

```
incode_starter_setup' ) ) ;
```



Program Studi Teknologi Informasi  
Fakultas Teknik  
Universitas Tidar



## **KATA PENGANTAR**

Puji syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa, karena atas rahmat dan karunia-Nya, modul praktikum Pemrograman Web ini dapat disusun. Modul ini disusun untuk menjadi pedoman mahasiswa mempelajari dasar-dasar pemrograman web meliputi HTML, CSS, JavaScript, PHP, dan MySQL. Dengan modul ini diharapkan mahasiswa mampu mengembangkan kemampuan dalam membangun aplikasi berbasis web yang dinamis, interaktif, dan sesuai standar industri.

Perkembangan teknologi web terus mengalami evolusi dari waktu ke waktu. Jika dulu HTML menjadi dasar utama pembangunan sebuah halaman web, maka kini dikembangkan berbagai teknologi pendukung guna meningkatkan fungsionalitas dan efisiensi dalam pengembangan aplikasi berbasis web. CSS hadir untuk mengatur tata letak dan desain, JavaScript memberikan interaktivitas, sedangkan PHP dan MySQL memungkinkan pengolahan data secara dinamis di sisi server. Oleh karena itu, pemahaman terhadap sejarah perkembangan pemrograman web, arsitektur yang mendasarinya, serta praktik terbaik dalam pengembangannya menjadi sangat penting bagi mahasiswa agar memiliki wawasan yang lebih luas dalam bidang ini.

Modul ini dibangun secara sistematis mulai dari pembahasan materi dasar hingga tingkatan lanjutan dengan metode pembelajaran berbasis praktik. Di setiap materi modul ini, mulai dari teorinya, langkah-langkah praktikum, hingga studi kasus dirancang guna memberikan kemudahan bagi mahasiswa dalam menguasai teknik pengembangan aplikasi web secara bertahap. Modul ini juga membahas tentang konsep pengelolaan database dan manajemen sesi dan keamanan pada pemrograman web dalam membangun sebuah aplikasi berbasis web secara modern. Harapan kami dengan modul ini, semoga berguna untuk menjadi referensi bagi teman-teman mahasiswa guna lebih memahami dan menguasai segala bentuk permasalahan mengenai pemrograman web secara teoritis. Oleh sebab itu kritik dan saran sangat dinanti demi menyempurnakan modul kami.

Terakhir, tak lupa kami mengucapkan terima kasih atas semua bantuan dan dukungan selama penyusunan modul ini. Semoga modul dapat memberikan sumbangsih yang lebih besar dan memberi bekal untuk mahasiswa terhadap pembangunan skill dalam Teknologi Informasi.

Magelang, 1 Januari 2025

Ikhwan Alfath Nurul Fathony

Dosen Teknologi Informasi, Universitas Tidar

# **SEJARAH PERKEMBANGAN PEMROGRAMAN WEB**

Perkembangan pemrograman web telah mengalami evolusi pesat sejak awal diperkenalkannya teknologi internet. Dari halaman web statis yang sederhana hingga aplikasi web dinamis dan interaktif yang kita kenal saat ini, setiap era membawa perubahan signifikan dalam cara situs web dibuat dan diakses.

## **1. Era Web Statis (1990-an)**

Pada tahun 1989, Tim Berners-Lee menciptakan World Wide Web (WWW) di CERN. Ia juga mengembangkan HTML atau HyperText Markup Language sebagai standar untuk membuat halaman-halaman web pertamanya. Di era ini, internet masih bersifat statis, terdiri hanya dari konten teks dan gambar sederhana dengan pengguna tak dapat berinteraksi secara lebih kompleks.

- 1991: Situs web pertama di dunia dibuat oleh Tim Berners-Lee dalam bentuk format sederhana HTML.
- 1993: Mosaic, salah satu browser web pertama, dirilis, memungkinkan lebih banyak orang mengakses web.
- 1995: HTML 2.0 introduced as the first standard for HTML.

## **2. Dinamis Web Era (Late 1990s – Early 2000s)**

Dengan meningkatnya kebutuhan interaksi pengguna, teknologi pemrograman web mulai berkembang untuk memungkinkan pembuatan situs yang lebih dinamis.

- 1995: JavaScript diperkenalkan oleh Netscape untuk memberikan interaktivitas pada halaman web tanpa harus melakukan permintaan ke server.
- 1995: PHP (Personal Home Page, kini dikenal sebagai PHP: Hypertext Preprocessor) diciptakan sebagai bahasa server-side yang memungkinkan pengolahan data dinamis.
- 1995: In MySQL, the relational database management system was developed to store and manage data in web applications.
- 1996: CSS, or Cascading Style Sheets, came into being and allowed the styling and layout of web pages to be separated from HTML.

During this time, the combination of HTML, CSS, JavaScript, PHP, and MySQL began to be used extensively for building more interactive web applications.

### 3. Era Web 2.0 (2000s – Early 2010s)

Konsep Web 2.0 mulai berkembang, ditandai dengan meningkatnya penggunaan teknologi yang memungkinkan pengguna berkontribusi dalam menciptakan konten.

- 2003: WordPress diluncurkan sebagai platform CMS yang memungkinkan pembuatan situs web tanpa perlu banyak pengetahuan tentang pemrograman.
- 2004: Ajax menjadi populer, memungkinkan komunikasi dengan server tanpa harus me-reload halaman web secara keseluruhan.
- 2005: YouTube dan jejaring sosial seperti Facebook mulai tumbuh dengan sangat cepat, mengandalkan teknologi web untuk interaksi pengguna yang lebih kompleks.

Saat itu, website tidak lagi sekadar informasi satu arah, tetapi menjadi platform interaktif dengan fitur media sosial, blog, dan e-commerce yang semakin berkembang.

### 4. Modern Web Era (2010s – present)

Developments on the Web go very fast with various technologies to support comfort in developing applications and users' experience.

- 2010: HTML5 and CSS3 came into view and introduced non-textual content without installing any plug-ins.
- 2011: Node.js was born as it began evolving. Now, JavaScript wasn't just on the client-side but was working on the server side too.
- 2013: Framework modern seperti React.js, Angular, dan Vue.js mulai diperkenalkan, membuat pengembangan aplikasi web lebih cepat dan modular.
- 2015: REST API dan GraphQL semakin banyak digunakan untuk komunikasi antara frontend dan backend aplikasi web.
- Sekarang: Teknologi seperti Progressive Web Apps (PWA), WebAssembly, dan AI-powered applications terus dikembangkan untuk memberikan pengalaman web yang lebih cepat dan responsif.

Dari dokumen statis sampai ke aplikasi interaktif dan kompleks, perkembangan pemrograman web melalui tahap-tahap yang berbeda. Maka dengan menggunakan teknologi seperti HTML, CSS, JavaScript, PHP, dan MySQL, pengembangan web semakin fleksibel dan dinamis. Sejarah ini memberikan kesempatan bagi pengembang untuk memahami prinsip-prinsip dasar dan mengikuti tren terbaru dalam dunia pemrograman web.

# FLOWCHART

Apa Itu Flowchart?

Flowchart adalah diagram visual yang menggambarkan alur logika atau proses dalam suatu sistem menggunakan simbol-simbol standar. Flowchart sering digunakan dalam pemrograman, bisnis, dan sistem engineering untuk mendesain dan menganalisis suatu proses.

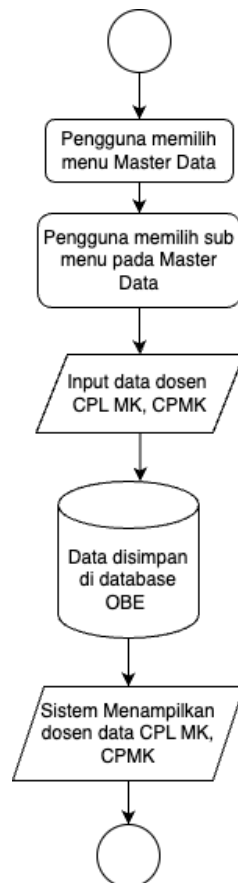
Fungsi Flowchart:

- ✓ Memvisualisasikan proses agar lebih mudah dipahami.
- ✓ Membantu perancangan algoritma sebelum menulis kode program.
- ✓ Mempermudah debugging dan menemukan kesalahan dalam logika sistem.
- ✓ Meningkatkan efisiensi kerja dengan mendokumentasikan alur proses secara sistematis.

Berikut beberapa simbol utama dalam flowchart:

	<b>Flow</b> Simbol yang digunakan untuk menggabungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga dengan Connecting Line.		<b>Input/output</b> Simbol yang menyatakan proses input atau output tanpa tergantung peralatan.
	<b>On-Page Reference</b> Simbol untuk keluar - masuk atau penyambungan proses dalam lembar kerja yang sama.		<b>Manual Operation</b> Simbol yang menyatakan suatu proses yang tidak dilakukan oleh komputer.
	<b>Off-Page Reference</b> Simbol untuk keluar - masuk atau penyambungan proses dalam lembar kerja yang berbeda.		<b>Document</b> Simbol yang menyatakan bahwa input berasal dari dokumen dalam bentuk fisik, atau output yang perlu dicetak.
	<b>Terminator</b> Simbol yang menyatakan awal atau akhir suatu program.		<b>Predefine Proses</b> Simbol untuk pelaksanaan suatu bagian (sub-program) atau prosedur.
	<b>Process</b> Simbol yang menyatakan suatu proses yang dilakukan komputer.		<b>Display</b> Simbol yang menyatakan peralatan output yang digunakan.
	<b>Decision</b> Simbol yang menunjukan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, yaitu ya dan tidak.		<b>Preparation</b> Simbol yang menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberikan nilai awal.

Berikut ini merupakan contoh sebuah flowchart untuk sistem informasi assessment berbasis Outcome Based Education (OBE). Flowchart ini menggambarkan proses pengelolaan data dosen terkait CPL (Capaian Pembelajaran Lulusan) dan CPMK (Capaian Pembelajaran Mata Kuliah) dalam sistem berbasis OBE (Outcome-Based Education). Berikut adalah alur ceritanya:



Gambar Flowchart

1. Pengguna memilih menu Master Data

- Pengguna (dosen/admin) mengakses sistem dan masuk ke menu Master Data, yang berisi berbagai pengelolaan data terkait pembelajaran.

2. Pengguna memilih sub-menu dalam Master Data

- Setelah masuk ke menu utama, pengguna memilih sub-menu yang relevan, dalam hal ini adalah pengelolaan data CPL MK dan CPMK.

### 3. Input data dosen untuk CPL MK dan CPMK

- Pengguna memasukkan data dosen yang berkaitan dengan CPL MK dan CPMK melalui form input di sistem.
- Data ini bisa berupa nama dosen, mata kuliah yang diajarkan, serta capaian pembelajaran yang terkait dengan mata kuliah tersebut.

### 4. Data disimpan dalam database OBE

- Setelah data dimasukkan, sistem akan menyimpannya ke dalam database OBE untuk memastikan data tersimpan dengan aman dan dapat diakses kapan saja.

### 5. Sistem menampilkan data dosen terkait CPL MK dan CPMK

- Setelah data tersimpan, sistem akan menampilkan data tersebut kepada pengguna, baik dalam bentuk tabel, laporan, atau dashboard.
- Pengguna dapat melihat data yang telah dimasukkan dan memastikan bahwa informasi yang tersimpan sudah benar.

### ← END Proses selesai

Dengan selesainya langkah ini, sistem telah berhasil mengelola data CPL MK dan CPMK secara otomatis. Flowchart ini menggambarkan alur input dan pengolahan data dosen dalam sistem OBE. Dengan sistem ini, dosen dapat memasukkan, menyimpan, dan melihat data CPL MK dan CPMK dengan lebih efisien, sehingga pengelolaan capaian pembelajaran dalam kurikulum menjadi lebih terstruktur dan terdokumentasi dengan baik. 🚀

## **TUGAS - Membuat Alur Cerita dan Flowchart Sistem**

### **Instruksi:**

Buatlah alur cerita dan flowchart untuk membangun sebuah sistem berbasis web atau aplikasi yang dapat membantu menyelesaikan suatu permasalahan. Sistem yang dibuat harus memiliki alur proses yang jelas dan dapat diterapkan dalam dunia nyata.

### **Ketentuan Tugas:**

1. Pilih jenis sistem yang akan dibuat, misalnya:

- Sistem pemesanan makanan online
- Sistem absensi berbasis QR code
- Sistem manajemen tugas mahasiswa
- Sistem e-commerce sederhana
- Sistem perpustakaan digital

Boleh menggunakan ide tema, selain contoh di atas. 😊

2. Buat alur cerita (naskah naratif) yang menjelaskan bagaimana sistem bekerja dari awal hingga akhir. Alur cerita harus mencakup:

- Siapa pengguna sistem (admin, pelanggan, mahasiswa, dsb.).
- Bagaimana pengguna berinteraksi dengan sistem (input, proses, dan output).
- Hasil akhir dari sistem yang dibuat.

3. Buat flowchart yang menggambarkan proses utama dalam sistem, dengan menggunakan simbol standar:

- Terminator (Start/End)
- Process (Proses yang terjadi dalam sistem)
- Input/Output (Masukan dari pengguna & keluaran sistem)
- Decision (Keputusan bercabang dalam sistem)
- Arrow (Alur proses dalam sistem)

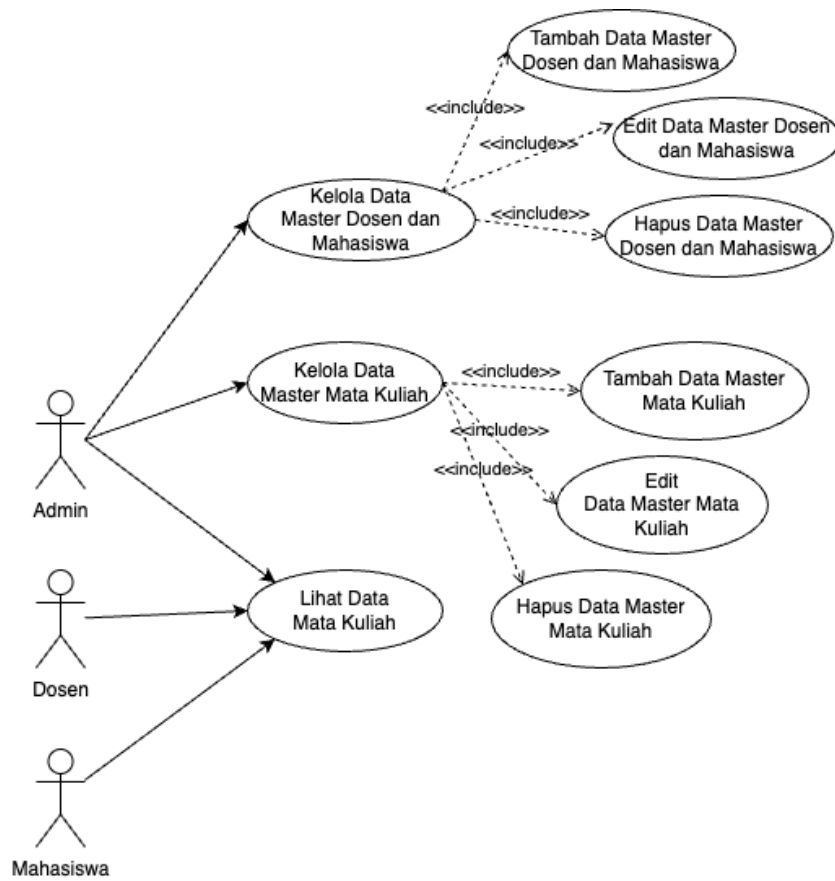


4. Gunakan tools digital atau gambar tangan untuk menggambar flowchart, kemudian sertakan dalam laporan tugas.

Hasil yang dikumpulkan:

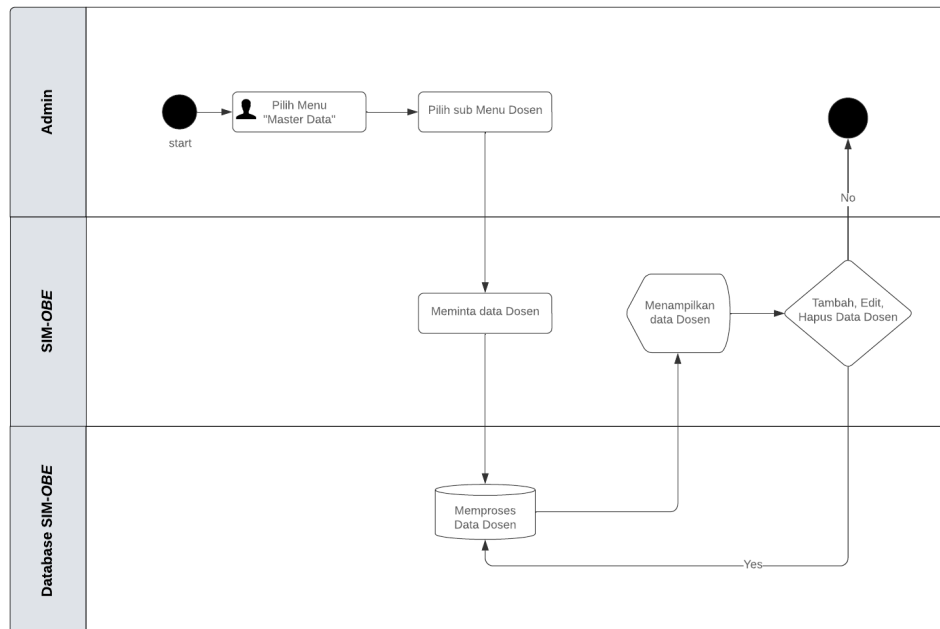
- ✓ Dokumen berisi alur cerita sistem dalam bentuk narasi.
- ✓ Flowchart yang jelas dan rapi, baik dibuat secara digital dengan tools **<https://draw.io>**
- ✓ File dikumpulkan dalam bentuk **link <https://medium.com/>**.

## DIAGRAM USE CASE



## DIAGRAM SWIMLANE

### Fungsionalitas 1 Tambah, Edit, Hapus data Dosen



#### Deskripsi

*Gambar Error! No text of specified style in document..1 Diagram Swimlane Tambah, Edit, Hapus Data Dosen*

Menambah, mengedit, menghapus data Dosen oleh Admin

Urutan Langkah

Langkah 1 Admin memilih sub menu Dosen

Langkah 2 sistem meminta data Dosen kepada database

Langkah 3 Database mencari data Dosen

Langkah 4 Sistem menampilkan data Dosen

Langkah 5 Admin memilih menu tambah data Dosen

Langkah 6 Sistem menampilkan form tambah data Dosen

Langkah 7 Admin mengisi data pada form tambah Dosen

Langkah 8 Admin menekan tombol simpan data Dosen

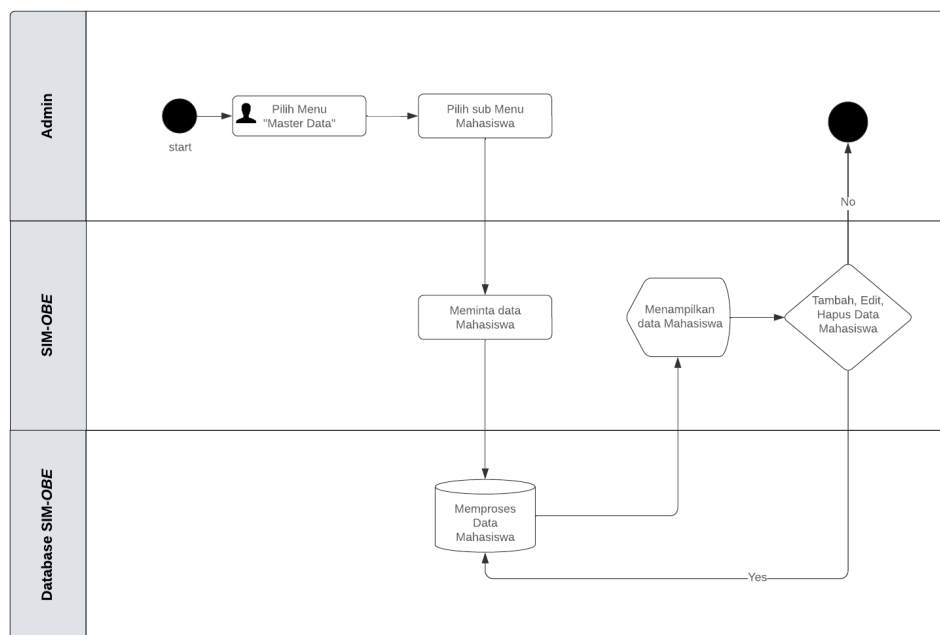
Langkah 9 Sistem menyimpan data Dosen ke dalam database

Langkah 10 Database menyimpan data Dosen

Kebutuhan Sistem

R01.1 Sistem harus bisa menambahkan, mengedit, menghapus data pada databases

## Fungsionalitas 2 Tambah, Edit, Hapus data Mahasiswa



Gambar *Error! No text of specified style in document.*2 Diagram Swimlane Tambah, Edit, Hapus Data Mahasiswa

### Deskripsi

Menambah, mengedit, menghapus data Mahasiswa oleh Admin

Urutan Langkah

Langkah 1 Admin memilih sub menu Mahasiswa

Langkah 2 sistem meminta data Mahasiswa kepada database

Langkah 3 Database mencari data Mahasiswa

Langkah 4 Sistem menampilkan data Mahasiswa

Langkah 5 Admin memilih menu tambah data Mahasiswa

Langkah 6 Sistem menampilkan form tambah data Mahasiswa

Langkah 7 Admin mengisi data pada form tambah Mahasiswa

Langkah 8 Admin menekan tombol simpan data Mahasiswa

Langkah 9 Sistem menyimpan data Mahasiswa ke dalam database

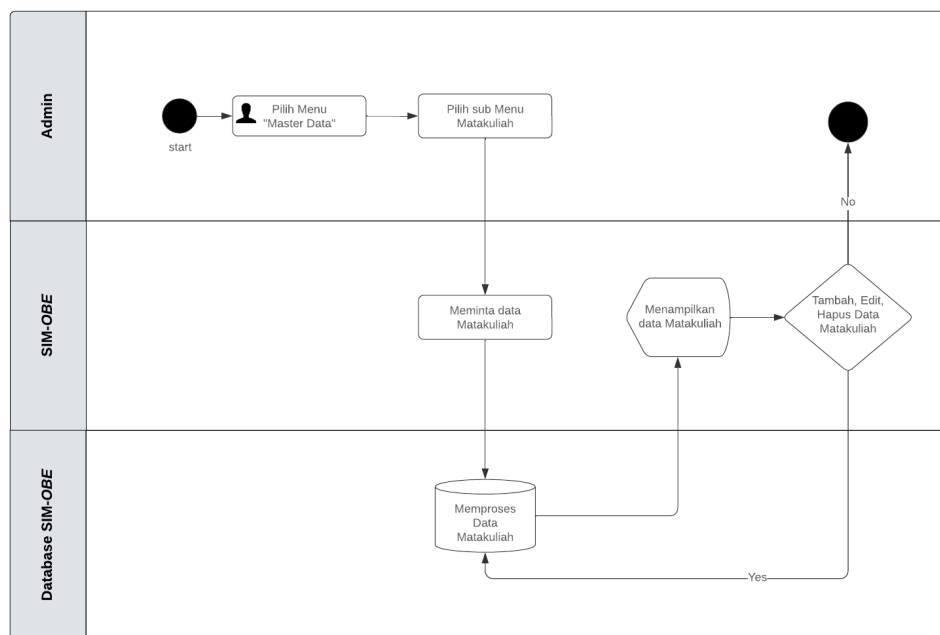
Langkah 10 Database menyimpan data Mahasiswa

Kebutuhan Sistem

R01.2 Sistem harus bisa menambahkan, mengedit, menghapus data pada

databases

### Fungsionalitas 3 Tambah, Edit, Hapus data Matakuliah



Gambar Error! No text of specified style in document..3 Diagram Swimlane Tambah, Edit, Hapus Data Matakuliah

#### Deskripsi

Menambah, mengedit, menghapus data Matakuliah oleh Admin

Urutan Langkah

Langkah 1 Admin memilih sub menu Matakuliah

Langkah 2 sistem meminta data Matakuliah kepada database

Langkah 3 Database mencari data Matakuliah

Langkah 4 Sistem menampilkan data Matakuliah

Langkah 5 Admin memilih menu tambah data Matakuliah

Langkah 6 Sistem menampilkan form tambah, edit, atau hapus data

Matakuliah

Langkah 7 Admin mengisi data pada form tambah, edit, atau hapus

Matakuliah

Langkah 8 Admin menekan tombol simpan data Matakuliah

Langkah 9 Sistem menyimpan data Matakuliah ke dalam database

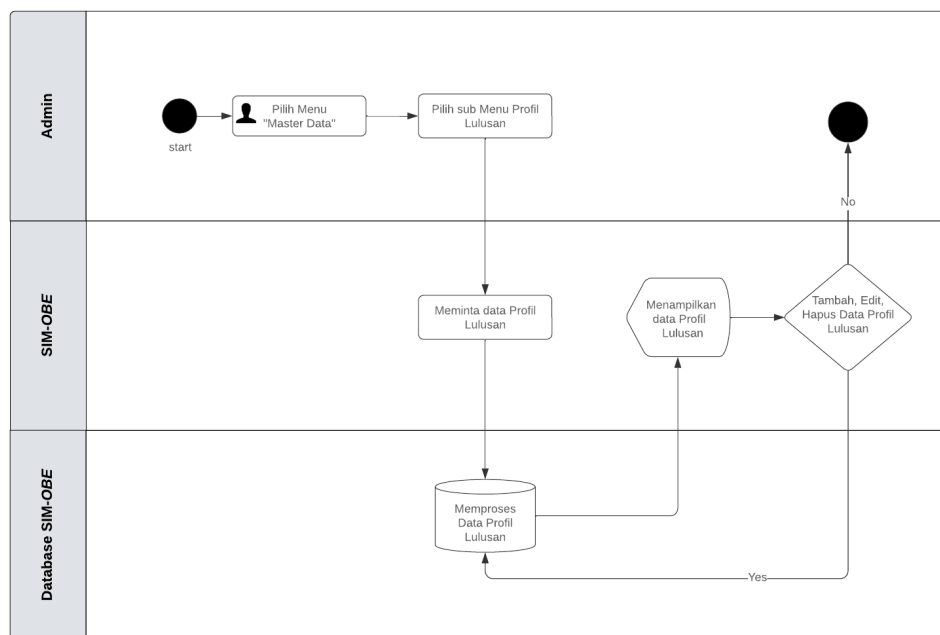
Langkah 10 Database menyimpan data Matakuliah

#### Kebutuhan Sistem

R01.3 Sistem harus bisa menambahkan, mengedit, menghapus data pada

databases

## Fungsionalitas 4 Tambah, Edit, Hapus data Profil Lulusan



Gambar *Error! No text of specified style in document.* 4 Diagram Swimlane Tambah, Edit, Hapus Data Profil Lulusan

### Deskripsi

Menambah, mengedit, menghapus data Profil Lulusan oleh Admin

### Urutan Langkah

Langkah 1 Admin memilih sub menu Profil Lulusan

Langkah 2 sistem meminta data Profil Lulusan kepada database

Langkah 3 Database mencari data Profil Lulusan

Langkah 4 Sistem menampilkan data Profil Lulusan

Langkah 5 Admin memilih menu tambah data Profil Lulusan

Langkah 6 Sistem menampilkan form tambah, edit, atau hapus data Profil Lulusan

Langkah 7 Admin mengisi data pada form tambah, edit, atau hapus Profil Lulusan

Langkah 8 Admin menekan tombol simpan data Profil Lulusan

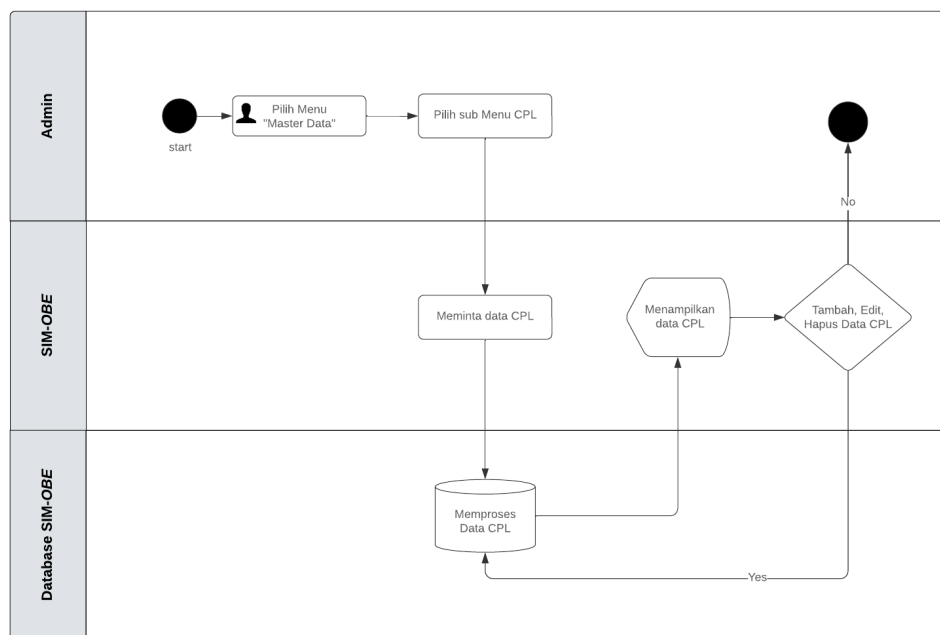
Langkah 9 Sistem menyimpan data Profil Lulusan ke dalam database

Langkah 10 Database menyimpan data Profil Lulusan

### Kebutuhan Sistem

R01.4 Sistem harus bisa menambahkan, mengedit, menghapus data pada databases

## Fungsionalitas 5 Tambah, Edit, Hapus data CPL



### Deskripsi

*Gambar Error! No text of specified style in document..5 Diagram Swimlane Tambah, Edit, Hapus Data CPL*

Menambah, mengedit, menghapus data CPL oleh Admin

Urutan Langkah

Langkah 1 Admin memilih sub menu CPL

Langkah 2 sistem meminta data CPL kepada database

Langkah 3 Database mencari data CPL

Langkah 4 Sistem menampilkan data CPL

Langkah 5 Admin memilih menu tambah data CPL

Langkah 6 Sistem menampilkan form tambah, edit, atau hapus data CPL

Langkah 7 Admin mengisi data pada form tambah, edit, atau hapus CPL

Langkah 8 Admin menekan tombol simpan data CPL

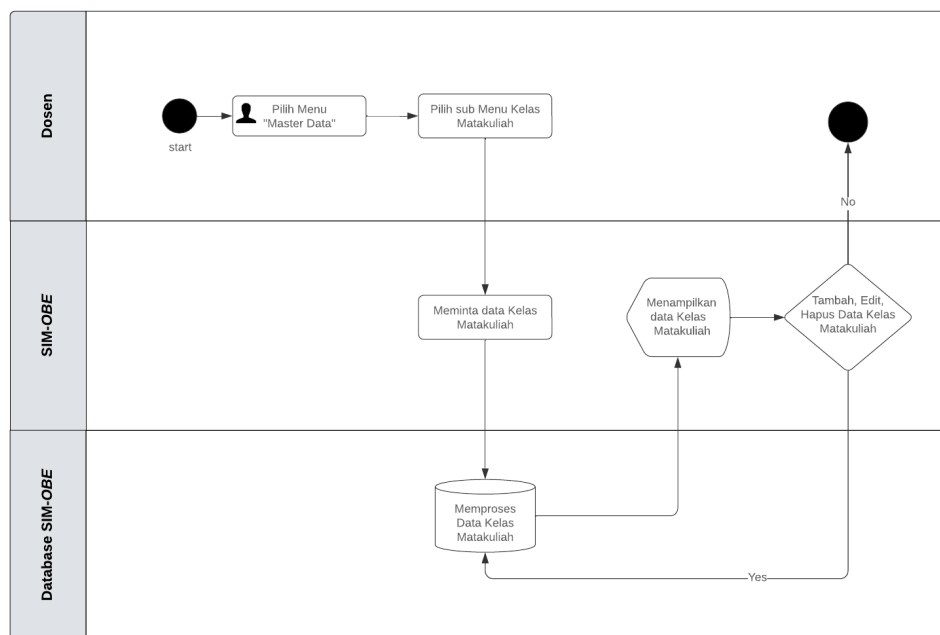
Langkah 9 Sistem menyimpan data CPL ke dalam database

Langkah 10 Database menyimpan data CPL

Kebutuhan Sistem

R01.5 Sistem harus bisa menambahkan, mengedit, menghapus data pada databases

## Fungsionalitas 6 Tambah, Edit, Hapus data Kelas Matakuliah



### Deskripsi

*Gambar Error! No text of specified style in document..6 Diagram Swimlane Tambah, Edit, Hapus Data Kelas Matakuliah*

Menambah, mengedit, menghapus data Kelas Matakuliah oleh Dosen

Urutan Langkah

Langkah 1 Dosen memilih sub menu Kelas Matakuliah

Langkah 2 sistem meminta data Kelas Matakuliah kepada database

Langkah 3 Database mencari data Kelas Matakuliah

Langkah 4 Sistem menampilkan data Kelas Matakuliah

Langkah 5 Dosen memilih menu tambah data Kelas Matakuliah

Langkah 6 Sistem menampilkan form tambah, edit, atau hapus data Kelas Matakuliah

Langkah 7 Dosen mengisi data pada form tambah, edit, atau hapus Kelas Matakuliah

Langkah 8 Dosen menekan tombol simpan data Kelas Matakuliah

Langkah 9 Sistem menyimpan data Kelas Matakuliah ke dalam database

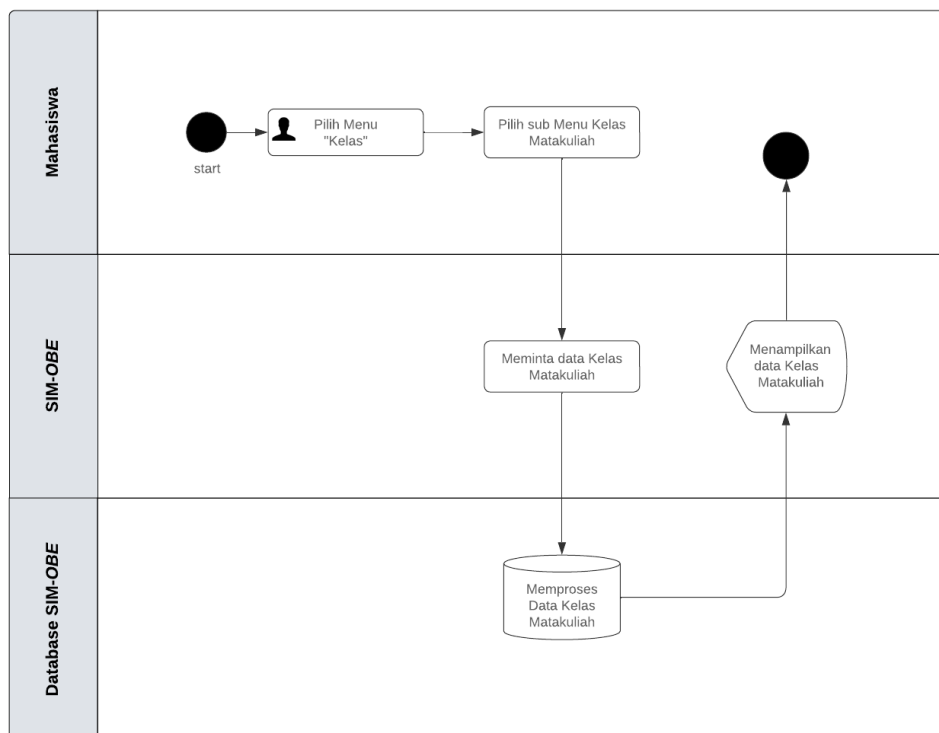
Langkah 10 Database menyimpan data Kelas Matakuliah

Kebutuhan Sistem

R01.6 Sistem harus bisa menambahkan, mengedit, menghapus data pada databases



## Fungsionalitas 7 lihat data Kelas Matakuliah



### Deskripsi

*Gambar Error! No text of specified style in document..7 Diagram Swimlane Tampilkan Data Kelas Matakuliah*

### Melihat data Kelas Matakuliah oleh Mahasiswa

#### Urutan Langkah

- Langkah 1 Mahasiswa memilih sub menu Kelas Matakuliah
- Langkah 2 sistem meminta data Kelas Matakuliah kepada database
- Langkah 3 Database mencari data Kelas Matakuliah
- Langkah 4 Sistem menampilkan data Kelas Matakuliah
- Langkah 5 Mahasiswa memilih menu tambah data Kelas Matakuliah

#### Kebutuhan Sistem

R01.7 Sistem harus bisa menampilkan data kelas matakuliah yang ada di databases

## PENGANTAR DESAIN UI DALAM PEMROGRAMAN WEB

Dalam pengembangan aplikasi web, **Desain User Interface (UI)** memegang peranan penting dalam menciptakan pengalaman pengguna yang optimal. UI tidak hanya berfokus pada tampilan visual, tetapi juga pada bagaimana pengguna berinteraksi dengan sistem. Desain UI yang baik harus memperhatikan aspek estetika, fungsionalitas, serta kemudahan penggunaan agar pengguna dapat menavigasi dan menggunakan aplikasi dengan nyaman dan efisien.

Seiring dengan berkembangnya teknologi, standar dalam desain UI juga terus mengalami perubahan. Penggunaan tata letak yang responsif, kombinasi warna yang harmonis, pemilihan tipografi yang tepat, serta elemen-elemen interaktif seperti tombol, formulir, dan navigasi menjadi faktor utama dalam menciptakan antarmuka yang menarik dan user-friendly. Selain itu, desain UI juga harus mempertimbangkan aspek **aksesibilitas**, sehingga dapat digunakan oleh berbagai kelompok pengguna dengan kebutuhan yang beragam.

Dalam proses pengembangan aplikasi web, desain UI biasanya dilakukan sebelum tahap pengkodean (development). Desain ini dibuat dalam bentuk **prototipe** atau **wireframe**, yang berfungsi sebagai gambaran awal struktur halaman dan interaksi yang akan diterapkan dalam aplikasi. Dengan adanya prototipe ini, tim pengembang dapat menguji dan menyempurnakan desain sebelum memasuki tahap implementasi, sehingga menghemat waktu dan biaya pengembangan.

Saat ini, terdapat berbagai alat berbasis cloud yang dapat digunakan untuk mendesain UI secara kolaboratif dan efisien, antara lain:

✅ **Figma** → Alat desain UI/UX berbasis cloud dengan fitur kolaborasi real-time yang memungkinkan tim bekerja secara bersama-sama.

🔗 <https://www.figma.com/>

✅ **Adobe XD** → Alat desain UI dengan fitur prototyping yang kuat untuk membuat desain interaktif dan mendukung animasi.

🔗 <https://www.adobe.com/products/xd.html>

✅ **Sketch** → Alat desain UI populer untuk macOS yang banyak digunakan dalam pembuatan wireframe dan prototipe.

🔗 <https://www.sketch.com/>

✅ **Canva** → Cocok untuk pembuatan desain UI sederhana, terutama bagi pemula atau yang tidak memiliki latar belakang desain.

🔗 <https://www.canva.com/>

Dengan memahami prinsip dasar dan menggunakan alat yang tepat dalam perancangan UI, pengembang dapat menciptakan aplikasi web yang tidak hanya menarik secara visual, tetapi juga memberikan pengalaman yang lebih baik bagi pengguna. Desain UI yang baik akan meningkatkan kepuasan pengguna, mempercepat adopsi aplikasi, serta mendukung keberhasilan proyek pengembangan web secara keseluruhan. 🚀

# DASAR – DASAR HTML

## Dasar Teori

**HTML** merupakan singkatan dari **HyperText Markup Language**, yaitu sebuah *script* yang digunakan untuk menyusun dokumen web. Dokumen HTML disimpan dalam bentuk teks biasa dan berisi tag-tag khusus yang memberi instruksi kepada peramban web (browser) untuk menjalankan perintah tertentu.

### 1. Penamaan Dokumen HTML

Dalam menyimpan suatu dokumen HTML dengan menggunakan komputer, penamaan filenya diakhiri dengan tanda .html. Misalnya suatu dokumen HTML dapat disimpan dengan nama namafile.html.

### 2. Struktur Dokumen HTML

Struktur dasar sebuah dokumen HTML terdiri dari tiga tag utama, yaitu HTML, HEAD, dan BODY.

- Tag <HTML> digunakan untuk mendefinisikan bahwa suatu dokumen adalah dokumen HTML.
- Tag <HEAD> berisi informasi metadata mengenai dokumen tersebut.
- Tag <BODY> berisi konten utama yang akan ditampilkan pada halaman web.

Dalam sebuah dokumen HTML terdapat dua bagian utama, yaitu Head dan Body.

#### a. Bagian Head

Bagian Head ditulis dengan menggunakan tag <HEAD>.</HEAD>. Di sini terdapat beberapa elemen, di antaranya adalah <TITLE>.</TITLE>, yang bisa menampilkan title dari suatu halaman di tab browser.

#### b. Bagian Body

Bagian ini dituliskan dengan menggunakan tag <BODY>.</BODY>. Bagian ini digunakan untuk menampilkan berbagai jenis elemen, seperti teks, gambar, link, dan elemen-elemen yang dapat dilihat oleh user.

## Contoh Struktur Dasar HTML

Berikut merupakan contoh secara sederhana dari struktur dokumen pada HTML:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Judul Halaman</title>
```

```
</head>
```

```
<body>
```

```
    <h1>Selamat Datang di Website</h1>
```

```
    <p>Ini adalah contoh dokumen HTML sederhana.</p>
```

```
</body>
```

```
</html>
```

Bagian Head

Bagian Body

Penjelasan :

*<! Doctype html>*

Merupakan tag awal dari setiap dokumen HTML, tag ini berfungsi untuk menginformasikan pada browser bahwa dokumen yang sedang dibuka adalah dokumen HTML.

*<html>.....</html>*

Awal dari isi dokumen HTML, semua kode HTML yang dibuat akan ditulis dalam tag ini.

*head>*

Tag head digunakan untuk menyimpan informasi tentang dokumen HTML.

*<title>.....</title>*

Salah satu contoh informasi yang terdapat didalam tag HEAD adalah title. Title akan menampilkan teks pada judul browser atau pada tab browser.

*<body>.....</body>*

Merupakan tag pembuka dari badan dokumen HTML.

### 3. Elemen dan Tag Pada Dokumen HTML

Dokumen HTML memiliki ciri khas utama berupa **tag dan elemen**. Elemen dalam HTML terbagi menjadi dua bagian utama:

- **Elemen <HEAD>**, yang berfungsi untuk memberikan informasi mengenai dokumen HTML, seperti metadata, judul halaman, dan referensi ke file eksternal.

- **Elemen <BODY>**, yang menentukan tampilan isi dokumen pada browser, seperti paragraf, daftar, tabel, dan elemen visual lainnya.

Sementara itu, **tag dalam HTML** ditandai dengan simbol **kurung siku**:

- **Tag pembuka** menggunakan tanda lebih kecil < (misalnya <p>).
- **Tag penutup** menggunakan tanda lebih besar > dan diawali dengan garis miring / (misalnya </p>).

Dalam penggunaannya, sebagian besar kode HTML harus berada di dalam **tag kontainer**, yang terdiri dari **tag pembuka** (<namatag>) dan **tag penutup** (</namatag>).

## Struktur Dokumen HTML

Sebuah dokumen HTML terdiri dari beberapa elemen yang saling berhubungan untuk membentuk struktur yang lengkap, di antaranya:

- **Elemen <html>** sebagai elemen utama yang menampung seluruh isi dokumen.
- **Elemen <head>** untuk informasi metadata.
- **Elemen <body>** untuk menampilkan konten yang terlihat di halaman web.
- **Elemen <h1> sampai <h6>** untuk mendefinisikan header 1 sampai 6
- **Elemen <p>** untuk mendefinisikan paragraph
- **Elemen <br>** untuk menambahkan line break (spasi baru)
- **Elemen <!-->** untuk mendefinisikan komentar

Setiap elemen dalam HTML diapit oleh **tanda pembatas** (< . . . >), yang dikenal sebagai **tag**.

- **Heading dan Paragraph**

Heading digunakan untuk memberikan penjudulan pada suatu dokumen HTML. Untuk melakukan format heading maka menggunakan tag <h1> . . . </h1> untuk judul utama dan untuk judul sub bab dapat menggunakan tag <h2> . . . </h2> sampai dengan <h6> . . . </h6>.

Untuk membuat sebuah paragraph dalam dokumen HTML maka digunakan tag <p> . . . </p>.

- **Link**

Link atau biasa disebut dengan anchor (pengait) dapat dibuat dengan menambahkan tag

`<a>...</a>` serta atribut *href* agar link tersebut dapat menyimpan alamat web yang akan dituju ketika link diklik. Sehingga sintaks link adalah :

```
<a href='https://alamat-web-yang-dituju.com'></a>
```

- **List**

Dalam dokumen HTML suatu daftar atau list adalah suatu kumpulan item tertentu, daftar item dapat ditulis secara berurutan atau tidak. Jenis list dalam HTML adalah :

- a. *Ordered list*

*Ordered list* dapat dibuat dengan menggunakan tag `<ol>...</ol>` dan untuk setiap listnya menggunakan tag `<li>...</li>`.

- b. *Unordered list*

*Unordered list* dapat dibuat dengan menggunakan tag `<ul>...</ul>` dan untuk setiap listnya menggunakan tag `<li>...</li>`.

- **Image**

Dalam dokumen HTML kita dapat memuat dan menampilkan gambar. Untuk memuat gambar menggunakan tag `<img>` untuk menentukan file gambar maka digunakan atribut *src*. Sehingga sintaks image adalah :

```
<img src ='nama-file-gambar.jpg'>
```

- **Form**

Sebuah form dalam dokumen HTML dapat memiliki bermacam kontrol, mulai dari text input, combo box, button dan lain sebagainya. Pembuatan sebuah form diawali dengan menggunakan tag :

```
<form>...</form>
```

Berikut beberapa kontrol form yang biasa digunakan dalam HTML:

- a. Label

Label digunakan untuk memberikan keterangan pada setiap input yang ada. Sintaks form label adalah :

```
<label for='nama'>keterangan input</label>
```

- b. Text

Kontrol input ini dapat diisi dengan teks yang memiliki panjang kata terbatas, biasa digunakan dalam form pencarian, nama dan lainnya. Sintaks form text adalah :

```
<input type='text' name='nama'>
```

- c. Combobox

Combobox adalah kontrol yang memiliki pilihan ketika diklik. Sintaks form combobox adalah :

```

<select>
  <option value="name value 1"></option>
  <option value="name value 2"></option>
  <option value="name value 3"></option>
</select>

```

## ▪ Tabel

Untuk menampilkan data dengan tipe tabel dalam HTML, dapat menggunakan tag :

```
<table>...</table>
```

Sedangkan untuk menampilkan baris tabel maka digunakan tag :

```
<tr>...</tr>
```

Sedangkan untuk menentukan banyaknya kolom maka menggunakan tag :

```
<td>...</td>
```

## ▪ Div

Penggunaan tag <div> adalah sebagai container atau pembatas satu struktur HTML dengan struktur lainnya. dalam struktur HTML terdapat beberapa bagian antara lain :

### a. Header

Bagian kepala website yang berisi logo, nama, slogan, menu dan lainnya. Sintaks

<div> untuk footer adalah :

```
<div id='header'>.....</body>
```

### b. Content

Berisi isi website itu sendiri, contoh content berisi postingan terbaru atau informasi lainnya. sintaks <div> untuk content adalah :

```
<div id='content'>.....</body>
```

### c. Sidebar

Berada disamping content dan biasanya berisi iklan, kategori artikel dan widget dan lainnya. Sintaks <div> untuk sidebar adalah :

```
<div id='sidebar'>.....</body>
```

### d. Footer

Bagian kaki dari website yang pada umumnya diisi dengan tag website tersebut, misalnya *copyright©2024 by me*. Sintaks <div> untuk footer adalah :

```
<div id='footer'>.....</body>
```



## KEGIATAN PRAKTIKUM

### Requirement Tools untuk Pemrograman Web

Dalam praktikum pemrograman web, Anda memerlukan beberapa tools utama untuk membantu dalam pengembangan dan pengujian kode. Berikut adalah daftar perangkat lunak yang direkomendasikan:

#### 1. XAMPP (Cross-Platform, Apache, MySQL, PHP, Perl)



Fungsi:

- Menyediakan server lokal untuk menjalankan skrip PHP dan mengelola database MySQL.
- Termasuk Apache sebagai web server dan phpMyAdmin sebagai antarmuka manajemen database.



Download:

-  <https://www.apachefriends.org/download.html>



Alternatif:



- WAMP (untuk Windows), MAMP (untuk macOS), atau Docker dengan container PHP & MySQL.

#### 2. Code Editor / IDE

Anda dapat menggunakan editor kode ringan atau IDE untuk menulis dan mengedit skrip HTML, CSS, JavaScript, dan PHP.



Pilihan Editor:

- **Sublime Text** → Ringan, cepat, dan mendukung banyak plugin.  
 <https://www.sublimetext.com/>
- **Visual Studio Code (VS Code)** → Gratis, fitur lengkap, dan mendukung ekstensi untuk PHP, MySQL, dan JavaScript.  
 <https://code.visualstudio.com/>



Alternatif:

- Notepad++ (untuk pengguna Windows yang menginginkan editor sederhana).
- PHPStorm (IDE premium dengan fitur lengkap untuk PHP & web development).

### 3. Sistem Operasi

- Windows (direkomendasikan untuk XAMPP dan Sublime Text/VS Code).
- Alternatif: Linux (Ubuntu) atau macOS juga dapat digunakan dengan tool yang sesuai seperti MAMP atau LAMP stack.

### 4. Web Browser

Digunakan untuk menguji tampilan dan fungsionalitas halaman web.

#### ✅ Rekomendasi Browser:

- **Google Chrome** (terbaik untuk debugging dengan DevTools).
- **Mozilla Firefox** (juga memiliki developer tools yang baik).
- **Microsoft Edge** (alternatif berbasis Chromium).

### 5. Version Control & Repository Management

Untuk menyimpan dan mengelola kode proyek secara online serta berkolaborasi dengan tim.

#### ✅ Git & GitLab

- Git → Digunakan untuk versi kontrol (Version Control System) pada kode.
- GitLab → Platform penyimpanan repository berbasis Git yang dapat digunakan untuk menyimpan, berbagi, dan berkolaborasi dalam proyek pemrograman web.

#### ✅ Download & Registrasi:

- **Git** → <https://git-scm.com/downloads>
- **GitLab** → <https://gitlab.com/>

#### ✅ Alternatif:

- GitHub → Alternatif lain untuk penyimpanan repository berbasis Git.
- Bitbucket → Platform penyimpanan kode yang mendukung Git dan Mercurial.

### 6. Ekstensi & Tools Tambahan

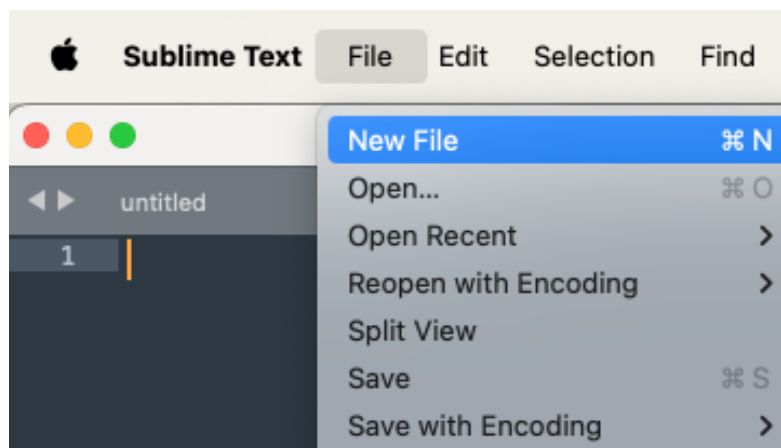
- **Live Server (untuk VS Code)** → Memungkinkan Anda melihat perubahan pada kode secara real-time di browser.
- **Postman** → Untuk menguji API jika proyek melibatkan komunikasi dengan backend.
- **Docker** → Untuk membuat lingkungan pengembangan yang lebih terisolasi dan mudah dikonfigurasi.

## LATIHAN

### 1. Membuat Heading dan Paragraph

Langkah-langkah praktikum

- Buka direktori C>xampp>htdocs kemudian buat folder dan beri nama pweb
- Buka editor Sublime Text
- Buatlah file baru dengan membuka file > new atau dengan *shortcut* ctrl + N



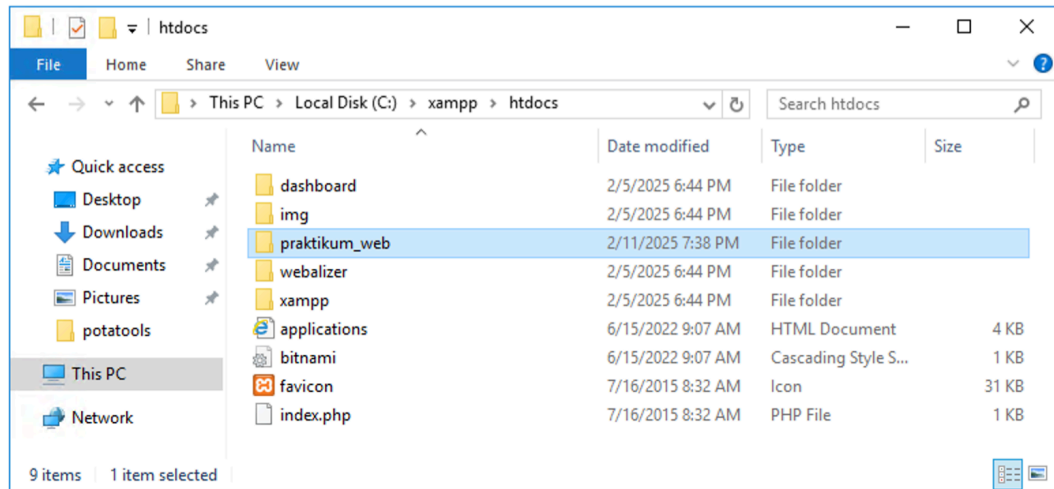
- Tuliskan *coding* berikut

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Belajar Membuat Heading dan Paragraph</title>
</head>
<body>
  <h1>Heading ke-1</h1>
  <h2>Heading ke-2</h2>
  <h3>Heading ke-3</h3>
  <h4>Heading ke-4</h4>
  <h5>Heading ke-5</h5>
  <h6>Heading ke-6</h6>

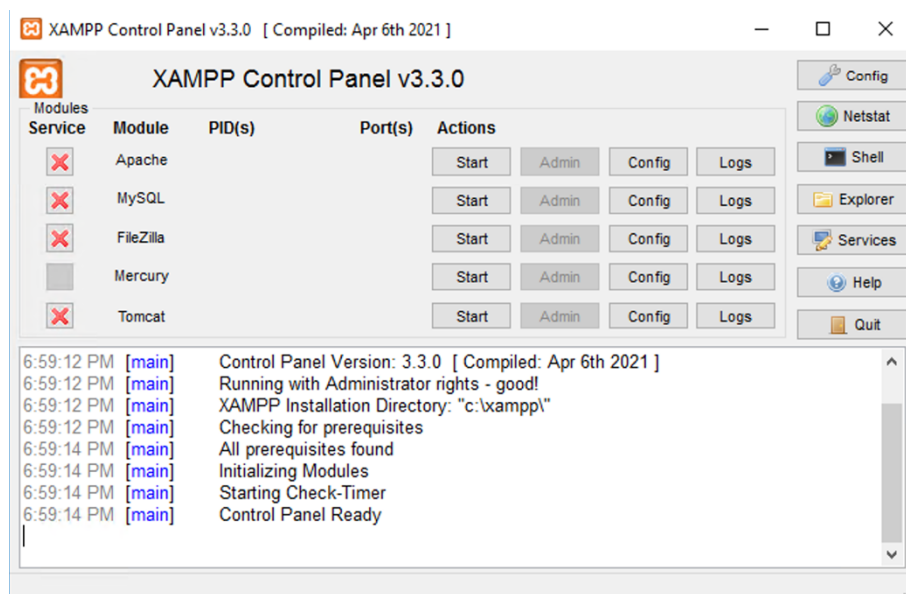
  <p> <strong>ini adalah contoh paragraph</strong></p>
</body>
</body>
</html>
```

- Simpan *coding* yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > praktikum\_web**, dan

beri nama file dengan “**prak1.html**” atau pilih “Hypertext Markup Language File” pada Save as type, klik *Save*.



- Pastikan di komputer sudah terinstall XAMPP. Kemudian buka *XAMPP* dan aktifkan *Apache* serta *MySQL*, dengan menekan tombol *Start* pada modul *Apache* dan *MySQL*.



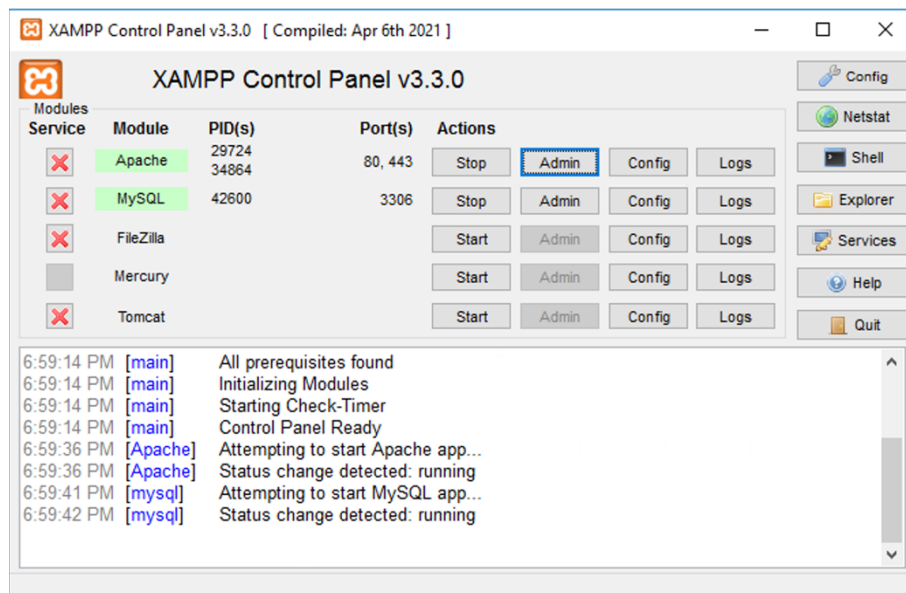
## Fungsi XAMPP

XAMPP adalah software yang menyediakan lingkungan server lokal untuk pengembangan web. Dengan XAMPP, kita bisa menjalankan aplikasi berbasis PHP dan MySQL di komputer sendiri tanpa perlu koneksi internet atau server eksternal.

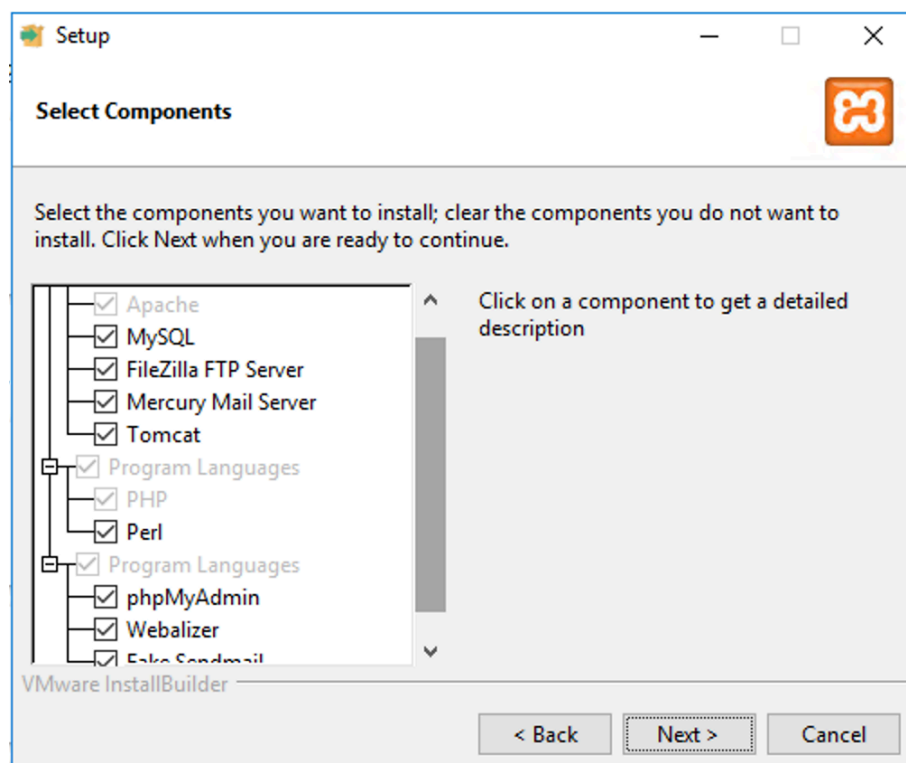
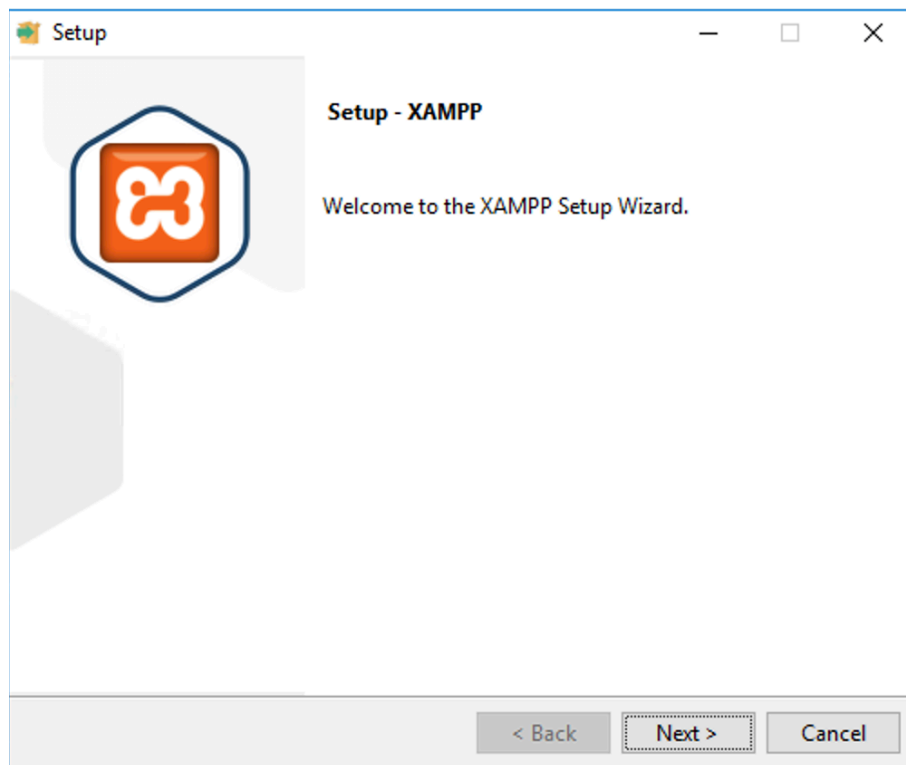
### Fungsi Utama XAMPP:

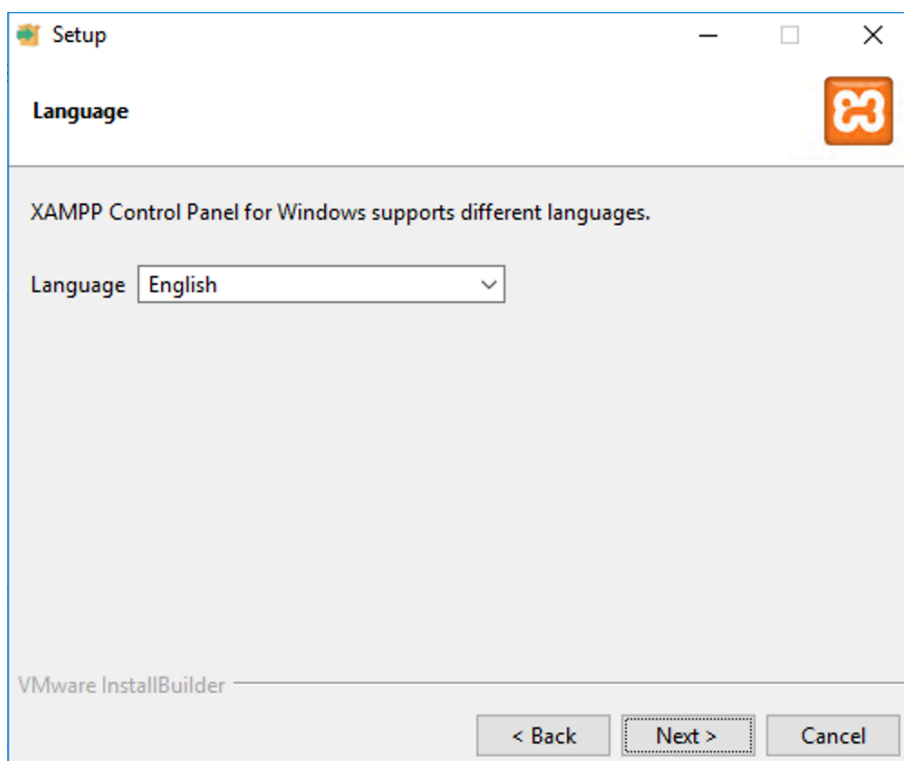
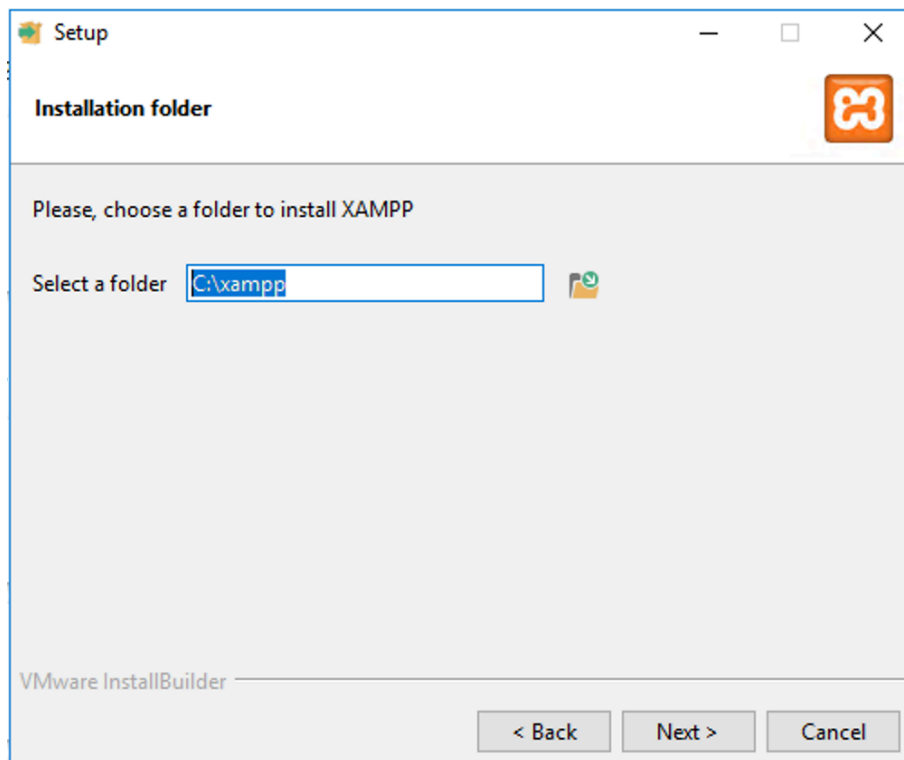
1. Server Lokal → Menjalankan aplikasi web di komputer sendiri tanpa perlu hosting online.
2. Mendukung PHP & MySQL → Memungkinkan pengembangan website berbasis PHP dan database MySQL.

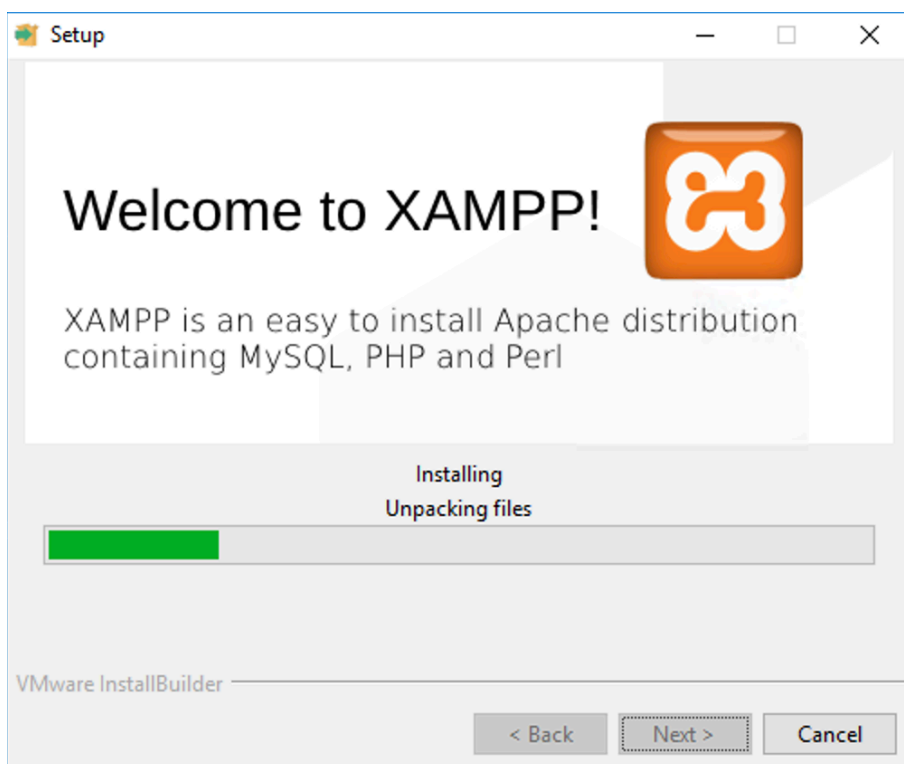
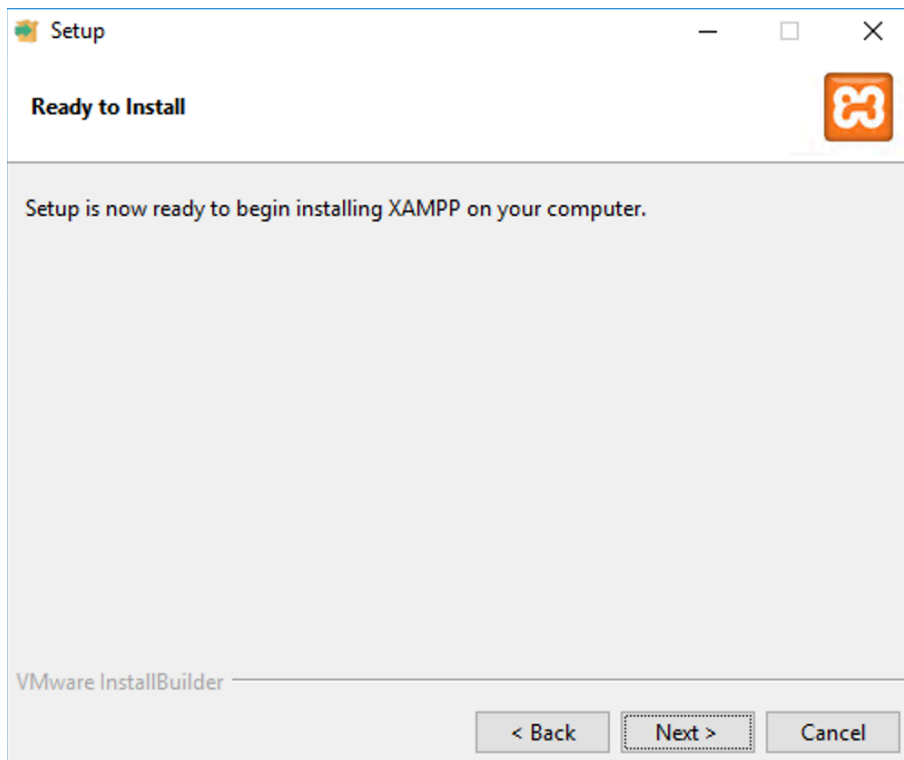
3. Menjalankan Apache Server → Mengaktifkan server web untuk mengakses website secara lokal.
4. Menyediakan phpMyAdmin → Memudahkan manajemen database MySQL melalui antarmuka web.
5. Testing dan Debugging → Memeriksa dan menguji aplikasi sebelum dipublikasikan secara online.
6. Mendukung Modul Tambahan → Bisa dikonfigurasi dengan berbagai ekstensi seperti Perl dan MariaDB.



**Notes:** Bagi computer yang belum terinstall XAMPP, silakan ikuti tutorial berikut ini.





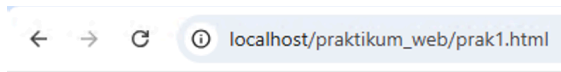


- Tunggu hingga proses instalasi selesai, kemudian buka *XAMPP* dan aktifkan *Apache* serta *MySQL*, dengan menekan tombol Start pada modul *Apache* dan *MySQL*.



- Lakukan kompilasi program melalui browser dengan mengetikkan <http://localhost/namafolder/namafile>, yaitu [http://localhost/praktikum\\_web/prak1.html](http://localhost/praktikum_web/prak1.html)

Maka pada browser akan muncul hasil sebagai berikut:



**Heading ke-1**

**Heading ke-2**

**Heading ke-3**

**Heading ke-4**

**Heading ke-5**

**Heading ke-6**

**ini adalah contoh paragraph**