



## Universitas Kristen Duta Wacana

### Fakultas Teknologi Informasi

#### Prodi Informatika

Jln. Dr. Wahidin Sudiro Husodo No 5-25

Daerah Istimewa Yogyakarta 55224

<b>Mata Kuliah</b>	: Pemrograman Website
<b>Kode Mata Kuliah</b>	: TI0243
<b>Periode</b>	: Semester Genap 2021/2022
<b>Dosen Pengampu</b>	: Danny Sebastian S.Kom., M.M., M.T. : Maria Nila Anggia Rini S.T., M.TI. : Agata Filiana S.Kom., M.Cs.

## BAB 1

### SILABUS, ARSITEKTUR & APLIKASI WEBSITE

#### A. CPL Pertemuan

- Mencari contoh-contoh aplikasi website yang ada di Internet.
- Mencoba Web Development Tools pada Browser
- Mempersiapkan environment yang dibutuhkan dalam pengembangan aplikasi website.

#### B. Alat & Bahan

Kegiatan praktikum pada bagian ini membutuhkan perangkat keras (*hardware*) atau komputer dengan spesifikasi minimum:

- Processor : Intel Core i3
- Memory : 2 GB
- Hard disk : kapasitas kosong penyimpanan sebesar 3 GB

Selain perangkat keras, kegiatan praktikum bagian ini juga membutuhkan perangkat lunak (*software*):

- Sistem Operasi (Windows, Linux, atau MacOSX)
- Text editor (Atom, Visual Studio Code, atau editor lain)
- Browser (Google Chrome, Opera, atau Mozilla Firefox)
- Xampp (Web Server Apache & Database MariaDB)

## C. Silabus Perkuliahan

### C.1. Kompetensi

- Mampu menjelaskan mengenai Arsitektur Internet dan Aplikasi Web
- Mampu merancang dan membuat implementasi sebuah Aplikasi Web statis
- Mampu merancang dan membuat implementasi sebuah Aplikasi Web dinamis
- Mengetahui *Frameworks, Library & Content Management System* yang digunakan dalam pembuatan aplikasi web

### C.2. Kelas Teori

No.	Komponen	Points	Keterangan
1	Kuis	46%	Test Menggunakan moodle, setiap hari Jumat-Minggu
2	Diskusi Kelas	20%	Laporan & Diskusi topik minggu terkait
3	UTS	20%	Test moodle, HTML dan CSS
4	UAS	20%	Test moodle HTML, CSS, PHP, Javascript
	<b>TOTAL</b>	<b>106%</b>	

No.	Materi
1	- Silabus, pengenalan - Arsitektur Internet - Penjelasan adanya Kapita Selekt - Penjelasan Project
2	- Basic HTML - Tag-tag HTML - Tabel: colspan, rowspan - Pengelompokan konten sejenis: Div/semantic box model
3	CSS #1 - How CSS Works - Using CSS - Coloring, background - More cascading rules: pendalaman masalah cascading rules.
4	CSS #2 - HTML display block dan inline - padding, margin, border - basic layouting: header, footer, sidebar/nav
5	Kapita Selekt 1: Database
6	PHP #1 - PHP: Hypertext Preprocessor - Static Request & Dynamic Request

	<ul style="list-style-type: none"> <li>- Syntax dasar PHP</li> <li>- Menggunakan PHP untuk menghasilkan halaman website dinamis</li> </ul>
7	PHP #2 <ul style="list-style-type: none"> <li>- HTML Form &amp; Input Type</li> <li>- Predefined Variables</li> <li>- PHP &amp; Database</li> <li>- PHP Lintas File</li> </ul>
8	UTS: Moodle
9	PHP 3: Upload File & Session
10	JS #1 <ul style="list-style-type: none"> <li>- syntax dasar JS: variable, loop, percabangan</li> <li>- cara pakai JS</li> <li>- demo singkat kalkulator</li> </ul>
11	JS #2 <ul style="list-style-type: none"> <li>- JS praktis untuk tambahan behavior pada halaman website</li> </ul>
12	JS #3 <ul style="list-style-type: none"> <li>- Perkembangan JS, frameworks, dll</li> </ul>
13	Kapita Selekt 2: Bootstrap
14	Kapita Selekt 3: Live the website, Github CDN
15	Next Level Website: <ul style="list-style-type: none"> <li>- web services</li> <li>- frameworks CSS</li> <li>- frameworks PHP</li> <li>- Static site generator</li> <li>- Progressive Website Application</li> </ul>
16	UAS: Moodle

### C.3. Praktikum

No.	Komponen	Points	Keterangan
1	UTS	15%	Tes praktik menggunakan komputer
2	Mini Project #1	15%	Tes praktik coding berdasarkan tema tertentu
3	Mini Project #2		
	Aplikasi website dinamis	25%	Hasil aplikasi website dengan kriteria penilaian, fitur yang wajib ada
	Presentasi	10%	Presentasi akhir mini project
4	Tes Kecil	45%	Tes praktik coding, diambil 10 terbaik
	<b>TOTAL</b>	<b>110%</b>	

No.	Materi
-----	--------

1	- Website developer tools - xampp - visual studio code
2	Hypertext Markup Language (HTML) - Syntax dasar tag HTML - Dokumen HTML - Struktur dasar HTML
3	CSS #1 - Penggunaan CSS dan Selector CSS - CSS untuk pengaturan style dasar
4	CSS #2 - CSS untuk pengaturan layout - Latihan terbimbing membuat sebuah layout halaman website
5	Mini Project #1 - membuat rancangan UI secara statis untuk project dia
6	PHP #1 - PHP: Hypertext Preprocessor - Static Request & Dynamic Request - Syntax dasar PHP - Menggunakan PHP untuk menghasilkan halaman website dinamis
7	PHP #2 - HTML Form & Input Type - Predefined Variables - PHP & Database - PHP Lintas File - PT 5
8	UTS
9	PHP #3 - Cookies dan Session pada PHP
10	PHP #4 - Upload File pada PHP
11	Javascript #1 - Cara penggunaan Javascript - Syntax dasar Javascript - Javascript untuk validasi HTML menggunakan message box
12	Javascript #2 - Modify HTML Structure - Modify CSS
13	Javascript #3 - Modify HTML Structure - Modify CSS
14	Website Dinamis - Praktik mengubah website dinamis dari halaman template statis dengan menambahkan PHP dan javascript
15	Mini Project #2
16	UAS

## D. Materi Kelas Teori

### D.1. Internet dan WWW

Untuk memahami Internet dan WWW, kita perlu memahami apa itu Network, Internet, dan World Wide Web. **Network** pada prinsipnya adalah **kumpulan computer** dan perangkat keras lainnya yang **saling terhubung** oleh sebuah **saluran komunikasi**, yang memungkinkan terjadinya pertukaran **informasi** dan **sumber daya**. Pada pemrograman website, pada dasarnya terjadi antara *client* dengan *server* atau 2 komputer. Akan tetapi pertukaran client dan server juga dapat terjadi pada 1 perangkat yang sama. Pertukaran informasi dan sumber daya tidak terbatas pada jaringan yang bersifat lokal pada perusahaan, maupun pada skala yang lebih besar seperti internet. Pada prinsipnya, internet merupakan jaringan komputer yang sangat besar, bersifat global atau seluruh dunia, dan tidak terbatas pada lokasi.

WWW atau Word Wide Web merupakan halaman website, gambar, video dan konten online lainnya yang dapat diakses melalui browser web. Komunikasi dilakukan menggunakan aturan **Hyper-Text Transfer Protocol** atau **HTTP**. Internet merupakan koneksi jaringan yang memungkinkan kita mengirimkan informasi dan mengakses World Wide Web. WWW dapat diakses menggunakan jaringan dengan skala tertentu, seperti Internet yang bersifat global, maupun Intranet yang bersifat lebih spesifik.

**Aplikasi website** merupakan sebuah software aplikasi yang berjalan diatas **remote server**. Pada umumnya, aplikasi website dapat diakses menggunakan **Web Browser**. Pada umumnya aplikasi website menggunakan basisdata sebagai tempat menyimpan informasi. Aplikasi website memiliki beberapa **keunggulan**,

- **Mudah diupdate.** Hanya perlu melakukan pembaruan file yang berada di server, secara otomatis semua pengguna akan mengakses halaman terbaru. Berbeda dengan aplikasi client, dimana pengguna perlu melakukan pembaruan secara manual atau download pembaruan file, kemudian melakukan instalasi.
- Hanya membutuhkan **sedikit sumber daya komputer**. Sumber daya komputer atau kemampuan komputasi komputer *client* bersifat ringan karena hanya bertugas menampilkan halaman *website*, sedangkan sumber daya komputer *server* bersifat berat karena perlu melakukan pembuatan/persiapan halaman *website* kemudian mengirimkan ke pengguna.

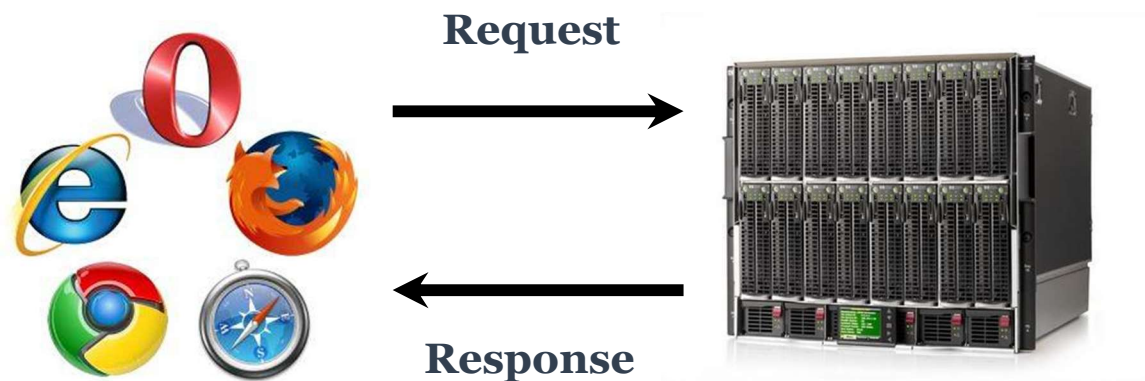
- **Kompatibilitas tinggi.** Tidak ada batasan Sistem operasi antara *client* dengan *server* saat mengakses aplikasi website. Dimana aplikasi website yang diletakkan pada server dengan system operasi Linux dapat diakses menggunakan perangkat mobile dengan system operasi Android. Yang dibutuhkan adalah Web Browser.
- **Tidak ada batasan lokasi.** Aplikasi website dapat diakses melalui perangkat komputer yang terhubung ke Internet. Tidak ada batasan lokasi antara *client* dengan *server*.
- **Dan lain-lain.**

Selain keunggulan tersebut, aplikasi website memiliki beberapa **kelemahan**,

- **Perlu memperhatikan ukuran layar.** Pada prinsipnya setiap aplikasi perlu memerhatikan ukuran layar perangkat yang digunakan, sedangkan aplikasi website diakses oleh beberapa jenis perangkat seperti perangkat komputer, tablet, dan handphone. Dari masing-masing perangkat memiliki ukuran layer yang berbeda-beda. Hal ini membuat permasalahan pada saat mengatur tampilan/development tampilan website. Solusi dari permasalahan ini adalah **Responsive Website Design**.
- **Standarisasi Browser.** Kita tidak dapat menentukan web browser yang digunakan untuk mengakses halaman dari aplikasi website. Masing-masing web browser memiliki standarisasi dan idealism nya masing-masing dalam menerjemahkan kode program aplikasi website. Hal ini membuat developer perlu menyesuaikan aturan dari masing-masing web browser. Solusi dari masalah ini adalah **CSS Reset atau CSS Normalization**. Saat ini sudah ada standarisasi yang dapat diakses di W3School.
- **Masalah Privacy.** Pada aplikasi yang dapat diakses oleh banyak orang dan terhubung pada jaringan Internet, pasti muncul permasalahan privacy atau kebocoran data.
- **Membutuhkan koneksi.** Aplikasi website tidak dapat berfungsi apabila tidak ada jaringan internet. Akan tetapi saat ini aplikasi website terus berkembang, sehingga aplikasi website dapat terus digunakan walau tanpa adanya koneksi internet (hanya perlu koneksi internet di awal). Contoh: Gmail, kita tetap dapat menulis email walaupun internet terputus. Dengan catatan halaman website sudah selesai di-load.

Pada hubungan komunikasi client-server, browser merupakan software yang digunakan pada client (Safari, Chrome, etc), sedangkan web server (HTTP Server) berada di Server. Web Server yang digunakan

bergantung kepada Bahasa pemrograman yang dipilih. Ada Apache HTTP Server, Apache Tomcat, Nginx, IIS, dan lain-lain. **Hyper-Text Transfer Protocol** atau **HTTP** merupakan aturan komunikasi antara client dan server. TCP/IP atau Transmission Control Protocol / Internet Protocol merupakan jalur komunikasi yang digunakan. Saat memberikan response, pesan yang dikirimkan selalu mengandung beberapa informasi, yaitu header dan body. Salah satu informasi penting di header adalah HTTP Response code<sup>1</sup>. Pada body, terdapat dokumen HTML yang akan “diterjemahkan” ke halaman browser.



Application Layer	Application Layer	<b>HTTP</b> , Telnet, FTP, TFTP, SNMP, DNS, SMTP, X Windows, etc..
Presentation Layer		
Session Layer		
Transport Layer	Transport Layer	TCP, UDP, RTP
Network Layer	Internet Layer	IP, ICMP, ARP, RARP
Datalink Layer	Network Access Layer	Ethernet, Token Ring, FDDI, X.25, Frame Relay, RS-232, v.35
Physical Layer		

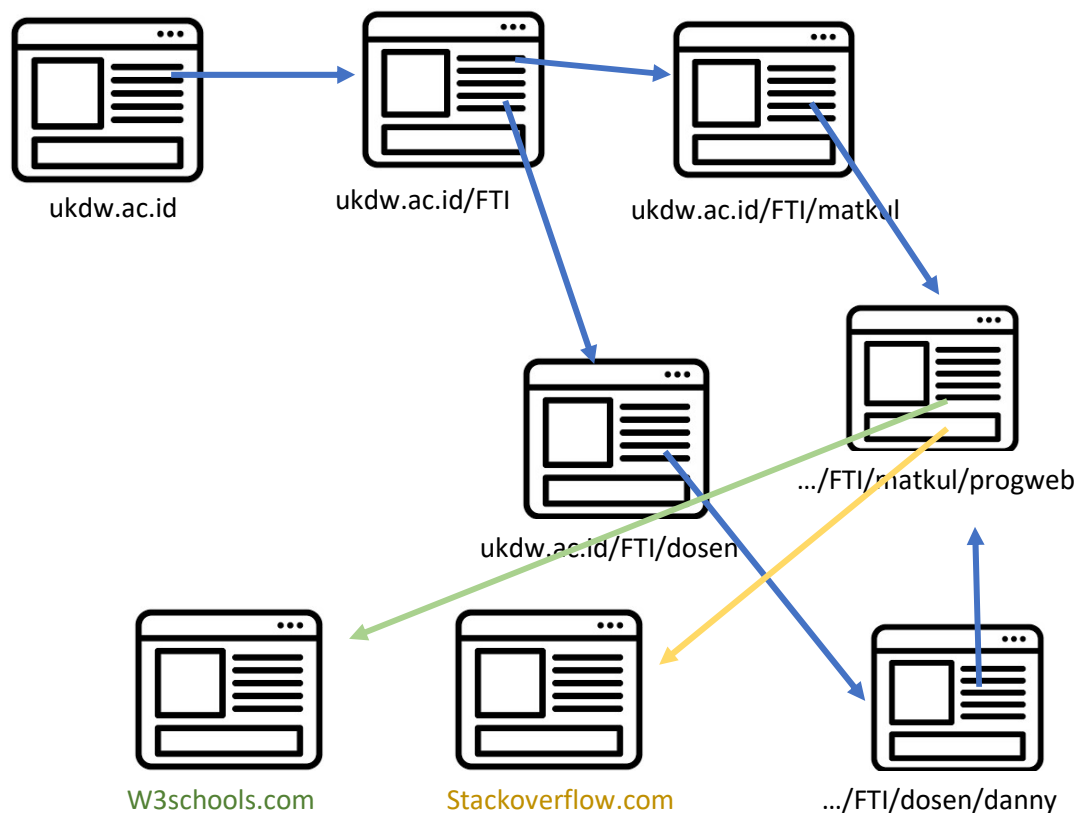
**7 Layers OSI Model**

**4 Layer TCP/IP Model**

HTTP Protocol menggunakan TCP Port 80 secara default. Tetapi kita dapat mengatur port yang digunakan pada web server. Protocol HTTP men-support *hypermedia files*. Untuk memahami hypermedia

<sup>1</sup> [https://en.wikipedia.org/wiki/List\\_of\\_HTTP\\_status\\_codes](https://en.wikipedia.org/wiki/List_of_HTTP_status_codes)

files, perlu dipahami *media files*. Contoh *Media Files*: Buku. Kita tidak bisa berpindah secara langsung ke Bab 5. Yang mungkin kita lakukan adalah membalik satu persatu halaman buku tersebut (atau secara ekstrim membalik beberapa halaman secara langsung). Hal ini berbeda dengan *hypermedia files*, dimana kita dapat berpindah menggunakan *hyperlinks* ke suatu halaman/titik yang kita mau. Contoh: DVD Player musik. Kita bisa berpindah ke lagu no 10 secara langsung. *Hyperlinks* pada website memungkinkan kita menghubungkan sebuah halaman website dengan halaman website lain, bahkan pada alamat website yang berbeda sekalipun. Untuk memahami konsep *hypermedia files* dan *hyperlinks*, perhatikan gambar dibawah ini:



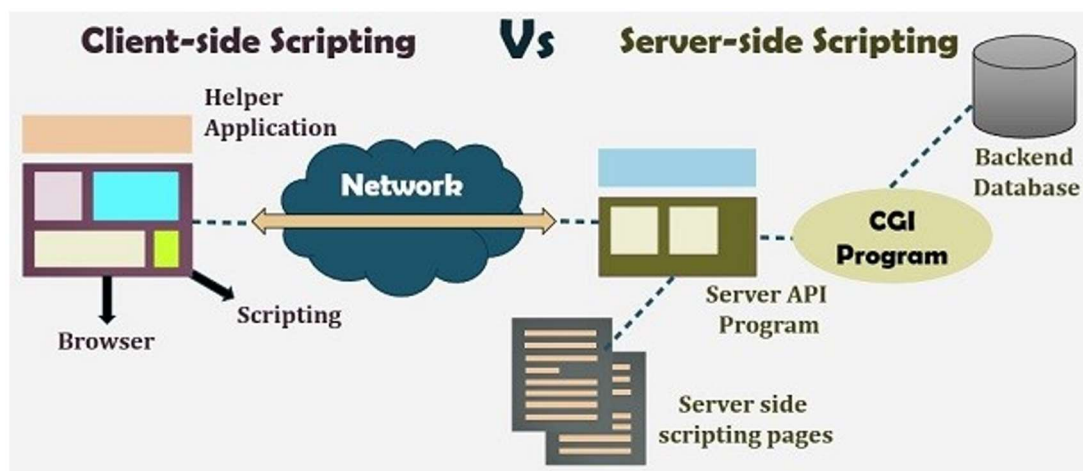
HTTP bersifat “**stateless**”, dimana tidak ada informasi yang masih disimpan antara komunikasi dari client ke server. Contoh: Saat HTTP client (*user agent*) melakukan request ke HTTP server, kemudian server akan memberikan response dan koneksi akan di close. Saat client melakukan request kedua, maka server sudah tidak mengenali client, dan akan menganggap sebagai request yang benar-benar baru. Kasus: Sebuah portal berita yang memungkinkan client menyimpan berita yang disukai. Ketika client memerintahkan “simpan berita A”, kemudian client “berpindah ke berita B”, server sudah tidak



mengetahui siapa client yang sudah “menyimpan berita A”. Sungguh tidak nyaman bukan? Solusinya: Cookies & Session.

## D.2. HTML, CSS, JS, PHP

Dalam kegiatan pembelajaran Pemrograman Website, digunakan 4 “Bahasa pemrograman”, yaitu HTML, CSS, Javascript, dan PHP. HTML bertugas membentuk struktur dokumen. Pada prinsipnya mirip seperti MS Words, dimana kita mendefinisikan mana yang merupakan judul? Mana yang merupakan paragraph? Mana yang merupakan table? Etc. Sedangkan CSS bertugas mempercantik halaman website. Elemen HTML memiliki default standard tampilan. CSS mendefinisikan, elemen HTML paragraph akan memiliki property bold dan judul memiliki properti italic. Javascript memberikan behavior pada elemen HTML. Misal, sebuah tombol kalau di klik akan melakukan fungsi apa? Halaman kalau di scroll akan melakukan fungsi apa? Sedangkan PHP bertugas membentuk dokumen HTML, CSS, dan JS secara dinamis berdasarkan data dari database. Dokumen HTML CSS JS ini yang akan dikirimkan ke pengguna/client.



Client Side	Server Side
<b>Client Hardware:</b> PC Client, perangkat jaringan	<b>Server Hardware:</b> PC Server, perangkat jaringan
<b>Web Browser:</b> Opera, Google Chrome, Safari, etc.	<b>Web Server software:</b> server-side programming language. PHP: Apache Web Server Java: Glassfish, Tomcat Etc.
<b>Client-side Scripting:</b> HTML, Javascript, CSS	<b>Database</b> (MySQL, MariaDB, SQLServer, Oracle, etc)
<b>Plugin lain:</b> Flash Player, Java Applet, etc	

### D.3. Frameworks vs Content Management System (CMS)

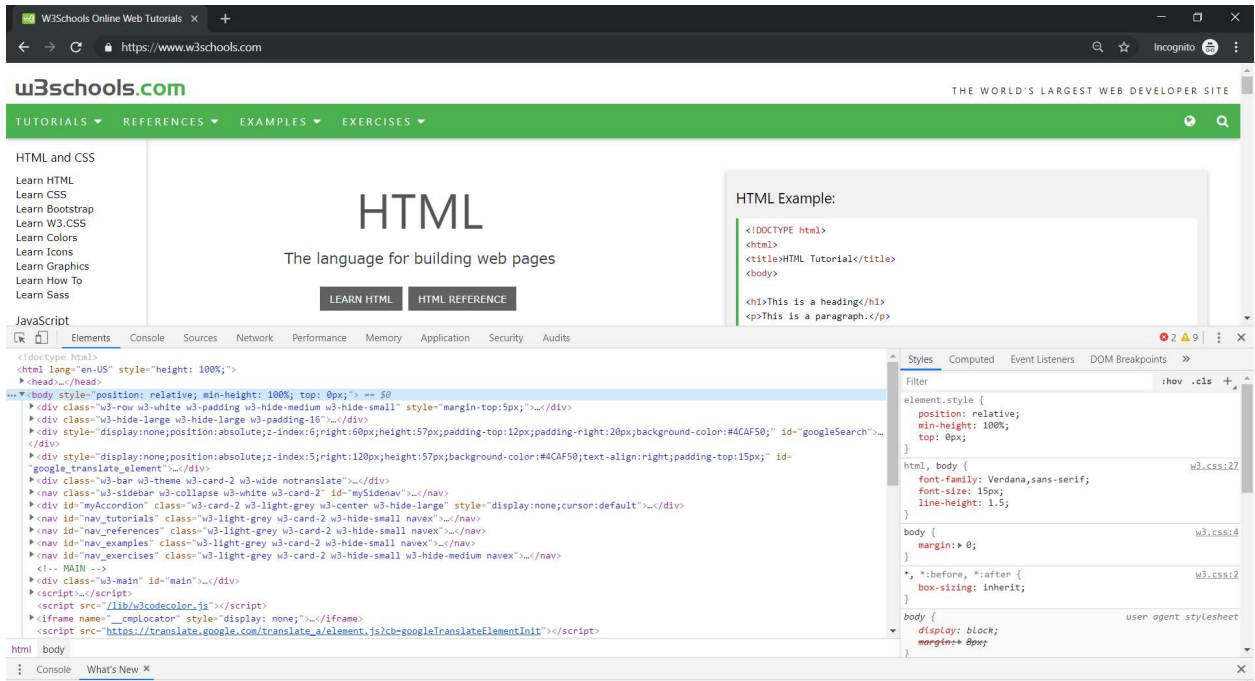
Pengembangan aplikasi Website dapat dilakukan menggunakan Frameworks maupun CMS. Frameworks dan CMS memiliki tujuan untuk mempercepat proses development (tidak hanya pada aplikasi website). Tanpa menggunakan CMS, kita harus membuat fungsi-fungsi pemrograman “from scratch” atau dari awal. CMS merupakan aplikasi website yang menyediakan kustomisasi secara terbatas, hal ini memungkinkan kita membuat website tanpa perlu melakukan koding sama sekali atau hanya sedikit koding. Contoh: Wordpress, blogspot, Joomla, Drupal. CMS sudah menyediakan antarmuka yang membedakan *public facing website* dan *private facing website*.

Frameworks merupakan kerangka development aplikasi, tidak terbatas pada aplikasi website. Dengan menggunakan frameworks kita bisa langsung focus mengembangkan proses bisnis, seperti proses checkout, proses bayar, dan lain sebagainya. Dalam menggunakan frameworks, ada aturan yang harus diikuti. Frameworks bergantung pada Bahasa pemrograman yang digunakan. Contoh frameworks PHP backend: Laravel, CodeIgniter, ZenPHP, CakePHP, dll. Sedangkan frameworks untuk front-end (CSS) aplikasi website: Bootstrap, Material Design, etc.

## E. Kegiatan Praktikum

### E.1. Penggunaan Web Developer Tools

Web development tools merupakan fitur browser untuk membantu developer website melakukan testing dan debug aplikasi website. Web development tools dapat melihat tag HTML, CSS, DOM HTML, javascript, dan parameter/komponen lain dalam website secara realtime melalui perangkat browser.

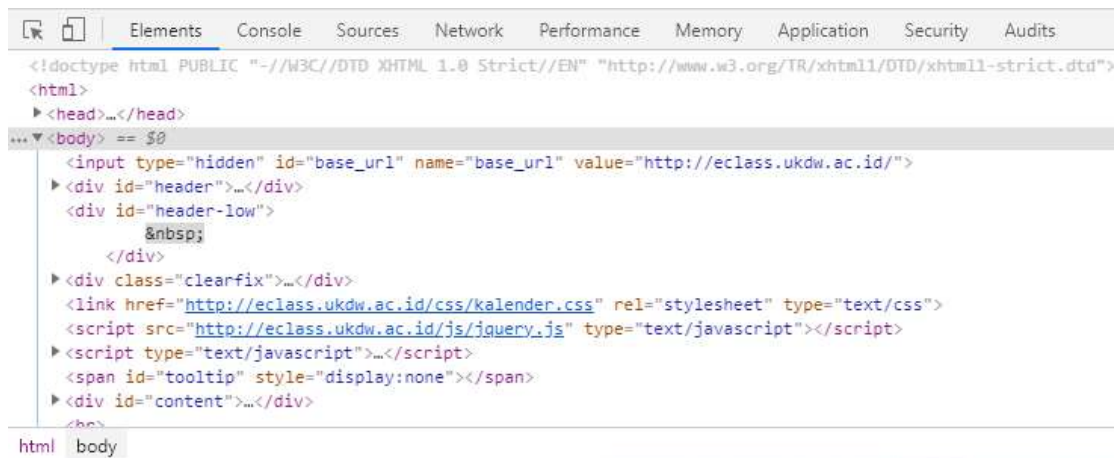


### Langkah Kerja:

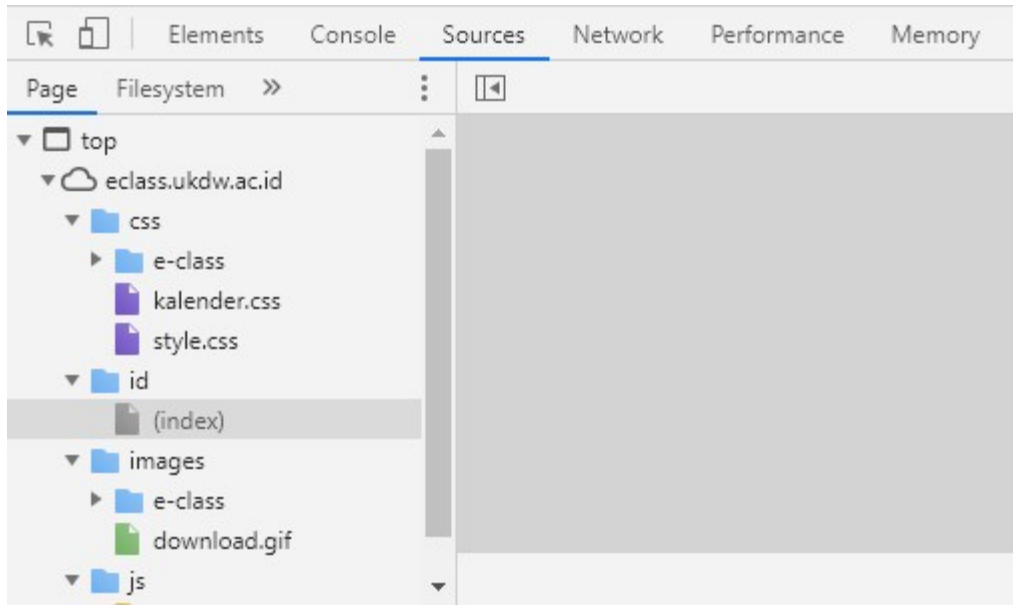
1. Buka Browser, dicontohkan menggunakan Google Chrome. Aplikasi website yang akan diperiksa adalah e-class UKDW.
2. Ada 2 cara membuka Web Developer Tools, pertama Klik kanan pada sebuah elemen pada HTML, kemudian pilih "Inspect". Web Developer Tools akan tampil, kemudian secara spesifik melihat struktur HTML dari elemen yang dipilih.



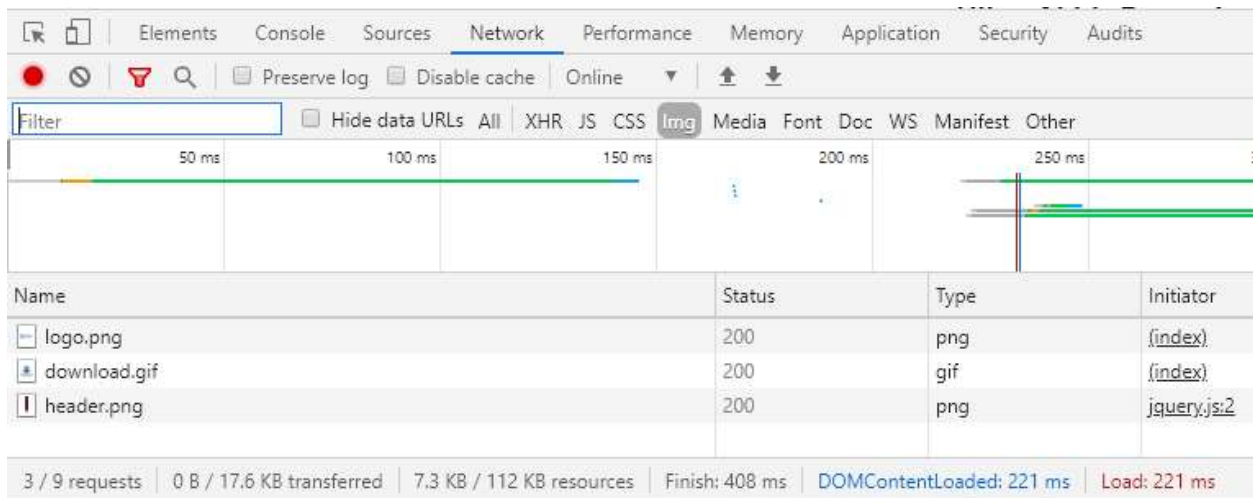
- Atau dengan cara `Ctrl+Shift+I`. Web developer Tools akan tampil, dan tidak menunjuk ke elemen apapun.
- Untuk melihat struktur HTML, pilih Tab **“Elements”**.



- Untuk memeriksa javascript, pilih Tab **“Console”**.
- Untuk melihat file apa saja yang dikirimkan untuk menampilkan halaman, pilih Tab **“Sources”**.



7. Untuk melihat kondisi network/jaringan yang dibutuhkan untuk menampilkan halaman, pilih Tab **“Network”**.



8. Tab lain tidak akan dibahas pada praktikum ini. Apabila mahasiswa tertarik, dapat melakukan eksplorasi sendiri.

## E.2. Instalasi Editor Visual Studio Code

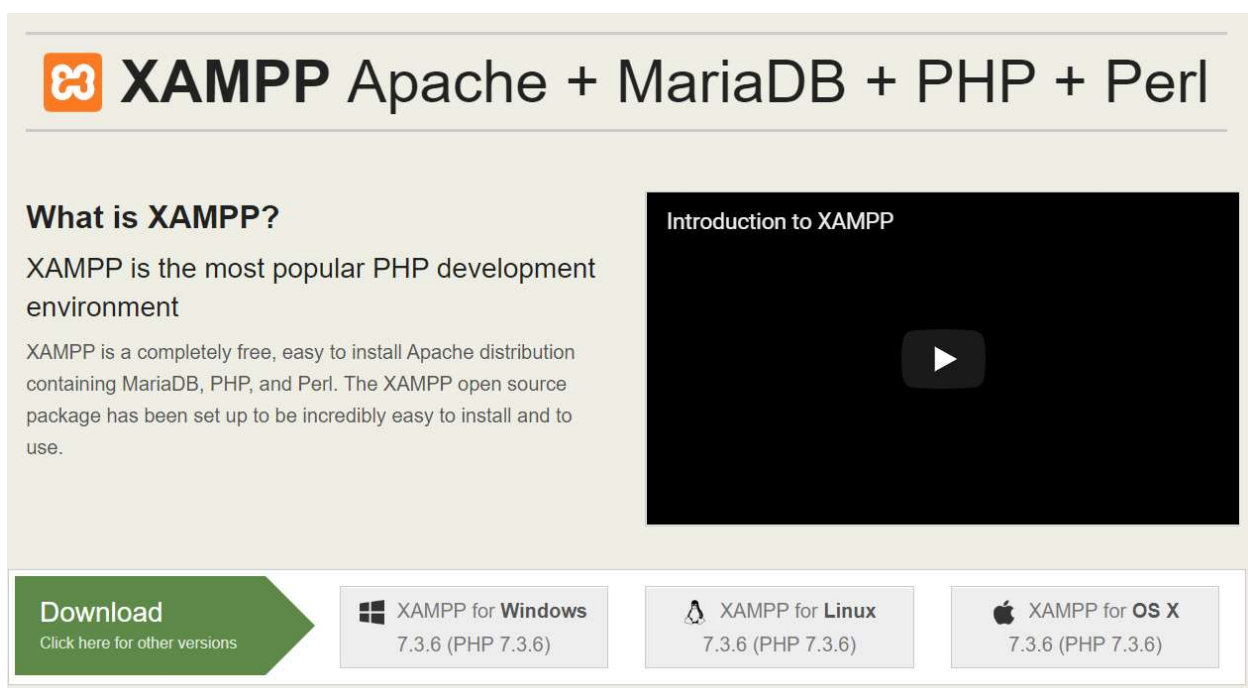
Visual Studio Code merupakan editor yang digunakan pada praktikum ini. Mahasiswa tetap diperbolehkan menggunakan editor selain Visual Studio Code, seperti Atom, Sublime, dan lain sebagainya.

Langkah Kerja:

1. Download visual studio code (<https://code.visualstudio.com/>)
2. Setelah melakukan download, ikuti proses instalasi.
3. Asisten mendemonstrasikan penggunaan snippets dan shortcut pada Visual Studio Code.

### E.3. Instalasi XAMPP

Xampp menyediakan bundling aplikasi yang dibutuhkan dalam pengembangan website, seperti Apache web server, MariaDB, MySQL, dan PHP interpreter. Pada materi bagian 2: Website dinamis, diperlukan web server berbasis PHP dan database untuk pengembangan aplikasi website.



#### Langkah Kerja:

1. Download bundling dari website xampp (<https://www.apachefriends.org/index.html>). Pilih sesuai dengan system operasi yang digunakan.
2. Setelah melakukan download, ikuti proses instalasi. Yang perlu diperhatikan adalah lokasi instalasi. Disarankan tidak melakukan instalasi pada folder yang terlalu dalam (sebaiknya pada root drive C: atau D: atau label lain).

### F. Project Praktikum

### F.1. Mini Project #1

Anda diminta untuk membuat sebuah online to-do list yang dapat digunakan untuk mahasiswa untuk menuliskan kegiatan penting. To-do list ini nantinya akan berbentuk seperti kalender dalam bentuk bulan (see Google Calendar), dimana setiap kotak adalah satu tanggal. Dalam setiap kotak akan berisi kegiatan yang ada pada hari tersebut. Informasi kegiatan meliputi:

- Nama kegiatan
- Tanggal mulai kegiatan
- Tanggal selesai kegiatan
- Level penting: biasa, sedang, sangat penting
- Durasi kegiatan
- Lokasi kegiatan
- Gambar kegiatan (opsional)

#### F.1.1. Instruksi

Mini Project #1 dilakukan secara berkelompok yang terdiri atas 3 orang (tidak boleh lebih). **Pilih anggota kelompok dengan baik karena akan berlanjut hingga Mini Project #2.** Anda akan memakai hasil Mini Project #1 untuk Mini Project #2. Pada Mini Project #1, anda diminta untuk membuat sebuah website statis dengan HTML dan CSS (tidak diperbolehkan menggunakan *template*) yang **memiliki 2 halaman** dengan ketentuan:

- Terdapat 1 halaman utama yang mengeluarkan template kalender seperti yang terlihat pada gambar di bawah ini. Pada setiap kotak diberi nomor 1 - 31 (untuk keterangan hari tidak perlu dinamis) dan pada setidaknya 5 kotak diisi dengan **nama kegiatan** yang memiliki *hyperlink* yang menuju ke halaman detail ketika diklik.

29	30	31	Feb 1	2	3
	Office				
5	6	7	8	9	10
	Office				
12	13	14	15	16	17
	Office				
19	20	21	22	23	24
	Office				
26	27	28	Mar 1	2	3
	Office				

- Halaman detail berisi informasi lengkap: tanggal mulai kegiatan, tanggal selesai kegiatan, level penting: biasa, sedang, sangat penting, durasi kegiatan, lokasi kegiatan, dan gambar kegiatan (opsional) dengan format yang rapi. Pada halaman ini harus ada *hyperlink* untuk kembali ke halaman utama. Halaman juga harus memiliki layout terdiri dari 2 atau 3 kolom.
- Pastikan pada setiap halaman terdapat *header* dan *footer* yang konsisten.
- Gunakan data yang real.
- Tidak menggunakan *frameworks* apapun. Menggunakan CSS murni.
- Halaman website bersifat statis.
- CSS external, bukan INLINE.

### F.1.2. Rubrik Penilaian

RUBRIK PENILAIAN MINI PROJECT #1 PRAKTIKUM PROG. WEB TOTAL = 22 POIN (+ Bonus 3 poin)			
Komponen	Kurang	Baik	Sangat Baik
Layouting	Minimalis, tanpa ada header, footer, konten. (maks. 1 poin)	Tidak ada layouting pada halaman detail. Header dan footer tidak konsisten pada setiap halaman. Tidak bisa mengakomodasi gambar. (maks. 3 poin)	Ada header dan footer di setiap halaman secara konsisten. Ada 2 atau 3 kolom pada halaman detail dan bisa mengakomodasi gambar. (maks. 6 poin)
CSS	Hanya hitam-putih tanpa CSS atau CSS Inline (maks. 1 poin)	Penggunaan CSS embedded atau external. Ada pewarnaan background basic. (maks. 2.5 poin)	Penggunaan CSS embedded atau external. Ada pewarnaan yang menarik. (maks. 5 poin)



Hyperlink	Tidak ada hyperlink (0 poin)	Ada 1 hyperlink pada salah satu halaman saja (maks. 1 poin)	Ada hyperlink yang jelas pada setiap halaman detail. (maks. 2 poin)
Halaman utama	Halaman tidak rapi, tidak ada format kalender (maks. 1 poin)	Terlihat ada format kalender namun tidak ada kegiatan atau tidak memenuhi syarat minimal 5 kegiatan yang terisi (maks. 2 poin)	Terdapat format kalender dengan penomoran yang jelas serta terisi minimal 5 kegiatan (maks. 5 poin)
Halaman Detail Kegiatan	Tidak ada halaman detail kegiatan (0 poin)	Data kegiatan terlihat namun tidak lengkap (maks. 1 poin)	Detail kegiatan terlihat dengan jelas sesuai dengan ketentuan (maks. 4 poin)
BONUS: Penerapan setidaknya dua prinsip IMK  Dijelaskan dan disertai screenshot pada dokumen PDF (dikumpul pada Moodle)	Tidak ada (0 poin)	Terdapat 1 prinsip yang diterapkan secara baik dan benar dengan deskripsi yang memadai. (maks. 2 poin)	Terdapat 2 prinsip yang diterapkan secara baik dan benar dengan deskripsi yang memadai. (maks. 3 poin)

## F.2. Mini Project #2