

## **BAB I SISTEM GALERI BUKU**

Perpustakaan merupakan suatu satuan kerja organisasi bada atau Lembaga. Satuan unit kerja tersebut dapat berdiri sendiri, tetapi dapat juga merupakan bagian dari oragnisasi diatasnya yang lebih besar. Perpustakaan yang berdiri sendiri seperti perpustakaan umum, unit pelaksana teknis perpustakaan pada universitas, dan perpustakaan nasional. Sedangkan perpustakaan yang merupakan bagian dari suatu oragnisasi yang lebih besar seperti perpustakaan khusus atau kedianasan dan perpustakan sekolah.

Fungsi Perpustakaan sekolah sebagai tempat baca yang sekarang semakin kurang diminati oleh siswa untuk membaca buku dan mencari literatur ilmu pengetahuan, hal ini disebabkan karena terkalahkan oleh media informasi seperti internet yang lebih mudah digunakan dalam pencarian berbagai macam ilmu dan sumber bacaan.

Dalam dunia pendidikan, Perpustakaan sangat dibutuhkan oleh siswa untuk mencari bahan referensi namun perpustakaan saat ini memiliki banyak keterbatasan buku, maka dibutuhkan sebuah sistem yang dapat terintegrasi dengan galeri buku yang telah dirancang sebelumnya. Integrasi dapat diartikan adanya saling keterkaitan antara satu sistem dengan sistem yang lain.

Kehadiran teknologi informasi dan komunikasi (TIK) saat ini menjadi suatu yang penting dalam sebuah instansi. Kemajuan TIK dalam proses pertukaran informasi membuat kualitas pelayanan dari suatu instansi meningkat. Oleh karena itu, agar pertukaran informasi dapat dilakukan dengan maksimal harus didukung dengan penggunaan teknologi dan komunikasi yang memadai.[1]

Instansi pendidikan membutuhkan sistem informasi dalam menjalankan kegiatan kerjanya, termasuk SMP 41 Bulukumba sebagai Sekolah Menengah Pertama. SMP 41 Bulukumba saat ini belum menggunakan Sistem Informasi Perpustakaan dalam menjalankan berbagai kegiatan kerjanya.

Perpustakaan dituntut untuk memberikan sebuah layanan informasi pada pengguna yang tepat dengan waktu yang tepat. Layanan dapat terlaksana dengan baik apabila perpustakaan dapat menghadirkan dan memanfaatkan perkembangan teknologi informasi dalam pengelolaan perpustakaan. [2]

Integrasi Sistem Informasi yang tepat dapat mendukung rencana dan pengembangan sistem informasi yang ada di SMPN 41 Bulukumba. Penerapan integrasi sistem jika penerapannya sesuai dengan tujuan yang dibangun dengan menerapkan strategi sistem yang selaras dengan strategi Sistem Galeri Buku. Ketersediaan SMPN 41 Bulukumba masih banyak buku yang tidak tersedia sehingga kurang membantu siswa yang ingin melakukan peminjaman. Sistem informasi belum juga tersedia sehingga dalam mengakses informasi ke sistem lainnya belum dapat dilakukan.

Oleh karena itu, akan dirancang sebuah Sistem informasi yang akan diintegrasikan dengan sistem yang telah dirancang sebelumnya, dimana sistem sebelumnya yaitu Sistem Galeri Buku akan memberikan akses kepada Sistem Informasi yang akan dirancang pada proyek ini.

Dengan intergarasi sistem yang akan dibangun ini memiliki kelebihan yaitu dapat melakukan pertukaran data atau informasi dan memiliki bahasa yang independent. Sehingga integrasi yang akan

dilakukan ini yaitu sistem informasi perpustakaan dengan memanfaatkan teknologi web service.

Dalam perancangan sistem ini menggunakan sebuah kerangka kerja yaitu *codeigniter* yang sudah banyak dipakai dalam pengembangan *website*, yang ringan dan cepat dalam membangun pemrograman website dinamis. Berbeda dengan cms yang memang dapat di unduh secara gratis namun kurang dapat dikembangkan karena masih bersifat statis sehingga perancangan *website* Galeri Buku ini menggunakan kerangka kerja yang dapat dikembangkan dengan dinamis.

## **BAB II LANDASAN TEORI**

### **2.1 Sistem**

Sistem adalah suatu kelompok dari komponen-komponen yang saling berhubungan dengan satu sama lain dengan tujuan yang sama untuk mencapai tujuan tertentu. Sedangkan informasi adalah berupa data yang diolah menjadi bentuk lebih berguna dan berarti bagi pemakainya.[4] Sehingga dapat disimpulkan bahwa system informasi adalah suatu komponen yang saling berhubungan yang bertujuan mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi.



*Gambar Sistem*

*Sumber : <http://www.naesys.com/>*

Sistem terdiri dari bagian-bagian yang saling berkaitan yang beroperasi bersama untuk mencapai beberapa sasaran atau maksud (Budi Sutedjo, 2006:11 ). Sistem terdiri dari elemen-elemen yang berinteraksi untuk mencapai tujuan tertentu suatu sistem mempunyai karakteristik atau sifat-sifat tertentu. Terdapat dua kelompok pendekatan dalam mendefinisikan sistem, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya.

Suatu sistem adalah suatu jaringan kerja untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Sedangkan pengertian prosedur itu sendiri menurut Richard F. Neuschel, prosedur suatu urutan operasi klerikal (tulis menulis), biasanya melibatkan beberapa orang dalam satu atau lebih departemen, yang diterapkan untuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis yang terjadi.

### **2.1.1 Karakteristik Sistem**

Suatu sistem memiliki karakteristik atau sifat-sifat tertentu, yaitu (Sutabri, 2004:12): 153

#### **1. Komponen-komponen (*Components*)**

Suatu sistem terdiri dari sejumlah komponen yang sering disebut dengan subsistem yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai

sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batas sistem (*Boundary*)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (scope) sistem itu sendiri.

3. Lingkungan luar sistem (*Environments*)

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4. Penghubung sistem (*Interface*)

Penghubung merupakan media penghubung antara subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya.

5. Masukan sistem (*Input*)

Masukan yaitu energi yang dimasukkan ke dalam sistem, dimana dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). Masukan perawatan adalah energi yang di inputkan supaya sistem tersebut dapat beroperasi, sedang masukan sinyal adalah energi yang diproses untuk didapatkan keluaran.

6. Keluaran sistem (*Output*)

Keluaran yaitu hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

7. Pengolah sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah *input* menjadi *output*.

8. Sasaran sistem (*Objective*)

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Apabila suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak ada gunanya.

### **2.1.2 Klasifikasi Sistem**

Sistem dapat diklasifikasikan dari berbagai sudut pandang, diantaranya adalah sebagai berikut (Sutabri, 2004:14):

1. Sistem abstrak dan sistem fisik.

Sistem abstrak adalah suatu sistem yang berupa ide-ide pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya teologi yaitu sistem yang berupa pemikiran-pemikiran hubungan antara manusia dengan Tuhan.

Sedangkan sistem fisik adalah suatu sistem yang berupa pemikiran atau ide-ide yang nyata atau yang ada secara fisik. Misalnya sistem komputer, sistem akuntansi, sistem produksi dan lain sebagainya.

2. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Misalnya: sistem perputaran

bumi. Sistem buatan manusia yang melibatkan interaksi antara manusia dengan mesin disebut dengan human-machine system atau ada yang menyebut dengan man-machine system.

3. Sistem tertentu dan sistem tak tentu

Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Contohnya: sistem komputer. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

4. Sistem tertutup dan sistem terbuka

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak berpengaruh dengan lingkungan luarnya. Sedangkan sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan.

Menurut Scott (1996) sistem terdiri dari unsur-unsur seperti masukan (*input*), pengolahan (*processing*), keluaran (*output*) (Hanif Al Fatta 2007:4). Menurut McLeod (2004) sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan (Yakub, 2012:1).

Sedangkan menurut Jogiyanto (1999) terdapat dua kelompok pendekatan sistem didalam mendefinisikan sistem yaitu pendekatan pada prosedur dimana sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu, dan pendekatan pada komponen-komponen atau elemen-elemen, pendekatan pada komponen dianggap lebih mudah dalam mempelajari sistem untuk tujuan dan perancangan system (Yakub, 2012:1).



## **2.2 Sistem Informasi**

Menurut Whitten, Bentley, Dittman “ Sistem informasi adalah pengaturan manusia, data, prosesproses, antarmuka yang saling berinteraksi untuk mendukung kebutuhan penyelesaian masalah dan pengambilan keputusan oleh manajemen dan para pengguna.” (1) “Sistem informasi adalah suatu sistem yang menggunakan teknologi informasi untuk mengambil, menyalurkan, menyimpan, mengambil kembali, memanipulasi, atau menampilkan informasi yang digunakan untuk proses bisnis” (2) Sedangkan menurut Wilkinson (Wilkinson, 1993:14), “Sistem Informasi adalah suatu kerangka kerja dengan mana sumberdaya (manusia,komputer) dikoordinasikan untuk mengubah masukan (data) menjadi keluaran (informasi), guna mencapai sasaran perusahaan”.

Dapat disimpulkan bahwa sistem informasi adalah kumpulan elemen yang saling berhubungan satu sama lain dengan memberikan masukan (data) untuk mendapatkan suatu keluaran (informasi) dengan fungsi untuk memproses, menyimpan, mendistribusikan informasi serta mendukung pengambilan keputusan.

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau mendatang. Sedangkan menurut McLeod dalam informasi sebagai data yang telah diolah menjadi bentuk yang lebih berarti bagi penerimanya (Ladjamudin, 2005:5-6).

### 2.2.1 Kualitas Informasi

Informasi yang baik adalah informasi yang berkualitas (Sutabri, 2004:25), informasi yang berkualitas ditentukan oleh beberapa hal, yaitu :

1. Akurat (*accurate*)

Informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan, informasi harus jelas mencerminkan maksudnya.

2. Tepat waktu (*time lines*)

Informasi yang dihasilkan atau dibutuhkan tidak boleh terlambat, karena nantinya tidak mempunyai nilai yang baik, sehingga apabila dijadikan dasar dalam pengambilan keputusan akan berakibat fatal atau kesalahan pengambilan keputusan dan tindakan.

3. Relevan (*relevance*)

Berarti informasi tersebut mempunyai manfaat bagi pemakainya. Informasi merupakan *Output* dari data yang telah diolah sedemikian rupa hingga menjadi suatu bentuk yang berguna dan telah mempunyai arti. *Output* yang tidak memiliki tiga unsur diatas (Akurasi, Tepat Waktu, Relevan) dapat dikatakan sebagai informasi yang berguna, tetapi merupakan sampah.

Sistem informasi bukan merupakan hal yang baru. Yang baru adalah komputerisasinya. Sebelum ada komputer, Teknik penyaluran informasi yang memungkinkan manajer merencanakan serta

mengendalikan operasi yang telah ada. Komputer menambahkan satu atau dua dimensi, seperti kecepatan, ketelitian dan penyediaan data dengan volume yang lebih besar yang memberikan bahan pertimbangan yang lebih banyak untuk mengambil keputusan.

Sistem informasi sistem yang didalamnya mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan (Sutabri, 2004:35).

Sistem Informasi Sistem adalah kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan dalam usaha mencapai suatu tujuan [5]. Sedangkan dalam buku yang berbeda [3], sistem adalah sekumpulan elemen atau subsistem yang saling bekerjasama atau yang dihubungkan dengan caracara tertentu sehingga membentuk satu kesatuan untuk melaksanakan suatu fungsi guna mencapai suatu tujuan.

Informasi adalah hasil pemrosesan data yang diperoleh dari setiap elemen sistem tersebut menjadi bentuk yang mudah dipahami dan merupakan pengetahuan yang relevan yang dibutuhkan oleh orang untuk menambah pemahamannya terhadap fakta-fakta yang ada [5]. Sedangkan pendapat yang berbeda [3], informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara

langsung saat itu juga atau secara tidak langsung pada saat mendatang.

Sistem informasi adalah kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan untuk mengintegrasikan data, memproses dan menyimpan serta mendistribusikan informasi [5]. Dapat pula diartikan bahwa sistem informasi adalah suatu kumpulan dari komponen-komponen dalam perusahaan atau organisasi yang berhubungan dengan proses penciptaan dan pengaliran informasi [3].

### **2.3 Integrasi Sistem**

Integrasi sistem didefinisikan sebagai suatu rekayasa dalam proses menyatukan subsistem komponen kedalam satu sistem (agregasi dari sub sistem yang bekerja sama sehingga sistem mampu memberikan fungsi menyeluruh) dan memastikan bahwa subsistem berfungsi bersama sebagai suatu sistem, dan dalam teknologi informasi sebagai proses menghubungkan bersama berbagai sistem yang terkomputasi dan aplikasi perangkat lunak secara fisik atau fungsional untuk bertindak sebagai keseluruhan yang terkoordinasi.



*Gambar Integrasi Sistem*

Sumber : [coins.newbium.com](https://coins.newbium.com)

Sistem integrasi memanfaatkan berbagai Teknik dalam mengintegrasikan dua sistem seperti jaringan computer, integrase aplikasi *enterprise*, manajemen proses bisnis atau pengguna pemrograman.

Didunia modern yang terhubung oleh internet integrase sistem sangat memiliki peran penting. Semakin banyak sistem dirancang untuk terhubung, baik dalam sistem yang akan dibangun maupun ke sistem yang sudah digunakan.

Integrasi Sistem Informasi sering disebut *Enterprise Information System*, yaitu sebuah platform teknologi yang memungkinkan organisasi mengintegrasikan dan mengkoordinasikan proses bisnis. Sedangkan Sandoe (2001) mengatakan bahwa integrasi Sistem Informasi bertujuan menggabungkan Sistem Informasi yang tadinya terpisah dengan tujuan sebuah sumber daya informasi yang lebih komplit dan menyeluruh bagi sebuah organisasi.[5]

## **2.4 Perpustakaan**

Perpustakaan adalah sebuah tempat yang digunakan oleh setiap orang mulai dari anak-anak hingga orang dewasa guna untuk menambah wawasan dengan cara membaca buku. Seiring dengan berkembangnya dunia teknologi di zaman sekarang, perpustakaan tradisional di perbaharui agar sesuai dengan era digital. Perpustakaan elektronik dalam pengelolaan yang terdapat di dalamnya mirip dengan yang ada di perpustakaan tradisional, yaitu akuisisi, katalog, penyimpanan, peminjaman, dan pencarian.[6]

*E-Library* dapat dianggap sebagai suatu sistem yang dapat mengelola buku ataupun informasi tentang ilmu pengetahuan dengan menggunakan digital atau secara terkomputerisasi. *E-Library* semakin memungkinkan pengguna mengakses berbagai sumber informasi secara jarak jauh tanpa mempertimbangkan kondisi penyimpanan fisik. *E-Library* sudah dianggap sebagai *effectife way* untuk mengatur sumber daya informasi dan dapat memberikan layanan yang terkait dengan pengguna.[7]

*E-Library* berkembang menjadi sebuah organisasi yang menyediakan sumber daya, termasuk didalamnya staff khusus, bertugas memilih, menyusun, dan menawarkan akses intelektual, menerjemahkan, mendistribusikan, memelihara integritas, menjamin keutuhan dari waktu ke waktu hasil koleksi digital sehingga karya – karya tersebut dapat dibaca dan secara ekonomis tersedia untuk dimanfaatkan oleh komunitas tertentu maupun sekumpulan komunitas. (Waters,1998)

Disampaikan oleh Cleveland (1998), bersumber pada beberapa jurnal dan hasil diskusi sebelumnya maka definisi karakteristik *E-libarary* antara lain :

- a. perpustakaan digital merupakan perpustakaan yang mewakili perpustakaan tradisional yang menyediakan baik koleksi digital dan koleksi tradisional, termasuk koleksi media. Sehingga perpustakaan tersebut memangkas biaya koleksi elektronik dan biaya kertas.
- b. Perpustakaan digital juga termasuk didalamnya adalah materi digital yang sebenarnya berada diluar perpustakaan secara fisik namun memiliki link dari perpustakaan digital lainnya.
- c. Perpustakaan digital juga akan berisi segala proses dan pelayanan yang menjadi tulang belakang dan jaringan syaraf dalam perpustakaan digital. Walau bagaimanapun, beberapa tradisional proses yang akan membangun pola kerja perpustakaan digital, yang akan disempurnakan dan ditingkatkan untuk mengakomodasi perbedaan antara

media digital yang baru dan media tradisional . Sushan Dhakal (2007), memfokuskan pada pemanfaatan Open

Digital Library pada rencana pendidikan masa kini, dan deskripsi teknik dari arsitekturnya. Paper tersebut berdasarkan pada penelitian dan pembangunan Digital Library Research Lab menggunakan OAI\_PMH dan PHP based harvester.

## **2.5 Framework**

*Framework* adalah kumpulan perintah atau fungsi dasar yang membentuk aturan-aturan tertentu dan saling berinteraksi satu sama lain sehingga dalam pembuatan aplikasi website, kiat harus mengikuti aturan dari *framework*.

Dengan *framework* kita tidak perlu memikirkan kode perintah/ fungsi dasar dari aplikasi website. Seperti bagaimana mengambil data dari database untuk ditampilkan. Kita hanya memikirkan apa kode SQLnya dan ditampilkan kemana? Hal penunjang lainnya seperti koneksi database, validasi form, GUI, dan keamanan telah disediakan oleh *framework* sehingga jumlah baris kode yang kita buat jauh lebih sedikit dibandingkan jika semua kode dari kita.

Jadi keuntungan yang dapat diperoleh dari penggunaan *framework* adalah:

1. Waktu pembuatan aplikasi website kita jauh lebih singkat.



2. Kode aplikasi website menjadi lebih mudah dibaca, karena sedikit dan sifatnya pokoknya. Detailnya adalah kode dari *framework* dan ini mungkin difikirkan, terjamin.
3. Website kita menjadi lebih mudah diperbaiki, karena kita tidak perlu focus kesemua komponen kode website terutama kode sistem *framework*.
4. Kita tidak perlu lagi membuat kode penunjang aplikasi website seperti koneksi database, validasi form GUI dan keamanan.
5. Fikiran kita menjadi lebih terfokus ke kode alur permasalahan website. Apa yang ditampilkan dan layanan apa saja yang diberikan dari aplikasi website tersebut.
6. Jika dikerjakan teamwork maka akan lebih terarah karena sistem akan *framework*, mengharuskan adanya keteraturan peletakan kode seperti bagian pengambilan database terpisah dengan bagian pengaturan tampilan untuk pengunjung.

*Codeigniter* dilengkapi dengan berbagai pustaka siap pakai untuk berbagai kebutuhan misalnya saja koneksi database, email, *session*, dan *cokkies*, keamanan dan manipulasi gambar dan banyak lagi sehingga mempermudah pekerjaan kita. Dan jika tidak tersedia kita dapat saja menambah pustaka yang ada dengan menciptakan pustaka sendiri yang di *includekan* kedalam *framework* tersebut.

## 2.6 CodeIgniter

*CodeIgniter* adalah kerangka kerja aplikasi web untuk membangun aplikasi web yang sifatnya *open source* yang digunakan untuk membangun aplikasi PHP yang dinamis. *CodeIgniter* memiliki Karakteristik yang sangat fleksibel dan ringan dalam memodifikasi dan mengintegrasikan suatu website. Dalam *CodeIgniter* dapat juga digunakan pola *Model View Controller* (MVC) sehingga struktur kode yang di hasilkan lebih terstruktur dan memiliki standar yang jelas.[8]

*CodeIgniter* adalah salah satu jawabannya bagaimana kita membuat website tidak dari nol. Dengan *codeigniter* kita tidak perlu pusing membuat website PHP karena *framewaork codeigniter* ini sudah menyediakan fasilitas untuk mempercepat pembuatan website. *Codeiginter* ini menyediakan fungsi-fungsi kode yang dapat digunakan kembali dalam pembuatan website dan ini mempercepat efisiensi dalam mengembangkan aplikasi web.

Ada banyak keuntungan penggunaan *codeigniter* antara lain penghematan waktu, *script website* menjadi lebih udah dibaca dan diupdate, mempermudah pembuatan sistem besar dan terstruktur, kode *script* kita menjadi lebih mudah dikelola dan karena adanya pengelompokan kode makan akan mempermudah pembagian kerja dalam tim jika kita membuat *website* secara gotong royong.

## 2.7 PHP (*Hypertext Preprocessor*)

PHP adalah singkatan dari *Hypertext Preprocessor* yang merupakan bahasa pemrograman yang bersifat *open source* yang berada didalam server yang diproses di server. PHP salah satu bahasa pemrograman yang memiliki script yang terintegrasi dengan HTML dan berada pada server (*server side HTML embedded scripting*).[9]

PHP merupakan bahasa pemrograman pelengkap HTML yang memungkinkan aplikasi web dinamis untuk pengolahan data, pemrosesan data dari *user via form*, membuat buku tamu, toko online, dan lain sebagainya. Dengan mudah PHP dapat melakukan koneksi ke database karen PHP memang dilengkapi fitur yang memungkinkan koneksi ke PHP dilakukan dengan mudah tanpa harus memusingkan (Tim EMS, 2016:55).

PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya akan dijalankan diserver tetapi disertakan pada dokumen HTML. Pembuatan web ini merupakan kombinasi antara php sendiri sebagai bahasa pemrograman dan HTML sebagai pembangun halaman web (Bimo sunarfrihantono, ST 2002:9).



*Gambar Hypertext Preprocessor*

Sumber : <https://www.webhozz.com/>

Menurut Andi Pramono dan M. Syafii (2004 : 2), PHP atau kependekan dari *Hypertext Preprocessor* adalah sebuah bahasa pemrograman berbasis *web* yang mempunyai banyak keunggulan dibandingkan dengan bahasa pemrograman berbasis *web* yang lain. PHP merupakan bahasa pemrograman yang bersumber dari Perl. Sedangkan Perl merupakan pengembangan dari bahasa C.

Oleh karenanya, struktur pemrograman yang ada di PHP sama dengan yang ada di bahasa C. Melihat bahwa PHP merupakan pengembangan dari bahasa C secara tidak langsung, maka PHP mempunyai banyak sekali fitur-fitur yang dapat digunakan. Misalnya, PHP dapat mengakses shell di Linux, mempunyai fungsi yang lengkap berhubungan dengan *networking*.

Salah satu keunggulan PHP disbanding bahasa pemrograman lainnya adalah PHP dapat diperoleh secara gratis, meskipun bukan berarti karena gratis kemampuannya menjadi pas-pasan. PHP sangat powefull. Terbutki dengan banyaknya website yang dibangun menggunakan PHP.

PHP juga terkenal lebih aman daripada bahasa pemrograman website yang lain. PHP juga sudah mendukung OOP (*Object Oriented Programming*) sehingga *maintenance* kode menjadi jauh lebih muda dibandingkan procedural.

PHP bisa berinteraksi dengan database, file dan folder, PHP ini termaksud bahasa *cross-platform*, yang mana PHP ini bisa berjalan di sistem operasi yang berbeda-beda (Windows, Linux, ataupun MAC).[10]

Berdasarkan pernyataan diatas dapat disimpulkan bahwa PHP adalah bahasa adalah sebuah bahasa pemrograman yang sangat fleksibel untuk digunakan dalam membuat sebuah website, dan juga PHP dapat dijalankan di bawah sistem operasi LINUX dan Windows.

## **2.8 MySQL (*My Structure Query Language*)**

MySQL adalah sebuah server database *open source* yang terkenal yang digunakan berbagai aplikasi terutama untuk server atau membuat web. MySQL juga adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS). Kehandalan suatu sistem basis data (DBMS) dapat diketahui dari cara kerja pengoptimasinya dalam melakukan proses perintah-perintah SQL yang dibuat oleh

pengguna maupun program-program aplikasi yang memanfaatkannya.[10]

Berdasarkan dari pengertian diatas kesimpulan dari MySQL adalah sekumpulan sistem database yang banyak digunakan untuk pengembangan suatu aplikasi berbasis web dan bersifat open source.

MySQL merupakan software yang tergolong sebagai DBMS (*Database Management System*) yang bersifat *Open Source*. Open Source menyatakan bahwa *software* ini dilengkapi dengan *source code*. MySQL pada awalnya dibuat oleh perusahaan konsultan bernama TcX yang berlokasi di Swedia. Saat ini pengembangan MySQL berada dibawah naungan perusahaan MySQL AB (Kadir, 2008:2).

Fitur yang terdapat pada MySQL:

1. *Multipatform*

MySQL tersedia pada beberapa platform (Windows, Linux, Unix dan lain-lain).

2. Andal, cepat dan mudah digunakan MySQL tergolong sebagai database server yang andal, dapat menangani database yang besar dengan kecepatan tinggi, mendukung banyak sekali fungsi untuk mengakses database, dan sekaligus mudah untuk digunakan.

3. Jaminan keamanan akses

MySQL mendukung pengamanan database dengan berbagai kriteria pengaksesan.

#### 4. Dukungan SQL

MySQL mendukung perintah SQL (*Structurd Query Language*). Sebagai diketahui, SQL merupakan standar dalam pengaksesan database relasional.

### 2.8. 1 Perkembangan MySQL

Pada bulan Mei 1996, MySQL versi 1.0 berhasil dirilis secara terbatas hanya untuk 4 orang saja. Namun di bulan Oktober pada tahun yang sama versi 3.11.0 dilepas ke publik. Namun mula-mula kode ini tidak diberikan dibawah lisensi GPL (*General Public License*), melainkan lisensi khusus. Pada tahun 1998-1999, yaitu pada versi-versi akhir 3.22, MySQL menjadi semakin populer dan dilirik orang karena kestabilan dan kecepatan yang meningkat. Jika pada versi 3.22, MySQL mulai diadopsi oleh banyak orang. Berbeda halnya dengan versi 3.23 dan 4.0 yang telah terjadi banyak peningkatan dari sisi teknologi (Sukarno, 2006:5-7).

MySQL memiliki beberapa keistimewaan, antara lain :

1. Portabilitas, MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. Perangkat lunak sumber terbuka. MySQL didistribusikan sebagai perangkat lunak sumber terbuka, dibawah lisensi GPL sehingga dapat digunakan secara gratis.

3. *Multi-user*. MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. *'Performance tuning'*, MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. Ragam tipe data. MySQL memiliki ragam tipe data yang sangat kaya, seperti *signed / unsigned integer, float, double, char, text, date, timestamp*, dan lainlain.
6. Perintah dan Fungsi. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (*query*).
7. Keamanan. MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta sandi terenkripsi.
8. Skalabilitas dan Pembatasan. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
9. Konektivitas. MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).
10. Lokalisasi. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.



11. Antar Muka. MySQL memiliki antar muka (*interface*) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).
12. Klien dan Peralatan. MySQL dilengkapi dengan berbagai peralatan yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk online.
13. Struktur tabel. MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya semacam PostgreSQL ataupun Oracle

## **2.9 UML (*Unified Modeling Language*)**

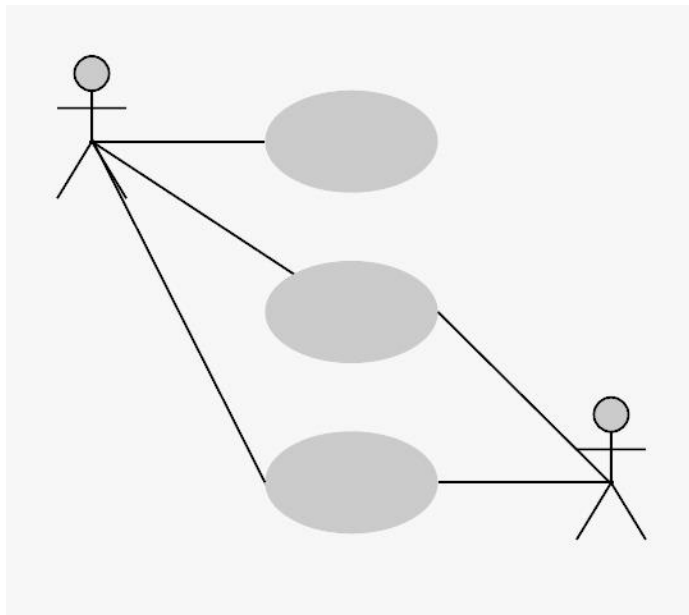
*The Unified Modeling Language (UML)* adalah standar de facto untuk analisis perangkat lunak berorientasi objek dan pemodelan desain.[11] UML membantu memodelkan berbagai aspek sistem melalui berbagai diagram yang di dukungnya. Setiap aspek dari suatu sistem disajikan dengan menggunakan jenis diagram UML tertentu dan satu set diagrama disebut sebagai model.[12]

UML mempunyai sejumlah elemen grafis yang bisa dikombinasikan menjadi diagram. Karena ini merupakan sebuah bahasa, UML memiliki sejumlah aturan untuk menggabungkan atau mengkombinasikan elemen-elemen tersebut.

### *1. Use case diagram*

*Use case diagram* menjelaskan apa yang akan dilakukan oleh sistem yang akan dibangun dan siapa yang berinteraksi dengan sistem. *Use case diagram* menjadi dokumen kesepakatan antara *customer*, *user* dan *developer*. *User*

menggunakan dokumen *use case diagram* ini untuk memahami sistem dan mengevaluasi bahwa benar yang dilakukan sistem adalah untuk memecahkan masalah yang user ajukan. *Use case diagram* memberikan gambaran statis dari sistem yang sedang dibangun dan merupakan artifak dari proses analisis (Hermawan, 2004:23).

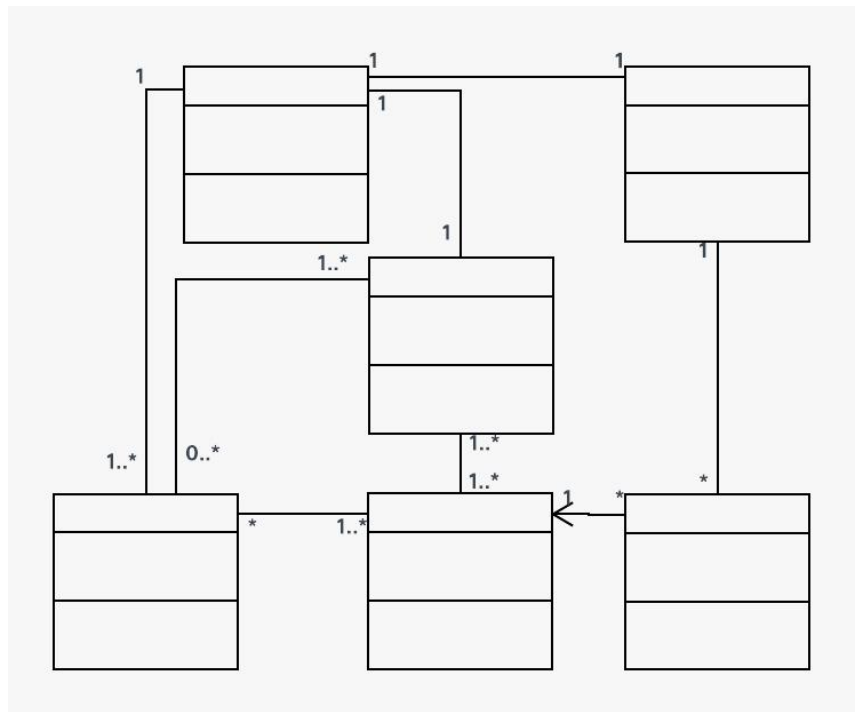


*Gambar Use Case Diagram*

## 2. *Class Diagram*

*Class diagram* merupakan diagram yang selalu ada di pemodelan sistem berorientasi obyek. *Class diagram* menunjukkan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai tujuan. *Class diagram* digunakan untuk

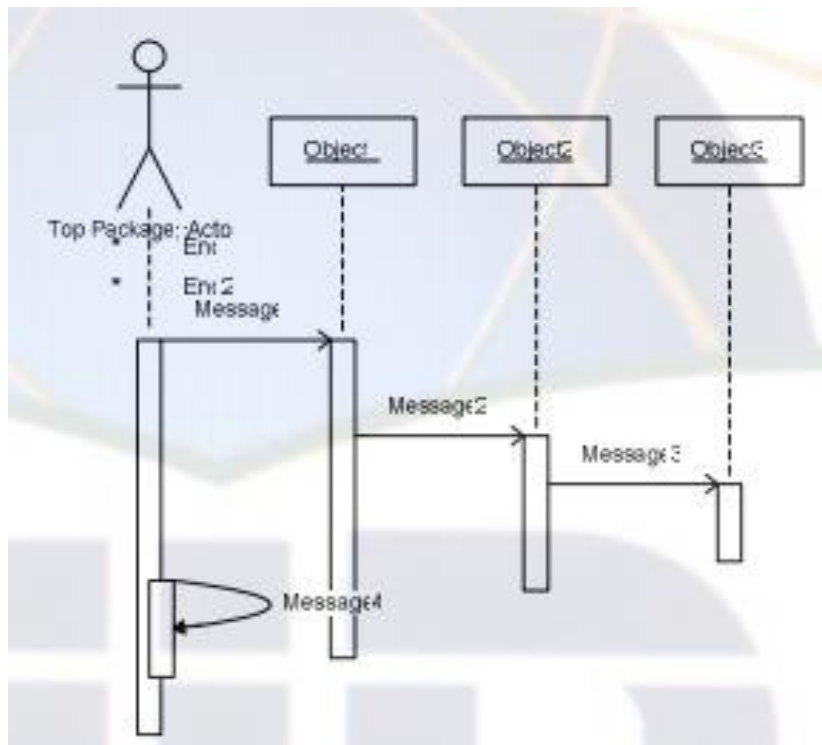
menggambarkan disain statis dari sistem yang sedang dibangun (Hermawan, 2004:27).



*Gambar Class Diagram*

### 3. *Sequence diagram*

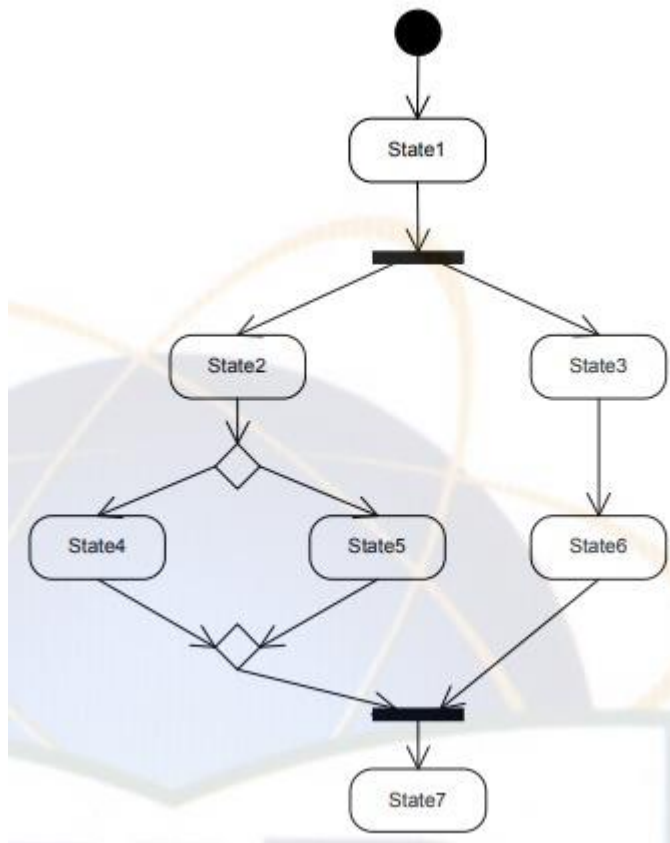
*Sequence diagram* menjelaskan secara detail uruta proses yang dilakukan dalam sistem untuk mencapai tujuan dari use case: interaksi yang terjadi antar *class*, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi. *Sequence diagram* menjelaskan aspek dinamis dari sistem yang sedang dibangun (Hermawan, 2004:24).



*Gambar Sequence Diagram*

#### 4. Activity Diagram

*Activity diagram* adalah teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. *Activity diagram* mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity diagram* mendukung perilaku paralel sedangkan *flowchart* tidak bisa (Munawar, 2005:109).

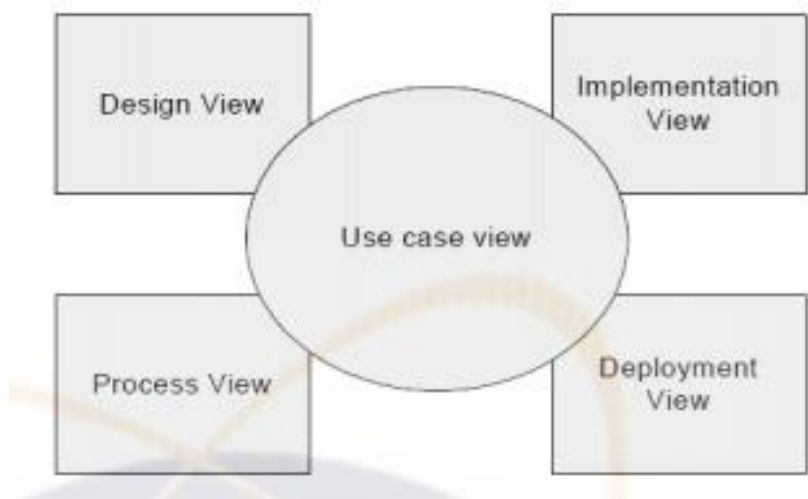


*Gambar Activity Diagram*

## 5. UML

UML dibangun atas model 4+1 view. Model ini didasarkan pada fakta bahwa struktur sebuah sistem dideskripsikan dalam 5 view dimana salah satu diantaranya *use case view*. *Use case view* ini memegang peran khusus diantaranya mengintegrasikan content ke *view* yang lain. Kelima view tersebut tidak berhubungan dengan diagram yang dideskripsikan di UML. Setiap *view* berhubungan dengan perspektif tertentu dimana sistem akan diuji. *View* yang

berbeda akan menekankan pada aspek yang berbeda dari sistem yang mewakili ketertarikan sekelompok *stakeholder* tertentu.



*Gambar UML Diagram*

Penjelasan lengkap gambar UML Diagram tentang sistem bisa dibentuk dengan menggabungkan informasi-informasi yang ada pada *view* pada kelima *view* tersebut.

1. *Use case view* mendefinisikan perilaku eksternal sistem.
2. *Design view* mendefinisikan struktur logika yang mendukung fungsi-fungsi yang dibutuhkan di *use case*. Informasi yang terkandung di *view* ini menjadi perhatian para programmer karena menjelaskan secara detail bagaimana fungsionalitas sistem akan diimplementasikan.

3. *Implementation view* menjelaskan komponen-komponen fisik dari sistem yang akan dibangun. Informasi yang ada di view ini relevan dengan aktifitas-aktifitas seperti manajemen konfigurasi dan integrasi sistem.
4. *Process sistem* berhubungan dengan hal-hal yang berkaitan dengan concurrency di dalam sistem.
5. *Deployment view* menjelaskan komponen-komponen fisik didistribusikan ke lingkungan fisik seperti jaringan komputer dimana sistem akan dijalankan.

## 2.10 Database

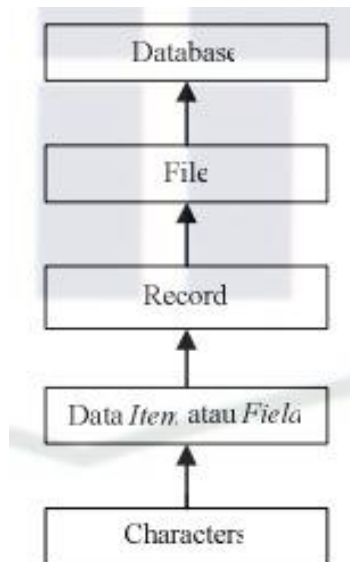


*Gambar Database*

Sumber : robicomp.com

*Database* didefinisikan sebagai kumpulan data yang saling berkaitan secara teknis, database juga sebagai tempat media

penyimpanan data kita dalam membuat sebuah program yang berisikan tabel, field, record, yang diselimuti dengan DBMS (*Database Management System*).[13]



Basis data (database) ini kumpulan dari data yang saling berhubungan satu dengan lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya (Jogiyanto, 2005:217). Jadi basis data merupakan suatu komponen utama sistem informasi karena semua informasi untuk pengambilan keputusan berasal dari data di basis data.

Sampai dengan membentuk suatu data base, data mempunyai jenjang yang dapat dilihat pada gambar (Jogiyanto, 1999:714-715):

*a. Characters*

*Characters* merupakan bagian data yang terkecil, dapat berupa karakter numerik, huruf atau pun karakter-karakter



husus (*special characters*) yang membentuk suatu item data atau field.

b. *Field*

*Field* menggambarkan suatu atribut dari record yang menunjukkan suatu item dari data, seperti misalnya nama, alamat dan lain sebagainya. Kumpulan dari field membentuk suatu record

1) Nama dari *field* (*field name*)

*Field* harus diberi nama untuk membedakan *field* yang satu dengan *field* yang lain.

2) Representasi dari *field* (*field representation*)

Representasi dari *field* menunjukkan tipe dari *field* (*field type*) dapat berupa tipe *numeric*, karakter atau huruf, tanggal, dan memo, serta lebar dari field (*field width*) menunjukkan ruang maksimum dari field yang dapat diisi dengan karakter-karakter data.

3) Nilai dari *field* (*field value*)

Nilai dari *field* menunjukkan isi dari *field* untuk masingmasing *record*.

c. *Record*

*Record* merupakan kumpulan dari *field* yang membentuk suatu *record*. *Record* menggambarkan suatu unit data individu tertentu. Kumpulan dari *record* membentuk suatu file. Misalnya file mahasiswa, tiap-tiap record dapat mewakili data tiap-tiap mahasiswa.

d. *File*

*File* terdiri dari *record-record* yang menggambarkan satu kesatuan data yang sejenis. Misalnya file mata kuliah berisi data tentang semua mata kuliah yang ada.

e. Database Database merupakan kumpulan dari file membentuk suatu database. Tujuan basis data yang efektif termuat di bawah ini:

- Memastikan bahwa data dapat dipakai diantara pemakai untuk berbagai aplikasi.
- Memelihara data baik keakuratan maupun konsistensinya.
- Memastikan bahwa semua data yang diperlukan untuk aplikasi sekarang dan yang akan datang akan disediakan secara cepat.
- Membolehkan basis data untuk berkembang dan kebutuhan pemakai untuk berkembang.
- Membolehkan pemakai untuk mengembangkan pandangan personalnya tentang data tanpa memperhatikan cara data disimpan secara fisik.

*DataBase Management System* (DBMS atau DMS) adalah paket perangkat lunak yang lengkap digunakan untuk memanipulasi database (Jogiyanto, 1999:731). Lebih detail lagi dijelaskan oleh Hariyanto bahwa DBMS adalah perangkat lunak untuk mendefinisikan, menciptakan, mengelola dan mengendalikan pengaksesan basisdata. Semua operasi *input* dan *output* yang berhubungan dengan database harus menggunakan DBMS. Bila pemakai akan mengakses database, DBMS menyediakan penghubung

(*interface*) antara pemakai dengan database (Jogiyanto, 1999:734).

Hubungan pemakai dengan database dapat dilakukan dengan cara:

- a. Secara interaktif menggunakan bahasa pertanyaan (*query language*).
- b. Dengan menggunakan program aplikasi.

Kegunaan Basis Data Penyusunan satu basis data digunakan untuk mengatasi masalah-masalah pada penyusunan data [2], yaitu:

- 1) Redudansi dan inkonsistensi data Jika file-file dan program aplikasi diciptakan oleh programmer yang berbeda pada waktu yang berselang cukup panjang, maka ada beberapa bagian data mengalami penggandaan pada file-file yang berbeda. Penyimpanan data yang berulang-ulang di beberapa file juga dapat mengakibatkan inkonsistensi (tidak konsisten).
- 2) Kesulitan Pengaksesan Data Suatu saat dibutuhkan untuk mencetak data siapa saja, padahal belum tersedia program yang telah tertulis untuk mengeluarkan data tersebut maka kesulitan tersebut timbul, dan penyelesaiannya untuk itu adalah kearah Sistem Manajemen Basis Data yang mengambil data secara langsung dengan bahasa yang familian dan mudah digunakan.
- 3) Isolasi data untuk standarisasi Jika data tersebar dalam beberapa file dalam bentuk format yang tidak sama, maka ini menyulitkan dalam menulis program aplikasi untuk mengambil dan menyimpan data, maka haruslah data dalam

satu basis data dibuat satu format sehingga mudah membuat program aplikasinya.

- 4) Masalah keamanan atau Security Setiap pemakai sistem basis data tidak semuanya diperbolehkan untuk mengakses semua data. Misalnya data mengenai gaji pegawai hanya boleh dibuka oleh bagian keuangan dan personalia. Keamanan ini dapat diatur lewat program yang dibuat oleh pemrogram atau fasilitas keamanan dari operating sistem.
- 5) Masalah Integrasi (Kesatuan) Basis Data berisi file yang saling berkaitan, masalah utama adalah bagaimana kaitan antara file tersebut terjadi. Meskipun diketahui bahwa file A berkaitan dengan file B, namun secara teknis maka ada file kunci yang mengaitkan kedua file tersebut.
- 6) Masalah data independence (kebebasan data) Aplikasi yang dibuat dengan bahasa yang diciptakan dari Sistem Manajemen Basis Data, apapun yang terjadi pada struktur file, setiap kali hendak melihat data cukuplah dengan utility USE, hendak menambah data cukup dengan APPEND, ini berarti perintah-perintah dalam paket Sistem Manajemen Basis Data bebas terhadap basis data. Perubahan apapun dalam basis data, semua perintah akan mengalami kestabilan tanpa perlu ada yang diubah.

Berdasarkan dari uraian di atas dapat disimpulkan bahwa database adalah sekumpulan data yang beris tabel, field, dan record yang disimpan secara sistematis.

## **2.11 API (*Application Programing Interface*)**

*Application Programming Interface* atau API adalah sebuah dokumentasi yang terdiri dari *interface*, kelas, fungsi, struktur dan sebagainya agar dapat membangun sebuah perangkat lunak. Dan API bisa dikatakan sebagai suatu kode pemrograman penghubung antara aplikasi atau web yang telah kita buat dengan fungsi yang dikerjakan [10]. Google+ api merupakan pemrograman yang menghubungkan ke Google+ [11].

Artinya dengan API, web PenA (*Penilaian Absensi*) akan terintegrasi dengan Rinfo+. Yang akan menghubungkan foto profile pada Rinfo+ dalam sistem PenA (*Penilaian Absensi*) . Dalam hal ini Google Api sebagai kode pemrograman yang disederhanakan yang dapat ditambahkan oleh sistem PenA (*Penilaian Absensi*) untuk mengakses dan terintegrasi dengan fitur Rinfo+.

*Application Programming Interface* (API) atau Antarmuka Pemrograman Aplikasi adalah sekumpulan perintah, fungsi, dan protokol yang dapat digunakan oleh programmer saat membangun perangkat lunak untuk sistem operasi tertentu. [14]



lain. API dapat dikatakan sebagai penghubung suatu aplikasi dengan aplikasi lainnya yang memungkinkan programmer menggunakan sistem function. Proses ini dikelola melalui sistem operasi. Keunggulan dari API ini adalah memungkinkan suatu aplikasi dengan aplikasi lainnya dapat saling berhubungan dan berinteraksi. [15]

*Application Programming Interface* (API) ini yang memfasilitasi pertukaran informasi atau data antara dua atau lebih aplikasi perangkat lunak. API adalah antarmuka virtual antara dua fungsi perangkat lunak yang saling bekerja sama, seperti antara sebuah word processor dan sebuah spreadsheet [7]. Sebuah API mendefinisikan bagaimana cara programmer memanfaatkan suatu fitur tertentu dari sebuah komputer. API tersedia untuk sistem windowing, sistem file, sistem database, serta sistem jaringan.

Sistem informasi berbasis API memungkinkan sebuah back-end dimanfaatkan dengan cara yang lebih luas karena logic pada sistem dengan logic pada antarmukanya terpisah. Sistem informasi berbasis API juga akan mempermudah sebuah sistem untuk berkolaborasi dengan sistem lain. Hal ini jadi sangat berguna jika pada waktu mendatang proses bisnis yang difasilitasi menuntut interaksi dengan sistem lain, atau minimal akan sangat berguna pada pengembangan modul-modul baru pada sistem informasi terkait.

Secara umum API merupakan ekspresi terfokus keseluruhan fungsional dalam suatu modul software yang dapat diakses oleh orang yang membutuhkan dengan cara yang telah ditentukan layanan.

Representasi terfokus dari fungsi yang dideklarasikan dalam API dimaksudkan untuk menyediakan rangkaian layanan yang spesifik untuk target tertentu. Jika dalam satu modul memiliki API ganda, hal ini sudah menjadi hal yang umum karena setiap API dimaksudkan untuk penggunaan yang spesifik dari modul terkait (Rama dan Avinash, 2015).

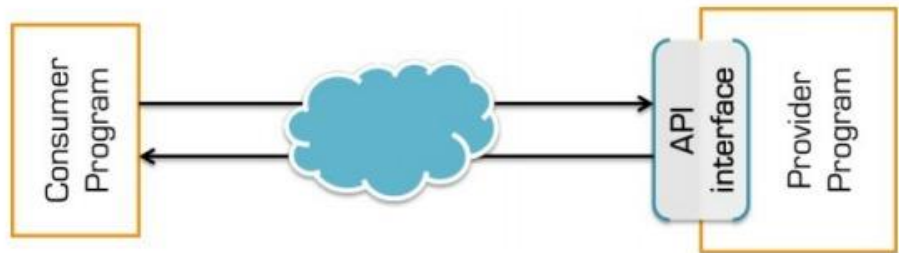
Web API adalah antar muka program dari sistem yang dapat diakses melalui *method* dan *header* pada protokol HTTP yang standar. Web API dapat diakses dari berbagai macam HTTP client seperti browser dan perangkat mobile. Web API juga memiliki keuntungan karena menggunakan infrastruktur yang juga digunakan oleh web terutama untuk penggunaan *caching* dan *concurrency* (Miller dkk, 2014).

*Windows Application Programming Interface* (API) terdapat dalam file-file DLL yang menyusun Sistem Operasi Windows[14]. Karena terdapat dalam file *library* (DLL) maka fungsi-fungsi dan konstanta-konstanta tersebut dapat digunakan oleh aplikasi atau library lainnya. Windows API biasanya disusun dalam bahasa pemrograman C/C++. Bahasa C dipilih karena kecepatan dan kedekatannya dengan hardware.

Windows menyediakan banyak sekali API yang memiliki fungsi berbedabeda. Berikut ini adalah beberapa API yang disediakan oleh Sistem Operasi Windows :



1. Win32 API, API yang disediakan Windows untuk melakukan pemrograman Windows secara umum. Fungsi-fungsi yang terdapat dalam Win32 API ini sangat lengkap mulai dari menu, button, window, eksekusi program dan lain-lain.
2. *Telephony* API (TAPI), API yang digunakan untuk melakukan fungsifungsi telephony dengan menggunakan Sistem Operasi Windows.
3. *Messaging* API (MAPI), API yang digunakan untuk melakukan pengiriman pesan (termasuk pengiriman email) dengan menggunakan Sistem Operasi Windows.
4. *Speech* API (SAPI), API yang disediakan Windows untuk menambahkan fungsi speech ke dalam aplikasi yang berjalan di atas Sistem Operasi Windows.
5. *Cryptography* API, API untuk melakukan fungsi-fungsi enkripsi dan dekripsi. Banyak sekali metode enkripsi/dekripsi yang disediakan oleh cryptography API ini.
6. dan masih banyak API lainnya (termasuk yang tidak didokumentasikan oleh Microsoft). Seperti telah disinggung sebelumnya bahwa API sebenarnya hanyalah sebuah fungsi. Seperti halnya fungsi-fungsi lainnya, fungsi tersebut dapat memiliki parameter-parameter dan dapat mengembalikan nilai. Bedanya hanyalah fungsi-fungsi tersebut ditulis dalam bahasa C/C++ sehingga bagi yang belum familiar dengan bahasa C/C++ biasanya akan mengalami kesulitan.



*Gambar 1 API*

API memungkinkan programmer untuk menggunakan fungsi standar untuk berinteraksi dengan sistem operasi lain. API diimplementasikan dengan menulis fungsi panggilan atau sintaks dalam program, yang menyediakan sarana yang diperlukan untuk meminta layanan program.

Pada dasarnya, program API mendefinisikan cara yang tepat bagi pengembang untuk meminta layanan dari program itu. Sebagai contoh, Amazon.com merilis API sehingga pengembang situs web dapat lebih mudah mengakses informasi produk Amazon, menggunakan Amazon API.

Sebuah situs web pihak ke-tiga dapat memposting link langsung ke produk Amazon dengan harga yang terupdate dan pilihan untuk “beli sekarang”. (Prasetyo, 2014) Dalam API terdapat fungsi-fungsi/perintah-perintah untuk menggantikan bahasa yang digunakan dalam *system calls* dengan bahasa yang lebih terstruktur dan mudah dimengerti oleh pengembang. Fungsi yang dibuat dengan menggunakan API tersebut kemudian akan memanggil *system calls*

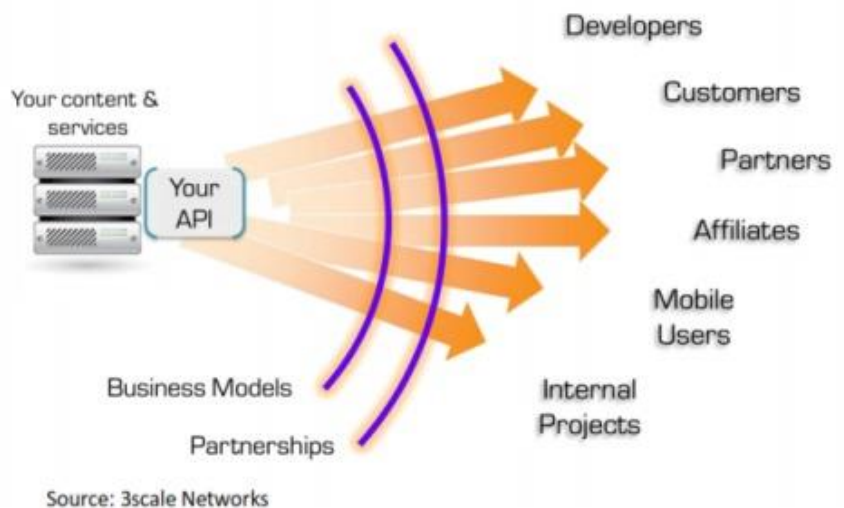
sesuai dengan sistem operasinya. Tidak tertutup kemungkinan nama dari systemcalls sama dengan nama di API.

Keuntungan memprogram dengan menggunakan API adalah:

1. Portabilitas

Pengembang yang menggunakan API dapat menjalankan programnya dalam sistem operasi mana saja asalkan sudah ter-install API tersebut. Sedangkan system call berbeda antar sistem operasi, dengan catatan dalam implementasinya mungkin saja berbeda (Gambar 1).

2. Lebih mudah dimengerti API menggunakan bahasa yang lebih terstruktur dan mudah dimengerti daripada bahasa system call. Hal ini sangat penting dalam hal editing dan pengembangan (Gambar 2).



*Gambar 2 API*

*System call interface* ini berfungsi sebagai penghubung antara API dan system call yang dimengerti oleh sistem operasi. System call interface ini akan menerjemahkan perintah dalam API dan kemudian akan memanggil system calls yang diperlukan. Untuk membuka suatu file tersebut user menggunakan program yang telah dibuat dengan menggunakan bantuan API, maka perintah dari user tersebut diterjemahkan dulu oleh program menjadi perintah `open()`. Perintah `open()` ini merupakan perintah dari API dan bukan perintah yang langsung dimengerti oleh kernel sistem operasi.

Oleh karena itu, agar keinginan user dapat dimengerti oleh sistem operasi, maka perintah `open()` tadi diterjemahkan ke dalam bentuk *system call* oleh *system call interface*. Implementasi perintah `open()` tadi bisa bermacam-macam tergantung dari sistem operasi yang kita gunakan (Setiawan, 2014).

## **2.12    *Web Application***

Dewasa ini *web application* dikenal sebagai aplikasi yang diakses *melalui web browser* dan melalui jaringan seperti Internet atau intranet. Kemampuan untuk memperbarui dan memelihara aplikasi web tanpa harus mendistribusikan dan menginstal perangkat lunak pada kemungkinan ribuan komputer klien merupakan keunggulan teknologi ini, selain juga untuk cross-platform compatibility. Termasuk aplikasi web common webmail, penjualan ritel online, online pelelangan, wiki dan banyak fungsi lainnya.

Pada jurnal yang ditulis oleh Xu, dkk, (2005), menitikberatkan pada efektifitas dan efisiensi sebuah testing terhadap aplikasi yang berbasis web application dengan membandingkan dua metode yaitu Semantic Label dan XML description technique. Lei Xu dan timnya mengembangkannya dengan melengkapi mekanisme feedback control pada pembangunan aplikasi agar lebih menyempurnakan kualitas sistem. Edinburgh (2005) membahas sebuah pendekatan pengujian pada web application. Dalam metode pendekatannya analisa aliran data akan dianggap sebagai *Function Level Testing*, *Function Cluster Level Testing*, *Object Level Testing* dan *Web Application Level Testing*, dari level terendah hingga level tertinggi.

### 2.13 XAMPP



*Gambar XAMPP*

Sumber : [commons.wikimedia.org](https://commons.wikimedia.org)

XAMPP adalah sebuah *software web server apache* yang didalamnya sudah tersedia database server MySQL dan dapat mendukung pemrograman PHP. XAMPP merupakan software yang mudah digunakan, gratis dan mendukung instalasi di Linux dan Windows. Keuntungan lainnya adalah cuma menginstal satu kali sudah tersedia Apache Web Server, MySQL Database Server, PHP Support (PHP 4 dan PHP 5) dan beberapa module lainnya.

Perbedaan versi untuk Windows operating system sudah dalam bentuk instalasi grafis dan yang Linux dalam bentuk file terkompresi tar.gz. Kelebihan lain yang berbeda dari versi untuk Windows adalah memiliki fitur untuk mengaktifkan sebuah server secara grafis, sedangkan Linux masih berupa perintah-perintah di dalam console.

#### **2.14    *CSC (Cascading Style Sheets***



*Gambar Logo CSS*

Sumber : petanikode.com

CSS (*Cascading Style Sheets*) Pada versi HTML yang terdahulu, web browser mengontrol tampilan (rendering) dari setiap halaman web. Jika menggunakan elemen H1 pada (*large heading*) pada web dokumen, browser akan merender elemen tersebut.

Dengan adanya CSS, *programmer* dapat mengontrol bagaimana *browser me-render* halaman web. Mengaplikasikan CSS pada halaman web dapat memberikan tampilan yang lebih menarik dan spesifik sesuai dengan tema pada sebuah web site.

Teknologi CSS memberikan fasilitas untuk menentukan style (misal;*spacing, margins*) dari elemen halaman web terpisah dari struktur dokumen web (*section headers, body text, links* ). Pemisahan tersebut memberikan peningkatan yang lebih besar dalam pengaturan *web pages*, dan membuat perubahan – perubahan *style* dalam dokumen dapat dilakukan lebih cepat dan lebih mudah.

## **2.15    *Local Host***

*Localhost* adalah sebuah akses local yang didapat dari sebuah aplikasi untuk dapat mengakses local server yang dimaksudkan untuk mengetahui bagaimana kekurangan dan kelebihan aplikasi yang akan digunakan sebelum dilakukan hosting.

*Hosting* adalah sebuah tempat dimana kita bisa menyimpan data-data website kita sehingga dapat diakses lewat

internet. Alasan peneliti menginstall di *localhost* sebelum *hosting* dikarenakan selain mengetahui kekurangan dan kelebihan aplikasi peneliti juga dapat menghemat penggunaan internet jika ada kesalahan yang dialami sebelum di *upload* pada *hosting* yang akan digunakan.

Localhost adalah server lokal yang ada pada komputer. Localhost adalah hostname dari komputer itu sendiri. Localhost dapat diakses ketika ada web server yang berjalan.

*Local Host* Menurut Kasiman Peranginangin (2006), *Localhost* adalah sebuah *host server* yang ada pada komputer Anda sendiri. *Localhost* biasanya digunakan untuk kebutuhan intranet atau untuk mempraktikkan pembuatan *web* sebelum di-*upload* ke *web hosting*. Salah satu penyedia server local adalah Xampp.

Xampp sendiri merupakan kepanjangan dari X, Apache, MySQL, PHP, dan Perl. X di sini maksudnya Xampp, yaitu cross (X) platform yang dapat digunakan pada sistem operasi Windows, Linux, Mac, dan Solaris. Sedangkan Apache ialah server yang digunakan. MySQL berarti basis data yang digunakan untuk localhost. PHP dan Perl merupakan script/bahasa pemrogramannya.

## **2.16    *Flow Map***

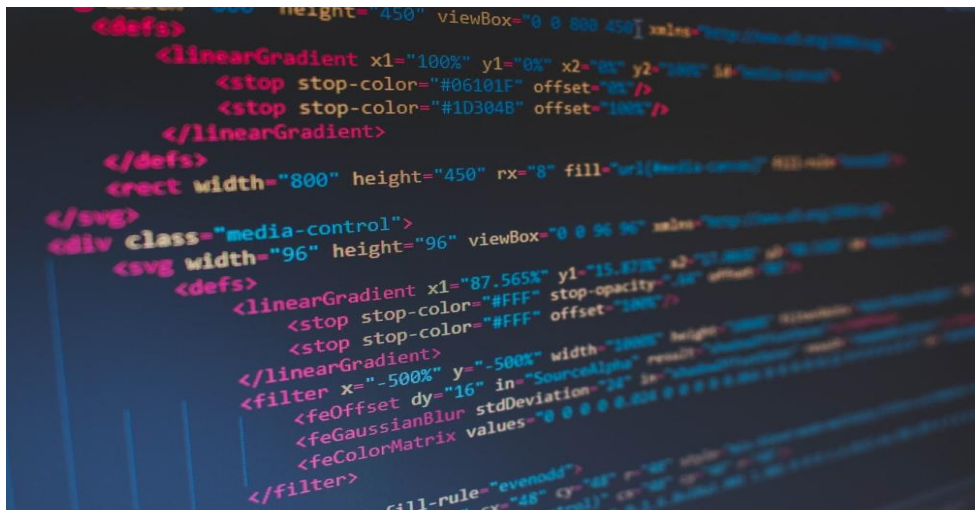
*Flowmap* adalah campuran peta dan *flow chart*, yang menunjukkan pergerakan benda dari satu lokasi ke lokasi lain, seperti jumlah orang dalam migrasi, jumlah barang yang diperdagangkan,



atau jumlah. *Flowmap* menolong analisis dan programmer untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.[5]

*Flowmap* merupakan diagram alir dokumen yang digunakan untuk menggambarkan hubungan antara *entity* yang terlibat berupa aliran-aliran dokumen yang ada. Untuk menjalankan prosedur sistem, digunakan *flowmap* yang terbentuk dari analisis prosedur. Bagan alir dokumen tersebut juga disebut bagan alir formulir yang merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan – tembusannya.

## 2.17 HTML



Gambar Script HTML

Sumber : animamedia.com

HTTP merupakan protokol yang digunakan untuk mentransfer data antara *web server* ke *web browser*. Protokol ini mentransfer dokumen-dokumen web yang ditulis atau berformat HTML. Dikatakan markup karena HTML berfungsi untuk "mepercantik" file text biasa untuk ditampilkan pada program web browser. Hal ini dilakukan dengan menambahkan tag-tag (perintah khusus) pada file teks biasa tersebut.

Tag HTML biasanya berupa tag-tag yang berpasangan dan ditandai dengan simbol `<` dan `>`. Pasangan dari sebuah tag ditandai dengan tanda `'/'`. misalnya awalnya perintah tagnya `<contoh>` maka diakhiri dengan `</contoh>`.

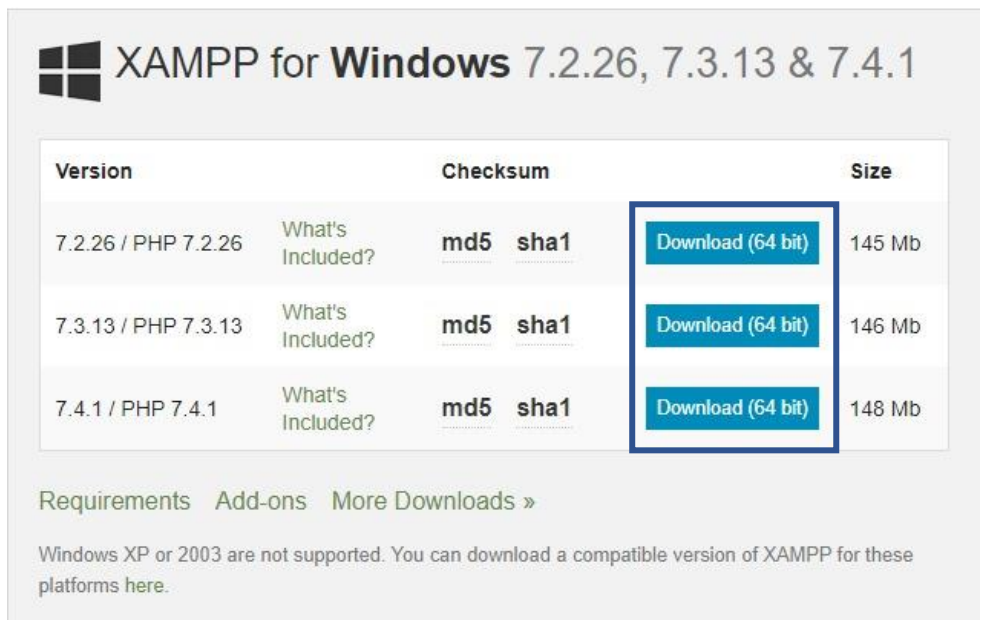
## BAB III

### TATA CARA INSTALASI DAN PENGGUNAAN *TOOLS* YANG DIGUNAKAN

#### 3.1 Xampp

##### 3.1.1 Cara Install Xampp

1. *Download* aplikasi Xampp terbaru , link : <https://www.apachefriends.org/download.html>. Pilih salah satu sesuai dengan yang anda butuhkan :



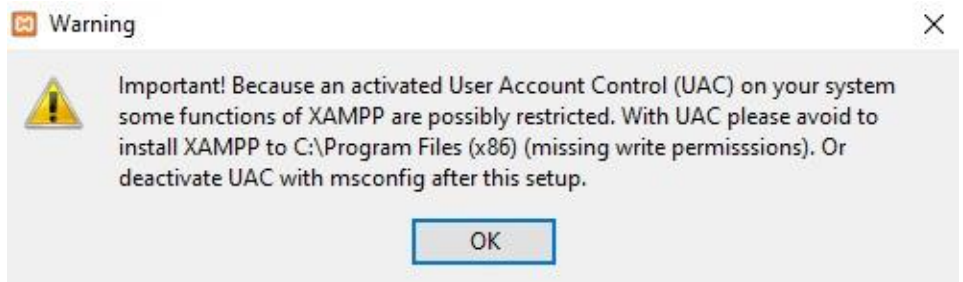
The screenshot shows the XAMPP for Windows download page. It features a table with three rows representing different versions: 7.2.26 / PHP 7.2.26, 7.3.13 / PHP 7.3.13, and 7.4.1 / PHP 7.4.1. Each row includes a 'What's Included?' link, 'md5' and 'sha1' checksum links, a 'Download (64 bit)' button (highlighted with a blue box), and a 'Size' column. Below the table, there are links for 'Requirements', 'Add-ons', and 'More Downloads'. A note at the bottom states: 'Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms [here](#).'

Version	Checksum	Size
7.2.26 / PHP 7.2.26	What's Included? md5 sha1	145 Mb
7.3.13 / PHP 7.3.13	What's Included? md5 sha1	146 Mb
7.4.1 / PHP 7.4.1	What's Included? md5 sha1	148 Mb

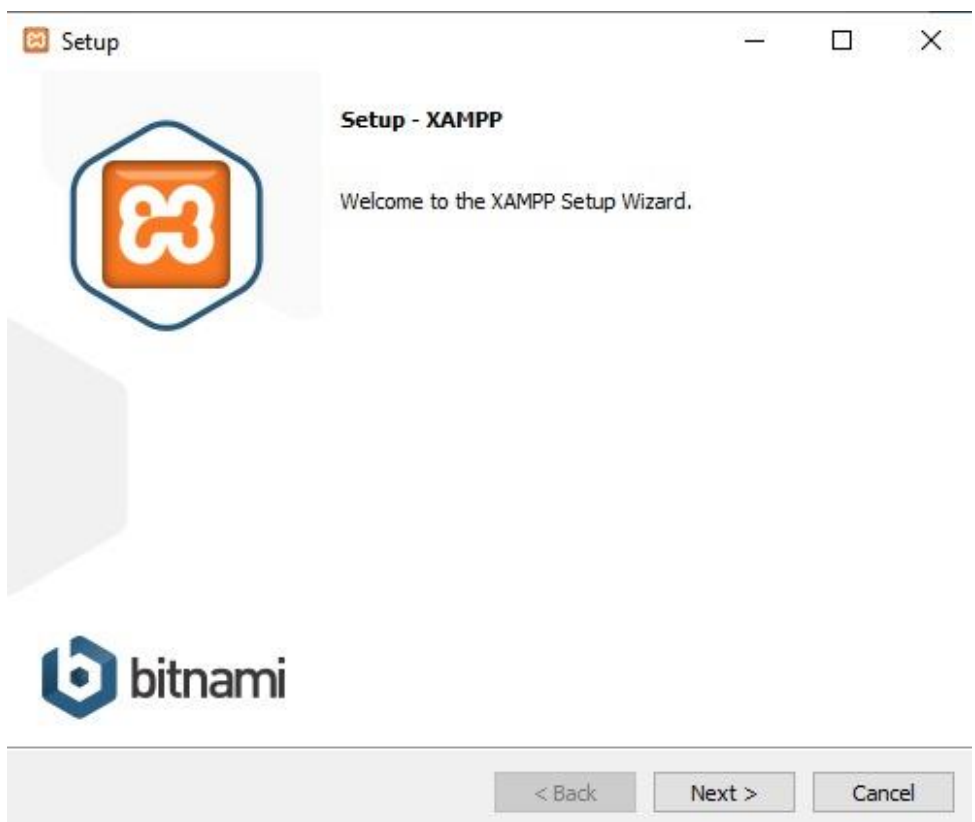
Requirements Add-ons More Downloads »

Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms [here](#).

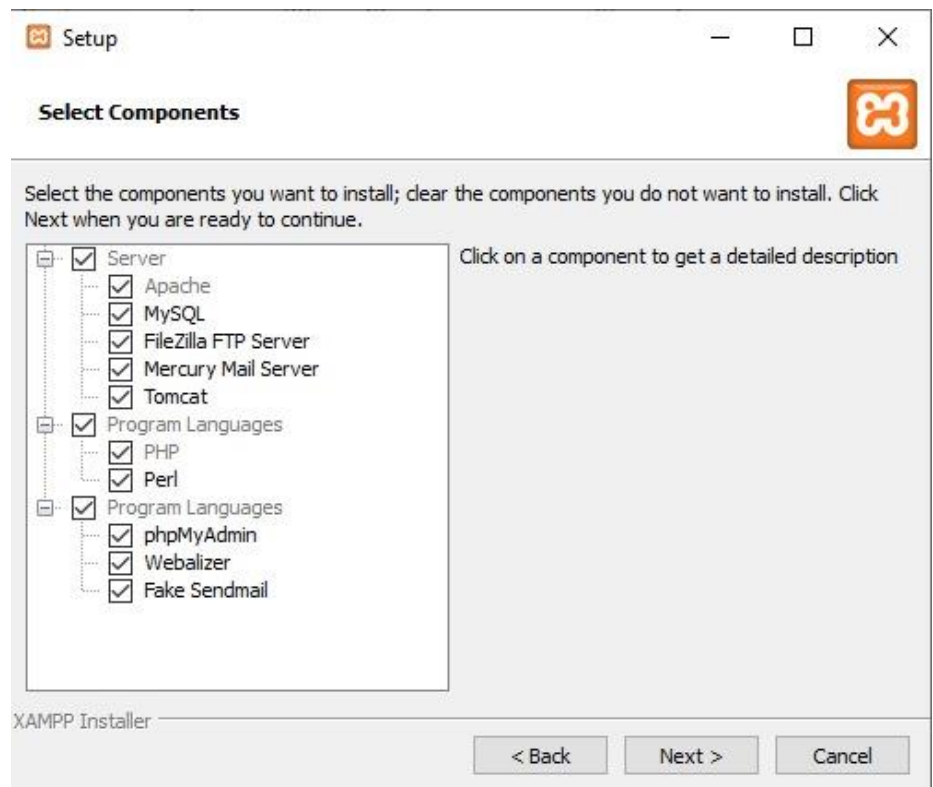
2. Klik dua kali pada file xampp yang baru saja anda *download*, jika muncul pesan *error* seperti gambar berikut abaikan saja dan pilih oke untuk melanjutkan instalasi



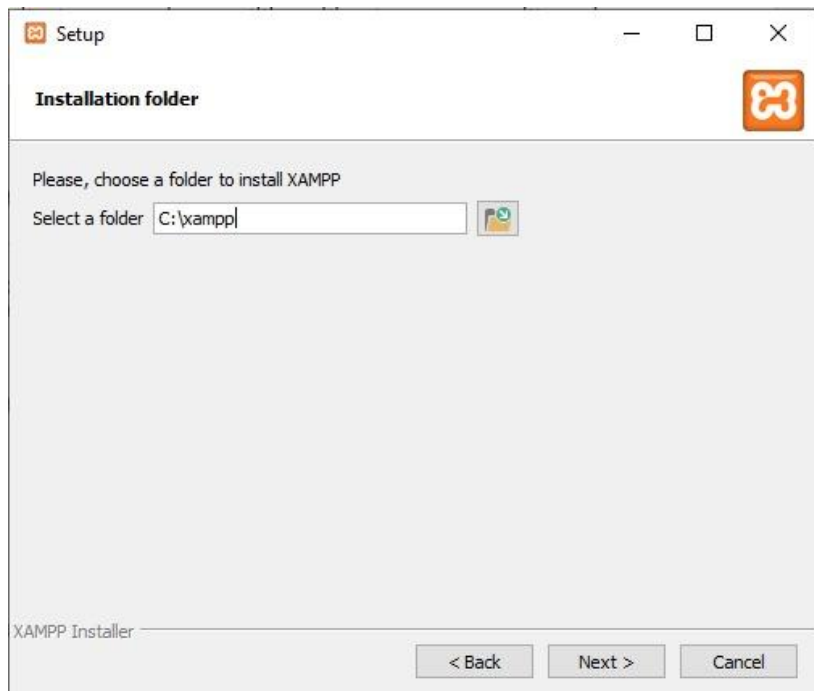
3. Selanjutnya anda akan dialihkan ke jendela yang isinya meminta anda untuk menginstall aplikasi Xampp



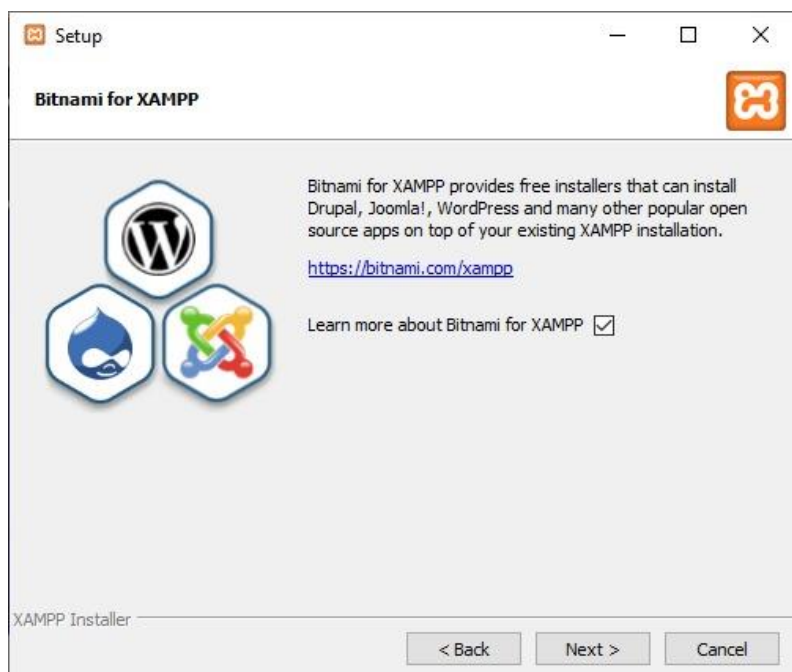
4. Klik next untuk melanjutkan.
5. Selanjutnya anda akan diminta untuk memilih aplikasi yang mau di install. Centang saja semua pilihan dan klik tombol next.



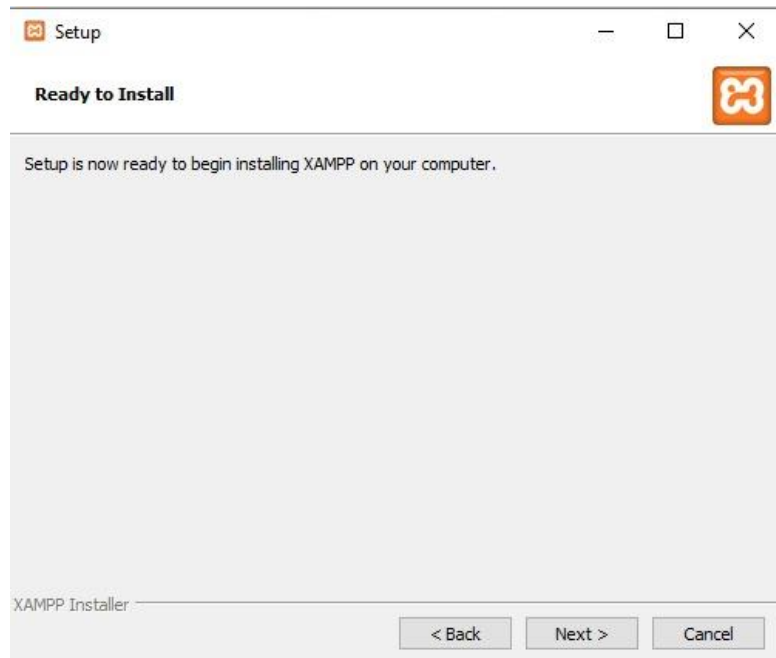
6. Kemudian anda akan diminta untuk menentukan lokasi folder penyimpanan file-file dan folder xampp. Secara default akan diarahkan ke lokasi C:\Xampp.
7. Tetapi jika anda ingin menyimpannya di folder lain anda bisa mengklik *browse* kemudian menentukan secara manual folder yang anda ingin gunakan.
8. Jika sudah selesai lanjutkan dan silahkan klik tombol install.



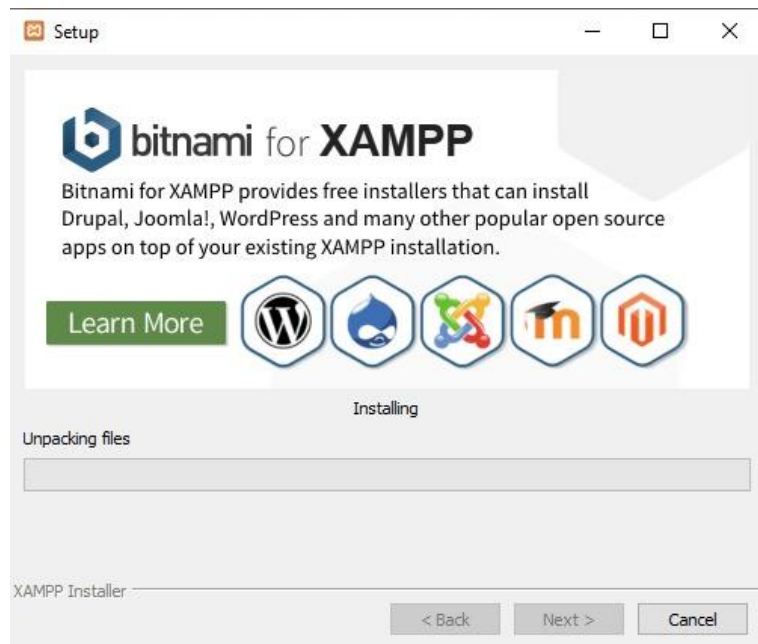
9. Selanjutnya akan muncul tampilan seperti gambit berikut, anda tinggal klik next saja.



10. Selanjutnya akan muncul tampilan bahwa xampp siap untuk di install. Anda tinggal klik *next* saja.

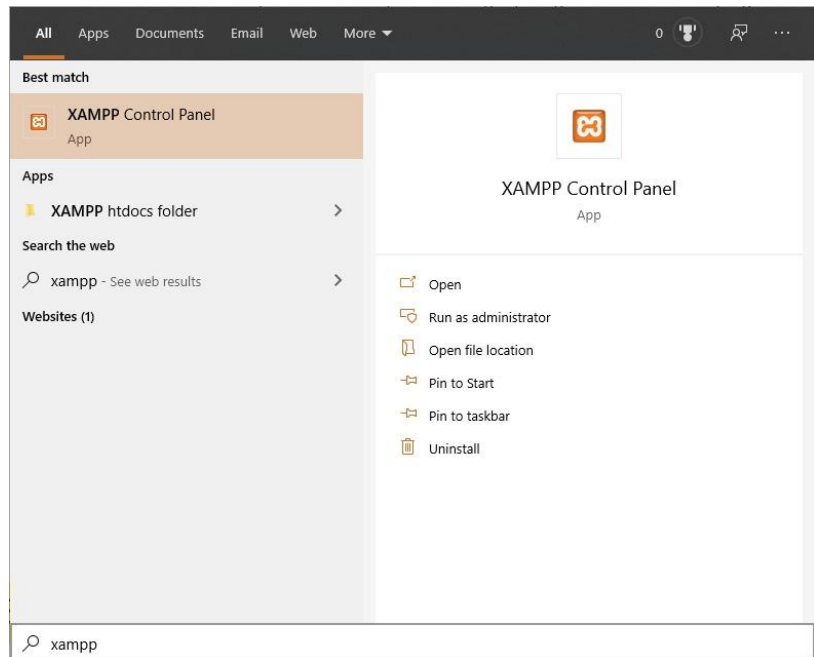


11. Tunggu sampai proses instalasi selesai.

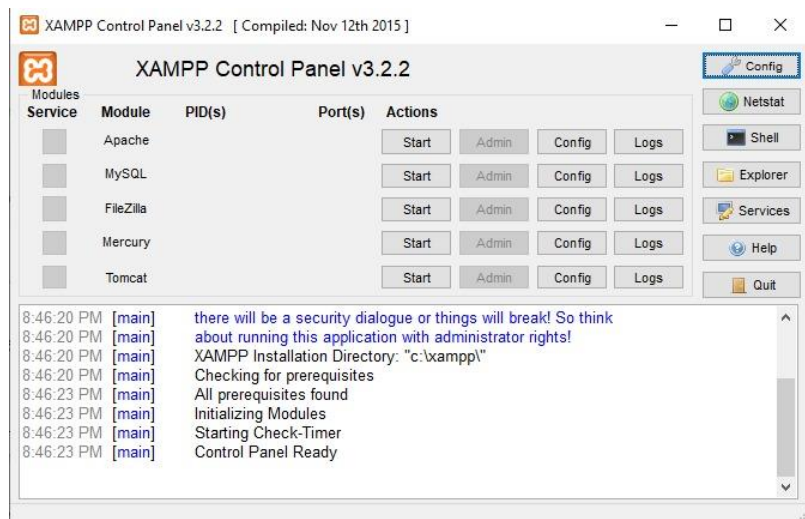


### 3.1.2 Cara Menjalankan Aplikasi Xampp

1. Bulakah aplikasi Xampp bisa melalui pencarian di task bar atau di desktop

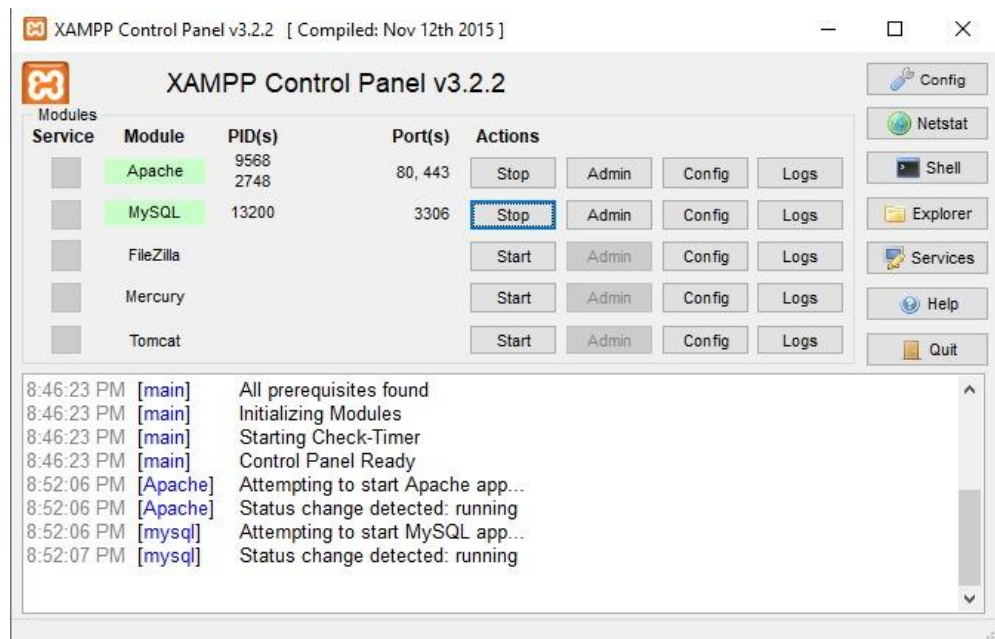


2. Klik dua kali pada xampp tersebut, maka akan muncul tampilan seperti berikut:

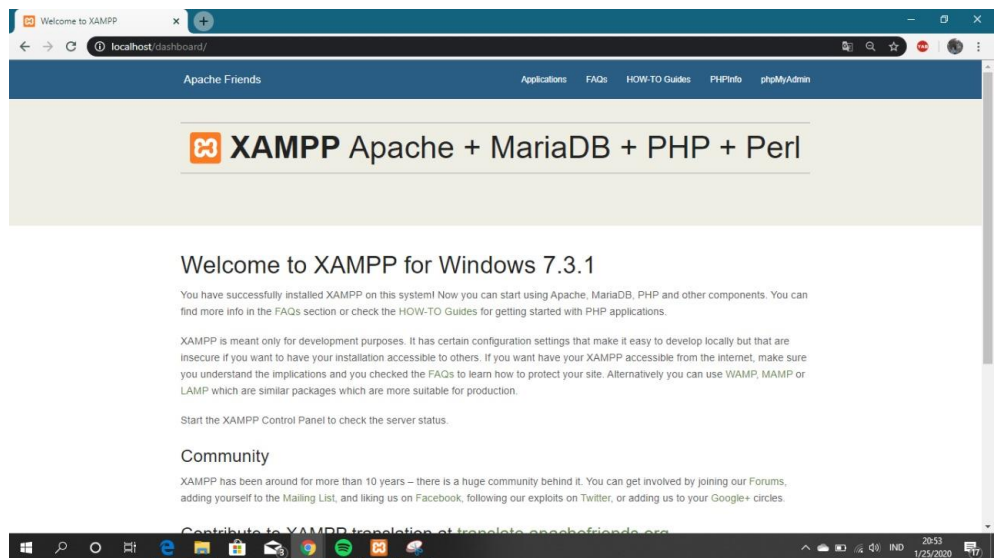




3. Silahkan klik tombol *start* pada kolom *action* sehingga tombol tersebut berubah menjadi *stop*.
4. Dengan mengklik tombol tersebut artinya bahwa aplikasi tersebut telah dijalankan. Biasanya saya menggunakan xampp yang start hanyalah aplikasi Apache dan MySQL karena saya tidak memerlukan aplikasi lainnya.



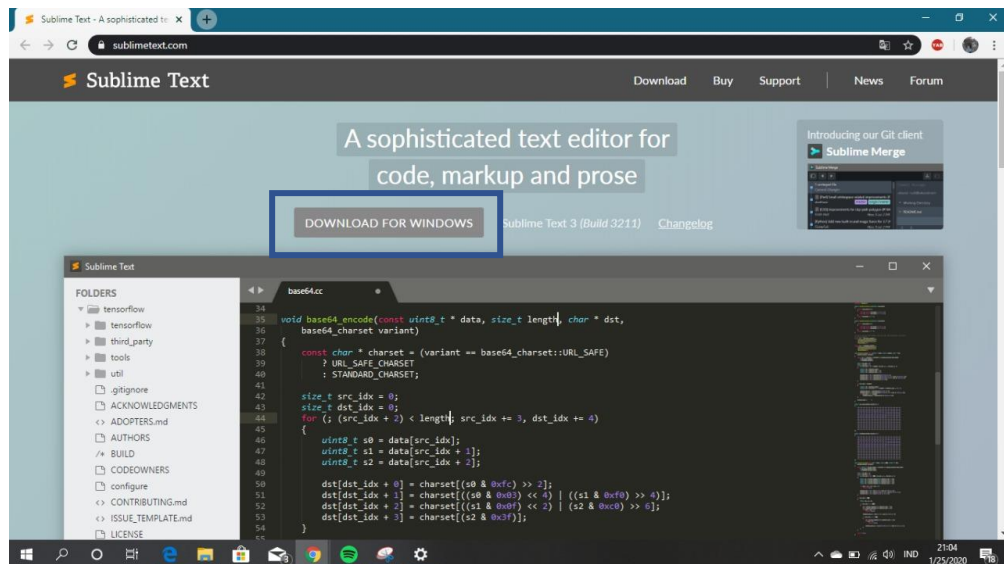
5. Selanjutnya buka browser anda dan cobq ketikkan <http://localhost/> di *address bar*, jika muncul tampilan seperti gambar dibawah ini berarti proses instalasi telah berhasil.



## 3.2 Sublime

1. Download terlebih dahulu file sublime di situs resmi sublime,

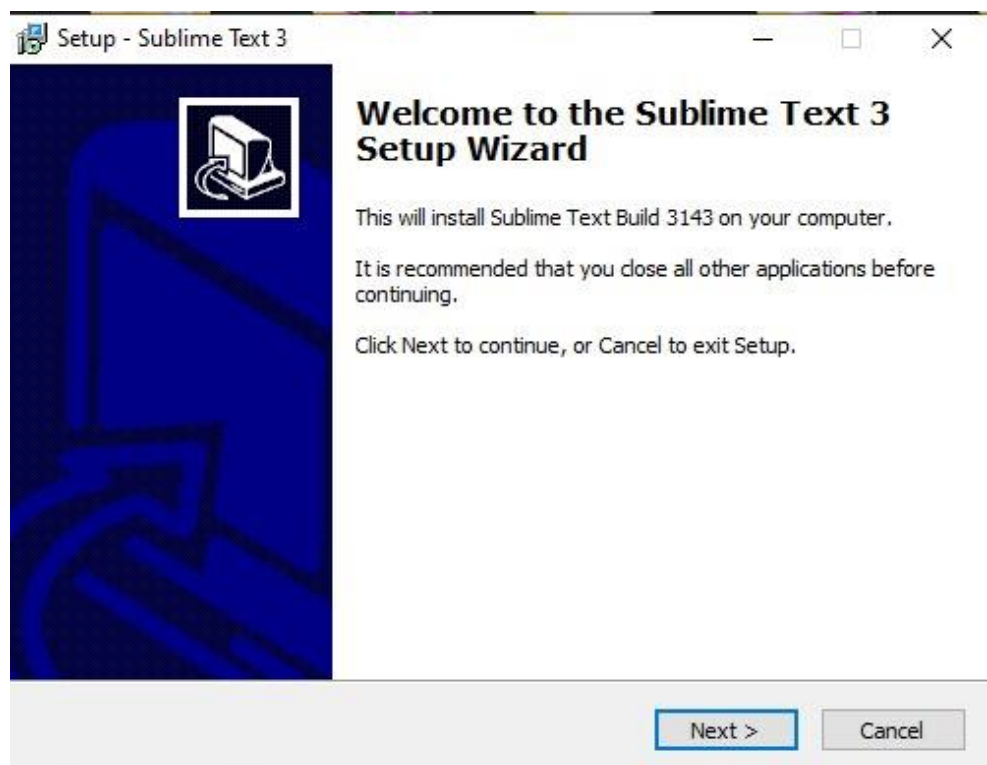
link : <https://www.sublimetext.com/>



2. Setelah selesai di *download* klik dua kali pada file .exe

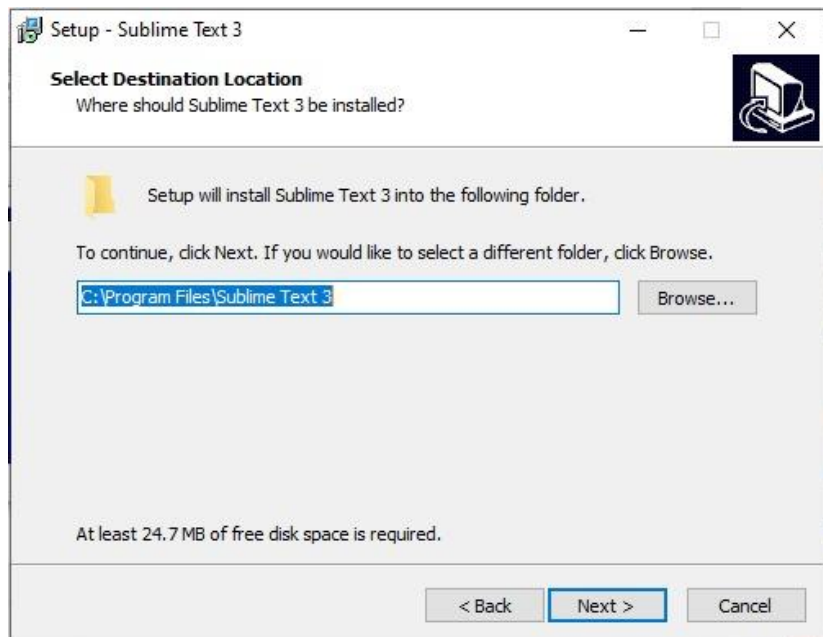


3. Selanjutnya akan muncul tampilan awal untuk proses instalasi.

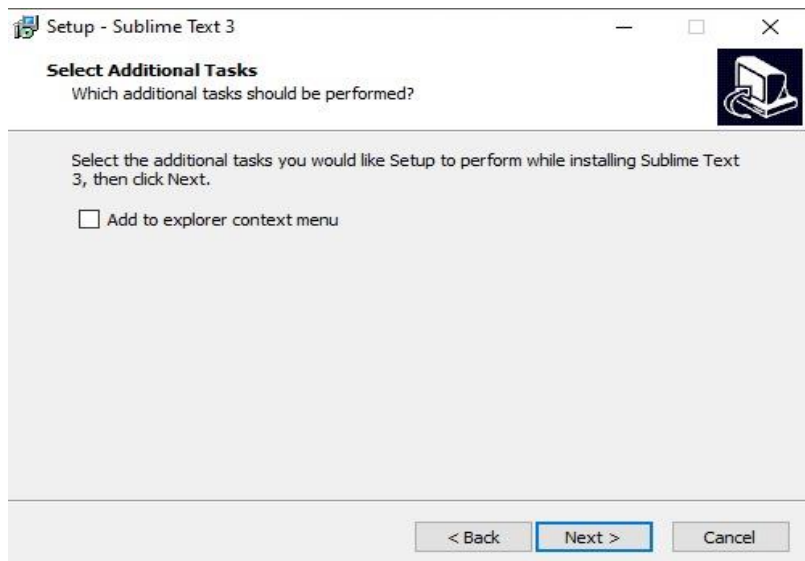


4. Klik next untuk melanjutkan.
5. Setelah itu akan muncul tampilan library penyimpanan sublime. Secara default akan diarahkan ke lokasi C:\Program Files\Sublime Text 3

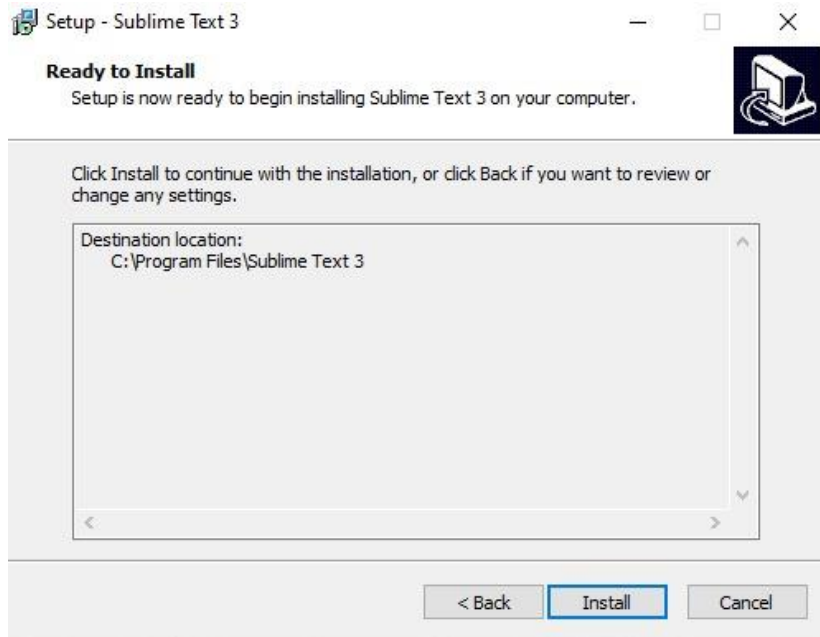
6. Tetapi jika anda ingin menyimpannya di folder lain anda bisa mengklik *browse* kemudian menentukan secara manual folder yang anda ingin gunakan.
7. Jika sudah selesai lanjutkan dan silahkan klik tombol *next*.



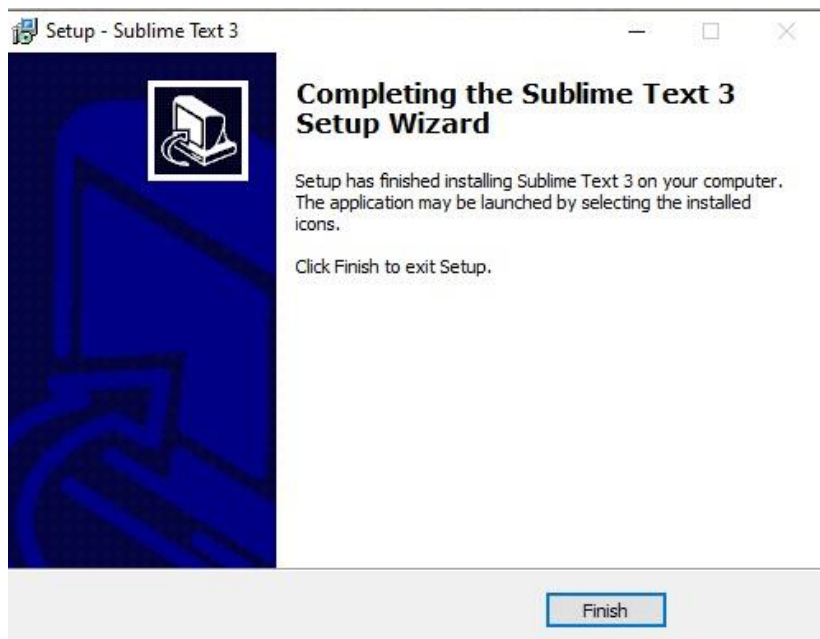
8. Selanjutnya akan muncul tampilan, abaikan dan klik *next*.



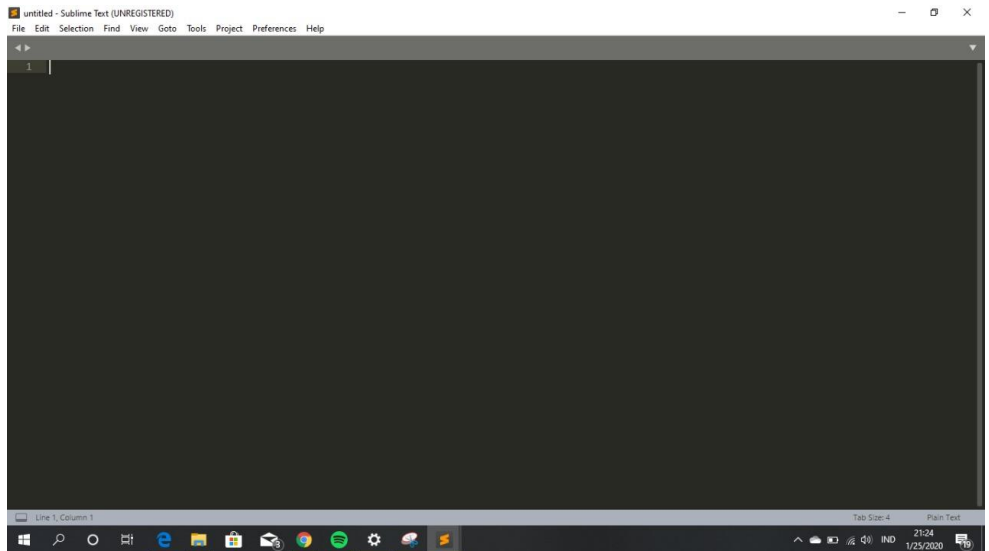
9. Setelah itu klik install untuk melanjutkan proses instalasi, tunggu sampai selesai.



10. Setelah proses instalasi berhasil klik *finish*.

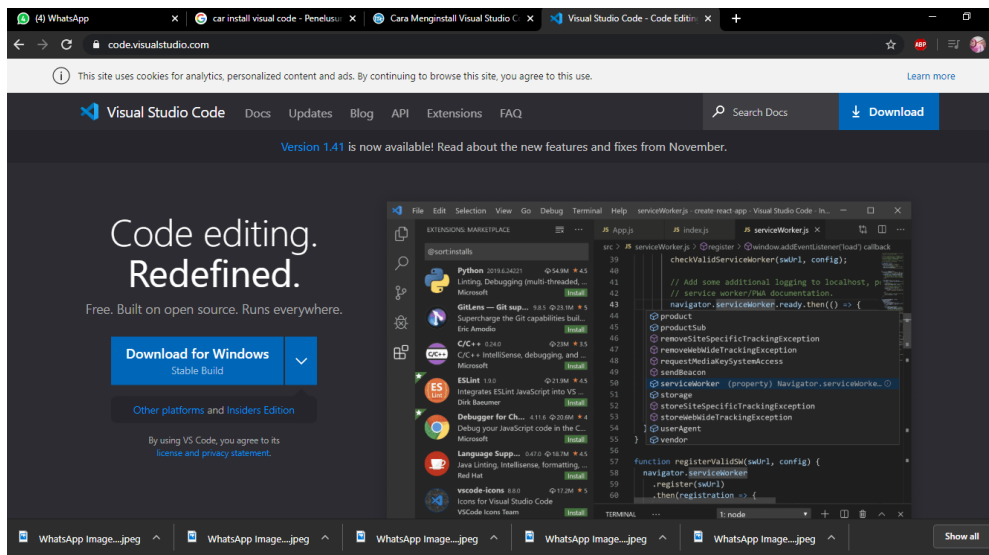


## 11. Tampilan awal sublime

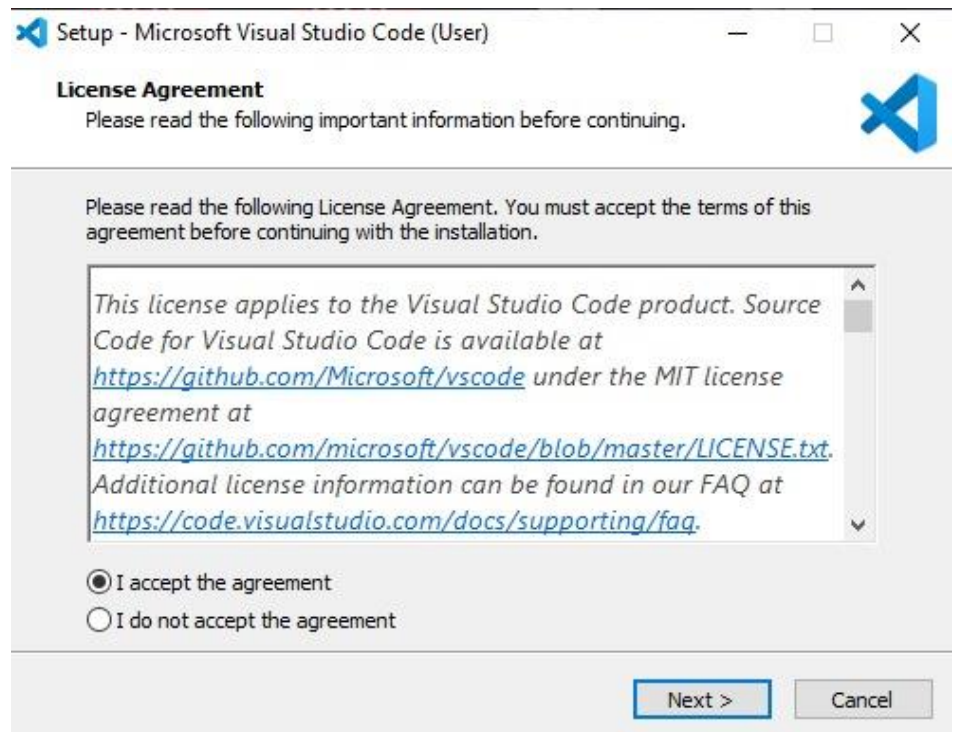


## 3.3 Visual Code

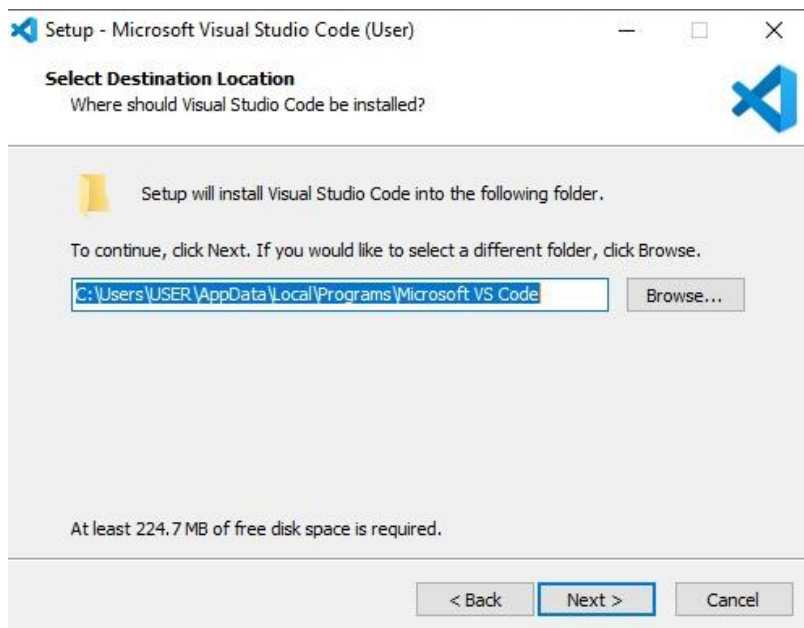
1. Pertama download aplikasi visual code, link <https://code.visualstudio.com/>.
2. Jika link sudah di buka klik download di pojok kanan . seperti pada gambar dibawah ini.



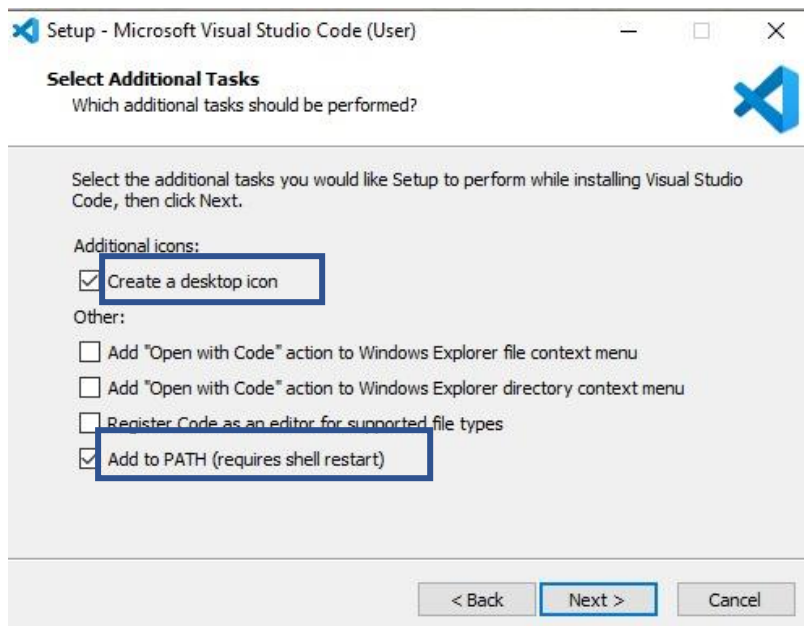
3. Setelah download selesai, buka aplikasi visual code yang telah didownload tadi, maka akan muncul *license agreement*.



4. Pilih "*I accept the agreement*" untuk menyetujui kebijakan visual code, lalu untuk melanjutkan klik *next*.
5. Setelah itu akan muncul tampilan library penyimpanan sublime. Secara default akan diarahkan ke lokasi C:\User\USER\AppData\Local\Programs\Microsoft VS Code
6. Tetapi jika anda ingin menyimpannya di folder lain anda bisa mengklik *browse* kemudian menentukan secara manual folder yang anda ingin gunakan.
7. Jika sudah selesai lanjutkan dan silahkan klik tombol *next*.

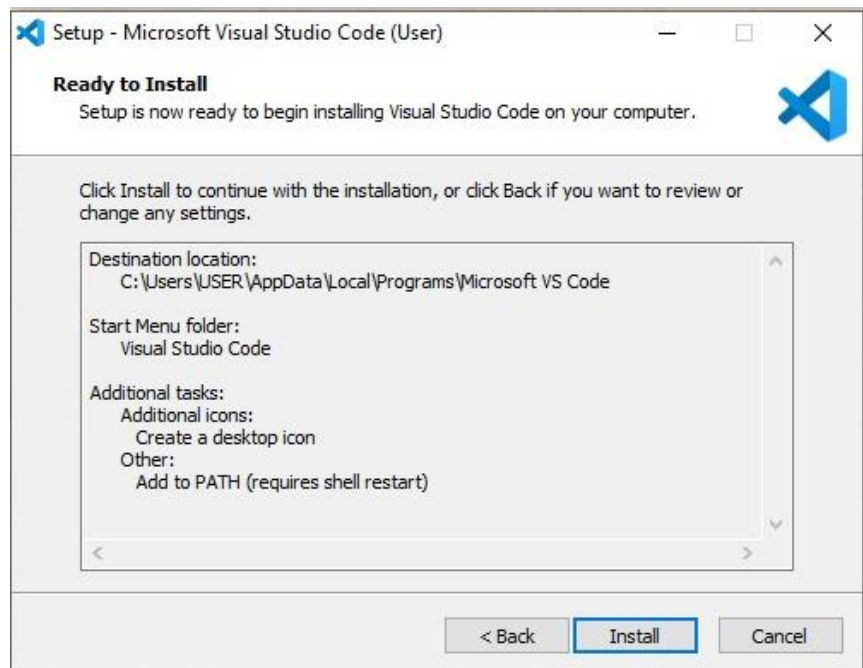


8. Setelah itu akan tampilan seperti pada gambar di bawah ini, ceklis “*Create Dekstop Icon*” jika ingin membuat shortcut VSCode nya kemudian ceklis “*Add to PATH(require shell restart)*” jika sudah klik *Next*.

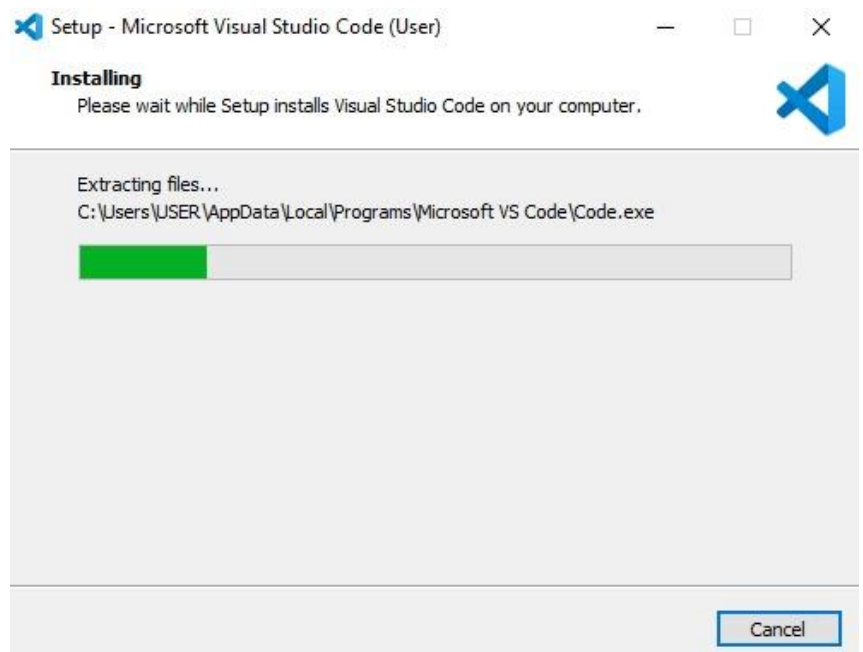




9. Selanjutnya klik *install* untuk melakukan proses instalasi.



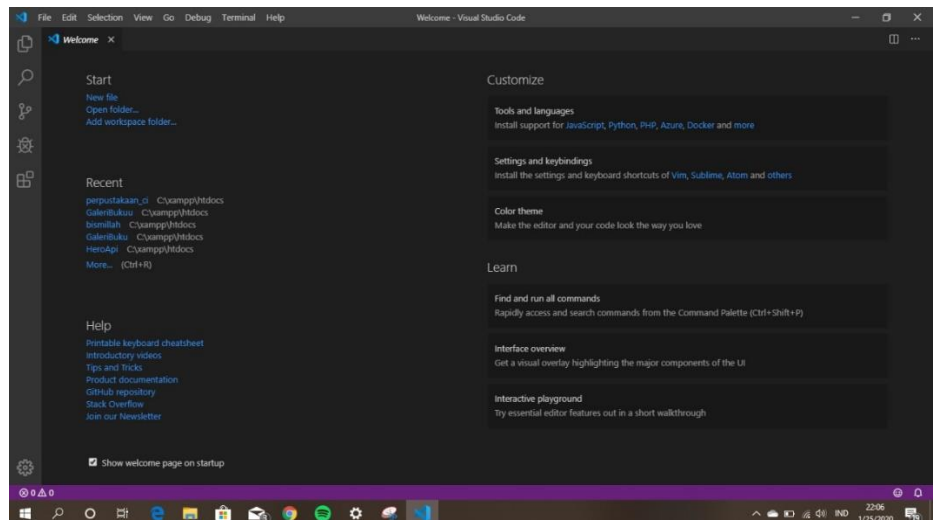
10. Proses instalasi dimulai, tunggu sampai proses selesai.



11. Setelah muncul tampilan seperti gambar dibawah ini, artinya instalasi telah berhasil.



12. Berikut adalah tampilan awal dari visual code.



## **BAB IV**

### **PENJELASAN *FRAMEWORK* DAN BAHASA YANG DAPAT DIGUNAKAN**

#### **4.1 *Framework CodeIgniter***

##### **4.1.1 Pengertian *Framework***

*Framework* adalah suatu kumpulan instruksi yang dikumpulkan dalam kelas dan *function-function* yang memiliki fungsi masing-masing untuk memudahkan *developer* dalam pemanggilan tanoa harus menuliskan *syntax* program yang sama berulang-ulang. Hal ini memiliki kegunaan untuk menghemat waktu dan mencegah penulisan *syntax* secara berulang-ulang agar terciptanya suatu *source code* yang bersih dan terstruktur.

Dalam *framework* sudah menyediakan fungsi *library* dan peralatan lainnya yang kita butuhkan. Salah satu *framework* yang cukup populer di Indonesia adalah *CodeIgniter*.

##### **4.1.2 Pengertian *CodeIgniter***

*CodeIgniter* adalah sebuah *framework* PHP yang bersifat *open source* dan menerapkan salah satu metode MVC (*Model View Controller*). Dalam *CodeIgniter* anda bisa menggunakannya secara gratis atau bersifat *free* dengan kata lain tidak berbayar. *Framework CodeIgniter* dibuat dengan tujuan yang sama seperti *framework* lainnya yaitu; memudahkan *developer* atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuat semuanya dari awal.

*CodeIgniter* dirilis pertama kali pada tanggal 28 Februari 2006. *CodeIgniter* cocok digunakan untuk membuat suatu aplikasi web seperti :

- Portal berita,
- Sistem informasi,
- *Web start up*,
- *Profile company*,
- *E-Commerce*, dan
- *Blog*.

#### **4.1.3 Kelebihan CodeIgniter**

Ada beberapa kelebihan *codeigniter* dibandingkan dengan *framework-framework* yang lain yaitu :

- Performa cepat  
*Framework* ini merupakan *framework* yang paling cepat dibandingkan dengan yang lain, karena tidak menggunakan *template engine* dan ORM yang dapat memperkambat suatu proses.
- Konfigurasi yang minim (*Nearly zero configuration*)  
 Untuk menyesuaikan dengan database dan keleluasaan *routine* tetap diizinkan melakukan konfigurasi dengan mengubah beberapa file seperti *database.php* atau *autoload.php*, namun dalam menggunakan CI hanya menerapkan aturan standar, kita hanya perlu mengubah sedikit saja pada file di folder *config*.
- Memiliki banyak komunitas  
 Di Indonesia komunitas CI lumayan ramai, selain itu tutorialnya pun mudah dicari dan kadang-kadang bisa dijadikan ajang diskusi dalam komunitas tersebut.

- Dokumentasi yang lengkap

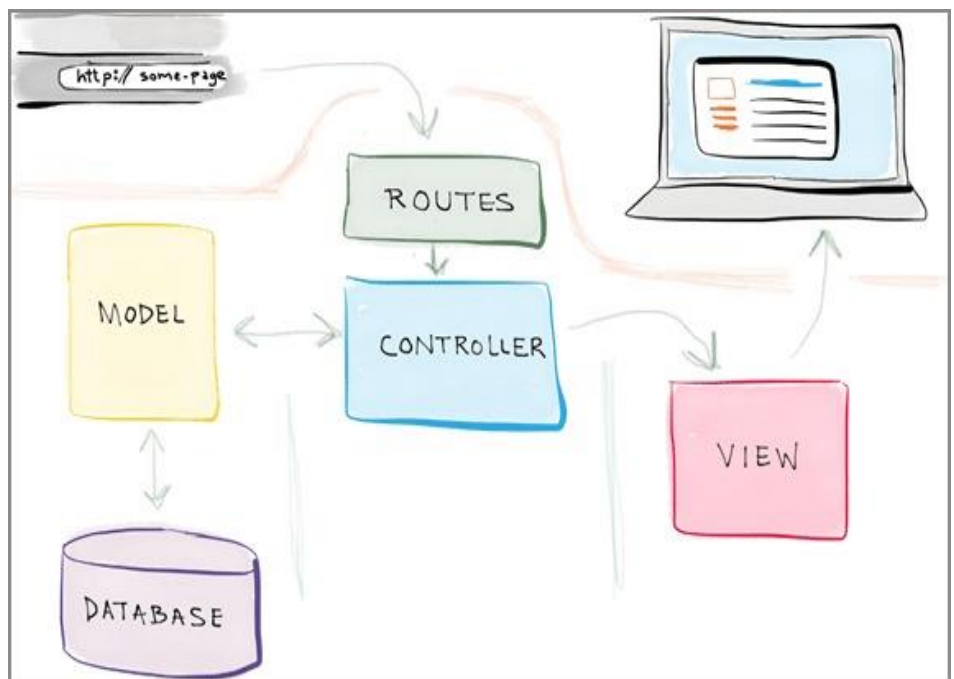
Ketika kita mendownload *CodeIgniter* didalam folder tersebut sudah disertai dengan *user\_guide* yang berisi dokumentasi yang lengkap dan bisa diakses secara *offline*.

- Mudah dipelajari pemula

Bagi pemula *codeigniter* sangat mudah dipelajari karena tidak perlu bergantung pada *tools* tambahan seperti composer ORM *template engine*.

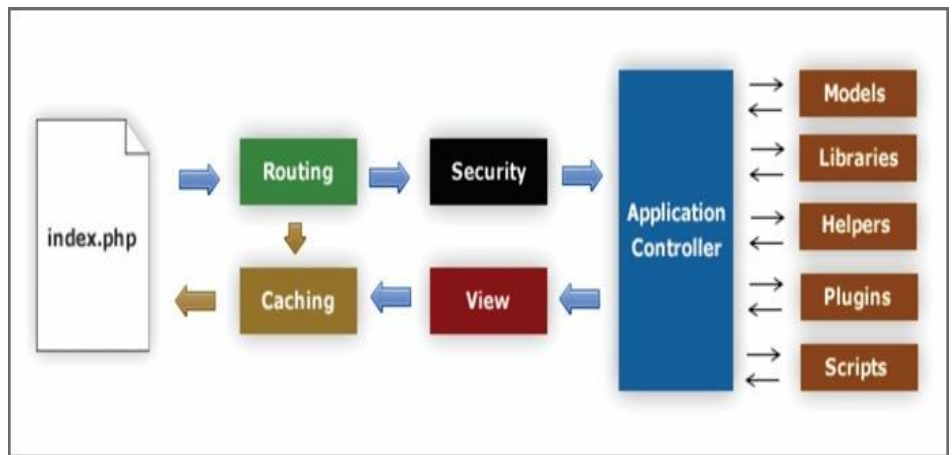
- Metode *Model View Controller*

*Model view controller* atau MVC adalah suatu Teknik atau konsep yang memisahkan komponen utama menjadi tiga komponen yaitu *model*, *view* dan *controller*.



- *Model*, *model* adalah suatu bagian penanganan yang berhubungan dengan pengolahan atau manipulasi database, misalnya mengambil data dari database, *meninput* dari pengolahan database lainnya, sehingga semua instruksi yang berhubungan dengan pengelolaan database diletakkan didalam model.
- *View*, *View* merupakan suatu bagian yang menangani halaman *user interface* atau halaman yang muncul pada *user*. Tampilan dari *user interface* dikumpulkan di *view* untuk memisah dengan *controller* dan *model* sehingga memudahkan *web designer* dalam melakukan pengembangan tampilan suatu halaman *website*.
- *Controller*, *controller* adalah suatu kumpulan instruksi atau aksi yang menghubungkan model dan *view* jadi *user* tidak akan berhubungan dengan model secara langsung begitu juga dengan *view* karena *controller* yang mengelolah instruksi.

Didalam *codeigniter* secara detail konsep MVC digambarkan sebagai berikut:



Pada gambar diatas file *index.php* memiliki peran sebagai *controller* utama yang memanggil fungs-fungsi dasar yang digunakan untuk menjalankan *controller*. *Router* memeriksa *HTTP request* selanjutnya memuttusakn *controller* mana yang akan digunakan untuk menangani *request* tersebut.

Apabila file *cache* tersedia alur aplikasi akan dilewati dan file *cache* tersebut akan dikirimkan ke *browser* pengguna. Sebelum *controller* dipanggil, *HTTP request* dan data yang dikirimkan pengguna akan di sortir terlebih dahulu dengan alasan keamanan. Kemudian *controller* memanggil file model, *library*, *helper* dan file lain yang dibutuhkan untuk menangani *HTTP request*. Hasil akhirnya akan ditampilkan oleh file *view* setelah itu dikirimkan ke *browser* untuk di tampilkan. Jika modus *chacing* diaktifkan hasil *view* akan di *chace* terlebih dahulu. Sehingga jika ada *request* yang sama bisa langsung bisa digunakan.

- Mendukung PHP4 dan PHP5

Saat ini PHP telah mencapai versi 5 bahkan versi tapi masih banyak juga orang-orang yang masih menggunakan PHP4,

oleh karena itu pengembang *framework codeigniter* memperhatikan betul setiap pengguna sehingga *framework codeigniter* dikembangkan agar mampu berjalan baik menggunakan PHP versi apapun

- *Codeigniter* mendukung banyak RDBMS (*Relational Database Management System*) seperti *MySQL*, *MySQLi*, *SQL Server*, *Oracle*, *Maria DB*, *PostgreSQL*, *SQLite*, dan lainnya.
- Mendukung SEO (*Search Engginie Optimization*) karena *codeigniter* pada dasarnya menganut *clean URL* sehingga setiap aplikasi yang dibangun menggunakan *codeigniter* lebih mudah di *index* oleh *search engine* populer seperti google, yahoo, msn dan lainnya.
- *Codeigniter* mengijinkan pengembangan web menggunakan *library* atau *helper* yang tidak disediakan oleh *codeigniter* seperti google map API, Facebook API, Fpdf dan lain sebagainya.
- *Codeigniter* bersifat tidak kaku sehingga memberika kebebasan kepada *developer web* untuk mengembangkan aplikasi berbasis *web* bahkan tanpa *framework*.
- Kompatibilitas dengan *Hosting*  
*Codeigniter* mampu berjalan pada hamper semua *platform hosting* karena *codeigniter* mendukung semua database yang paling umum atau yang sering digunakan.
- Sangat Mudah di Integrasikan  
*Codeigniter* sangat mengerti tentang pengembangan berbagai *library* saat ini. Maka dari itu *codeigniter* memberikan



kemudahan untuk diintegrasikan dengan *library-library* yang tersedia saat ini.

- *Small Footprint*

*Small footprint* yaitu minim jejak, atau lebih sederhananya ketika *user* melihat tampilan website *codeigniter* maka akan sulit ditebak. Hal ini sangat berguna bagi *developer*, ketika ada musuk *attacker* lebih sulit melihat *framework* yang digunakan contohnya saja penggunaan *wordpress* yang dapat langsung dilihat hanya menggunakan meta datanya.

#### **4.1.4 Kekurangan CodeIgniter**

Dalam *codeigniter* tidak hanya memiliki kelebihan saja sama seperti *framework* lainnya *codeigniter* juga memiliki kekurangan yaitu diantaranya:

- *Codeigniter* tidak ditujukan atau disarankan untuk pembuatan web dengan skala yang besar.
- *Library* yang sangat terbatas. Hal ini dikarenakan sangat sulit mencari *pluggin* tambahan yang terverifikasi secara resmi, karena pada situsnya sendiri tidak menyediakan *plugin-plugin* tambahan untuk mendukung pengembangan aplikasi dengan CI.
- Belum memiliki editor khusus *codeigniter* sehingga dalam membuat suatu *project* dan modul-modulnya harus berpindah-pindah antara satu folder dengan folder yang lainnya.
- *Codeigniter* dikembangkan oleh Ellislab yang dimana bukanlah suatu komunitas sehingga menyebabkan *update core engine* nya tidak secepat *framework* yang lain.

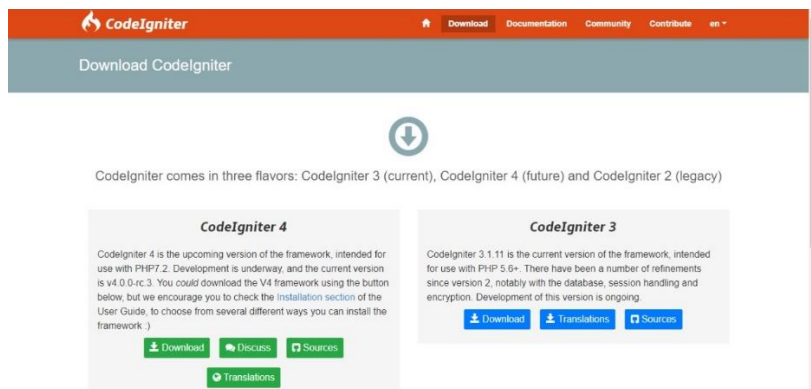
- Memiliki banyak kelonggaran dalam hal *coding* contohnya bebas menambahkan file.
- *Codeigniter* tidak mencerminkan MVC yang sesungguhnya, contohnya penulisan *echo* masih dapat dilakukan pada file *controller*.

Dari penjelasan MVC diatas dapat di simpulkan bahwa *controller* sebagai penghubung *view* dan model. Contohnya pada aplikasi yang menampilkan data dengan menggunakan metode MVC, *controller* memanggil instruksi pada model yang mengambil data pada database, kemudian *controller* yang meneruskannya pada *view* untuk ditampilkan.

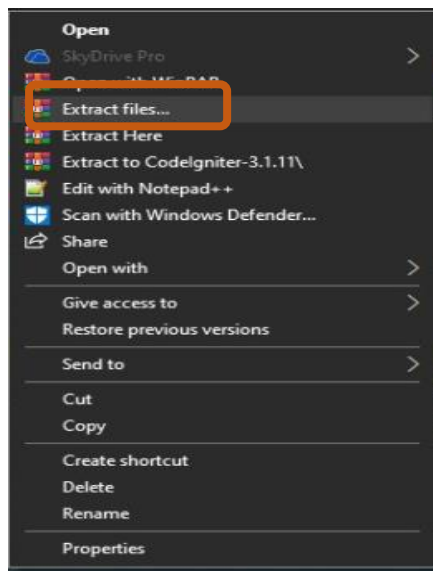
Jadi sudah jelas bahwa dalam pengembangan aplikasi dengan menggunakan konsep MVC ini sangat mudah dan membantu *web designer* atau *frontend developer* tidak perlu lagi berhubungan dengan *view* untuk mendesign tampilan aplikasi, sedangkan bagian *backend developer* mereka yang menangani bagian *controller* dan modelnya. Jadi pembagian tugas pun menjadi pun dan pengembangan aplikasi dapat dilakukan dengan cepat.

#### **4.1.5 Cara Menggunakan CodeIgniter**

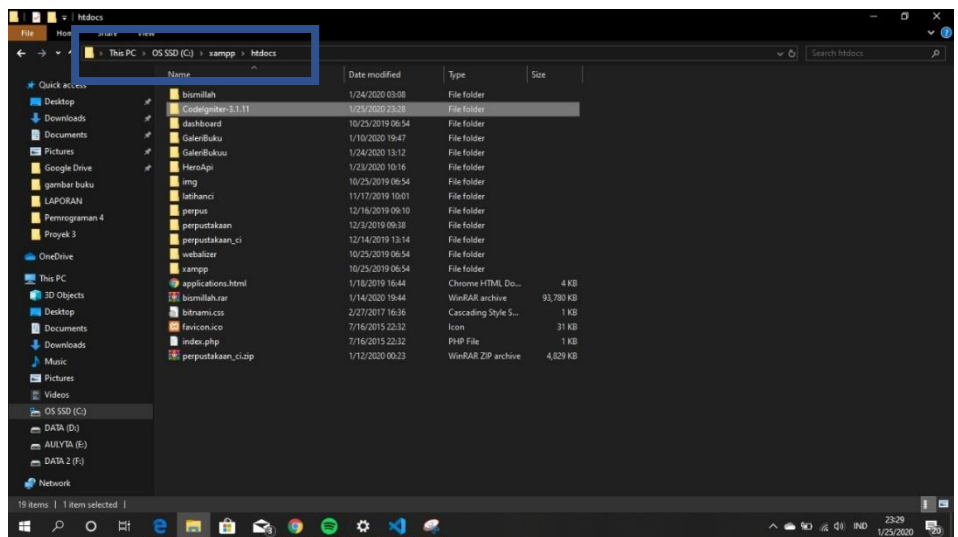
1. Download terlebih dahulu *codeigniter* di link <https://codeigniter.com/download>
2. setelah link tersebut di buka , pilih sesuai dengan kebutuhan masing masing



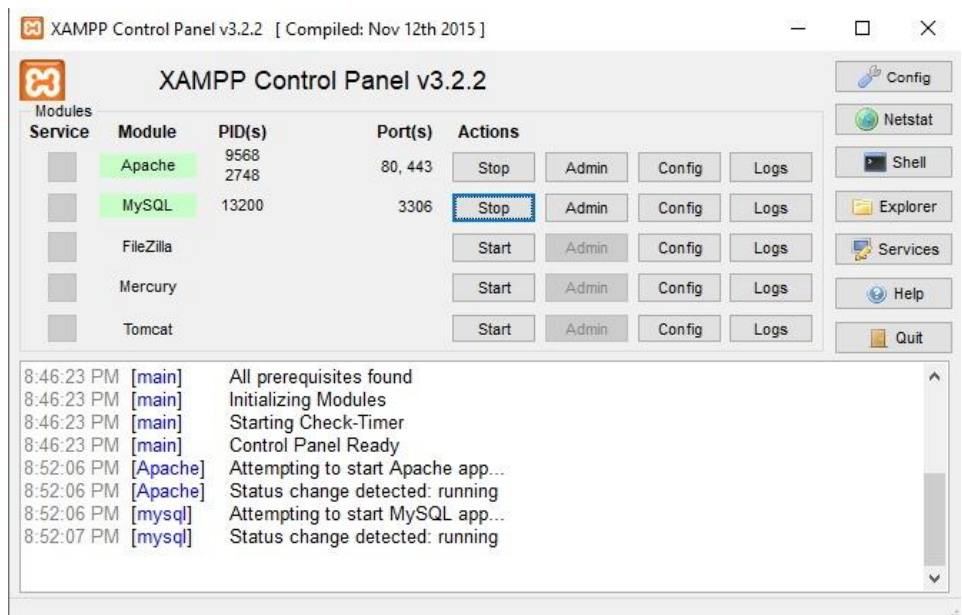
3. kemudian *extract* file yang telah berhasil di *download*.



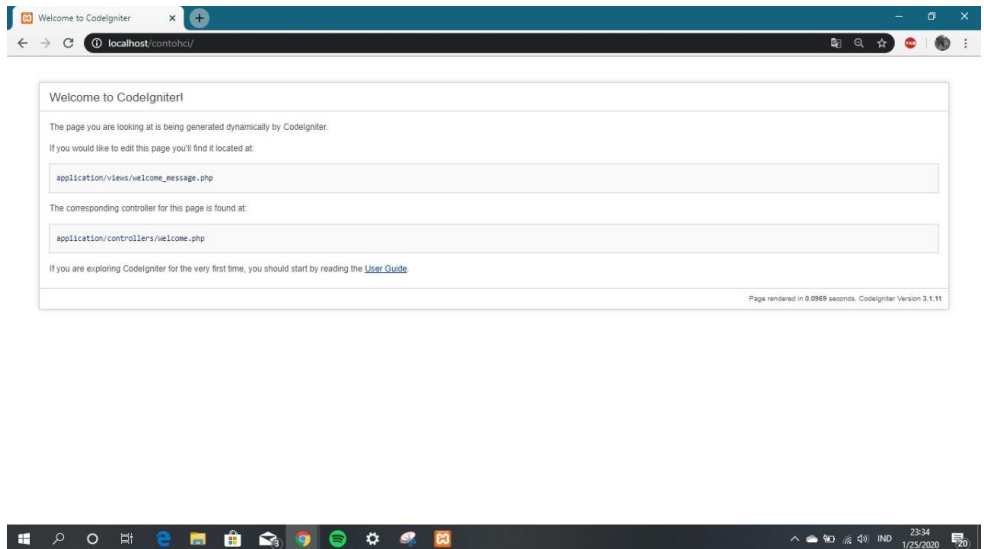
4. Setelah itu pindahkan folder yang telah di *extract* tadi kedalam folder htdocs (C:\xampp\htdocs)



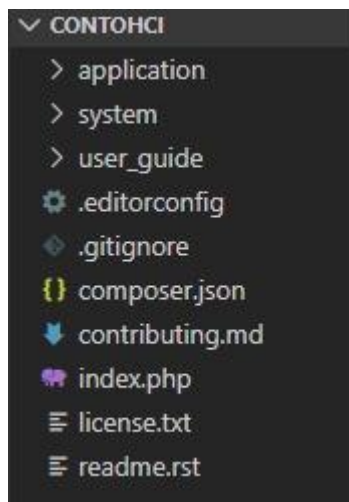
5. Kemudian ubah nama file tersebut sesuai dengan yang anda inginkan, disini saya menggunakan nama file contohCI.
6. Setelah itu buka aplikasi Xampp dan jalankan xampp dengan mengklik pada bagian *actions* sampai berubah menjadu *stop*.



7. Selanjutnya buka *browser* dan ketikkan <http://localhost/contohci/>

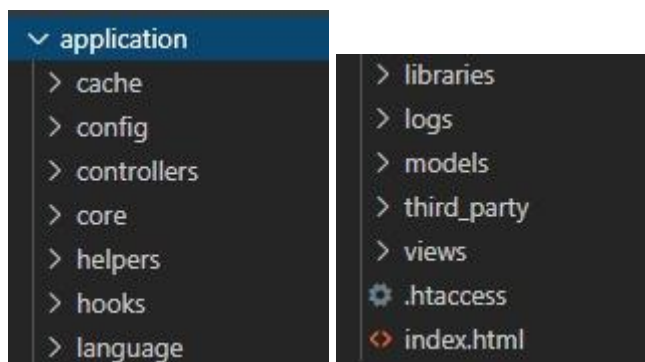


8. Selamat *codeigniter* berhasil di install.
9. Setelah itu buka VisualCode yang telah di install tadi, hal ini dilakukan untuk mengenali struktur direktori pada *codeigniter*.
- Berikut adalah struktur direktori *codeigniter*



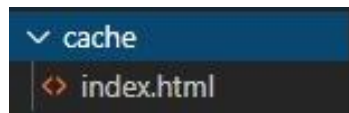
Terdapat dua direktori penting didalam CI yaitu folder *application* dan folder *System* . selain juga terdapat direktori folder *user\_guide* dan beberapa file. Berikut adalah beberapa penjelasan yang terdapat dalam CI:

- Folder *Application*

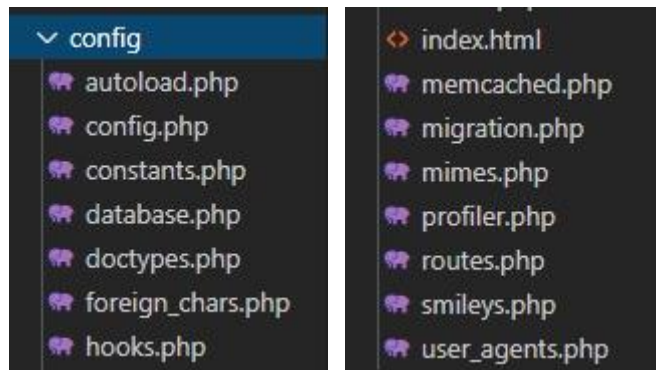


Folder *application* berisi code aplikasi. Disinilah kita akan menulis semua baris kode aplikasi yang akan kita bangun. Dalam folder *application* terdapat beberapa direktori yaitu:

- Folder *chace*, adalah suatu folder yang berisi *chace* dari aplikasi.

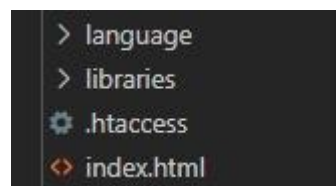
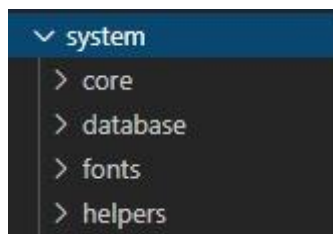


- Folder *config*, adalah suatu folder yang berisi konfigurasi aplikasi.



- Autoload.php*, merupakan suatu tempat yang mendefinisikan *autoload*.
- Config.php*, adalah suatu file untuk mengkonfigurasi aplikasi.
- Constants.php*, merupakan suatu file yang berisi konstanta.
- Database.php*, merupakan suatu file yang berisi konfigurasi database aplikasi.
- Doctypes.php*, adalah suatu file yang berisi definisi untuk *doctype html*.
- Foreign\_chars.php*, merupakan suatu file yang berisi *character* dan *symbol*.
- Hooks.php*, merupakan suatu file yang berisi konfigurasi *hooks*.
- Index.html*, merupakan suatu file untuk mencegah *direct access*.

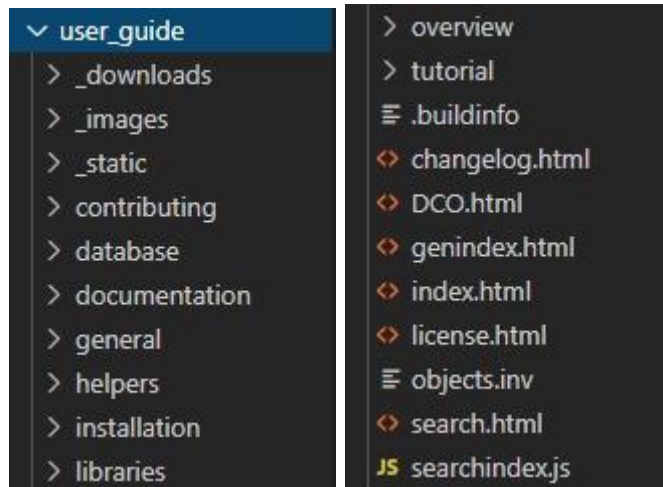
- i. *Memchaced.php* , merupakan suatu file yang berisi suatu konfigurasi untuk *menchace*.
  - j. *Migration.php*, merupakan suatu file yang berisi untuk migrasi.
  - k. *Mimes.php*, merupakan suatu file yang berisi definisi type file
  - l. *Profiler.php*, merupakan suatu file yang berisi konfigurasi untuk *profiler*.
  - m. *Routes.php*, merupakan suatu file yang digunakan sebagai tempat untuk menulis *route* aplikasi.
  - n. *Smileys.php*, berisi kode untuk emoji.
  - o. *User\_agents.php*, merupakan suatu file yang berisi definisi untuk *user\_agents*.
- Folder *System*, Folder ini berisi kode-kode inti dari *codeigniter*. Perlu diingat bahwa janganlah anda mengubah apapun didalam direktori ini.jika anda ingin *menuprage* versi ke yang baru anda cukup *mereplace* direktori tersebut dengan yang baru.



- Folder *User-guide*, Folder ini berisi dokumentasi lengkap dari *codeigniter*, dengan folder ini kita bisa mengakses



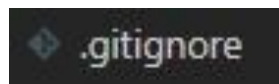
dokumentasi CI secara *offline*. Anda bisa menghapus direktori ini saat *web* yang anda buat telah jadi.



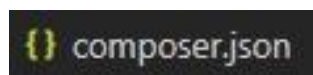
- File *.editorconfig*, File ini berisi konfigurasi untuk *text* editor.



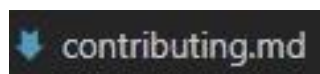
- File *.gitignore*, File ini berisi daftar file dan folder yang akan diabaikan oleh git.



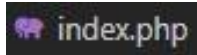
- File *Composer .json*, File ini adalah suatu yang berisi keterangan project dan keterangan *library* yang digunakan, selain itu file ini dibutuhkan oleh composer



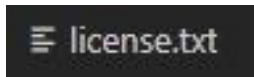
- File *Contributing.md*, file ini adalah suatu file yang berisi penjelasan tentang cara berkontribusi diproyek CI. File ini dapat dihapus apabila *web* sudah jadi.



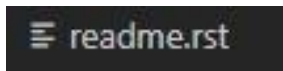
- *File Index.php*, file ini adalah suatu file utama dari *codeigniter* yang merupakan file yang akan dibuka pertama kali saat kita mengakses *web*.

A screenshot of a file icon for 'index.php' in a code editor. The icon features a purple elephant logo to the left of the text 'index.php'.

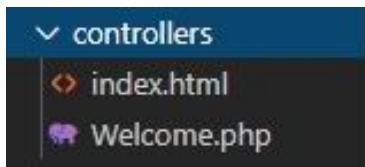
- File *license.txt*, file ini berisi keterangan lisensi dari *codeigniter*

A screenshot of a file icon for 'license.txt' in a code editor. The icon features a purple elephant logo to the left of the text 'license.txt'.

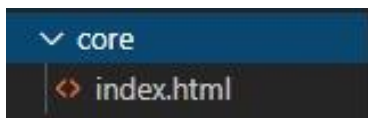
- File *Readme.rst*, File ini sama seperti dengan file *contributing.md*. file ini berisi penjelasan dan informasi tentang proyek *codeigniter*. File ini dapat dihapus saat *web* sudah selesai.

A screenshot of a file icon for 'readme.rst' in a code editor. The icon features a purple elephant logo to the left of the text 'readme.rst'.

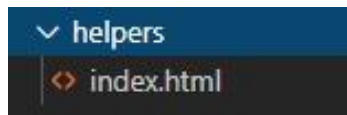
- Folder *Controller*, merupakan suatu folder yang berisi kode-kode *controller*.

A screenshot of a file explorer showing a folder named 'controllers'. The folder is expanded, revealing two files: 'index.html' with a code icon and 'Welcome.php' with a purple elephant icon.

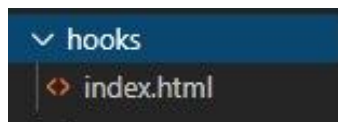
- Folder *Core*, merupakan suatu folder yang berisi kode-kode untuk *custom core*.

A screenshot of a file explorer showing a folder named 'core'. The folder is expanded, revealing one file: 'index.html' with a code icon.

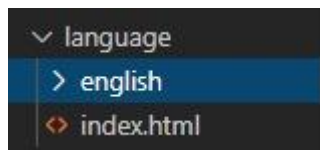
- Folder *Helpers*, merupakan suatu folder adalah fungsi-fungsi helper.



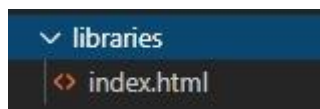
- Folder *Hooks*, merupakan suatu folder yang berisi kode untuk *script hook*.



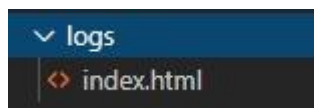
- Folder *language*, merupakan suatu folder yang berisi *string* untuk bahasa apabila web mendukung multi bahasa.



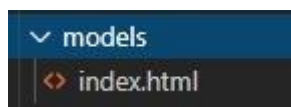
- Folder *Libraries*, merupakan suatu folder yang berisi *library*.



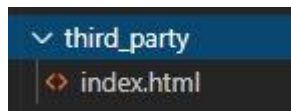
- Folder *Logs*, merupakan suatu folder yang berisi *logs* dari aplikasi.



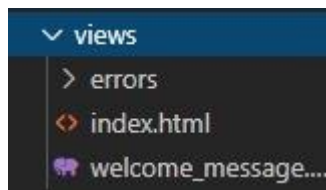
- Folder *models*, berisi kode-kode untuk model



- Folder *Third\_party*, merupakan suatu folder yang berisi *library* dari pihak ketiga

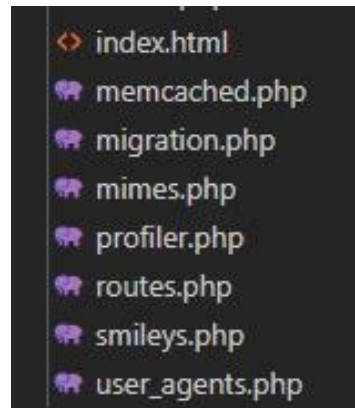
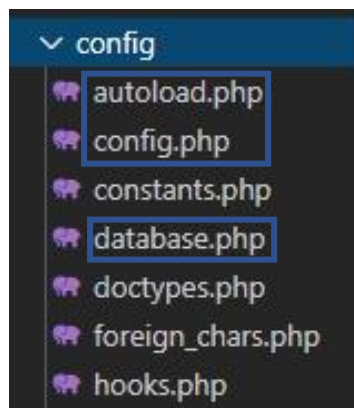


- Folder *Views*, merupakan suatu folder yang berisi kode suatu *view*.



#### 4.1.6 Konfigurasi Dasar *CodeIgniter*

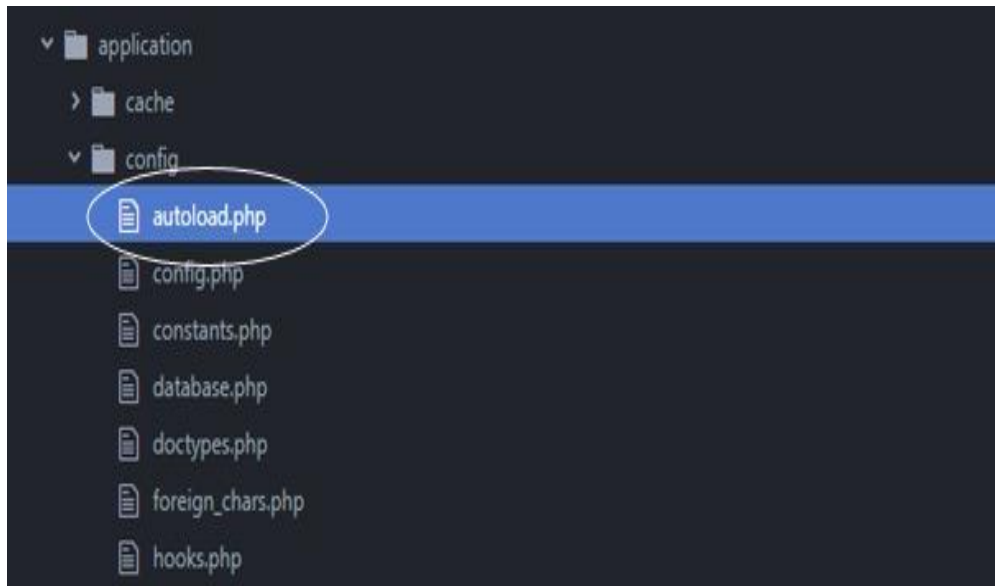
Dalam memulai *codeigniter* ada beberapa konfigurasi yang harus dilakukan pada beberapa file yaitu file *autoload.php*, file *config*, dan *database.php*. Semua konfigurasi tersebut terletak didalam folder *application/config*.



- *Autoload.php*

Pada file ini digunakan untuk mengatur fungsi-fungsi yang akan di muat secara otomatis diawal ketika suatu program dijalankan. Untuk melakukan konfigurasi pada file

*autoload.php* silahkan buka folder *application/config/autoload.php*.



Ada beberapa hal yang bisa di *load* secara otomatis dalam *codeigniter* yaitu *package*, *libraries*, *drivers*, *helper files*, *custom config files*, *language files*, dan *models*.

Untuk konfigurasi dasar yang perlu anda ketahui adalah *libraries* dan *helper files* hal tersebut memiliki tujuan agar beberapa *library* dan *helper* tertentu berjalan otomatis.

Untuk melakukan konfigurasi pada *libraries* buka file *autoload.php* dengan *text editor* kemudia temukan kode berikut :

```
$autoload['libraries'] = array();
```

Setelah itu atur menjadi seperti gambar berikut:

```
$autoload['libraries'] = array('database');|
```

Pada kode diatas artinya kita akan *meload library database* secara otomatis, sehingga anda dapat menggunakan fungsi-fungsi database pada *codeigniter*.

Selanjutnya kita akan melakukan konfigurasi pada *helper files*, kemudian temukan kode berikut:

```
$autoload['helper'] = array();
```

Setelah itu atur menjadi seperti gambar berikut:

```
$autoload['helper'] = array('url');
```

Pada kode diatas artinya kita akan *meload helper URL* secara otomatis. Dengan demikian anda dapat menggunakan fungsi-fungsi URL pada *codeigniter*, misalnya fungsi *base\_url()*, *site\_url()*, *URI Segment*, dan sebagainya.

- *Config.php*

Pada file ini terdapat beberapa konfigurasi yang secara standar sudah terkonfigurasi, namun terdapat beberapa konfigurasi yang perlu diperhatikan:

- *Base\_url*, merupakan url dasar dari *project* anda. Temukan kode berikut:

```
$config['base_url'] = '';
```

Kemudian atur menjadi gambar berikut:

```
$config['base_url'] = 'http://localhost/contohci/';
```

- *Database.php*

Dilihat dari nama filenya anda pasti sudah mengetahui apa fungsi dari file ini. File *database.php* berfungsi untuk melakukan konfigurasi yang berkaitan dengan konfigurasi database dari *website* yang akan di buat. Konfigurasi yang perlu diperhatikan tersebut diantaranya: *hostname*, *username*, *password*, dan *database*.

Untuk melakukan konfigurasi pada *database.php* buka file *database.php* dengan *text* editor.



Kemudian temukan kode berikut :

```

$active_group = 'default';
$query_builder = TRUE;

$db['default'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => '',
    'password' => '',
    'database' => '',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);

```

Setelah itu ubah kode tersebut seperti gambar dibawah ini :

```

$active_group = 'default';
$query_builder = TRUE;

$db['default'] = array(
    'dsn' => '',
    'hostname' => 'localhost',
    'username' => 'root',
    'password' => '',
    'database' => 'contohci',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);

```



## 4.2 Bahasa Pemrograman Yang Dapat Digunakan

### 4.2.1 PHP (*Hypertext Preprocessor*)

PHP adalah singkatan dari *Hypertext Preprocessor* yang merupakan bahasa pemrograman yang bersifat *open source* yang berada didalam server yang diproses di server. PHP salah satu bahasa pemrograman yang memiliki script yang terintegrasi dengan HTML dan berada pada server (*server side HTML embedded scripting*).[9]

PHP merupakan bahasa pemrograman pelengkap HTML yang memungkinkan aplikasi web dinamis untuk pengolahan data, pemrosesan data dari *user via form*, membuat buku tamu, toko online, dan lain sebagainya. Dengan mudah PHP dapat melakukan koneksi ke database karen PHP memang dilengkapi fitur yang memungkinkan koneksi ke PHP dilakukan dengan mudah tanpa harus memusingkan (Tim EMS, 2016:55).

PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya akan dijalankan diserver tetapi disertakan pada dokumen HTML. Pembuatan web ini merupakan kombinasi antara php sendiri sebagai bahasa pemrograman dan HTML sebagai pembangun halaman web (Bimo sunarfrihantono, ST 2002:9).



*Gambar Hypertext Preprocessor*

Sumber : <https://www.webhozz.com/>

Menurut Andi Pramono dan M. Syafii (2004 : 2), PHP atau kependekan dari *Hypertext Preprocessor* adalah sebuah bahasa pemrograman berbasis *web* yang mempunyai banyak keunggulan dibandingkan dengan bahasa pemrograman berbasis *web* yang lain. PHP merupakan bahasa pemrograman yang bersumber dari Perl. Sedangkan Perl merupakan pengembangan dari bahasa C.

Oleh karenanya, struktur pemrograman yang ada di PHP sama dengan yang ada di bahasa C. Melihat bahwa PHP merupakan pengembangan dari bahasa C secara tidak langsung, maka PHP mempunyai banyak sekali fitur-fitur yang dapat digunakan. Misalnya, PHP dapat mengakses shell di Linux, mempunyai fungsi yang lengkap berhubungan dengan *networking*.

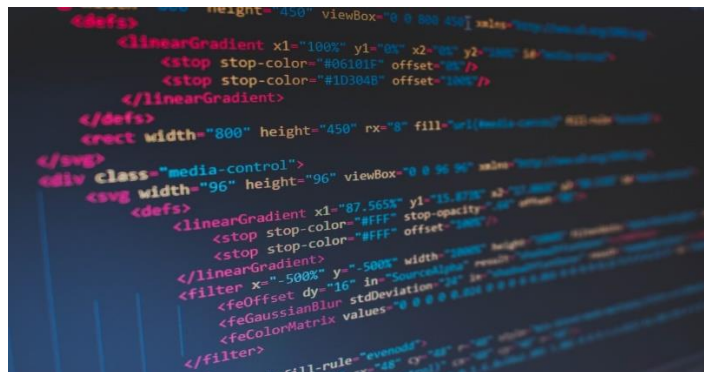
Salah satu keunggulan PHP dibanding bahasa pemrograman lainnya adalah PHP dapat diperoleh secara gratis, meskipun bukan berarti karena gratis kemampuannya menjadi pas-pasan. PHP sangat powefull. Terbutki dengan banyaknya website yang dibangun menggunakan PHP.

PHP juga terkenal lebih aman daripada bahasa pemrograman website yang lain. PHP juga sudah mendukung OOP (*Object Oriented Programming*) sehingga *maintenance* kode menjadi jauh lebih muda dibandingkan procedural.

PHP bisa berinteraksi dengan database, file dan folder, PHP ini termaksud bahasa *cross-platform*, yang mana PHP ini bisa berjalan di sistem operasi yang berbeda-beda (Windows, Linux, ataupun MAC).[10]

Berdasarkan pernyataan diatas dapat disimpulkan bahwa PHP adalah bahasa adalah sebuah bahasa pemrograman yang sangat fleksibel untuk digunakan dalam membuat sebuah website, dan juga PHP dapat dijalankan di bawah sistem operasi LINUX dan Windows.

## 4.2.2 HTML



### *Gambar Script HTML*

Sumber : enimamedia.com

HTTP merupakan protokol yang digunakan untuk mentransfer data antara *web server* ke *web browser*. Protokol ini mentransfer dokumen-dokumen web yang ditulis atau berformat HTML. Dikatakan markup karena HTML berfungsi untuk "mepercantik" file text biasa untuk ditampilkan pada program web browser. Hal ini dilakukan dengan menambahkan tag-tag (perintah khusus) pada file teks biasa tersebut.

Tag HTML biasanya berupa tag-tag yang berpasangan pasangan dan ditandai dengan simbol `<dan>`. Pasangan dari sebuah tag ditandai dengan tanda `'/'`. misalnya awalnya perintah tagnya `<contoh>` maka diakhiri dengan `</contoh>`.

Supaya dapat menghasilkan tampilan wujud yang terintegrasi Pemformatan hiperteks sederhana ditulis dalam berkas format ASCII sehingga menjadi halaman web dengan perintah-perintah HTML.

HTML merupakan sebuah bahasa yang bermula bahasa yang sebelumnya banyak dipakai di dunia percetakan dan penerbitan yang disebut Standard Generalized Markup Language (SGML).

Sekarang ini HTML merupakan standar Internet yang dikendalikan dan didefinisikan pemakaiannya oleh World Wide Web Consortium (W3C). Pada tahun 1989, HTML dibuat oleh kolaborasi Berners-lee Robert dengan Caillau TIM pada saat mereka bekerja di CERN (CERN merupakan lembaga penelitian fisika energi tinggi di Jenewa).

#### 4.2.2.1 Fungsi HTML (*HyperText Markup Language*)

HTML (*HyperText Markup Language*) adalah suatu bahasa yang menggunakan tanda-tanda tertentu (*tag*) untuk menyatakan kode-kode yang harus ditafsirkan oleh *browser* agar halaman tersebut dapat ditampilkan secara benar.

Secara umum, fungsi HTML adalah untuk mengelola serangkaian data dan informasi sehingga suatu dokumen dapat diakses dan ditampilkan di Internet melalui layanan *web*.

Fungsi HTML yang lebih spesifik yaitu :

- Membuat halaman *web*.
- Menampilkan berbagai informasi di dalam sebuah *browser Internet*.
- Membuat link menuju halaman web lain dengan kode tertentu (*hypertext*).
- HTTP atau *Hypertext Transfer* Protokol merupakan protokol yang digunakan untuk mentransfer data atau document yang berformat HTML dari *web server* ke *web browser*. Dengan HTTP inilah yang memungkinkan Anda menjelajah internet dan melihat halaman *web*.

#### 4.2.2.2 Sejarah HTML

HTML dibuat oleh Tim Berners-Lee, seorang ahli fisika di lembaga penelitian CERN yang berlokasi di Swiss. Dia memiliki ide tentang sistem *hypertext* yang berbasis internet.

*Hypertext* merujuk pada teks yang memuat referensi (link) ke teks lain yang bisa diakses langsung oleh viewer. Tim merilis versi pertama HTML pada tahun 1991, dan di dalamnya terdiri atas 18 HTML tag. Sejak saat itu, setiap kali bahasa HTML

merilis versi teranyarnya, selalu ada *tag* dan *attribute* (*tag modifier*) terbaru.

Berdasarkan *HTML Element Reference* milik Mozilla *Developer Network*, untuk saat ini, ada 140 HTL tag meskipun sebagiannya sudah usang (tidak lagi didukung oleh versi terbaru browser).

Berkat popularitasnya yang terus meningkat, HTML kini dianggap sebagai web standard yang resmi. Spesifikasi HTML di-*maintain* dan dikembangkan oleh *World Wide Web Consortium* (W3C). Cek versi terbaru dari bahasa ini di website W3C.

*Upgrade* HTML besar-besaran terjadi pada tahun 2014, dan hasilnya adalah pengenalan HTML5. Pada *upgrade* tersebut, terdapat semantic baru yang memberitahukan arti dari kontennya sendiri, seperti <artcile>, <header>, dan <footer>.

#### **4.2.2.3 Cara Kerja HTML**

Dokumen HTML adalah file yang diakhiri dengan ekstensi .html atau .htm. Ekstensi file ini bisa dilihat dengan menggunakan web browser apa pun (seperti Google Chrome, Safari, atau Mozilla Firefox). Browser tersebut membaca file HTML dan me-*render* kontennya sehingga user internet bisa melihat dan membacanya.

Biasanya, rata-rata situs web menyertakan sejumlah halaman HTML yang berbeda-beda. Contohnya, beranda utama, halaman ‘tentang kami’, halaman kontak yang semuanya memiliki dokumen HTML terpisah.

Masing-masing halaman HTML terdiri atas seperangkat tags (bisa disebut juga elements), yang mengacu pada building block

halaman website. Tag tersebut membuat hirarki yang menyusun konten hingga menjadi bagian, paragraf, heading, dan *block* konten lainnya.

Sebagian besar element HTML memiliki tag pembuka dan penutup yang menggunakan syntax `<tag></tag>`.

Berikut contoh kode dari susunan atau struktur HTML:

```
<div>
  <h1>The Main Heading</h1>
  <h2>A catchy subheading</h2>
  <p>Paragraph one</p>
  
  <p>Paragraph two with a <a
href="https://example.com">hyperlink</a></p>
</div>
```

Elemen teratas dan terbawah adalah division sederhana (`<div></div>`) yang bisa Anda gunakan untuk mark up bagian konten yang lebih besar.

Susunan HTML di atas terdiri atas heading (`<h1></h1>`), subheading (`<h2></h2>`), dua paragraf (`<p></p>`), dan satu gambar (`<img>`).

Paragraf kedua meliputi sebuah link (`<a></a>`) dengan attribute href yang terdiri atas URL tujuan.

Tag gambar memiliki dua attribute, src untuk path gambar dan alt untuk deskripsi gambar.

#### 4.2.2.4 Gambaran Umum Tentang Tag HTML yang Paling Sering Digunakan

Tag HTML memiliki dua tipe utama: *block-level* dan *inline tags*. Elemen *block-level* memakai semua space yang tersedia dan selalu membuat line baru di dalam dokumen. Contoh dari *tag block* adalah *heading* dan paragraf.

Elemen *inline* hanya memakai *space* sesuai dengan kebutuhannya dan tidak membuat line baru di halaman. Biasanya elemen ini akan memformat isi konten dari elemen *block-level*. Contoh dari *tag inline* adalah link dan *emphasized strings*.

- *Tag Block-Level*

Tiga *tag block-level* yang harus dimiliki oleh setiap dokumen HTML adalah `<html>`, `<head>`, dan `<body>`.

Tag `<html></html>` adalah elemen level tertinggi yang menyertakan setiap halaman HTML.

Tag `<head></head>` menyimpan informasi meta, seperti judul dan charset halaman.

Tag `<body></body>` melampirkan semua konten yang muncul pada suatu halaman.

```
<html>
  <head>
    <!-- META INFORMATION -->
  </head>
  <body>
    <!-- PAGE CONTENT -->
  </body>
</html>
```



*Heading* memiliki 6 level di HTML. Level tersebut bervariasi, mulai dari `<h1></h1>` sampai ke `<h6></h6>`, di mana h1 merupakan level heading tertinggi dan h6 adalah level terendah. Paragraf dibuka dan ditutup dengan tag `<p></p>`, sedangkan *blockquote* menggunakan tag `<blockquote></blockquote>`.

*Division* merupakan bagian konten yang lebih besar dan biasanya terdiri atas beberapa paragraf, gambar, kadang-kadang *blockquote*, dan elemen lebih kecil lainnya. Kita bisa membuat markup dengan menggunakan tag `<div></div>`. Di dalam elemen div juga terdapat tag div lainnya.

Anda juga bisa menggunakan tag `<ol></ol>` untuk list yang berurutan dan `<ul></ul>` untuk list yang tidak berurutan. Masing-masing list item harus dibuka dan ditutup dengan tag `<li></li>`. Sebagai contoh, di bawah ini adalah tampilan dasar dari list tidak berurutan dalam HTML:

```
<ul>
  <li>List item 1</li>
  <li>List item 2</li>
  <li>List item 3</li>
</ul>
```

- *Tag Inline*

Sebagian besar *tag inline* digunakan untuk memformat teks. Sebagai contoh, tag `<strong></strong>` akan *render* elemen ke format *bold*, sedangkan tag `<em></em>` akan ditampilkan dalam format *italic*.

*Hyperlink* adalah elemen *inline* yang mewajibkan adanya tag `<a></a>` dan attribute href untuk mengindikasikan tujuan link:

```
<a href="https://example.com/">Click me!</a>
```

Gambar (*image*) juga merupakan elemen *inline*. Anda dapat menambahkan satu gambar dengan menggunakan `<img>` tanpa harus membubuhkan *tag* penutup. Hanya saja, Anda disarankan menggunakan *attribute* `src` untuk menentukan path gambar, misalnya:

```

```

Jika ingin mempelajari lebih dalam tentang tag HTML, silakan baca artikel kami tentang HTML *cheat sheet* (kami juga menyediakan link untuk diunduh).

#### 4.2.2.5 Kelebihan dan Kekurangan HTML

Sama seperti hal teknis lainnya dalam dunia web, HTML juga punya kelebihan dan kekurangannya.

- Kelebihan
  - Bahasa yang digunakan secara luas dan memiliki banyak sumber serta komunitas yang besar.
  - Dijalankan secara alami di setiap *web browser*.
  - Memiliki *learning curve* yang mudah.
  - *Open-source* dan sepenuhnya gratis.
  - Bahasa markup yang rapi dan konsisten.
  - *Standard web* yang resmi di-maintain oleh *World Wide Web Consortium* (W3C).
  - Mudah diintegrasikan dengan bahasa *backend*, seperti PHP dan Node.js.

- Kekurangan

Paling sering digunakan untuk halaman *web* statis. Untuk fitur dinamis, Anda bisa menggunakan *JavaScript* atau bahasa backend, seperti PHP.

HTML tidak memungkinkan user untuk menjalankan logic. Alhasil, semua halaman web harus dibuat terpisah meskipun menggunakan elemen yang sama, seperti header dan footer. Fitur-fitur baru tidak bisa digunakan secara cepat di sebagian browser.

Terkadang perilaku browser susah untuk diprediksi (misalnya, browser lama tidak selalu bisa render tag yang lebih baru).

#### 4.2.3 CSS (*Cascading Style Sheets*)

CSS (*Cascading Style Sheets*) Pada versi HTML yang terdahulu, web browser mengontrol tampilan (rendering) dari setiap halaman web. Jika menggunakan elemen H1 pada (*large heading*) pada web dokumen, browser akan merender elemen tersebut.

Dengan adanya CSS, *programmer* dapat mengontrol bagaimana *browser me-render* halaman web. Mengaplikasikan CSS pada halaman web dapat memberikan tampilan yang lebih menarik dan spesifik sesuai dengan tema pada sebuah web site.

Teknologi CSS memberikan fasilitas untuk menentukan style (misal; *spacing, margins*) dari elemen halaman web terpisah dari struktur dokumen web (*section headers, body text, links*). Pemisahan tersebut memberikan peningkatan yang lebih besar dalam pengaturan *web pages*, dan membuat perubahan –

perubahan *style* dalam dokumen dapat dilakukan lebih cepat dan lebih mudah.

#### 4.2.3.1 Kelebihan dan Kekurangan CSS

##### a. Kelebihan

- *Update* tampilan lebih mudah
- Beban *bandwidth* lebih kecil
- Modifikasi *Web template* lebih mudah
- Lebih mudah digunakan pada *mobile phone*
- *Search engine friendly*
- Memisahkan desain dengan konten halaman web
- Mengatur desain seefisien mungkin
- Jika kita ingin mengubah suatu tema halaman web, cukup modifikasi pada css saja
- Menghadirkan sesuatu yang tidak dapat dilakukan oleh HTML
- Lebih mudah didownload karena lebih ringan ukuran filenya
- Satu CSS dapat digunakan banyak halaman web
- Menghemat penulisan kode, karena satu css dapat dipakai beberapa kali
- Mempersingkat waktu kerja, baik saat membuat maupun saat modifikasi halaman *Web/Blog*, dan masih banyak lagi.

##### b. Kekurangan

- Tampilan pada browser berbeda-beda.
- Kadang juga terdapat *browser* yang tidak support CSS (*browser* lama).
- Harus tahu cara menggunakannya.

- Dibutuhkan waktu lebih lama dalam membuatnya.
- Belum lagi ada *bug/error* dalam CSS