

**BUKU TUTORIAL MEMBUAT SISTEM INFORMASI  
MENGGUNAKAN CODEIGNITER DENGAN  
NOTIFIKASI TELEGRAM**

**BUKU TUTORIAL MEMBUAT SISTEM INFORMASI  
MENGGUNAKAN CODEIGNITER DENGAN  
NOTIFIKASI TELEGRAM**

**(Studi Kasus: PT. X)**

Buku ini dibuat untuk memenuhi persyaratan kelulusan  
matakuliah Program Internship I



**Dibuat Oleh,  
1.16.4.036 Farid Ariyanto Saputra**

**PROGRAM DIPLOMA IV TEKNIK INFORMATIKA  
POLITEKNIK POS INDONESIA  
BANDUNG  
2020**

# BUKU TUTORIAL MEMBUAT SISTEM INFORMASI MENGGUNAKAN CODEIGNITER DENGAN NOTIFIKASI TELEGRAM



Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan laporan praktek kerja lapangan dengan judul ”Analisis penambahan Digital Invoice dan Settlement Pada Pengembangan Aplikasi Telkom Partner Network (Studi kasus: PT.Telkomunikasi Indonesia)”. Penulisan laporan Internship I ini dan penulis menyadari bahwa penyusunan laporan ini masih belum sempurna. Ini mengingat keterbatasan pengetahuan, pengalaman serta kemampuan penulis. Penulis megharapkan kritik dan saran yang sifatnya membangun dari pembaca. Untuk itu penulis mengucapkan terima kasih kepada:

1. Allah S.W.T yang telah melimpahkan karunia-Nya sehingga kami dapat menyelesaikan tugas ini;
2. Kedua orang tua dan keluarga saya yang telah memberi motivasi kepada saya;
3. Hezekieli Gulo. selaku pembimbing eksternal yang telah membantu selama kegiatan intership.
4. Bapak Mohamad Nurkamal Fauzan, S.T., M.T. selaku dosen pembimbing internship 1.
5. Ibu Nisa Hanum Harani, S.T., M.T.I selaku Koordinator Internship I.
6. Bapak M. Yusril Helmi Setyawan, S.Kom., M.Kom. selaku Ketua Prodi D4 Teknik Informatika.
7. Pihak Telekomunikasi Indonesia yang telah banyak membantu dalam memberikan solusi dan saran terhadap penelitian yang saya buat;

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membala segala kebaikan semua pihak yang telah membantu. Semoga buku ini membawa manfaat bagi pengembangan ilmu.

Bandung, 20 Januari 2020

Penulis

## DAFTAR ISI

DAFTAR ISI.....	5
DAFTAR GAMBAR .....	8
BAB I PENGENALAN SISTEM,INFROMASI,FRAMEWORK,WEB.....	10
1. Pengenalan Sistem,Infromasi, Framework,Web .....	10
1.1 Pengantar Sistem .....	10
1.2 Pengantar Informasi.....	16
1.3 Pengantar Framework.....	24
1.3.1 Laravel .....	24
1.3.2 Yii .....	26
1.3.3 Code Igniter .....	29
1.3.4 Symfony.....	32
1.3.5 Zend Framework.....	34
1.3.6 Cake PHP .....	35
1.3.7 Fuel PHP .....	36
1.3.8 Pengantar Web .....	37
1.3.11 Hubungan <i>HTML</i> dan <i>PHP</i> .....	51
1.3.12 Pemahaman Pemrograman Web .....	52
1.3.13 Web Statis dan Web Dinamis .....	53
1.3.14 Basis Data .....	55
BAB II PENGENALAN FRAMEWORK CODE IGNITER DAN NOTIFIKASI TELEGRAM .....	61
2.1 Framework CodeIgniter.....	61
2.2 Sejarah CodeIgniter .....	61
2.3 Konsep MVC Pada CodeIgniter .....	62
2.4 Notifikasi Telegram.....	67

BAB III .....	69
PENJELASAN TOOLS YANG DIGUNAKAN DAN BAHASA	
PEMROGRMAN YANG DIGUNAKAN .....	69
3.1 XAMPP .....	69
3.2 Visual Studio Code.....	76
3.3 Bot Chat Telegram.....	78
3.4 Bahasa Pemrograman .....	81
3.5 Hubungan Antara PHP dan HTML .....	133
BAB IV .....	135
INSTALASI TOOLS YANG DIGUNAKAN.....	
4.1INSTAL XAMPP.....	135
4.1.1 Download Xampp .....	135
4.1.2 Cara menjalan XAMPP .....	142
4.2 INSTAL VISUAL STUDIO CODE .....	143
4.3 INSTAL FRAMEWORK CODE IGNITER .....	147
4.3.1 Download File CodeIgniter.....	147
4.4 INSTAL BOT TELEGRAM .....	150
4.4.1 Mendaftar ke BotFather .....	150
BAB V .....	154
MEMBUAT HELLO WORLD MENGGUNAKAN CODEIGNITER ....	
5.1 Membuat Hello World.....	154
BAB VI .....	158
MEMBUAT SISTEM INFROMASI MENGGUNAKAN CODEIGNITER	
DENGAN NOTIFIKASI TELEGRAM .....	158
6.1 Membuat database .....	158
6.2 Mendownload Template Untuk Aplikasi Yang akan di Bangun .....	162
6.3 Pembuatan Sistem Informasi menggunakan Codeigniter dengan notifikasi Telegram .....	171

BAB VII.....	218
TAMPILAN ANTARMUKA .....	218
7.1 Login .....	218
7.2 Halaman Menu Utama .....	218
7.3 Halaman Permintaan Baru .....	219
7. 4 Halaman Pengisian Survey .....	219
7.5 Halaman Mitra Kerjasama.....	221
7.6 Halaman Pengiriman Pesan Telegram.....	222

## **DAFTAR GAMBAR**

Gambar 1.1 Elemen-Elemen Pembentuk Sistem.....	12
Gambar 1.2 Ilustrasi Informasi .....	16
Gambar 1.3 Laravel .....	24
Gambar 1.4 Yii.....	26
Gambar 1.5 Codeigniter.....	29
Gambar 1.6 Ilustrasi Web .....	37
Gambar 1.7 Pengaksesan Informasi Melalui Hypertext .....	38
Gambar 1.8 Contoh Situs Http: Browser Google Chrome Menampilkan Status Website Dengan Protokol Http .....	40
Gambar 1.9 Status “Not Secure” Pada Browser Google Chrome .....	40
Gambar 1.10 Gambar Contoh Situs Https: Status “Secure” Pada Browser Google Chrome.....	42
Gambar 1.11 Gambar Cara Kerja Web.....	52
Gambar 1.12 Gambar Web Statis Vs Web Dinamis.....	55
Gambar 2.1 Framework Codeigniter .....	61
Gambar 2.2 Konsev Mvc .....	62
Gambar 2.3 Tampilan Model.....	64
Gambar 2.4 Tampilan View.....	64
Gambar 2.5 Tampilan Controller.....	65
Gambar 3.1 Xampp .....	69
Gambar 3.1 Versi Xampp .....	74
Gambar 3.3 Visual Studio Code .....	76
Gambar 3.4 Tampilan Visual Studio Code.....	77
Gambar 3.5 Tampilan Botfather .....	78
Gambar 3.6 Php .....	81
Gambar 3.7 Perluasan File Php.....	92

Gambar 3.8 Perintah Mencetak Hasil .....	93
Gambar 3.9 Variabel Nilai.....	93
Gambar 3.10 Tipe Data Dari Variabel .....	93
Gambar 3.11 Deklarasi Variabel.....	94
Gambar 3.12 Scope Lokal.....	95
Gambar 3.13 Scope Global .....	96
Gambar 3.14 Scope Global 2 .....	96
Gambar 3.15 Scope Statik.....	96
Gambar 7.1 Login .....	218
Gambar 7.2 Halaman Menu Utama .....	218
Gambar 7.3 Halaman Permintaan Baru .....	219
Gambar 7.4 Halaman Pengisian Survey .....	219
Gambar 7.4.1 Halaman Survey Desain Inventory .....	220
Gambar 7.4.2 Halaman Pembagunan Kontruksi.....	221
Gambar 7.5 Halaman Mitra Kerjasama .....	221
Gambar 7.6 Halaman Pengiriman Pesan Telegram .....	222
Gambar 7.6.1pesan Terkirim .....	223

# **BAB I**

## **PENGENALAN SISTEM,INFROMASI, FRAMEWORK,WEB**

### **1. Pengenalan Sistem,Infomasi, Framework,Web**

Ada beberapa istilah yang akan digunakan dalam penyusunan buku ini, diantaranya adalah sebagai berikut :

1. Sistem
2. Informasi
3. Framework
4. *Web*

Penulis akan memaparkan penjelasan mengenai ketiga istilah tersebut.

#### **1.1 Pengantar Sistem**

Apa yang dimaksud dengan sistem (*system*)? Secara umum, pengertian sistem adalah suatu kesatuan, baik obyek nyata atau abstrak yang terdiri dari berbagai komponen atau unsur yang saling berkaitan, saling tergantung, saling mendukung, dan secara keseluruhan bersatu dalam satu kesatuan untuk mencapai tujuan tertentu secara efektif dan efisien.

Ada juga yang mengatakan definisi sistem adalah suatu paduan yang terdiri dari beberapa unsur/ elemen yang dihubungkan menjadi satu kesatuan sehingga memudahkan aliran informasi dan materi/ energi untuk mewujudkan suatu tujuan tertentu.

Secara etimologis, istilah “sistem” berasal dari bahasa Latin (*systēma*) dan bahasa Yunani (*sustēma*) yang sering dipakai untuk memudahkan dalam menggambarkan interaksi di dalam suatu entitas.

### **1.1.1 Pengertian Sistem Menurut Para Ahli**

#### **1. Harijono Djojodihardjo**

Menurut Harijono Djojodihardjo, pengertian sistem adalah gabungan obyek yang memiliki hubungan secara fungsi dan hubungan antara setiap ciri obyek, secara keseluruhan menjadi suatu kesatuan yang berfungsi.

#### **2. Indrajit**

Menurut Indrajit, pengertian sistem adalah kumpulan dari komponen-komponen yang saling memiliki unsur keterkaitan antara satu dengan lainnya.

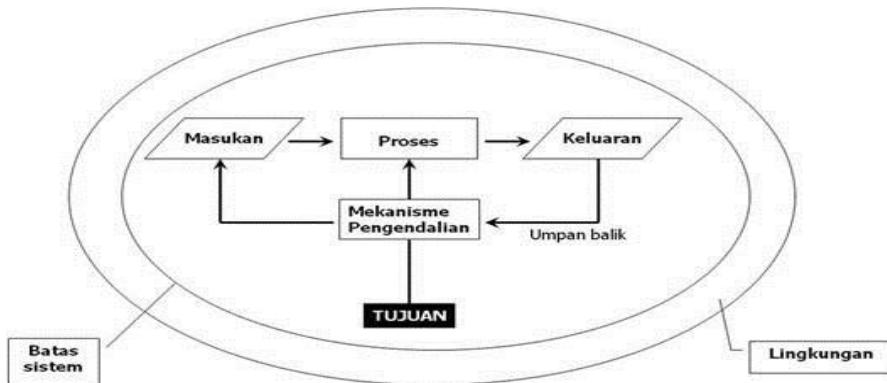
#### **3. Jogianto**

Menurut Jogianto, definisi sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu yang menggambarkan berbagai kejadian dan kesatuan yang nyata, seperti tempat, benda dan orang-orang yang betul-betul ada dan terjadi.

### **1.1.2 Unsur-unsur untuk membentuk suatu sistem**

- a. Obyek, di dalam sistem terdapat sekumpulan obyek (fisik/ abstrak) dalam bentuk elemen, bagian, atau variabel.
- b. Atribut, sesuatu yang menentukan mutu atau sifat kepemilikan suatu sistem dan obyeknya.
- c. Hubungan internal, setiap elemen saling terikat menjadi satu kesatuan.
- d. Lingkungan, tempat atau wilayah dimana sistem berada.

### 1.1.3 Elemen-Elemen Pembentuk Sistem



Gambar 1.1 Elemen-Elemen Pembentuk Sistem

Elemen pembentuk sistem itu meliputi diantaranya, yaitu:

1. Masukan, yang berarti semua yang masuk dalam sistem akan diproses, apakah itu merupakan objek fisik ataupun objek abstrak.
2. Tujuan, yang berarti sistem dibuat demi mencapai tujuan atau output tertentu yang ingin dicapai.
3. Proses, yaitu transformasi dari masukan yang diubah menjadi keluaran yang memiliki nilai lebih tinggi.
4. Keluaran atau *output*, yang merupakan hasil dari pemrosesan yang mana wujudnya dapat berbentuk informasi, cetakan laporan, sara, produk, dan lain sebagainya.
5. Batas, yakni sesuatu yang memisahkan antara sistem dan daerah yang berada di luar sistem. Batas inilah yang akan menentukan ruang lingkup, konfigurasi, dan banyak hal lain sebagainya.
6. Pengendalian dan umpan balik, yang mana mekanismenya bisa dilakukan dengan menggunakan *feedback* terhadap keluaran untuk mengendalikan elemen masukan atau proses.

7. Lingkungan, yaitu segala sesuatu di luar sistem yang memiliki pengaruh terhadap sistem, baik yang menguntungkan ataupun merugikan.

#### **1.1.4 Jenis-Jenis Sistem**

##### **1. Berdasarkan Keterbukaan**

Jenis sistem yang pertama adalah berdasarkan keterbukaan, maka sistem itu terbagi menjadi dua yakni sistem terbuka dan sistem tertutup:

- i. Sistem Terbuka, yakni suatu sistem yang bisa dipengaruhi oleh pihak luar karena adanya akses yang terbuka.
  - ii. Sistem Tertutup, yakni kebalikan dari sistem terbuka yaitu sistem yang tidak dapat dipengaruhi oleh pihak luar dikarenakan akses yang tertutup.
2. Berdasarkan Komponen

Berdasarkan komponennya sistem dibagi atas dua jenis. Yakni sistem fisik dan non-fisik. Perbedaan keduanya adalah sebagai berikut:

- i. Sistem Fisik, yakni suatu sistem yang mempunyai komponen energi dan materi.
- ii. Sistem non-fisik, yakni suatu sistem yang berbentuk abstrak, contohnya: dapat berupa ide, konsep, dan gagasan serta banyak contoh lainnya.

### **1.1.5 Karakter Sistem**

Karakter sistem adalah sistem yang mempunyai komponen-komponen, batas sistem, penghubung, masukan, keluaran, pengolahan dan sasaran.

#### **1. Komponen**

Elemen-elemen yang lebih kecil yang disebut *sub sistem*, misalkan sistem computer terdiri dari sub sistem perangkat keras, perangkat lunak dan manusia.

Elemen-elemen yang lebih besar yang disebut *supra sistem*, misalkan apabila perangkat keras adalah sebuah sistem yang memiliki sub sistem *CPU*, perangkat *I/O* dan memori, maka supra sistem perangkat keras adalah sistem computer.

#### **2. *Boundary* (Batasan Sistem)**

Batas sistem merupakan daerah yang membatasi suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya.

Batas sistem ini menunjukkan ruang lingkup dari sistem tersebut.

#### **3. *Environment* (Lingkungan Luar Sistem)**

Lingkungan dari sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan untuk sistem itu sendiri. Lingkungan luar yang menguntungkan merupakan energi dari sistem dari sistem dan demikian harus tetap dijaga dan dipelihara. Sedang lingkungan luar yang harus merugikan harus ditahan dan dikendalikan, jika tidak akan menganggu kelangsungan hidup dari sistem itu sendiri.

#### **4. *Interface (Penghubung Sistem)***

Penghubung merupakan media perantara antar sub sistem. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya. *Output* dari satu sub sistem akan menjadi input untuk subsistem yang lainnya dengan melalui penghubung.

Dengan penghubung satu subsistem dapat berinteraksi dengan sub sistem yang lainnya membentuk satu kesatuan.

#### **5. *Input (Masukan)***

Masukan adalah energi yang dimasukkan ke dalam sistem.

Masukan dapat berupa *maintenance input* dan *sinyal input*. *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Sinyal input* adalah energi yang diproses untuk didapatkan keluaran.

#### **6. *Output (Keluaran)***

Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

#### **7. *Proses (Pengolahan Sistem)***

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi.

## 8. *Objective and Goal* (Sasaran dan Tujuan Sistem)

Suatu sistem pasti mempunyai tujuan atau sasaran. Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

### 1.2 Pengantar Informasi

Informasi adalah data yang diolah menjadi bentuk yang sangat berguna untuk membuat sebuah keputusan. Informasi berguna untuk membuat keputusan karena informasi dapat menurunkan ketidakpastian pada data, karena berdasarkan informasi itu para pengelola dapat mengetahui kondisi obyektif. Informasi tersebut merupakan hasil dari pengolahan data atau fakta yang sudah dikumpulkan menggunakan metode tertentu.



*Gambar 1.2 Ilustrasi Informasi*

### **1.2.1 Ciri-Ciri Informasi**

Ciri-ciri informasi yang berkualitas bisa dilihat sebagai berikut:

1. Akurat, artinya informasi mencerminkan keadaan sebenarnya.
2. Tepat waktu, artinya informasi harus ada saat diperlukan.
3. Relevan, informasi yang diberikan harus sesuai dengan yang dibutuhkan.
4. Lengkap, artinya informasi harus utuh, tidak setengah-setengah.

### **1.2.2 Jenis-Jenis Informasi**

1. **Absolute Information**, merupakan “pohonnya” sebuah informasi yaitu jenis informasi yang disajikan dengan sebuah jaminan serta tidak membutuhkan penjelasan lebih lanjut.
2. **Substitutional Information**, merupakan jenis yang merujuk pada kasus dimana konsep informasi digunakan untuk sejumlah informasi.
3. **Philosophic Information**, merupakan jenis informasi yang berkaitan dengan perasaan dan infomasi manusia. Adanya infomasi ini bergantung kepada orang yang menyajikannya.
4. **Objective Information**, merupakan jenis informasi yang merujuk pada karakter logis informasi tertentu.
5. **Cultural Information**, merupakan informasi yang memberikan tekanan pada dimensi *cultural* (budaya).

### **1.2.3 Contoh Informasi**

1. Kenaikan harga BBM adalah Rp. 500 per liter untuk jenis pertamax.
2. Minuman jus wortel kaya akan vitamin A.
3. Harga laptop dan *smartphone* terbaru dengan harga di atas 2 juta.
4. Harga satu kilogram telur adalah 22.000 per kilonya.

#### **1.2.4 Sumber Data atau Informasi**

Sumber infomasi bisa didapat dari:

1. Lingkungan kerja
2. Lembaga pendidikan
3. Media masa
4. Instansi pemerintah
5. Masyarakat

#### **1.2.5 Cara Mendapatkan Data atau Informasi**

Ada beberapa cara untuk mendapatkan data atau informasi,diantaranya sebagai berikut:

1. Hasil penelitian yang dilakukan sebelumnya.
2. Data yang telah lewat dengan memperhatikan *trend* dan taksiran di masa depan.
3. Mengambil dari pusat data seperti Badan Pusat Statistik (BPS) dan Pusat Data Informasi Pertanian (Pustadin).
4. Media elektronik seperti televisi, radi, dan internet.
5. Media cetak seperti buku, majalah, karya ilmiah, koran, proposal, dan lain sebagainya.
6. Forum, seperti seminar, pelatihan, dan pendidikan.

#### **1.2.6 Kualitas Informasi**

Kualitas informasi ditentukan oleh berapa faktor yaitu sebagai berikut:

1. Keakuratan dan teruji kebenarannya

Informasi harus bebas dari kesalahan-kesalahan serta tidak menyesatkan.

2. Kesempurnaan Informasi

Informasi harus disajikan dengan lengkap tanpa pengurangan dan penambahan, serta adanya pengubahan.

### 3. Tepat Waktu

Informasi harus disajikan tepat waktu, karena menjadi dasar dalam pengambilan sebuah keputusan nantinya.

### 4. Relevansi

Informasi akan mempunyai manfaat dengan nilai yang sangat tinggi, jika informasi tersebut dapat diterima oleh mereka yang membutuhkan.

### 5. Mudah dan Cerah

Apabila cara serta biaya untuk memperoleh informasi sulit serta mahala, maka orang menjadi tidak berminat untuk memperolehnya, atau akan mencari alternatif lainnya untuk memperoleh infomasi itu sendiri.

### 6. Kualitas suatu informasi tergantung dari 3 hal, yaitu:

- i. Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan harus jelas mencerminkan maksudnya.
- ii. Tepat pada waktunya, berarti informasi yang diterima tidak boleh terlambat atau dengan kata lain harus tepat waktu.
- iii. Relevan, berarti informasi tersebut mempunyai manfaat dari pemakainya.

## 1.2.7 Tujuan Klasifikasi Data atau Informasi

1. Melindungi perjanjian kontrak (infomasi) dengan mitra bisnis atau konsumennya.
2. Memberikan pengamanan yang sesuai, sehingga menghemat sumber daya organisasi dan membuat pengelolaan infomasi menjadi efisien dan efektif.
3. Membantu meningkatkan kualitas data/infromasi yang digunakan sebagai bahan untuk mengambil keputusan.

### **1.2.8 Manfaat Data atau Informasi**

Data atau Informasi memiliki manfaat yaitu:

1. Meninkatkan wawasan dan pengetahuan organisasi.
2. Mengurangi resiko kesalahan dalam pengambilan keputusan.
3. Menggambarkan kondisi yang terjadi di masa kini.
4. Memberi gambaran *trend* atau kecendrungan di masa depan.
5. Mengurangi ketidak pastian kondisi karena adanya keimangsiuran fenomena.
6. Menjadi dasar bagi pemecahan masalah.
7. Menghasilkan arus kerja menjadi lebih efektif dan efisien.
8. Meningkatkan citra positif perusahaan.
9. Menambah relasi.
10. Meningkatkan kepercayaan pemegang saham.
11. Memberi arahan bagi promosi yang lebih jelas.
12. Menjadi dasar pertanggung jawaban atas segala tindakan yang sudah diambil.
13. Memberikan bukti, bukan kesan, isu, atau opini dari pihak lain.

### **1.2.9 Komponen Informasi**

Komponen-komponen informasi meliputi:

1. ***Root of Information***, yaitu komponen inti datu informasi berada pada tahap keluaran pertama sebuah proses pengolahan data yang biasanya disampaikan oleh orang pertama.
2. ***Bar of Information***, yaitu merupakan badan/batangnya dari informasi yang disajikan dan memerlukan informasi pendukung, agar informasi inti dapat diketahui secara utuh. Contoh : headline surat kabar agar pembaca jelas maka harus membaca informasi selanjutnya.

3. ***Branch of Information***, yaitu informasi dapat dipahami apabila informasi sebelumnya telah dipahami. Misalnya, ketika kita membaca glosarium atau indeks ketika membaca sebuah buku.
4. ***Stick of Information***, yaitu komponen informasi yang sederhana dari cabang informasi. Bentuk dari informasi ini biasanya berbentuk pengayaan pengetahuan, kedudukannya hanya sebagai pelengkap, terhadapa informasi yang ada.
5. ***Bud of Information***, yaitu komponen informasi yang sifatnya semi mikro namun sangat dibutuhkan, sehingga diwaktu mendatang informasi ini akan berkembang dan dicari orang, misalnya informasi tentang multiple intelligence, hypoteaching, kurikulum masa depan, pembelajaran abad ke 21, dan lain-lain.
6. ***Leaf of Information***. yaitu merupakan informasi pelindung untuk menjelaskan kondisi dan situasi ketika informasi itu mucul ke permukaan, seperti informasi tentang prakiraan cuaca, prakiraan kemarau panjang, prakiraan gempa atau gerhana matahari/bulan.

### 1.2.10 Perubahan Data Menjadi Informasi

Pemrosesan data (Inggris: *data processing*) adalah jenis pemrosesan yang dapat mengubah data menjadi informasi atau pengetahuan. Pemrosesan data ini sering menggunakan komputer sehingga bisa berjalan secara otomatis. Setelah diolah, data ini biasanya mempunyai nilai yang informatif jika dinyatakan dan dikemas secara terorganisir dan rapi, maka istilah pemrosesan data sering dikatakan sebagai sistem informasi.

Kedua istilah ini mempunyai arti yang hampir sama,pemrosesan data mengolah dan memanipulasi data mentah menjadi informasi (hasil pengolahan), sedangkan sistem informasi memakai data sebagai bahan masukan dan menghasilkan informasi sebagai produk keluaran.Pada saat ini kegiatan Data Processing sudah semakin luas, baik yang berorientasi kepada ilmu pengetahuan, komersil/bisnis maupun kegiatan pemerintahan, sehingga data yang diolahpun akan bermacam-macam sesuai dengan bidang pekerjaan tersebut.

### **1.2.11 Fungsi Informasi**

#### **1. Menjadi Sumber Pengetahuan Baru**

Informasi valid yang didapatkan oleh seseorang dapat menjadi pengetahuan baru dan menambah wawasan di bidang tertentu. Misalnya informasi mengenai cara mengatasi masalah kesehatan yang didapatkan dari konten di internet.

Mungkin informasi tersebut adalah sesuatu yang umum dan sudah banyak diketahui orang. Namun, mungkin saja ada seseorang yang belum mengetahui informasi tersebut.

#### **2. Menghapus Ketidakpastian**

Kurangnya informasi tentang sesuatu akan menimbulkan ketidakpastian. Untuk menghapus ketidak pastian tersebut maka diperlukan informasi lengkap dan valid dari sumber terpercaya.

#### **3. Sebagai Media Hiburan**

Informasi juga dapat berfungsi sebagai media hiburan bagi masyarakat. Misalnya informasi mengenai objek wisata di suatu tempat yang disajikan dengan bahasa dan gambar-gambar yang menarik.

#### **4. Sebagai Sumber Berita**

Suatu informasi mengenai hal tertentu bisa dipakai sebagai sumber berita yang disampaikan kepada khalayak. Misalnya, informasi tentang Asian Games yang didapatkan dari media Televisi, Radio, dan situs berita online.

#### **5. Untuk Sosialisasi Kebijakan**

Informasi adalah komponen penting dalam berkomunikasi dengan pihak lain. Salah satunya adalah untuk menyampaikan suatu kebijakan dari pemerintah kepada masyarakat yang dilakukan dengan cara sosialisasi.

#### **6. Untuk Mempengaruhi Khalayak**

Penyampaian informasi melalui media massa biasanya dilakukan untuk mempengaruhi khalayak. Misalnya informasi mengenai suatu produk melalui Televisi yang tujuannya agar masyarakat mengenal dan tertarik untuk menggunakannya.

#### **7. Menyatukan Pendapat**

Di era media sosial seperti sekarang ini, sangat mudah untuk menyampaikan pendapat ke ruang publik. Namun, tidak semua pendapat tersebut sesuai dengan fakta yang ada.

Adanya informasi yang valid dari sumber terpercaya akan bermanfaat untuk menilai setiap pendapat yang dikemukakan di ruang publik apakah sesuai dengan informasi tersebut.

### **1.3 Pengantar Framework**

Framework adalah sebuah software untuk memudahkan para programer untuk membuat sebuah aplikasi web yang di dalamnya ada berbagai fungsi diantaranya plugin, dan konsep untuk membentuk suatu sistem tertentu agar tersusun dan terseruktur dengan rapih.

Dengan menggunakan framework bukan berarti kita akan terbebas dengan pengkodean. Karena kita sebagai seorang pengguna framework haruslah menggunakan fungsi – fungsi dan variable yang ada di dalam sebuah framework yang kita gunakan. Untuk saat ini framework terbagi dua yaitu framework PHP dan framework CSS, untuk framework PHP terdiri dari :

#### **1.3.1 Laravel**



*Gambar 1.3 Laravel*

Laravel diluncurkan sejak tahun 2011 dan mengalami pertumbuhan yang cukup eksponensial. Di tahun 2015, Laravel adalah framework yang paling banyak mendapatkan bintang di Github. Sekarang framework ini menjadi salah satu yang populer di dunia, tidak terkecuali di Indonesia.

Laravel fokus di bagian end-user, yang berarti fokus pada kejelasan dan kesederhanaan, baik penulisan maupun tampilan, serta menghasilkan fungsionalitas aplikasi web yang bekerja sebagaimana

mestinya. Hal ini membuat developer maupun perusahaan menggunakan framework ini untuk membangun apa pun, mulai dari proyek kecil hingga skala perusahaan kelas atas.

Laravel mengubah pengembangan website menjadi lebih elegan, ekspresif, dan menyenangkan, sesuai dengan jargonnya “*The PHP Framework For Web Artisans*”. Selain itu, Laravel juga mempermudah proses pengembangan website dengan bantuan beberapa fitur unggulan, seperti *Template Engine*, *Routing*, dan *Modularity*.

### 1.3.1.1 Manfaat Laravel

Laravel menawarkan beberapa keuntungan ketika Anda mengembangkan website menggunakan dasar framework ini.

- Pertama, website menjadi lebih *scalable* (mudah dikembangkan).
- Kedua, terdapat *namespace* dan tampilan yang membantu Anda untuk mengorganisir dan mengatur sumber daya website.
- Ketiga, proses pengembangan menjadi lebih cepat sehingga menghemat waktu karena Laravel dapat dikombinasikan dengan beberapa komponen dari framework lain untuk mengembangkan website.

### 1.3.1.2 Tool Andalan Laravel

#### 1. Composer

Composer merupakan tool yang di dalamnya terdapat *dependencies* dan kumpulan *library*. Seluruh *dependencies* disimpan menggunakan format file composer.json sehingga dapat ditempatkan di dalam folder utama website.

## 2. Artisan

Artisan merupakan command line interface yang dimiliki oleh Laravel. Artisan mencakup sekumpulan perintah yang membantu Anda untuk membangun sebuah website atau aplikasi web.

### 1.3.2 Yii



*Gambar 1.4 Yii*

Framework YII adalah sebuah software yang berfungsi sebagai kerangka kerja php yang berbasis komponen dengan performansi tinggi untuk pembuatan sebuah aplikasi web bersekalai besar, yii menyediakan resuabilitas maksimum dalam pemrograman Web dan bisa mengakselerasi proses pengembangan secara signifikan.

Seperti kebanyakan framework php lain framework yii adalah MVC framework. Yii bisa melampaui framework php yang lain dalam hal efisiensi, karena yii telah menyediakan fitur-fitur yang sudah di dokumentasikan dengan jelas. Yii di desain sangat hati-hati agar dapat sesuai dengan pengembangan aplikasi web secara serius.

Yii telah memimplementasikan pola desain MVC (model-view-controller) yang diadopsi secara luas dalam pemrograman Web. Selain implementasi MVC, Yii juga memperkenalkan front controller(controller depan), yang disebut Aplikasi, yang akan mengenkapsulasi konteks eksekusi untuk memproses sebuah request.

Dalam mempelajari Yii kita harus mempunyai skil OOP( object oriented programming ) dan untuk kebutuhan software dan hardware nya adalah

1. Laptop / komputer
2. Web server seperti XAMPP
3. Aplikasi Yii
4. Editor web seperti notepad++ atau sublimeText

Yii juga memiliki beberapa fitur yaitu:

1. **MVC (model view controller)** : Yii menggunakan sebuah arsitektur programing MVC yaitu menggunakan konsep Model, View, Controller. Dan hampir seluruh framework PHP menggunakan konsep MVC termasuk CMS (Content management system).
2. **ORM(Object Relational Mapping)** : Dan untuk fitur berikutnya yang ada di framework Yii adalah Object Relational Mapping atau ORM yang akan mengurusi segala pengelolaan sebuah database. Dengan ORM maka pekerjaan seorang developer akan terbantu karena tidak akan lagi menulis query SQL yang rumit, cukup hanya dengan memanggil fungsi-fungsi tertentu di dalam library Yii.

3. **Widget** : Yang selanjutnya fitur yang ada di Yii framework adalah fitur widget. Widget merupakan konsep Yii yang telah menyediakan komponen-komponen user interface yang siap pakai, misalnya data grid, autocomplete, tree view dan lain-lain. Sebagian widget-widget ini ada yang terintegrasi dengan library Jquery sehingga memungkinkan untuk mekanisme AJAX.
4. **Extension** : Framework Yii memiliki banyak extensions. Extension merupakan kontribusi dari pengguna-pengguna Yii dengan membuat widget, library ataupun component yang bisa dipakai dan belum ada di Yii secara built-in.
5. **Thema** : Dengan framework yii kita dapat merubah tampilan dengan mudah karena Yii framework menggunakan konsep skinning jadi kita bisa mengcostumize tampilan komponen user interface pada aplikasi menggunakan CSS. Sedangkan dengan konsep theming kita bisa merubah seluruh tampilan aplikasi sesuai dengan yang kita inginkan.
6. **Security** : Jika kalian menginginkan sebuah web yang aman dari berbagai cookie atau yang lainnya tenang, karna framework yii mempunyai security yang dapat mencegah serangan-serangan seperti SQL Injection, XSS, CSRF, Cookie Tampering dll.
7. **Web service** : Fitur yang selanjutnya dari Yii framework adalah fasilitas Web service, dengan fitur Web Service kita dapat menggenerate spesifikasi WSDL service secara otomatis.
8. **Translate** : Pemakai Yii framework pasti terdapat di berbagai daerah, dan karna itulah Yii menyediakan fitur translate (penterjemah) untuk pesan, tanggal dan format waktu sesuai zona waktu dan daerah si pemakai.

9. **Caching** : Fitur yang selanjutnya dari Yii framework adalah fitur caching. Dengan fitur caching akan membuat aplikasi yang dibangun menggunakan Yii berjalan ringan ketika diakses.
10. **Autentikasi error** : Untuk kodingan yang sering eror tenang saja karna,Yii menyediakan pesan error dan logging yang bisa memilih untuk menampilkan langsung ataupun disimpan ke suatu file di server. Dengan begitu jika kodingan kita ada yang eror yi akan langsung memberikan pesan eror kepada kita.
11. **Pihak ke tiga** : Yii framework di desain sedemikian rupa sehingga kita bisa memasang library external dari pihak ketiga untuk di sematkan dalam aplikasi,dan kita bisa memasang teknologi bootstap untuk teknologi tampilan yii.

### 1.3.3 CodeIgniter



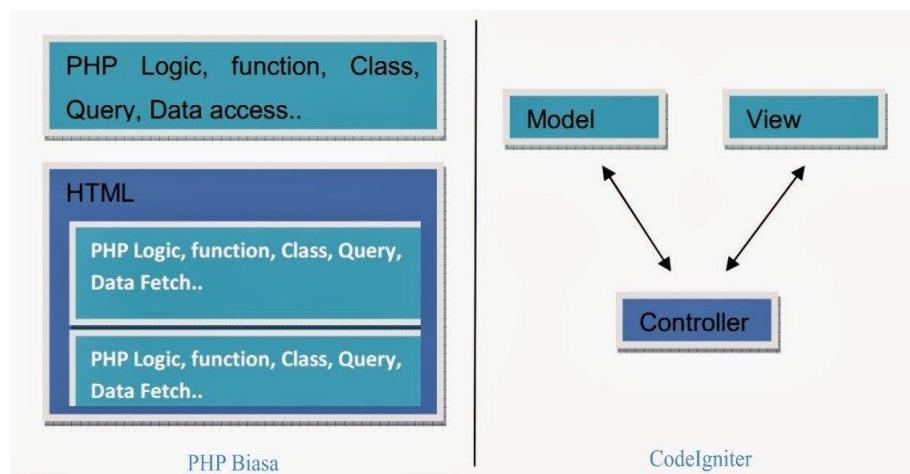
*Gambar 1.5 CodeIgniter*

CodeIgniter adalah sebuah web application network yang bersifat open source yang digunakan untuk membangun aplikasi php dinamis.

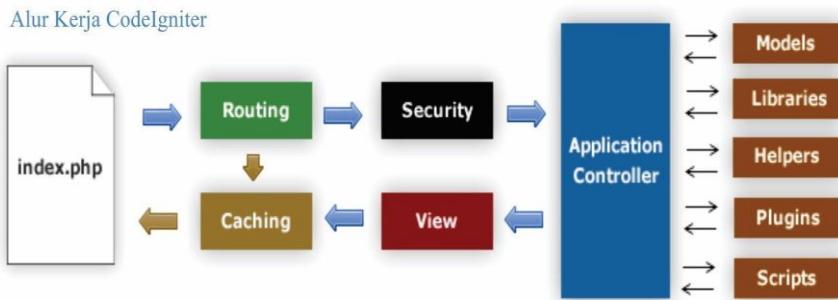
CodeIgniter menjadi sebuah framework PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis

dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi web. Selain ringan dan cepat, CodeIgniter juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kodennya. Dokumentasi yang lengkap inilah yang menjadi salah satu alasan kuat mengapa banyak orang memilih CodeIgniter sebagai framework pilihannya. Karena kelebihan-kelebihan yang dimiliki oleh CodeIgniter, pembuat PHP Rasmus Lerdorf memuji CodeIgniter di frOSCon (Agustus 2008) dengan mengatakan bahwa dia menyukai CodeIgniter karena “it is faster, lighter and the least like a framework.”

CodeIgniter pertamakali dikembangkan pada tahun 2006 oleh Rick Ellis. Dengan logo api yang menyala, CodeIgniter dengan cepat “membakar” semangat para web developer untuk mengembangkan web dinamis dengan cepat dan mudah menggunakan framework PHP yang satu ini.



### 1.3.3.1 Alur kerja framework CodeIgniter



- **Index.php:** Index.php disini berfungsi sebagai file pertama dalam program yang akan dibaca oleh program.
- **The Router:** Router akan memeriksa HTTP request untuk menentukan hal apa yang harus dilakukan oleh program.
- **Cache File:** Apabila dalam program sudah terdapat “cache file” maka file tersebut akan langsung dikirim ke browser. File cache inilah yang dapat membuat sebuah website dapat di buka dengan lebih cepat. Cache file dapat melewati proses yang sebenarnya harus dilakukan oleh program codeigniter.
- **Security:** Sebelum file controller di load keseluruhan, HTTP request dan data yang disubmit oleh user akan disaring terlebih dahulu melalui fasilitas security yang dimiliki oleh codeigniter.
- **Controller:** Controller akan membuka file model, core libraries, helper dan semua resources yang dibutuhkan dalam program tersebut.
- **View:** Hal yang terakhir akan dilakukan adalah membaca semua program yang ada dalam view file dan mengirimkannya ke browser supaya dapat dilihat. Apabila file view sudah ada yang di “cache”

maka file view baru yang belum ter-cache akan mengupdate file view yang sudah ada.

#### 1.3.4 Symfony

Symfony adalah salahsatu dari sekian banyak framework yang ditulis dalam php, symfony di rilis pada tahun 2005 oleh Fabien Potencier. Symfony disusun sesuai standar PSR (PHP standard recommendations) sehingga kualitas dari symfony tidak perlu dipertanyakan lagi. Symfony secara keseluruhan dibuat dengan PHP 5, dan untuk koneksi database, symfony menggunakan doctrine sebagai providernya, symfony juga murni berbasis OOP (object oriented programming), lain halnya dengan php murni, di symfony kita tidak perlu menuliskan lagi kode PHP dari awal, karna symfony akan otomatis menggenerate class object untuk kita.

Framework symfony memiliki beberapa kelebihan, diantaranya:

- Mudah untuk di install dan di konfigurasi pada kebanyakan platform
- Kompatible dengan banyak engine database seperti : MYSQL, ORACLE, MS SQL SERVER, dll)
- Kode yang mudah dibaca
- Mudah untuk dikembangkan
- Disediakan integrasi dari vendor library yang lain.

Selain kelebihan ada juga beberapa kekurangan dalam framework symfony ini

- Tidak mendukung PHP4
- Butuh waktu yang cukup lama untuk mengerti framework ini
- Proses instalasi dan konfigurasi yang cukup rumit

Selain memiliki kelebihan, symfony juga memiliki beberapa fitur yang sangat bermanfaat

## **1. Built-in email library**

Dengan built-in email library, kita tidak perlu lagi menginstall email library untuk mengirim email melalui aplikasi, kita cukup setting dan konfigurasi aplikasi symfony, kita sudah bisa mengirim email dengan cepat dan mudah.

## **2. Templating engine**

Untuk templating engine symfony menggunakan Twig, twig adalah sebuah template engine buatan sensiolabs. Dengan Twig kita dapat dengan mudah berkolaborasi dengan frontend developer karena syntaxnya yang mudah dan friendly sehingga akan membuat proses eksekusi jauh lebih cepat.

## **3. Built-in security**

Untuk security yang dimiliki symfony dibuat Out Of The Box dimana sebuah aplikasi dan sistem keamanan dibuat secara terpisah. Dengan begitu kita bisa membuat aplikasi secara utuh tanpa harus memikirkan lagi keamanan terlebih dahulu. Dan dengan konsep ini kita akan terbebas dari permasalahan user management yang biasanya sering terjadi perubahan yang radikal sehingga sistem yang dibuat harus menyesuaikan dan banyak kode yang harus kita ubah.

## **4. Annotation**

Annotation adalah sebuah block doc PHP yang akan dibaca dan di parsing sebagai program. Dengan ini kita bisa merubah flow aplikasi, menambahkan security, bahkan bisa memanipulasi parameter dengan cara yang sangatlah mudah tanpa merubah code program sama sekali.

## **5. Human friendly configuration**

Untuk jenis konfigurasi yang direkomendasikan oleh symfony adalah jenis konfigurasi yml. Yml adalah singkatan dari yahoo markup language yaitu sebuah markup language yang sangat friendly karena tidak perlu menggunakan tag seperti html maupun xml.

## **6. Backward compatibility**

Bagi seorang developer, backward compatibility sangatlah penting, karena dengan dukungan backward compatibility, seorang developer akan lebih percaya diri dalam mengupgrade program karena tidak akan terjadi masalah baik itu bug ataupun error.

### **1.3.5 Zend Framework**

Zend framework adalah sebuah framework yang bersifat open source untuk membangun dan mengembangkan sebuah aplikasi web dengan PHP. Struktur dan komponen dari zend framework bisa di bilang agak unik karena, setiap komponen telah dirancang dengan beberapa dependensi pada komponen lainnya, zend framework juga sangatlah fleksibel, dan arsitektur yang bisa ditambah sehingga memungkinkan pengembang untuk menggunakan komponen secara individual atau yang sering kita sebut dengan “Use-At-Will” desain.

framework juga menggunakan model MVC dan memiliki beberapa kelebihan diantaranya:

- Dibuat oleh Zend yang merupakan pengembangan engine PHP
- Dibangun diatas teknologi PHP 5
- Lisensi opensource yang berbasis BSD yang sangat fleksibel untuk pengembangan aplikasi
- Memiliki dokumentasi yang lengkap

- Proses pengembangan Zend framework menyertakan unit test yang lengkap
- Libary yang menyediakan fungsi yang lengkap sehingga hampir semua kebutuhan pengembang aplikasi web terpenuhi
- Mendukung pengembangan aplikasi MVC
- URL yang bagus dan search engine yang friendly

Selain keunggulan atau kelebihannya, pastilah zend framework juga memiliki beberapa kekurangan yaitu:

- agak rumit karena menggunakan OOP (object oriented programming)
- proses penginstalannya yang agak rumit.

### **1.3.6 Cake PHP**

CakePHP adalah framework MVC yang *opensource*. Itu membuat mengembangkan, dan maintaining aplikasi lebih mudah. CakePHP memiliki jumlah *libraries* untuk mengurangi kelebihan beban tugas-tugas yang paling umum.

Berikut adalah keuntungan menggunakan CakePHP.

- Open Source
- MVC Framework
- Templating Engine
- Caching operations
- Search Engine Friendly URL
- Memudahkan CRUD (Create, Read, Update, Delete) Database interaksi.
- Libraries dan Helpers
- Built-in validation
- Lokalisasi

- Email, Cookie, keamanan, sesi, dan permintaan yang menangani komponen
- Dapat Melihat bantuan untuk AJAX, JavaScript, HTML dan lain - lain.

### 1.3.7 Fuel PHP

FuelPHP merupakan salah satu *web framework* PHP yang memiliki pengguna cukup banyak. FuelPHP memiliki dukungan yang baik dari komunitas *developer*. FuelPHP merupakan *web framework* yang sederhana dan fleksibel. Dengan portabilitasnya, hampir dapat berjalan di semua *server* dan memiliki kebanggaan karena mempunyai *clean syntax*. Selain memiliki *clean syntax*,

FuelPHP memiliki dokumentasi yang sangat baik dan jelas, sehingga *newbie* sekalipun dapat mempelajari FuelPHP dengan mudah. Kompatibilitasnya dibuat berdasarkan penggunaan fungsional yang ada di PHP. FuelPHP hanya menggunakan ekstensi yang menjadi bagian kode PHP dan terdokumentasi di *website* PHP. Oleh karena itu FuelPHP memiliki portabilitas tinggi.

Tentu saja *web framework* elegan ini dikelola oleh *developer* yang terpilih dan berkomitmen menjadi *mantainer*.

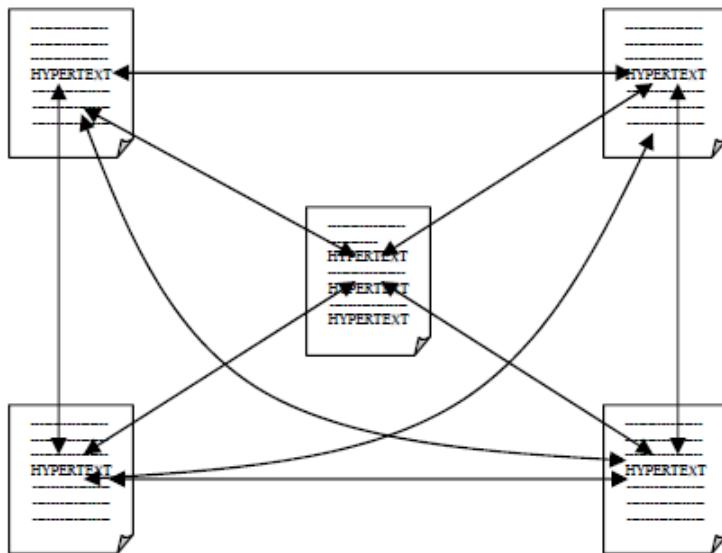
### 1.3.8 Pengantar Web

*Webstite* atau biasa disebut *web* adalah kumpulan dari halaman-halaman situs yang terangkum dalam sebuah *domain* atau *subdomain* yang berada di dalam *World Wide Web (WWW)* yang diakses melalui jaringan internet. *World Wide Web* atau *WWW* adalah gabungan atas semua situs yang dapat diakses publik. Sehingga *WWW* merupakan tempat pengaksesan dari semua situs *Web* yang ada.



*Gambar 1.6 Ilustrasi Web*

Sebuah halaman web biasanya berupa dokumen yang ditulis dalam format *HTML (Hyper Text Markup Language)* yang diakses melalui *HTTP (Hyper Text Transfer Protokol)*, yaitu sebuah protokol yang menyampaikan informasi dari *server website* untuk ditampilkan kepada para pemakai melalui *web browser*.



*Gambar 1.7 Pengaksesan Informasi Melalui Hypertext*

### 1.3.9 Sejarah Web

Awal perkembangan web dimulai pada bulan maret 1989 saat tim berner-lee yang bekerja di laboratorium fisika partikel eropa atau yang dikenal dengan nama *CERN (conseil european pour la recherche nucléaire)* yang terletak di genewa swiss, mengajukan protokol (bahasa atau prosedur yang digunakan untuk menghubungkan antara komputer yang satu dengan lainnya) sistem distribusi informasi internet yang digunakan untuk berbagai informasi di antara para fisikawan.

Protokol inilah yang selanjutnya dikenal sebagai protokol *world wide web* dan dikembangkan oleh *world wide web consortium (w3c)*. w3c adalah konsorsium dari sejumlah organisasi yang berkepentingan dalam pengembangan berbagai standar yang berkaitan dengan web.

*HTTP (hypertext transfer protocol)* merupakan protokol yang digunakan untuk mentransfer data antara *web server* ke *web browser*. Protokol ini mentransfer dokumen-dokumen web yang ditulis atau berformat *HTML (hypertext markup language)*. Dikatakan markup

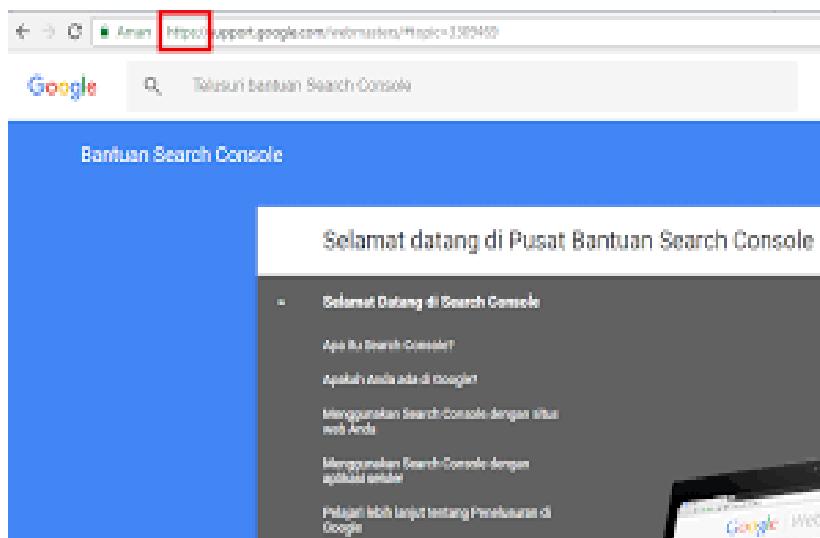
language karena *HTML* berfungsi untuk memperindah file teks biasa untuk ditampilkan pada program web browser. Hal ini dilakukan dengan menambahkan tag-tag (perintah khusus) pada file teks biasa tersebut.

### **1.3.10 HTTP (Hypertext Transfer Protocol) dan HTTPS (Hyper Text Transfer Protocol Secure)**

#### **1. *HTTP (Hypertext Transfer Protocol)***

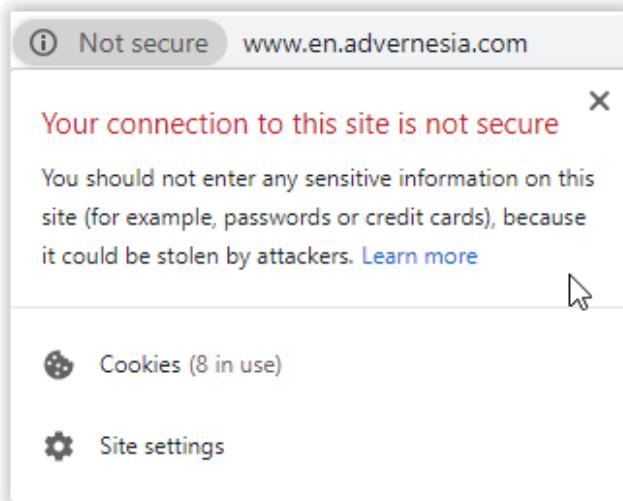
*HTTP* kepanjangan dari *Hyper Text Transfer Protocol*, adalah protokol standar yang digunakan sebuah website untuk melakukan transfer data antar komputer *server* (misalnya server hosting) dengan komputer *client* (komputer yang mengakses website). *HTTP* memegang peranan yang penting untuk mengatur aliran data dari komputer server terkait data apa saja yang akan diberikan kepada komputer *client* dan memberikan instruksi kepada komputer server untuk merespon komunikasi dari komputer *client*.

Semua website menggunakan protokol *HTTP*. Pada umumnya browser tidak menampilkan protokol *HTTP* yang digunakan, melainkan status koneksi website tersebut. *Browser* akan menampilkan status “*Not secure*” atau “**Tidak aman**” untuk protokol *HTTP*.



Gambar 1.8 Contoh situs *http*: Browser Google Chrome menampilkan status website dengan protokol HTTP

Status “*Not secure*” disebabkan karena pengelola *website* hanya menggunakan protokol *HTTP* saja tanpa memberikan perlindungan keamanan data pada *website* tersebut.



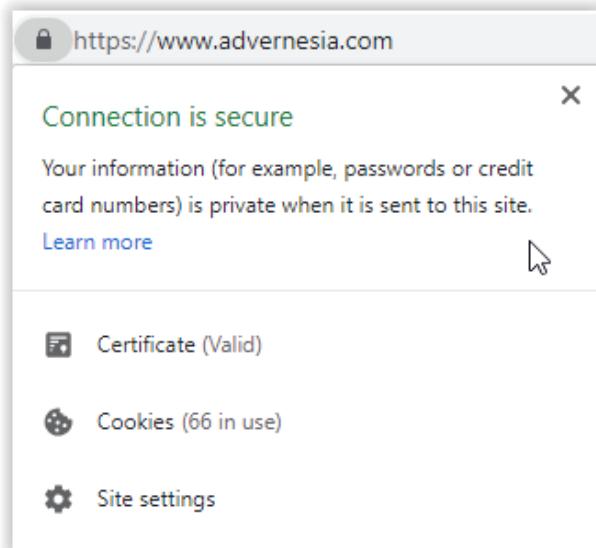
Gambar 1.9 Status “*Not secure*” pada browser Google Chrome

Status tersebut memperingati pengguna website untuk berhati-hati ketika memasukkan data pribadi, akun, password, hingga kartu kredit pada website bersangkutan. Hal ini disebabkan karena protokol HTTP sangat mudah untuk diretas.

## 2. *HTTPS (Hyper Text Transfer Protocol Secure)*

*HTTPS* kepanjangan *Hyper Text Transfer Protocol Secure*, merupakan protokol *HTTP* yang dilengkapi dengan sistem keamanan (security) berupa *SSL (Secure Socket Layer)*. Ibarat kabel telepon, *SSL* merupakan lapisan yang melindungi serat kabel. *HTTPS* memberikan perlindungan aliran data dari serangan peretasan. *HTTPS* dapat melindungi data website, privasi pengguna, akun bank online, akun pengguna website dari tindakan *cybercrime* seperti peretasan dan mengurangi resiko menipu pengguna dengan web tipuan (*phising*). Teknologi *HTTPS* sebenarnya sangat dibutuhkan oleh semua website, tidak hanya website yang dapat menyimpan data pengguna seperti toko online, website bank dan sosial media. Website kecil seperti blog pribadi dan profil perusahaan juga berpotensi dari peretasan.

Browser akan menampilkan status “**Connection is secure**” atau “**Koneksi aman**” saat mengakses suatu website yang menggunakan protokol *HTTPS*.



*Gambar 1.10 Gambar Contoh situs https: Status “Secure” pada browser Google Chrome*

Status “*Connection is secure*” memberikan informasi bahwa website tersebut dapat melindungi data yang dimasukkan pengguna website baik akun, password, atau kartu kredit.

#### 1.4.1 *HTML (Hyper Text Markup Language)*

##### 1. Pengenalan *HTML*

*HTML (Hyper Text Markup Language)* adalah merupakan sebuah dasar ataupun pondasi bahasa pemrograman sebuah webpage, HTML muncul sebagai standar baru dari kemajuan dan berkembangnya internet, pada pertama kali muncul internet masih dalam keadaan berbasis text dimana tampilan sebuah halaman web hanya berisikan sebuah text yang monoton tanpa sebuah format dokumen secara visual, bayangkan saja sebuah dokumen text yang dikemas dalam bungkus format seperti tipe file .txt atau sering disebut notepad, tanpa paragraph, satu warna, satu ukuran huruf tanpa gambar serta tidak adanya visual format dokumen seperti halnya Ms. Word, hal ini akan sangat

membosankan dalam membaca. Dan selain itu pertama kali muncul internet user mengakses masih menggunakan sebuah terminal, hal itu jelas sangatlah tidak friendly. Pemrograman HTML muncul seiring perkembangan teknologi dan informasi.

## 2. Sejarah *HTML*

*Hyper Text Markup Language (HTML)* pertama kali diciptakan dan dikembangkan oleh Tim Berners-Lee pada awal tahun 1990-an yang pada saat itu masih bekerja di CERN. HTML diciptakan dengan tujuan sebagai cara sederhana namun efektif untuk mengkodekan dokumen elektronik. HTML pertama kali dipopulerkan dengan menggunakan browser Mosaic.

Tahun 1980, IBM memikirkan pembuatan suatu dokumen yang akan mengenali setiap elemen dari dokumen dengan suatu tanda tertentu. IBM kemudian mengembangkan suatu jenis bahasa yang menggabungkan teks dengan perintah-perintah pemformatan dokumen. Bahasa ini dinamakan Markup Language, sebuah bahasa yang menggunakan tanda-tanda sebagai basisnya. IBM menamakan sistemnya ini sebagai Generalized Markup Language atau GML.

Tahun 1986, ISO menyatakan bahwa IBM memiliki suatu konsep tentang dokumen yang sangat baik, dan kemudian mengeluarkan suatu publikasi (ISO 8879) yang menyatakan markup language sebagai standar untuk pembuatan dokumen-dokumen. ISO membuat bahasa ini dari GML milik IBM, tetapi memberinya nama lain, yaitu SGML (Standard Generalized Markup Language). ISO dalam publikasinya meyakini bahwa SGML akan sangat berguna untuk pemrosesan informasi teks dan sistem-sistem perkantoran. Tetapi diluar perkiraan ISO, SGML dan terutama subset dari SGML, yaitu HTML juga berguna

untuk menjelajahi internet. Khususnya bagi mereka yang menggunakan World Wide Web.

Mulai pada tahun 1989, sebuah nama HTML muncul dari pemikiran Caillau Tim yang bekerja sama dengan Banners Lee Robert yang ketika itu masih bekerja di CERN memulai mengembangkan bahasa pemrograman ini, dan dipopulerkan pertama kali dengan browser Mosaic. Dan mulailah dari tahun 1990 HTML sangat berkembang dengan cepat hingga mencapai versi HTML versi 5.0 yang digarap pada 4 Maret 2010 kemarin oleh W3C.

Sejarah dari standar HTML:

1. HTML 2.0 (RFC 1866) disetujui sebagai standar 22 September 1995
2. HTML 3.2 14 Januari 1996
3. HTML 4.0 18 Desember 1997
4. HTML 4.01 (minor Fixes) 24 Desember 1999
5. ISO/IEC 15445:2000 (“ISO HTML”, berdasar pada HTML 4.01 Strict) 15 Mei 2000
6. HTML 5 masih dalam draft pengembangan Januari 2008

### 3. Versi *HTML*

#### a. *HTML 1.0*

Ini adalah awal mula dari HTML (pendahulunya). Pada versi ini masih terlihat beberapa kelemahan dan masih sangat sederhana. Kemampuan yang dimiliki oleh versi 1.0 ini hanya terbatas pada heading, paragraph, hypertext, list, dan setak tebal atau miring pada teks.

b. *HTML 2.0*

Versi 2.0 pada 14 Januari 1996, pada versi ini ada beberapa tambahan kemampuan diantaranya penambahan form comment, hal ini menyebabkan adanya sebuah interaktif dan mulai dari versi ini yang menjadikan sebuah pioneer dalam perkembangan homepage interaktif.

c. *HTML 3.0*

Dirilis pada 18 Desember 1997 yang sering disebut sebagai HTML+ yang mempunyai kemampuan dalam beberapa fasilitas diantaranya adalah penambahan fitur table dalam paragraph, akan tetapi versi ini tidak bertahan lama.

d. *HTML 3.2*

Dan pada bulan Mei 1996 dikeluarkan versi baru sebagai pengganti dan penyempurnaan versi 3.0 ini yaitu HTML veri 3.2, keluarnya versi ini dikarenakan adanya beberapa kasus yang timbul pada pengembang browser yang telah melakukan pendekatan dengan cara lain yang justru hal tersebut menjadi popular, maka dibakukan versi 3.2 untuk mengakomodasi praktek yang banyak digunakan oleh pengembang browser dan diterima secara umum, dapat dikatakan bahwa versi 3.2 ini merupakan versi 3.0 yang dikembangkan oleh beberapa pengembang browser seperti Netscape dan Microsoft.

e. *HTML 4.0*

Yang terakhir perombakan terjadi pada tahun 1999 tepatnya tanggal 24 Desember yaitu HTML versi 4.0, seperti yang kita kenal HTML pada saat ini penambahan link, meta, imagemaps. Image dan lain-lain sebagai penyempurnaan versi 3.2. Di samping itu versi ini ditambahkan tag-tag baru seperti ABBR, ACRONYM, BUTTON, PARAM, BUTTON, TBODY, THEAD dan lain sebagainya.

f. *HTML 5.0*

Pada tanggal 4 Maret 2010, terdapat sebuah informasi bahwasannya HTML versi 5.0 masih dikembangkan oleh W3C (World Wide Web Consortium) dan IETF (Internet Engineering Task Force) yaitu sebuah organisasi yang menangani HTML sejak versi 2.0.

4. Struktur Dasar *HTML*

Struktur dasar dokumen HTML adalah sebagai berikut :

```
<html>
<head>
<title>Disini Judul Dokumen HTML</title>
</head>
<body>
```

Disini penulisan informasi Web

```
</body>
</html>
```

Dari struktur dasar HTML di atas dapat dijelaskan sebagai berikut:

a. Tag

Adalah teks khusus (*markup*) berupa dua karakter "<" dan ">", sebagai contoh <body> adalah tag dengan nama body. Secara umum tag ditulis secara berpasangan, yang terdiri atas **tag pembuka** dan **tag penutup** (ditambahkan karakter "/" setelah karakter "<"), sebagai contoh <body> ini adalah tag pembuka isi dokumen HTML, dan </body> ini adalah tag penutup isi dokumen HTML.

b. Element

Element terdiri atas tiga bagian, yaitu tag pembuka, isi, dan tag penutup. Sebagai contoh untuk menampilkan judul dokumen HTML pada web browser digunakan element title, dimana:

ini adalah tag penutup judul dokumen HTML. Tag-tag yang ditulis secara berpasangan pada suatu element HTML, tidak boleh saling tumpang tindih dengan pasangan tag-tag lainnya.

Contoh penulisan tag-tag yang benar

```
<p>  
<b>  
.....  
</b>  
</p>
```

Contoh penulisan tag-tag yang salah

```
<p>  
<b>  
.....  
</p>  
</b>
```

### c. Attribute

Attribute mendefinisikan property dari suatu element HTML, yang terdiri atas nama dan nilai. Penulisannya adalah sebagai berikut:

```
<TAG>  
nama-attr="nilai-attr"  
nama-attr="nilai-attr"  
.....  
>  
.....  
</TAG>
```

Secara umum nilai attribute harus berada dalam tanda petik satu atau dua.

d. Element HTML

Menyatakan pada browser bahwa dokumen Web yang digunakan adalah HTML.

Sintaks:

**<html>**

.....

**</html>**

e. Element HEAD

Merupakan kepala dari dokumen HTML. Tag dan tag terletak di antara tag dan tag.

Sintaks:

**<head>**

.....

**</head>**

f. Element Title

Merupakan judul dari dokumen HTML yang ditampilkan pada judul jendela browser. Tag **<title>** dan tag **</title>** terletak di antara tag **<head>** dan tag **</head>**.

Sintaks:

**<title>**

.....

**</title>**

g. Element Body

Element ini untuk menampilkan isi dokumen HTML. Tag **<body>** dan tag **</body>** terletak di bawah tag **<head>** dan tag **</head>**. Element BODY mempunyai attribute-attribute yang

menspesifikasikan khususnya warna dan latarbelakang dokumen yang akan ditampilkan pada browser.

Sintaks:

```
<body text="v" bgcolor="w" background="uri" link="x"  
alink="y" vlink="z">  
.....  
</body>
```

Attribute text memberikan warna pada teks, bgcolor memberikan warna pada latarbelakang dokumen HTML, background memberikan latar belakang dokumen HTML dalam bentuk gambar, link memberikan nilai warna untuk link, alink memberikan warna untuk link yang sedang aktif, vlink memberikan warna untuk link yang telah dikunjungi. Jika attribute bgcolor dan background keduanya dispesifikasikan maka attribute background yang akan digunakan, akan tetapi jika nilai attribute background (gambar) tidak ditemukan pada dokumen HTML maka attribute bgcolor yang akan digunakan.

## 5. Kelebihan dan Kekurangan *HTML*

Kelebihan *HTML*:

- a. Merupakan bahasa pengkodean yang lintas platform (cross platform), maksudnya HTML dapat digunakan pada berbagai jenis mesin komputer yang berbeda dan berbagai macam sistem operasi yang berbeda. Jadi berdifikat fleksibel karena ditulis cukup dengan menggunakan editor karakter ASCII.
- b. Dapat disisipi gambar baik gambar statis atau dinamis (animasi) termasuk menggunakan gambar untuk dijadikan hyperlink. Gambar disini digunakan untuk merujuk pada suatu halaman web, dimana setiap titik-titik yang sudah didefinisikan berupa rectangular (kotak),

poligon (kurva tak beraturan) atau lingkaran digunakan untuk ‘jump’ ke halaman lain, atau link ke halaman di luar web yang bersangkutan.

- c. Dapat disisipi animasi berupa Java Applet atau file-file animasi dari Macromedia Flash atau Macromedia Shockwave (untuk keperluan ini, browser harus memiliki plug-in khusus untuk menjalankan file-file animasi ini).
- d. Dapat disisipi bahasa pemrograman untuk mempercantik halaman web seperti Javascript, Vbscript, Active Server Pages, Perl, Tcl, PHP, dan sebagainya.
- e. Bukan merupakan bahasa pemrograman jadi tidak memerlukan kompiler. Cara menjalankannya cukup dengan menggunakan browser.

**Kekurangan HTML:**

- a. Menghasilkan halaman yang statis, untuk memperoleh halaman yang dinamis harus menggunakan bahasa pemrograman tertentu seperti Javascript atau Vbscript dan animasi seperti Flash atau Shockwave.
- b. Memiliki tag-tag yang begitu banyak sehingga susah dipelajari untuk yang masih awam.
- c. Tidak dapat menghasilkan halaman yang interaktif. Interaktif disini maksudnya client dapat berinteraksi dengan server. Untuk keperluan itu HTML harus disisipi bahasa pemrograman yang dapat menangani hal tersebut, contohnya Perl dan Tcl.

### **1.3.11 Hubungan *HTML* dan *PHP***

Halaman web biasanya disusun dari kode-kode html yang disimpan dalam sebuah file berekstensi .html. File html ini dikirimkan oleh server (atau file) ke browser, Kemudian browser menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah. Lain halnya dengan program php, program ini harus diterjemahkan oleh web-server sehingga menghasilkan kode html yang dikirim ke browser agar dapat ditampilkan.

Program ini dapat berdiri sendiri ataupun disisipkan di antara kode-kode html sehingga dapat langsung ditampilkan bersama dengan kode-kode html tersebut. Program php dapat ditambahkan dengan mengapit program tersebut diantara tanda.

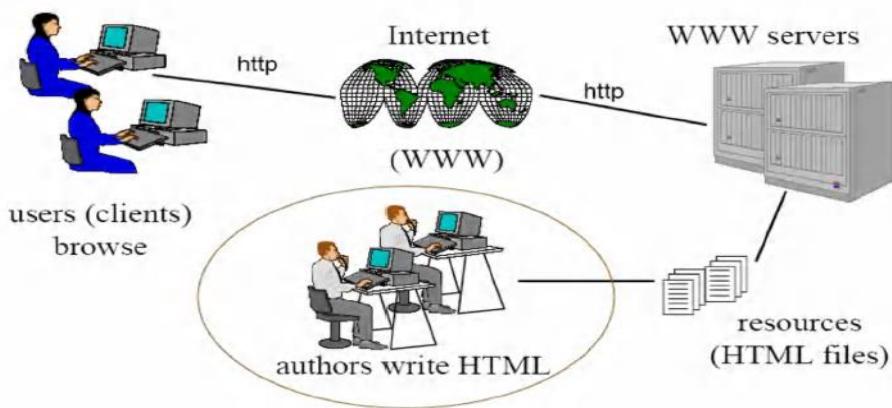
Tanda-tanda tersebut biasanya disebut tanda untuk escaping (kabur) dari kode html. File html yang telah dibubuhinya program php harus diganti ekstensi-nya menjadi .php3 atau php. Php merupakan bahasa pemograman web yang bersifat server-side HTML=embedded scripting, di mana script-nya menyatu dengan HTML dan berada si server. Artinya adalah sintaks dan perintah-perintah yang kita berikan akan sepenuhnya dijalankan di server tetapi disertakan HTML biasa. PHP dikenal sebagai bahasa scripting yang menyatu dengan tag HTML, dieksekusi di server dan digunakan untuk membuat halaman web yang dinamis seperti ASP (Active Server Pages) dan JSP (Java Server Pages).

### 1.3.12 Pemahaman Pemrograman Web

1. Pemrograman: Suatu usaha menulis sesuatu perintah (program aplikasi) sehingga computer dapat menjalankan apa yang kita inginkan.
2. Pemrograman Web: Membuat program aplikasi berbasis web
3. Aplikasi berbasis web: Aplikasi yang dibuat dengan memanfaatkan mekanisme dan aplikasi yang sudah ada pada sistem web (WWW).

### 1.4.2 Membuat Aplikasi Berbasis Web

1. Memperkaya fungsi web server dengan cara menambahkan program pada dokumen yang akan dieksekusi oleh server ketika file dokumen web tersebut diakses oleh web server.  
Misalnya: Program yang mengambil data ke basis data untuk ditampilkan ke web browser.
2. Memperkaya interaktivitas dokumen dengan cara menambahkan program pada dokumen yang akan dieksekusi oleh web browser ketika file dokumen tersebut ditampilkan oleh web browser.  
Misalnya: Program yang memvalidasi data masukan pada form sebelum disubmit ke web server.



Gambar 1.11 Gambar Cara Kerja Web

### **1.4.3 Keunggulan Dan Kekurangan Aplikasi Berbasis Web**

1. Keunggulan
  - a. Dapat diakses kapanpun dan dari mana pun selama ada koneksi internet.
  - b. Dapat diakses hanya dengan menggunakan browser (umumnya sudah tersedia di PC, PDA, dan handphone).
  - c. Tidak perlu menginstal aplikasi client khusus.
2. Kekurangan
  - a. Antarmuka yang dibuat terbatas sesuai spesifikasi standar untuk membuat dokumen web.
  - b. Keterbatasan kemampuan web browser untuk menampilkannya.
  - c. Terbatasnya kecepatan internet mungkin membuat respon aplikasi menjadi lebih lambat.

### **1.3.13 Web Statis dan Web Dinamis**

#### **1. Web Statis**

Web statis ialah web yang berisi tentang informasi yang memiki sifat statis (tetap) atau pengguna tidak dapat berinteraksi dengan website tersebut, web statis dapat dilihat dari tampilan website tersebut jika suatu web hanya berhubungan dengan halaman web lain yang berisi informasi tetap maka web tersebut termasuk kedalam kategori web statis, pada web statis pengguna hanya dapat melihat isi web tersebut dan jika di klik hanya akan berpindah pada halaman lainnya. Dalam web statis interaksi pengguna sangatlah terbatas

## 2. Web Dinamis

Web dinamis adalah web yang dapat menampilkan informasi serta dapat membuat pengguna berinteraksi seperti dengan form input, button sehingga dapat mengolah informasi yang di tampilkan pada web tersebut, web dinamis bersifat tidak kaku dan terlihat lebih enak dipandang.

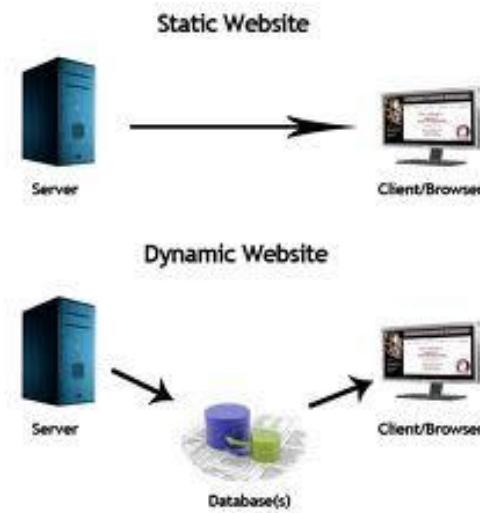
## 3. Perbedaan Web Statis dan Web Dinamis

### a. Web Statis

1. Sebagian besar halaman web statis.
2. Isi (teks/link/gambar) yang sama setiap kali diakses *HyperText Markup Language (HTML)* digunakan untuk menetukan teks/ format gambar.
3. Contoh: dokumen online, kebanyakan homepage.

### b. Web Dinamis

1. Sebagai web dinamis mengarah ke layanan online/ e-commerce.
2. Halaman webnya juga harus menyediakan konten dinamis.
3. Halaman harus update, berubah-ubah (misalnya: berputar banner, artikel ganti)
4. Harus mampu bereaksi terhadap tindakan info pengguna, permintaan dan proses, pemesanan jasa, dan lain-lain.



*Gambar 1.12 Gambar Web Statis vs Web Dinamis*

### 1.3.14 Basis Data

#### 1. Pengertian Basis Data

Basis Data terdiri dari kata basis dan data. Basis dapat diartikan sebagai markas atau gudang. Sedangkan data adalah catatan atas kumpulan fakta dunia nyata yang mewakili objek seperti manusia, barang, hewan, konsep, peristiwa dan sebagainya yang diwujudkan dalam bentuk huruf, angka, simbol, gambar, teks, bunyi atau kombinasinya.

Sebagai suatu kesatuan maka pengertian basis data atau biasa disebut *database* adalah sebagai berikut:

#### Pengertian Basis Data atau *Database*

1. Himpunan kelompok data yang saling terhubung dan diorganisasi sedemikian rupa supaya kelak dapat dimanfaatkan kembali secara cepat dan mudah.
2. Kumpulan data dalam bentuk file/tabel/arsip yang saling berhubungan dan tersimpan dalam media penyimpanan elektronis,

untuk kemudahan dalam pengaturan, pemilahan, pengelompokan dan pengorganisasian data sesuai tujuan.

Dengan basis data seseorang dapat menyimpan sebuah informasi, seperti data mahasiswa, kepegawaian atau produk ke dalam media penyimpanan elektronis seperti cakram magnetis (*disk*) melalui perangkat komputer, Untuk kemudian data tersebut dapat kita gunakan sesuai keperluan. Database mempunyai 8 operasi dasar diantaranya adalah *Create database, Drop database, create table, Drop table, Insert, Read, Update dan Delete.*

## 2. Pengertian Sistem Basis Data

Sistem basis data adalah sistem yang terdiri atas kumpulan tabel data yang saling berhubungan dan kumpulan program yang memungkinkan beberapa pemakai atau program lain untuk mengakses dan memanipulasi tabel tabel data tersebut.

## 3. Komponen Sistem Basis Data

### a. Perangkat Keras

Perangkat keras atau hardware yang umumnya terdapat dalam sistem basis data adalah komputer, hard disk, memori sekunder offline (removable disk, fd), perangkat komunikasi jaringan.

### b. Sistem Operasi

Sistem operasi adalah program yang dirancang untuk mengaktifkan sistem komputer dan mengendalikan seluruh sumber daya yang ada di dalamnya termasuk operasi- operasi dasar komputer. seperti Windows, Unix dan Linux.

### c. Basis Data

Komponen adalah sekumpulan data yang terorganisir dengan baik sehingga data tersebut mudah disimpan, diakses, dan juga dapat dimanipulasi.

Sistem basis data dapat terdiri dari beberapa basis data yang memiliki data masing-masing.

d. Database Management System (DBMS)

DBMS atau database management system adalah program aplikasi khusus yang dirancang untuk membuat dan juga mengelola database yang tersedia. Sistem ini berisi koleksi data dan set program yang digunakan untuk mengakses database tersebut.

DBMS adalah software yang berperan dalam mengelola, menyimpan, dan mengambil data kembali. Adapun mekanisme yang digunakan sebagai pelengkap adalah pengaman data, konsistensi data dan pengguna data bersama.

**Contoh dari DBMS** adalah *Microsoft Access, MySql, Oracle database, Sybase, Borland-Interbasi, PostgreSQL dll.*

e. Pemakai atau User

User adalah salah satu komponen database yang berinteraksi secara langsung dengan database. Ada beberapa tipe user, diantaranya, programmer aplikasi, User mahir (casual user), user umum (end user) dan user khusus (specialized user)

f. Aplikasi atau Perangkat Lain

Aplikasi ini tergantung kebutuhan, pemakai basis data bisa dibuatkan program khusus untuk melakukan pengisian, pengubahan atau pengambilan data yang mudah dalam pemakaiannya. Program tersebut ada yang tersedia langsung dalam DBMS atau dibuat menggunakan aplikasi lain seperti misalnya Visual Basic.

#### 4. Bahasa Basis Data (Database Language)

Bahasa database merupakan bahasa data yang dapat ditempelkan kedalam bahasa pemrograman yang lain, sebut saja Java, Pascal, Fortran dst. Bahasa dimana instruksi data base menempel disebut inang. Beberapa komponen Bahasa data base menurut fungsinya dibagi menjadi:

##### a. *Data Definition Language*

*Data definition language* adalah sekumpulan definisi yang disimpan di dalam data dictionary.

##### b. *Data Manipulation Language*

*Data Manipulation Language* berisi akumulasi dari operasi manipulasi basis data yang dilakukan. Ini biasa disebut dengan bahasa query sebab biasanya digunakan untuk meminta informasi yang ada dari basis data tersebut.

#### 5. Fungsi dan Tujuan Basis Data

Fungsi basis data cukup banyak dan cakupannya pun luas dalam mendukung keberadaan lembaga atau organisasi, diantaranya adalah:

##### a. Ketersediaan/ *Availability*

Fungsi basis data yang pertama adalah untuk menyediakan data-data penting saat sedang diperlukan. Ya, ini adalah fungsi penting dari basis data yang meskipun tidak terletak dalam satu lokasi, dan tersimpan dalam bentuk disk, akan tetapi dengan cara penyimpanan yang sistematik, informasi tersebut mudah untuk didapatkan.

*b. Mudah dan Cepat/ Speed*

Selanjutnya, fungsi dari basis data ini adalah agar Anda sebagai pengguna bisa dengan mudah mengaksesnya saat sedang membutuhkan. Tidak perlu tunggu nanti, apalagi harus mengalokasikan waktu tertentu untuk memanggilnya.

*c. Kelengkapan/ Completeness*

Basis data harus menyimpan data yang lengkap, yang bisa melayani keperluan penggunanya secara keseluruhan. Meski kata lengkap yang dipakai disini sifatnya relatif, namun setidaknya data tersebut membantu memudahkan untuk menambah koleksi data, dan menjamin mudahnya pengguna untuk memodifikasi struktur data yang ada, sebut saja field-field data yang tersedia

*d. Accuracy dan Security*

Fungsi data base selanjutnya adalah untuk accuracy atau keakuratan. Jadi, agar kesalahan dapat ditekan semaksimal mungkin, Anda bisa lakukan pengorganisasian file-file database dengan baik untuk menghindari kesalahan pada proses data entry dan juga dalam proses penyimpanan atau datastore.

Selain itu, fungsi database adalah untuk security atau keamanan. Ada fasilitas pengaman data yang disediakan oleh sistem basis data yang baik sehingga data tidak bisa dimodifikasi, diakses, diubah maupun dihapus oleh yang tidak mendapatkan hak untuk melakukannya.

*e. Storage Efficiency*

Pengorganisasian data dilakukan dengan baik dengan tujuan untuk menghindari duplikasi data yang berpengaruh pada bertambahnya ruang penyimpanan dari basis data tersebut.

pengkodean dan juga relasi data bermanfaat untuk menghemat space penyimpanan dalam basis data.

## **BAB II**

### **PENGENALAN FRAMEWORK CODE IGNITER DAN NOTIFIKASI TELEGRAM**

#### **2. Pengenalan Framework codeigniter dan notifikasi telegram**

Pada Bab II ini akan dijelaskan mengenai Framework CodeIgniter dan Notifikasi lewat Telegram yang akan digunakan dalam pembuatan aplikasi ini

##### **2.1 Framework CodeIgniter**



*Gambar 2.1 Framework CodeIgniter*

CodeIgniter adalah sebuah web application network yang bersifat open source yang digunakan untuk membangun aplikasi php dinamis.

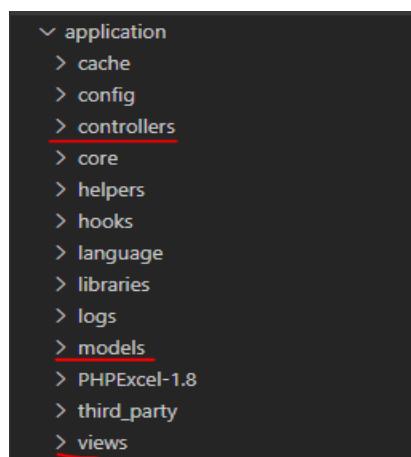
##### **2.2 Sejarah CodeIgniter**

CodeIgniter adalah sebuah framework berbasis PHP yang kuat dengan footprint yang sangat kecil, dibangun untuk pengembang yang membutuhkan toolkit sederhana dan elegan untuk membuat aplikasi web dengan fitur lengkap. Framework atau dalam bahasa indonesia dapat diartikan sebagai "kerangka kerja" merupakan sebuah tool yang bekerja pada suatu konsep tertentu dan terdiri dari

berbagai fungsi yang dapat dengan mudah dijabarkan atau dialihgunakan untuk membuat fungsi-fungsi lain yang lebih kompleks. Maksudnya, jika seorang Programmer menggunakan sebuah framework sebagai lingkup kerjanya, dia harus mematuhi segala ketentuan dari framework tersebut. Lebih lanjut ia dapat membangun berbagai fungsi yang rumit dan kompleks dengan menggunakan fungsi-fungsi yang telah disediakan oleh Framework tersebut. Jadi dia tidak perlu lagi menulis ulang semua coding, Ia cukup mempelajari cara menggunakan kerangka (frame) yang telah tersedia dari framework tersebut.

CodeIgniter merupakan aplikasi sumber terbuka yang berupa framework PHP dengan model MVC (Model, View, Controller) untuk membangun websitedinamis dengan menggunakan PHP. CodeIgniter memudahkan developer untuk membuat aplikasi web dengan cepat mudah dibandingkan dengan membuatnya dari awal. CodeIniter dirilis pertama kali pada 28 Februari 2006. Versi stabil terakhir adalah versi 3.0.4

### 2.3 Konsep MVC Pada CodeIgniter



Gambar 2.2 Konsep MVC

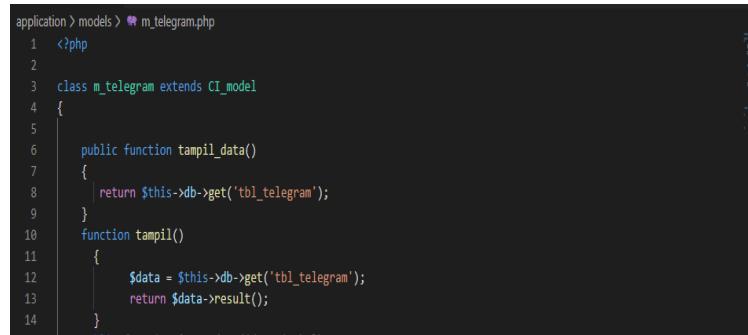
## **1. Pengertian MVC**

MVC adalah konsep dasar yang harus diketahui sebelum mengenal CodeIgniter. MVC ( Model, View, Controller) merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web, berawal pada bahasa pemrograman Small Talk, yang memisahkan bisnis logic (alur piker), data logic (penyimpanan data) dan presentation logic (antarmuka aplikasi) atau secara sederhana adalah memisahkan antara desain, data dan proses. Ada 3 komponen yang membangun suatu MVC yaitu :

- a. Model,

Biasanya berhubungan dengan data dan interaksi ke database atau webservice. Model juga merepresentasikan struktur data dari aplikasi yang bisa berupa basis data maupun data lain, misalnya dalam bentuk file teks, file XML maupun webservice. Biasanya didalam model akan berisi class dan fungsi untuk mengambil, melakukan update dan menghapus data website. Sebuah aplikasi web biasanya menggunakan basis data dalam menyimpan data, maka pada bagian Model biasanya akan berhubungan dengan perintah-perintah query SQL.

## Contoh tampilan Model pada CodeIgniter :



```
application > models > m_telegram.php
1  <?php
2
3  class m_telegram extends CI_model
4  {
5
6      public function tampil_data()
7      {
8          | return $this->db->get('tbl_telegram');
9      }
10     function tampil()
11     {
12         $data = $this->db->get('tbl_telegram');
13         return $data->result();
14     }

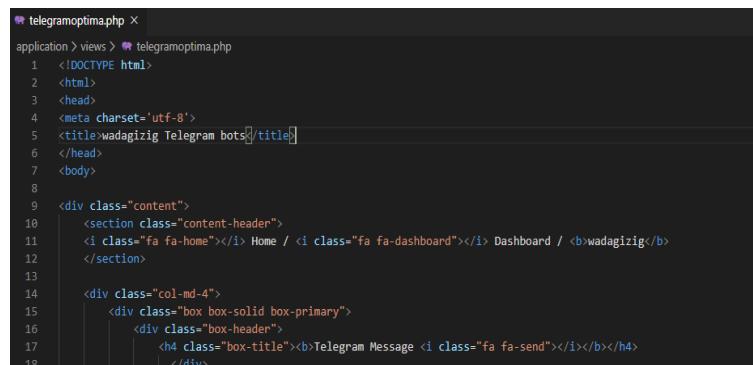
```

*Gambar 2.3 tampilan model*

## b. View,

Merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML, yang diatur oleh controller. View berfungsi untuk menerima dan merepresentasikan data hasil dari model dan controller kepada user. View tidak memiliki akses langsung terhadap bagian model.

Contoh tampilan View pada CodeIgniter :



```
application > views > telegramoptima.php
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <meta charset='utf-8'>
5  <title>wadagizig Telegram bots</title>
6  </head>
7  <body>
8
9  <div class="content">
10    <section class="content-header">
11      <i class="fa fa-home"></i> Home / <i class="fa fa-dashboard"></i> Dashboard / <b>wadagizig</b>
12    </section>
13
14    <div class="col-md-4">
15      <div class="box box-solid box-primary">
16        <div class="box-header">
17          <h4 class="box-title"><b>Telegram Message <i class="fa fa-send"></i></b></h4>
18        </div>

```

*Gambar 2.4 tampilan view*

c. Controller,

Merupakan bagian yang mengatur hubungan antara bagian model dan bagian view. Pada controller terdapat class-class dan fungsi-fungsi yang memproses permintaan dari View ke dalam struktur data di dalam model. Controller juga tidak boleh berisi kode untuk mengakses basis data Karena tugas megakses data telah diserahkan kepada model. Tugas controller adalah menyediakan berbagai variable yang akan ditampilkan di view, memanggil model untuk melakukan akses ke basis data, menyediakan penanganan kesalahan/error, mengerjakan proses logika dari aplikasi serta melakukan validasi atau cek terhadap input.

Contoh tampilan Controller pada CodeIgniter :

```
application > controllers > home.php
1  <?php
2
3  class home extends CI_Controller{
4
5      function __construct(){
6          parent::__construct();
7          $this->load->model('M_sales');
8          $this->load->helper('url');
9          $this->load->library(array('googlemaps'));
10     }
11 }
```

Gambar 2.5 tampilan controller

## 2. Fungsi CodeIgniter

- Mempercepat dan mempermudah kita dalam pembuatan website.
- Menghasilkan struktur pemrograman yang sangat rapi, baik dari segi kode maupun struktur file phpnya.
- Memberikan standar coding sehingga memudahkan kita atau orang lain untuk mempelajari kembali system aplikasi yang dibangun.

### **3. Kelebihan CodeIgniter**

1. Berukuran sangat kecil. File download nya hanya sekitar 2MB, itupun sudah included dokumentasinya yang sangat lengkap.
2. Dokumentasi yang bagus. Saat anda mendownloadnya, telah disertakan dengan dokumentasi yang berisi pengantar, tutorial, bagaimana panduan penggunaan, serta referensi dokumentasi untuk komponen-komponennya.
3. Kompitabilitas dengan Hosting. CodeIgniter mampu berjalan dengan baik pada hampir semua platform hosting. CodeIgniter juga mendukung database-database paling umum, termasuk MySQL.
4. Tidak ada aturan coding yang ketat. Terserah anda jika anda hanya ingin menggunakan Controller, tanpa View, atau tidak menggunakan Model, atau tidak salah satu keduanya. Namun dengan menggunakan ketiga komponennya adalah pilihan lebih bijak.
5. Kinerja yang baik. Codeigniter sangat cepat bahkan mungkin bisa dibilang merupakan framework yang paling cepat yang ada saat ini.
6. Sangat mudah diintegrasikan. CodeIgniter sangat mengerti tentang pengembangan berbagai library saat ini. Karenanya CodeIgniter memberikan kemudahan untuk diintegrasikan dengan library-library yang tersedia saat ini.

7. Sedikit Konfigurasi. Konfigurasi CodeIgniter terletak di folder application/config. CodeIgniter tidak membutuhkan konfigurasi yang rumit, bahkan untuk mencoba menjalankannya, tanpa melakukan konfigurasi sedikitpun ia sudah bisa berjalan.
8. Mudah dipelajari. Disamping dokumentasi yang lengkap, ia juga memiliki berbagai forum diskusi.

#### **4. Kekurangan CodeIgniter**

1. CodeIgniter tidak ditujukan untuk pembuatan web dengan skala besar.
2. Library yang sangat terbatas. Hal ini dikarenakan sangat sulit mencari plugin tambahan yang terverifikasi secara resmi, karena pada situsnya CodeIgniter tidak menyediakan plugin-plugin tambahan untuk mendukung pengembangan aplikasi dengan CI.
3. Belum adanya editor khusus CodeIgniter, sehingga dalam melakukan create project dan modul-modulnya harus berpindah-pindah folder.

### **2.4 Notifikasi Telegram**

#### **1. Bot Telegram**

- Telegram adalah sebuah sistem perpesanan yang lintas *platform* dan berpusat pada keamanan dan kerahasiaan pribadi penggunanya.
- Bot adalah program komputer yang melakukan pekerjaan tertentu secara otomatis.

Aplikasi perpesanan pastinya telah jamak dalam kehidupan sehari-hari kita, namun mengapa ada *bot* dalam Telegram? Bot adalah sebuah mesin, dibuat untuk meringankan pekerjaan manusia. Dalam kaitan bot dengan Telegram, bayangkan jika Anda ingin mengirimkan sebuah gambar ke teman bicara dalam Telegram, alih-alih membuka peramban web dan mengetikkan kata kunci di mesin pencari kemudian mengirimkan hasilnya di Telegram, Anda tinggal memanggil bot dan mengetikkan kata kunci semisal: [/img <gambar\_yang\_dicari>], maka hasilnya langsung keluar dalam aplikasi Telegram.

Bot pencari gambar hanyalah salah satu kegunaan bot yang sederhana. Bot bisa digunakan untuk mencari di mesin pencari, wiki, youtube, torrent, cuaca, nilai tukar, alih bahasa, pengaturan grup, dan bahkan program yang sangat rumit.

Daya tarik utama Telegram adalah ia dapat dijalankan pada beragam perangkat dan sistem operasi, tidak hanya telepon genggam, namun juga komputer dan perangkat pintar serupa komputer lainnya. Telegram dan bot dapat memudahkan kehidupan keseharian kita tanpa harus terpaku di depan komputer.

## **BAB III**

### **PENJELASAN TOOLS YANG DIGUNAKAN DAN BAHASA PEMROGRMAN YANG DIGUNAKAN**

#### **3. Penjelasan Tools Yang Digunakan Dan Bahasa Pemrograman Yang Digunakan**

Tools yang digunakan digunakan dalam pembuatan aplikasi ini adalah :

- 1. XAMPP**
- 2. Visual Studio Code**
- 3. Bot Chat Telegram**

Bahasa pemrograman yang digunakan adalah :

- 1. PHP**

##### **3.1 XAMPP**



*Gambar 3.1 XAMPP*

XAMPP adalah perangkat lunak (free software) bebas, yang mendukung untuk banyak sistem operasi, yang merupakan kompilasi dari beberapa program. Fungsi XAMPP sendiri adalah sebagai server

yang berdiri sendiri (localhost), yang terdiri beberapa program antara lain : Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl.

Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah untuk digunakan yang dapat menampilkan halaman web yang dinamis. Untuk mendapatkanya XAMPP anda dapat mendownload langsung dari web resminya. Dan berikut beberapa definisi program lainnya yang terdapat dalam XAMPP.

Nama XAMPP sendiri merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Masing-masing huruf yang ada di dalam nama XAMPP memiliki arti sebagai berikut ini:

- a. X = Cross Platform

Merupakan kode penanda untuk software cross platform atau yang bisa berjalan di banyak sistem operasi.

- b. A = Apache

Apache adalah aplikasi web server yang bersifat gratis dan bisa dikembangkan oleh banyak orang (open source).

- c. M = MySQL / MariaDB

MySQL atau MariaDB merupakan aplikasi database server yang dikembangkan oleh orang yang sama. MySQL berperan dalam mengolah, mengedit, dan menghapus daftar melalui database.

- d. P = PHP

Huruf “P” yang pertama dari akronim kata XAMPP adalah inisial untuk menunjukkan eksistensi bahasa pemrograman

PHP. Bahasa pemrograman ini biasanya digunakan untuk membuat website dinamis, contohnya dalam website berbasis CMS WordPress.

e. P = Perl

Sementara itu, untuk huruf P selanjutnya merupakan singkatan dari bahasa pemrograman Perl yang kerap digunakan untuk memenuhi berbagai macam kebutuhan. Perl ini bisa berjalan di dalam banyak sistem operasi sehingga sangat fleksibel dan banyak digunakan.

Berikut beberapa definisi program lainnya yang terdapat dalam XAMPP

1. Apache

Fungsi dari Apache yaitu menampilkan halaman web sesuai dengan script php yang telah dibuat sebelumnya.

2. PHP

Merupakan bahasa pemrograman yang sering digunakan untuk web server. PHP memungkinkan pengguna atau user untuk membuat web dinamis (dapat diubah). PHP dapat berkerja di berbagai macam Operating System seperti, windows, linux, mac os, dan lainnya.

3. MySQL

SQL atau Structured Query Language merupakan software yang khusus digunakan untuk mengolah database. Hal ini memungkinkan SQL untuk dapat menambah, mengubah, menghapus data yang terdapat dalam database. SQL merupakan software yang bersifat rational atau dalam artian program ini menggunakan tabel data untuk memisahkan beberapa data yang memungkinkan untuk menghindari duplicate data.

## 4. PHPmyAdmin

Dengan fitur PHPmyAdmin ini, kita akan dapat dengan mudah membuat baris data ataupun database tanpa harus mengingat perintah-perintahnya.

Server HTTP Apache atau Server Web/WWW Apache merupakan server web yang dapat dijalankan di banyak sistem operasi seperti (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta platform lainnya) yang berguna untuk melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani fasilitas.

### 3.1.1 Fungsi XAMPP

Program aplikasi XAMPP berfungsi sebagai server lokal untuk mengampu berbagai jenis data website yang sedang dalam proses pengembangan. Dalam prakteknya, XAMPP bisa digunakan untuk menguji kinerja fitur ataupun menampilkan konten yang ada didalam website kepada orang lain tanpa harus terkoneksi dengan internet, atau istilahnya website offline. XAMPP bekerja secara offline layaknya web hosting biasa namun tidak bisa diakses oleh banyak orang. Maka dari itu, XAMPP biasanya banyak digunakan oleh para mahasiswa maupun pelajar untuk melihat hasil desain website sebelum akhirnya dibuat online menggunakan web hosting yang biasa dijual dipasaran.

### **3.1.2 Komponen XAMPP**

Berikut ini terdapat beberapa komponen xampp, terdiri atas:

*XAMPP 1.8.3 untuk Windows, Termasuk :*

- Apache 2.4.4
- MySQL 6.5.11
- PHP 5.5.0
- phpMyAdmin 4.0.4
- FileZilla FTP Server 0.9.41
- Tomcat 7.0.41 (with mod\_proxy\_ajp as connector)
- Strawberry Perl 5.16.3.1 Portabel
- XAMPP Control Panel 3.2.1 (dari hackattack142)

*XAMPP 1.8.3 untuk Linux, Termasuk :*

- Apache 2.4.4
- MySQL 6.5.11
- PHP 5.5.0
- phpMyAdmin 4.0.4
- OpenSSL 1.0.1e

### 3.1.3 Versi XAMPP

Berikut ini merupakan daftar versi dari XAMPP

XAMPP	Apache	PHP 5	PHP 4	MySQL
1.8.3	2.4.4	5.5.3		5.6.11
1.8.2	2.4.4	5.4.19		5.5.32
1.8.1	2.4.3	5.4.7		5.5.27
1.8.0	2.4.2	5.4.4		5.5.25a
1.7.7	2.2.21	5.3.8		5.5.16
1.7.5	2.2.21	5.3.8		5.5.15
1.7.4	2.2.17	5.3.5		5.5.8
1.7.3	2.2.14	5.3.1		5.1.41
1.7.2	2.2.12	5.3.0		5.1.37
1.7.1	2.2.11	5.2.9		5.1.33
1.7.0	2.2.11	5.2.8		5.1.30
1.6.8	2.2.9	5.2.6	4.4.9	5.0.67
1.6.7	2.2.9	5.2.6	4.4.8	5.0.51b
1.6.6a	2.2.8	5.2.5	4.4.8	5.0.51a
1.6.6	2.2.8	5.2.5	4.4.8 (RC2)	5.0.51
1.6.5	2.2.6	5.2.5	4.4.7	5.0.51
1.6.4	2.2.6	5.2.4	4.4.7	5.0.45
1.6.3a	2.2.4	5.2.3	4.4.7	5.0.45
1.6.3	2.2.4	5.2.3	4.4.7	5.0.54
1.6.2	2.2.4	5.2.2	4.4.7	5.0.41
1.6.1	2.2.4	5.2.1	4.4.6	5.0.37
1.6.0a	2.2.4	5.2.1	4.4.5	5.0.33
1.6.0	2.2.3	5.2.1	4.4.5	5.0.33

*Gambar 3.1 Versi XAMPP*

### **3.1.4 Kelebihan XAMPP**

Kelebihan XAMPP antara lain:

- a. Database Storage Engine ini banyak digunakan oleh programmer apalagi oleh web developer karena sifatnya yang free. Untuk yang expert sudah ada yang bayar.
- b. Kemampuannya sudah bisa diandalkan, mempunyai kapasitas yang cukup mumpuni sekitar 60.000 tabel dengan jumlah record mencapai 5.000.000.000 bahkan untuk yang terbaru sudah lebih.
- c. Keamanan datanya cukup aman walaupun tidak sehebat Postgre apalagi Oracle.
- d. Engine ini multiplatform sehingga mampu diaplikasikan di berbagai sistem operasi. My Sql cocok diaplikasikan diaplikasi kelas kecil dan menengah.
- e. Kelebihan paling utama engine ini adalah kecepatannya.

### **3.1.5 Kekurangan XAMPP**

Kekurangan XAMPP antara lain:

- a. Tidak cocok untuk menangani data dengan jumlah yang besar, baik untuk menyimpan data maupun untuk memproses data.
- b. Memiliki keterbatasan kemampuan kinerja pada server ketika data yang disimpan telah melebihi batas maksimal kemampuan daya tampung server karena tidak menerapkan konsep Technology Cluste.

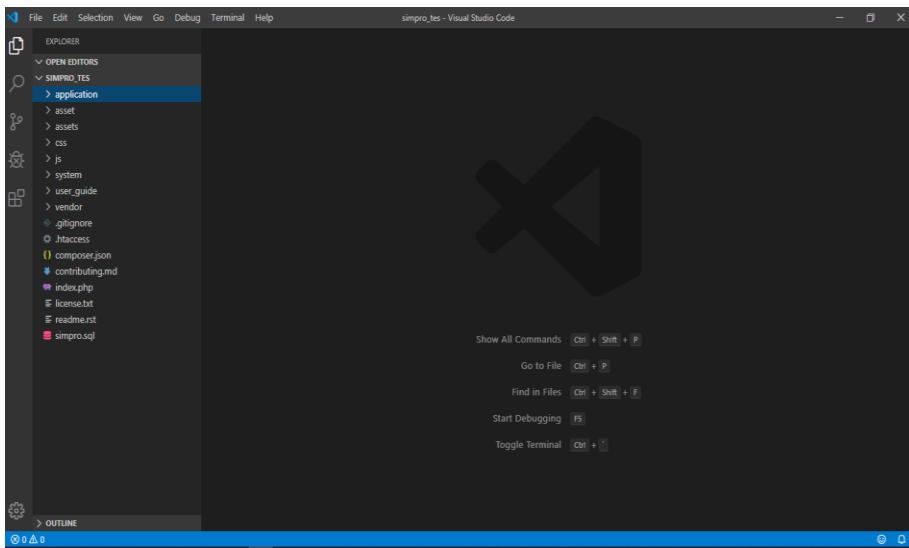
### 3.2 Visual Studio Code



*Gambar 3.3 Visual Studio Code*

Visual Studio Code (VS Code) adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi multiplatform, artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung bahasa pemrograman JavaScript, Typescript, dan Node.js, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang via marketplace Visual Studio Code (seperti C++, C#, Python, Go, Java, dst).

Banyak sekali fitur-fitur yang disediakan oleh Visual Studio Code, diantaranya Intellisense, Git Integration, Debugging, dan fitur ekstensi yang menambah kemampuan teks editor. Fitur-fitur tersebut akan terus bertambah seiring dengan bertambahnya versi Visual Studio Code. Pembaruan versi Visual Studio Code ini juga dilakukan berkala setiap bulan, dan inilah yang membedakan VS Code dengan teks editor-teks editor yang lain.



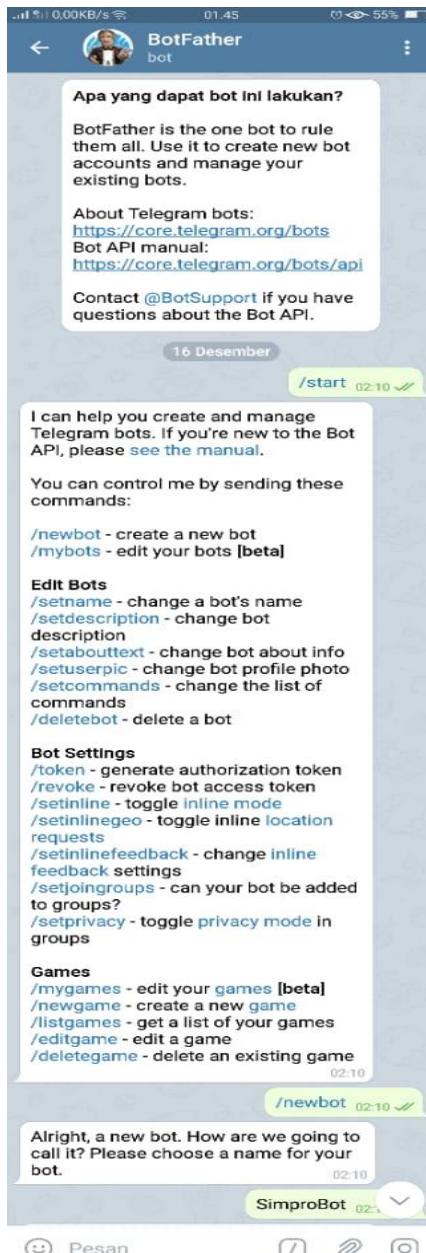
Gambar 3.4 tampilan visual studio code

Teks editor VS Code juga bersifat open source, yang mana kode sumbernya dapat kalian lihat dan kalian dapat berkontribusi untuk pengembangannya. Kode sumber dari VS Code ini pun dapat dilihat di *link Github*. Hal ini juga yang membuat VS Code menjadi favorit para pengembang aplikasi, karena para pengembang aplikasi bisa ikut serta dalam proses pengembangan VS Code ke depannya.

VS Code dapat digunakan secara langsung tanpa perlu ekstensi selama alur pemrogramannya sudah didukung langsung. Namun ada beberapa alur pemrograman yang tidak didukung langsung misalnya penggunaan snippets code dan kemampuan Intellisense yang masih kurang. Disini letak kegunaan ekstensi karena dapat menambah kemampuan dukungan alur pemrograman yang diinginkan. Ekstensi VS Code ini banyak dibuat oleh pengembang-pengembang pihak ketiga yang juga menjadi kontributor dari VS Code.

### 3.3 Bot Chat Telegram

#### 3.3.1 BotFather



Gambar 3.5 Tampilan BotFather

BotFather merupakan bot yang disediakan oleh telegram. Botfather adalah bot yang gunanya untuk menciptakan bot tetapi masih belum

bisa dijalankan, maksudnya di botfather kamu hanya bisa membuat botnya saja belum untuk membuat bot itu berjalan

Nah disinilah bot-bot keren di buat ada bot yang menggunakan bahasa pemograman ada juga yang menggunakan bot perancang seperti manybot dan juga Chatfuel

Jika di \start botfather maka akan tampil seperti dibawah ini :

I can help you create and manage Telegram bots. If you're new to the Bot API, please see the manual.

You can control me by sending these commands:

/newbot - create a new bot

/mybots - edit your bots [beta]

/mygames - edit your games [beta]

Edit Bots

/setname - change a bot's name

/setdescription - change bot description

/setabouttext - change bot about info

/setuserpic - change bot profile photo

/setcommands - change the list of commands

/deletebot - delete a bot

Bot Settings

/token - generate authorization token

/revoke - revoke bot access token

/setinline - toggle inline mode

/setinlinegeo - toggle inline location requests

/setinlinefeedback - change inline feedback settings

/setjoininggroups - can your bot be added to groups?

/setprivacy - toggle privacy mode in groups

Games

/newgame - create a new game

/listgames - get a list of your games

/editgame - edit a game

/deletegame - delete an existing game

Inilah keterangan dari perintah di atas :

/newbot - Untuk membuat bot

Ini adalah perintah untuk membuat bot. Maksimal pembuatan bot adalah 20

/mybots - Untuk edit bot [beta]

Dalam beta lebih mudah untuk kamu untuk edit bot kamu kamu tinggal menekan tombol saja untuk menjalankan semua yang ada di perintah edit bot

/mygames - Digunakan untuk edit game [beta]

Edit Bots

/setname - Untuk menganti nama

/setdescription - Untuk membuat diskripsi bot

/setabouttext - Untuk memberi info tentang bot anda

/setuserpic - Untuk mengubah Gambar bot

/setcommands - Untuk menambahkan perintah bot

/deletebot - Untuk hapus bot

Bot Settings

/token - Untuk lihat kode toke

/revoke - Untuk mengubah kode token

/setinline - untuk masuk toggle mode

/setinlinegeo - lokasi toggle inline requests

/setinlinefeedback - untuk umpan balik

/setjoininggroups - untuk membuat bot bisa join tidaknya ke dalam bot

/setprivacy - privasi ke group

Games

/newgame - Untuk membuat bot

/listgames - listgame bot  
/editgame - untuk edit game  
/deletegame - Untuk hapus game

### **3.4 Bahasa Pemrograman**

#### **3.4.1 Pengertian PHP**



*Gambar 3.6 php*

PHP Merupakan singkatan dari "PHP: Hypertext Preprocessor", yang merupakan sebuah bahasa scripting (kode untuk membangun suatu program) yang dikombinasikan pada HyperText Markup Language (HTML). Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamik dengan cepat.

#### **3.4.2 Sejarah PHP**

PHP adalah singkatan dari “PHP: Hypertext Preprocessor”, yaitu bahasa pemrograman disisi server yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan HTML. Ketika Anda mengakses sebuah URL, maka web browser akan melakukan request ke sebuah web server.

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilisan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP: Hypertext Preprocessing.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah

paradigma berorientasi objek. Server web bawaan ditambahkan pada versi 5.4 untuk mempermudah pengembang menjalankan kode PHP tanpa menginstall software server.

Versi terbaru dan stabil dari bahasa pemrograman PHP saat ini adalah versi 7.0.16 dan 7.1.2 yang resmi dirilis pada tanggal 17 Februari 2017.

### **3.4.3 Kelebihan Dari PHP**

#### **1. Memiliki Community yang besar**

Programmer Web mana yang tidak mengetahui PHP, semua web programmer paling tidak pasti pernah mencoba PHP. Banyak sekali website yang menggunakan PHP sebagai bahasa pemrograman untuk membuat aplikasi web atau website nya. Facebook, Yahoo, Wikipedia, WordPress adalah contoh website terkenal yang menggunakan PHP. Forum untuk membahas dan juga saling bertukar pikiran dalam pemrograman PHP juga telah banyak muncul di berbagai situs. Kebanyakan kuliah di bidang IT mengajarkan PHP sebagai bahasa pemrograman awal untuk mahasiswanya yang berkuliah di jurusan website development ( baca juga: Daftar Mata Kuliah Teknik Informatika)

#### **2. Mudah Dipelajari**

PHP mudah di install dan dikonfigurasi. membuatnya menjadi bahasa pemrograman tingkat entry level yang mudah dipelajari bagi seseorang yang baru mulai belajar pengembangan web. Tutorial untuk memulai belajar pemrograman PHP dapat diperoleh dengan mudah secara online, di toko buku, ataupun di lembaga bimbingan kursus pengembangan website.

### **3. Pengembangan Cepat**

Membuat Aplikasi menggunakan PHP jauh lebih cepat daripada mengembangkan aplikasi web menggunakan bahasa pemrograman lain. banyak sekali tools, boiler yang tersedia secara open source untuk bahasa pemrograman PHP. hal ini mempercepat proses dari start sampai dengan finish sebuah projek pembuatan aplikasi web.

### **4. Ringkas**

Bagi Programmer web yang pernah mencoba bahasa ASP maupun java pasti mengetahui betul satu kelebihan ini. Mulai dari proses install yang tidak perlu setting berlebihan, konfigurasi dengan database yang mudah. hingga proses pengembangan yang tidak memerlukan waktu kompilasi. membuat PHP terasa sangat ringkas dan praktis berbeda dengan bahasa pemrograman lain yang membutuhkan proses kompilasi untuk dapat melihat website yang telah diselesaikan pembuatan kodennya. Bahkan, bahasa pemrograman php dapat digunakan didalam dokumen html, hmm ringkas bukan. (baca juga: Dasar Dasar HTML)

### **5. Maintenance mudah**

Sekali web yang menggunakan PHP berjalan, programmer dapat dengan mudah melakukan update dari software PHP dengan mudah jika memang diperlukan. karena sifat PHP yang merupakan interpreter. Aplikasi web yang dibuat dengan menggunakan PHP dapat dengan mudah diupgrade versi PHP tanpa harus melakukan kompilasi ulang source code. berbeda sekali dengan bahasa pemrograman lain yang membutuhkan kompilasi ulang jika melakukan upgrade versi dari bahasa pemrograman. PHP juga dapat berjalan pada berbagai macam web server seperti apache, nginx, dan IIS. ( baca juga : Pengertian Web Server Menurut Para Ahli )

## **6. Open Source**

PHP merupakan sebuah projek Open source dengan license yang dikeluarkan oleh PHP group yaitu PHP license V3.01. Inti dari license ini adalah setiap pengguna program PHP bebas menggunakan PHP secara gratis tanpa harus memberikan royalty apapun ke PHP group namun tetap wajib mencantumkan licensi atas PHP yang dimiliki PHP Group. Dengan kata lain selama pemakai program PHP tidak mengakui produk PHP adalah buatannya maka perjual belian program yang menggunakan PHP diperbolehkan tanpa harus membayar licensi apapun.

## **7. Perkembangan Pesa**

Karena sifat PHP yang open source, banyak sekali bermunculan projek projek open source besar yang menggunakan PHP seperti Prestashop, WordPress, Drupal, dan lain lain. Hal ini menjadi keunggulan yang sangat besar bagi orang yang menguasai pemrograman PHP. Dengan sangat luasnya perkembangan PHP, maka kesempatan untuk bisnis ataupun kerja pada bidang pemrograman PHP sangatlah luas

### **3.4.4 Kekurangan Dalam PHP**

Seperti yang telah di katakan, Selain memiliki kelebihan PHP, PHP juga memiliki beberapa kekurangan. Namun kekurangannya sangat sedikit. Lalu apa saja kekurangan dari PHP, yaitu diantaranya :

1. Tidak ideal jika untuk pengembangan skala besar.
2. Tidak mempunyai sistem pemrograman berorientasi objek yang sesungguhnya.
3. Tidak dapat memisahkan antara tampilan dengan logik dengan baik (Meskipun penggunaan template bisa memperbaikinya)

4. PHP mempunyai kelemahan security tertentu yang mana jika programmer tidak jeli dalam melakukan pemrograman dan kurang memperhatikan isu dan konfigurasi PHP
5. Kode PHP bisa dibaca semua orang, dan kompilasi hanya bisa dilakukan dengan tool yang mahal dari Zend (\$2000)

### **3.4.5 Perbedaan Dari beberapa Versi PHP**

- **PHP 3**

PHP 3.0 dibuat karena melihat kelemahan dari PHP/FI yang digunakan dalam aplikasi e-commerce. Kemudian, mereka menulisnya ulang dengan masih mengacu pada PHP/FI. Setelah PHP 3.0 dikeluarkan, mereka menyarankan untuk menghentikan proyek PHP/FI karena PHP 3.0 masih lebih baik. Alasan untuk mulai mengembangkan PHP, merupakan akronim dari Hypertext Preprocessor, dan memfokuskan diri pada PHP 3.0 ialah pengembangan versi ini secara meluas dalam mendukung berbagai jenis database, protokol dan API. Salah satu kelebihan PHP 3 adalah fleksibilitas fitur, penambahan API dan protokol, kemampuan mengakses berbagai macam database, kemampuan OOP, serta kemudahan syntax bahasa. Kelebihan-kelebihan inilah yang mengundang ribuan developer utk bergabung dalam proyek PHP 3 untuk menambahkan modul-modul baru. Hal ini juga yang akhirnya merubah singkatan PHP yang pada awalnya ‘Personal Home Page’ dan kemudian secara resmi berganti menjadi ‘PHP Hypertext Processor’. Dengan dukungan yang semakin besar dari berbagai pihak yang menyumbangkan berbagai modul maka, pada tahun 1998, 10% dari seluruh webserver yang ada kala itu telah menginstalasi PHP versi 3.0.

- **PHP 4**

PHP versi 4 diluncurkan untuk menangani kelemahan PHP 3, yaitu PHP4 peningkatan performa aplikasi dan peningkatan kemampuan modular codebase secara keseluruhan. Kurangnya efisiensi waktu dan kinerja buruk diperbaiki dan ditulis ulang dari inti PHP 3. Dengan penambahan fitur baru, seperti session, output buffering dan penanganan input, menjadikan PHP 4 aman dari berbagai jenis bahasa pemrograman berbasis web. Selain itu, inti perbedaan mereka terletak pada penggunaan Engine baru yang dinamakan Zend Engine (dari kata Zeev dan Andi) adalah core dari PHP 4 yg mampu meningkatkan performa aplikasi dan secara resmi di release pada tahun 2000, hampir 2 tahun setelah pertama kali PHP 4 di kembangkan. Zend engine merupakan inti dari PHP. Sebagai bagian dari inti PHP, secara fungsional ia bertugas menangani input, menerjemahkan dan mengeksekusinya. Ia juga berperan menerjemahkan fungsi.

### **Fitur PHP 4**

1. Meningkatkan penanganan pada resource yang digunakan. Kekurangan utama dari PHP 3.0 adalah skalabilitas. Para programmer masih menganggap PHP 3.0 belum mampu digunakan pada aplikasi-aplikasi berskala besar. Hal ini membuat para developer PHP berkerja keras untuk meningkatkan kemampuan penanganan resource, sehingga PHP mampu dijalankan pada aplikasi-aplikasi berskala besar.
2. Dukungan pada pemrograman berorientasi objek. Fitur ini memainkan peranan penting bagi user untuk dapat bekerja dengan bahasa pemrograman berorientasi objek.

3. Dukungan pada penanganan session. Penanganan pada session HTTP sebenarnya juga sudah ada pada versi PHP 3.0. Namun, masih menggunakan dukungan pihak ketiga, yaitu PHPLIB. Nah, pada versi 4 ini, session sudah mampu ditangani langsung oleh PHP. Fitur ini menawarkan interaksi dari website. contoh yang paling umum digunakan adalah pada proses login dan logout.
4. Enkripsi. MCrypt sudah digabung pada distribusi asli. Fitur ini menawarkan algoritma enkripsi diantaranya Blowfish, MD5, SHA1, dan tripleDES.
5. Dukungan ISAPI. Fitur ini menghubungkan webserver IIS (dari Microsoft) dan PHP dan dianggap sebagai modul ISAPI.
6. Dukungan COM/DCOM. Fitur ini memungkinkan PHP untuk mengakses objek COM dan menambah interoperabilitas dengan aplikasi yang berjalan di Windows.
7. Dukungan terhadap Java
8. Perl Compatible Regular Expressions (PCRE) library

## • **PHP 5**

PHP 5 merupakan kelanjutan dari evolusi yang berjalan pada PHP. Walaupun pada PHP 4 sudah banyak library yang ditambahkan, PHP 5 dikembangkan dengan Zend Engine 2.0 dengan kemampuan jauh lebih powerfull dibandingkan PHP 4, terutama di sisi OOP, sehingga kemampuan PHP 5 nantinya dapat disajarkan dengan bahasa pemrograman lain yang lebih kompleks seperti Java, ASP.NET, dsb. Survey Netcraft pada Februari 2004 melaporkan bahwa 15 juta server telah di install PHP, atau sekitar 32% dari keseluruhan domain yang ada di internet.

## **Fitur PHP 5**

1. Peningkatan dari kemampuan pemrograman berorientasi objek
2. Exception handling, yang menstandarisasi logika atas pemberitahuan kesalahan pemrograma
3. Peningkatan terhadap penanganan string
4. Peningkatan dukungan terhadap XML dan Web Service, yang menggunakan libxml
5. Dukungan terhadap SQLite database server
6. PHP versi 5 dapat membuat file swf dan applet java

### **• PHP 7**

Di versi PHP 4 menggunakan mesin Zend, di versi PHP 5 menggunakan Zend II, untuk versi PHP 7 sendiri tetap menggunakan mesin Zend tapi menggunakan model mesin baru yang disebut PHPNG atau PHP Next Generation. Model terbaru ini bisa meningkatkan kinerja dua kali lipat penggunaan memori. Perusahaan pengembang membuktikannya dengan menggunakan model PHP terbaru tetapi dengan perangkat pengdukung yang lebih sedikit dan tetap melayani pengguna sebelumnya dengan total pengguna yang tidak jauh berbeda.

## **Operator Tambahan**

PHP 7 membuat kejutan dengan menambahkan operator baru di versinya, operator dengan nama **Spaceship** berkode <=>.

## **Fitur Tambahan**

Setelah PHP 7 mendukung 64-bit secara konsisten, masih banyak lagi fitur-fitur lain yang disematkan di versi PHP 7. Diantaranya dukungan untuk Anonymous class, Isset Ternary Operator, Abstract syntax tree, null coalesce operator, lots of libraries, generator return expressions, generator delegation

### **3.4.6 Tiga Model Penggunaan PHP**

Sebelum menginstall, pastikan terlebih tujuan penggunaan PHP.

Terdapat 3 model penggunaan

PHP, yaitu:

- Website dan Aplikasi Web (Server-Side Scripting)
- Command Line Scripting
- Aplikasi Desktop (GUI)

Model pertama merupakan penggunaan paling umum, dibutuhkan 3 hal: PHP, sebuah server web, dan browser web. Untuk platform Linux dan MacOS X, dapat menggunakan server web Apache, server web IIS digunakan untuk platform Windows. Browser web biasanya sudah ada dalam setiap komputer, terutama yang menggunakan Windows, yaitu Internet Explorer (IE). Selain IE, ada Mozilla FireFox, Google Chrome, Opera, dan lain-lain. Selain itu, dapat juga menyewa space web dari suatu layanan hosting di Internet. Dengan cara ini, yang perlu dilakukan hanya menulis script PHP, mengupload ke server yang disewa, dan melihat hasilnya pada browser. Dalam hal penggunaan PHP dengan cara menginstall sendiri, ada 2 pilihan metode menghubungkan PHP ke server. Pada sebagian server PHP mempunyai Direct Module Interface (SAPI).

Termasuk kedalam kelompok server ini adalah Apache, Microsoft Internet Information 6 Server, Netscape dan server iPlanet. Sebagian server yang lain memiliki dukungan untuk ISAPI, Microsfot Module Interface (contohnya OmniHTTPd). Jika PHP tidak memiliki dukungan Modul untuk server web Anda, maka gunakan PHP sebagai processor CGI atau FastCGI. Artinya, server disetup untuk menggunakan CGI executable dari PHP untuk memproses semua file PHP yang diminta pada server. Jika PHP akan digunakan sebagai Command Line Scripting, misalnya untuk menulis script yang membangkitkan gambar secara otomatis secara offline, atau memproses file teks yang dikendalikan oleh beberapa argumen yang dikirimkan kepada script, maka dibutuhkan Command Line Executable. Dalam kasus ini, tidak diperlukan server dan browser. PHP juga dapat digunakan untuk menulis aplikasi dekstop GUI. Untuk itu diperlukan ekstensi PHP-GTK. Cara ini memiliki pendekatan yang berbeda dari membuat halaman-halaman web, seperti tidak ada output HTML yang dibuat, tetapi mengelola windows dan objek yang berada didalamnya. PHP-GTK tidak disertakan dalam distribusi resmi PHP.

### **3.4.7 Sintaks PHP**

Skrip PHP dieksekusi di server dan hasil eksekusi yang berupa kode HTML dikirim ke computer klien.

#### **1. Simbol Pembuka dan Penutup Skrip PHP**

Skrip PHP selalu diawali dengan tanda ‘<?php’ dan ditutup dengan ‘?>’. Skrip PHP dapat diletakkan dimana saja dalam suatu dokumen HTML. Beberapa server yang sudah diatur konfigurasi directive ‘shorthand-support’, dapat mengawali skrip dengan tanda

‘<?’ dan diakhiri dengan ‘?>’. Tetapi demi kompatibilitas maksimum, disarankan menggunakan bentuk standar ‘<?php’

## 2. Perluasan File PHP

File PHP harus disimpan dengan perluasan ‘.php’. File PHP biasanya berisi tag-tag HTML dan beberapa kode skrip PHP. Contoh:

```
1 <html>
2 <body>
3 <?php
4 echo "Hello World";
5 ?>
6 </body>
7 </html>
```

Gambar 3.7 perluasan file php

## 3. Simbol Pemisah Setiap Perintah PHP

Setiap perintah PHP harus diakhiri dengan simbol semikolon ‘;’. Tugas simbol ini untuk membedakan atau memisahkan satu perintah PHP dengan perintah PHP lainnya.

## 4. Perintah Mencetak Hasil

Ada dua jenis perintah PHP untuk mencetak keluaran atau hasil yaitu ‘echo’ dan ‘print’. Contoh:

```
9  <html>
10 <body>
11 12
12 <?php
13 echo "Hello World";
14 print "I'm learning PHP Script Language now";
15 ?>
16 </body></html>
```

Gambar 3.8 perintah mencetak hasil

#### 3.4.7.1 Variabel PHP

Sintaks Perintah Membuat Variabel Suatu variabel digunakan untuk menyimpan suatu nilai, dapat berupa teks, angka, atau array. Variabel dalam PHP menggunakan simbol ‘\$’ di awal namanya. Sintaks perintah membuat variabel:

```
14 $nama_var = nilai;
```

Gambar 3.9 Variabel Nilai

### 5. Tipe Data dari Variabel

Tipe data variabel tidak perlu dideklarasikan, PHP akan otomatis mengkonversi atau menentukan tipe data variabel berdasarkan nilai yang disimpannya. Contoh:

```
21
22 <?php
23 $nama='Alvina Khansa';
24 $nilai=90;
25 ?>
26
27
```

Gambar 3.10 Tipe data dari variabel

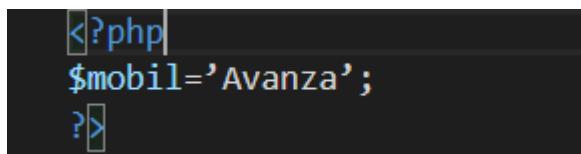
Catatan (Variabel nama diatas otomatis akan bertipe string, variabel nilai akan bertipe integer).

## 6. Aturan Pemberian Nama Variabel

- Harus dimulai dengan huruf atau garis bawah (underscore) ‘\_’.
- Hanya dapat menggunakan karakter alphanumeric dan underscore
- (A-Z / a-z / 0 - 9, dan \_)
- Sebaiknya tidak menggunakan spasi, jika nama variabel terdiri lebih dari satu kata, pisahkan dengan underscore (\$nama\_depan, \$nilai\_tugas) atau kapitalisasi (\$namaDepan, \$nilaiTugas)

## 7. Mendeklarasikan/Membuat Variabel

Tidak ada perintah khusus dalam PHP untuk membuat variabel. Sebuah variabel akan terbentuk pada saat sebuah nilai ditentukan kepadanya. Contoh perintah:



```
<?php  
$mobil='Avanza';  
?>
```

Gambar 3.11 deklarasi variabel

Setelah perintah tersebut dieksekusi maka variabel \$mobil akan berisi Avanza.

Catatan: untuk memberikan suatu nilai berupa teks/string kepada sebuah variabel, nilai tersebut harus diapit dengan tanda petik tunggal atau ganda.

## 8. Scope Variabel

Scope atau ruang lingkup variabel adalah bagian dari skrip yang dapat mereferensikan variabel tersebut. Ada 4 scope variabel dalam PHP:

## 9. Scope Local

Suatu variabel yang dibuat pada suatu fungsi akan menjadi variabel lokal (memiliki scope local) dan hanya bisa diakses di dalam fungsi. Nama variabel yang sama dapat dibuat dalam fungsi yang berbeda, sebab variabel lokal hanya dikenali oleh fungsi yang membentuk variabel tersebut. Variabel lokal akan dihapus setelah fungsi usai dieksekusi. Contoh:

```
26 <?php
27 function dicoba(){
28 $lokal='Saya hanya bisa diakses dari fungsi dimana saya berada';
29 echo $lokal; //mencetak var lokal
30 }
31 echo $lokal; //akan terjadi error
32 ?>
```

Gambar 3.12 Scope Lokal

## 10. Scope Global

Scope global dimiliki oleh variabel yang dibuat diluar fungsi. Variabel dengan scope global dapat diakses dari bagian manapun dari program selama perintah tersebut ditulis diluar suatu fungsi. Variabel global dapat diakses dari dalam suatu fungsi dengan menggunakan kata kunci ‘**global**’. Contoh:

```
35 <?php
36 $a = 10; // scope global
37 function myTest()
38 {
39 // mengacu ke variabel scope global
40 echo global $a;
41 }
42 myTest();
43 ?>
```

Gambar 3.13 Scope Global

PHP menyimpan semua variabel global dalam array bernama **\$GLOBAL** []. Indeks dari array adalah nama dari variabel-variabel tersebut. Array ini dapat diakses dari dalam fungsi, tetapi juga dapat digunakan untuk mengupdate variabel global secara langsung. Contoh:

```
46 <?php
47 global $a;
48 $a="abc";
49 // adalah sama dengan menggunakan perintah:
50 $GLOBALS["a"]="abc";
51 ?>
```

Gambar 3.14 Scope Global 2

## 7. Scope Statik

Ketika suatu fungsi selesai digunakan, secara normal semua variabelnya akan dihapus. Jika diinginkan variabel-variabel tersebut tidak dihapus ketika fungsi selesai dipakai, gunakan kata kunci ‘static’ saat membuat variabel. Contoh pembuatan variabel static

```
53 <?php
54 static $varstatik
55 ?>
```

Gambar 3.15 Scope statik

variabel \$varStatik sekarang menjadi variabel statik.

### 3.4.7.2 Operator

Operator Hitung

Operator	Penjelasan	Contoh	Hasil
+	Pertambahan	<b>X=2</b> <b>Y=x+2</b>	<b>Y=4</b>
-	Pengurangan	<b>x=2</b> <b>y=5-x</b>	<b>Y=3</b>
*	Perkalian	<b>x=4</b> <b>y=x*5</b>	<b>y=20</b>
/	Pembagian	<b>y=15/5</b>	<b>y=3</b>
%	Sisa hasil bagi	<b>x=10%5</b> <b>y=10%8</b> <b>z=5%2</b>	<b>x=0</b> <b>y=2</b> <b>z=1</b>
++	Inkremen	<b>x=5</b> <b>x++</b>	<b>x=6</b>
--	dekremen	<b>x=5</b> <b>x--</b>	<b>x=4</b>

### Contoh 1

```
59 <?php  
60 //menghitung penjualan bersih  
61 $jual=100000;  
62 17  
63 $potongan=0.1;  
64 $net=$jual-($jual*$potongan);  
65 echo "Penjualan bersih = Rp $net,00";  
66 ?>
```

### Contoh 2

```
68 <?php  
69 //penggunaan operator modulus/sisa hasil bagi  
70 $hal=10;  
71 if ($hal % 2 == 0)  
72 echo 'Halaman genap';  
73 else  
74 echo 'Halaman ganjil';  
75 ?>
```

### Contoh 3

```
77 <?php  
78 //penggunaan inkrement  
79 for($i=1;$i<=100;$i++){  
80 echo "$i ";  
81 }  
82 ?>
```

## Operator Penugasan

<b>Operator</b>	<b>Contoh</b>	<b>Sama Dengan</b>
=	x=y	x=y
+=	x+=y	x=x+y
-=	x-=y	x=x-y
*=	x*=y	x=x*y
/=	x/=y	x=x/y
.=	x.=y	x=x.y
%=	x%-=y	x=x%y

## Operator Perbandingan

<b>Operator</b>	<b>Penjelasan</b>	<b>Contoh</b>
==	sama dengan	5==8 mengembalikan nilai false
!=	tidak sama dengan	5!=8 mengembalikan nilai true
<>	tidak sama dengan	5<>8 mengembalikan nilai true
>	lebih besar dari	5>8 mengembalikan nilai false
<	lebih kecil dari	5<8 mengembalikan nilai true
>=	lebih besar dari atau sama dengan	5>=8 mengembalikan nilai false

<code>&lt;=</code>	lebih kecil dari atau sama dengan	<code>5&lt;=8</code> mengembalikan nilai true
--------------------	-----------------------------------	--

### Operator Logika

Operator	Penjelasan	Contoh
<code>&amp;&amp;</code>	and	<code>x=6; y=3;</code> <code>(x &lt; 10 &amp;&amp; y &gt; 1)</code> mengembalikan true
<code>  </code>	or	<code>x=6; y=3; 19</code> <code>(x==5    y==5)</code> mengembalikan false
<code>!</code>	not	<code>x=6; y=3;</code> <code>!(x==y)</code> mengembalikan true

#### 3.4.7.3 Struktur Kendali

Seringkali alur eksekusi suatu program tidak berjalan lurus dari baris kode pertama sampai baris terakhir. Kadang-kadang pada suatu baris tertentu alur program bercabang (struktur keputusan). Di lain baris alur program akan kembali ke baris kode sebelumnya untuk mengulangi sekelompok perintah (struktur perulangan). Struktur keputusan dalam PHP mengenal struktur if dan switch. Untuk menerapkan suatu perulangan baris-baris kode program menggunakan struktur **for**, **while**, **do...while**, **dan foreach**. Berikut ini penjelasan masing-masing struktur tersebut.

### 3.4.7.4 Struktur Keputusan

Ketika Anda menulis kode program, suatu saat Anda perlu menggunakan mekanisme percabangan sehingga berdasarkan kondisi program Anda akan melakukan aksi yang berbeda. Misalnya saja Anda ingin membuat program yang dapat menampilkan salam yang berbeda tergantung jenis hari pada saat itu, maka kode programnya akan seperti ini:

```
84 <?php  
85 $hari=date("D");  
86 if ($hari=="Mon")  
87 echo "Selamat berlibur";  
88 else  
89 echo "Selamat bekerja dan berkarya";  
90 ?> |
```

Gambar 3.16 Struktur keputusan

Jika program diatas dijalankan pada hari Minggu, maka Anda akan memperoleh ucapan ‘Selamat berlibur’, tapi jika dijalankan selain hari Minggu akan memperoleh ucapan ‘Selamat bekerja dan berkarya’. Ya, contoh diatas adalah salah satu penggunaan struktur kondisi (**if**) yang dimiliki PHP. Struktur **if** mempunyai beberapa format, berikut penjelasannya.

#### IF

Struktur ini memiliki 3 jenis format pemakaian, yaitu:

- Pertama, bila struktur ini hanya akan mengeksekusi beberapa kode program hanya jika nilai kondisi true, maka format yang digunakan:  
*if(kondisi) kode\_yang\_akan\_dieksekusi\_bila\_nilai\_kondisi\_true*

Contoh:

```
<?php  
$hari=date("D");  
if ($hari=="Mon") echo "Selamat berlibur";  
?>
```

- Kedua, baik nilai kondisi if true atau false akan sama-sama mengeksekusi suatu kode program, maka format yang digunakan:

*if(kondisi)*

*kode\_yang\_akan\_dieksekusi\_bila\_nilai\_kondisi\_true;*

*else*

*kode\_yang\_akan\_dieksekusi\_bila\_nilai\_kondisi\_false;*

Contoh:

```
<?php  
$hari=date("D");  
if ($hari=="Mon")  
echo "Selamat berlibur";  
else  
echo "Selamat bekerja dan berkarya";  
?> |
```

- Ketiga, bila ada beberapa kondisi yang perlu dievaluasi, maka format yang digunakan:

*if(kondisi\_1)*

*kode\_yang\_akan\_dieksekusi\_bila\_nilai\_kondisi\_1\_true;*

*elseif(kondisi\_2)*

*kode\_yang\_akan\_dieksekusi\_bila\_nilai\_kondisi\_2\_true;*

*else*

*kode\_yang\_akan*

*dieksekusi\_bila\_nilai\_kondisi\_1\_dan\_2\_false;*

Contoh:

```
<?php  
$hari=date("D");  
if ($hari=="Mon")  
echo "Selamat berlibur";  
elseif ($hari=="Sat")  
echo "Selamat libur panjang";  
else  
echo "Selamat bekerja dan berkarya";  
?> |
```

Jika kode program yang akan dieksekusi dalam if lebih dari satu, maka kode-kode program tersebut harus dikurung dengan tanda kurung kurawal buka dan tutup seperti ini:

```
<?php  
$hari=date("D");  
if ($hari=="Mon"){  
echo "Hai coy,<br />";  
echo "Selamat berlibur<br />";  
echo "Jangan lupa pesenanku ya";  
}  
?>
```

## Switch

Jika dalam program ada banyak pilihan, maka lebih baik menggunakan switch. Sintaksnya:

```
switch(n) {  
case label_1 : pernyataan_yg_dieksekusi_jika_n=label_1;  
break;  
case label_2 : pernyataan_yg_dieksekusi_jika_n=label_2;  
break;
```

```
default :
dieksekusi_bila_nilai_n_bukan_label2_atau_label_1;
23
}
```

Contoh:

```
<html>
<body>
<?php
$x=1;
switch ($x)
{
case 1:
echo "Number 1";
break;
case 2:
echo "Number 2";
break;
case 3:
echo "Number 3";
break;
default:
echo "No number between 1 and 3";
}
?>
</body>
</html>
```

## Struktur Perulangan (Loop)

PHP mempunyai beberapa struktur perulangan:4

### for

Struktur perulangan for sering digunakan dalam pemrograman.

Sintaksnya:

```
for (inisialisasi; kondisi; inkrement)
{
kode yang akan dieksekusi;
}
```

Tiga parameteranya:

- **inisialisasi** : nilai awal konter
- **kondisi** : parameter yang akan dievaluasi pada setiap iterasi perulangan. Jika hasil evaluasi true, perulangan akan dilanjutkan, bila false perulangan akan dihentikan
- **inkremen** : nilai pertambahan konter setiap satu iterasi perulangan diselesaikan

Setiap parameter di atas bersifat opsional.

Contoh:

```
<html>
<body>
<?php
for ($i=1; $i<=5; $i++)
{
echo "Bilangan ke-$i: " . $i . "<br />";
}
?>
</body>
</html>
```

Contoh for tanpa parameter kondisi:

```
<html>
<body>
<?php
for ($i=1;; $i++)
25
{
echo "The number is " . $i . "<br />";
if ($i>=5) exit;
}
?>
</body>
</html>
```

## Foreach

Perulangan ini disediakan PHP untuk memudahkan kita mengakses elemen-elemen suatu array. Sintaks:

```
foreach ($array as $value)
{
    kode yang akan dieksekusi;
}
```

Pada setiap iterasi perulangan, nilai dari elemen array yang sedang ditunjuk oleh pointer akan diletakkan ke variabel \$value kemudian pointer akan bergerak menunjuk elemen array berikutnya sehingga pada iterasi selanjutnya, elemen array berikutnya yang akan diolah.

Contoh berikut akan mencetak nilai “satu”, “dua”, dan “tiga” secara berturut-turut:

```
<html>
<body>
<?php
$x=array("satu","dua","tiga");
foreach ($x as $value)
{
    echo $value . "<br />";
}
?>
</body>
26
</html>
```

## While

Perulangan while akan menjalankan suatu blok kode (sekelompok kode) selama kondisinya bernilai true. Sintaks perulangan ini:

```
while (condition)
{
```

*blok kode yang akan dieksekusi;*  
}

Contoh:

```
<html>
<body>
<?php
$i=1;
while($i<=5)
{
echo "Bilangan ke-$i: " . $i . "<br />";
$i++;
}
?>
</body>
</html>
```

Contoh while di atas akan terus menjalankan blok kodennya selama nilai dari variabel \$i lebih kecil atau sama dengan 5. Dan pada setiap iterasinya, nilai variabel \$i akan bertambah 1 (\$i++).

### **Do...while**

Satu yang unik dari perulangan ini adalah, dia pasti akan melakukan iterasi, minimal satu kali, meskipun nilai kondisinya tidak pernah true. Hal ini disebabkan struktur perulangan yang menyebabkan blok kode akan dijalankan lebih dulu, lalu kondisinya dievaluasi belakangan. Sintaks perulangan ini:

```
Do
{
    kode yang akan dieksekusi;
}
while (kondisi);
```

Contoh:

```
<html>
<body>
<?php
$i=1;
do
{
    $i++;
    echo "Bilangan ke-$i: " . $i . "<br />";
}
while ($i<=5);
?>
</body>
</html>
```

Contoh di atas akan menjalankan kode `$i++` dan `echo "Bilangan ke-$i: ". $i. "<br />"`; terlebih dahulu, setelah itu baru kondisinya dievaluasi.

### 3.4.7.5 Fungsi

Fungsi merupakan salah satu teknik pemrograman modular. Sebuah aplikasi besar disusun dari modul-modul yang berupa sebuah fungsi atau prosedur. Fungsi berisi sekelompok kode dengan tugas dan tujuan spesifik. Fungsi tidak akan dieksekusi ketika program dijalankan. Fungsi hanya akan dieksekusi jika dilakukan pemanggilan terhadapnya. Pemanggilan dapat dilakukan dari mana saja dalam program. Keuntungan teknik ini, modul-modul yang dibuat dapat digunakan berkali-kali (reuse).

#### 1. Membuat Fungs

Sebuah fungsi dibuat dengan aturan sintaks:

```
function namaFungsi()
{
    kode-kode yang akan dieksekusi;
```

}

Beberapa petunjuk dalam membuat sebuah fungsi:

- Namai fungsi yang menggambarkan fungsinya
- Nama fungsi dimulai dengan huruf atau garis bawah (underscore), tidak boleh angka.

## 2. Pemanggilan Fungsi

Ketika fungsi sudah dibuat, dia dapat dijalankan dengan cara dipanggil. Pemanggilan suatu fungsi mengikuti pola: nama fungsi lalu diikuti tanda kurung dan nilai parameter jika ada. Contoh:

*tambah(10,20);*

Memanggil sebuah fungsi bernama tambah dengan nilai parameter 10 dan 20. Jika tidak ada nilai parameter, maka pemanggilan fungsi:

*cetak();*

## 3. Parameter Fungs

Untuk menambah daya guna fungsi dapat ditambahkan parameter fungsi yang tidak lain adalah serupa variabel. Parameter ini dituliskan sesudah nama fungsi didalam tanda kurung. Dengan parameter ini, hasil dari fungsi dapat diatur sesuai dengan keinginan.

Contoh:

```
<html>
29
<body>
<?php
function namaProg ($fprogram) {
echo $fname . "<br />"}
echo "Bahasa membuat struktur dan konten adalah ".namaProg("HTML");
echo "Unsur interaktif diberikan oleh ".namaProg("Javascript");
echo "Memperindah tampilan fungsi dari ".namaProg("CSS");
echo "Aplikasi web pengolahan data menggunakan ".namaProg("PHP");
?>
</body>
</html>
```

Hasil dari fungsi diatas (nama program) dapat diubah-ubah dengan memberi nilai berbeda pada parameter \$fprogram saat pemanggilan fungsi.

#### 4. Nilai Balik Fungsi

Fungsi dapat diatur agar mengembalikan hasil berupa nilai dengan cara menggunakan kata kunci **return**.

Contoh:

```
<html>
<body>
<?php
function tambah(){
$total = $x + $y;
return $total;
}
echo "$x + $y = " . tambah(5,20);
?>
</body>
</html>
```

Nilai yang dikembalikan pada fungsi diatas adalah jumlah dari nilai variabel \$x dan \$y yang ada didalam variabel \$total. Hasil di layar browser adalah tampilan  $5 + 20 = 25$ .

##### 3.4.7.6 Array

Array adalah jenis tipe data khusus yang menyimpan sejumlah nilai data dalam sebuah variabel. Nilai-nilai data tersebut, yang disebut elemen array, memiliki indeks yang menunjukkan urutannya dalam array. Elemen array pertama akan memiliki indeks 0, elemen kedua berindeks 1, dan seterusnya. Dengan array, proses pencarian data tertentu akan mudah dilakukan. Misalkan Anda memiliki data

nama barang elektronik yang disimpan dalam variabel tunggal seperti ini:

```
$brg1="DVD";  
$brg2="Televisi";  
$brg3="Lemari es"
```

Bagaimana jika kita ingin mencari barang elektronik tertentu dengan menggunakan perulangan terhadap kelompok variabel tersebut? Tentu akan repot dengan berbagai kode yang harus kita buat untuk menentukan nomor urut variabel tersebut terlebih dahulu dan sebagainya. Lain halnya jika menggunakan array, kita hanya perlu menggunakan indeks array untuk menemukan barang elektronik yang kita cari.

## 5. Membuat Array

Membuat array dalam PHP menggunakan fungsi array().

Contoh:

```
$brg = array ("DVD", "Televisi", "Lemari es");
```

Untuk mengakses elemen array tersebut dengan menggunakan indeksnya. Nilai data “Televisi” dapat diakses dengan kode \$brg[1]. Misalnya dicetak, maka perintahnya adalah:

```
echo "Elemen array kedua adalah : " . $brg[1];
```

Array juga dapat dibuat dengan cara menentukan indeksnya langsung seperti:

```
$brg[0] = "DVD";  
$brg[1] = "Televisi";  
$brg[2] = "Lemari es";
```

## 6. Jenis Array

Dalam PHP terdapat 3 jenis array, yaitu array numerik (indexed arrays), array assosiatif (associative array), dan array multidimensi (multidimensional array). Array numerik Jenis array yang berindeks numeris seperti contoh sebelumnya. Array assosiatif Jenis array yang memiliki indeks berupa string. Array multidimensi Jenis array yang indeksnya juga array. Artinya, elemen dari array jenis ini berupa suatu array juga.

Contoh:

```
$mhs = array (
    array ('A12.2010.04567', 'Anita Larasati', 3.5);
    array ('A12.2010.05678', 'Dude Harmono', 3);
    array ('A12.2010.06789', 'Ernawati Listyani', 2.75);
```

Untuk mengakses elemen array multidimensi, misalkan akan dicetak, maka kodenya:

```
<?php
echo "NIM : " . $mhs[0][0]. "Nama : " . $mhs[0][1] . "IPK : " . $mhs[0][2] . "<br />";
echo "NIM : " . $mhs[1][0]. "Nama : " . $mhs[1][1] . "IPK : " . $mhs[1][2]; . "<br />";
echo "NIM : " . $mhs[2][0]. "Nama : " . $mhs[2][1] . "IPK : " . $mhs[2][2];
?>
```

## 7. Mencari Panjang Suatu Array

Seperti diketahui array terdiri dari sejumlah elemen array. Untuk mengetahui jumlah elemen dalam suatu array atau panjang suatu array dapat menggunakan fungsi count () .

Contoh:

```
$mobil = array ("Volvo","Jaguar", "Mercedez");
echo "Panjang array adalah : " . count($mobil);|
```

## 8. Mencetak Seluruh Elemen Array

Untuk mencetak seluruh elemen array dapat digunakan suatu perulangan for sebagai berikut:

```
$mobil = array ("Volvo","Jaguar", "Mercedez");
$jumlah = count($mobil);
for ($i=0; $i<$jumlah; $i++) {
    echo $mobil[$i] . '<br />';
}|
```

## Array Assosiatif

Penulisan indeks untuk jenis array ini bagi Anda yang tidak terbiasa akan terbilang ganjil. Tetapi jika sudah terbiasa tidak akan menjadi masalah. Kode pembuatan array jenis ini jika menggunakan fungsi array () tampak seperti berikut ini:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
echo "Joni berusia " . $umur("Joni") . " tahun";|
```

Tampak dalam menentukan elemen-elemen array diatas menggunakan simbol “=>” yang dibuat dengan simbol sama dengan (“=” ) dikombinasikan dengan simbol lebih besar (“>”). Untuk mencetak seluruh nilai elemen array jenis ini dapat menggunakan perulangan foreach, seperti ini:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
foreach ($umur as $x=>$nilaiX) {
    echo "Indeks " . $x . " bernilai " . $nilaiX . "<br />";
}
```

## 5. Menyortir Array

Elemen-elemen array dapat diurutkan baik secara alfabet maupun numeris, menaik (ascending) atau menurun (descending). Berikutnya penjelasan tentang jenis-jenis fungsi pengurutan array dalam PHP.

### Fungsi sort ()

Fungsi ini akan mengurutkan elemen secara menaik atau ascending. Jika elemen-elemen array berupa string, maka akan diurutkan menurut urutan alfabet (a, b, c, dan seterusnya). Dan jika berupa nilai numerik, maka elemen array akan diurutkan secara numeris (1, 2, 3, dan seterusnya). Berikut ini contoh keduanya :

*Pengurutan terhadap data string:*

```
$mobil = array ("Volvo", "Jaguar", "Mercedez");
sort($mobil);
$pjg = count($mobil);
for ($i=0; $i<$pjg; $i++) {
    echo $mobil[$i] . "<br />";
}
```

*Pengurutan terhadap data numerik:*

```
$bilangan = array (5, 7, 3, 4, 2, 1, 6);
sort($bilangan);
$pjg = count($bilangan);
for ($i=0; $i<$pjg; $i++) {
    echo $bilangan[$i] . "<br />";
}
```

## Fungsi rsort()

Fungsi ini akan mengurutkan elemen array secara menurun atau descending. Contoh untuk data string dan numerik:

*Pengurutan terhadap data string:*

```
$mobil = array ("Volvo", "Jaguar", "Mercedez");
rsort($mobil);
$pjg = count($mobil);
for ($i=0; $i<$pjg; $i++) {
    echo $mobil[$i] . "<br />";
}
```

*Pengurutan terhadap data numeris:*

```
$bilangan = array (5, 7, 3, 4, 2, 1, 6);
rsort($bilangan);
$pjg = count($bilangan);
for ($i=0; $i<$pjg; $i++) {
    echo $bilangan[$i] . "<br />";
}
```

## Fungsi assort ()

Fungsi ini digunakan untuk jenis array assosiatif yang akan mengurutkan array secara menaik berdasarkan nilai data elemen array.

Contoh:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
asort($umur);
foreach ($umur as $x=>$nilaiX) [
    echo "Indeks " . $x . " bernilai " . $nilaiX . "<br />";
]
```

Karena elemen ketiga memiliki nilai data terkecil (16) maka akan dicetak sebagai yang pertama.

## Fungsi ksort ()

Fungsi akan mengurutkan secara menaik array assosiatif berdasarkan nilai indeks elemen array.

Contoh:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
ksort($umur);
foreach ($umur as $x=>$nilaiX) {
    echo "Indeks " . $x . " bernilai " . $nilaiX . "<br />";
}
```

Karena elemen kedua memiliki indeks berupa “Indra” yang memiliki urutan alfabetis lebih dahulu (“I” lebih dahulu daripada “J” atau “S”) maka akan dicetak sebagai yang pertama.

### Fungsi arsort ()

Fungsi ini merupakan gabungan sifat dari fungsi asort() dan rsort() yang akan menghasilkan pengurutan array assosiatif secara menurun (descending).

Contoh:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
arsort($umur);
foreach ($umur as $x=>$nilaiX) [
    echo "Indeks " . $x . " bernilai " . $nilaiX . "<br />";
]
```

Akan menghasilkan elemen kedua (“18”) dicetak sebagai yang pertama.

### Fungsi krsort ()

Fungsi ini gabungan sifat dari fungsi ksort() dan rsort() yang akan menghasilkan pengurutan array assosiatif secara menurun (descending) berdasarkan nilai indeks elemen array.

Contoh:

```
$umur = array ("Joni"=>"17", "Indra"=>"18", "Susi"=>"16");
arsort($umur);
foreach ($umur as $x=>$nilaiX) {
    echo "Indeks " . $x . " bernilai " . $nilaiX . "<br />";
}
```

Akan menghasilkan elemen ketiga (“Susi”) dicetak sebagai yang pertama.

### **3.4.7.7 Mengelola Data Waktu dan Tanggal**

#### **a. Nilai Timestamp**

Sebelum mempelajari fungsi-fungsi yang dapat digunakan, pertama kali yang harus kita ketahui adalah informasi waktu dasar yang digunakan dalam memanipulasi data waktu dan tanggal. Informasi ini disebut epoch timestamp atau UNIX timestamp atau timestamp saja. Istilah ini merujuk pada jumlah detik yang dihitung mulai pukul 0:0:0 tanggal 1 Januari 1970 GMT sampai dengan waktu yang kita tentukan. Nilai inilah yang menjadi dasar perhitungan informasi waktu dan tanggal dalam fungsi-fungsi terkait di PHP. Misalnya ingin mencari nilai timestamp pada tanggal 2 Januari 1970 maka nilai timestamp yang kita peroleh adalah 86400 yang berasal dari perhitungan  $60 * 60 * 24$ . Bagaimana dengan tanggal saat ini, berapa nilai timestampnya?

#### **b. Fungsi-fungsi Waktu dan Tanggal**

Nilai timestamp tersebut dapat dimanipulasi untuk menghasilkan informasi tertentu dari waktu dan tanggal. PHP menyediakan banyak fungsi untuk hal ini. Salah satu fungsi dasar yang perlu diketahui adalah date () .

#### **c. Fungsi date ()**

Fungsi date () ini akan mengembalikan nilai waktu dan tanggal saat ini dari komputer yang digunakan. Sintaks fungsi: date (format, timestamp) Parameter format wajib digunakan yang akan menentukan format waktu yang ingin dihasilkan. Parameter timestamp bersifat opsional yang menentukan waktu timestamp yang akan dimanipulasi.

Nilai default parameter ini adalah waktu timestamp saat ini. Karakter yang dapat digunakan dalam parameter format untuk menghasilkan informasi waktu dan tanggal yang kita inginkan antara lain:

- **d** : menghasilkan nilai hari dalam sebulan (01 s/d 31)
- **m** : menghasilkan nilai bulan dalam setahun (01 s/d 12)
- **Y** : menghasilkan nilai tahun dengan panjang 4 digit.

Karakter lain yang dapat digunakan dalam parameter format antara lain “/”, “.”, atau “-” .

Contoh:

```
echo date("Y/m/d");
echo date("d-m-Y");
✓ echo date("m.d.Y");
```

#### d. Fungsi mktime ()

Nilai timestamp dapat diperoleh dengan fungsi mktime () .

Sintaks fungsi:

*Mktime (jam, menit, detik, bulan, tanggal, tahun);*

Contoh penggunaan dari fungsi mktime() ini misalkan kita ingin menampilkan informasi tanggal untuk hari besok. Maka kodennya akan seperti berikut ini:

```
$tomorrow = mktime(0, 0, 0, date("m"), date("d")+1, date("Y"));
echo "Besok adalah tanggal " . date("Y/m/d", $tomorrow);
```

Dalam kode diatas pertama ditentukan terlebih dahulu nilai timestamp untuk hari besok. Pada baris berikutnya, nilai tersebut digunakan sebagai parameter kedua dari fungsi date (). Sebagai catatan, untuk menggunakan fungsi ini Anda perlu melihat versi PHP

yang Anda gunakan sebab fungsi ini telah dihapus pada PHP versi 5.1.0 keatas. Sebagai gantinya, gunakan fungsi time () berikut ini.

#### e. Fungsi time ()

Fungsi ini menggantikan fungsi mktime() untuk PHP versi 5.1.0 keatas. Seperti fungsi mktime (), fungsi ini juga akan mengembalikan nilai timestamp. Fungsi time tidak memiliki parameter. Sebagai contoh, dimisalkan akan ditampilkan informasi tanggal seminggu dari saat ini, maka kode programnya:

```
$mgdpn= time() + (7 * 24 * 60 * 60);  
echo "Hari ini tanggal : " . date('d-m-Y') . "<br />";  
echo "Seminggu lagi tanggal : " . date('d-m-Y', $mgdpn);  
|
```

#### f. Fungsi getdate ()

Fungsi ini akan mengembalikan nilai dalam bentuk array assosiatif berisi informasi tanggal dari nilai timestamp yang diberikan. Jika tidak menyertakan nilai timestamp, maka waktu lokal (waktu komputer atau server) yang akan digunakan. Array assosiatif yang dihasilkan oleh fungsi berisi sekumpulan informasi tentang waktu yang tersimpan didalam masing-masing elemen array:

"seconds"	berisi nilai numeris menunjukkan detik	0 s/d 59
"minutes"	berisi nilai numeris menunjukkan menit	0 s/d 59
"mday"	berisi nilai numeris menunjukkan hari dalam sebulan	1 s/d 31

<b>"wday"</b>	berisi nilai numeris menunjukkan hari dalam seminggu	0 (sunday) s/d 6 (saturday)
<b>"mon"</b>	berisi nilai numeris menunjukkan bulan	1 through 12
<b>"year"</b>	berisi nilai numeris menunjukkan tahun dengan panjang 4 digit	contoh: 1999 atau 2014
<b>"yday"</b>	berisi nilai numeris menunjukkan hari dalam setahun	0 s/d 365
<b>"weekday"</b>	berisi teks menunjukkan nama hari	Sunday s/d Saturday
<b>"month"</b>	berisi teks menunjukkan nama bulan	January s/d December
<b>0</b>	Jumlah detik sejak dari UNIX epoch, sama seperti nilai yang dikembalikan oleh fungsi time () dan yang digunakan dalam fungsi date ()	Tergantung sistem yang digunakan, biasanya 2147483648 s/d 2147483647.

Elemen array pertama berisi informasi tentang detik dari waktu saat ini yang indeksnya bernama “seconds”, lalu informasi menit pada

eleman array kedua dengan indeks “minutes”, demikian dan seterusnya. Jadi, untuk menampilkan informasi waktu dan tanggal dapat menggunakan kode seperti ini:

```
$tgl = getdate();
echo "Waktu saat ini menunjukkan pukul " . $tgl['hours'] . ":" . $tgl['minutes'] . ":" .
| | | | $tgl['seconds'];
```

Hasilnya jika ditampilkan pada layar browser akan menampilkan teks ‘Waktu saat ini menunjukkan pukul 14:7:14’.

#### g. Fungsi checkdate()

Fungsi ini dapat digunakan untuk memeriksa kebenaran dari data waktu yang diberikan. Sintaks fungsi:

*checkdate(bulan, tanggal, tahun)*

Parameter bulan dapat berisi nilai 1-12 yang menunjukkan bulan, parameter tanggal dapat berisi 1-31 tergantung bulannya, sedangkan parameter tahun dapat berisi nilai 1-32767. Fungsi akan mengembalikan nilai boolean true (angka 1) jika tanggal valid dan false jika ternyata tanggal tidak valid (misalnya tanggal 30 Pebruari jelas invalid).

Contoh:

```
if (checkdate(2,29,2014)){
    echo "Tanggal valid";
} else {
    echo "Tanggal invalid";
}
```

Kode diatas akan menghasilkan teks ‘Tanggal invalid’ karena tanggal 29 Pebruari 2014 tidak ada.

#### **h. Fungsi date\_default\_timezone\_set ()**

Seperti telah dijelaskan diatas, informasi tanggal dan waktu berhubungan dengan berbagai zona waktu dan format tanggal yang berbeda. Jika program manipulasi tanggal yang kita buat tidak menghasilkan informasi waktu yang tepat, misalnya menghasilkan informasi waktu pukul 1 padahal waktu saat itu menunjukkan pukul 9, maka kita perlu mengatur zona waktu yang akan digunakan terlebih dahulu agar sesuai dengan zona waktu tempat kita berada. Fungsi yang dapat digunakan untuk mengatur zona waktu lokal ini adalah date\_default\_timezone\_set (). Sintaks fungsi ini:

```
date_default_timezone_set('nama_zona_waktu')
```

Parameter nama zona waktu merupakan data string yang berbentuk ‘nama kawasan/nama ibukota’. Untuk Indonesia nama zona waktunya adalah ‘Asia/Jakarta’. Contoh penggunaan:

```
$today = date('h:i:s, d-m-Y');
echo $today;
```

Contoh diatas akan menghasilkan informasi waktu ‘03:28:20 18-02-2014’ sementara pada saat program ini dijalankan waktu menunjukkan pukul 10:30:00 dan tanggal 18-02-2014. Disini ada perbedaan informasi waktu yang dihasilkan. Maka agar informasi waktu sesuai dengan waktu saat ini dimana program ini dijalankan (dijalankan di Indonesia), fungsi date\_default\_timezone\_set() perlu digunakan, sehingga kode sebelumnya akan menjadi:

```
date_default_timezone_set('Asia/Jakarta');
$today = date('h:i:s, d-m-Y');
echo $today;
```

Kode sekarang akan menghasilkan informasi waktu yang sesuai dengan waktu dimana program dijalankan (Dalam hal ini waktu yang sesuai dengan waktu komputer atau server yang digunakan).

### **Variabel Superglobal**

Variabel super global adalah variabel terpasang siap pakai yang selalu tersedia dalam semua lingkup aplikasi web yang dibuat. Tidak seperti variabel biasa yang hanya bisa digunakan pada sebuah file program (misalnya program tambah data mahasiswa) dimana variabel tersebut dibuat

yang artinya lingkupnya hanya sebatas program tersebut. Beberapa variabel yang telah ditetapkan dalam PHP adalah superglobal, yang berarti bahwa variabel tersebut selalu dapat diakses, terlepas dari lingkupnya, dan kita dapat mengaksesnya dari fungsi, kelas atau file program tanpa harus melakukan sesuatu yang khusus.

Variabel-variabel superglobal adalah:

- **`$_GLOBALS`**
- **`$_SERVER`**
- **`$_REQUEST`**
- **`$_POST`**
- **`$_GET`**
- **`$_FILES`**
- **`$_ENV`**
- **`$_COOKIE`**
- **`$_SESSION`**

### **Variabel `$_GLOBAL`**

`$_GLOBAL` adalah variabel super global PHP yang digunakan untuk mengakses variabel global dari mana saja didalam script PHP (juga dari dalam fungsi atau metode). PHP menyimpan semua

variabel global dalam sebuah array yang disebut \$GLOBALS [index]. Indeks berupa nama variabel. Contoh berikut memperlihatkan bagaimana penggunaan \$GLOBAL ini:

```
?php
$x = 75;
$y = 25;
function addition()
{
    $GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
}
41
addition();
echo $z;
?>
```

Dalam contoh di atas, karena z adalah variabel dalam array \$GLOBALS, maka variabel ini dapat diakses dari luar fungsi.

## Variabel \$\_SERVER

\$\_SERVER adalah variabel super global PHP yang menyimpan informasi tentang header, path, dan lokasi file program atau script. Berikut contoh penggunaan \$\_SERVER:

Berikut contoh penggunaan \$\_SERVER:

```
?php
echo $_SERVER['PHP_SELF'];
echo "<br>";
echo $_SERVER['SERVER_NAME'];
echo "<br>";
echo $_SERVER['HTTP_HOST'];
echo "<br>";
echo $_SERVER['HTTP_REFERER'];
echo "<br>";
echo $_SERVER['HTTP_USER_AGENT'];
echo "<br>";
echo $_SERVER['SCRIPT_NAME'];
?>
```

Tabel dibawah ini berisi daftar elemen-elemen yang paling sering digunakan didalam `$_SERVER`

Elemen	Deskripsi
<code>\$_SERVER['PHP_SELF']</code>	mengembalikan nama file script yang sedang berjalan
<code>\$_SERVER['GATEWAY_INTERFACE']</code>	mengembalikan versi CGI yang digunakan server
<code>\$_SERVER['SERVER_ADDR']</code>	mengembalikan alamat IP server host
<code>\$_SERVER['SERVER_NAME']</code>	mengembalikan nama dari server host (misalnya <a href="http://www.w3schools.com">www.w3schools.com</a> )
<code>\$_SERVER['SERVER_SOFTWARE']</code>	mengembalikan string yang mengidentifikasi server (misalnya Apache/2.2.24)
<code>\$_SERVER['SERVER_PROTOCOL']</code>	mengembalikan nama dan versi dari protokol informasi (misalnya HTTP/1.1)
<code>\$_SERVER['REQUEST_METHOD']</code>	mengembalikan metode request/permintaan yang digunakan untuk mengakses halaman (misalnya POST)
<code>\$_SERVER['REQUEST_TIME']</code>	mengembalikan timestamp dari awal

	permintaan (misalnya 1377687496)
<code>\$_SERVER['QUERY_STRING']</code>	mengembalikan string query jika halaman diakses melalui sebuah string query
<code>\$_SERVER['HTTP_ACCEPT']</code>	mengembalikan header Accept dari request saat ini
<code>\$_SERVER['HTTP_ACCEPT_CHARSET']</code>	mengembalikan header Accept_Charset dari request saat ini (misalnya utf-8, ISO-8859-1)
<code>\$_SERVER['HTTP_HOST']</code>	mengembalikan header Host dari request saat ini
<code>\$_SERVER['HTTP_REFERER']</code>	mengembalikan URL lengkap dari halaman saat ini (tidak dapat diandalkan karena tidak semua user agent (browser) mendukungnya)
<code>\$_SERVER['HTTPS']</code>	adalah script yang dikirimkan melalui protokol HTTP yang aman
<code>\$_SERVER['REMOTE_ADDR']</code>	mengembalikan alamat IP dari perangkat yang digunakan pengguna

	untuk mengakses halaman saat ini
<code>\$_SERVER['REMOTE_HOST']</code>	mengembalikan nama host yang digunakan pengguna untuk mengakses halaman saat ini
<code>\$_SERVER['REMOTE_PORT']</code>	mengembalikan port yang digunakan pada perangkat pengguna untuk berkomunikasi dengan server web
<code>\$_SERVER['SCRIPT_FILENAME']</code>	mengembalikan nama path absolut dari script yang dijalankan saat ini
<code>\$_SERVER['SERVER_ADMIN']</code>	mengembalikan nilai yang diberikan ke direktif SERVER_ADMIN dalam file konfigurasi server web (jika script anda berjalan diatas host virtual, maka akan berisi nilai yang didefinisikan untuk virtual host tersebut.
<code>\$_SERVER['SERVER_PORT']</code>	mengembalikan port pada komputer server yang digunakan oleh server web

	untuk berkomunikasi (misalnya 80)
<code>\$_SERVER['SERVER_SIGNATURE']</code>	mengembalikan versi server dan nama host virtual yang ditambahkan ke halaman yang digenerate oleh server
<code>\$_SERVER['PATH_TRANSLATED']</code>	mengembalikan path berbasis file sistem yang mengarah ke script saat ini
<code>\$_SERVER['SCRIPT_NAME']</code>	mengembalikan path script saat ini
<code>\$_SERVER['SCRIPT_URI']</code>	mengembalikan URI dari halaman web saat ini

### Variabel \$\_REQUEST

`$_REQUEST` digunakan untuk mengumpulkan data setelah pengiriman (submitting) sebuah formHTML terjadi. Contoh di bawah ini memperlihatkan sebuah form dengan field input dan tombol submit. Ketika pengguna mengirimkan data dengan mengklik "Submit", maka data form dikirim ke file yang ditentukan dalam atribut action dari tag `<form>`. Dalam contoh ini, kami arahkan pengiriman ke file ini sendiri untuk mengolah data form. Jika Anda ingin menggunakan file PHP lain untuk memproses data form, ganti nilai atribut action dengan nama file pilihan Anda. Setelah pengiriman, maka kita bisa menggunakan variabel super global `$_REQUEST` untuk mengumpulkan nilai dari field input:

```
<html>
<body>
<form method="post" action=<?php echo $_SERVER['PHP_SELF'];?>>
Name: <input type="text" name="fname">
<input type="submit">
</form>
<?php
$name = $_REQUEST['fname'];
echo $name;
?>
</body>
</html>
```

## Variabel \$\_POST

\$\_POST secara luas digunakan untuk mengumpulkan data form setelah proses pengiriman sebuah form HTML dengan method="post". \$\_POST juga banyak digunakan untuk melewatkkan variabel. Contoh di bawah ini memperlihatkan sebuah form dengan field input dan tombol submit. Ketika pengguna mengirimkan data dengan mengklik "Submit", maka data form dikirim ke file yang ditentukan dalam atribut action dari tag <form>. Dalam contoh ini, kami arahkan pengiriman ke file ini sendiri untuk mengolah data form. Jika Anda ingin menggunakan file PHP lain untuk memproses data form, ganti nilai atribut action dengan nama file pilihan Anda. Setelah pengiriman, maka kita bisa menggunakan variabel super global \$\_POST untuk mengumpulkan nilai dari field input:

```
<html>
<body>
<form method="post" action=<?php echo $_SERVER['PHP_SELF'];?>>
Name: <input type="text" name="fname">
<input type="submit">
</form>
<?php
$name = $_POST['fname'];
echo $name;
?>
</body>
</html>
```

## Variabel \$\_GET

\$\_GET juga dapat digunakan untuk mengumpulkan data form yang dikirimkan oleh form HTML dengan method="get". \$\_GET Juga dapat digunakan untuk mengumpulkan data yang dikirim melalui sebuah URL. Anggaplah kita memiliki halaman HTML yang berisi hyperlink dengan parameter.

```
<html>
<body>
<a href="test_get.php?subject=PHP&web=W3schools.com">Test $GET</a>
</body>
</html>
```

Ketika pengguna mengklik link "Test \$GET", maka parameter subject dan web dikirim ke file test\_get.php, kemudian nilai-nilai parameter tersebut di file test\_get.php dapat diakses dengan \$\_GET.

Berikut ini kode file test\_get.php yang digunakan:

```
<html>
<body>
<?php
echo "Belajar " . $_GET['subject'] . " di " . $_GET['web'];
?>
</body></html>
```

## Form

Seandainya saja tidak ada form HTML, tentulah programmer web akan kelimpungan ketika berusaha mengumpulkan data dari pemakai/pengunjung melalui aplikasinya. Form adalah alat pengumpul data yang utama bagi seorang programmer web. Inilah alasannya mengapa Anda harus menguasai alat ini dengan baik jika ingin menjadi seorang programmer web. Script PHP bekerjasama secara baik dengan form HTML. Elemen-elemen form pada sebuah halaman web secara otomatis akan tersedia pada script PHP.

Berikut contoh sebuah form (form.php):

```
<html>
<body>
<form action="welcome.php" method="post">
Name: <input type="text" name="nama" />
Age: <input type="text" name="usia" />
<input type="submit" />
</form>
</body>
</html>
```

Ketika seorang user mengisi isian data pada form diatas lalu menekan tombol submit, maka data yang diisikan kedalam form akan dikirim ke file PHP (welcome.php) yang ditentukan.

Berikut isi file welcome.php

```
<html>
<body>
Welcome <?php echo $_POST["nama"]; ?>!<br />
You are <?php echo $_POST["usia"]; ?> years old.
</body>
</html>
```

Variabel \$\_POST digunakan untuk mengambil data dari form. Ada satu variabel jenis ini, yaitu \$\_GET, yang akan digunakan jika method form menggunakan get. Catatan: Variabel-variabel \$\_POST dan \$\_GET adalah variabel super global yang disediakan oleh PHP untuk keperluan pengiriman data form. Artinya, ketika tombol submit diklik maka data form akan diletakkan dalam variabel ini secara otomatis. Maka ketika PHP akan mengambil data yang dikirim oleh form tersebut, akan menggunakan variabel-variabel ini untuk mengaksesnya.

## Validasi FORM

Input user sebaiknya divalidasi pada sisi browser menggunakan script klien, misalnya javascript, bila memungkinkan. Ini akan mempercepat proses validasi dan mengurangi komunikasi dengan server. Jika input user akan disimpan kedalam database, perlu dipertimbangkan melakukan proses validasi di server. Cara yang disarankan dalam melakukan proses validasi di server adalah mengirimkan data form ke form itu sendiri. Maka isi dari atribut action form berupa nama file dari form itu sendiri. Dalam contoh diatas, isi dari atribut action, yaitu berisi welcome.php, diganti form.php. Kemudian isi file welcome.php ditulis dibawah kode yang ada pada file form.php sehingga isi dari file form.php akan menjadi:

```
<html>
<body>
<form action="welcome.php" method="post">
Name: <input type="text" name="nama" />
Age: <input type="text" name="usia" />
<input type="submit" />
</form>
Welcome <?php echo $_POST["nama"]; ?>!<br />
You are <?php echo $_POST["usia"]; ?> years old.
</body>
</html>
```

Bagian yang dicetak tebal adalah kode yang semula ditulis di file welcome.php

### 3.5 Hubungan Antara PHP dan HTML

Halaman web biasanya disusun dari kode-kode html yang disimpan dalam sebuah file berekstensi .html. File html ini dikirimkan oleh server (atau file) ke browser, Kemudian browser menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah. Lain halnya dengan program php, program ini harus diterjemahkan oleh web-server sehingga menghasilkan kode html yang dikirim ke browser agar dapat ditampilkan. Program ini dapat berdiri sendiri ataupun disisipkan di antara kode-kode html sehingga dapat langsung ditampilkan bersama dengan kode-kode html tersebut. Program php dapat ditambahkan dengan mengapit program tersebut diantara tanda. Tanda-tanda tersebut biasanya disebut tanda untuk escaping (kabur) dari kode html. File html yang telah dibubuh program php harus diganti ekstensi-nya menjadi .php3 atau php. Php merupakan bahasa pemograman web yang bersifat server-side HTML=embedded scripting, di mana script-nya menyatu dengan HTML dan berada si server.

Artinya adalah sintaks dan perintah-perintah yang kita berikan akan sepenuhnya dijalankan di server tetapi disertakan HTML biasa. PHP dikenal sebagai bahasa scripting yang menyatu dengan tag HTML, dieksekusi di server dan digunakan untuk membuat halaman web yang dinamis seperti ASP (Active Server Pages) dan JSP (Java Server Pages).

## BAB IV

# INSTALASI TOOLS YANG DIGUNAKAN

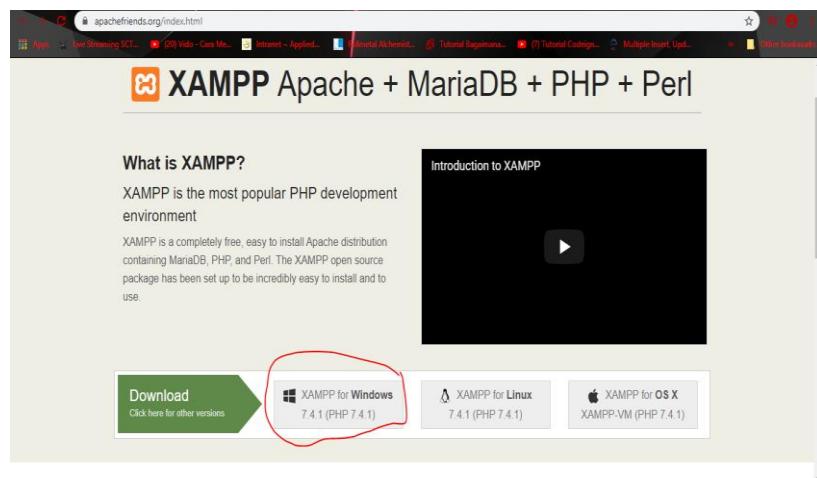
### 4. INSTALASI TOOLS YANG DIGUNAKAN

Pada Bab IV ini akan ditujukan cara instalasi Tools-tools yang akan digunakan dalam pembuatan sistem informasi ini.

#### 4.1 INSTAL XAMPP

##### 4.1.1 Download Xampp

- a. Kunjungi situs <https://www.apachefriends.org/index.html>
- b. klik langsung tombol download untuk Windows, Linux, dan Mac



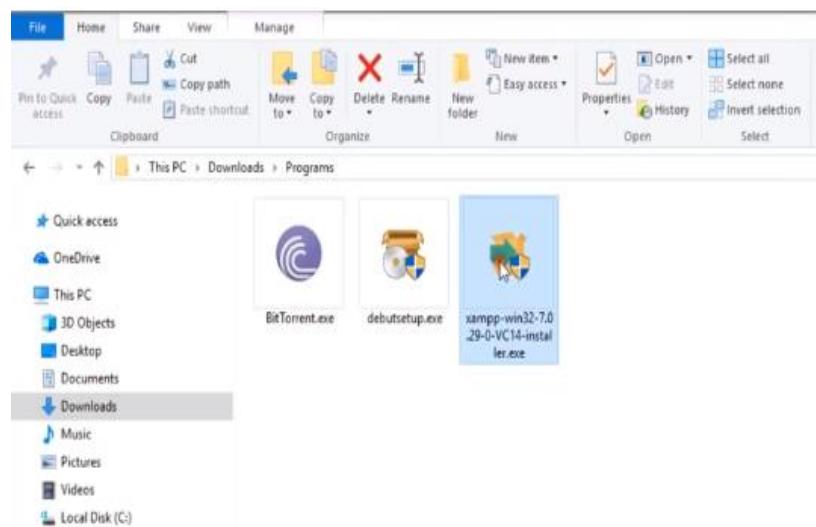
- c. Untuk mendownload versi yang lain silahkan klik tombol **Download** : *Click here for other versions*

The screenshot shows the 'Download' section of the XAMPP website. It features a table with three rows of software versions and their details:

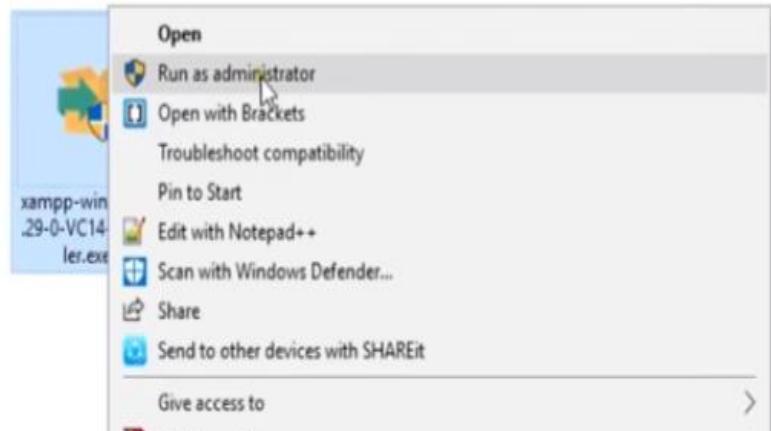
Version	Checksum	Size
7.2.26 / PHP 7.2.26	md5 sha1	Download (64 bit) 145 Mb
7.3.13 / PHP 7.3.13	md5 sha1	Download (64 bit) 146 Mb
7.4.1 / PHP 7.4.1	md5 sha1	Download (64 bit) 148 Mb

Below the table, there's a note about compatibility: "Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms here." To the right, there are sections for 'Documentation/FAQs' (with links to Linux, Windows, OS X, and OS X XAMPP-VM FAQs), 'Add-ons and Themes' (with icons for WordPress, Google Sheets, and others), and a note about Bitnami.

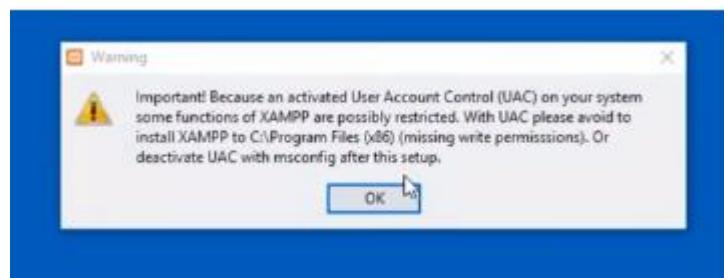
- d. Klik **download** dan tunggu sampai dengan selesai.  
e. Jika sudah silahkan buka **file.exe** dari XAMPP pada folder yang sudah terdownload tadi



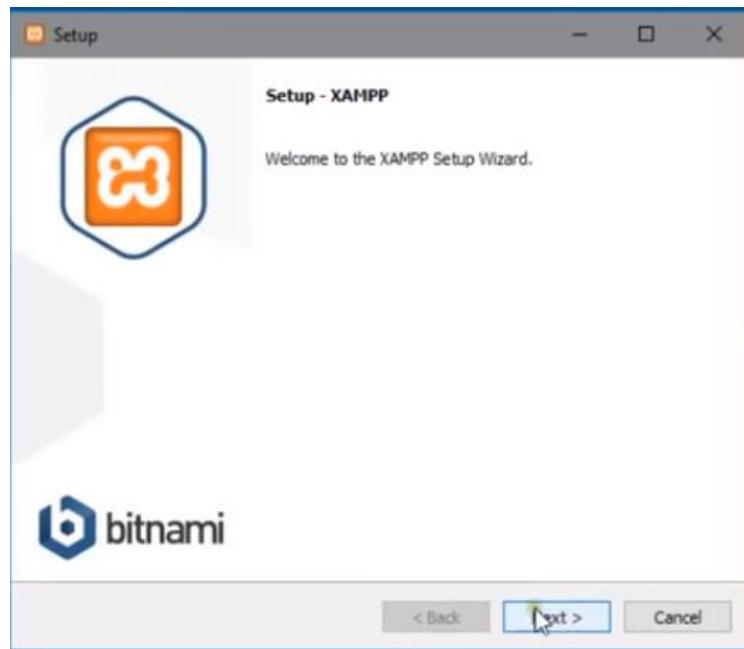
- f. Klik kanan pada file exe lalu pilih ***run as administrator***,  
Selanjutnya pilih YES



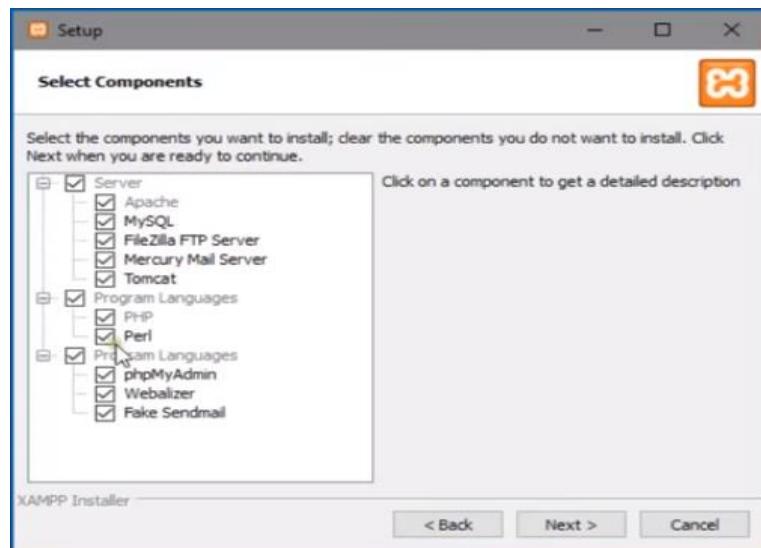
- g. Proses awal penginstallan biasanya akan ditampilkan icon bitnami, Klik **Ok** pada peringatan yang muncul



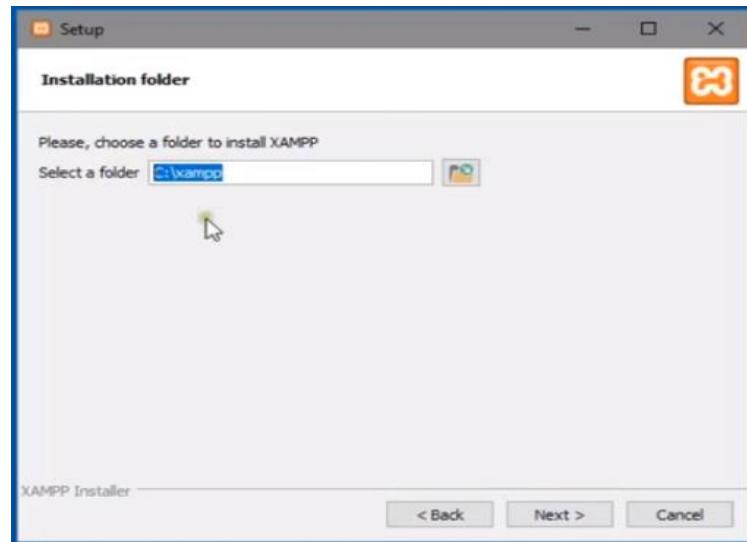
- h. Klik **Next** ketika jendela awal untuk menginstall XAMPP sudah muncul



- i. Pastikan semua komponen seperti : **Apache, MySQL, Filezilla, Phpmyadmin**, dan lainnya harus kita centang. Lalu klik **Next**



- j. Pilih direktori yang akan kita gunakan untuk menyimpan semua data-data XAMPP (termasuk data web yang akan kita buat nanti). Defaultnya yaitu direktori C. Jika sudah klik **Next**



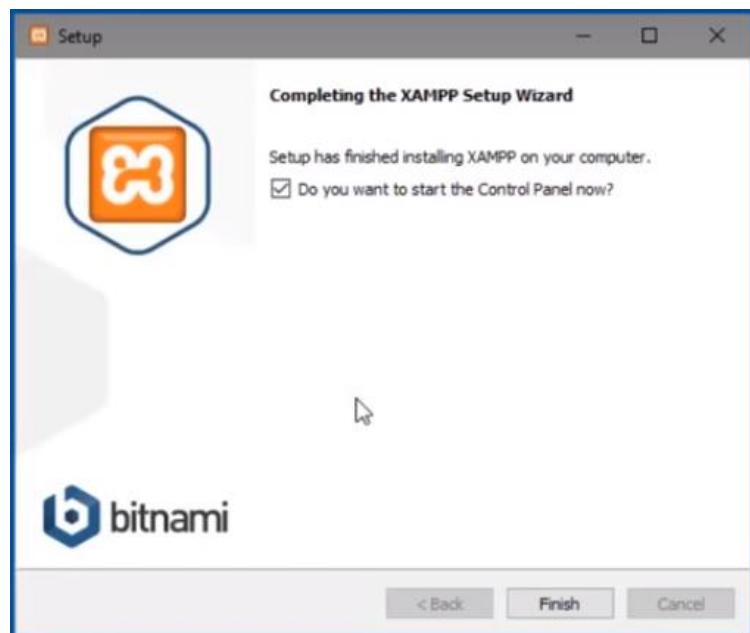
- k. Klik **Next** pada tampilan information bitnami



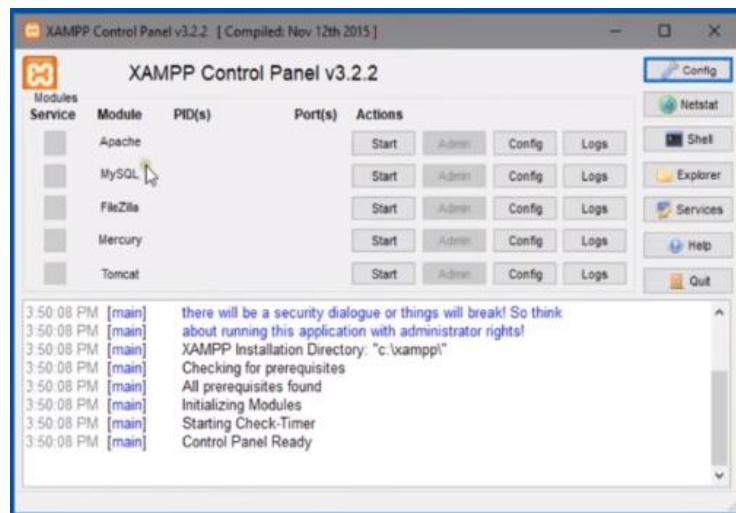
1. Silahkan klik next kembali pada jendela **Ready to Install**



- m. Penginstallan akan dimulai dan tunggu sampai selesai, jika sudah 100 % maka proses install akan selesai dan akan tampil gambar dibawah ini

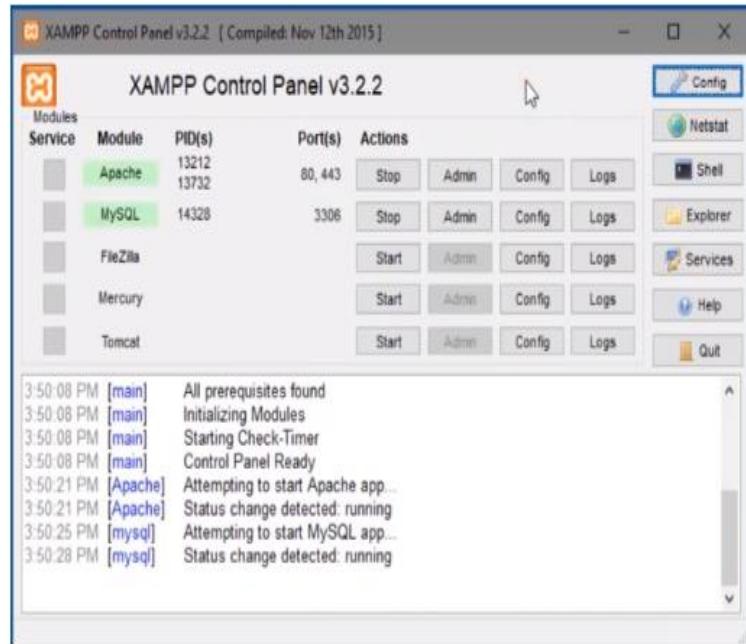


- n. Klik **Finish**, lalu jendela control panel XAMPP akan mulai dijalankan. Pilih bahasa yang akan digunakan lalu klik **Save**

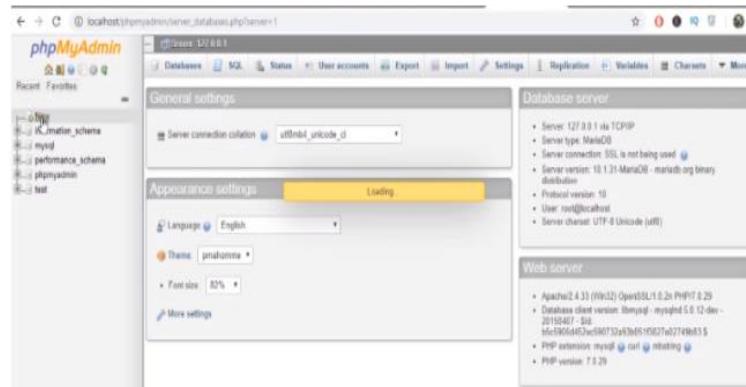


#### 4.1.2 Cara menjalan XAMPP

- f. Setelah jendela Control Panel dari XAMPP terbuka langkah selanjutnya adalah mengklik tombol **action start Apache dan MySQL**

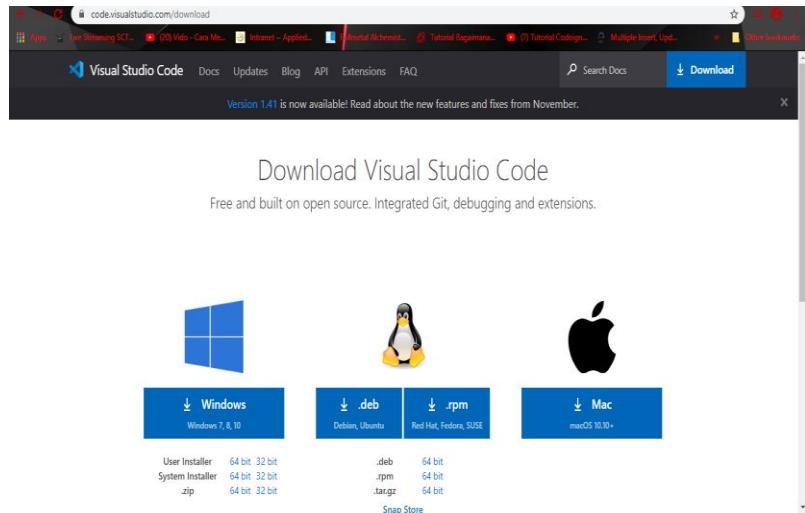


- g. Selanjutnya buka browser, disini saya menggunakan Google Chrome lalu ketikan Localhost/phpmyadmin pada address bar

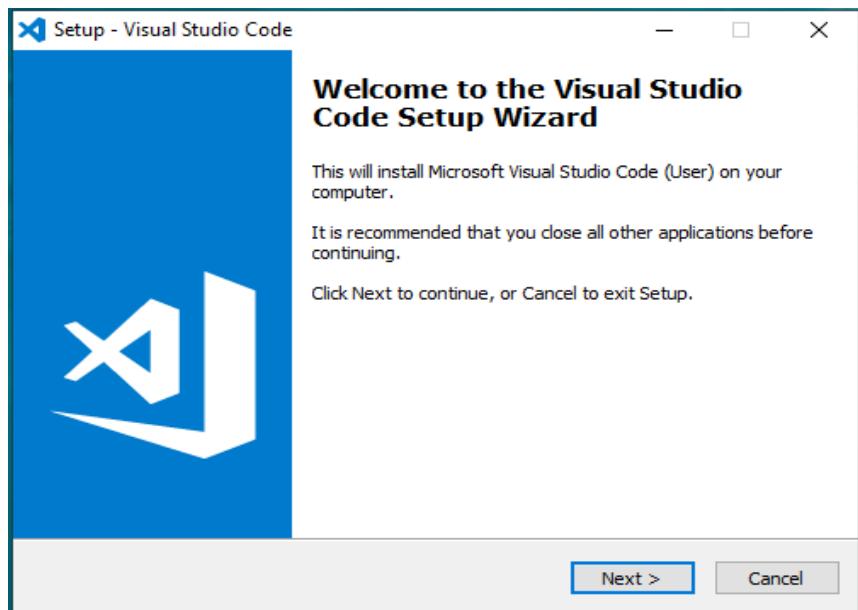


## 4.2 INSTAL VISUAL STUDIO CODE

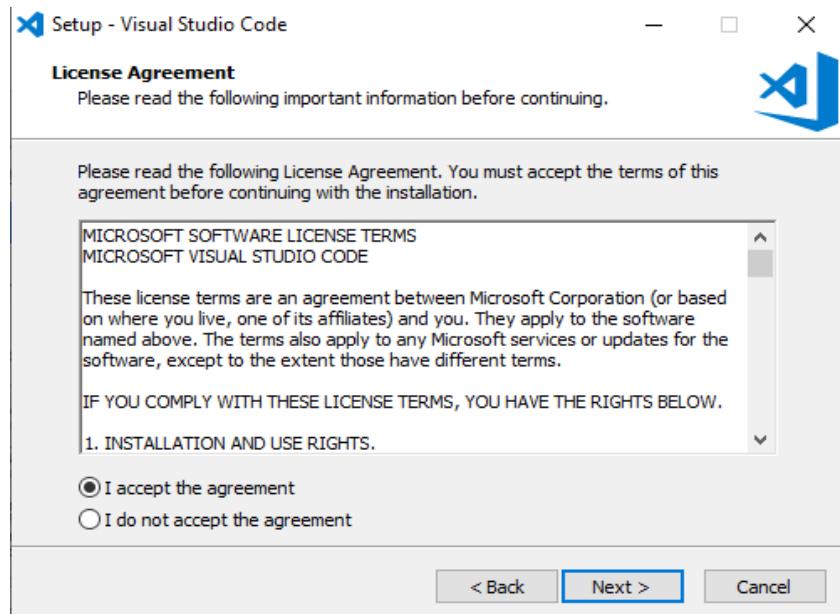
- c. Klik download di pojok kanan atas atau juga bisa dengan mengklik “*Download the Visual Studio Code installer for Windows.*” seperti gambar di bawah ini.



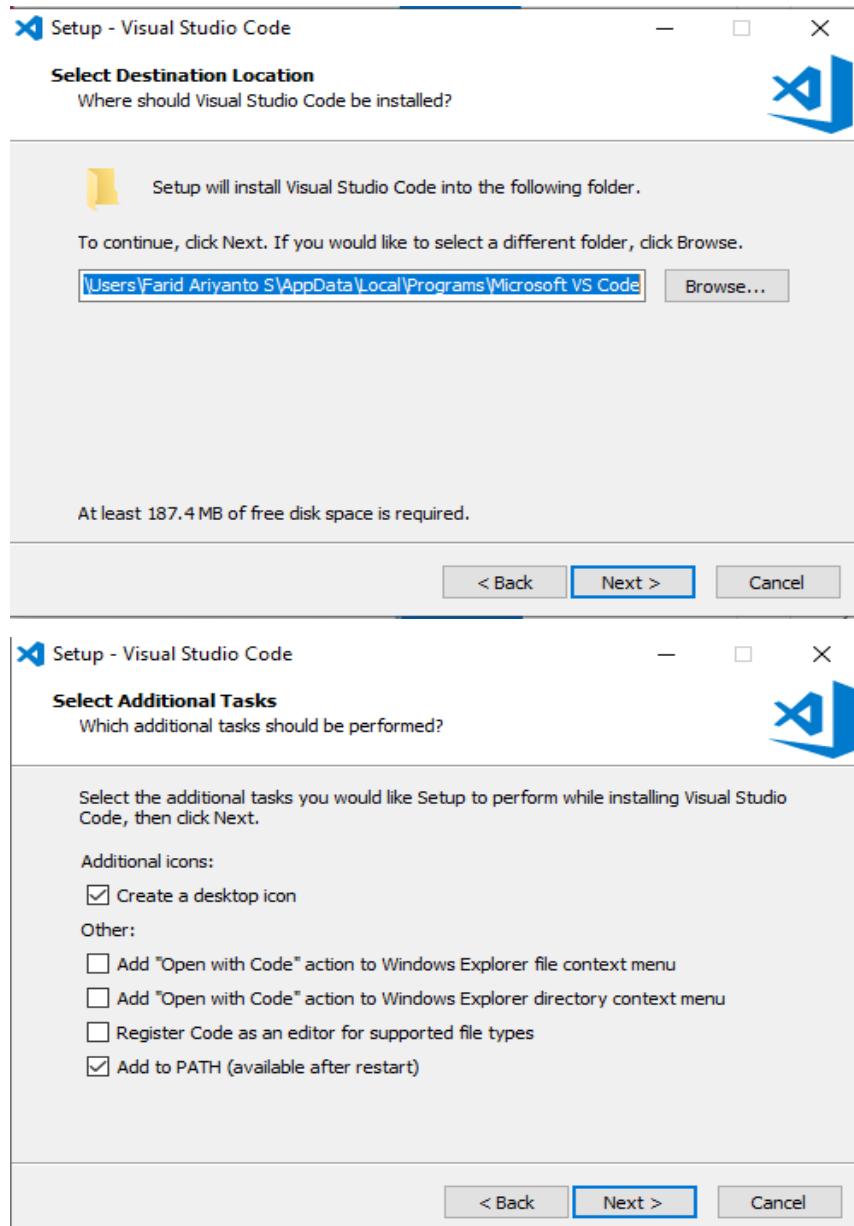
- d. klik/buka Aplikasi VSCode yang sudah di download, Kemudian ikuti kotak merah pada gambar klik *Next*.



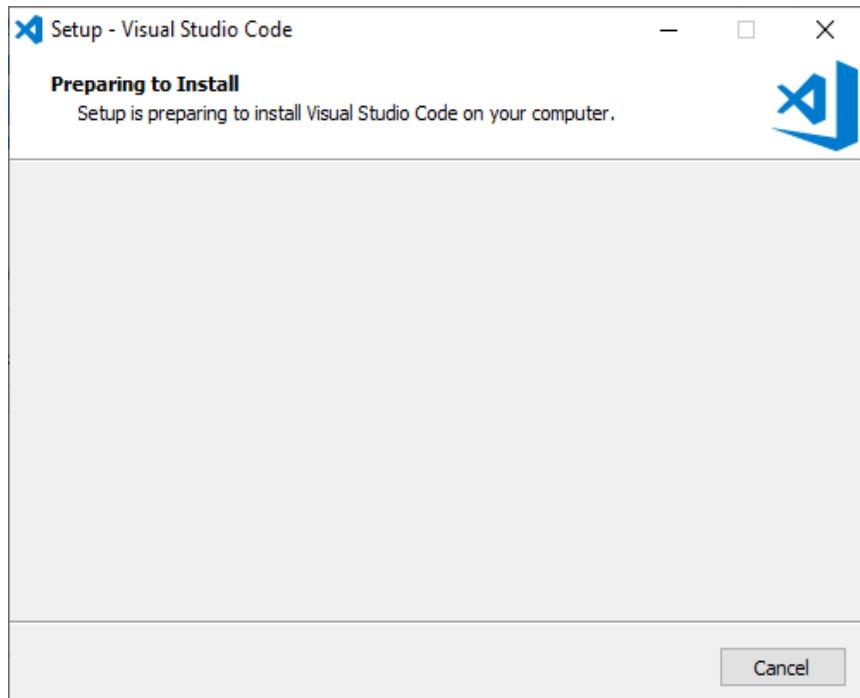
- e. Kemudian pilih “*I accept the agreement*” untuk menyetujui kebijakan dari VSCode lalu klik *Next*.



- f. Kemudian klik *Next* lagi itu hanya penempatan direktori file programnya, lalu Next untuk membuat folder nya, Kemudian Ceklis “*Create Desktop Icon*” jika ingin membuat shortcut VSCode nya kemudian ceklis “*Add to PATH* (available after restart) jika sudah klik *Next*.



- g. Kemudian Klik *Install* tunggu sampai “*Setup has finished installing Visual Studio Code on your komputer*” . kemudian klik *Finish* seperti gambar dibawah ini.



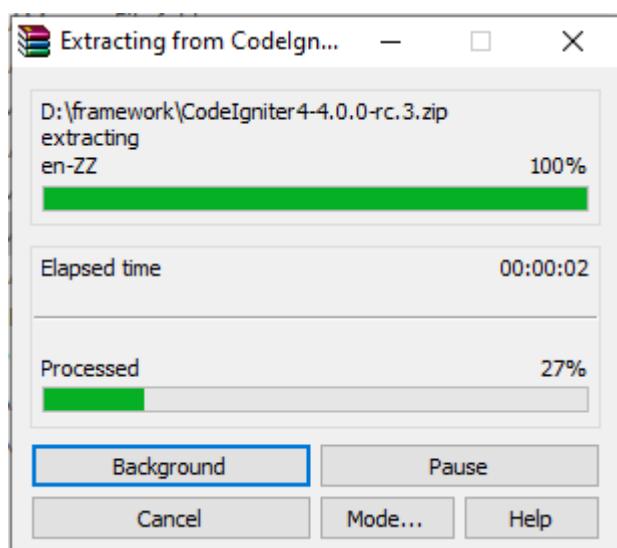
## 4.3 INSTAL FRAMEWORK CODE IGNITER

### 4.3.1 Download File CodeIgniter

- a. klik link <https://codeigniter.com/> lalu download framework codeigniter terbaru, saat ini yang terbaru adalah bcit-ci-CodeIgniter-3.1.11-0-gb73eb19 , dan akan terus berlanjut ke versi berikutnya.



- b. Ekstrak dan Install Codeigniter Framework



 cache	9/19/2019 7:08 PM	File folder
 config	9/19/2019 7:08 PM	File folder
 controllers	9/19/2019 7:08 PM	File folder
 core	9/19/2019 7:08 PM	File folder
 helpers	9/19/2019 7:08 PM	File folder
 hooks	9/19/2019 7:08 PM	File folder
 language	9/19/2019 7:08 PM	File folder
 libraries	9/19/2019 7:08 PM	File folder
 logs	9/19/2019 7:08 PM	File folder
 models	9/19/2019 7:08 PM	File folder
 third_party	9/19/2019 7:08 PM	File folder
 views	9/19/2019 7:08 PM	File folder
 .htaccess	9/19/2019 7:08 PM	HTACCESS File 1 KB
 index	9/19/2019 7:08 PM	Chrome HTML Do... 1 KB

### c. Konfigurasi Base Url pada CodeIgniter

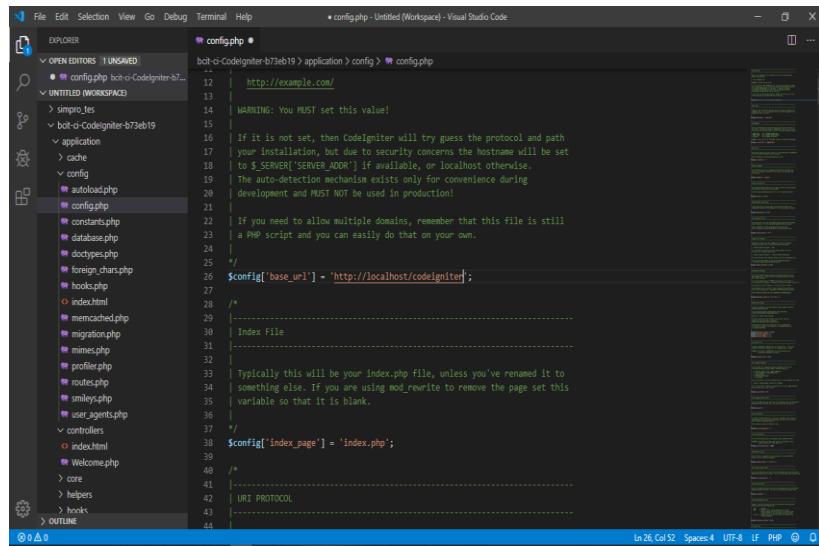
Setelah selesai melakukan instalasi codeigniter maka selanjutkan melakukan konfigurasi base url yang terdapat di folder application/config/config.php silahkan anda buka dengan teks editor Visual Studio Code. Lalu pada bagian kode dibawah ini:

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

Ubahlah menjadi dibawah ini:

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

Nilai tersebut harus sesuai dengan alamat/nama folder yang tentukan ketika menyalin file codeigniter ke dalam folder root web server.

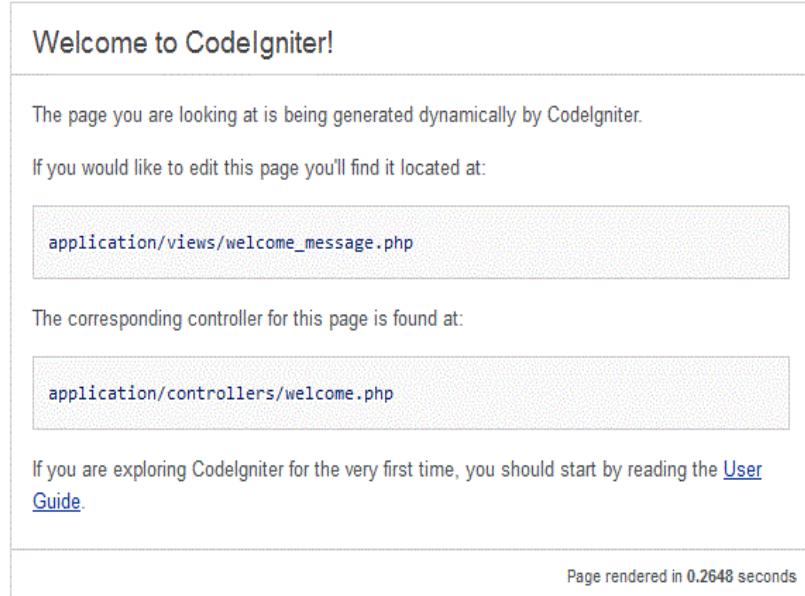


The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Debug, Terminal, Help.
- Editor Title:** config.php - Untitled (Workspace) - Visual Studio Code
- Explorer:** Shows the project structure under "UNLISTED (WORKSPACE)".
  - config.php (selected)
  - constants.php
  - database.php
  - doctypes.php
  - foreign\_chars.php
  - hooks.php
  - index.html
  - memcached.php
  - migration.php
  - mines.php
  - profiler.php
  - routes.php
  - smiley.php
  - user\_agents.php
  - controllers
  - indexhtml
  - Welcome.php
  - core
  - helpers
  - hooks
- Code Editor:** Displays the content of config.php.

```
1 http://example.com/
2
3 WARNING: You MUST set this value!
4
5 If it is not set, then CodeIgniter will try guess the protocol and path
6 to your installation, but due to security concerns the hostname will be set
7 to $_SERVER['SERVER_NAME'] if available, or localhost otherwise.
8 The auto-detection mechanism exists only for convenience during
9 development and MUST NOT be used in production!
10
11 If you need to allow multiple domains, remember that this file is still
12 a PHP script and you can easily do that on your own.
13
14 /*
15  $config['base_url'] = 'http://localhost/codeigniter';
16 */
17
18 /**
19  |-----[ Index File
20  |
21  | Typically this will be your index.php file, unless you've renamed it to
22  | something else. If you are using mod_rewrite to remove the page set this
23  | variable so that it is blank.
24  |
25  */
26 $config['index_page'] = 'index.php';
27
28 /**
29  |-----[ URI PROTOCOL
30  |
31  |
32  |
33  */
34
35 /**
36  |
37  */
38
39 /**
40  |
41  |
42  |
43  |
44  |-----
```
- Bottom Status Bar:** Line 26, Column 52, Spaces: 4, UTF-8, LF, PHP, etc.

d. Buka url di browser <http://localhost/codeigniter>



The screenshot shows a web browser window with the following content:

**Welcome to CodeIgniter!**

The page you are looking at is being generated dynamically by CodeIgniter.

If you would like to edit this page you'll find it located at:

`application/views/welcome_message.php`

The corresponding controller for this page is found at:

`application/controllers/welcome.php`

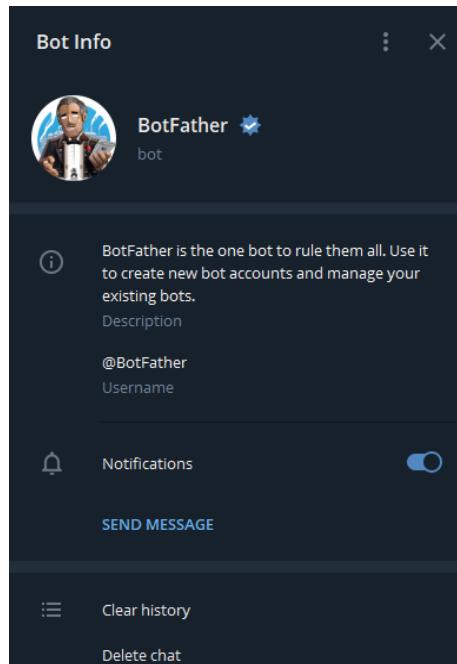
If you are exploring CodeIgniter for the very first time, you should start by reading the [User Guide](#).

Page rendered in 0.2648 seconds

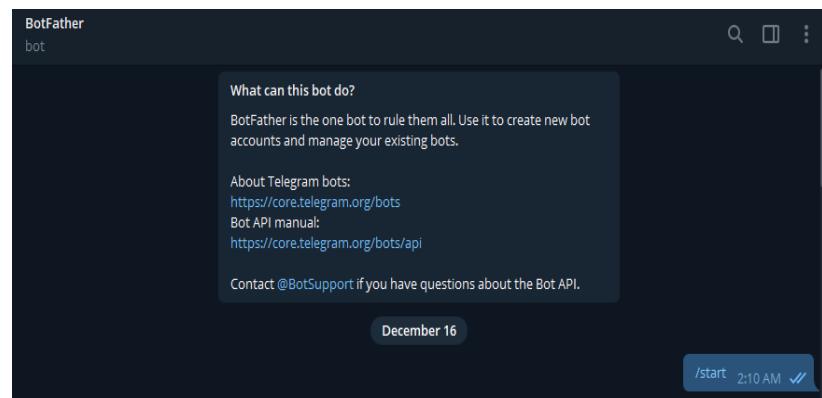
## 4.4 INSTAL BOT TELEGRAM

### 4.4.1 Mendaftar ke BotFather

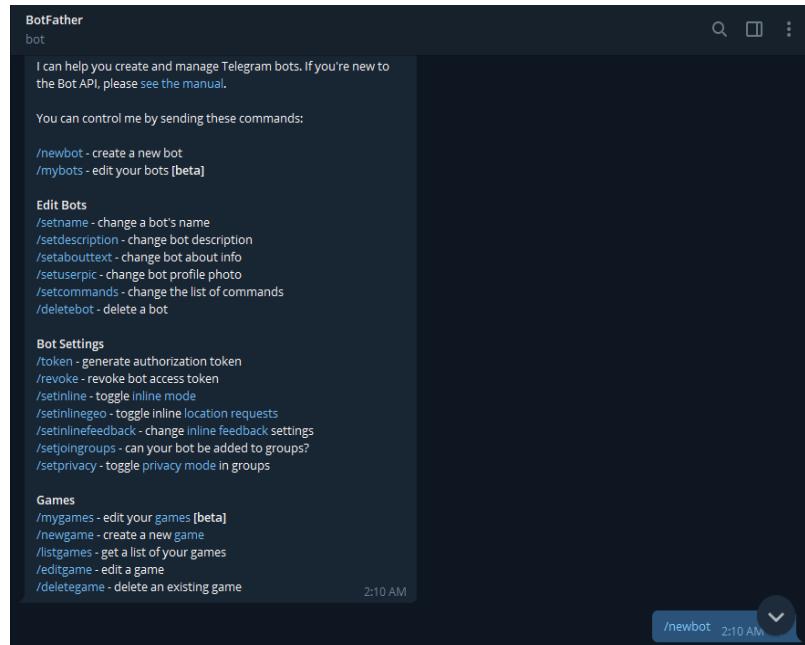
- a. Pilih menu pencarian dan ketikan “Botfather” atau “@Botfather”.



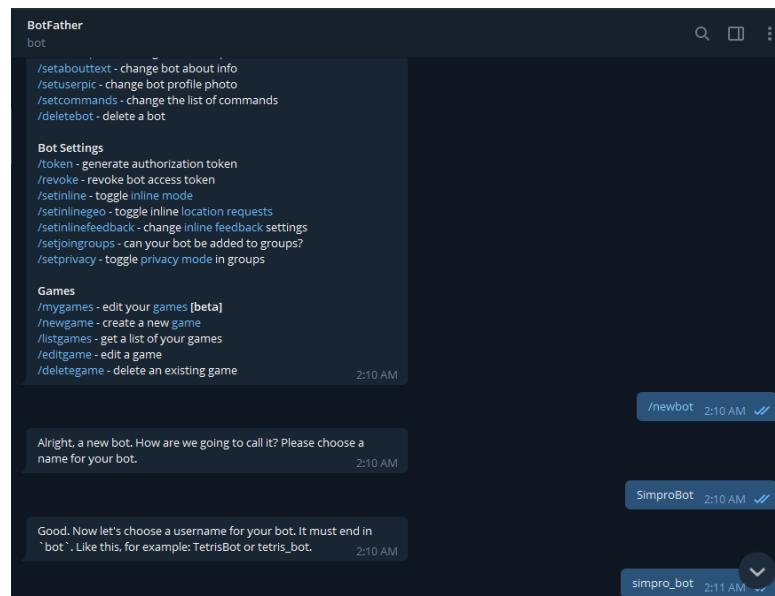
- b. Lalu klik **START** untuk memulai interaksi dengan @Botfather. Kita akan disuguhkan menu pilihan.



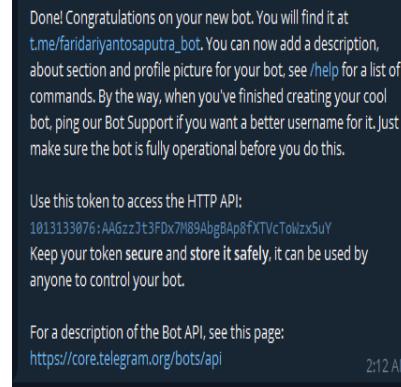
- c. Karena tujuan kita membuat bot maka kirim: /newbot. Jawab pertanyaan-pertanyaan dari Botfather.



- Nama bot, sebagai contoh saya mengisinya: SimproBot
- Username bot, wajib menggunakan akhiran kata “bot”. Saya isi: faridaritantosaputra\_**bot**



- d. Jika username tersedia, saat itu juga kita akan mendapatkan TOKEN yang merupakan identifier dari bot itu sendiri.



Done! Congratulations on your new bot. You will find it at [t.me/faridariyantosaputra\\_bot](https://t.me/faridariyantosaputra_bot). You can now add a description, about section and profile picture for your bot, see /help for a list of commands. By the way, when you've finished creating your cool bot, ping our Bot Support if you want a better username for it. Just make sure the bot is fully operational before you do this.

Use this token to access the HTTP API:  
1013133076:AAGzzJt3FDx7M89AbgBAp8fXTVcToWzx5uY  
Keep your token **secure** and store it **safely**, it can be used by anyone to control your bot.

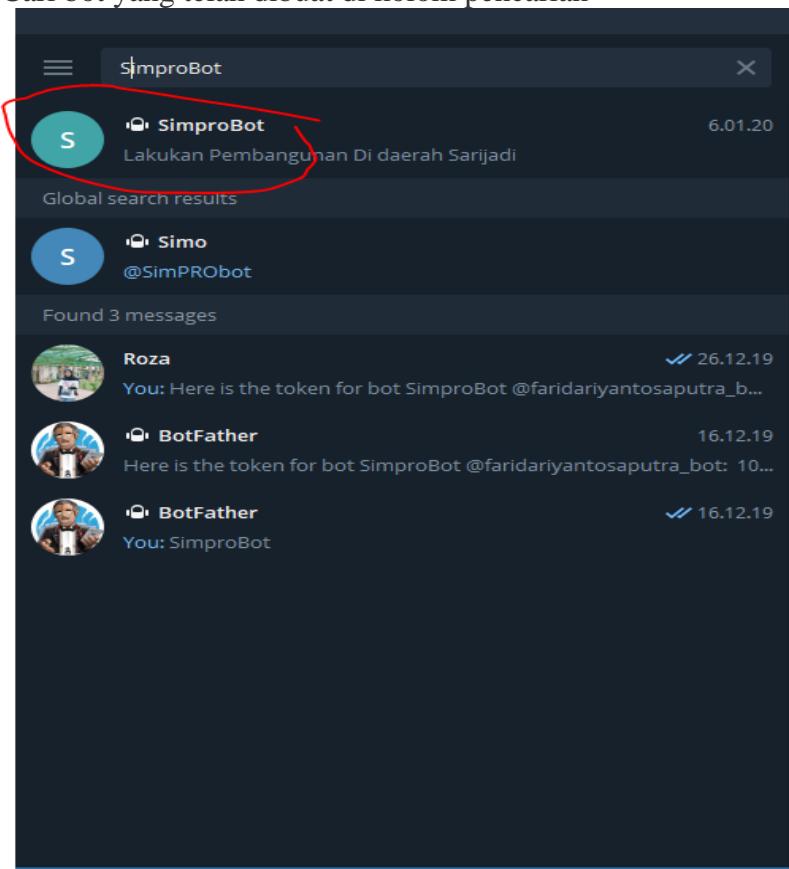
For a description of the Bot API, see this page:  
<https://core.telegram.org/bots/api>

2:12 AM

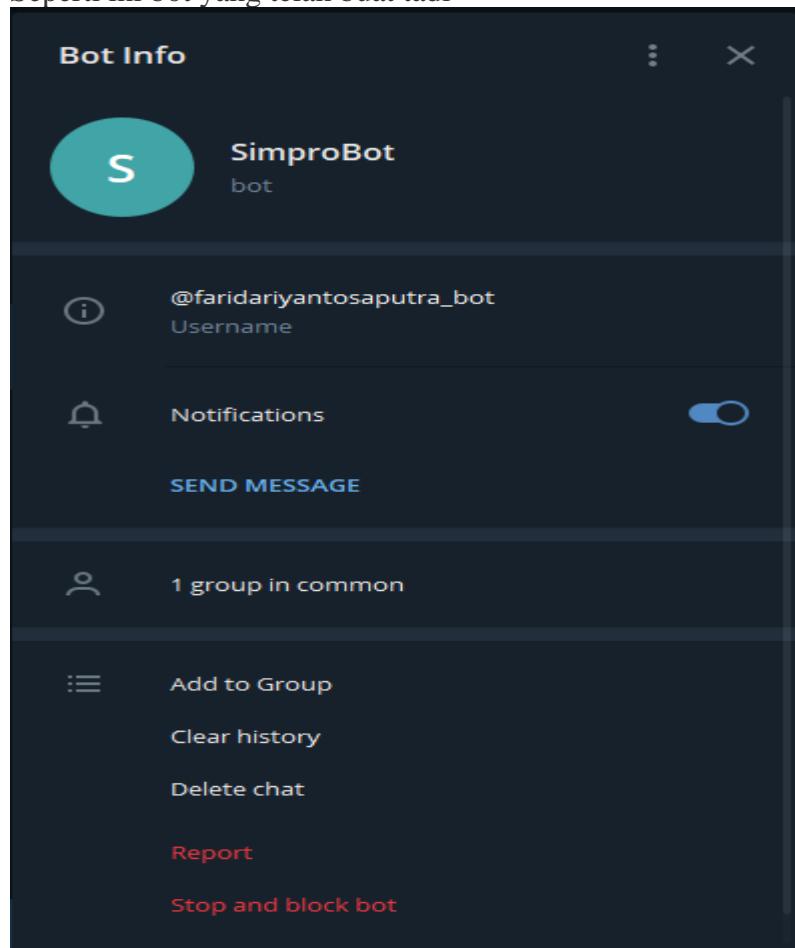
Catat baik-baik TOKEN yang didapat dari BotFather. Pada contoh ini TOKEN adalah:

1013133076:AAGzzJt3FDx7M89AbgBAp8fXTVcToWzx5uY

- e. Cari bot yang telah dibuat di kolom pencarian



- f. Seperti ini bot yang telah buat tadi



# BAB V

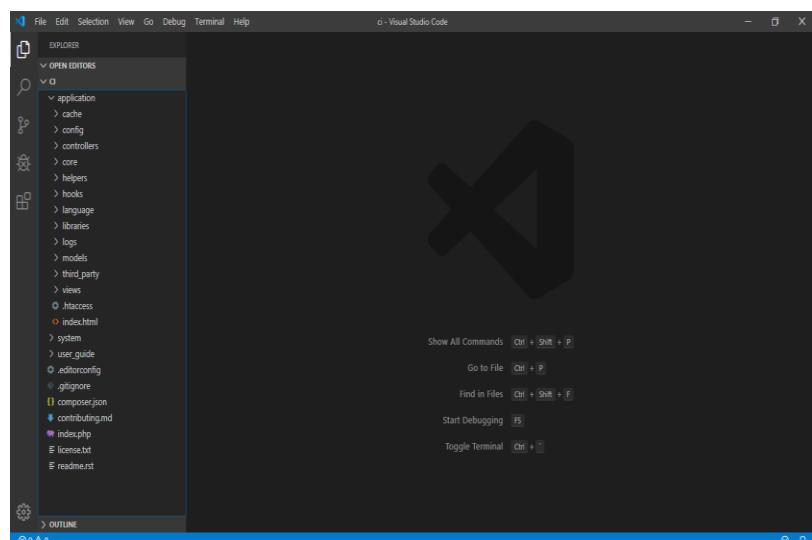
## MEMBUAT HELLO WORLD MENGGUNAKAN CODEIGNITER

### 5. Membuat Hello World Menggunakan Codeigniter

Sebelum kita membuat aplikasi menggunakan codeigniter dan tools-tools yang telah dijelaskan tadi. Kita terlebih dahulu akan mencoba membuat Hello World menggunakan CodeIgniter.

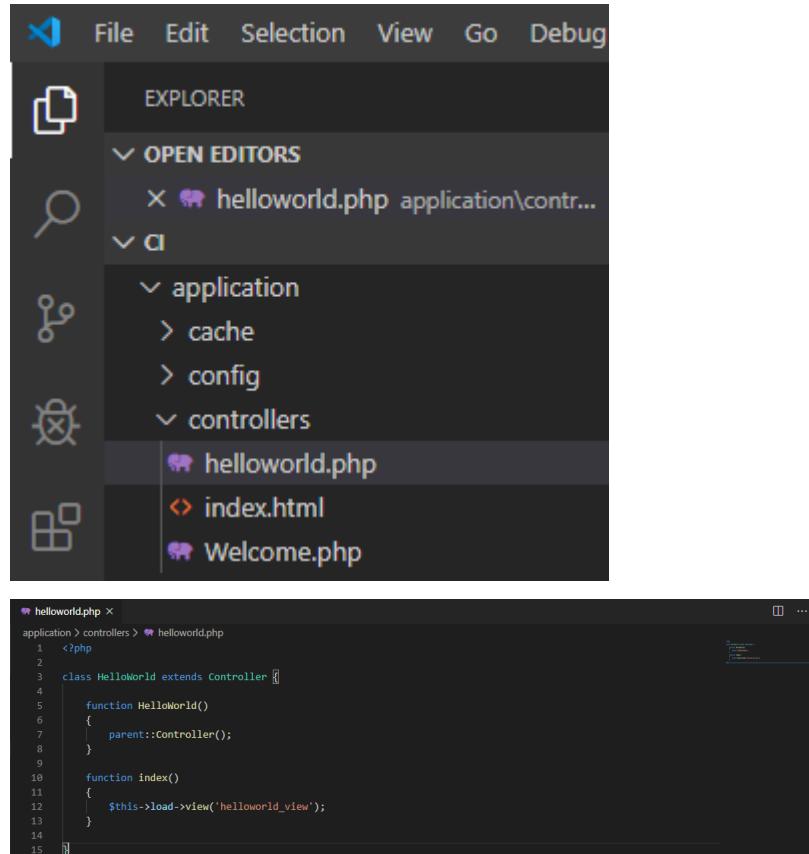
#### 5.1 Membuat Hello World

- Buka folder CodeIgniter yang telah didownload di dalam Visual Studio Code



Gambar 5.1 Membuat Hello world

- b. Buatlah file bernama **helloworld.php** dan taruh didalam folder controllers. Kemudian ketikkan sintaks seperti dibawah :



The screenshot shows a code editor interface with a dark theme. At the top is a menu bar with File, Edit, Selection, View, Go, and Debug. Below the menu is an Explorer sidebar with icons for File, Open Editors, and CI. The Open Editors section shows a file named "helloworld.php" under the application\contr... path. The CI section shows application, cache, config, and controllers. Under controllers, there are three files: helloworld.php, index.html, and Welcome.php. The main editor area has a tab for "helloworld.php" which contains the following PHP code:

```
application > controllers > helloworld.php
1 <?php
2
3 class HelloWorld extends Controller []
4
5     function HelloWorld()
6     {
7         parent::Controller();
8     }
9
10    function index()
11    {
12        $this->load->view('helloworld_view');
13    }
14
15 }
```

The screenshot shows the Visual Studio Code interface. The left sidebar displays a file tree with the following structure:

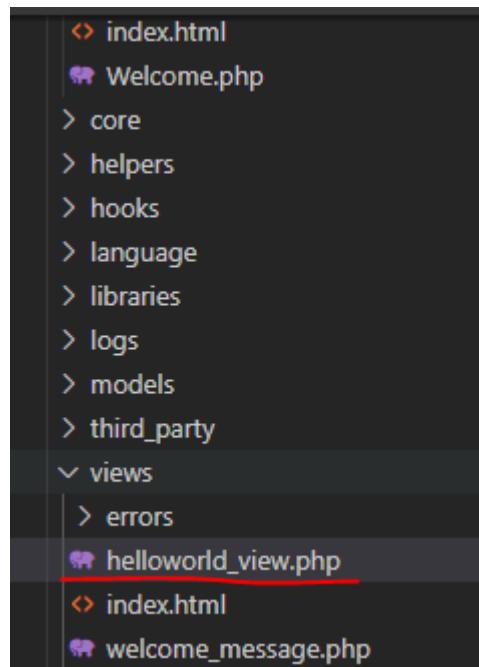
- EXPLORE
- OPEN EDITORS
- application
- helloworld.php
- application\_controller.php
- cache
- config
- controllers
- helloworld.php
- index.html
- Welcome.php
- core
- helpers
- hooks
- language
- libraries
- logs
- models
- third\_party
- views
- index.php
- index.html
- helloworld\_view.php
- welcome\_message.php

The right pane shows the code editor with the file `helloworld.php` open. The code is as follows:

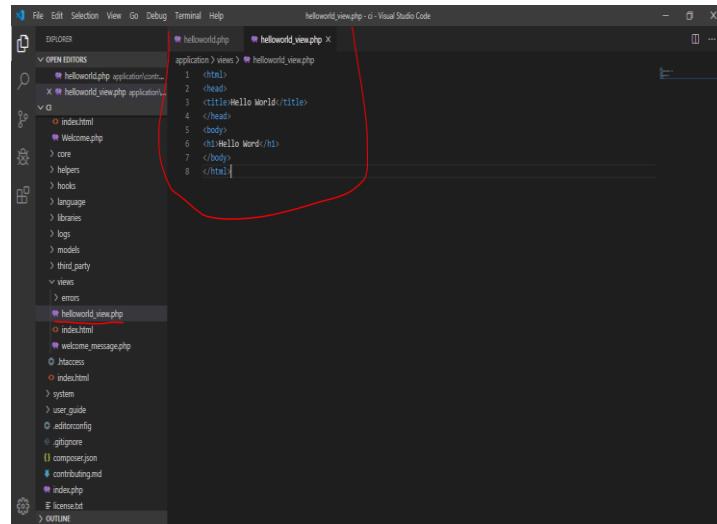
```
<?php  
class HelloWorld extends Controller  
{  
    function HelloWorld()  
    {  
        parent::Controller();  
    }  
    function index()  
    {  
        $this->load->view('helloworld_view');  
    }  
}  
?
```

Fungsi **HelloWorld()** merupakan fungsi constructor pada class **HelloWorld**. Nama class harus sama dengan nama fungsi constructor-nya. Dan juga nama file harus sama dengan nama classnya.

- c. Buatlah file bernama **helloworld\_view.php** dan taruh dalam folder views. Kemudian ketikkan sintaks seperti dibawah :



```
application > views > helloworld_view.php
1 <html>
2 <head>
3 <title>Hello World</title>
4 </head>
5 <body>
6 <h1>Hello Word</h1>
7 </body>
8 </html>
```



- d. Sekarang anda akses lagi URL

**Hello Word**

# BAB VI

## MEMBUAT SISTEM INFROMASI MENGGUNAKAN CODEIGNITER DENGAN NOTIFIKASI TELEGRAM

### 6. Membuat Sistem Infromasi Menggunakan Codeigniter Dengan Notifikasi Telegram

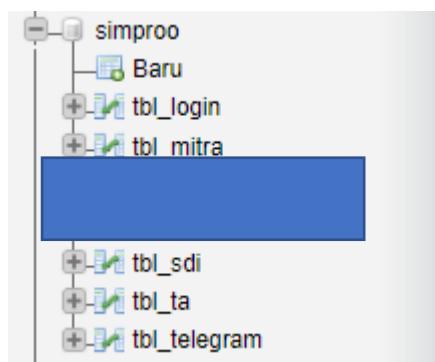
Pada Bab VI ini akan menjelaskan bagaimana cara membuat sistem informasi mengguakan codeigniter dengan notifikasi melalui telegram.

#### 6.1 Membuat database

1. Buat database dengan nama simproo



2. Buat beberapa table seperti gambar dibawah :



3. Buat tabel login dengan nama tabel `tbl_login` dengan field seperti berikut :

The screenshot shows the MySQL Workbench interface for creating the `tbl_login` table. The table structure is defined as follows:

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	<code>id_user</code>	int(20)			Tidak	Tidak ada			Ubah  Hapus  Lainnya
2	<code>username</code>	varchar(30)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus  Lainnya
3	<code>password</code>	varchar(255)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus  Lainnya
4	<code>level</code>	enum('1', '2', '3', '4', '5')	latin1_swedish_ci	Ya	NULL	1=Teknis;2=Optima;3=Kabis;4=Mitra;5=Optimaa			Ubah  Hapus  Lainnya
5	<code>status</code>	int(11)			Tidak	Tidak ada			Ubah  Hapus  Lainnya
6	<code>tgl</code>	datetime			Tidak	Tidak ada			Ubah  Hapus  Lainnya
7	<code>notel</code>	varchar(12)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus  Lainnya

Below the table structure, there are buttons for printing, previewing, normalizing, and adding columns. A 'Tambahkan' button is also present.

**Indeks** (Indexes) section:

Tindakan	Nama kunci	Jenis	Unik	Dipadatkan	Kolom	Kardinalitas	Penyortiran	Tak Ternilai	Komentar
Ubah  Hapus	PRIMARY	BTREE	Ya	Tidak	<code>id_user</code>	5	A	Tidak	

Buttons for creating an index and specifying the column are shown below.

4. Buat tabel mitra dengan nama tabel `tbl_mitra` dengan field seperti berikut :

The screenshot shows the MySQL Workbench interface for creating the `tbl_mitra` table. The table structure is defined as follows:

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	<code>id_mitra</code>	int(10)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah  Hapus  Lainnya
2	<code>nama_mitra</code>	varchar(30)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus  Lainnya

Below the table structure, there are buttons for printing, previewing, normalizing, and adding columns. A 'Tambahkan' button is also present.

**Indeks** (Indexes) section:

Tindakan	Nama kunci	Jenis	Unik	Dipadatkan	Kolom	Kardinalitas	Penyortiran	Tak Ternilai	Komentar
Ubah  Hapus	PRIMARY	BTREE	Ya	Tidak	<code>id_mitra</code>	4	A	Tidak	

Buttons for creating an index and specifying the column are shown below.

5. Buat tabel mitra dengan nama tabel `tbl_mitra` dengan field seperti berikut :

#	Nama	Jenis	Penyortiran	Atribut	Tak Ter nilai	Bawaan	Komentar	Ekstra	Tindakan
1	<code>id_boq</code>	int(11)			Tidak	Tidak ada	AUTO_INCREMENT		Ubah Hapus Lainnya
2	<code>pekerjaan</code>	varchar(100)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
3	<code>sto</code>	varchar(100)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
4	<code>lokasi</code>	varchar(100)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
5	<code>desimator</code>	enum('DC-OF-SM-48D', 'DC-OF-SM-288D', 'AC-OF-SM-12...')		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
6	<code>uraian_pekerjaan</code>	enum('Pengadaan dan pemasangan Kabel Duct Fiber Optic', 'Pemasangan Kabel Duct Fiber Optic')		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
7	<code>satuan</code>	enum('meter', 'core', 'pcs', 'tlik')		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
8	<code>status</code>	enum('sedang dikerjakan', 'selesai', 'batai', '')		latin1_swedish_ci	Ya	NULL			Ubah Hapus Lainnya

6. Buat tabel ta dengan nama tabel `tbl_ta` dengan field seperti berikut :

#	Nama	Jenis	Penyortiran	Atribut	Tak Ter nilai	Bawaan	Komentar	Ekstra	Tindakan
1	<code>id_survey</code>	int(11)			Tidak	Tidak ada	AUTO_INCREMENT		Ubah Hapus Lainnya
2	<code>sto</code>	varchar(100)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
3	<code>lokasi</code>	varchar(100)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
4	<code>gambar_olt</code>	varchar(100)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
5	<code>gambar_ftr</code>	varchar(100)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
6	<code>gambar_odc</code>	varchar(100)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
7	<code>gambar_otp</code>	varchar(100)		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya
8	<code>status</code>	enum('sedang dikerjakan', 'selesai', 'batai', '')		latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Lainnya

Tindakan	Nama kunci	Jenis	Unik	Dipadatkan	Kolom	Kardinalitas	Penyortiran	Tak Ter nilai	Komentar
Ubah Hapus	PRIMARY	BTREE	Ya	Tidak	<code>id_survey</code>	1	A	Tidak	

Buat indeks pada	1	kolom	Kirim
------------------	---	-------	-------

7. Buat tabel telegram dengan nama tabel `tbl_telegram` dengan field seperti berikut :

#	Nama	Jenis	Penyortiran	Atribut	Tak Ter nilai	Bawaan	Komentar	Ekstra	Tindakan
1	<code>id</code>	int(12)			Tidak	Tidak ada		<code>AUTO_INCREMENT</code>	Ubah  Hapus  Lainnya
2	<code>no_telp</code>	varchar(12)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus  Lainnya
3	<code>id_telegram</code>	varchar(12)	latin1_swedish_ci		Tidak	Tidak ada			Ubah  Hapus  Lainnya

Pilih Semua Dengan pilihan: Jelajahi Ubah Hapus Utama Unik Indeks Teks penuh Add to central columns Remove from central columns

Cetak Usulkan struktur tabel Lacak tabel Move columns Normalisasi

Tambahkan 1 kolom setelah `id_telegram`

Indeks Tindakan Nama kunci Jenis Unik Dipadatkan Kolom Kardinalitas Penyortiran Tak Ter nilai Komentar Ubah Hapus PRIMARY BTREE Ya Tidak id 2 A Tidak

Buat indeks pada 1 kolom

Partitions No partitioning defined!

Partition table

8. Konfigurasi nama database ke dalam codeigniter

```

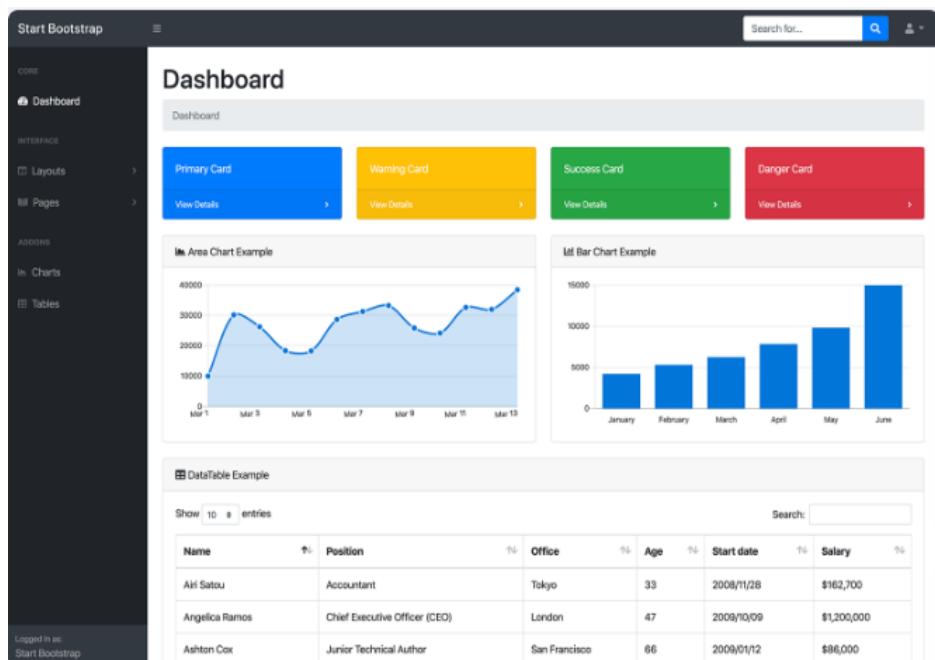
application > config > database.php
64 | However, this may cause high memory usage, especially if you run
65 | a lot of SQL queries ... disable this to avoid that problem.
66 |
67 | The $active_group variable lets you choose which connection group to
68 | make active. By default there is only one group (the 'default' group).
69 |
70 | The $query_builder variables lets you determine whether or not to load
71 | the query builder class.
72 */
73 $active_group = 'default';
74 $query_builder = TRUE;
75
76 $db['default'] = array(
77     'dsn' => '',
78     'hostname' => 'localhost',
79     'username' => 'root',
80     'password' => '',
81     'database' => 'simproo',
82     'dbdriver' => 'mysqli',
83     'dbprefix' => '',
84     'pconnect' => FALSE,
85     'db_debug' => (ENVIRONMENT !== 'production'),
86     'cache_on' => FALSE,
87     'cachedin' => '',
88     'char_set' => 'utf8',
89     'dbcollat' => 'utf8_general_ci',
90     'swap_pre' => '',
91     'encrypt' => FALSE,
92     'compress' => FALSE,
93     'stricton' => FALSE,
94     'failover' => array(),
95     'save_queries' => TRUE
96 );

```

## 6.2 Mendownload Template Untuk Aplikasi Yang akan di Bangun

Dalam pembuatan aplikasi dengan CodeIgniter ini membutuhkan sebuah template, disini template yang digunakan adalah template SB ADMIN.

Template SB Admin ini dapat di download di website Bootstrap langsung atau bisa langsung kunjungi <https://startbootstrap.com/templates/sb-admin>.



Template SB Admin adalah template dari Bootstrap 4, Dengan fitur :

- Tema berbasis SCSS dengan skrip bawaan untuk mengkompilasi Pug dan CSS
- Layout berbasis flexbox dengan opsi navigasi tetap dan statis
- Menu bilah samping yang dapat diubah dengan status sakelar yang intuitif
- Penataan panel khusus

- Grafik interaktif dengan plugin Chart.js
- Tabel interaktif dengan plugin dataTables
- Login, Registrasi, Lupa Kata Sandi, 404, dan halaman HTML starter Kosong

### 6.2.1 Konfigurasi Template Kedalam CodeIgiter

Sebelum memulai membuat template, silahkan lakukan konfigurasi terlebih dahulu :

- Extract file yang telah di download melalui <https://startbootstrap.com/templates/sb-admin>.
- Konfigurasi Base URL Cara mengatur Base URL, silahkan buka file config/config.php. Lalu isi bagian \$config['base\_url'] = ""; menjadi seperti ini:



```

config.php X
application > config > config.php
12 | http://example.com/
13 |
14 | WARNING: You MUST set this value!
15 |
16 | If it is not set, then CodeIgniter will try guess the protocol and path
17 | of your installation, but due to security concerns the hostname will be set
18 | to $_SERVER['SERVER_ADDR'] if available, or localhost otherwise.
19 | The auto-detection mechanism exists only for convenience during
20 | development and MUST NOT be used in production!
21 |
22 | If you need to allow multiple domains, remember that this file is still
23 | a PHP script and you can easily do that on your own.
24 |
25 |
26 $config['base_url'] = 'http://localhost/simpro_tes';
27 |
28 /*

```

- Berikutnya kita harus meload sebuah *helper* ke memori secara otomatis melalui autoload.php.

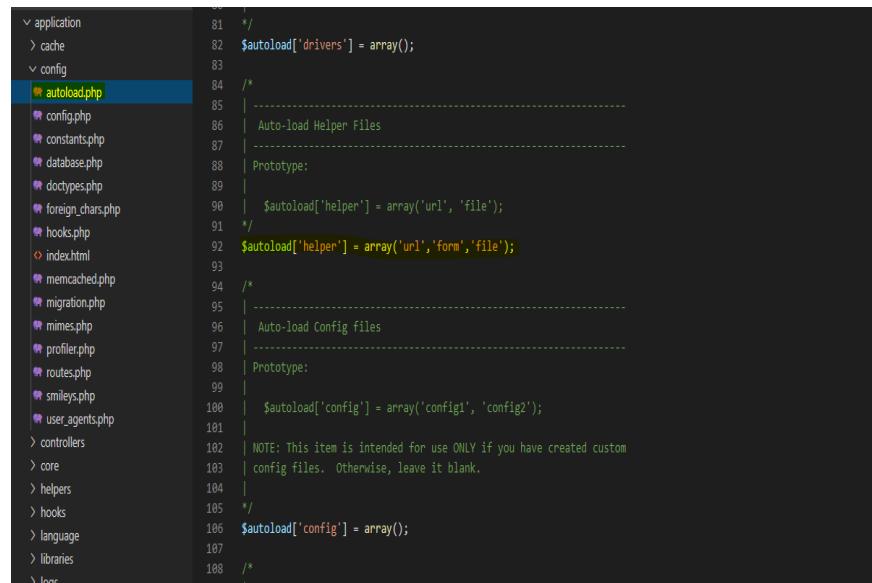
*Helper* adalah fungsi-fungsi Codeigniter yang akan membantu kita dalam berbagai hal, seperti: membuat form, mengakses URL, dll.

Salah satu *helper* yang kita butuhkan dalam pembuatan template adalah helper url.

Helper url berisi fungsi-fungsi untuk mengakses URL seperti base\_url(), site\_url(), \$this->uri, dll.

Cara mengaktifkan helper url:

Pertama buka file config/autoload.php, lalu cari \$autoload['helper']. Tambahkan url, sehingga menjadi seperti ini:



```
81 */
82 $autoload['drivers'] = array();
83 /*
84 | -----
85 | Auto-load Helper Files
86 | -----
87 |
88 | Prototype:
89 |
90 | $autoload['helper'] = array('url', 'file');
91 */
92 $autoload['helper'] = array('url', 'form', 'file');
93 /*
94 | -----
95 | Auto-load Config files
96 | -----
97 |
98 | Prototype:
99 |
100 | $autoload['config'] = array('config1', 'config2');
101 |
102 | NOTE: This item is intended for use ONLY if you have created custom
103 | config files. Otherwise, leave it blank.
104 |
105 */
106 $autoload['config'] = array();
107 /*
108 | -----
```

- Membuat konstanta, buat konstanta SITE\_NAME untuk menyimpan nama web. Konstanta ini nanti kita perlukan untuk mengambil judul web pada template.

Silahkan buka file config/constants.php, lalu tambahkan kode berikut :

The screenshot shows a code editor with two tabs open: 'config.php' and 'constants.php'. The left sidebar lists several files under 'SIMPRO\_TES' and 'application'. The 'constants.php' file is currently selected. The code in 'constants.php' defines various exit codes and their meanings.

```
application > config > constants.php

65 | are as follows:
66 |
67 | Standard C/C++ Library (stdlib.h):
68 | http://www.gnu.org/software/libc/manual/html\_node/Exit\_Status.html
69 | (This link also contains other GNU-specific conventions)
70 | BSD sysexists.h:
71 | http://www.gsp.com/cgi-bin/man.cgi?section=3&topic=sysexists
72 | Bash scripting:
73 | http://tldp.org/LDP/abs/html/exitcodes.html
74 |
75 */
76 defined('EXIT_SUCCESS') OR define('EXIT_SUCCESS', 0); // no errors
77 defined('EXIT_ERROR') OR define('EXIT_ERROR', 1); // generic error
78 defined('EXIT_CONFIG') OR define('EXIT_CONFIG', 3); // configuration error
79 defined('EXIT_UNKNOWN_FILE') OR define('EXIT_UNKNOWN_FILE', 4); // file not found
80 defined('EXIT_UNKNOWN_CLASS') OR define('EXIT_UNKNOWN_CLASS', 5); // unknown class
81 defined('EXIT_UNKNOWN_METHOD') OR define('EXIT_UNKNOWN_METHOD', 6); // unknown class member
82 defined('EXIT_USER_INPUT') OR define('EXIT_USER_INPUT', 7); // invalid user input
83 defined('EXIT_DATABASE') OR define('EXIT_DATABASE', 8); // database error
84 defined('EXIT_AUTO_MIN') OR define('EXIT_AUTO_MIN', 9); // lowest automatically-assigned error code
85 defined('EXIT_AUTO_MAX') OR define('EXIT_AUTO_MAX', 125); // highest automatically-assigned error code
86 define('SITE_NAME', 'simpro_tes');
```

- Menambahkan file SB Admin ke dalam proyek Codeigniter.

*Copy/paste* file dan folder yang dibutuhkan. Berikut ini folder yang harus kita *copy*:

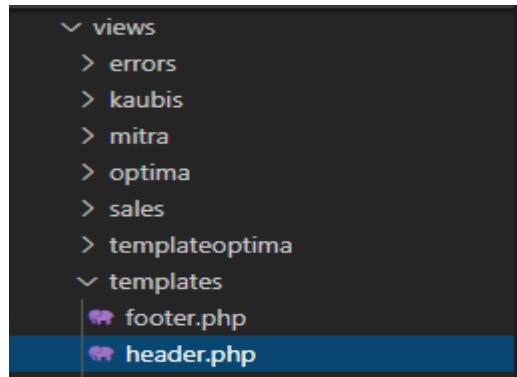
1. Folder css
  2. Folder js
  3. Folder vendor rename menjadi -> assets

css	1/31/2020 11:11 PM	File folder
dist	1/31/2020 11:11 PM	File folder
js	1/31/2020 11:11 PM	File folder
scripts	1/31/2020 11:11 PM	File folder
scss	1/31/2020 11:11 PM	File folder
src	1/19/2020 4:38 AM	File folder
vendor	1/31/2020 11:12 PM	File folder

Dirubah menjadi :

assets	12/18/2019 6:28 PM	File folder
css	12/18/2019 6:29 PM	File folder
js	12/18/2019 6:29 PM	File folder

- Buka Index.php dalam folder SB admin ke dalam Visual Studio Code.
- Buat folder dengan nama templates dan buat file dengan nama header.php didalam folder view codeigniter.



File header.php nantinya akan diisi dengan bootstrap untuk bagian atas tampilan aplikasi.

- Selanjutnya copy bagian header pada index.php didalam folder SB admin ke dalam header.php

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>Mitra - Dashboard</title>

    <!-- Bootstrap core CSS-->
    <link href=<?php echo base_url('asset/bootstrap/css/bootstrap.min.css') ?>" rel="stylesheet">

    <!-- Custom fonts for this template-->
    <link href=<?php echo base_url('asset/fontawesome-free/css/all.min.css') ?>" rel="stylesheet" type="text/css">

    <!-- Page level plugin CSS-->
    <link href=<?php echo base_url('asset/datatables/dataTables.bootstrap4.css') ?>" rel="stylesheet">

    <!-- Custom styles for this template-->
    <link href=<?php echo base_url('css/sb-admin.css') ?>" rel="stylesheet">
```

Header disini berfungsi untuk desain bagian atas dari aplikasi.

```
<!-- Bootstrap core CSS-->
<link href=<?php echo base_url('asset/bootstrap/css/bootstrap.min.css') ?>" rel="stylesheet">

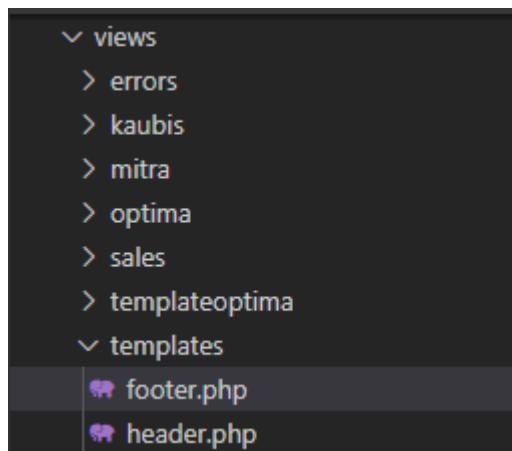
<!-- Custom fonts for this template-->
<link href=<?php echo base_url('asset/fontawesome-free/css/all.min.css') ?>" rel="stylesheet" type="text/css">

<!-- Page level plugin CSS-->
<link href=<?php echo base_url('asset/datatables/dataTables.bootstrap4.css') ?>" rel="stylesheet">

<!-- Custom styles for this template-->
<link href=<?php echo base_url('css/sb-admin.css') ?>" rel="stylesheet">
```

Fungsi dari kodingan di atas adalah untuk memanggil bootstrap dari sebagai desain bagian atas aplikasi.

- Buat file dengan nama footer.php didalam folder view codeigniter.



File footer.php nantinya akan diisi dengan bootstrap untuk bagian bawah tampilan aplikasi.

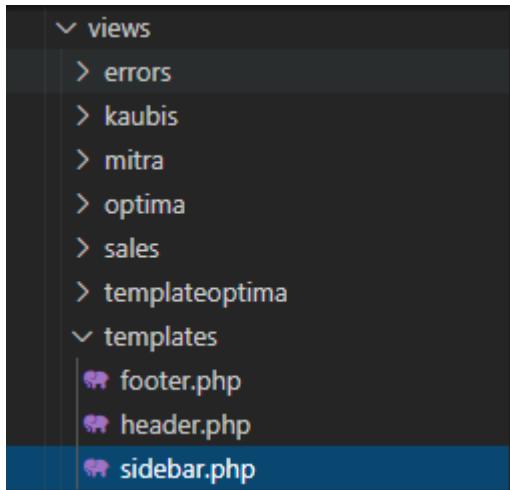
- Selanjutnya copy bagian header pada index.php didalam folder SB admin ke dalam footer.php

```
<!-- Bootstrap core JavaScript-->
<script src=<?php echo base_url('asset/jquery/jquery.min.js') ?>"</script>
<script src=<?php echo base_url('asset/bootstrap/js/bootstrap.bundle.min.js') ?>"></script>

<!-- Core plugin JavaScript-->
<script src=<?php echo base_url('asset/jquery-easing/jquery.easing.min.js') ?>"></script>
<!-- Page level plugin JavaScript-->
<script src=<?php echo base_url('asset/chart.js/Chart.min.js') ?>"></script>
<script src=<?php echo base_url('asset/datatables/jquery.dataTables.js') ?>"></script>
<script src=<?php echo base_url('asset/datatables/dataTables.bootstrap4.js') ?>"></script>
<!-- Custom scripts for all pages-->
<script src=<?php echo base_url('js/sb-admin.min.js') ?>"></script>
<!-- Demo scripts for this page-->
<script src=<?php echo base_url('js/demo/datatables-demo.js') ?>"></script>
<script src=<?php echo base_url('js/demo/chart-area-demo.js') ?>"></script>
</body>
</body>

</html>
```

- Buat file dengan nama sidebar.php didalam folder view codeigniter.



File sidebar.php nantinya akan diisi dengan bootstrap untuk bagian samping kiri tampilan aplikasi.

- Selanjutnya copy bagian header pada index.php didalam folder SB admin ke dalam sidebar.php

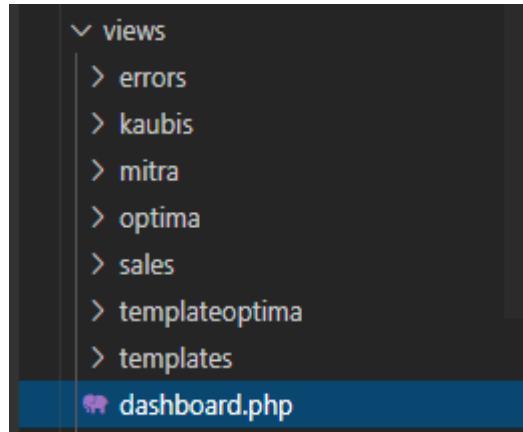
```

1  <!-- Sidebar -->
2  <ul class="sidebar navbar-nav">
3      <li class="nav-item active">
4          <a class="nav-link" href="#"><?php echo base_url().'index.php/home2/index' ;?><?php echo base_url().'index.php/permintaan/index' ;?><?php echo base_url().'index.php/sdi/index' ;?>Survey Desain Inventory <br>(SDI)
22             <a class="dropdown-item" href="#"><?php echo base_url().'index.php/ta/index' ;?>Pembangunan Konstruksi</a>
23         </div>
24     </li>
25     <li class="nav-item">
26         <a class="nav-link" href="#"><?php echo base_url().'index.php/nonta/index' ;?>
27             <i class="fas fa-fw fa-folder"></i>
28             | <span>Mitra Kerjasama</span></a>
29         </li>
30     </ul>
31     <div id="content-wrapper">
32         <div class="container-fluid">

```

Fungsi dari kodingan sidebar.php ini adalah untuk mengisi bagian menu aplikasi.

- Buat View dengan nama file dashboard.php



Masukkan codingan berikut kedalam dashboard.php

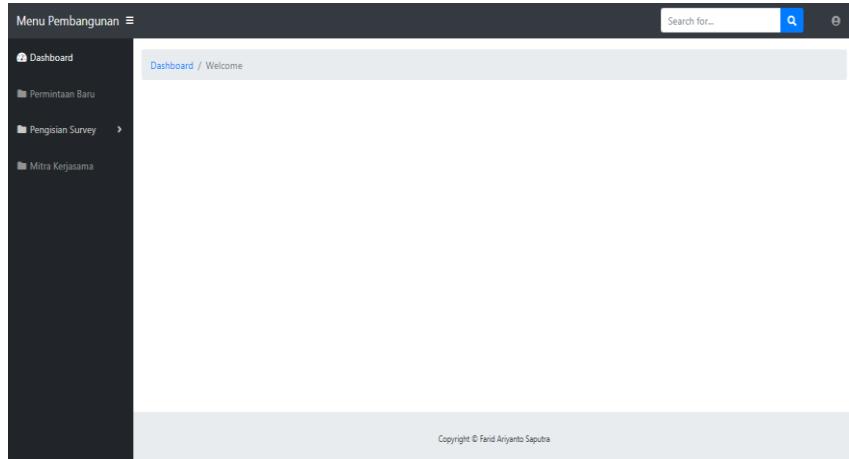
```
application > views > dashboard.php
1  <!-- Breadcrumbs-->
2  <ol class="breadcrumb">
3  | <li class="breadcrumb-item">
4  | | <a href="#">Dashboard</a>
5  | </li>
6  | <li class="breadcrumb-item active">Welcome</li>
7  </ol>
```

- Buat Controller dengan nama file home.php

```
application > controllers > home.php
1  <?php
2  defined('BASEPATH') OR exit('No direct script access allowed');
3
4  class home extends CI_Controller {
5
6      public function index()
7      {
8          $this->load->view('templates/header');
9          $this->load->view('templates/sidebar');
10         $this->load->view('dashboard');
11         $this->load->view('templates/footer');
12     }
13 }
```

File home.php ini berfungsi untuk memanggil template header, footer, dan sidebar yang telah kita buat tadi

- Setelah itu, coba buka kembali [http://localhost/simpro\\_tes/index.php/home/](http://localhost/simpro_tes/index.php/home/), maka kita akan mendapatkan tampilan seperti ini:



### **6.3 Pembuatan Sistem Informasi menggunakan Codeigniter dengan notifikasi Telegram**

Kita telah membuat dashboard dan kerangka pada aplikasi ini selanjutnya adalah kita membuat isi dari kerangka dari aplikasi tsb.

Kita akan membuat sistem informasi tentang pembangunan jaringan pada suatu daerah, ada beberapa hal yang harus di buat dalam aplikasi ini

#### 1. Permintaan Baru

Permintaan baru ini adalah pengguna dari aplikasi ini akan menerima permintaan pembangunan disuatu daerah/lokasi.

#### 2. Pengisian Survey

Di dalam pengisian survey terdapat 2 sub menu yaitu

- Survey Desain
- Pembangunan Kontruksi

#### 3. Mitra Kerjasama

Mitra kerjasama adalah perusahaan yang bekerjasama dalam pembangunan yang akan dilakukan.

Berikut langkah-langkah dalam pembuatan aplikasi/sistem informasi :

- **Permintaan Baru**

Tambahkan codingan berikut pada view/sidebar.php

```
<li class="nav-item">
<a class="nav-link" href="=php echo base_url().'index.php/permintaan/index' ;?&gt;"&gt;
&lt;i class="fas fa-fw fa-folder"&gt;&lt;/i&gt;
| &lt;span&gt;Permintaan Baru&lt;/span&gt;&lt;/a&gt;
&lt;/li&gt;</pre
```

Di dalam kodingan sidebar.php terdapat sebuah class untuk membuat menu “Permintaan Baru”. Kodingan tersebut nantinya akan berfungsi sebagai tampilan untuk aplikasi. Di dalam kodingan ada fungsi :

```
<?php echo base_url().'index.php/permintaan/index;?>
```

Yang artinya adalah memanggil file controller dengan nama permintaan. Nantinya fungsi tersebut akan memanggil class dari controller “permintaan.php”.

- Selanjutnya buat file didalam view dengan nama permintaan.php

```
permintaan.php
```

```
<div class="content-wrapper">
<section class="content-header">
<h1>
    Dashboard
    <small>/ Optima Survey</small>
</h1>
<ol class="breadcrumb">
    <li><a href="#"><i class="fa fa-dashboard"></i> Home</a></li>
    <li class="active">/Permintaan Baru</li>
</ol>
```

```

</section>
<section class="content">
<table class="table">
<tr>
<th>No</th>
<th>STO</th>
<th>LOKASI</th>
<th>Lanjutkan Survey</th>
</tr>

<?php
$no = 1;
foreach($tbl_optimarequest as $mtr){
?>
<tr>
<td><?php echo $no++ ?></td>
<td><?php echo $mtr->sto?></td>
<td><?php echo $mtr->lokasi?></td>
<td><a class="btn btn-
info" href="<?php echo base_url().'index.php/sdi/index' ;?>">
<i class="fa fa-fa-print"></i>Lanjutkan Survey</a></td>
</tr>
<?php } ?>
</table>
</section>
</div>

```

```
<div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title" id="exampleModalLabel"></h4>
      </div>
      <div class="modal-body">
        <div class="form-group">
          <label>STO</label>
          <input type="text" name="sto" class="form-control">
          <label>LOKASI</label>
          <input type="text" name="lokasi" class="form-control">
        </div>
      </div>
      <div class="modal-footer">
        <button type="Reset" class="btn btn-info" data-dismiss="modal">Cancel</button>
        <button type="Submit" class="btn btn-primary">Simpan</button>
      </div>
      <?php echo form_close();?>
    </div>
  </div>
</div>
```

,

- Selanjutnya buat file di dalam controller dengan nama permintaan.php

```

pertmintaan.php

<?php
defined('BASEPATH') OR exit('No direct script access allowed');
);

class permintaan extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        $this->load->model('m_optimarequest');
        $this->load->helper('url');
    }

    public function index()
    {
        $data['tbl_optimarequest'] = $this->m_optimarequest-
>tampil_data()->result();
        $this->load->view('templates/header');
        $this->load->view('templates/sidebar');
        $this->load->view('permintaan', $data);
        $this->load->view('templates/footer');
    }

    public function tambah_aksi()
    {
        $sto    = $this->input->post('sto');
        $lokasi = $this->input->post('lokasi');
    }
}

```

```

$data = array(
    'sto'    => $sto,
    'lokasi' => $lokasi,
);

$this->m_optimarequest-
>input_data($data,'tbl_optimarequest');

redirect('optimarequest');

}
}

```

- Jalankan pada browser maka tampilannya akan jadi seperti berikut:

No	STO	LOKASI	Lanjutkan Survey
1	sarimanah	sarimanah	<a href="#">Lanjutkan Survey</a>
2	sarimanah	sarjadi blok 12	<a href="#">Lanjutkan Survey</a>

- **Pengisian survey**

Pertama kita akan membuat Survey Desain Inventory

Tambahkan pada sidebar.php codingan berikut :

```

<li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle" href="#" id="pagesDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
        <i class="fas fa-fw fa-folder"></i>
        <span>Pengisian Survey</span>
    </a>
    <div class="dropdown-menu" aria-labelledby="pagesDropdown">
        <h6 class="dropdown-header">Pilih Survey:</h6>
        <a class="dropdown-item" href="=php echo base_url().'index.php/sdi/index' ;?&gt;"Survey Desain Inventory <br>(SDI)
        <a class="dropdown-item" href="=php echo base_url().'index.php/ta/index' ;?&gt;"Pembangunan Konstruksi</a>
    </div>

```

Buat file di dalam view dengan nama sdi.php

```
sdi.php

<div class="content-wrapper">
    <section class="content-header">
        <h1>
            Dashboard
            <small>/ SDI Survey</small>
        </h1>
        <ol class="breadcrumb">
            <li><a href="#"><i class="fa fa-dashboar...
                dashboard"></i>Home</a></li>
            <li class="active">/Data Survey</li>
        </ol>
    </section>
    <section class="content">
        <button class="btn btn-primary" data-toggle="modal" data-target="#exampleModal">
            <i class="fa fa-plus"></i> Tambah Survey SDI</button>
        <a class="btn btn-warning" href="<?php echo base_url('Sdi/pdf')?>">
            <i class="fa fa-file"></i>Download PDF</a>
        <table class="table">
            <thead>
                <tr>
                    <th>No</th>
                    <th>Pekerjaan</th>
                    <th>STO</th>
                    <th>Lokasi</th>
                    <th>Designator</th>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td>1</td>
                    <td>Pekerjaan</td>
                    <td>STO</td>
                    <td>Lokasi</td>
                    <td>Designator</td>
                </tr>
            </tbody>
        </table>
    </section>
</div>
```

```

<th>Uraian Pekerjaan</th>
<th>Satuan</th>
<th>status</th>
<th>Detail Data</th>
<th>Hapus</th>
<th>Kirim Pesan</th>

</tr>
<?php
$no = 1;
foreach($tbl_sdi as $mtr){
?>
<tr>
<td><?php echo $no++ ?></td>
<td><?php echo $mtr->pekerjaan?></td>
<td><?php echo $mtr->sto?></td>
<td><?php echo $mtr->lokasi?></td>
<td><?php echo $mtr->desigmator?></td>
<td><?php echo $mtr->uraian_pekerjaan?></td>
<td><?php echo $mtr->satuan?></td>
<td><?php echo $mtr->status?></td>
<td><?php echo anchor('sdi/detail/'.$mtr-
>id_boq,'Tampilkan'); ?></td>

<td><?php echo anchor('sdi/hapus/'.$mtr-
>id_boq,'Hapus',[ 'class'=>'btn btn-danger']); ?></td>
<td><a class="btn btn-
info" href=<?php echo base_url().'index.php/indextelegram';?>
" ><i class="fa fa-fa-print"></i> Kirim Pesan</a></td>

```

```

        </tr>
    <?php } ?>
</table>
</section>
</div>
<!-- Modal -->
<div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
<div class="modal-dialog" role="document">
<div class="modal-content">
<div class="modal-header">
<h4 class="modal-
title" id="exampleModalLabel">Form Input Hasil Survey</h4>
<button type="button" class="close" data-
dismiss="modal" aria-label="Close">
<span aria-hidden="true">&times;</span>
</button>
</div>
<div class="modal-body">
<form method="post" action ="<?php echo base_url().'index
.php/sdi/tambah_aksi';?>">
<div class="form-group">
<label>Pekerjaan</label>
<input type="text" name="pekerjaan" class="form-
control">
<label>Sto</label>
<input type="text" name="sto" class="form-control">

```

```
<label>Lokasi</label>
<input type="text" name="lokasi" class="form-control">
<label>Desigmator</label>
<select class="form-
control" name="desigmator" id="id_boq">
<option>-- Pilih Designator --</option>
<option>DC-OF-SM-48D</option>
<option>DC-OF-SM-288D</option>
<option>AC-OF-SM-12-SC</option>
<option>AC-OF-SM-24-SC</option>
<option>SC-OF-SM-288</option>
<option>OS-SM-1 Eks</option>
<option>OS-SM-12</option>
<option>OS-SM-24</option>
<option>OS-SM-48</option>
<option>ODC-C-288 Splitter</option>
<option>ODP-CA-8</option>
<option>PS-1-4-ODC</option>
<option>PS-1-8-ODP</option>
<option>5 PU-S7.0-140</option>
<option>GB-G3</option>
<option>TC-XX-ODC</option>
<option>SC-OF-SM-288</option>
<option>DD-BM-HDPE-1</option>
<option>DD-HDPE-1</option>
<option>BC-TR-5</option>
<option>BD-SK</option>
<option>HH-PIT-Portable</option>
```

```
</select>
<label>Uraian Pekerjaan</label>
<select class="form-control" name="uraian_pekerjaan" id="id_boq">
    <option>--Pilih Uraian--</option>
    <option>Pengadaan dan pemasangan Kabel Duct Fiber Optic Single Mode kap.48 Core G.652D </option>
    <option>Pengadaan dan pemasangan Kabel Duct Fiber Optic Single Mode kap.288 Core G.652D </option>
    <option>Pengadaan dan pemasangan kabel udara FO Aerial kap.12 Core, Single Mode G.652</option>
    <option>Pengadaan dan pemasangan kabel udara FO Aerial kap.24 Core, Single Mode G.652</option>
    <option>Idem 12 – 288 core </option>
    <option>Penyambungan Kabel Optik Single Mode kap 1 core dengan cara fusion splice </option>
    <option>Idem 12 Core</option>
    <option>Idem 24 Core</option>
    <option>Idem 48 Core</option>
    <option>Pengadaan dan pemasangan Drop Cable FO atas tanah /aerial 1 Core Single Mode, G.65</option>
    <option>Pengadaan dan pemasangan kabinet ODC (Outdoor) kap 288 core dengan space untuk spliter modular termasuk material adaptor SC/UPC, pigtail, pondasi berlapis keramik, lantai kerja, patok pengaman (5 buah), berikut pelabelan</option>
    >
```

<option>Pengadaan dan pemasangan ODP type Closure Aerial Kap 8 core berikut space pasive spliter (1:8), adapter SC/UPC,berikut pelabela</option>

<option>Idem, 1 : 4</option>

<option>Idem, 1 : 8</option>

<option>Pengadaan dan Pemasangan Tiang Besi 7 meter, berikut cat & cor pondasi dan assesories – kekuatan tarik 140 kg </option>

<option>Pengadaan dan Pemasangan Grounding 3 titik rod pada ODC dengan maks 1 ohm </option>

<option>Pengadaan dan Pemasangan Rise Pipe untuk pengaman kabel optik ke ODC Pole / titik naik KU diamater 1 panjang 3 meter</option>

<option>Boring manual/mesin untuk HDPE / Micro Duct diameter dalam 33 mm 1 pipa kedalaman 1,5 meter</option>

>

<option>Pengadaan dan Pemasangan 1 pipa HDPE/H DPE BIG untuk kabel FO, Include Soket HDPE </option>

<option>Galian, Pengurukan kembali dan perbaikan kembali, pengisian pasir, deeksten dan tanda rute kabel tanah dan tempat sambung kedalaman 1,5 meter </option>

<option>Pembobokan dan perbaikan Dinding untuk lubang Sparing Kabel</option>

</select>

<label>Satuan</label>

<select class="form-control" name="satuan" id="id\_boq">

<option>meter</option>

```

<option>core</option>
<option>titik</option>
<option>pcs</option>
</select>
<label>Satuan</label>
<select class="form-
control" name="status" id="id_boq">
<option>sedang dikerjakan</option>
</select>
</div>
</div>
<div class="modal-footer">
<button type="Reset" class="btn btn-info" data-
dismiss="modal">Reset</button>
<button type="Submit" class="btn btn-
primary">Simpan</button>
</div>
</div>
</div>
</div>

```

- Selanjutnya buat file didalam controller dengan nama Sdi.php

Sdi.php
<pre> &lt;?php defined('BASEPATH') OR exit('No direct script access allowed'); class sdi extends CI_Controller { </pre>

```

public function __construct()
{
    parent::__construct();
    $this->load->model('m_sdi');
    $this->load->helper('url');
}

public function index()
{
    $data['tbl_sdi'] = $this->m_sdi->tampil_data()->result();
    $dataku['tbl_sdi'] = $this->m_sdi->get();
    $this->load->view('templates/header');
    $this->load->view('templates/sidebar');
    $this->load->view('sdi',$data);
    $this->load->view('templates/footer');

}

public function tambah_aksi()
{
    $pekerjaan      = $this->input->post('pekerjaan');
    $sto           = $this->input->post('sto');
    $lokasi         = $this->input->post('lokasi');
    $desigmator     = $this->input->post('desigmator');
    $uraian_pekerjaan = $this->input-
>post('uraian_pekerjaan');

    $satuan        = $this->input->post('satuan');
    $status         = $this->input->post('status');

    $data = array(

```

```

        'pekerjaan' => $pekerjaan,
        'sto'       => $sto,
        'lokasi'    => $lokasi,
        'desigmator' => $desigmator,
        'uraian_pekerjaan' => $uraian_pekerjaan,
        'satuan'     => $satuan,
        'status'      => $status,
    );
    $this->m_sdi->input_data($data,'tbl_sdi');
    redirect('sdi');
}
public function print($id){
    $data = array(
        'record' => $this->db-
>query("SELECT * FROM tbl_sdi where id_boq='$id'"),
    );
    $this->load->view('print_sdi',$data);
    $html = $this->output->get_output();
    $this->load->library('dompdf_gen');
    $this->dompdf->load_html($html);
    $this->dompdf->render();
    $sekarang=date("d:F:Y:h:m:s");
    $this->dompdf-
>stream("BoQ".$sekarang.".pdf",array('Attachment'=>0));
}
public function detail($id_boq){
    //controller untuk form detail data
    $where = array('id_boq' => $id_boq);
}

```

```

    $data['tbl_sdi']=$this->m_sdi->edit_data($where,'tbl_sdi')-
>result();
    $this->load->view('detail',$data, false);
}

function hapus($id_boq){
    $where = array('id_boq' => $id_boq);
    $this->m_sdi->hapus_data($where,'tbl_sdi');
    redirect('sdi');
}

public function pdf(){
    $this->load->library('dompdf_gen');
    $data['tbl_sdi'] = $this->m_sdi->tampil_data('tbl_sdi')-
>result();
    $this->load->view('laporan_pdf',$data);
    $paper_size = 'A4';
    $orientation = 'landscape';
    $html = $this->output->get_output();
    $this->dompdf->set_paper($paper_size,$orientation);
    $this->dompdf->load_html($html);
    $this->dompdf->render();
    $this->dompdf-
>stream("BoQ.pdf", array('Attachment' =>0));
}
}

```

- Selanjutnya buat file di dalam model dengan nama m\_sdi.php

M_sdi.php
<?php

```
class m_sdi extends CI_model
{
    public function tampil_data()
    {
        return $this->db->get('tbl_sdi');
    }
    function tampil()
    {
        $data = $this->db->get('tbl_sdi');
        return $data->result();
    }
    public function input_data($data,$tabel)
    {
        $this->db->insert($tabel,$data);
    }
    public function get()
    {
        $dataku = $this->db->get('tbl_sdi');
        return $dataku->result();
    }
    function tampil_list()
    {
        $data = $this->db->get('tbl_sdi');
        return $data->result();
    }
    function edit_data($where,$table){

```

```

        return $this->db->get_where($table,$where);

    }

    function hapus_data($where,$table){
        $this->db->where($where);
        $this->db->delete($table);
    }
}

```

Ketika dijalankan pada browser akan keluar tampilan sebagai berikut :

- Kedua kita akan membuat Pembangunan Kontruksi  
Buat file di dalam view dengan nama ta.php

ta.php
<div class="content-wrapper"> <section class="content-header"> <h1> Dashboard <small>/ Mitra Survey</small>

```

</h1>

<ol class="breadcrumb">
    <li><a href="#"><i class="fa fa-
dashboard"></i>Home</a></li>
    <li class="active">/Data Survey</li>
</ol>
</section>

<section class="content">
    <button class="btn btn-primary" data-toggle="modal" data-
target="#exampleModal">
        <i class="fa fa-plus"></i> Upload Hasil Survey</button>
        <a class="btn btn-
warning" href="<?php echo base_url('ta/pdf')?>">
            <i class="fa fa-file"></i> Dowbload PDF</a>
        <table class="table">
            <tr>
                <th>No</th>
                <th>Sto</th>
                <th>Lokasi</th>
                <th>Foto Pemasangan OLT</th>
                <th>Foto Pemasangan FTM/ODF</th>
                <th>Foto Pemasangan ODC</th>
                <th>Foto Pemasangan ODP</th>
                <th>Status</th>
                <th>Detail Data</th>
                <th>Aksi</th>
            </tr>

```

```

<?php
$no = 1;
foreach($tbl_ta as $mtr){
?>
<tr>
<td><?php echo $no++ ?></td>
<td><?php echo $mtr->sto?></td>
<td><?php echo $mtr->lokasi?></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td><?php echo $mtr->status?></td>
<td><?php echo anchor('ta/detailta/'.$mtr-
>id_survey,'Tampilan'); ?></td>
<!-- <td><?php echo anchor('ta/edit/'.$mtr-
>id_survey,'Edit',[ 'class'=>'btn btn-success' ]); ?> -->
<td><?php echo anchor('ta/hapus/'.$mtr-
>id_survey,'Hapus',[ 'class'=>'btn btn-danger' ]); ?></td>

```

```

        </tr>
        <?php } ?>
    </table>
</section>
</div>

<!-- Modal --&gt;
&lt;div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true"&gt;
    &lt;div class="modal-dialog" role="document"&gt;
        &lt;div class="modal-content"&gt;
            &lt;div class="modal-header"&gt;
                &lt;h4 class="modal-
title" id="exampleModalLabel"&gt;Form Input Hasil Survey&lt;/h4&gt;
            &lt;
                &lt;button type="button" class="close" data-dismiss="modal" aria-label="Close"&gt;
                    &lt;span aria-hidden="true"&gt;&amp;times;&lt;/span&gt;
                &lt;/button&gt;
            &lt;/div&gt;
            &lt;div class="modal-body"&gt;
                &lt;?php echo form_open_multipart('ta/tambah_aksi'); ?&gt;

                &lt;div class="form-group"&gt;
                    &lt;label&gt;STO&lt;/label&gt;
                    &lt;input type="text" name="sto" class="form-control"&gt;
                    &lt;label&gt;LOKASI&lt;/label&gt;
</pre>

```

```
<input type="text" name="lokasi" class="form-control">
<label>Upload Foto OLT</label>
<input type="file" name="gambar_olt" class="form-control">
<label>Upload Foto FTM</label>
<input type="file" name="gambar_ftm" class="form-control">
<label>Upload Foto ODC</label>
<input type="file" name="gambar_odc" class="form-control">
<label>Upload Foto ODP</label>
<input type="file" name="gambar_odp" class="form-control">
<label>Satuan</label>
<select class="form-control" name="status" id="id_boq">
<option>sedang dikerjakan</option>
</select>
</div>
</div>
<div class="modal-footer">
<button type="Reset" class="btn btn-info" data-dismiss="modal">Cancel</button>
<button type="Submit" class="btn btn-primary">Simpan</button>
</div>
<?php echo form_close();?>
```

```
</div>
</div>
</div>
```

- Tambahkan file didalam folder controller beri nama ta.php

```
ta.php

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class ta extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        $this->load->model('m_ta');
        $this->load->helper('url');
    }
    public function index()
    {
        $data['tbl_ta'] = $this->m_ta->tampil_data()->result();
        $this->load->view('templates/header');
        $this->load->view('templates/sidebar');
        $this->load->view('ta',$data);
        $this->load->view('templates/footer');
    }
    public function tambah_aksi()
```

```

{
$sto    = $this->input->post('sto');
$lokasi  = $this->input->post('lokasi');
$gambar_olt    = $_FILES['gambar_olt'];
if ($gambar_olt!=""){ }else{
$config['upload_path'] = './asset/images/';
$config['allowed_types'] = 'jpg|png|gif|jpeg';
$this->load->library('upload',$config);
if(!$this->upload->do_upload('gambar_olt')){
echo "upload gagal"; die();
}else{
$gambar_olt=$this->upload->data('file_name');
}
}
$gambar_ftm    = $_FILES['gambar_ftm'];
if ($gambar_ftm!=""){ }else{
$config['upload_path'] = './asset/images/';
$config['allowed_types'] = 'jpg|png|gif|jpeg';
$this->load->library('upload',$config);
if(!$this->upload->do_upload('gambar_ftm')){
echo "upload gagal"; die();
}else{
$gambar_ftm=$this->upload->data('file_name');
}
}
$gambar_odc    = $_FILES['gambar_odc'];
if ($gambar_odc!=""){ }else{
$config['upload_path'] = './asset/images/';

```

```

$config['allowed_types'] = 'jpg|png|gif|jpeg';
$this->load->library('upload',$config);
if(!$this->upload->do_upload('gambar_odc')){
echo "upload gagal"; die();
}else{
$gambar_odc=$this->upload->data('file_name');
}
}

$gambar_otp      = $_FILES['gambar_otp'];
if ($gambar_otp==""){ }else{
$config['upload_path'] = './asset/images/';
$config['allowed_types'] = 'jpg|png|gif|jpeg';
$this->load->library('upload',$config);
if(!$this->upload->do_upload('gambar_otp')){
echo "upload gagal"; die();
}else{
$gambar_otp=$this->upload->data('file_name');
}
}

$status   = $this->input->post('status');
}

}

$data = array(
'sto'    => $sto,
'lokasi'  => $lokasi,
'gambar_olt'    => $gambar_olt,
'gambar_ftm'    => $gambar_ftm,
'gambar_odc'    => $gambar_odc,

```

```

'gambar_odp'      => $gambar_odp,
'status'          =>$status

);

$this->m_ta->input_data($data,'tbl_ta');

redirect('ta');

}

function hapus($id_survey){

$where = array('id_survey' => $id_survey);

$this->m_ta->hapus_data($where,'tbl_ta');

redirect('ta');

}

public function detailta($id_survey){

//controller untuk form detail data

$where = array('id_survey' => $id_survey);

$data['tbl_ta']=$this->m_ta->edit_data($where,'tbl_ta')-

>result();

$this->load->view('detailta',$data, false);

}

public function pdf(){

$this->load->library('dompdf_gen');

$data['tbl_ta'] = $this->m_ta->tampil_data('tbl_ta')->result();

$this->load->view('laporan_pdf_ta',$data);

$paper_size = 'A4';

$orientation = 'landscape';

$html = $this->output->get_output();

$this->dompdf->set_paper($paper_size,$orientation);

```

```
    $this->dompdf->load_html($html);
    $this->dompdf->render();
    $this->dompdf->stream("BoQ.pdf", array('Attachment' =>0));
}
}
```

- Tambahkan file didalam folder models beri nama m\_ta.php

```
m_ta.php

<?php

class m_ta extends CI_model
{

    public function tampil_data()
    {
        return $this->db->get('tbl_ta');
    }

    function tampil()
    {
        $data = $this->db->get('tbl_ta');
        return $data->result();
    }

    public function input_data($data,$tabel)
    {
        $this->db->insert($tabel,$data);
    }

    function hapus_data($where,$table){
```

```

        $this->db->where($where);
        $this->db->delete($table);
    }

    function edit_data($where,$table){
        return $this->db->get_where($table,$where);
    }
}

```

- Jalankan pada browser tampilannya akan jadi seperti berikut

No	Sto	Lokasi	Foto Pemasangan OLT	Foto Pemasangan FTM/ODF	Foto Pemasangan ODC	Foto Pemasangan ODP	Status	Detail Data	Aksi
1	sanjadi	Sanjadi, blok 12 no 131					sedang dikerjakan	<a href="#">Tampilkan</a>	<a href="#">Hapus</a>

- **Mitra Kerjasama**

Buat file baru di dalam folder view bari nama mitra.php

Tambahkan pada sidebar.php codingan berikut :

```

<li class="nav-item">
    <a class="nav-link" href="php echo base_url().'index.php/mitra/index' ;?"&gt;
        &lt;i class="fas fa-fw fa-folder"&gt;&lt;/i&gt;
        &lt;span&gt;Mitra Kerjasama&lt;/span&gt;&lt;/a&gt;
    &lt;/li&gt;
</pre

```

Dalam file mitra.php tambahkan codingan berikut :

```
mitra.php

<div class="content-wrapper">
  <section class="content-header">
    <h1>
      Dashboard
      <small>/ Mitra baru</small>
    </h1>
    <ol class="breadcrumb">
      <li><a href="#"><i class="fa fa-dashboar- dashboard"></i>Home</a></li>
      <li class="active">/Mitra</li>
    </ol>
  </section>

  <section class="content">
    <button class="btn btn-primary" data-toggle="modal" data-target="#exampleModal">
      <i class="fa fa-plus"></i>Tambah Mitra </button>
    <!-- Data -->
    <div class="card mb-3">
      <div class="card-header">
        <i class="fas fa-table"></i>
        Data Simpro </div>
      <div class="card-body">
        <div class="table-responsive">
```

```
<table class="table table-
bordered" id="dataTable" width="100%" cellspacing="20"
>
<thead>
<tr>
<th>NO </th>
<th>Nama</th>
</tr>
</thead>
<?php
foreach($tbl_mitra as $mtr){
?>
<tbody>
<tr>
<td><?php echo $mtr->id_mitra ?></td>
<td><?php echo $mtr->nama_mitra ?></td>
</td>
</td>
</tr>
<?php } ?>
</tbody>
</table>

</div>
</section>
</div>
```

```

<!-- Modal -->
<div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-
title" id="exampleModalLabel">Form Input Mitra Baru</h
4>
        <button type="button" class="close" data-
dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <div class="modal-body">
        <form method="post" action = "<?php echo base_url('i
ndex.php/nonta/tambah_aksi');?>">
          <div class="form-group">
            <label>Nama Mitra</label>
            <input type="text" name="namamitra" class="form-
control">
          </div>
          <div class="modal-footer">
            <button type="Reset" class="btn btn-info" data-
dismiss="modal">Reset</button>
            <button type="Submit" class="btn btn-
primary">Simpan</button>
          </div>
        </form>
      </div>
    </div>
  </div>
</div>

```

```
</div>
</div>
</div>
</div>
```

- Tambahkan controller dengan nama mitra.php

```
mitra.php

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class mitra extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        $this->load-
>model('m_nonta'); // di jadikan constructor agar langsung di
jalankan saat di akses dan bisa di akses ke semua fungction
        $this->load->helper('url');
    }

    public function index()
{
```

```
$data['tbl_mitra'] = $this->m_nonta->tampil_data()->result();

$this->load->view('templates/header');
$this->load->view('templates/sidebar');
$this->load->view('nonta',$data);
$this->load->view('templates/footer');

}

public function tambah_aksi()
{
    $nama_mitra    = $this->input->post('namamitra');

    $data = array(
        'nama_mitra'    => $nama_mitra,
    );
    $this->m_nonta->input_data($data,'tbl_mitra');
    redirect('nonta');

}

public function edit($id){
    $where = array('id' => $id_mitra);
    $data['tbl_mitra'] = $this->m_ta->edit_data($where,'upload')->result();
}

}
```

- Tambahkan models dengan nama m\_mitra.php

```
m_mitra.php

<?php

class m_mitra extends CI_model
{

    public function tampil_data()
    {
        return $this->db->get('tbl_mitra');
    }

    function tampil()
    {
        $data = $this->db->get('tbl_mitra');
        return $data->result();
    }

    public function input_data($data,$tabel)
    {
        $this->db->insert($tabel,$data);
    }
}
```

- Jalankan pada browser maka akan keluar tampilan seperti berikut :

NO	Nama
1	PT. INTI
2	PT. TELKOM AKSES
3	PT.INTI
4	PT.RAHMI ROZA

- **Pembuatan telegram bot**
- Buat file dengan nama telegram.php didalam folder view

```
telegram.php

<!DOCTYPE html>
<html>
<head>
<meta charset='utf-8'>
<title>wadagizig Telegram bots</title>
</head>
<body>

<div class="content">
    <section class="content-header">
        <i class="fa fa-home"></i> Home / <i class="fa fa-dashboard"></i> Dashboard / <b>wadagizig</b>
    </section>
```

```
<div class="col-md-4">
    <div class="box box-solid box-primary">
        <div class="box-header">
            <h4 class="box-
title"><b>Telegram Message <i class="fa fa-
send" style="color: #007bff;"></i></b></h4>
        </div>

        <div class="box-body">
            <form method="post" action="<?php echo base_url() . 'index.php/telegramoptima/kirim' ;?>">
                <div class="form-group row">
                    <label class="col-md-4 col-form-
label">Telegram ID</label>
                    <div class="col-md-8">
                        <input type="text" class="form-
control" name="telegram_id" placeholder="Telegram ID" va-
lue="805296561">
                </div>

                </div>
            </div>
            <div class="form-group row">
                <label class="col-md-4 col-form-
label">Messages</label>
                <div class="col-md-8">
```

```

<input type="text" class="form-control" name="message_text" placeholder="Custom Text Message">
</div>
</div>

<button type="submit" class="btn btn-primary pull-right">Send <i class="fa fa-send"></i></button>

</form>
</div>
</div>
</body>
</html>

```

- Buat file dengan nama SendMessage.php di folder view

SendMessage.php
<pre> &lt;?php  function sendMessage(\$telegram_id, \$message_text, \$secret_token) {     \$url = "https://api.telegram.org/bot" . \$secret_token . "/sendMessage?parse_mode=markdown&amp;chat_id=" . \$telegram_id;     \$url = \$url . "&amp;text=" . urlencode(\$message_text);     \$ch = curl_init();     \$optArray = array( </pre>

```

CURLOPT_URL => $url,
CURLOPT_RETURNTRANSFER => true
);

curl_setopt_array($ch, $optArray);
$result = curl_exec($ch);
curl_close($ch);

}

/*
-----only basic POST method :-----*/
$telegram_id = $_POST ['telegram_id'];
$message_text = $_POST ['message_text'];
/*
-----Isi TOKEN dibawah ini:-----*/
$secret_token = "1013133076:AAGzzJt3FDx7M89AbgB
Ap8fXTVcToWzx5uY";
sendMessage($telegram_id, $message_text, $secret_token)
;
echo "<script>alert('Pesan berhasil terkirim!'); window.location.href = './';</script>";

```

Buat controller dengan nama telegram.php

telegram.php
<pre> &lt;?php defined('BASEPATH') OR exit('No direct script access allowed'); </pre>

```
class telegramoptima extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        $this->load->model('m_telegram');
        $this->load->helper('url');
    }

    public function index()
    {

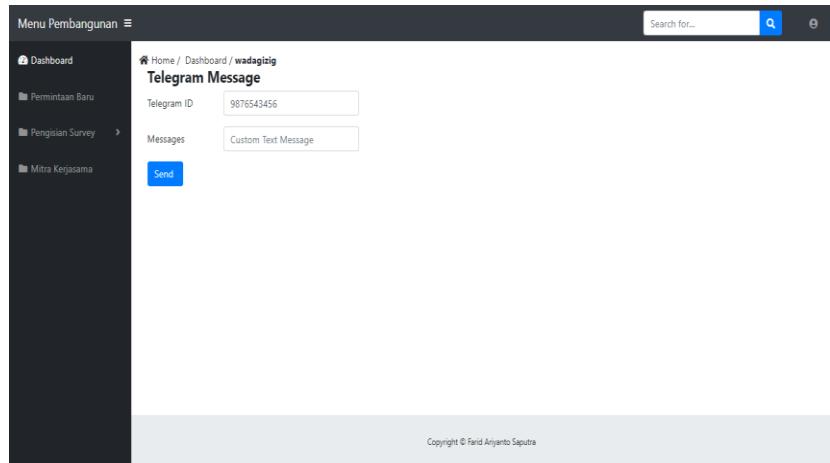
        $this->load->view('templateoptima/header');
        $this->load->view('templateoptima/sidebar');
        $this->load->view('telegramoptima');
        $this->load->view('templateoptima/footer');

    }

    public function kirim()
    {
        $this->load->view('templateoptima/header');
        $this->load->view('templateoptima/sidebar');
        $this->load->view('sendMessaageOptima');
        $this->load->view('templateoptima/footer');

    }
}
```

- Jalankan pada browser maka tampilannya akan jadi seperti berikut :



- **Pembuatan Login**
- Buat file di dalam folder view dengan nama login.php

```
login.php

<!DOCTYPE html>
<html lang="en">

<head>

    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge" >
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" >
    <meta name="description" content="">
    <meta name="author" content="">

    <title>LOGIN</title>

    <!-- Custom fonts for this template-->
    <link href="<?php echo base_url('assets/vendor/fontawesome-free/css/all.min.css') ?>" rel="stylesheet" type="text/css">
```

```

<!-- Page level plugin CSS-->
<link href=<?php echo base_url('assets/vendor/dataTables/dataTables.bootstrap4.css') ?>" rel="stylesheet">

<!-- Custom styles for this template-->
<link href=<?php echo base_url('assets/css/admin.css') ?>" rel="stylesheet">

</head>

<body class="bg-light">

    <div class="container">
        <div class="card card-login mx-auto mt-5">
            <div class="card-header" style="text-align: center">LOGIN SIMPRO</div>
            <center> <img src=<?php echo base_url(); ?> asset/images/Capture1.PNG" width="150" height="150" ></center>
            <div class="card-body">
                <form action=<?= base_url('login/login') ?>" method="POST">
                    <div class="form-group">
                        <div class="form-label-group">
                            <input type="text" name="username" class="form-control" placeholder="Username" required="required" autofocus="autofocus">
                            <label for="username">Username</label>
                        </div>
                    </div>
                    <div class="form-group">
                        <div class="form-label-group">
                            <input type="password" name="password" class="form-control" placeholder="Password" required="required">
                        </div>
                    </div>
                </form>
            </div>
        </div>
    </div>

```

```
        <label for="inputPassword">Password</
label>
            </div>
        </div>
        <div class="form-group">

            </div>
            <button class="btn btn-primary btn-
block" name="submit">Login</button>
        </form>
        <div class="text-center">

            </div>
        </div>
    </div>
</div>

<!-- Bootstrap core JavaScript--&gt;
&lt;script src=&lt;?php echo base_url('assets/vendor/j
query/jquery.min.js')?&gt;"&gt;&lt;/script&gt;
&lt;script src=&lt;?php echo base_url('assets/vendor/b
ootstrap/js/bootstrap.bundle.min.js')?&gt;"&gt;&lt;/script&gt;

<!-- Core plugin JavaScript--&gt;
&lt;script src=&lt;?php echo base_url('assets/vendor/j
query-easing/jquery.easing.min.js')?&gt;"&gt;&lt;/script&gt;

<!-- Page level plugin JavaScript--&gt;
&lt;script src=&lt;?php echo base_url('assets/vendor/c
hart.js/Chart.min.js')?&gt;"&gt;&lt;/script&gt;
&lt;script src=&lt;?php echo base_url('assets/vendor/d
atables/dataTables.jquery.js')?&gt;"&gt;&lt;/script&gt;
&lt;script src=&lt;?php echo base_url('assets/vendor/d
atables/dataTables.bootstrap4.js')?&gt;"&gt;&lt;/script&gt;

<!-- Custom scripts for all pages--&gt;
&lt;script src=&lt;?php echo base_url('assets/js/sb-
admin.min.js')?&gt;"&gt;&lt;/script&gt;

<!-- Demo scripts for this page--&gt;</pre>
```

```

<script src=<?php echo base_url('assets/js/demo/
datatables-demo.js')?>></script>
<script src=<?php echo base_url('assets/js/demo/
chart-area-demo.js')?>></script>

</body>

</html>

```

- Buat file baru di dalam controller beri nama login.php

login.php

```

<?php

class Login extends CI_Controller{
    function __construct(){
        parent::__construct();
        $this->load->model('M_login');
    }
    function index(){
        $this->load->view('login');
    }

    function login()
    {
        if (isset($_POST['submit'])) {

            $username = $this->input-
>post('username',true);
            $password = $this->input-
>post('password',true);

            $akun = $this->M_login-
>cek_akun($username,$password);

            if($akun == 1) {
                $this->session-
>userdata('username');
                $this->load->view('sales/header');
                $this->load->view('sales/home');
            }
        }
    }
}

```

```
        $this->load->view('sales/footer');

    } else if ($akun == 3) {
        $this->session-
>userdata('username');
        $this->load->view('kaubis/footer');
        $this->load->view('kaubis/header');
        $this->load->view('kaubis/home');

    } else if ($akun == 2) {
        $this->session-
>userdata('username');
        $this->load->view('optima/footer');
        $this->load->view('optima/header');
        $this->load->view('optima/home');
    }else if ($akun == 4) {
        $this->session-
>userdata('username');
        $this->load-
>view('templates/header');
        $this->load-
>view('templates/sidebar');
        $this->load->view('dashboard');
        $this->load-
>view('templates/footer');
    }
    else if ($akun == 5) {
        $this->session-
>userdata('username');
        $this->load-
>view('templateoptima/header');
        $this->load-
>view('templateoptima/sidebar');
        $this->load-
>view('dashboardoptima');
        $this->load-
>view('templateoptima/footer');
    }
    else
    {
```

```
        echo "<script>alert('Username & Pas  
sword Salah')</script>";
        $this->load->view('login');
    }
}
else
{
    echo "<script>alert('BATAL')</script>";
    $this->load->view('login');
}

}

function logout()
{
    $username = $this->session-
>userdata('status_login');
    $this->M_login-
>logout($username);
    $this->session-
>set_flashdata('pesan', '<div class="alert alert-
success" role="alert"><button type="button" class="
close sucess-op" data-dismiss="alert" aria-
label="Close">
        <span class="icon-sc-
cl" aria-
hidden="true">x</span></button> Logouot Berhasil, Te
rima Kasih </div>');
    redirect(base_url()."Login/index");
}
}
```

- Selanjutnya buat file baru di dalam folder model beri nama M\_login.php

```

M_login.php

<?php

class M_login extends CI_Model
{

    function cek_akun($username,$password)
    {
        $query = $this->db-
>get_where('tbl_login',array('username'=>$username,
'password'=>$password));

        foreach ($query->result_array() as $qy )
        {
            $this->session-
>set_userdata('username', $qy['username']);
            $this->session-
>set_userdata('level', $qy['level']);
        }

        $row = $query->row_array();
        if ($query->num_rows()>0)
        {
            date_default_timezone_set('Asia/Jakarta
');

            $date    =    date("Y/m/d H:i:s");
            $data    =    array('tgl'=>$date, 'status
'=>1);

            $this->db-
>where('username', $username);
            $this->db->update('tbl_login',$data);
            return $row['level'];
        } else {
            return 0;
        }
    }
}

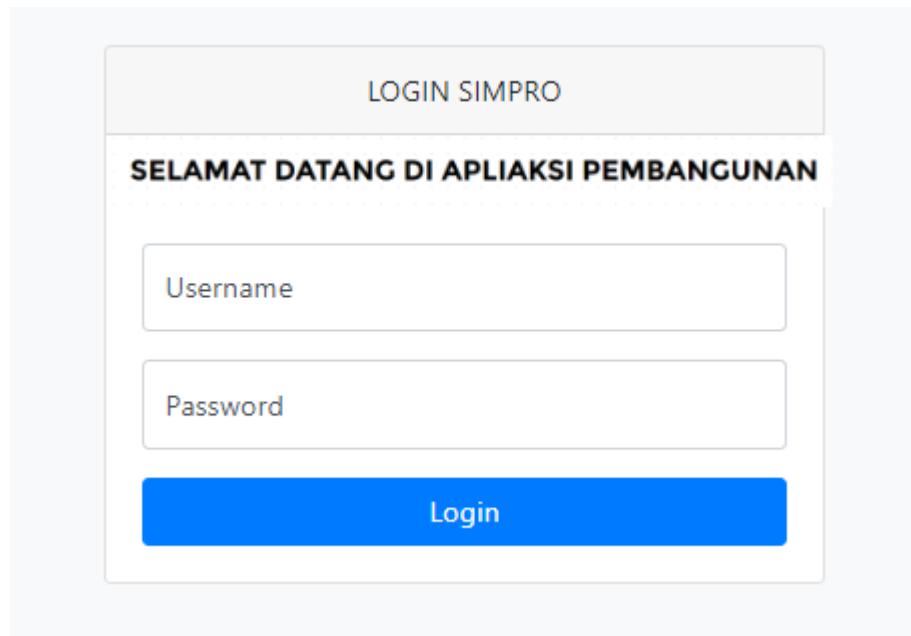
```

```
        }

    }

    function logout($username)
    {
        $this->db->set(status, 0);
        $this->db->where(username, $username);
        $this->db->update(tbl_login);
    }
}
```

- Jalankan pada browser

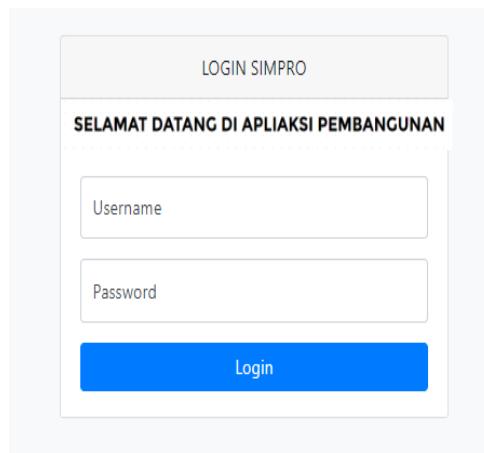


## BAB VII

# TAMPILAN ANTARMUKA

### 7. TAMPILAN ANTARMUKA

#### 7.1 Login



Gambar 7.1 login

Petunjuk pemakaian :

1. Masukkan username sesuai database yang telah dibuat.
2. Masukkan password sesuai database yang telah dibuat.

#### 7.2 Halaman Menu Utama



Gambar 7.2 Halaman Menu Utama

Petunjuk Pemakaian :

1. Lakukan login.
2. Untuk menu utama terdapat 3 menu, permintaan baru, pengisian survey, mitra kerjasama.

### 7.3 Halaman Permintaan Baru

No	STO	LOKASI	Lanjutkan Survey
1	sarimanah	sarimanah	<a href="#">Lanjutkan Survey</a>
2	sarimanah	sarijadi blok 12	<a href="#">Lanjutkan Survey</a>

Gambar 7.3 Halaman permintaan Baru

Petunjuk pemakaian :

1. Lakukan login
2. Pilih menu Permintaan Baru
3. Pilih lanjukan Survey untuk melakukan suervey

### 7.4 Halaman Pengisian Survey

Di dalam Halaman pengisian survey terdapat 2 sub menu yaitu survey desain inventory dan pembangunan kontruksi.

SELAMAT DATANG DI APLIKASI PEMBANGUNAN

Dashboard / Welcome

Menu Pembangunan

pilih Survey:  
Survey Desain Inventory (SDI)  
pembangunan Konstruksi

Mitra Kerjasama

Gambar 7.4 Halaman pengisian survey

Petunjuk pemakaian :

1. Lakukan Login
2. Pilih menu yang akan digunakan

#### 7.4.1 Halaman Survey desain inventory

The screenshot shows a web-based application interface titled 'Dashboard / SDI Survey'. On the left, there is a sidebar with navigation links: 'Dashboard', 'Permintaan Baru', 'Pengisian Survey' (which is currently selected), and 'Mitra Kerjasama'. The main content area has a search bar at the top right. Below the search bar, there are two buttons: '+ Tambah Survey SDI' (in blue) and 'Download PDF' (in yellow). The main area displays a table of survey data:

No	Pekerjaan	STO	Lokasi	Designator	Uraian Pekerjaan	Satuan	status	Detail Data	Hapus	Kirim Pesan
1	pasang	sanjadi	sanjadi	SC-OF-SM-288	Pengadaan dan Pemasangan Tiang Besi 7 meter, berikut cat & cor pondasi dan assesories – kekuatan tarik 140 kg	core	sedang dikerjakan	<a href="#">Tampilkan</a>	<button>Hapus</button>	<button>Kirim Pesan</button>

At the bottom of the page, there is a copyright notice: 'Copyright © Farid Arifianto Saputra'.

Gambar 7.4.1 Halaman Survey desain inventory

Petunjuk pemakaian :

1. Lakukan login
2. Pilih menu survey desain inventory
3. Klik button tambah survey SDI untuk menambah survey
4. Untuk melihat data klik detail data
5. Untuk menghapus survey yang telah dibuat klik button hapus
6. Setelah selesai menambahkan survey baru klik button kirim pesan untuk memberitahukan bahwa survey telah selesai dilakukan
7. Untuk mendownload hasil survey klik button Download PDF

#### 7.4.2 Halaman Pembangunan Kontruksi

The screenshot shows a dashboard titled 'Dashboard / Mitra Survey'. On the left is a sidebar with menu items: Dashboard, Permintaan Baru, Pengisian Survey, and Mitra Kerjasama. The main area has a search bar at the top right. Below it, there are two buttons: '+Upload Hasil Survey' and 'Download PDF'. A table lists survey entries. The first entry has the following details:

No	Sto	Lokasi	Foto Pemasangan OLТ	Foto Pemasangan FTM/ODF	Foto Pemasangan ODC	Foto Pemasangan ODP	Status	Detail Data	Aksi
1	sanjadi	Sanjadi, blok 12 no 131					sedang dikerjakan	<a href="#">Tampilkan</a>	<a href="#">Hapus</a>

At the bottom right of the main area, it says 'Copyright © Farid Ariyanto Saputra'.

Gambar 7.4.2 Halaman Pembagunan Kontruksi

Petunjuk pemakaian :

1. Lakukan login
2. Pilih menu Pembangunan Kontruksi
3. Klik button tambah Hasil Survey untuk menambah survey
4. Untuk melihat data klik detail data
5. Untuk menghapus survey yang telah dibuat klik button hapus
6. Untuk mendownload hasil survey klik button Download PDF

#### 7.5 Halaman Mitra Kerjasama

The screenshot shows a dashboard titled 'Dashboard / Mitra baru'. The sidebar includes 'Dashboard', 'Permintaan Baru', 'Pengisian Survey', and 'Mitra Kerjasama'. The main area features a search bar at the top right and a button '+Tambah Mitra'. Below is a table titled 'Data Simpro' with a search input field. The table lists four partner entries:

Show	10	entries
NO	Nama	
1	PT. INTI	
2	PT. TELKOM AKSES	
3	PT.INTI	
4	PT. RAHMI ROZA	

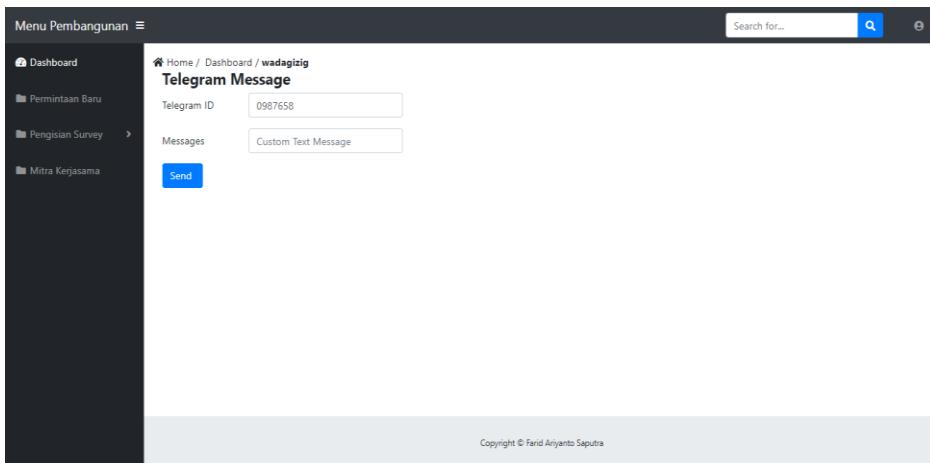
At the bottom, it says 'Showing 1 to 1 of 1 entries' and has navigation buttons for 'Previous' and 'Next'.

Gambar 7.5 Halaman Mitra kerjasama

Petunjuk pemakaian :

1. Lakukan login
2. Pilih menu mitra kerjasama
3. Klik button tambah mitra untuk menambahkan mitra yang akan bekerjasama

## 7.6 Halaman Pengiriman Pesan Telegram



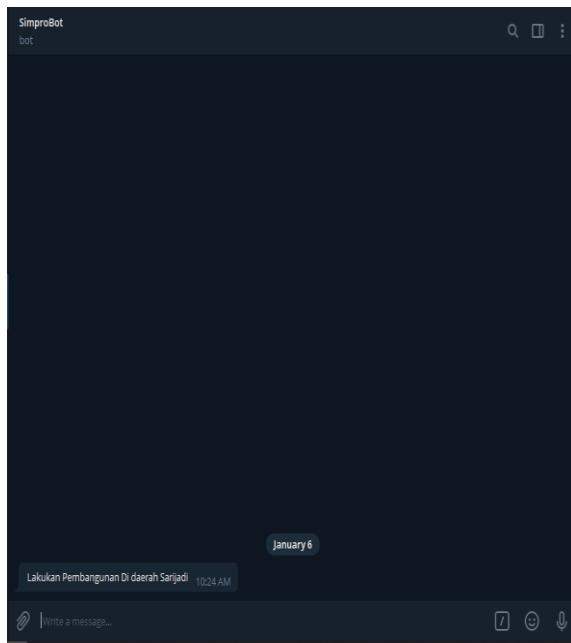
Gambar 7.6 Halaman Pengiriman Pesan Telegram

Petunjuk pemakaian :

1. Lakukan login
2. Klik menu pengisian survey
3. Klik sub menu SDI
4. Klik button kirim pesan
5. Masukkan id telegram yang dituju
6. Isikan pesan yang akan dikirim
7. Klik button Send
8. Tunggu notifikasi dari aplikasi mucul yang memberitahukan pesan telah terkirim
9. Klik Oke

10. Buka telegram

11. Berikut tampilan pesan terkirim



*Gambar 7.6.1 Pesan Terkirim*