

BAB II

LANDASAN TEORI

A. Pengertian Rumah Makan (*Restaurant*)

Definisi *restaurant* atau rumah makan ialah tempat bisnis yang menyediakan dan menjajakan makanan maupun minuman atau suatu bentuk olahan kepada pelanggan dengan imbalan uang dengan harapan pelanggan mendapatkan kepuasan dan pelayanan yang memuaskan. Baik makanan atau minuman atau bentuk olahan lainnya yang disajikan dan dinikmati ditempat, namun tak jarang banyak *restaurant* yang juga menawarkan dan memberi layanan pesan antar makanan dan minuman atau bentuk olahan lainnya yang dapat dikonsumsi.

Restaurant sangat bervariasi dalam penampilan dan penawaran layanan, termasuk berbagai macam masakan dan model layanan mulai dari jenis *restaurant* cepat saji dan kafetaria, hingga *restaurant* keluarga dengan harga menengah sampai harga tinggi. Banyak juga *restaurant* yang saat ini sedang booming ialah *restaurant* yang menu yang disediakan masih mentah atau belum dapat dikonsumsi namun mengharuskan pelanggan untuk memasak sendiri ditempat atau dibawa pulang.

Beberapa *restaurant* bisa menyajikan semua jenis menu sarapan, makan siang dan juga makan malam. Ada juga beberapa yang hanya menyajikan satu dari ketiga menu diatas. Layaknya sebuah *trend* baru bahwa *restaurant* sudah menjadi tempat pariwisata. Bahkan tak jarang *restaurant* tak hanya melegakan perut dari rasa lapar dan dahaga saja namun juga melegakan mata setiap orang yang datang.

Tak jarang juga kita temui rumah makan/*restaurant* yang disulap menjadi area bermain sekaligus sarana bereksplorasi bagi anak-anak. Contoh kecil dengan menyediakan tempat bermain yang asik bagi anak-anak.

Dapat penulis simpulkan *restaurant* adalah tempat untuk menjual makanan dan minuman yang mengutamakan fasilitas dan pelayanan yang ramah sebagai nilai jualnya walaupun harga yang ditawarkan bisa lebih mahal.

B. Pengenalan *Codeigniter*

Codeigniter sendiri merupakan kerangka yang dipakai dalam pemrograman bahasa *PHP*. Kalau kita berbicara bahasa pemrograman *PHP* maka akan terbesit dalam benak kita bahwa bisa kita beranggapan kalau *PHP* itu bisa diibaratkan sebagai bahasa Nasional. Sedangkan *Codeigniter* merupakan bahasa daerah. Maka dari pemaparan yang penulis berikan bisa diartikan kalau *Codeigniter* bukan Cuma kerangka dalam pemrograman *PHP* namun juga sebagai bahasa yang dibentuk dari pengolahan bahas pemrograman *PHP*.

Kalau kita berbicara tentang bahasa daerah dan bahasa Nasional maka bisa kita artikan kalau bahasa daerah lebih mudah kita pelajari. Sebagai contoh orang jawa akan lebih mudah berbicara jawa begitupun orang sunda akan lebih mudah juga berbahasa sunda. Namun perlu diperhatikan kalau mempelajari bahasa daerah lain akan lebih susah kalau kita belum mengenal bahasa Nasional. Jadi catatan pada pemaparan mengenai apa itu *Codeigniter* maka dari itu penulis menganggapnya sebagai bahasa daerah yang mana baha nasional adalah bahasa pemrograman *PHP*.

Hal yang membedakan kalau kita menggunakan kerangka atau *framework* dan tidak menggunakan atau *native* itu sangat terlihat entah itu dari efisiensi waktu yang digunakan maupun dari segi yang lain yang tentunya dengan adanya sebuah kerangka penulisan coding kita akan jauh lebih efisien. Karena pada prinsip dasarnya sebuah hal inovasi yang dilakukan tak lain memiliki tujuan yang amat penting yakni untuk mempermudah kinerja user atau bisa dibilang *developer* dalam mengembangkan aplikasi. *Codeigniter* sendiri kalau kita buka pada situs resminya menyediakan berbagai modul juga *library* yang cukup lengkap mulai dari instalasi hingga yang lebih kompleks. Dari beberapa pengalaman *developer* entah itu di artikel maupun di grup-grup yang membahas tentang *CodeIgniter*. *CodeIgniter* sendiri memiliki tingkat keamana yang lebih baik.

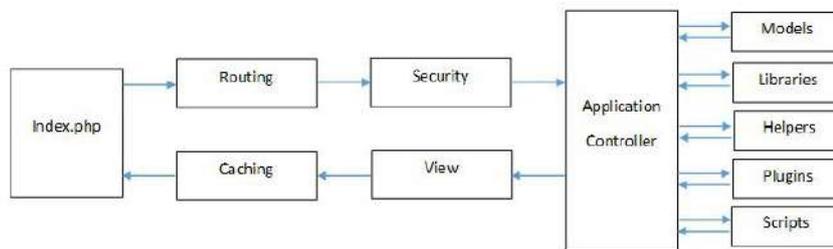
Sebagai *web framework* yang populer yang menggunakan bahasa pemrograman *PHP*, *Codeigniter* mempunyai berbagai keunggulan seperti yang disebutkan di dokumentasinya. Menurut Supono dan putratama (2009:109) menyatakan bahwa ada 9 keunggulan *Codeigniter* sebagai berikut:

1. Mempercepat sekaligus mempermudah pengembangan aplikasi *web*.
2. Dari segi *maintenance* lebih mudah dikendalikan bila saja sewaktu waktu terdapat bug atau error pada program. Ini dikarenakan *Codeigniter* menggunakan konsep *M-V-C* yakni *Model-View-Controller*.
3. Pada *framework Codeigniter* sudah terdapat banyak sekali *library* dan juga fasilitas-fasilitas pendukung lainnya yang membuat *Codeigniter* menjadi mudah untuk digunakan.

4. Lebih mudah dalam segi hal pengembangan. Itu karena *Codeigniter* bersifat *open source*.

C. Cara kerja *Codeigniter*

Untuk lebih memahamai cara kerja *Codeigniter*, berikut ini adalah sebuah diagram yang menjelaskan bagaimana *Codeigniter* bekerja menurut Supono dan putratama (2009:110).

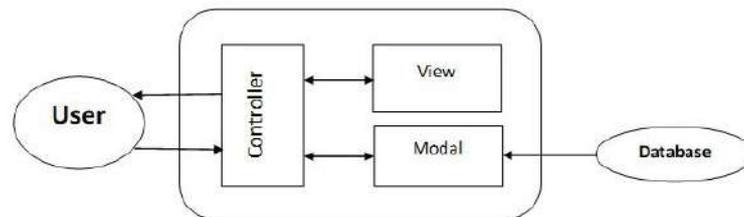


Gambar 2.1 alur bekerja *Codeigniter*

Berikut adalah cara kerja dari *Codeigniter*:

1. *Index.php* bertindak sebagai *controller* terdepan, dan menginisialisasi *resource* yang diperlukan untuk menjalankan *Codeigniter*.
2. *Router* memeriksa *HTTP request* untuk menentukan apa yang harus dikerjakan.
3. Jika *cache file* ada, maka akan ditampilkan langsung, dengan melewati eksekusi normal sistem.
4. Sebelum memuat *controller*, *HTTP request* akan memeriksa apa yang *disubumit user* dan memfilternya untuk keamanan.
5. *Controller* memuat *model*, *core libraries*, *plugin*, *helper*, dan *resource* lainnya untuk memproses permintaan tertentu.

6. *View* ditampilkan di *browser* sesuai proses yang dikerjakan *controller*. Jika *caching* dijalankan, *view* akan di-*cache* terlebih dahulu agar dapat ditampilkan di *request* selanjutnya.



Gambar 2.2 MVC Codeigniter

D. Bahasa Pemrograman Yang Digunakan

1. PHP

Pengertian *PHP* menurut Sibero (Supono dan Putratama, 2018:3) menyatakan bahwa *PHP (Personal Home Page)* adalah pemrograman (*interpreter*) adalah proses penerjemahan baris sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan.

Menurut Kustiyahningsih (Supono dan Putratama, 2018:3) *PHP* (atau resminya *PHP : Hypertext Preprosesor*) adalah skrip bersifat *severside* yang ditambahkan ke dalam HTML. *PHP* sendiri merupakan singkatan dari *Personal Home Page Tools*. Skrip ini akan membuat suatu aplikasi dapat diintegrasikan ke dalam HTML. Sehingga suatu halaman *web* tidak lagi bersifat statis, namun menjadi bersifat dinamis. Sifat *server side* berarti pengerjaan kode program dilakukan di *server*, baru kemudian hasilnya dikirimkan ke *browser*.

Berdasarkan pengertian di atas, dapat ditarik kesimpulan bahwa *PHP (PHP : Hypertext Preprosesor)* adalah suatu bahasa pemrograman yang digunakan

untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat server-side yang ditambahkan ke HTML. (Supono dan Putratama, 2018:3)

2. *AJAX*

AJAX merupakan bahasa pemrograman *web server-side* yang terbentuk dari penggabungan *HTML*, *JavaScript*, *DHTML* dan *DOM* yang menyerupai *PHP* dan *ASP*, sehingga membentuk suatu *web* yang interaktif dan menarik.

AJAX sendiri memiliki kepanjangan dari *Asynchronous JavaScript and XML*. *AJAX* sendiri bisa dibilang sebuah teknik penulisan suatu bahasa pemrograman yang lebih *simple*. Teknik penulisan *AJAX* sendiri bisa dituliskan pada halaman *html* sebuah *website* dengan menyisipkan *code AJAX* maka kita tidak perlu lagi untuk memerlukan sebuah ruang baru atau *file extension* baru untuk mere*load* suatu perintah. Bisa juga untuk menampilkan jendela halaman agar lebih efisien dalam pengembangan suatu aplikasi (Abdulloh, 2017:5).

E. Aplikasi Pendukung Dalam Pembuatan Sistem

1. *Visual Studio Code*

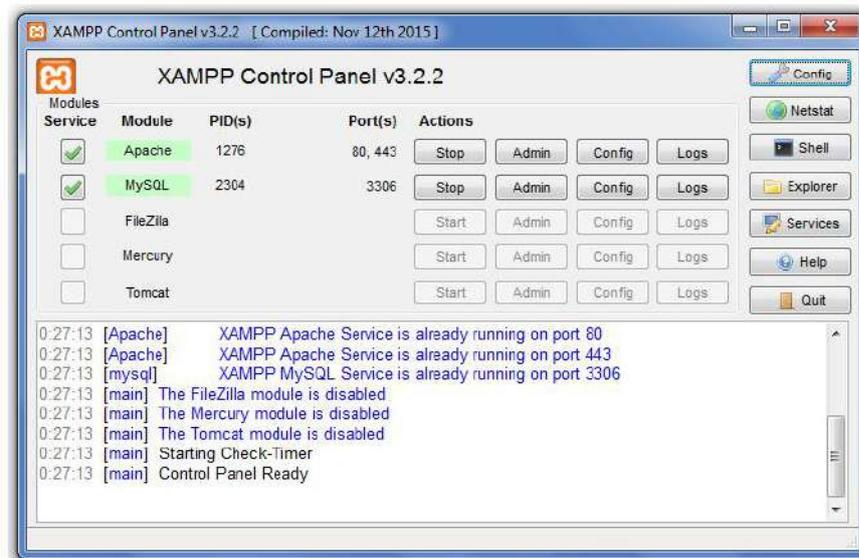
Dikutip dari buku seri belajar *ASP.net core mvc* dan *postgreSQL* dengan *visual studio code* yang ditulis oleh Faisal (2017:13). Dalam pembuatan aplikasi penulis dalam hal ini menggunakan sebuah teks editor visual studio code atau bahasa kerennya *Visual studio code*. *Visual studio code* adalah perangkat lunak yang lengkap yang dapat dilakukan untuk melakukan pengembangan aplikasi, baik aplikasi *website* maupun *mobile app* yang dikembangkan oleh *Microsoft* dan

bisa dilajankan di semua perangkat *operating system* baik *MAC*, *Windows*, maupun *Linux*. Awal pengembangan *Visual studio code* sendiri di rilis sejak setahun lalu dengan versi awal 1.0.

VS Code merupakan *text editor* yang memiliki banyak keunggulan dari *text editor* lainnya mulai dari segi penampilan yang *powerfull* hingga fitur yang mendukung hampir semua bahasa pemrograman.

2. *XAMPP*

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program *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 digunakan dalam melayani tampilan *web* yang dinamis.



Gambar 2.3 Tampilan *Start Xampp*

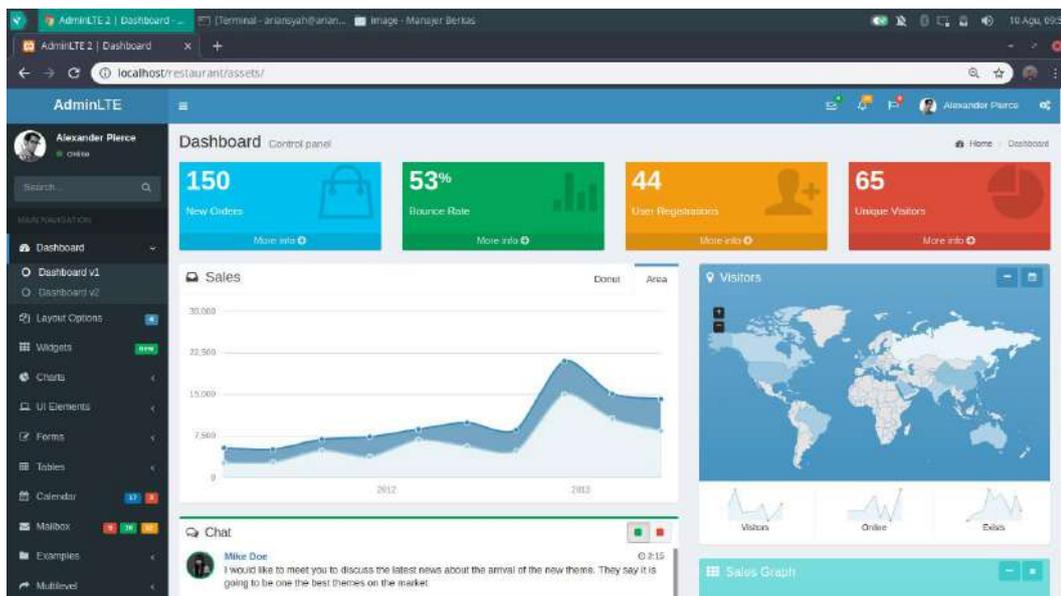
Menurut Asyikin (2019:4) menyatakan bahwa *XAMPP* merupakan singkatan dari *multi platform (X)*, *Apache (A)*, *MySQL (M)*, *PHP (P)*, *perl (P)*. Aplikasi ini simpel, ringan dan sangat memudahkan kita sebagai developer *web* untuk membuat *web server* lokal dengan berbagai macam kebutuhan misalnya maupun sebagai *server real*. *XAMPP* ini bisa berjalan pada berbagai macam sistem operasi seperti windows, linux, maupun mac *OS*.

Sebenarnya ada berbagai macam aplikasi serupa yang berjalan pada sistem operasi yang lebih spesifik seperti *LAMPP* pada *linux*, dan *MAMPP* pada mac *OS*. Tetapi kelebihan dari si *XAMPP* ini adalah *interface* yang sangat *user friendly* bagi para pemula yang baru memasuki dunia pemrograman *web* khususnya yang menggunakan *PHP* dan *MySQL*. Saat kita berpindah sistem operasi misalnya dari windows ke linux, kita juga tidak akan direpotkan lagi untuk mengkonfigurasi ulang *web server* lokal kita lagi karena *XAMPP* yang *interface* dan konfigurasinya tetap sama.

3. *Template Bootstrap ADMIN LTE*

Admin LTE merupakan template yang menyajikan tampilan yang *powerfull* sesuai dengan kebutuhan *developer*. Kita tidak perlu lagi repot-repot membuat tampilan *CSS* lagi. Karena disini semua fitur sudah sangat lengkap. Tinggal kita panggil saja hal yang kita perlukan di dalam mempercantik tampilan. Mulai dari *icon* sampai kebutuhan lainnya. Dan yang terpenting adalah template ini *free* alias *open source*. Dan bisa digunakan oleh siapa pun.

Berikut ini adalah gambaran tampilan *admin LTE*



Gambar 2.4 Tampilan *Admin LTE*