

BAB II

TINJAUAN PUSTAKA

I. Landasan Teori

A. Website

World Wide Web biasa dikenal dengan *Web* adalah layanan yang menyajikan informasi menggunakan konsep *hyperlink* (tautan) memfasilitasi pekerjaan pengguna Internet (istilah ini merujuk pada pengguna komputer yang menjelajah atau mencari informasi di Internet). Fungsionalitas ini telah menjadikan *web* sebagai layanan yang tumbuh paling cepat. *Web* memungkinkan kita untuk menyorot (menggarisbawahi) kata atau gambar dalam dokumen untuk ditautkan atau diarahkan ke media lain seperti dokumen frasa klip video atau file audio. *Web* dapat menautkan dari mana saja dalam dokumen atau gambar ke mana saja di dokumen lain. Dengan *browser* yang memiliki *Graphical User Interface* (GUI) tautan afiliasi dapat ditautkan ke tujuan mereka dengan mengarahkan *mouse* ke tautan dan mengkliknya. (Susilo, 2018).

Web adalah ruang yang dapat menampung informasi di internet dalam *browser*, dengan menambahkan kemampuan untuk memproses beberapa data dan kode yang biasa dikenal sebagai *beacon* (pemisah) dan kemampuan untuk melompat (*link*) dari halaman satu ke halaman lain. (Fitriani, Nurmiati dan Utomo, 2016).

Berdasarkan sifatnya, suatu *website* dibagi menjadi dua yakni:

1. *Website Statis*

Website Statis adalah *website* dimana pengguna tidak dapat merubah konten dari *web* tersebut secara langsung menggunakan *browser*. Interaksi yang terjadi antara pengguna dan server hanyalah seputar pemrosesan link saja. Halaman – halaman *web* tersebut tidak memiliki *database*, data dan informasi yang ada pada *web* statis tidak berubah – ubah kecuali diubah sintaksnya. Dokumen *web* yang dikirim *client* akan sama isinya seperti apa yang ada di *web server*.

2. *Website Dinamis*

Dalam *website dinamis*, interaksi yang terjadi antara pengguna dan *server* sangat kompleks. Pengguna *website* dapat mengubah konten dari halaman tertentu dengan menggunakan *browser*. Permintaan dari pengguna dapat diproses oleh *server* yang kemudian ditampilkan dalam isi yang berbeda – beda menurut alur programnya. Halaman – halaman *web* tersebut memiliki *database*. *Website dinamis* memiliki data dan informasi yang berbeda – beda tergantung input yang disampaikan *client*. Dokumen yang sampai di *client* akan berbeda dengan dokumen yang ada di *web server*. Perbedaan *web statis* dan *web* dinamis:

- a. Interaksi antar pengunjung dan pemilik *web* dalam *web* statis tidak dimungkinkan terjadinya interaksi antar pengunjung dengan pemilik *web*. Sementara dalam *web* dinamis terdapat interaksi antar pengunjung dengan pemilik *web*.

- b. *Web* statis hanya menggunakan satu bahasa *script* yaitu HTML, atau setidaknya ditambahkan dengan CSS. Sedangkan *web* dinamis menggunakan ahasa pemrograman *web* yang lebih kompleks seperti PHP, ASP, serta *Javascript*.
- c. *Website* statis tidak menggunakan *database* karena tidak ada data untuk disimpan dan diproses. Sedangkan *web* dinamis menggunakan *database* seperti MySQL Oracle dll. untuk menyimpan dan mengolah data.
- d. Konten *web* statis hanya disediakan oleh pemilik situs *web* dan jarang diupdate, sedangkan konten di *web* dinamis dapat berasal dari pengunjung dan lebih sering diupdate. Kemudian konten di *web* dinamis dapat diambil dari *database* sehingga konten mungkin berbeda meskipun kita membuka halaman *web* yang sama.

B. Framework

1. Pengertian *Framework*

Pengertian *Framework* adalah kerangka kerja yang digunakan untuk memfasilitasi pembuatan dan pengembangan aplikasi oleh pengembang perangkat lunak. *Framework* berisi perintah-perintah dasar dan fungsi-fungsi yang biasa digunakan untuk membuat perangkat lunak aplikasi sehingga diharapkan aplikasi dapat dibuat lebih cepat dan memiliki struktur yang cukup rapi. *Framework* juga dapat dipahami sebagai komponen pemrograman matang yang siap digunakan setiap saat sehingga

pengembang aplikasi tidak perlu lagi memuat skrip yang sama untuk tugas yang sama.

Banyak manfaat menggunakan *framework* dalam membangun sebuah *website*. Salah satu kelebihanannya adalah menyediakan struktur yang baik dalam program yang dihasilkan karena *framework* memiliki *library* atau fungsi yang dapat digunakan secara langsung. Selain itu *framework* memudahkan untuk bekerja sebagai tim dalam program karena ketika membangun situs *web* kita harus menyesuaikan gaya kerangka yang digunakan. (Ambriani dan Nurhidayat, 2019)

Salah satu *framework* yang banyak digunakan oleh *programmer* adalah *framework laravel*.

2. Fungsi *Framework*

- a. Mempercepat pengembangan aplikasi baik itu aplikasi *desktop* seluler atau *web*.
- b. Membantu para pengembang dalam merencanakan dan memelihara aplikasi.
- c. Aplikasi yang dihasilkan menjadi lebih stabil dan andal, karena *framework* telah menjalani proses pengujian untuk stabilitas dan keandalan.
- d. Mempermudah pengembang untuk membaca kode program dan menemukan *bug* dengan lebih mudah.
- e. Memiliki tingkat keamanan yang lebih tinggi itu karena Kerangka mengantisipasi kemungkinan lubang keamanan.

- f. Mempermudah pengembang untuk mendokumentasikan aplikasi yang sedang dibangun.

Dapat disimpulkan pengertian dari *framework* adalah sebuah tempat berupa kerangka kerja yang berisi kode pustaka untuk memudahkan dan mempercepat proses pembuatan aplikasi *website* tanpa harus membuat fungsi *class* dari awal sehingga akan lebih mudah dalam mengelola aplikasi *web* dan pengembangannya.

C. *Laravel*

Laravel merupakan *framework* berbasis PHP yang bersifat *open source* dan menggunakan konsep *model – view – controller*. *Laravel* dilisensikan di bawah lisensi MIT menggunakan *Github* sebagai *platform* berbagi kode.

Berikut adalah dasar – dasar *laravel*:

1. *Artisan*

Artisan adalah *command line* atau perintah yang dijalankan melalui terminal dan menyediakan beberapa baris perintah yang dapat digunakan saat mengembangkan dan membangun aplikasi. Salah satu fungsi dari php artisan yaitu php artisan *serve*. Php artisan *serve* berfungsi untuk membuka *website* yang telah dibuat tanpa menggunakan *web* server lokal.

2. *Routing*

Routing adalah suatu proses yang bertujuan untuk membawa suatu barang yang diinginkan ke tujuannya. Menggunakan perutean dimungkinkan untuk menentukan halaman mana yang akan muncul

ketika pengguna membukanya. *Setting routing* di *laravel* biasanya terdapat di file `we.php`. File `we.php` dapat ditemukan di rute.

3. *Controller*

Controller adalah sebuah proses yang tujuannya adalah untuk mengambil permintaan memuat *instance* memanggil model untuk mengirim ke tampilan. Ada dua cara untuk memuat *controller* di *laravel*. Yang pertama adalah membuat file pengontrol secara manual dan menulis kode pengontrol yang meluas ke dalamnya. Cara kedua adalah memuat file *controller* menggunakan *command line* dengan menulis php Artian `make controller filename_controller`.

4. *View (blade templating)*

Blade adalah *template engine laravel* secara *default*. *Blade* menghasilkan kode *laravel* dengan lebih mudah. Pemuatan *file.blade* dilakukan secara manual dengan membuat `file_name.php.blade` di folder *views*. Bagian dalam pisau bisa dibuat *template* master dan *template inheritance*. Pembuatan *template* master dan turunannya ini bertujuan agar elemen yang sama tidak ditulis secara berulang-ulang. Pada *template inheritance* diberikan kode `extend (nama_layout)` dan `section (nama_content)`.

5. *Middleware*

Middleware adalah perantara antara permintaan yang masuk dan pengontrol tujuan. Cara memuat *middleware* secara manual dengan

mengetikkan `php Artisan make: middleware file_name`. File *middleware* terletak di folder *middleware*.

6. *Session*

Session adalah metode yang digunakan untuk *hosting* di server dan penyimpanan ini digunakan di beberapa halaman termasuk situs itu sendiri. Ada dua cara untuk menggunakan sesi. Cara pertama untuk memuat sesi adalah dengan menggunakan *Permintaan*. Metode kedua dapat digunakan oleh fitur sesi bantuan global.

7. *Migration*

Migration adalah fitur yang disertakan dalam *laravel* dan sistem kontrol versi untuk *database*. Dengan menggunakan transformasi penulis dapat membuat tabel data dengan lebih mudah dan cepat. Migrasi menghasilkan atau menghasilkan file migrasi sebagai sistem kontrol. Ini adalah file yang dapat dikirim oleh pengembang satu sama lain saat membangun aplikasi. Cara migrasi menggunakan artisan dengan mengetik `php artisan make: convert create_namatable_table - create = namatable`.

8. *Model*

Model merupakan salah satu bagian MVC bertanggung jawab untuk menangani *database* secara langsung. Kita juga dapat mengatakan bahwa model adalah penghubung untuk setiap aliran program terikat data. Model yang terhubung ke *database* kemudian digunakan atau dipanggil melalui pengontrol selama implementasi konsep MVC. Cara

memuat model menggunakan artisan dengan mengetikkan php Artisan
make: model_model_name.

D. Studio

Studio dapat diartikan sebagai ruang bengkel atau tempat di mana seseorang aktif menghasilkan karya. Studio adalah tempat yang digunakan untuk memproduksi film, video, foto dan karya seni lainnya. Secara umum sebuah studio perlu menyediakan berbagai peralatan untuk dapat mencakup semuanya mulai dari tahap awal hingga tahap penyelesaian akhir dari sebuah produksi audiovisual. Saat membangun studio ada beberapa persyaratan yang perlu diperhatikan yaitu persyaratan suara pencahayaan, AC, dan lain-lain (Purnamaria, Azizah dan Ramadhani, 2019).

Berdasarkan uraian diatas, dapat disimpulkan pengertian studio adalah tempat kerja para photographer atau videographer untuk memproduksi foto, video dan karya yang lainnya. Vibesroom Studio merupakan studio foto sekaligus video yang dimana kita dapat mengabadikan momen yang nantinya bisa di cetak.

E. Basis Data

Database atau basis data adalah kumpulan informasi yang disimpan secara sistematis dalam komputer sehingga dapat diverifikasi oleh program komputer untuk mengambil informasi dari *database*. Penggunaan utama dari sistem *database* adalah bahwa pengguna dapat membangun tampilan data abstrak. Hal ini bertujuan untuk menyederhanakan interaksi antara pengguna dan sistem dan *database* dapat menyajikan pandangan

yang berbeda kepada pengguna *programmer* dan administrator (Aprilyana, Munti dan Adeswastoto, 2021).

F. PHP

PHP adalah Bahasa dirancang khusus untuk digunakan di *web*. PHP adalah alat untuk membuat halaman *web* dinamis. Awalnya PHP adalah singkatan dari Personal Site. PHP pertama kali diuat oleh Rasmus Lerdorf pada tahun 1995. Saat itu PHP masih dikenal dengan nama FI (*Form Interpreted*) yang berupa sekumpulan *script* yang digunakan untuk mengolah data *form* dari *web*. Saat ini PHP adalah singkatan dari PHP: *Hypertext Preprocessor* yang merupakan singkatan dari *recursive* yang merupakan plesetan yang akronimnya terdiri dari singkatan itu sendiri: PHP: *Hypertext* (Lutfi, 2017).

G. MYSQL

MySQL adalah salah satu aplikasi DBMS (*Database Management System*) yang sudah banyak digunakan oleh para *programmer* aplikasi *web*. Dalam sistem basis data *non-relasional* semua informasi disimpan dalam satu bidang besar yang terkadang sulit dan memakan waktu untuk mengakses data yang dikandungnya. Tetapi MySQL adalah sistem *database* relasional sehingga dapat mengelompokkan informasi ke dalam tabel atau kelompok informasi terkait. Setiap tabel berisi bidang terpisah mewakili setiap bit informasi. MySQL menggunakan indeks untuk mempercepat pencarian baris informasi tertentu. MySQL membutuhkan setidaknya satu

indeks per tabel. Biasanya akan menggunakan kunci utama atau pengenal unik untuk membantu melacak data (Lutfi, 2017).

H. HTML

HTML atau *Hyper Text Markup Language* adalah salah satu format yang digunakan untuk membuat dokumen dan aplikasi yang berjalan di halaman *web*. *Hypertext Markup Language* (HTML) adalah bahasa *markup* yang digunakan untuk membuat halaman *web* menampilkan berbagai informasi di *browser web* Internet dan format *hypertext* sederhana yang ditulis dalam file format ASCII. Dengan demikian konsep HTML (*Hyper Text Markup Language*) adalah bahasa *markup* yang digunakan untuk membuat halaman *web* dan menampilkan berbagai informasi di *browser* Internet (Lidiyasari, 2020).

I. CSS

CSS merupakan Salah satu bahasa desain *web* (*Style Sheet Language*) yang mengontrol format tampilan halaman *web* yang ditulis dalam kode bahasa *markup*. Biasanya CSS digunakan untuk mendesain halaman HTML dan XHTML tetapi sekarang CSS dapat diterapkan ke semua dokumen XML termasuk SVG dan XUL dan bahkan android (Tampubolon, 2018).

CSS merupakan singkatan dari *Cascading Style Sheet*. Tujuannya adalah untuk mengatur tampilan dokumen HTML misalnya dengan mengatur spasi baris teks warna dan format atas bahkan tampilan file gambar baris. CSS dikembangkan oleh W3C. Organisasi pengembangan

teknologi internet. Tujuannya tak lain untuk mempermudah proses penataan halaman (Setiawan dan Rostianingsih, 2020).

Meskipun HTML memiliki kemampuan untuk menyesuaikan tampilan dan nuansa halaman *web* kemampuannya sangat teratas. Fungsi CSS adalah untuk memerikan parameter yang lebih lengkap sehingga struktur halaman *web* yang terbuat dari HTML lebih bersih dan indah. (Josi, 2017).

J. Pemesanan

Pemesanan adalah kegiatan yang dilakukan oleh konsumen sebelum membeli. Untuk mencapai kepuasan pelanggan perusahaan harus memiliki sistem pemesanan yang baik. Mendefinisikan pemesanan adalah “proses, perbuatan, cara memesan (tempat, barang, dsb) kepada orang lain” (Purwanto dan Nugroho, 2018)

Pemesanan dalam arti umum perjanjian pemesan antara 2 (dua) pihak atau lebih perjanjian pemesanan yang dapat berupa perjanjian pemesanan aula tata letak dan lain-lain pada waktu tertentu dan disertai dengan produk dan layanannya. Produk atau jasa yang dimaksud adalah jasa yang diberikan dalam kontrak pemesanan misalnya pada perusahaan penerangan atau pelayaran yaitu perpindahan orang atau barang dari satu titik (kota) ke titik lain (kota) (Purwanto dan Nugroho, 2018).

Dalam melakukan pemesanan hal yang harus diperhatikan adalah sebagai berikut:

1. *Product*

Produk, adalah segala sesuatu yang dapat ditawarkan ke pasar untuk diperhatikan diperoleh dan digunakan atau dikonsumsi untuk memuaskan kebutuhan dan keinginan termasuk barang fisik jasa orang titik tempat organisasi dan ide.

2. *Price*

Harga yaitu jumlah yang harus dibayar pelanggan untuk memperoleh suatu produk atau jasa untuk memperoleh suatu produk dan harganya harus terjangkau oleh kantong konsumen.

3. *Place*

Salurana atau lokasi distribusi termasuk kegiatan usaha untuk menyediakan produk atau jasa kepada konsumen. Akses layanan yang mudah bagi pelanggan. Dimana suatu produk berada dan didistribusikan menentukan seberapa mudah bagi pelanggan untuk mengakses layanan. Tempat yang menyediakan produk di sejumlah saluran distribusi dan agen untuk membantu konsumen membeli produk dengan mudah.

4. *Promotion*

Promosi mengacu pada kegiatan yang mencakup berbagai metode yaitu periklanan promosi penjualan komunikasi produk dan persuasi pelanggan. Jelaskan berbagai cara yang digunakan bisnis untuk menjual produk kepada konsumen. Penjualan langsung dan hubungan

masyarakat komunikasi produk dan persuasi. Jelaskan berbagai cara bisnis menjual produk kepada konsumen.

K. *Responsive Web Design*

Responsive Web Design (RWD) adalah pendekatan pengemangan *web* yang memuat perubahan dinamis pada tampilan dan nuansa halaman *web* tergantung pada ukuran layar dan orientasi perangkat yang menampilkannya. RWD adalah pendekatan untuk memecahkan masalah desain untuk berbagai perangkat yang tersedia bagi pelanggan dari ponsel kecil hingga monitor *desktop* besar. (Ramsari dan Rifaldi, 2018)

RWD menggunakan apa disebut *breakpoints* untuk menentukan bagaimana tata letak halaman *web* akan muncul. Satu desain digunakan di atas *breakpoint* dan desain lainnya diterapkan di bawah *breakpoint*. *Breakpoint* biasanya didasarkan pada lebar *browser* (Ramsari dan Rifaldi, 2018).

Ini karena dengan menggunakan desain responsif desainer dapat menerapkan solusi untuk resolusi layar kepadatan dan rasio yang berbeda di berbagai perangkat. Desain responsif yang mampu mengelola sumber daya multimedia secara efisien.

Ada tiga elemen utama dalam teknologi *responsive web design* yaitu:

1. Tata letak fleksibel berbasis grid
2. Gambar dan media fleksibel
3. *Media queries*

Website responsive penelitian ini sangat bermanfaat bagi mahasiswa karena aplikasi dapat dibuka melalui handphone sehingga memudahkan mahasiswa untuk mengadakan segregasi. (Melani, 2019).

L. UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. Dapat didefinisikan UML adalah standart bahasa untuk mendefinisikan dari *requirement*, membuat analisa dan desain dan menggambarkan arsitektur dalam pemrograman yang berorientasi pada objek (Josi, 2017)

Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan pengembang sistem untuk membuat cetak baru atas visi mereka dalam bentuk yang baku. Sebagai sebuah sketsa, UML berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dari sistem (Sunge dan Awaludin, 2020).

UML sendiri juga memberikan standar penulisan sebuah *sistem blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam *system software* (Lesmana, 2016).

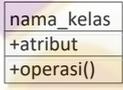
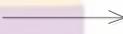
Dari pengertian diatas dapat disimpulkan bahwa UML merupakan sebuah pemodelan untuk membangun sebuah perangkat lunak dengan teknik pemrograman OOP atau berorientasi objek untuk memberi

gambaran dan merancang, membuat model dan mendokumentasikan aspek-aspek dari sebuah sistem.

a. *Class Diagram*

Class Diagram adalah spesifikasi yang akan menghasilkan objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut atau properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut.

Tabel 2.1 Simbol Diagram Kelas

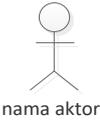
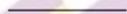
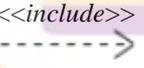
No	Nama	Simbol	Keterangan
1	Operasi		Kelas pada struktur sistem.
2	Antarmuka/ <i>interface</i>		Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3	Asosiasi berarah		Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain.
4	Generalisasi		Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
5	Kebergantungan		Relasi antar tabel dengan makna kebergantungan antar kelas.
6	Agresi		Relasi antar kelas dengan makna semua bagian.

Sumber (Maria, 2019)

b. *Usecase Diagram*

Usecase Diagram merupakan diagram yang harus dibuat pertama kali saat pemodelan pemrograman perangkat lunak berorientasi objek dilakukan. *Usecase Diagram* akan menggambarkan apa yang dikerjakan oleh aktor.

Tabel 2.2 Simbol Diagram *Use Case*

No	Nama	Simbol	Keterangan
1	<i>Use case</i>		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
2	Aktor/ <i>actor</i>		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
3	Asosiasi/ <i>association</i>		Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4	<i>Generalisasi</i>		Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
5	<i>Extend</i>		Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan itu.
6	<i>Include</i>		Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.

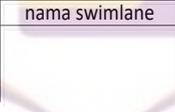
Sumber (Maria, 2019)

c. *Activity Diagram*

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya. Diagram ini sangat mirip dengan *flowchart* karena memodelkan *workflow* dari suatu aktifitas ke aktifitas yang lainnya, atau dari aktifitas ke status. Pembuatan diagram *activity* pada awal pemodelan proses dapat membantu memahami

keseluruhan proses. *Activity* diagram juga digunakan untuk menggambarkan interaksi antara beberapa *use case*.

Tabel 2.3 Simbol Diagram Aktivitas

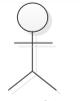
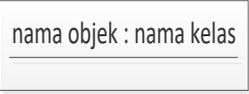
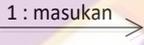
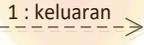
No	Nama	Simbol	Keterangan
1	Status awal		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2	Aktivitas		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3	<i>Decision</i>		Pilihan untuk pengambilan keputusan.
4	<i>Join</i>		Aktivitas untuk menunjukkan kegiatan yang dilakukan secara paralel atau bercabang.
5	Status akhir		Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6	<i>Swimlane</i>		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
7	Percabangan		Menunjukkan adanya dekomposisi
8	Penggabungan		Tanda penerima

Sumber (Maria, 2019)

d. *Sequence Diagram*

Menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem

Tabel 2.4 Simbol Diagram *Sequence*

No	Nama	Simbol	Keterangan
1	Aktor	 nama aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat.
2	Garis hidup/ <i>lifeline</i>		Menyatakan kehidupan suatu objek
3	Objek		Menyatakan objek yang berinteraksi pesan
4	Waktu aktif		Menyatakan objek dalam keadaan aktif dan berinteraksi pesan
5	Pesan tipe <i>send</i>		Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
6	Pesan tipe <i>return</i>		Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.

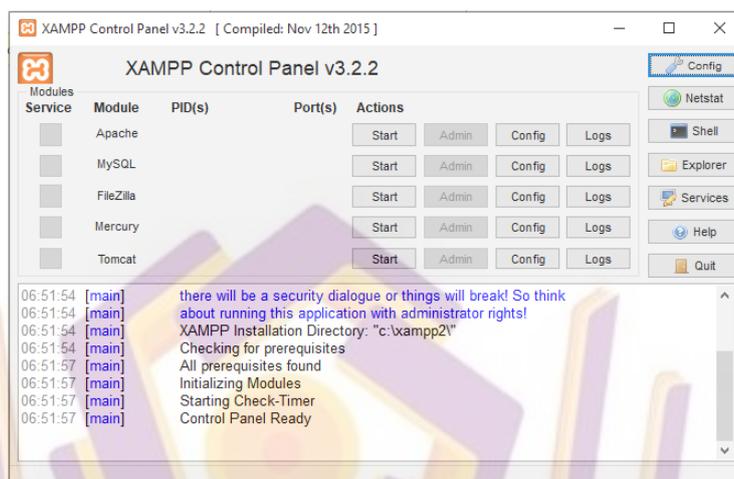
Sumber (Maria, 2019)

M. XAMPP

Xampp adalah paket *web* server populer yang digunakan untuk pengujian di *Windows* karena mudah dipasang. Xampp adalah *software* gratis yang mendukung banyak sistem operasi yang merupakan gabungan dari beberapa program (Sarwindah, 2018).

Fungsionalitasnya adalah server yang dihosting sendiri (*localhost*) yang terdiri dari server HTTP *Apache*. *Database* MySQL dan terjemahan bahasa ditulis dalam bahasa pemrograman PHP. Nama Xampp mengacu pada empat sistem operasi yaitu *Apache* MySQL PHP dan Perl (Sarwindah, 2018).

Berdasarkan penjelasan diatas dapat disimpulkan pengertian XAMPP adalah perangkat lunak *open source* yang mendukung berbagai sistem operasi untuk pengembangan aplikasi dengan basis LAMP (*Linux, Apache, MySQL, PHP dan Perl*).



Gambar 2.1 Tampilan Software Xampp

Gambar 2.1 Menggambarkan tampilan awal dari software XAMPP Control Panel versi 3.2.2.

II. Penelitian Sebelumnya

Penelitian sebelumnya dari peneliti lainnya yang dijadikan sebagai acuan atau bahan perbandingan, penelitian sebelumnya merupakan hal yang sangat perlu dan dapat dijadikan sebagai data pendukung. Dilihat dari beberapa kesamaan pembahasan dari penelitian terdahulu yang dijadikan tolak ukur oleh peneliti. Fokus penelitian terdahulu yang di jadikan acuan adalah *website*

booking, oleh karena itu peneliti melakukan kajian terhadap hasil – hasil penelitian sebelumnya atau jurnal terdahulu.

Penelitian sebelumnya tentang *website booking* dilakukan oleh (Anharudin, Fernando dan Putri, 2018) yang berjudul Rancang bangun sistem informasi *E-Booking* ruang karaoke berbasis *Web*. Penelitian ini bertujuan untuk mempermudah perancangan aplikasi pemesanan elektronik dan penggunaannya, mempersingkat waktu dan dapat diakses kapan saja, dimana saja, sehingga memudahkan calon pelanggan dalam mencari informasi mengenai tempat karaoke.

Penelitian selanjutnya selanjutnya dilakukan oleh (Putra, Wara dan Hariyanto, 2020) yang berjudul Aplikasi pencarian dan *booking* hotel pada travel *agent* berbasis *Web* menggunakan *Framework Codeigneter*. Penelitian ini bertujuan untuk para konsumen dapat mengakses dan memesan secara langsung pada *website* tersebut, serta dapat mengefisiensikan dan megefektifkan prooduktifitas kinerja dari para karyawan.

Penelitian selanjutnya dilakukan oleh (Purwanto dan Nugroho, 2018) yang berjudul Analisis Rancangan teknologi *mobile application* Pada *e-booking* ruangan untuk kegiatan ukm. Penelitian ini bertujuan untuk mengatasi peminjaman ruangan tanpa harus datang ke ruangan tersebut dan juga dapat memonitoring penggunaan ruangan tersebut. Aplikasi ini memungkinkan pelanggan melakukan registrasi dan melakukan pemesanan ruangan sesuai dengan waktu yang dibutuhkan serta jenis ruangan dengan kapasitas ruangan.

Penelitian selanjutnya dilakukan oleh (Triyono, Minarsih dan Oktavia, 2018) yang berjudul Perancangan sistem informasi *booking* buku berbasis *Web* pada perpustakaan SMK Pancakarya Tangerang. Penelitian ini bertujuan untuk merancang sistem informasi reservasi buku pada perpustakaan *online* yang berfungsi untuk pemesanan buku pinjaman *online* di SMK Pancakarya Tangerang, kemudian mempelajari sistem informasi perpustakaan yang bekerja di SMK Pancakarya Tangerang, untuk mengetahui kebutuhan informasi mengenai keanggotaan proses pelayanan di Perpustakaan SMK Pancakarya Tangerang.

Penelitian selanjutnya dari (Muljono, Gunadi dan Nugroho, 2020) yang berjudul Rancang bangun *Website* pemesanan makanan kedai *Twins* menggunakan *Laravel PHP Framework*. Penelitian ini bertujuan agar transaksi menjadi lebih mudah dan cepat, sehingga administrator dan pengguna memiliki manfaat yang sama, proses pelayanan juga cepat dapat dilakukan dimana saja dan sering.

Tabel 2.1 Penelitian sebelumnya

No	Nama Penelitian	Persamaan dengan penelitian sebelumnya	Perbedaan dengan penelitian sebelumnya
1	Anharudin, Donny Fernando, dan Novi Khristina Putri. 2018	Penelitian sebelumnya membahas tentang <i>booking</i> berbasis <i>website</i> dan menggunakan UML sebagai perancangan sistemnya	a) Penelitian sebelumnya menggunakan metode pengembangan <i>waterfall</i> . Sedangkan penelitian ini menggunakan metode pengembangan <i>Extreme Programming</i> b) Sistem hanya bisa berjalan di satu <i>personal computer</i> (PC). Sedangkan sistem pada penelitian dapat dijalankan oleh beberapa pengguna dengan menggunakan komputer ataupun <i>smartphone</i>
2	Andreas Adi Putra Wara dan Susanto Hariyanto. 2020	Membahas tentang <i>booking</i> berbasis <i>website</i> dan pengelolaan <i>database</i> menggunakan <i>MySQL</i>	a) Penelitian sebelumnya mengelola data hotel. Sedangkan pada penelitian ini mengelola data studio foto b) Penelitian sebelumnya menggunakan <i>framework Codeigniter</i> sedangkan pada penelitian ini menggunakan <i>framework Laravel</i>
3	Purwanto dan Nugroho. 2018	Penelitian sebelumnya membahas tentang <i>booking</i> berbasis <i>website</i>	a) Penelitian sebelumnya menggunakan aplikasi <i>Android</i> studio sedangkan penelitian ini menggunakan <i>visual studio</i> b) Penelitian sebelumnya digunakan untuk civitas sekolah sedangkan penelitian bisa digunakan untuk umum
4	Triyono, Diah Minarsih dan Dwi Oktavia. 2018	Penelitian ini membahas <i>website</i> dan melakukan metode <i>booking</i>	a) Penelitian sebelumnya, <i>website</i> tersebut digunakan sebagai pengecekan ketersediaan buku yang hanya dilakukan di perpustakaan. Sedangkan penelitian ini bisa digunakan kapan saja dan dimana saja karena <i>website</i> ini bersifat responsif b) Penelitian sebelumnya terintegrasi
5	Niko Cahyono Slamet Muljono, Devina Gunadi dan Agus Cahyo Nugroho. 2020	Penelitian ini membahas rancang bangun <i>website</i> menggunakan <i>framework Laravel</i>	a) Penelitian sebelumnya hanya ditujukan kepada si pemilik, Sedangkan penelitian ini ditujukan kepada konsumen dan juga si pemilik b) Penelitian sebelumnya mengelola data menu makananan. Sedangkan penelitian ini mengelola data pembookingan studio