

SKRIPSI

RANCANG BANGUN SISTEM PENJARINGAN UMPAN BALIK TINGKAT

KEPUASAN PEMANGKU KEPENTINGAN MENGGUNAKAN

ARSITEKTUR MODEL VIEW CONTROLLER

Disusun dan diajukan oleh :

MUHAMMAD SHADIQ

D42114319



DEPARTEMEN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

MAKASSAR

2021

LEMBAR PENGESAHAN SKRIPSI
RANCANG BANGUN SISTEM PENJARINGAN UMPAN BALIK
TINGKAT KEPUASAN PEMANGKU KEPENTINGAN MENGGUNAKAN
ARSITEKTUR MODEL VIEW CONTROLLER

Disusun dan diajukan oleh

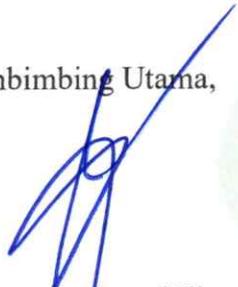
MUHAMMAD SHADIQ

D42114319

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian Studi Program Sarjana Program Studi Teknik Informatika Fakultas Teknik Universitas Hasanuddin pada tanggal 06 Oktober 2021 dan dinyatakan telah memenuhi syarat kelulusan.

Menyetujui,

Pembimbing Utama,



Dr. Amil Ahmad Ilham, ST.,MIT.
Nip. 197310101998021001

Pembimbing Pendamping,



Dr. Eng. Jr. Zulkifli Tahir, ST.,M.Sc.
Nip. 198404032010121004

Ketua Program Studi,



Dr. Amil Ahmad Ilham, S.T., M.IT.
Nip. 197310101998021002

PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini :

Nama : MUHAMMAD SHADIQ

NIM : D42114319

Program Studi : S1 Teknik Informatika

Menyatakan dengan sebenar-benarnya bahwa skripsi yang berjudul:

**RANCANG BANGUN SISTEM PENJARINGAN UMPAN BALIK TINGKAT
KEPUASAN PEMANGKU KEPENTINGAN MENGGUNAKAN
ARSITEKTUR MODEL VIEW CONTROLLER**

Adalah karya ilmiah saya sendiri dan sepanjang pengetahuan saya di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan/ditulis/diterbitkan sebelumnya, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila dikemudian hari ternyata didalam naskah skripsi ini terdapat unsur-unsur dijiplakan, saya bersedia menerima sanksi atas perbuatan tersebut dan diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2000, pasal 25 ayat 2 dan pasal 70).

Makassar, 14 Oktober 2021

Yang membuat Pernyataan,



MUHAMMAD SHADIQ

ABSTRAK

Keberhasilan pengelolaan survey kepuasan pemangku kepentingan Perguruan Tinggi tergantung pada keakuratan data dan kecepatan dalam penyediaan informasi sesuai yang dibutuhkan. Untuk mendukung hal tersebut, maka telah dibangun sebuah sistem yang dapat mencatat dan sekaligus mengolah data hasil umpan balik kepuasan pemangku kepentingan. Sistem ini memakai bahasa pemrograman PHP dengan menggunakan arsitektur Model View Controller, salah satunya yaitu framework Laravel mengacu pada MySQL sebagai DBMS (*Database Management System*). Sebagai studi kasus teknologi ini akan di manfaatkan untuk menunjang proses evaluasi pengelolaan survey kepuasan pemangku kepentingan Perguruan Tinggi. Proses analisis penelitian ini, membandingkan aplikasi *web* menggunakan *tool* apache jmeter dan *developer tools google chrome*. Dalam analisis performansi dengan menggunakan *tool* apache jmeter dan *developer tools google chrome*, hasil menunjukkan bahwa, Semakin besar jumlah thread (user) yang mengakses sistem secara bersamaan, maka semakin besar pula waktu rata-rata *response time* yang dihasilkan, Semakin besar jumlah thread (user) yang mengakses sistem secara bersamaan, maka semakin stabil dan konsisten pula throughput/sec rata-rata yang dihasilkan. Hasil yang didapatkan dari proses pengujian menggunakan metode *black box* memberikan keluaran yang diharapkan pada setiap proses *input, edit, dan delete* data. Dengan demikian, diharapkan sistem ini dapat diimplementasikan pada pengelolaan survey kepuasan pemangku kepentingan di Perguruan Tinggi.

Kata Kunci: *web, tool apache jmeter dan developer tools google chrome*

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Segala puji dan syukur Penulis panjatkan ke hadirat Allah S.W.T, Tuhan Yang Maha Esa yang dengan limpahan rahmat dan hidayah-Nya sehingga tugas akhir dengan judul “*Rancang Bangun Sistem Penjaringan Umpan Balik Tingkat Kepuasan Pemangku Kepentingan Menggunakan Arsitektur Model View Controller*” ini dapat diselesaikan sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Dalam penyusunan penelitian ini disajikan hasil penelitian terkait judul yang telah diangkat dan telah melalui proses pencarian dari berbagai sumber baik jurnal penelitian, prosiding pada seminar-seminar nasional ataupun internasional, buku, dan dari berbagai situs-situs terpercaya yang ada di internet.

Penulis menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, mulai dari masa perkuliahan sampai dengan penyusunan tugas akhir, sangatlah sulit untuk menyelesaikan tugas akhir ini. Oleh karena itu, pada kesempatan ini Penulis menyampaikan ucapan terima kasih sedalam-dalamnya kepada :

- 1) Allah S.W.T karena atas semua berkat, karunia serta pertolongan-Nya yang tiada batas, yang telah diberikan kepada Penulis disetiap langkah dalam pembuatan tugas akhir ini hingga penulisan laporan skripsi ini;
- 2) Kedua orang tua Penulis, Bapak Syaharuddin Kasim dan Ibu Johar yang selalu mendidik Penulis dan menjadi tempat untuk berkeluh kesah serta selalu memberikan dukungan, doa dan semangat kepada Penulis sejak kecil. Terima kasih untuk selalu ada bagi Penulis;
- 3) Bapak Dr. Amil Ahmad Ilham, ST., M.IT., selaku pembimbing I dan Bapak Dr. Eng. Ir. Zulkifli Tahir, ST.M.Sc., selaku pembimbing II yang selalu menyediakan waktu, tenaga, pikiran dan perhatian yang luar biasa untuk mengarahkan Penulis dalam penyusunan tugas akhir ini;
- 4) Bapak Robert, Bapak Zainuddin, Ibu Yuanita dan Ibu Santi serta segenap staff Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah membantu kelancaran penyelesaian tugas akhir Penulis;
- 5) Saudara Hermawan, ST., yang telah membantu saya dalam penelitian dan memberikan motivasi;

- 6) Saudara Alriefqy Dasmito, ST., yang telah membantu saya dalam pengujian dan memberikan inspirasi;
- 7) Saudara seperjuangan Penulis Teknik 2014 yang telah menemani perjalanan penulis dalam dunia pengkaderan sebagai anak teknik;
- 8) Keluarga besar Rectifier'14 yang juga menjadi rekan sekaligus tempat berbagi keluh dan kesah selama mengarungi dunia kampus sebagai anak teknik;
- 9) Teman-teman Adhyaksa Boys khususnya Muh. Nur Alamsyah, Yakip, Syarif Hidayatullah, ST, Rahmat Firman, Muh. Ardiansyah, ST, Ahmad Setiadi, Jamaluddin, Arya Jaka Putra, Siswono Nasir, Ulfah Rojiyyah yang selalu menemani Penulis dalam mengarungi segala rintangan dalam dunia perkuliahan;
- 10) Seluruh pihak yang tidak sempat disebutkan satu persatu yang telah banyak meluangkan tenaga, waktu dan pikiran selama penyusunan laporan tugas akhir ini.

Akhir kata, penulis berharap semoga Allah SWT. berkenan membalas segala kebaikan dari semua pihak yang telah banyak membantu. Semoga Tugas Akhir ini dapat memberikan manfaat bagi pengembangan ilmu. Aamiin.

Wassalammualaikum wr.wb
Makassar, 14 Oktober 2021

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PENGESAHAN SKRIPSI	ii
PERNYATAAN KEASLIAN	iii
ABSTRAK	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	x
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan dan Batasan Masalah	2
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	3
1.5 Sistematika Penulisan	3
BAB II. TINJAUAN PUSTAKA	4
2.1 Sistem Informasi	4
2.2 Pengertian Website	8
2.3 Website Modern	9
2.4 PHP	11
2.5 Compuser	14
2.6 Xampp (local host)	14
2.7 Basis Data	17
2.8 MySQL	22
2.9 Gitlab	23
2.10 Visual Studio Code	23
2.11 MVC	24
2.12 Framework Laravel	27
2.13 Pengembangan Sistem	32
BAB III METODOLOGI PENELITIAN.....	34
3.1 Lokasi dan Waktu Penelitian	34
3.2 Instrumen Penelitian	34
3.3 Prosedur Penelitian	35

3.4 Tahap Persiapan	36
3.5 Gambaran Umum Sistem	36
3.5.1 Tahap Desain Perancangan Sistem	36
3.5.2 Use Case Diagram	37
3.5.3 System Activity Diagram	42
3.6 Skenario Pengujian	51
3.6.1 Response Time.....	51
3.6.2 Throughput.....	52
BAB IV HASIL DAN PEMBAHASAN	53
4.1 Hasil Pembuatan Sistem dan Pengujian Black Box	53
4.2 Hasil Pengujian pada Web	59
4.2.1 Response Time	59
4.2.2 Throughput	60
BAB V PENUTUP	62
5.1 Kesimpulan	62
5.2 Saran	62
DAFTAR PUSTAKA	63
LAMPIRAN	65

DAFTAR GAMBAR

Nomor	halaman
Gambar 2.1 Arsitektur website statis	9
Gambar 2.2 Arsitektur web dinamis	10
Gambar 2.3 Contoh aplikasi web framework	10
Gambar 2.4 Konsep sistem basis data	18
Gambar 2.5 Model MVC dan Interaksinya	25
Gambar 3.1 Diagram tahapan penelitian	35
Gambar 3.2 Use case diagram system.....	37
Gambar 3.3 Skema Database Sistem	38
Gambar 3.4 Activity diagram saat menampilkan data periode pada Dashboard	42
Gambar 3.5 Activity diagram saat menampilkan data tahunan pada Dashboard	43
Gambar 3.6 Activity diagram saat menampilkan detail pertahun pada Dashboard	44
Gambar 3.7 Activity diagram saat proses download rekap pada Dashboard	45
Gambar 3.8 Activity diagram membuat questioner pada halaman kelola quistioner ...	45
Gambar 3.9 Activity diagram mengedit questioner pada halaman kelola quistioner ...	46
Gambar 3.10 Activity diagram menghapus questioner pada halaman kelola questioner	47
Gambar 3.11 Activity diagram membuat pertanyaan di halaman kelola quistioner ...	47
Gambar 3.12 Activity diagram mengedit pertanyaan di halaman kelola quistioner ...	48
Gambar 3.13 Activity diagram menghapus pertanyaan di halaman kelola questioner	49
Gambar 3.14 Activity diagram menjawab questioner di halaman jawab quistioner ...	50
Gambar 3.15 Activity diagram saat mengimport data/menambah user	51
Gambar 4.1 Halaman dashboard saat menampilkan data sesuai request user	53
Gambar 4.2 Halaman dashboard saat proses download rekap responden	54
Gambar 4.3 Halaman kelola quistioner	55
Gambar 4.4 Halaman saat edit quistioner	55
Gambar 4.5 Halaman saat memilih untuk menghapus sebuah quistioner	56
Gambar 4.6 Halaman saat edit pertanyaan	57
Gambar 4.7 Halaman saat memilih untuk menghapus sebuah pertanyaan	57
Gambar 4.8 Halaman saat memilih questioner untuk di jawab	58
Gambar 4.9 Halaman saat mengimport data/tambah user	59
Gambar 4.10 Grafik hasil pengujian waktu rata-rata	60
Gambar 4.11 Grafik hasil throuhput rata-rata	61

DAFTAR TABEL

Nomor	halaman
Tabel 4.1 Pengujian black box saat menampilkan data pada Dashboard	53
Tabel 4.2 Pengujian black box saat proses Download Rekap Responden	54
Tabel 4.3 Pengujian black box menambahkan dan mengedit questioner	55
Tabel 4.4 Pengujian black box hapus questioner	56
Tabel 4.5 Pengujian black box menambahkan dan mengedit pertanyaan	56
Tabel 4.6 Pengujian black box hapus pertanyaan	57
Tabel 4.7 Pengujian black box jawab questioner	58
Tabel 4.8 Pengujian black box import data/tambah role	58
Tabel 4.9 Hasil pengujian Avarage Secara Online pada Jaringan 4G	59
Tabel 4.10 Hasil pengujian Thraughput pada web online	60

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan semakin cepatnya perkembangan teknologi informasi maka teknologi informasi telah merambah ke segala bidang dalam kehidupan manusia, termasuk dalam pengelolaan perguruan tinggi. Keberhasilan perguruan tinggi (PT) dalam mencapai visi dan misinya, salah satunya dipengaruhi oleh kemampuan PT memanfaatkan teknologi ini dalam menilai kepuasan pemangku kepentingan terhadap layanan kinerja yang diberikan. Dalam konsep pelayanan, kepuasan pemangku kepentingan menjadi sangat penting sehingga dipandang sebagai salah satu alat untuk mencapai keunggulan kompetitif perguruan tinggi tersebut, terutama pada pengembangan software berbasis teknologi website (Mustamiin, M, dkk, 2020).

Hal ini didukung pula oleh adanya kriteria baru pada akreditasi program studi yang menitik beratkan pada luaran dan capaian tridharma perguruan tinggi, khususnya pada point 2 yaitu penelusuran lulusan, umpan balik dari pengguna lulusan, dan persepsi publik terhadap lulusan sesuai dengan capaian pembelajaran lulusan yang ditetapkan oleh Program Studi dan Perguruan Tinggi dengan mengacu pada KKNI (Permendikbud, 2020).

Kualitas pelayanan memiliki beberapa dimensi atau unsur kualitas pelayanan yaitu *Tangibles*, *Reliability*, *Responsiveness*, *Assurance* dan *Empathy*. Kelima dimensi kualitas pelayanan ini sejatinya harus benar-benar diterapkan oleh perguruan tinggi dalam memberikan layanan terhadap pemangku kepentingan. Survey kepuasan pemangku kepentingan dilakukan untuk menjangkau tingkat kepuasan terhadap layanan yang diberikan oleh perguruan tinggi. Hasil survey ini dipakai sebagai masukan dalam rangka perbaikan maupun peningkatan kualitas pelayanan perguruan tinggi di masa yang akan datang (Parasuraman, et.al, 1988).

Keberhasilan pelaksanaan dan pengelolaan survey kepuasan pemangku kepentingan tergantung pada keakuratan data dan kecepatan dalam penyediaan informasi sesuai yang dibutuhkan. Untuk mendukung hal tersebut, maka diperlukan sebuah sistem yang dapat mencatat dan sekaligus mengolah data hasil umpan balik kepuasan pemangku kepentingan. Sistem ini akan dibangun dengan menggunakan bahasa pemrograman PHP dengan menggunakan arsitektur Model View Controller, salah satunya yaitu framework Laravel (framework PHP) dengan mengacu pada MySQL sebagai DBMS (Database Management System). Framework Laravel untuk

pengembangan PHP ini memiliki banyak manfaat dan merupakan salah satu perkembangan terpenting dalam desain dan praktik pengembangan selama beberapa tahun terakhir (Sutiono, 2019).

Beberapa alasan memilih Framework Laravel untuk pengembangan PHP, terutama karena beberapa manfaatnya yaitu website menjadi lebih *scalable* (mudah dikembangkan dengan bantuan beberapa fitur unggulan, seperti *Template Engine*, *Routing*, dan *Modularity*), terdapat *namespace* dan tampilan yang membantu untuk mengorganisir dan mengatur sumber daya website dan proses pengembangan menjadi lebih cepat sehingga menghemat waktu karena Laravel dapat dikombinasikan dengan beberapa komponen dari framework lain untuk mengembangkan website. Selain itu Laravel mengubah pengembangan website menjadi lebih elegan, ekspresif, dan menyenangkan, sesuai dengan jargonnya “*The PHP Framework For Web Artisans*” (<https://www.niagahoster.co.id/blog/laravel>).

Berdasarkan hal tersebut di atas, maka dirancang suatu penelitian berjudul “*Rancang Bangun Sistem Penjaringan Umpan Balik Tingkat Kepuasan Pemangku Kepentingan Menggunakan Arsitektur Model View Controller*”

1.2 Rumusan Masalah dan Batasan Masalah

Rumusan Masalah:

- Bagaimana cara menggunakan arsitektur model view controller dalam membangun sebuah sistem informasi?
- Bagaimana cara mengembangkan sistem penjaringan umpan balik tingkat kepuasan pemangku kepentingan di perguruan tinggi menggunakan arsitektur model view controller?
- Bagaimana kinerja dari website yang dibangun menggunakan arsitektur model view controller?

Batasan Masalah:

- Framework yang digunakan adalah framework laravel versi 8
- Database yang digunakan adalah Mysql
- Framework front end yang digunakan adalah Bootstrap 4
- Untuk system login mahasiswa dan dosen menggunakan service (API) login dari Neosia (Mahasiswa) dan Apps (Dosen)
- Jenis kuesioner yang digunakan pada sistem adalah Multiple Choice.

1.3 Tujuan Penelitian

- Untuk mengetahui bagaimana cara menggunakan arsitektur model view controller dalam membangun sebuah sistem informasi.
- Untuk mengetahui bagaimana cara mengembangkan sistem penjarangan umpan balik tingkat kepuasan pemangku kepentingan di perguruan tinggi menggunakan arsitektur model view controller.
- Untuk mengetahui kinerja dari website yang dibangun menggunakan arsitektur model view controller.

1.4 Manfaat penelitian

Manfaat dari penelitian ini adalah mempermudah pimpinan perguruan tinggi dalam mengambil keputusan saat melakukan evaluasi kepuasan pemangku kepentingan.

1.5 Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Pada bab ini akan dijelaskan teori-teori yang menunjang percobaan yang dilakukan.

BAB III METODOLOGI PENELITIAN

Bab ini berisi analisis kebutuhan sistem, perancangan sistem, dan skenario pengujian.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi hasil penelitian dan pembahasan penjabaran dari penelitian yang dilakukan.

BAB V PENUTUP

Bab ini berisi kesimpulan dari hasil penelitian dan saran.

BAB II

TINJAUAN PUSTAKA

2.1 Sistem Informasi

2.1.1. Konsep Dasar Sistem

Sistem menurut Sutabri (2012:10) bahwa “Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain dan terpadu”. Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.

1. Karakteristik Sistem

Model umum sebuah sistem terdiri dari input, proses dan output. Hal ini merupakan konsep sebuah sistem yang sangat sederhana mengingat sebuah sistem dapat mempunyai beberapa masukan dan keluaran sekaligus. Sistem mempunyai karakteristik atau sifat-sifat tertentu menurut Sutabri (2012:13), yaitu:

- a. **Komponen Sistem (Components)** Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem.
- b. **Batasan Sistem (Boundary)** Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.
- c. **Lingkungan Luar Sistem (Environment)** Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem.
- d. **Penghubung Sistem (Interface)** Sebagai media yang menghubungkan sistem dengan sub sistem yang lain disebut dengan penghubung sistem atau interface. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lain.
- e. **Masukan Sistem (Input)** Energi yang dimasukkan kedalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (maintenance input) dan sinyal (signal input).

- f. Keluaran Sistem (Output) Merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain.
- g. Pengolahan Sistem (Proses) Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.
- h. Sasaran Sistem (Objective) Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2. Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang di antaranya sebagai berikut:

- a. Sistem Abstrak (Abstract Sistem) dan Sistem Fisik (Physical Sistem)⁹
Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan, sedangkan sistem fisik merupakan sistem yang ada secara fisik, seperti sistem komputer, sistem produksi, sistem penjualan, sistem administrasi personalia, dan lain sebagainya.
- b. Sistem Alamiah (Nature Sistem) dan Sistem Buatan (Human Made Sistem)
Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem peputaran bumi, terjadinya siang malam, pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan human machine system. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.
- c. Sistem Tertentu (Deterministic Sistem) dan Sistem Tak Tent (Probabilistic Sistem)
Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministic. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sedangkan sistem yang bersifat probabilistic adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilitas.

- d. Sistem Tertutup (Closed Sistem) dan Sistem Terbuka (Open Sistem) Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

2.1.2 Pengertian Sistem Informasi

Informasi adalah hasil proses atau hasil pengolahan data melalui gabungan, analisis, penyimpulan dan pengolahan sistem informasi komputerisasi. Menurut Sutabri (2012:29) adalah “Informasi adalah data yang telah diklasifikasikan atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan”. Kualitas Informasi tergantung dari 3 (tiga) hal yaitu:

- a. Akurat (*Accurate*) Informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Akurat juga berarti informasi harus jelas dan mencerminkan maksudnya. Informasi harus akurat karena biasanya dari sumber informasi sampai penerima informasi ada kemungkinan terjadi gangguan (*noise*) yang dapat mengubah atau merusak informasi tersebut.
- b. Tepat Waktu (*Timelines*) Informasi yang datang pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi karena informasi merupakan landasan dalam pengambilan keputusan. Bila pengambilan keputusan terlambat maka dapat berakibat fatal bagi organisasi. Dewasa ini, mahalnya informasi disebabkan karena harus cepatnya informasi tersebut dikirim atau didapat sehingga diperlukan teknologi mutakhir untuk mendapat, mengolah, dan mengirimkannya.
- c. Relevan (*Relevance*) Informasi tersebut mempunyai manfaat untuk pemakaiannya. Relevansi informasi untuk orang satu dengan yang lain berbeda, misalnya informasi sebab musabab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan apabila ditunjukkan kepada ahli teknik perusahaan. Sebaliknya, informasi mengenai harga pokok produksi untuk ahli teknik merupakan informasi yang kurang relevan, tetapi akan relevan untuk seorang akuntan perusahaan.

Menurut Sutabri (2012:38) “Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang

mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu”. Sistem Informasi merupakan komponen-komponen dari subsistem yang saling berhubungan dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi. Sistem informasi terdiri dari komponen-komponen yang disebutnya dengan istilah blok bangunan (building block), yaitu blok masukan (input blok), blok model (model block), blok keluaran (output block), blok teknologi (technology block), blok basis data (database block) dan blok kendali (controls blok). Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasarnya.

2.1.3 Pengujian Sistem

Pengujian sistem dapat dilakukan dengan ujicoba whitebox dan blackbox. Ujicoba blackbox memfokuskan pada keperluan fungsional dari software. Oleh karena itu uji coba blackbox memungkinkan pengembang software untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program. Ujicoba blackbox bukan merupakan alternatif dari ujicoba whitebox, tetapi merupakan pendekatan yang melengkapi untuk menemukan kesalahan lainnya, selain menggunakan metode whitebox.

Menurut Rizky (2011:261) “White box testing secara umum merupakan jenis testing yang lebih berkonsentrasi terhadap “isi” dari perangkat lunak itu sendiri. Jenis ini banyak berkonsentrasi kepada source code dari perangkat lunak yang dibuat sehingga membutuhkan proses testing yang jauh lebih lama dan lebih “mahal” dikarenakan membutuhkan ketelitian dari para testes serta kemampuan teknis pemrograman bagi para testernya”. “Black Box Testing adalah tipe testing yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya. Sehingga para tester memandang perangkat lunak seperti layaknya sebuah “Kotak Hitam” yang tidak penting dilihat isinya, tapi cukup dikenai proses testing dibagian luar”.

Beberapa keuntungan yang diperoleh dari jenis testing ini antara lain (a) Anggota tester tidak harus dari seseorang yang memiliki kemampuan teknis dibidang pemograman, (b) Kesalahan dari perangkat lunak ataupun bug seringkali ditemukan oleh komponen tester yang berasal dari pengguna, (c) Hasil dari black box testing dapat memperjelas kontradiksi ataupun keracunan yang mungkin timbul dari eksekusi

sebuah perangkat lunak dan (d) Proses testing dapat dilakukan lebih cepat dibandingkan white box testing.

2.2 Pengertian Website

Website atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman. Hubungan antara satu halaman web dengan halaman web yang lainnya disebut hyperlink, sedangkan teks yang dijadikan media penghubung disebut hypertext (Batubara, 2015).

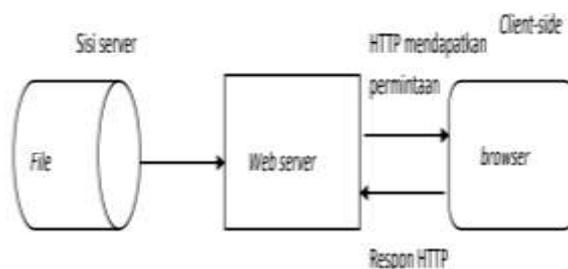
Dalam beberapa dekade, website telah menjadi lebih besar dan kompleks. Pada awalnya website digunakan hanya untuk mempermudah tukar menukar dan melakukan perbaruan informasi kepada sesama peneliti yang dilakukan oleh Sir Tomothy Jon. Namun dalam perkembangannya website dapat melakukan manajemen konten seperti video dan gambar (Butkiewicz, 2011).

Secara teknis, web adalah sebuah sistem dimana informasi dalam bentuk teks, gambar, suara, dan lain-lain yang tersimpan dalam sebuah internet web server dipresentasikan dalam bentuk hypertext. Informasi lainnya disajikan dalam bentuk grafis (dalam format GIF, JPG, PNG), suara (dalam format AU, WAV), dan objek multimedia lainnya (seperti MIDI, Shockwave, Quicktime Movie, 3D World). Web dapat diakses oleh perangkat lunak web client yang secara populer disebut sebagai browser. Browser membaca halaman-halaman yang tersimpan dalam webserver melalui protokol yang disebut dengan HTTP (Hypertext Transfer Protocol). Sebagai dokumen hypertext, dokumen-dokumen di web dapat memiliki link dengan dokumen lain, baik yang tersimpan dalam web server yang sama maupun di web server lainnya. Link memudahkan para pengakses web berpindah dari satu halaman ke halaman lainnya, dan “berkelana” dari satu server ke server lain. Kegiatan penelusuran halaman web ini biasa diistilahkan sebagai browsing, ada juga yang menyebutnya sebagai surfing (berselancar). Seiring dengan semakin berkembangnya jaringan internet di seluruh dunia, maka jumlah situs web yang tersedia juga semakin meningkat. Hingga saat ini, jumlah halaman web yang bisa diakses melalui internet telah mencapai angka miliaran. Untuk memudahkan penelusuran halaman web,

terutama untuk menemukan halaman yang memuat topik-topik yang spesifik, maka para pengakses web dapat menggunakan suatu mesin pencari (search engine). Penelusuran berdasarkan search engine dilakukan berdasarkan kata kunci (keyword) yang kemudian akan dicocokkan oleh search engine dengan basis data miliknya (Batubara, 2015).

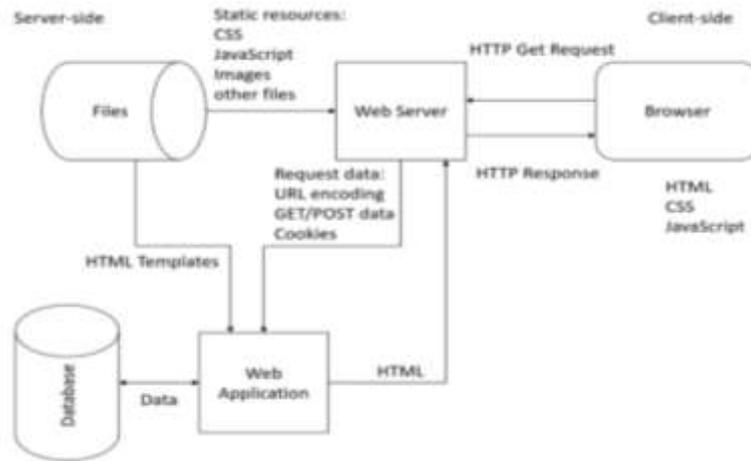
2.3 Website Modern

Saat ini, ada banyak cara dalam melakukan pengembangan dan pembuatan aplikasi web. Namun pengembangan website dimulai dengan memahami bagaimana arsitektur web yang akan digunakan, web statik atau dinamis dan juga penentuan pengembangan tool dan service yang digunakan. Sebuah website statis terdiri atas beberapa halaman HTML, CSS yang secara bersamaan saling terhubung oleh hyperlinks. Website dinamis memiliki lokasi penyimpanan konten pada basis data dan ditampilkan berdasarkan permintaan pengguna. Namun HTML dan CSS juga dapat digunakan dalam website dinamis ketika memiliki javascript yang mengandung pemrograman back-end. Website modern dibangun menggunakan pemrograman front-end yang dieksekusi oleh browser dan beralan pada sisi client yaitu HTML, CSS, dan pemrograman back-end yang dieksekusi pada sisi server yaitu havascript, PHP, Python dan yang lainnya yang sering digunakan oleh pengembangan web. Pemrograman front-end berhubungan dengan apa yang pengguna dapat lihat (Hannonen, 2017).



Gambar 2.1 Arsitektur website statis

Pada website statis ketika pengguna ingin melakukan perpindahan halaman maka browser mengirimkan permintaan HTTP “GET” yang menunjukkan URL. Sisi server File Client-side Web server browser Respon HTTP HTTP mendapatkan permintaan Server kemudian mengirimkan dokumen permintaan dari sistem file menuju browser pengguna.



Gambar 2.2 Arsitektur web dinamis

Dari kedua gambar yaitu gambar 2.1 dan gambar 2.2, masing-masing arsitektur memiliki kekurangan dan kelebihan. Website statis memiliki kelebihan yaitu sederhana dan cepat dalam melakukan proses pengembangan. Namun kekurangannya adalah pengembang harus melakukan perubahan kode baik HTML maupun CSS setiap konten mengalami perubahan. Berbeda dengan website dinamis dimana memiliki kemudahan dalam melakukan perubahan konten karena bekerja dengan data yang dinamis. Namun dalam mengembangkan arsitektur web dinamis dibutuhkan lebih dari satu pengembang untuk memastikan web yang dibuat berjalan dengan semestinya dan memiliki desain yang menarik.

Seiring perkembangannya, hari ini pengembangan web memiliki cara pengembangan yang lebih mudah dan disukai oleh pengembang, yaitu dengan menggunakan framework dalam melakukan pengembangan web. Framework memudahkan para pengembang karena kemampuannya menyediakan manajemen pengerjaan web yaitu dengan memisahkan bentuk pengerjaan dari sisi front-end dan back-end.



Gambar 2.3 Contoh aplikasi web framework

Framework sisi server adalah framework back-end yang memudahkan pengembang dalam melakukan penulisan dan pemeliharaan aplikasi web. Framework menyediakan pustaka yang sederhana yang sering digunakan dalam membangun sebuah sistem web, termasuk URL routing, interaksi dengan basis data, mendukung manajemen session maupun otorisasi pengguna (user authorizaion), manajemen masukan (HTML, JSON, XML), dan peningkatan keamanan terhadap serangan web (Hannonen, 2017).

Sedangkan front-end Framework sangat membantu dalam melakukan desain proses dan masukan dari berbagai fitur yang ada, termasuk fitur kanvas, menu, dan model yang ada. Framework front-end dibagi menjadi dua yaitu javascript framework (Angular, ReactJS, MeterJS) dan CSS framework (Bootstrap, Foundation, Sematic UI, Ulkit, MaterialUI). (Hannonen, 2017).

2.4 PHP

2.4.1 Pengertian PHP

PHP merupakan singkatan dari “PHP: Hypertext Preprocessor”, adalah sebuah bahasa scripting yang terpasang pada HTML. Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamik dengan cepat.

2.4.2 Hubungan PHP dengan 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 di antara tanda <? dan ?>. Tanda-tanda tersebut biasanya disebut tanda untuk *escaping* (kabur) dari kode html. File html yang telah dibubuhi program php harus diganti ekstensi-nya menjadi *.php* atau *.php3*.

PHP merupakan bahasa pemrograman 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).

PHP pertama kali dibuat oleh Rasmus Lerdroft, seorang programmer C. Semula PHP digunakannya untuk menghitung jumlah pengunjung di dalam webnya. Kemudian ia mengeluarkan *Personal Home Page Tools* versi 1.0 secara gratis. Versi ini pertama kali keluar pada tahun 1995. Isinya adalah sekumpulan script PERL yang dibuatnya untuk membuat halaman webnya menjadi dinamis. Kemudian pada tahun 1996 ia mengeluarkan PHP versi 2.0 yang kemampuannya telah mampu mengakses database dan dapat terintegrasi dengan HTML.

Pada tahun 1998 tepatnya pada tanggal 6 Juni 1998 keluarlah PHP versi 3.0 yang dikeluarkan oleh Rasmus sendiri bersama kelompok pengembang softwarena. Versi teranyar yaitu PHP 4.0 keluar pada tanggal 22 Mei 2000 merupakan versi yang lebih lengkap lagi dibandingkan dengan versi sebelumnya. Perubahan yang paling mendasar pada PHP 4.0 adalah terintegrasinya Zend Engine yang dibuat oleh Zend Suraski dan Andi Gutmans yang merupakan penyempurnaan dari PHP3 scripting engine. Yang lainnya adalah build in HTTP session, tidak lagi menggunakan library tambahan seperti pada PHP3.

Tujuan dari bahasa scripting ini adalah untuk membuat aplikasi-aplikasi yang dijalankan di atas teknologi web. Dalam hal ini, aplikasi pada umumnya akan memberikan hasil pada web browser, tetapi prosesnya secara keseluruhan dijalankan web server.

2.4.3 Kelebihan PHP

Ketika e-commerce semakin berkembang, situs-situs yang statispun semakin ditinggalkan Karena dianggap sudah tidak memenuhi keinginan pasar karena situs tersebut harus tetap dinamis selama setiap hari. Pada saat ini bahasa PERL dan CGI sudah jauh ketinggalan jaman sehingga sebagian besar designer web banyak beralih ke bahasa server-side scripting yang lebih dinamis seperti PHP.

Seluruh aplikasi berbasis web dapat dibuat dengan PHP. Namun kekuatan yang paling utama PHP adalah pada konektivitasnya dengan system database di dalam web. Sistem database yang dapat didukung oleh PHP adalah :

1. Oracle
2. MySQL
3. Sybase
4. PostgreSQL
5. dan lainnya

PHP dapat berjalan di berbagai system operasi seperti windows 98/NT, UNIX/LINUX, solaris maupun macintosh. PHP merupakan software yang *open source* yang dapat anda download secara gratis dari situs resminya yaitu <http://www.php.net>. Software ini juga dapat berjalan pada web server seperti PWS (Personal Web Server), Apache, IIS, AOLServer, fhttpd, phttpd dan sebagainya. PHP juga merupakan bahasa pemrograman yang dapat kita kembangkan sendiri seperti menambah fungsi-fungsi baru. Keunggulan lainnya dari PHP adalah PHP juga mendukung komunikasi dengan layanan seperti protocol IMAP, SNMP, NNTP, POP3 bahkan HTTP.

PHP dapat diinstal sebagai bagian atau modul dari apache web server atau sebagai CGI script yang mandiri. Banyak keuntungan yang dapat diperoleh jika menggunakan PHP sebagai modul dari apache di antaranya adalah :

1. Tingkat keamanan yang cukup tinggi
2. waktu eksekusi yang lebih cepat dibandingkan dengan bahasa pemrograman web lainnya yang berorientasi pada server-side scripting.
3. Akses ke system database yang lebih fleksibel. seperti MySQL.

Dalam modul ini kita akan mempelajari PHP sebagai server-side scripting yang menggunakan apache sebagai webserver. Versi PHP yang kita gunakan adalah PHP4 untuk windows.

PHP merupakan singkatan dari PHP Hypertext Preprocessor. PHP digunakan sebagai bahasa script server side dalam pengembangan web yang disisipkan pada dokumen HTML. Penggunaan PHP memungkinkan web dapat dibuat dinamis sehingga maintenance web menjadi lebih mudah dan efisien. PHP ditulis menggunakan bahasa C. PHP memiliki banyak kelebihan yang tidak dimiliki oleh bahasa scripting lainnya. PHP difokuskan pada pembuatan script server side yang bisa melakukan apa saja yang dilakukan oleh CGI, seperti mengumpulkan data dari form, menghasilkan isi halaman web dinamis, dan kemampuan mengirim serta menerima cookies, bahkan lebih daripada kemampuan CGI. PHP tidak terbatas pada hasil keluaran HTML, namun PHP juga memiliki kemampuan untuk mengolah gambar, file

PDF, dan movie flash. PHP juga dapat menghasilkan teks seperti XHTML dan file XML lainnya.

Salah satu fitur yang dapat diandalkan oleh PHP adalah dukungan terhadap database, salah satunya adalah MySQL. PHP hanya mengeksekusi kode yang ditulis dalam pembatas sebagaimana ditentukan oleh dasar sintaks PHP. Apapun di luar pembatas tidak diproses oleh PHP (meskipun teks PHP ini masih mengendalikan struktur yang dijelaskan dalam kode PHP). Pembatas yang paling umum adalah "

Beberapa keunggulan bahasa pemrograman PHP diantaranya:

- a. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. Web Server yang mendukung PHP dapat ditemukan di mana - mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.
- c. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
- d. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.
- e. PHP adalah bahasa open-source yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.

2.5 Composer

Composer adalah tools dependency manager pada PHP, Dependency (ketergantungan) diartikan ketika project PHP yang dikerjakan masih membutuhkan atau memerlukan library dari luar. Composer berfungsi sebagai penghubung antara project PHP dengan library dari luar. Composer adalah tools dependency manager multiplatform sehingga bisa di install di berbagai sistem operasi. Untuk system requirements tools ini sendiri membutuhkan PHP 5.3.2+ untuk dijalankan. Saat menggunakan installer segala ketidaksesuaian akan diperingatkan, seperti pengaturan PHP sensitif dan kompilasi flag.

2.6 Xampp (Localhost)

2.6.1 Pengertian Xampp

XAMPP adalah perangkat lunak bebas, yang mendukung banyak system operasi, merupakan kompilasi dari beberapa program. XAMPP merupakan tool

yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi web server Apache, PHP dan MySQL secara manual. XAMPP akan menginstallasi dan mengkonfigurasikannya secara otomatis untuk anda atau auto konfigurasi.

XAMPP merupakan salah satu paket instalasi Apache, PHP dan MySQL instant yang dapat kita gunakan untuk membantu proses instalasi ketiga produk tersebut. Selain paket instalasi instant XAMPP versi 1.6.4 juga memberikan fasilitas pilihan penggunaan PHP4 atau PHP5. Untuk berpindah versi PHP yang ingin digunakan juga sangat mudah dilakukan dengan menggunakan bantuan PHP-Switch yang telah disertakan oleh XAMPP, dan yang terpenting XAMPP bersifat free atau gratis untuk digunakan.

XAMPP merupakan tool yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi web server Apache, PHP dan MySQL secara manual. XAMPP akan menginstallasi dan mengkonfigurasikannya secara otomatis. Merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkannya dapat mendownload langsung dari web resminya.

2.6.2 Fungsi XAMPP

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. Nama XAMPP sendiri merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, 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 mendapatkannya XAMPP anda dapat mendownload langsung dari web resminya. Dan berikut beberapa definisi program lainnya yang terdapat dalam XAMPP.

2.6.3 Fitur-Fitur XAMPP

Berikut ini terdapat beberapa fitur-fitur xampp, terdiri atas:

1. Apache

Apache adalah perangkat lunak sumber terbuka yang menjadi alternatif dari server web Netscape. Server HTTP Apache atau Server Web/WWW Apache merupakan server web yang dapat dijalankan di banyak sistem operasi yang berguna untuk melayani dan memfungsikan situs web. Apache dikembangkan oleh komunitas terbuka yang terdiri dari pengembang-pengembang dibawah naungan Apache Software Foundation.

2. MySQL

MySQL adalah singkatan “My Structured Query Language”. Program ini berjalan sebagai server menyediakan multi-user mengakses ke sejumlah database. MySQL umumnya digunakan oleh perangkat lunak bebas yang memerlukan fitur penuh sistem manajemen database, seperti WordPress, phpBB dan perangkat lunak lain yang dibangun pada perangkat lunak LAMP. Ia juga digunakan dalam skala sangat tinggi World Wide Web, termasuk produk-produk Google dan Facebook.

3. PHP

PHP adalah bahasa pemrograman script yang banyak dipakai untuk memrogram situs web dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain. Contoh terkenal dari aplikasi PHP adalah phpBB dan MediaWiki (software di belakang Wikipedia). Contoh terkenal dari aplikasi PHP adalah phpBB dan MediaWiki (software di belakang Wikipedia).

2.6.4 Bagian-Bagian XAMPP

Berikut ini terdapat beberapa bagian-bagian XAMPP, terdiri atas:

a. Htdocs

Htdocs adalah sebuah folder yang digunakan sebagai tempat penyimpanan berkas seperti PHP, HTML, dan script lain yang digunakan dalam sebuah halaman website. Secara kapasitas penyimpanan, XAMPP tergantung dari seberapa besar kapasitas hardisk di laptop atau komputer anda. Sedangkan bila menggunakan hosting online, maka tergantung pilihan waktu membeli sebuah hosting.

b. phpMyadmin

phpMyadmin adalah sebuah tempat yang digunakan untuk mengelola database MySQL yang berada di komputer atau laptop. Untuk mengakses

phpMyadmin yakni dengan membuka browser internet (Mozilla atau chrome) lalu ketikkan alamat <http://localhost/phpMyadmin> maka akan muncul tampilannya.

c. Control Panel

Control Panel adalah sebuah layanan untuk mengelola XAMPP baik itu mengontrol (start atau stop XAMPP) serta layanan service lainnya. Secara online di dalam hosting atau VPS dikenal CPanel.

2.6.5 Kelebihan dan Kekurangan XAMPP

Berikut ini terdapat beberapa kelebihan dan kekurangan XAMPP, terdiri atas:

Kelebihan :

1. Database Storage Engine ini banyak digunakan oleh programmer apalagi oleh web developer karena sifatnya yang free. Untuk yang expert sudah ada yang bayar.
2. 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.
3. Keamanan datanya cukup aman walaupun tidak sehebat Postgre apalagi Oracle.
4. Engine ini multiplatform sehingga mampu diaplikasikan di berbagai sistem operasi. My Sql cocok diaplikasikan diaplikasi kelas kecil dan menengah.
5. Kelebihan paling utama engine ini adalah kecepatannya.

Kekurangan :

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

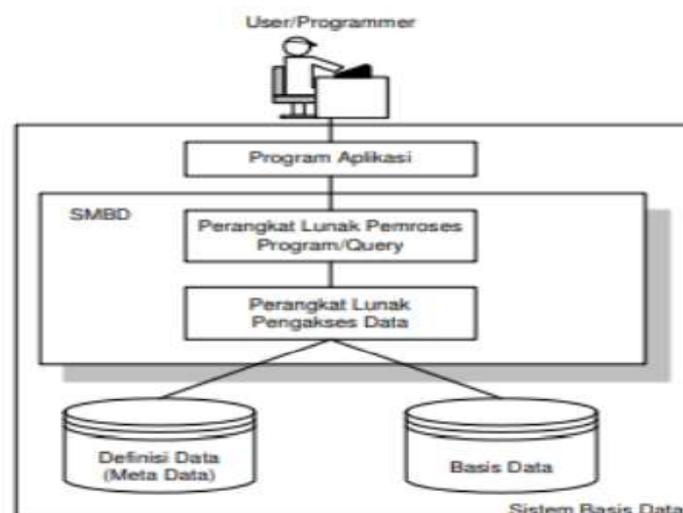
2.7 Basis Data

Data merupakan fakta mengenai suatu objek seperti manusia, benda, peristiwa, konsep, keadaan dan sebagainya yang dapat dicatat dan mempunyai arti secara implisit. Data dapat dinyatakan dalam bentuk angka, karakter atau simbol, sehingga bila data dikumpulkan dan saling berhubungan maka dikenal dengan istilah basis data (database). Sedangkan menurut George Tsu-der Chou basis data merupakan

kumpulan informasi bermanfaat yang diorganisasikan ke dalam aturan yang khusus. Informasi ini adalah data yang telah diorganisasikan ke dalam bentuk yang sesuai dengan kebutuhan seseorang. Menurut Encyclopedia of Computer Science and Engineer, para ilmuwan di bidang informasi menerima definisi standar informasi yaitu data yang digunakan dalam pengambilan keputusan. Definisi lain dari basis data menurut Fabbri dan Schwab adalah sistem berkas terpadu yang dirancang terutama untuk meminimalkan duplikasi data. Menurut Ramez Elmasri mendefinisikan basis data lebih dibatasi pada arti implisit yang khusus, yaitu (Haidar Dzacko, 2007):

- a. Basis data merupakan penyajian suatu aspek dari dunia nyata (real world)
- b. Basis data merupakan kumpulan data dari berbagai sumber yang secara logika mempunyai arti implisit. Sehingga data yang terkumpul secara acak dan tanpa mempunyai arti, tidak dapat disebut basis data
- c. Basis data perlu dirancang, dibangun dan data dikumpulkan untuk suatu tujuan. Basis data dapat digunakan oleh beberapa user dan beberapa aplikasi yang sesuai dengan kepentingan user.

Dari beberapa deifinisi-definisi tersebut, dapat dikatakan bahwa basis data mempunyai berbagai sumber data dalam pengumpulan data, bervariasi derajat interaksi kejadian dari dunia nyata, dirancang dan dibangun agar dapat digunakan oleh beberapa pengguna untuk berbagai kepentingan.



Gambar 2.4 Konsep sistem basis data

C.J Date menyatakan bahwa sistem basis data dianggap sebagai tempat untuk sekumpulan berkas data yang terkomputerisasi dengan tujuan untuk memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan (Haidar Dzacko 2007).

Basis data yang membutuhkan sebuah media untuk melakukan penyimpanan dan pengelolaan data yang ada diolah oleh program sql maupun mysql. Keduanya adalah media yang digunakan untuk melakukan pengolahan basis data yang digunakan sesuai dengan kebutuhan pengembang.

2.7.1 NoSQL Basis Data

Istilah NoSQL diciptakan oleh Carlo Strozzi pada tahun 1998 dan mengacu pada basis data non-relasional, pada tahun 2009 Eric Evans memperkenalkan kembali istilah NoSQL. Baru-baru ini, istilah ini memiliki makna lain, yaitu “Not Only SQL”, istilah yang lebih baik dari sebelumnya yang lebih dikenal dengan anti relasional. NoSQL mengakomodasi data yang tidak terstruktur, kehadirannya bukan untuk menggantikan SQL namun kedua teknologi ini dapat saling berdampingan. Perbedaan utama kedua basis data ini adalah SQL memiliki skema yang kaku sementara basis data NoSQL menawarkan desain yang fleksibel yang dapat diubah tanpa downtime atau gangguan lainnya. NoSQL juga dirancang untuk menyimpan data yang dapat didistribusikan untuk kebutuhan data dalam skala besar; misalnya Facebook memiliki 500 juta pengguna dan Twitter terakumulasi terabyte data, basis data NoSQL telah memiliki popularitas yang tinggi, sehingga basis data ini diklaim lebih baik dari basis data SQL. Basis data NoSQL dimotivasi oleh kesederhanaan, horizontal scaling dan kontrol yang lebih dalam ketersediaan data. NoSQL menjadi solusi dalam penanganan data dalam jumlah besar yang berkembang pesat saat ini. Data ini biasanya non-terstruktur, kompleks dan tidak cocok digunakan dalam model relasional. Contoh data yang bisa kita rasakan adalah data yang berasal dari smartphone yang mencatat lokasi broadcast setiap saat, video dan kamera bahkan halaman website yang berisi banyak informasi serta dokumen (No dan Junaidi 2016).

Teknologi NoSQL memiliki keunggulan sendiri dalam melakukan manajemen basis data sehingga penggunaan jenis basis data ini lebih diminati oleh pengembang. Ada banyak jenis basis data NoSQL yang sering digunakan oleh pengembang sesuai dengan kebutuhan, seperti IndexedDB, RXDB, PouchDB, dan CouchDB. Kelima

basis data tersebut memiliki keunggulan yaitu mampu menjadi basis data yang memungkinkan pengembang untuk membuat sebuah media penyimpanan offline bagi sistem yang bekerja, sehingga cocok digunakan pada sebuah aplikasi yang bekerja online maupun offline.

Sistem berbasis NoSQL biasanya digunakan dalam database yang sangat besar, yang sangat rentan terhadap masalah kinerja yang disebabkan oleh keterbatasan SQL dan model relasional dari database.

2.7.1.1 MongoDB

MongoDB adalah salah satu software NoSQL yang termasuk dalam kategori Document Store / Document-Oriented Database, yaitu data disimpan dalam bentuk dokumen. Suatu dokumen bisa diibaratkan seperti suatu record dalam basis data relasional dan isi dari masing-masing dokumen tersebut bisa berbeda-beda dan ada pula yang sama. Hal ini berbeda dengan basis data relasional yang menetapkan keseragaman kolom serta tipe data dengan data yang NULL jika tidak terdapat data. MongoDB menyimpan data dalam bentuk dokumen dengan menggunakan format JSON (javascript object notation). Konse dasar yang harus dipahami dalam MongoDB sebagai document-oriented database adalah documents dan collections. Sama halnya dengan basis data relasional, MongoDB menyimpan data dalam suatu basis data. Di dalam basis data tersebut terdapat collections yang bisa diibaratkan seperti tabel dalam basis data relasional, documents adalah records.

Pembuatan database pada MongoDB sangat berbeda dengan pembuatan database di SQL. Dalam SQL Kita harus membuat database terlebih dahulu sebelum melakukan perintah 'use' dengan menggunakan perintah create database dbs_name. Dalam MongoDB tidak perlu menggunakan perintah create Kita bisa langsung menggunakan perintah use meskipun Kita tahu bahwa database yang dimaksud belum ada dalam MongoDB, namun MongoDB mengizinkan Kita untuk melakukan hal itu. Tetapi saat keluar dari database tersebut, maka database akan hilang, hal ini dikarenakan Kita tidak membuat collections dan mengisikan documents dalam collections.

Seguin mengatakan bahwa ada enam konsep dasar yang perlu diketahui mengenai MongoDB, yaitu sebagai berikut :

- a. MongoDB memiliki konsep yang sama dengan database pada umumnya seperti MySQL dan Windows SQL Server. MongoDB dapat memiliki nol atau lebih database
- b. Sebuah database dapat memiliki nol atau lebih collections yang dapat disetarakan dengan table pada database umumnya
- c. Sebuah collections terdiri dari nol atau lebih dokumen yang dapat disetarakan dengan basis pada database umumnya
- d. Sebuah dokumen terdiri dari satu atau lebih fields yang dapat disertakan dengan kolom pada database umumnya
- e. MongoDB memiliki indeks yang memiliki fungsi utama seperti indeks pada database umumnya
- f. Data dari MongoDB akan dikembalikan dalam bentuk kursor.

MongoDB bersifat client-server. Sisi server sebagai tempat proses data dan untuk penulisan instruksi pada sisi client. Pengaksesan database dilakukan dengan instruksi `use database_name` seperti `use learn`. Instruksi tersebut akan mengakses database `learn` yang sudah ada atau membuat baru jika belum ada ketika proses insert pertama. Instruksi yang akan dieksekusi pada database dieksekusi pada collection dituliskan dengan menggunakan objek `db` seperti `db.help()` atau `db.stats()`. Instruksi yang akan dieksekusi pada collection dituliskan dengan menggunakan objek `db.collection_name` seperti `db.unicorns.help()` atau `db.unicorns.count()`.

MongoDB tidak menggunakan JOIN seperti yang biasa dimiliki oleh database pada umumnya, tetapi dapat tetap dihubungkan secara manual seperti penggunaan foreign key di SQL. Langkah yang dapat dilakukan adalah dengan menyimpan `_id` dari satu dokumen dalam dokumen yang lain. Pilihan lainnya adalah dengan menggunakan DBRef yang sudah disediakan oleh MongoDB. DBRef dapat digunakan untuk menghubungkan antara lebih dari satu dokumen dalam collection atau database yang berbeda. DBRef memiliki beberapa field sebagai berikut :

- a. `$ref` – field `$ref` akan berisi nama collection dimana dokumen yang akan dihubungkan berada
- b. `$id` – field `$id` akan berisi nilai `_id` dari dokumen yang akan dihubungkan
- c. `$db` – field `$db` bersifat optional dan akan berisi nama database dimana dokumen yang akan dihubungkan berada.

Dalam MongoDB juga disediakan beberapa fitur tambahan lagi yang mendukung performa. Berikut ini akan dibahas beberapa hal yang akan sering digunakan pada database MongoDB (Yovita Turnadi 2019).

a. Indexes

Index pada MongoDB berfungsi sama dengan indeks pada database relasional pada umumnya yang membantu performa query dan sorting data. Penambahan indeks dapat dilakukan dengan menggunakan instruksi `dropIndex`. Pembuatan unique inde dapat dilakukan dengan menambahkan parameter kedua. Pengurutan indeks secara ascending ditandai dengan angka satu dan descending ditandai dengan angka -1.

b. Explain

Pada MongoDB dapat dilakukan pengecekan apakah suatu collection dan query menggunakan indeks. Instruksi yang digunakan adalah `explain`. Penggunaan instruksi `db.unicorns.find()` akan menampilkan kursor yang digunakan adalah `BasicCursor` yang menandakan tidak adanya penggunaan indeks. Penggunaan instruksi `db.unicorns.find({name:'pilot'}).explain()` akan menampilkan kursor yang digunakan adalah `BtreeCursor` adanya penggunaan indeks (Yovita Turnadi 2019).

c. Backup dan Restore

Backup dan restore pada MongoDB dapat dilakukan dengan menggunakan `fileexecuteable` yang sudah disediakan oleh MongoDB. Backup dilakukan dengan menjalankan `mongodump` dengan mengikuti pilihan database, collection, dan nama filebackup. Sedangkan untuk restore dengan menjalankan proses `mongorestore` dengan pilihan yang sama. Selain backup dan restore, MongoDB juga memungkinkan untuk proses import dan export `fileexecuteable` yang digunakan adalah `mongoimport` dan `mongoexport` (Yovita Turnadi 2019).

2.8 MySQL

MySQL dikembangkan oleh perusahaan swedia bernama MySQL AB yang pada saat ini bernama Tcx DataKonsult AB sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak tahun 1979. Awalnya Tcx merupakan perusahaan

pengembang software dan konsultan database, dan saat ini MySQL sudah diambil alih oleh Oracle Corp. Kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya sehingga mudah untuk digunakan, kinerja query cepat, dan mencukupi untuk kebutuhan database perusahaan-perusahaan yang berskala kecil sampai menengah, MySQL juga bersifat open source (tidak berbayar) .

Menurut Arief (2011d:152) “MySQL adalah salah satu jenis database server yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang menggunakan database sebagai sumber dan pengolahan datanya”. MySQL merupakan database yang pertama kali didukung oleh bahasa pemrograman script untuk internet (PHP dan Perl). MySQL dan PHP dianggap sebagai pasangan software pembangun aplikasi web yang ideal. MySQL lebih sering digunakan untuk membangun aplikasi berbasis web, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman script PHP. MySQL didistribusikan dengan lisensi open source GPL (General Public License) mulai versi 3.23 pada bulan juni 2000. Software MySQL bisa diunduh melalui website resminya di <http://www.MySQL.org> atau di <http://www.mysql.com>.

2.9 GitLab

GitLab adalah sebuah manajer repositori Git berbasis web dengan fitur wiki dan pelacakan masalah. Gitlab digunakan bersamaan dengan sourcetree. Gitlab digunakan sebagai repasitori dalam proses pengembangan website ini agar manajemen selama Bekerja di rumah (WFH) berjalan dengan lancar.

2.10 Visual Studio Code

Visual Studio Code adalah editor source code yang dikembangkan oleh Microsoft untuk Windows, Linux dan MacOS. Ini termasuk dukungan untuk debugging, GIT Control yang disematkan, penyorotan sintaks, penyelesaian kode cerdas, cuplikan, dan kode refactoring. Hal ini juga dapat disesuaikan, sehingga pengguna dapat mengubah tema editor, shortcut keyboard, dan preferensi. Visual Studio Code gratis dan open-source, meskipun unduhan resmi berada di bawah lisensi proprietary. Kode Visual Studio didasarkan pada Elektron, kerangka kerja yang digunakan untuk menyebarkan aplikasi Node.js

untuk desktop yang berjalan pada Blinklayout. Meskipun menggunakan kerangka Elektron, Visual Studio Code tidak menggunakan Atom dan menggunakan komponen editor yang sama (diberi kode nama "Monaco") yang digunakan dalam Visual Studio Team Services yang sebelumnya disebut Visual Studio Online (Lardinois, 2015).

2.11 MVC

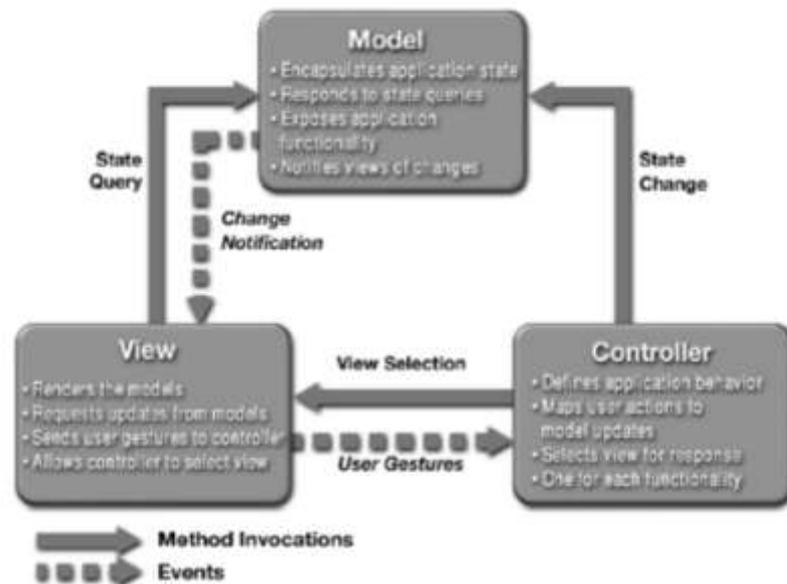
Menurut Daqiqil (2011:5) “MVC adalah singkatan dari Model View Controller. MVC adalah sebuah teknik pemrograman yang memisahkan bisnis logic (alur pikir), data logic (penyimpanan data) dan presentation logic (antarmuka aplikasi) atau secara sederhana adalah memisahkan antara desain (model), data (view) dan proses (controller)”. Saat ini kebanyakan framework dalam aplikasi website adalah berbasis arsitektur MVC. MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, antarmuka pengguna, dan bagian yang menjadi kontrol dalam sebuah aplikasi web.

Arsitektur MVC adalah satu pola yang terbukti dapat dipakai membangun sistem secara lebih efektif. Hal itu dapat dilakukan dengan memilah komponen antara Model, View dan Controller pada bagian – bagian dalam sistem. Aplikasi apapun yang digunakan, maka bagian dalam kode yang sering mengalami perubahan adalah user interface. User interface adalah bagian yang paling sering terlihat oleh user dan bagaimana bagian ini berinteraksi dengan aplikasi, sehingga menjadi titik fokus perubahan untuk kemudahan penggunaan. *Business-logic* yang rumit pada user-interface membuat perubahan pada user interface menjadi lebih kompleks dan mudah terjadi kesalahan. Perubahan pada satu bagian memiliki potensi keterkaitan dengan keseluruhan aplikasi.

Arsitektur MVC secara sederhana dirancang dan diadaptasikan dalam penggunaannya dalam *Web-Application*. Arsitektur yang dihasilkan kemudian disebut dengan Arsitektur Model 2. Aplikasi Model 2 umumnya memiliki (a) Controller Servlet menyediakan akses tunggal terhadap keseluruhan aplikasi. Controller ini bertanggungjawab menyediakan manajemen terpusat terhadap alur aplikasi dan juga service lain seperti penanganan keamanan dan manajemen pengguna (user). (b) Controller Servlet umumnya menggunakan konfigurasi XML untuk menentukan alur aplikasi dan pemrosesan perintah. Hal itu juga membuat komponen pembantu yang berfungsi sebagai obyek perintah. Hal ini berarti komponen pembantu terasosiasikan dengan tindakan pengguna dan dibuat/dipanggil untuk menangani tindakan yang

terjadi, dengan memanggil komponen Model sebagaimana diperlukan. Hal ini berfungsi untuk memisahkan antara controller servlet dari Model (Jeni, Pengenalan MVC).

Pola MVC menyediakan sebuah solusi terhadap permasalahan pada user-interface dengan membagi aplikasi menjadi bagian – bagian tersendiri, Model, View dan Controller, memisahkan antar bagian tersebut dan membuat tata interaksi diantaranya.



Gambar 2.5 Model MVC dan Interaksinya

Diagram di atas menunjukkan 3 komponen yang terdapat dalam pola MVC dan interaksi yang terjadi, yaitu:

a. Model

Model 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 di dalam 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.

Terdapat beberapa kelebihan dalam pendekatan ini. Pertama, dipakai membuat detail data dan operasinya dapat ditempatkan pada area yang ditentukan (Model) dibandingkan tersebar dalam keseluruhan lingkup aplikasi. Hal ini memberikan

keuntungan dalam proses maintenance aplikasi. Kedua, dengan pemisahan total antara data dengan implementasi interface, komponen model dapat digunakan kembali oleh aplikasi lain yang memiliki kegunaan yang hampir sama. Operasi dasar CRUD (Create, Read, Update, Delete) terjadi dalam database, dalam hal ini adalah bagian Model.

b. View

View berhubungan dengan segala sesuatu yang akan ditampilkan ke end-user. Bisa berupa halaman web, rss, javascript dan lain-lain. Sedapat mungkin menghindari adanya logika atau pemrosesan data di view. Di dalam view hanya berisi variabel-variabel yang berisi data yang siap ditampilkan. View dapat dikatakan sebagai halaman website yang dibuat dengan menggunakan HTML dan bantuan CSS atau JavaScript. Di dalam view jangan pernah ada kode untuk melakukan koneksi ke basis data. View hanya dikhususkan untuk menampilkan data-data hasil dari model dan controller.

Layer ini mengandung keseluruhan detail dari implementasi user interface. Disini, komponen grafis menyediakan representasi proses internal aplikasi dan menuntun alur interaksi user terhadap aplikasi. Tidak ada layer lain yang berinteraksi dengan user, hanya View. Penggunaan layer View memiliki beberapa kelebihan yaitu memudahkan penggabungan divisi desain dalam development team. Divisi desain dapat berkonsentrasi pada style, look & feel, dan sebagainya dalam aplikasi tanpa harus memperhatikan lebih pada detail yang lain dan juga, memiliki layer View yang terpisah yang memungkinkan ketersediaan multiple interface dalam aplikasi. Jika inti dari aplikasi terletak pada bagian lain (dalam Model), multiple interface dapat dibuat, secara keseluruhan memiliki tampilan yang berbeda namun mengeksekusi komponen Model sesuai fungsionalitas yang diharapkan.

c. Controller

Controller bertindak sebagai penghubung data dan view. Di dalam controller inilah 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 mengakses data telah diserahkan kepada model. Tugas controller adalah menyediakan berbagai variabel 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.

Layer ini menyediakan detail alur program dan transisi layer, dan juga bertanggungjawab terhadap penampungan events yang dibuat oleh user dari View dan melakukan update terhadap komponen Model menggunakan data yang dimasukkan oleh user. Kelebihan dalam penggunaan layer Controller secara terpisah yaitu: Pertama, dengan menggunakan komponen terpisah untuk menampung detail dari transisi layer, komponen view dapat didesain tanpa harus memperhatikan bagian lain secara berlebih. Hal ini memudahkan team pengembang multiple interface bekerja secara terpisah dari yang lain secara simultan. Interaksi antar komponen View terabstraksi dalam Controller. Kedua, dengan menggunakan layer terpisah yang melakukan update terhadap komponen Model, detail tersebut dapat dihapus dari layer presentasi. Layer presentasi kembali pada fungsi utamanya untuk menampilkan data kepada user. Detail tentang bagaimana data dari user mengubah ketetapan aplikasi yang disembunyikan oleh Controller. Hal ini memisahkan dengan jelas antara *presentation logic* dengan *business logic*. Pembagian aplikasi dalam 3 bagian terpisah meningkatkan kompleksivitas.

2.12 Framework Laravel

2.12.1 Pengertian Framework

Framework adalah struktur konseptual dasar yang berisi kumpulan fungsi untuk tujuan tertentu yang sudah siap untuk digunakan, sehingga pembuatan aplikasi dapat dilakukan dengan lebih cepat karena kode programnya tidak di buat dari awal. Beberapa alasan digunakannya framework dalam membuat aplikasi adalah aplikasi akan memiliki standar pemrograman yang universal, menghindari repetitive work, memudahkan dalam team work, memudahkan dalam maintenance dan pengembangan aplikasi di masa mendatang dan hemat waktu dan biaya.

2.12.2 Laravel

Laravel adalah framework bahasa Pemrograman Hypertext Preprocessor (PHP) yang ditujukan untuk pengembangan aplikasi berbasis web dengan menerapkan konsep Model View Controller (MVC). Framework ini dibuat oleh Taylor Otwell dan pertama kali dirilis pada tanggal 9 Juni 2011. Laravel berlisensi open source yang artinya bebas digunakan tanpa harus melakukan pembayaran. Alamat website resmi dari framework Laravel adalah <https://laravel.com>. Fitur-fitur modern Laravel yang sangat membantu

developer dalam membuat aplikasi adalah Bundles, Eloquent ORM (Object-Relational Mapping), Query Builder, Application Logic, Reverse Routing, Resource Controller, Class Auto Loading, View Composers, Blade, IoC Containers, Migration, Database Seeding, Unit Testing, Automatic Pagination, Form request, dan Middleware.

Framework Laravel juga memiliki beberapa keunggulan sebagai berikut.

1. Menggunakan Command Line Interface (CLI) Artisan.
2. Menggunakan package manager PHP Composer.
3. Penulisan kode program lebih singkat, mudah dimengerti, dan ekspresif.

Kemudian untuk cara instalasi framework Laravel dapat dilakukan dengan 3 cara yaitu melalui: (1) Installer Laravel, (2) Menggunakan Composer dengan mengetikkan perintah create-project dan (3) Download source code Laravel melalui GitHub dengan alamat <https://github.com/laravel/laravel/>.

Pada tanggal 23 Agustus 2016 framework Laravel versi 5.3 resmi dirilis. Versi 5.3 akan mendapatkan dukungan bug fixes selama enam bulan dan security fixes selama satu tahun. Untuk menggunakan Laravel versi 5.3 komputer atau server yang digunakan harus memenuhi persyaratan sebagai berikut:

1. PHP \geq 5.6.4
2. OpenSSL PHP Extension
3. PDO PHP Extension
4. Mbstring PHP Extension
5. Tokenizer PHP Extension
6. XML PHP Extension

Fitur framework Laravel yang ditekankan pada penelitian ini adalah Blade, Migration, Eloquent ORM, Resource Controller, dan Middleware. Berikut adalah penjelasan mengenai lima fitur tersebut.

1. Blade

Blade adalah template engine. Pada dasarnya Blade adalah view namun dengan menggunakan Blade akan mempermudah untuk mengatur tampilan website dan menampilkan data. Cara untuk membuat file view

menjadi file Blade adalah dengan menambahkan ekstensi `.blade.php` pada file view dan cara untuk memanggil file Blade sama dengan cara untuk memanggil file view biasa. Contoh program berikut adalah perbandingan antara file view biasa dengan file Blade.

```
mahasiswa.php
<div id="Mahasiswa">
  <h1>Mahasiswa</h1>
  <?php if(!empty($mahasiswa)): ?>
    <ul>
      <?php foreach($mahasiswa as $mhs): ?>
        <li> <?= $mhs ?> </li>
      <?php endforeach ?>
    </ul>
  <?php else: ?>
    <p>Tidak ada data Mahasiswa.</p>
  <?php endif ?>
</div>

mahasiswa.blade.php
<div id="Mahasiswa">
  <h1>Mahasiswa</h1>
  @if(!empty($mahasiswa))
    <ul>
      @foreach($mahasiswa as $mhs)
        <li> {{ $mhs }} </li>
      @endforeach
    </ul>
  @else
    <p>Tidak ada data Mahasiswa.</p>
  @endif
</div>
```

Dari kedua contoh program diatas dapat disimpulkan jika dengan menggunakan file Blade maka penulisan program akan menjadi lebih singkat dan rapi.

2. Migration

Migration adalah fitur yang menyediakan cara baru untuk membuat database. Dengan menggunakan migration cara membuat database melalui Command Line Interface (CLI) database atau dengan menggunakan aplikasi database manager digantikan dengan menggunakan class. Tahapan menggunakan migration adalah membuat class kemudian melakukan perintah migrate melalui Command Line Interface (CLI) artisan. Keuntungan menggunakan migration adalah class yang dibuat bisa dipakai untuk membuat database pada berbagai macam Relation Database Management System (RDBMS) yang didukung oleh Laravel. Sebagai contoh misalnya aplikasi yang digunakan selama ini menggunakan database MySQL, kemudian karena alasan pengembangan aplikasi maka akan dilakukan penggantian database ke PostgreSQL. Dalam proses penggantian tersebut tidak perlu membuat class lagi, tinggal melakukan perintah migrate melalui Command Line Interface (CLI)

artisan. Keuntungan lain dari menggunakan migration adalah semua perubahan yang dilakukan pada database akan disimpan pada suatu tabel. Sehingga bisa dilakukan pembatalan (rollback) pada database jika melakukan perubahan yang tidak benar.

3. Eloquent ORM

Eloquent ORM adalah implementasi dari Active Record yang digunakan untuk mengatur relasi antar tabel di data base. Pada Eloquent ORM tabel direpresentasikan dalam bentuk kelas dan data yang tersimpan didalam tabel direpresentasikan dalam bentuk objek. Relasi yang dapat diatur menggunakan Eloquent ORM adalah sebagai berikut.

- a. One-to-One yaitu relasi satu ke satu. Pada relasi ini digunakan method `hasOne` dan `belongsTo`.
- b. One-to-Many yaitu relasi satu ke banyak. Pada relasi ini digunakan method `hasMany` dan `belongsTo`.
- c. Many-to-One yaitu relasi banyak ke satu. Pada relasi ini digunakan method `belongsTo` dan `hasMany`.
- d. Many-to-Many yaitu relasi banyak ke banyak. Pada relasi ini digunakan method `belongsToMany`.

4. Resource Controller

Resource Controller adalah fitur yang digunakan untuk mempercepat pembuatan controller. Sebagai contoh misalnya ada controller yang menangani semua HTTP request terhadap data dosen, untuk membuat controller tersebut hanya perlu mengetikkan perintah berikut. `php artisan make:controller DosenController --resource` Perintah diatas akan menghasilkan controller `DosenController.php` yang disimpan pada folder `app/Http/Controllers`. Tabel dibawah ini adalah daftar action yang dapat dilakukan oleh controller `DosenController.php`. Tabel 2.2 Daftar Action `DosenController.php`

No	Verb	URI	Action	Route Name
1	GET	/dosen	index	dosen.index
2	GET	/dosen/create	create	dosen.create
3	POST	/dosen/store	store	dosen.store
4	GET	/dosen/{dosen}	show	dosen.show
5	GET	/dosen/{dosen}/edit	edit	dosen.edit
6	PUT/PATCH	/dosen/{dosen}	update	dosen.update
7	DELETE	/dosen/{dosen}	destroy	dosen.destroy

Setelah membuat controller `DosenController.php` hal yang harus dilakukan selanjutnya adalah membuat satu baris kode program pada route.

Route:resource('dosen', 'DosenController'); Satu baris kode program pada route diatas akan menangani semua route untuk melihat, menambah, mengedit, dan menghapus data dosen. Jadi dapat disimpulkan dengan menggunakan fitur Resource Controller dapat mempercepat pembuatan controller serta dapat menyederhanakan route untuk controller.

5. Middleware

Middleware adalah fitur yang menyediakan mekanisme untuk memfilter HTTP request yang masuk ke aplikasi. Laravel memiliki beberapa Middleware yaitu Authenticate, EncryptCookies, RedirectIfAuthenticated, dan VerifyCsrfToken. Sebagai pembahasan akan dibahas Middleware Authenticate. Middleware tersebut akan memeriksa apakah user sudah login atau belum. Jika user sudah login maka request akan dilanjutkan ke halaman yang dikehendaki oleh user. Tetapi jika user belum login maka Middleware Authenticate akan mengarahkan user ke halaman login. Jika Middleware yang sudah ada pada Laravel kurang sesuai dengan kebutuhan ataupun tidak sesuai dengan kebutuhan maka dapat dibuat sendiri Middleware yang sesuai dengan kebutuhan. Struktur folder dari framework Laravel 5.3 yang masih default sebagai berikut:

- a. *Folder app* adalah folder yang berisi kode program inti dari aplikasi yang akan dibuat. Model dan controller tersimpan pada folder ini.
- b. *Folder bootstrap* adalah folder yang berisi konfigurasi autoloading dan terdapat juga folder cache yang menyimpan file-file yang dihasilkan secara otomatis oleh Laravel untuk mengoptimasi kinerja dari sistem yang dihasilkan.
- c. *Folder config* adalah folder yang berisi semua file konfigurasi aplikasi.
- d. *Folder database* adalah folder yang berisi file database migration dan seeds.
- e. *Folder public* adalah folder yang berisi file index.php. File tersebut digunakan sebagai entry point untuk menangani semua request yang masuk ke aplikasi. Pada folder ini juga dapat disimpan beberapa aset dari aplikasi seperti gambar, JavaScript, dan CSS.

- f. *Folder resources* adalah folder yang berisi file view dari aplikasi yang dibuat. Selain itu terdapat juga file language yang digunakan aplikasi.
- g. *Folder routes* adalah folder yang berisi file yang digunakan untuk mendefinisikan semua route ke aplikasi. Secara default ada tiga file route yang disediakan Laravel yaitu `api.php`, `console.php`, dan `web.php`.
- h. *Folder storage* adalah folder yang berisi template Blade yang dikompilasi, file session, file cache, dan file lainnya yang dihasilkan secara otomatis oleh Laravel.
- i. *Folder tests* adalah folder yang berisi semua file test yang dibuat untuk aplikasi.
- j. *Folder vendor* adalah folder yang menyimpan semua library yang digunakan.

2.12.3 Kelebihan Framework Laravel

Pengembangan Web menggunakan framework laravel mempunyai kelebihan sebagai berikut:

1. Waktu yang dibutuhkan untuk mengembangkan projek website dengan menggunakan framework ini menjadi lebih cepat.
2. Dapat meningkatkan pengunjung webiste karena teknologi framework ini dapat digunakan di segala browser dan berbagai perangkat dengan baik.
3. Laravel dilengkapi dengan utilitas pemrograman untuk membantu proses pengembangan aplikasi web dan juga moderasi dengan cara terbaik. Ini dikemas dengan Modular Packaging System (MPS) dengan pengaturan ketergantungan yang lengkap.

2.13 Pengembangan Sistem

Membangun aplikasi komputer (program) bukan merupakan hal yang sederhana, karena didalamnya terdapat beberapa aspek yang perlu diperhatikan. Aplikasi dapat di ibaratkan sebuah bangunan, rumah tentunya merupakan sebuah bangunan besar yang di bentuk oleh potongan-potongan yang lebih kecil seperti pondasi, dinding dan atap begitu pula dengan aplikasi, ia terbentuk juga dari sesuatu yang lebih kecil dan saling berhubungan sehingga membentuk sebuah aplikasi

lengkap, didalamnya dibangun dari sekumpulan kode-kode dengan tugas dan fungsi tertentu, kode ini bahkan bisa sampai beribu-ribu baris.

Oleh karena kerumitan dari aplikasi tersebut tentunya proses pembuatannya harus di rencanakan dengan matang, aspek ini harus diperhatikan dengan serius karena merubah aplikasi yang sudah selesai membutuhkan proses yang rumit dan membutuhkan waktu yang lama, karena komponen diantara software tersebut saling berkaitan, sedikit perubahan bisa saja mempengaruhi sebagian atau bahkan seluruh komponen-komponen lainnya. Jika mengambil kasus dari sebuah program yang berbasis web dari sisi kode program kita bisa melihat bahwa sebuah aplikasi terdiri dari tiga bagian yaitu kode yang berfungsi mengatur antarmuka/interface program atau dikenal dengan istilah front-end, kode yang mengatur interaksi database dan kode yang berhubungan dengan logika atau di kenal dengan istilah back-end.

Dalam pengembangan aplikasi tersebut biasanya ketiga bagian tergabung menjadi satu bagian namun semakin banyak modul aplikasi yang akan dibuat maka model ini kurang baik diterapkan karena dapat membingungkan programmer dalam melakukan pengembangan aplikasi, hal ini disebabkan karena aplikasi harus lebih teliti dalam melihat kode program yang sedang dibuat, selain itu sudah lazim diketahui bahwa aplikasi web idealnya minimal dikerjakan oleh dua orang programmer, satu programmer menyelesaikan logika aplikasi (back-end) dan programmer lainnya mengerjakan antarmuka dari aplikasi (front-end).

Model yang sama juga akan menimbulkan masalah karena sulit rasanya dua orang programmer mengerjakan aplikasi secara bersama karena ketika programmer mengubah kode yang berurusan dengan aplikasi biasanya mempengaruhi kode-kode yang berhubungan dengan antarmuka program, ini menyulitkan terjadinya koodinasi yang baik diantara kedua programmer tersebut.

Salah satu konsep yang dibuat oleh pakar pemrograman adalah memecah sebuah aplikasi web menjadi tiga bagian yaitu model yang berkaitan dengan operasi yang berhubungan dengan basis data, view yang berhubungan dengan antarmuka aplikasi dan terakhir controller yang berhubungan dengan logika aplikasi dan mengendalikan alur data antara view dan controller konsep ini merupakan pengembangan dari konsep pemrograman berorientasi objek dan dalam pemrograman dikenal dengan MVC (Hasyrif SY dan Rismayani, 2016).