**APLIKASI SISTEM PENJUALAN**

**RESTORAN KONAKITO**



Makalah

Web Programming

Diajukan Untuk Memenuhi Mata Kuliah Pada Program DiplomaIII

**Sandiko Prasetyo 12163302**

**Ade Gunawan 12165969**

**Putra Cipta Wijaya 12165179**

**Theodora Manalu 12164259**

**Jurusan Manajemen Informatika**

**Akademi Manajemen Informatika dan Komputer Bina Sarana Informatika**

**Tangerang**

**2018**

# PERSETUJUAN Makalah

Tugas Program ini telah disetujui untuk dinilai pada Tahun Akademik 2018/2019 di Semester Lima.

DOSEN AKADEMIK

KELAS 12.5E.27

(Mulyadi S,Kom)

# KATA PENGANTAR

Puji syukur kami panjatkan kehadirat ALLAH SWT yang telah melimpahkanrahmat dan karunia-Nya sehingga pada akhirnya kami dapat menyelesaikan tugas Web Programming ini dengan baik dimana tugas makalah ini penulis sajikan dalam bentuk buku yang sederhana, adapun judul penulisan laporan kuliah kerja praktek yang saya ambil adalah sebagai berikut:

**ANALISA SISTEM INFORMASI PELAKSANAAN PEMBUATAN PROGRAM**

Tujuan penulisan kuliah kerja praktek ini dibuat untuk memenuhi salah satu mata kuliah Web Programming pada program Diploma III ( D3 ) AMIK BSI BSD. Sebagai bahan penulisan diambil berdasar kan hasil penelitian,beberapa sumber literatur yang mengandung penulisan ini. Penulis menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, maka penulisan Web Program dan makalah ini tidak akan lancar, oleh karena itu pada kesempatan ini saya menyampaikan ucapan terima kasih kepada:

1. Bapak Ir. Naba Aji Notoseputro selaku Direktur AMIK BSI.
2. Bapak Mulyadi S,Kom selaku Dosen matakuliah Web Programming
3. Staf / Karyawan / dosen dilingkungan AMIK BSI.
4. Orang Tua tercinta yang telah memberi dorongan moril maupun spirituil kepada penulis.
5. Serta rekan-rekan sejawat dan seperjuangan yang ikut memberikan bantuannya dalam pembuatan kuliah kerja praktek, khususnya MI.5E

Serpong, Desember 2018

Penulis

# DAFTAR ISI

Contents

[PERSETUJUAN Makalah i](#_Toc531640925)

[KATA PENGANTAR ii](#_Toc531640926)

[DAFTAR ISI iii](#_Toc531640927)

[DAFTAR SIMBOL 2](#_Toc531640928)

[DAFTAR GAMBAR 4](#_Toc531640929)

[DAFTAR TABEL 5](#_Toc531640930)

# DAFTAR SIMBOL

1. **Simbol Use Case Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **GAMBAR** | **NAMA** | **KETERANGAN** |
| 1 |  | *Actor* | Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan *use case*. |
| 2 |  | *Dependency* | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*independent*). |
| 3 |  | *Generalization* | Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*). |
| 4 |  | *Include* | Menspesifikasikan bahwa *use case* sumber secara *eksplisit*. |
| 5 |  | *Extend* | Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan. |
| 6 |  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
| 7 |  | *System* | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
| 8 |  | *Use Case* | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor |
| 9 |  | *Collaboration* | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi). |
| 10 |  | *Note* | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi |

1. **Simbol Activity Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **GAMBAR** | **NAMA** | **KETERANGAN** |
| **1** |  | *Activity* | Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain |
| **2** |  | *Action* | State dari sistem yang mencerminkan eksekusi dari suatu aksi |
| **3** |  | *Initial Node* | Bagaimana objek dibentuk atau diawali. |
| **4** |  | *Activity Final Node* | Bagaimana objek dibentuk dan dihancurkan |
| **5** |  | *Fork Node* | Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran |

# DAFTAR GAMBAR

# DAFTAR TABEL

# DAFTAR LAMPIRAN

# BAB I PENDAHULUAN

## Latar belakang Masalah

Di era kemajuan teknologi seperti sekarang ini, Website menjadi hal yang sudah tidak asing lagi terdengar pada telinga kita, khususnya dalam dunia internet. Meskipun sudah banyak orang mempunyai hompage di internet baik yang menggunakan ISP maupun server web gratis, tetapi pada umumnya homepage tersebut bersifat statis dan tidak dapat berinteraksi dengan pengunjung web.

Kebanyakan orang pada umumnya hanya perperan sebagai pengguna dan sebagian kecil saja yang mengetahui proses dibaliknya dikarenakan tidak mengusai programman web. Pada dasar pembuatan nya, sebuah homepage yang dapat berinteraksi dengan pengunjung website atau homepage dinamis tidak diperlukan kemampuan programan yang tinggi.

## Maksud dan Tujuan

Adapun maksud penulisan Makalah ini adalah :

1. Menerapkan dan mempraktekan ilmu pengetahuan yang telah di dapat dalam perkuliahan.
2. Membentuk pola pikir mahasiswa untuk menjadi pribadi yang memiliki wawasan pengetahuan dan dapat lebih mengembangkan potensi diri agar tidak hanya menjadi pengguna saja tetapi dapat berperan sebagai pembuat suatu alamat web (*homepage*) yang dinamis.
3. Melakukan penelitian mengenai suatu masalah di bawah bimbingan yang cermat dari dosen pembimbing, untuk menghasilkan pengetahuan baru dari penelaahan yang original.

Sedangkan tujuan dalam penulisan makalah web ini adalah untuk memenuhi UAS (ujian akhir semester) mata kuliah Web Programming pada Semester V (Lima) Jurusan Akademik Manajemen Informatika dan Komputer di Bina Sarana Informatika.

## 1.3 Metode Penelitian

Metode penelitian merupakan langkah penting dalam penyusunan makalah Web khususnya bagi perancangan system. Di dalam kegiatan penelitian penulis

melakukan pengumpulan data melalui cara :

1. Wawancara (Interview)

Dalam penulisan Makalah ini,untuk mendapatkan informasi secara lengkap maka penulis melakukan suatu metode suatu tanya jawab mengenai semua kegiatan yang berhubungan dengan sistem informasi Web(*homepage*).

1. Pengamatan (Observation)

Penulis melakukan pengamatan-pengamatan langsung terhadap kegiatan yang berhubungan dengan tema Web(*homepage*) yang dibuat. Hasil dari pengamatan tersebut langsung dicatat oleh penulis dan dari kegiatan observasi dapat diketahui kesalahan atau proses dan kegiatan tersebut.

1. Studi pustaka

Selain melakukan kegiatan diatas penulis juga melakukan studi kepustakaan melalui literatur-literatur atau referensi-referensi yang ada di perpustakaan Akademi Manajemen Informatika Bina Sarana Informatika maupun diperpustakaan lainnya.

## 1.4 Ruang Lingkup

Didalam penulisan Makalah ini, Agar tidak menyimpang dari permasalahan yang di bahas dan sistem yang telah dibuat, penulis membatasi pokok bahasan seputar pembuatan program web dengan menggunakan PHP.

## 1.5 Sistematika Penulisan

Untuk mengetahui secara ringkas permasalahan dalam Makalah program Web ini, maka digunakan sistematika penulisan yang bertujuan untuk memper mudah pembaca menelusuri dan memahami Makalah ini.

1. **PENDAHULUAN**

Pada bab ini penulis menguraikan tentang latar belakang secara umum, maksud dan tujuan, ruang lingkup yang membatasi permasalahan, metode penelitian , serta sistematika penulisan secara keseluruhan.

1. **LANDASAN TEORI**

Pada bab ini penulis menjelaskan tentang konsep dasar sistem, informasi dan peralatan pendukung (tools sistem).

1. **ANALISA SISTEM BERJALAN**

Pada bab ini penulis membahas mengenai hal yang bersifat

umum,tujuan perusahaan dengan menguraikan sejarah perusahaan dan struktur organisasi dan prosedur system berjalan, bab ini juga menguraikan tentang permasalahan dan alternatif pemecahan masalah.

1. **PENUTUP**

Pada bab ini penulis menjelaskan mengenai kesimpulan serta saran yang berhasil ditarik dari seluruh pembuatan program Web (*homepage*) ini. Yang mungkin bermanfaat bagi mahasiswa.

# BAB II LANDASAN TEORI



## Konsep Dasar Website

Website merupakan halaman situs sistem informasi yang dapat diakses secara cepat. Website ini didasari dari adanya perkembangan teknologi informasi dan komunikasi. Melalui perkembangan teknologi informasi, tercipta suatu jaringan antar komputer yang saling berkaitan. Jaringan yang dikenal dengan istilah internet secara terus-menerus menjadi pesan–pesan elektronik, termasuk e-mail, transmisifile, dan komunikasi dua arah antar individu atau komputer.

Keseluruhan halaman-halaman web yang terdapat dalam sebuah domain yang mengandung informasi. Sebuah website biasanya dibangun atas banyak halaman web yang saling berhubungan. Hubungan antara satu halaman web dengan halaman web lainnya disebut dengan hiperlink, sedangkan teks yang dijadikan media penghubung disebut hypert. Untuk membangun sebuah web page dibutuhkan sebuah Bahasa pemrograman yang lebih dikenal dengan sebutan web scripting. Dikatakan script karena perintah kode program tersebut akan diinterpreter dan tidak ada kompilasi untuk menjadikannya executable. Berdasarkan letak proses interpreter maka web scripting dibagi menjadi dua kategori, yaitu yang bersifat client side dan server side. Client side dilakukan oleh web browser seperti Internet Explorer, Netscape, Opera, dan Firefox. Untuk contoh bahasa client side adalah HTML, CSS, Javascript, VBscript, dan XML. Sedangkan server side dilakukan oleh web server seperti PWS (Personal Web Server untuk Sistem Operasi Windows 98), IIS (untuk Sistem Operasi Windows 2000/Windows XP), Apache, Tomcat, Xitami, dan ZOPE. Untuk contoh bahasa server side adalah ASP (.Net), PHP, JSP, CFM, dan CGI/PL.

Website ini dibuat untuk mempermudah masyarakat dalam penyewaan kapal di Indonesia yang belum banyak di ketahui. Pada makalah ini penulis akan membahas mengenai bagaimana langkah-langkah untuk membuat web penyewaan kapal termasuk cara pemeliharaan web. Sehingga dengan adanya web ini akan membantu mempermudah untuk menyediakan informasi maupun promosi kepada masyarakat melalui web.

1. **Server**

**Web server** adalah sebuah software yang memberikan layanan berbasis data dan berfungsi menerima permintaan dari HTTP atau [HTTPS](https://bukainfo.com/apakah-https-akan-menjadi-bagian-dari-seo/) pada klien yang dikenal dan biasanya kita kenal dengan nama web browser dan untuk mengirimkan kembali yang hasilnya dalam bentuk beberapa halaman web dan pada umumnya akan berbentuk dokumen HTML dalam bentuk sederhana web server akan mengirim data HTML kepada permintaan web Browser sehingga akan terlihat seperti pada umumnya yaitu sebuah tampilan website.

1. **Client**

Aplikasi web e-commerce beroperasi melibatkan dua sisi yakni sisi mesin server dan sisi client atau sering disebut server/client. Server bertugas menyediakan bermacam-macam jenis layanan misalnya adalah pengaksesan berkas, peripheral, database dan dihubungkan dengan berbagai client.Sedangkan client adalah sebuah terminal yang menggunakan layanan tersebut.

Sebuah terminal client melakukan pemrosesan data di terminalnya sendiri dan hal itu menyebabkan spesifikasi dari server tidaklah harus memiliki performansi yang tinggi, dan kapasitas penyimpanan data yang besar karena semua pemrosesan data yang merupakan permintaan dari client dilakukan di terminal client Jaringan komputer yang terdistribusi seperti LAN, WAN dan juga internet yang digunakan untuk ecommerce membutuhkan kontrol dan keamanan yang lebih kompleks, sehingga akuntan terutama fungsi internal audit perlu untuk mengevaluasi lingkungan pengendalian yang relevan dengan jaringan atau web servernya .

1. **HTML**

HTML (Hypertext Markup Language) adalah bahasa dasar untuk web scripting bersifat client side yang memungkinkan untuk menampilkan informasi dalam bentuk teks, grafik, serta multimedia dan juga untuk menghubungkan antartampilan web page (hyperlink).

1. **CSS**

Cascading Style Sheet (CSS) adalah satu kumpulan kode-kode untuk memformat, yang mengandalkan satu tampilan isi pada halaman web. Penggunaan style CSS pada format satu halaman diletakan terpisah dari tampilan halaman. Isi dari halman kode HTML terletak di dalam file HTML, sedangkan kode CSS dapat berupa tampilan kode yang berada dalam file lain atau salah satu bagian dari dokumen HTML, dan biasanya diletakan dibagian kepala atau tag <head>.

1. **JavaScript**

JavaScript adalah bahasa scripting kecil, ringan, berorientasi objek yang ditempelkan pada kode HTML dan di proses di sisi client. JavaScript digunakan dalam pembuatan website agar lebih interaktif dengan memberikan kemampuan tambahan terhadap HTML melalui eksekusi perintah di sisi browser. JavaScript dapat merespon perintah user dengan cepat dan menjadikan halaman web menjadi responsif. JavaScript memiliki struktur sederhana, kodenya dapat disisipkan pada dokumen HTML atau berdiri sebagai satu kesatuan aplikasi.

1. **Struktur Navigasi**

Struktur navigasi adalah struktur atau alur cerita dari suatu program yang merupakan rancangan hubungan dan rantai kerja dari beberapa area yang berbeda dan dapat membantu mengorganisasikan seluruh elemen-elemen yang digunakan dalam program. Menentukan struktur navigasi merupakan hal yang sebaiknya dilakukan sebelum membuat suatu aplikasi. Pada struktur navigasi user termasuk struktur navigasi non linear karena pada struktur non linear diperkenankan membuat penjejakan bercabang. Percabangan ini berbeda dengan percabangan pada struktur hirarki. Pada percabangan non linear walaupun terdapat percabangan tetapi tiap-tiap tampilan mempunyai kedudukan yang sama tidak ada pada master page, slave page, dan halaman pendukung.

1. **CRUD**

Pada umumnya sebuah sistem yang berupa aplikasi perangkat lunak memiliki fungsi dasar yang dikenal dengan istilah CRUD (Create Update Retrieve Delete), terutama untuk aplikasi yang memanfaatkan basis data. Pengerjaan kode program berfokus pada keempat fungsi dasar ini. Proses CRUD menitik beratkan pada bagaimana data dapat dimanipulasi dan menampilkan informasi sesuai dengan kebutuhan pengguna. Alur bisnis proses suatu perusahaan perusahaan berbeda satu dengan yang lainnya, maka informasi yang disajikan berbeda pula. Di dalam project, seorang analis bertugas membuat analisa berkaitan dengan bisnis proses yang berjalan. Hasilnya adalah sebuah desain sistem yang siap dikerjakan oleh programmer. Pada kenyataannya manipulasi data melalui proses CRUD menjadi bagian yang tak terpisahkan dari pengerjaan code program dan meliputi hampir keseluruhan fase implementasi penger jaan proyek perangkat lunak.

1. **API (Aplication Program Interface)**

Merupakan suatu dokumentasi yang terdiri dari interface, fungsi, kelas, struktur dan sebagainya untuk membangun sebuah perangkat lunak. Dengan adanya API ini, maka memudahkan programmer untuk “membongkar” suatu software untuk kemudian dapat dikembangkan atau diintegrasikan dengan perangkat lunak yang lain. API dapat dikatakan sebagai penghubung suatu aplikasi dengan aplikasi lainnya yang memungkinkan programmer menggunakan sistem function. Proses ini dikelola melalui operating system. Keunggulan dari API ini adalah memungkinkan suatu aplikasi dengan aplikasi lainnya dapat saling berhubungan dan berinteraksi. Bahasa pemrograman yang digunakan oleh Google Maps yang terdiri dari HTML, Javascript dan AJAX serta XML, memungkinkan untuk menampilkan peta Google Maps di website lain.

1. **REST (Representational State Transfer)**

REST adalah filosofi desain yang mendorong kita untuk menggunakan protokol dan fitur yang sudah ada pada Web untuk memetakan permintaan terhadap sumber daya pada berbagai macam representasi dan manipulasi data di Internet (Scribner, 2009). REST adalah gaya arsitektural yang memiliki aturan seperti antar muka yang seragam, sehingga jika aturan tersebut diterapkan pada web services akan dapat memaksimalkan kinerja web services terutama pada performa, skalabilitas, dan kemudahan untuk dimodifikasi. Pada arsitektur REST data dan fungsi dianggap sebagai sumber daya yang dapat diakses lewat Uniform Resource Identifier (URI), biasanya berupa tautan pada web. REST menggunakan protokol HTTP yang bersifat stateless. Perintah HTTP yang bisa digunakan adalah fungsi GET, POST, PUT atau DELETE. Hasil yang dikirimkan dari server biasanya dalam bentuk format XML atau JSON sederhana tanpa ada protokol pemaketan data, sehingga informasi yang diterima lebih mudah dibaca dan diparsing disisi client. Dalam penerapannya, REST lebih banyak digunakan untuk web service yang berorientasi pada resource. Maksud orientasi pada sumber daya adalah orientasi yang menyediakan sumber daya sebagai layanannya dan bukan kumpulan-kumpulan dari aktifitas yang mengolah sumber daya itu. Bentuk web service menggunakan REST style sangat cocok digunakan sebagai backend dari aplikasi berbasis mobile karena cara aksesnya yang mudah dan hasil data yang dikirimkan berformat JSON sehingga ukuran file menjadi lebih kecil.

1. **Model View Controller (MVC)**

Model View Controller pertama kali dipublikasikan oleh peneliti XEROX PARAC yang bekerja dalam pembuatan bahasa pemrograman Smalltalk sekitar tahun 1970-1980. MVC adalah sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data (Model) dari tampilan (View) dan cara bagaimana memprosesnya (Controller). Dalam implementasinya 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.

Bagian Model View Controller :

* 1. Model Model mewakili struktur data. Biasanya model berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data seperti memasukkan data ke basis data, pembaruan data dan lain-lain.
  2. View View adalah bagian yang mengatur tampilan ke pengguna. Bisa di katakan berupa halaman web.
  3. Controller Controller merupakan bagian yang menjembatani model dan view. Controller berisi perintah-perintah yang berfungsi untuk memproses suatu data dan mengirimkannya ke halaman web.

1. **Templating Engine**

Sebuah (komponen) perangkat lunak yang dirancang untuk menggabungkan sebuah template dengan data untuk dijadikan dokumen hasil. Pada umumnya di dunia web, dokumen yang dihasilkan oleh template engine merupakan dokumen HTML, walaupun tidak jarang juga kita menghasilkan data lain seperti PDF atau JSON. Pada beberapa kasus ekstrim, template engine bahkan digunakan untuk menghasilkan kode program tertentu berdasarkan masukan pengguna.

Sebagai komponen perangkat lunak yang kerap digunakan dalam berbagai lingkungan dan situasi, *template engine* memiliki beberapa fitur umum, yaitu:

1. Perulangan dan percabangan.
2. Pembuatan fungsi.
3. Fungsi bawaan (biasanya berhubungan dengan teks).
4. Blok dokumen.
5. Penggabungan file (include).
6. Pewarisan template.

Seperti yang dapat dilihat di atas, *template engine* pada umumnya memiliki fitur yang mirip dengan sebuah bahasa pemrograman (perulangan dan percabangan, pembuatan fungsi, dan fungsi bawaan). Hal ini disebabkan karena pada pembuatan *template* seringkali kita akan memerlukan cara untuk menampilkan data secara berulang. Misalnya, kita dapat saja ingin menampilkan data di dalam tabel, dan baris dari tabel merupakan data yang diambil dari basis data. Metode paling efektif untuk menampilkan data tersebut tentunya dengan melakukan perulangan yang mengeluarkan baris table.

## 2.1.2 Bahasa Pemograman

## Back End

Back end atau sering di sebut server side pada dasarnya adalah tempat dimana proses suatu aplikasi atau sistem berjalan di back end ini data di proses ditambahkan, diubah atau dihapus.Back end mengurusi segala sesuatu yang biasanya tidak dilihat atau berinteraksi langsung kepada user, seperti database dan server. Biasanya orang yang bekerja sebagai back end developer adalah programmer atau developer yang fokus pekerjaannya pada keamanan, desain sistem dan managemen data pada sistem. Back end developer dibutuhkan dalam pengembangan sistem atau aplikasi dinamis yang memiliki data yang selalu berubah ubah, contoh website dinamis antara lain facebook dan google. Bahasa pemrograman yang biasanya dikuasai oleh seorang backend developer adalah bahasa pemrograman yang dapat digunakan untuk mengelola database, mengolah file dan I/O seperti PHP, ASP, NodeJs, atau C#.

1. **Node.Js**

Menurut Equan Pr (2013), Node.js dikembangkan dari engine javaScript yang dibuat oleh Google untuk Browser Chrome / Chronium (V8) ditambah dengan libUV serta bebrapa pustaka internal lainnya. Dengan menggunakan Node.js semua penembangan akan dilakukan dengan javaScript, baik pada sisi client maupun server. Pengembangan aplikasi dengan menggunakan Node.js dapat dilakukan secara moduler yaitu dengan memisahkan berbagai komponen kedalam pustaka (library). Pustaka tersebut dapat dikelola dengan npm yang terdapat di Node.js. pada dasarnya, Node.js sebuah runtime environment dan script library. Sebuah runtime environment adalah sebuah software yang berfungsi untuk mengeksekusi, menjalankan dan mengimplementasikan fungsifungsi serta cara kerja inti dari suatu bahasa pemrograman. Sedangkan script library adalah kumpulan, kompilasi atau bank data berusi script atau kode-kode pemrograman.

Kalemi & Tola, (2013), Node.js adalah sebuah platform yang dibangun di atas Chrome JavaScript Runtime untuk membangun aplikasi yang memiliki performa baik dan cepat. Node.js menggunakan model event-driven, model non-blocking I/O yang membuatnya ringan dan efisien sehingga cocok untuk aplikasi real-time dengan data intensif yang berjalan pada perangkat yang tersebar.

Kelebihan utama Node.js adalah pada fitur non-blocking I/O yang dimilikinya, sehingga Node.js dapat melayani banyak permintaan dari klien sekaligus tanpa harus menunggu permintaan sebelumnya selesai diproses.

1. **Express.JS**

Menurut Muhammad Arslan (2016), Express.js adalah satu web framework paling popular di dunia Node.js. Dokumentasinya yang lengkap dan penggunaannya yang cukup mudah, dapat membuat kita mengembangkan berbagai produk seperti aplikasi web ataupun RESTful API. Express.js pun dapat digunakan menjadi pijakan untuk membangun web framaework yang lebih kompleks seperti, Sails.js, MEAN (MongoDB, Express.js, Angular.js, Node.js) dan MERN (MongoDB, Express.js, React.js, Node.js). Express.js dibuat oleh TJ Holowaychuck dan sekarang dikelola oleh komunitas.

1. **MongoDB**

MongoDB adalah salah satu NoSQL yang berorientasi dokumen yang dikembangkan secara kode terbuka (Open Source) oleh 10gen. Beberapa karakteristik penting dari MongoDB adalah penyimpanan data dalam collections dan masing – masing collections memiliki documents yang diserialisasi dengan menggunakan BSON (Binary JSON).

Perbandingan antara MongoDB dan SQL Server pernah dilakukan dalam lingkungan sistem pengambilan keputusan dan layanan data secara interaktif yang mengatakan sisi kinerja SQL Server lebih unggul meskipun aspek lainnya NoSQL memiliki keunggulan yang berbeda. Perbedaan antara berbagai NoSQL berbasis dokumen termasuk MongoDB dibandingkan jenis lain juga pernah diulas oleh Kaur. Aspek keamanan dari MongoDB yang masih banyak kekurangan juga pernah menjadi bahan penelitian. MongoDB dan berbagai jenis NoSQL lainnya juga menjadi subyek studi komparasi. Penelitian ini berbeda dibandingkan dengan penelitian sebelumnya dimana aspek yang dibandingkan adalah aspek koneksi, desain, dan kinerja dari Data Manipulation Language dari MongoDB apabila dibandingkan dengan SQL Server dengan menggunakan aplikasi yang identik.

1. **Templating Engine**

## 2.3 Front End

Front end adalah segala sesuatu yang mengubungkan antara user dengan sistem back end. Biasanya merupakan sebuah user interface dimana user akan berinteraksi dengan sistem. Pekerjaan yang sering muncul sebagai seorang front end developer adalah desainer user interface dan desainer user experience. Seorang front end developer tidak akan membuat program atau aplikasi yang berjalan di logic bisnis tapi fokusnya akan lebih banyak ke antarmuka, desain grafis (user interface designer) dam bagaimana membuat desain yang nyaman digunakan oleh user (user experience designer). Bahasa pemrograman yang biasanya digunakan dalam pengembangan front end adalah html.

Framework diartikan sebagai alat yang digunakan untuk membantu dan memudahkan pembuatan aplikasi dalam konteks ini adalah aplikasi web. Framework juga didefinisikan sebagai kumpulan script (class dan function) yang dapat membantu developer menangani berbagai masalah seperti koneksi ke database, pemanggilan variable dan fungsi-fungsi lainnya sehingga developer dapat lebih fokus dan lebih cepat membangun aplikasi. Sebuah framework umumnya telah menyertakan perintah-perintah siap pakai yang dibutuhkan dalam membuat suatu aplikasi, namun pihak developer tetap harus menulis kode sendiri dan harus menyesuaikan dengan lingkungan framework yang digunakan.

Laravel merupakan framework PHP yang menekankan pada kesederhanaan dan fleksibilitas pada desainnya. Laravel dirilis di bawah lisensi MIT dengan sumber kode yang disediakan di Github. Sama seperti framework PHP lainnya, Laravel dibangun dengan basis MVC (Model-View-Controller). Laravel dilengkapi command line tool yang bernama “Artisan” yang bisa digunakan untuk packaging bundle dan installation bundle.

Menurut survey popularitas framework PHP yang lakukan oleh Sitepoint.com, Laravel menduduki urutan teratas, sehingga menjadikan Laravel sebagai framework PHP terbaik untuk tahun 2015. Lavarel merupakan framework dengan versi PHP yang up-to-date, karena Laravel mensyaratkan PHP versi 5.3 atau lebih tinggi. Menurut survey yang dilakukan oleh Sitepoint.com pada tahun 2015 bahwa framework PHP yang paling terbaik di tahun 2015 adalah Laravel.

Materialize merupakan salah satu framework material design ,yang banyak digunakan oleh para Web developer. Di dalam Materialize terdapat fitur pengenalan framework, mulai dari about yang berisi pengenalan produk, getting started tahapan sebelum memulai, CSS, component dan javascript bagian yang memperlihatkan beberapa fitur yang tersedia pada Materialize untuk mobile, sampai showcase bagian yang memperlihatkan demo dari web yang menggunakan Materialize.

## 2.4 Analisis Sistem Yang Berjalan

Analisis sistem yang berjalan adalah menganalisis sistem yang sedang berjalan untuk menemukan domain permasalahan dan menjabarkan domain permasalahan tersebut sebagai bahan acuan dalam perencanaan sistem. Analisis tersebut meliputi analisis dokumen, analisa prosedur, *flow map*, diagram konteks dan *data flow diagram*. Namun, dalam pembahasan kali ini, penulis menggunakan UML (*Unified Modelling Language*) sebagai pengganti *data flow diagram.*

## 2.5 Peralatan Pendukung

Adapun peralatan pendukung yang dimaksud untuk merancang model system yang baru pada penulisan tugas ini adalah :

### 2.5.1 UML

Unified Modeling Language ( UML ) adalah tujuan umum, perkembangan, bahasa pemodelan di bidang rekayasa perangkat lunak , yang dimaksudkan untuk menyediakan cara standar untuk memvisualisasikan desain sistem.

Menurut Windu Gata (2013 : 4) Hasil pemodelan pada OOADterdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak.

Sedangkan menurut Sukamto dan Shalahuddin (2013:133), UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

Rosa A.S dan M. Shalahudin (2014:140), pada UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Berikut ini penjelasan singkat dari pembagian kategori tersebut.

1. Structure diagram, yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan. Structure diagram terdiri dari class diagram, object diagram, component diagram, composite structure diagram, package diagram dan deployment diagram.
2. Behavior diagram yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem. Behavior diagram terdiri dari Use case diagram, Aktivity diagram, State Machine System.
3. Interaction diagram yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem. Interaction diagram terdiri dari Sequence 13 Politeknik Negeri Sriwijaya Bab II Tinjauan Pustaka Diagram, Communication Diagram, Timing Diagram, Interaction Overview Diagram.

Dalam pembahasan kali ini, penulis hanya menggunakan 2 *tools system* untuk membantu dalam mendeskripsikan *workflow* pada bahasan ini, adalah sebagai berikut :

1. Activity Diagram.
2. Use case Diagram.

### 2.5.2 Activity Diagram

Activity Diagram adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek.

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

### Use Case Diagram

Use case diagram merupakan pemodelan untuk kelakukan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

Diagram ini memperlihatkan himpunan use-case dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.