

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Untuk mendapatkan hasil yang optimal, penulis melakukan kajian dari penelitian-penelitian dan atau jurnal-jurnal terdahulu yang linear dan terkait dengan tugas akhir ini sehingga bisa dijadikan referensi, Diantaranya :

1. “*Two Factor Authentication*” (Amin, 2017). Makalah ini berfokus pada implementasi metode otentikasi dua faktor dengan menggunakan kata sandi alfanumerik tradisional yang ramah pengguna dan kata Sandi grafis sebagai *gateway* untuk otentikasi. Upaya telah dilakukan dengan menggunakan dua faktor Otentikasi, dan dalam makalah ini kami menjelaskan dua faktor desain sistem otentikasi dan implementasi desain. Jadi menambahkan kata sandi tambahan lapisan keamanan tambahan.
2. “Implementasi HMAC-SHA-3-Based *One Time Password* pada Skema *Two-Factor Authentication*”(Fakhrusy, 2016). Penelitian ini menunjukkan Abstract—skema *two-factor authentication* dibuat untuk memberikan tingkat keamanan tambahan bagi *user*. Salah satu contoh penerapan *two-factor authentication* pada proses *login user* adalah autentikasi dengan hash *password* ditambah dengan autentikasi menggunakan *one time password*. Pada jurnal ini, pembahasan akan fokus pada implementasi HMAC-SHA3 untuk algoritma HMAC-based *one time password* yang masih menggunakan HMAC-SHA-1. Hal ini dilakukan karena SHA3 akan memberikan keamanan yang lebih baik dibandingkan SHA-1
3. “Implementasi Algoritma Time-Based *One Time Password* dalam Otentikasi Token Internet *Banking*” (Ungkawa dkk, 2017). Pada penelitian ini 2FA diaplikasikan pada sistem internet *banking* di mana antara token virtual dan server dipasang algoritma TOTP untuk menghasilkan *password* sebagai otentikasi tambahan dalam proses transaksi finansial seperti pengiriman uang, pembayaran tagihan, dan pembayaran tiket. Sebelum transaksi diproses, server meminta *user* untuk memasukkan pssword OTP

yang dihasilkan token virtual kemudian server melakukan validasi terhadap *password* OTP tersebut. Dari hasil pengujian yang dilakukan bahwa *password* OTP tidak muncul secara berulang dan secret key yang dihasilkan secara acak juga tidak muncul secara berulang tetapi mempunyai prosentasi kemiripan tertinggi sebesar 0,03%.

4. “Implementasi *Two Factor Authentication* Dan *Protocol Zero Knowledge Proof* Pada Sistem *Login*” (Raharjo dkk, 2017). Makalah ini menjelaskan sistem *login* menggunakan Dua Faktor Otentikasi dan *Zero Knowledge Proof* menggunakan Schnorr NIZK. Sistem yang diusulkan dirancang untuk mencegah kata sandi bocor ketika dikirim melalui jaringan tidak aman atau ketika digunakan dalam perangkat yang tidak dipercaya. *Zero Knowledge Proof* digunakan untuk menjaga kerahasiaan kata sandi dan *Two Factor Authentication* digunakan untuk mengamankan proses *login* perangkat yang tidak dipercaya. Sistem yang diusulkan telah diuji dan hasil awal menunjukkan bahwa sistem tersebut dapat mengamankan proses masuk tanpa membocorkan kata sandi pengguna.
5. “Sistem *Two-Factor Authentication* Dengan Algoritma Time-Based One-Time *Password* pada Aplikasi Web Menggunakan Perangkat *Android*” (Wijaya dan Purwanti, 2016). Pada jurnal ini dijelaskan bahwa sistem *two-factor authentication* dapat dibangun dengan menggunakan kombinasi *username* dan *password* serta divalidasikan kepemilikannya dengan *password* dinamis *one-time password*. Salah satu metode untuk membangkitkan *one-time password* adalah Time-based *One Time Password* (TOTP), metode ini membangkitkan sebuah *password* dinamis yang berubah mengikuti suatu jeda waktu tertentu. Tujuan dari penelitian ini adalah bagaimana sebuah perangkat *mobile android* dapat digunakan sebagai pembangkit atau *generator One-time password* yang dapat digunakan dalam sistem *two-factor authentication*. Dalam mengimplementasikan *two-factor authentication* ini, penulis hanya membuat prototipe bagian sistem saja, sehingga *model* ini belum teruji pada aplikasi web yang sesungguhnya.

6. “Implementasi TOTP (Time-Based One-Time Password) Untuk Meningkatkan Keamanan Transaksi E-Commerce” (Daqiqil dkk, 2016). Penelitian ini bertujuan untuk meningkatkan keamanan transaksi e-commerce menggunakan metode TOTP (Time-Based One-Time Password). Sebagian besar *website* atau aplikasi e-commerce hanya mengandalkan *username* dan *password* untuk melindungi akun dan transaksi mereka. Jika *username* dan *password* tersebut dapat dicuri oleh pihak lain, maka dengan mudah pihak tersebut dapat menggunakan *username* dan *password* tersebut untuk bertransaksi. Salah satu solusi yang ditawarkan adalah menggunakan metode *two-factor authentication*. Dengan metode tersebut, Sistem e-commerce tidak hanya bergantung pada apa yang diketahui pengguna (*username* dan *password*) tetapi juga membutuhkan apa yang dimiliki pengguna (token). TOTP adalah sebuah mekanisme *two-factor authentication* dimana *password* hanya dapat digunakan satu kali dalam kurun waktu tertentu. Pengguna harus menginstall aplikasi TOTP client di ponsel mereka sebagai token untuk bertransaksi. Kesimpulan yang dapat ditarik dari referensi ini adalah dipelru mekanisme tambahan untuk membuktikan bahwa pengguna tersebut valid. Salah satu cara yang dapat dilakukan adalah menggunakan SMS ataupun konfirmasi melalui E-mail.
7. “Penerapan Metode Smart Authentication Dalam Layanan E-Banking Menggunakan Two Channel Authentication Dan QR-Code Pada Perangkat Mobile Android” (Yusuf, 2015). Pada jurnal yang dijadikan referensi ini dikatakan saat ini E-banking masih menggunakan teknik otentikasi yang menggunakan kerahasiaan *user ID* dan *password*. Selain itu juga terdapat juga penambahan token security. Akan tetapi teknik otentikasi tersebut saat ini me jadi sangat rentan seiring berkembangnya serangan yang bersifat langsung seperti *phising* dan *man-in-the-middle/browser*. *Two channel authentication* muncul sebagai salah satu solusi yang menawarkan jaminan perbaikan keamanan atas lemahnya keamanan pada teknik *single factor authentication*. Selain itu, saat ini berkembang pula penerapan QR-code untuk layanan berbasis smart authentication yang menjamin kecepatan

proses otentikasi pada transaksi di layanan internet *banking*. Dari hasil penerapan kedua teknik tersebut mampu memberikan perlindungan keamanan pada layanan internet *banking*.

2.2 Two Factor Authentication

Two factor Authentication adalah mekanisme yang mengimplementasikan dua faktor yang telah disebutkan diatas dan diketahui lebih kuat dan lebih aman daripada implementasi sistem *one factor authentication* sederhana. Mekanisme ini melibatkan banyak faktor-faktor pendukung dalam proses otentikasi yang biasa disebut dengan "*Multi Factor Authentication*". Faktor-faktor yang digunakan dalam *multi factor authentication* di antaranya adalah: *something you know*, *something you have*, *something you are*, *something you do*.

1. Something you know

Faktor *something you know* melibatkan pengetahuan informasi rahasia yang memungkinkan *user* meng-otentikasi dirinya sendiri ke sebuah server. Contoh dari faktor ini adalah sebuah *password* dan sebuah *personal identification number* (PIN).

2. Something you have

Faktor *something you have* melibatkan bahwa *user* harus memiliki alat secara fisik. Jika tanpa adanya alat tersebut maka *user* tidak dapat meng-otentifikasi dirinya sendiri ke server sistem computer. Contoh dari faktor ini menggunakan sebuah token dan *smart card* (kartu cerdas)

3. Something you are

Faktor *something you are* melibatkan bahwa *user* memiliki karakteristik yang unik yang membedakan dirinya dengan *user* lain untuk mengidentifikasi dirinya sendiri. Faktor ini menggunakan metode identifikasi biometrik untuk meng-otentikasi *user*. Contoh dari faktor *something you are* meliputi sidik jari, pemindaian retina mata, dan garis tangan seseorang.

4. *Something you do*

Faktor *something you do* melibatkan bahwa tiap *user* ketika melakukan sesuatu atau ketika menggunakan sesuatu dengan cara yang berbeda. Contoh dari faktor ini penggunaan analisis suara (*voice recognattion*) atau analisis tulisan tangan. (Arief. TI STMIK AMIKOM Yogyakarta)

Two-factor authentication dalam tugas akhir ini menggunakan dua kombinasi dari “*something you know*” yaitu *password* dan “*something you have*” yaitu token atau kode unik acak 6 digit angka.

2.3 Pembangkit Bilangan Acak

Pembangkit Bilangan Acak atau *Random Number Generator* adalah suatu peralatan komputasional yang dirancang untuk menghasilkan suatu urutan nilai tersebut dapat dianggap sebagai suatu keadaan acak (*random*). Bilangan acak adalah bilangan yang kemunculan berikutnya tidak dapat diprediksi, konsep bilangan acak ini banyak digunakan dalam kriptografi, misalnya untuk pembangkitan parameter kunci pada algoritma kunci-public seperti *Elliptic Curve* dan RSA, atau dalam pembangkitan *initialization vector* pada algoritma kunci-simetri, dan masih banyak lainnya.

Pada dasarnya tidak ada komputasi yang benar-benar menghasilkan deret bilangan acak secara sempurna. Pembangkitan bilangan acak ini dapat didekati dengan bilangan acak semu, dengan batasan parameter tertentu yang didefinisikan sebagai ‘syarat cukup’ bilangan acak. Bilangan acak yang dihasilkan dengan formula matematika disebut bilangan acak semu (*pseudorandom number*) karena pembangkitan bilangannya dapat diulang kembali. Pembangkit bilangan yang menghasilkan bilangan *pseudorandom* disebut *pseudorandom number generator* (PRNG). (Luqman; 2016)

2.4 *Mersenne Twister*

Mersenne Twister merupakan suatu *pseudo random number generator* yang ditemukan oleh Makoto Matsumoto dan Takuji Nishimura dari Departemen

Matematika, Keio University, Jepang. *Mersenne Twister* merupakan pembangkit bilangan pertama yang algoritmanya dibuat menawarkan generasi yang cepat dari bilangan acak kualitas tinggi setelah menghilangkan masalah yang dihadapi oleh pembangkit acak semu sebelumnya. Memiliki periode yang besar $2^{19937} - 1$ bilangan *random* terdistribusi (*equidistribution dimensional high order*) dengan baik dan empat kali lebih cepat dari `rand()` di C. Algoritmanya merupakan suatu *twisted generalised shift feedback register* atau TGSFR. Kata *twist* merupakan suatu transformasi dimana menjamin *equidistribution* dari nilai dalam 623 dimensi. (Luqman; 2016)

2.5 Website

2.5.1 Defenisi Website

Website atau Situs web adalah suatu halaman web yang saling berhubungan yang umumnya berada pada server yang sama berisikan kumpulan informasi yang disediakan secara perorangan, kelompok, atau organisasi.

Sebuah situs web biasanya ditempatkan setidaknya pada sebuah server web yang dapat diakses melalui jaringan seperti Internet, ataupun jaringan wilayah lokal (LAN) melalui alamat Internet yang dikenali sebagai URL. Gabungan atas semua situs yang dapat diakses publik di Internet disebut pula sebagai World Wide Web atau lebih dikenal dengan singkatan WWW.

Sebuah halaman web merupakan berkas yang ditulis sebagai berkas teks biasa (plain text) yang diatur dan dikombinasikan sedemikian rupa dengan instruksi-instruksi berbasis HTML atau XHTML, kadang-kadang pula disisipi dengan sekelumit bahasa skrip. (Sumber : Sulastri)

2.5.2 Macam-macam Website

Menurut Abdulloh (2018 : 1) secara umum *website* dibagi menjadi beberapa jenis seperti di bawah ini:

1. *Website* Statis

Website statis yaitu jenis *website* yang isinya tidak diperbaharui secara berkala, sehingga isinya dari waktu ke waktu akan selalu tetap. *Website* jenis

ini biasanya hanya digunakan untuk menampilkan profil dari pemilik *website* seperti profil perusahaan atau organisasi.

2. *Website* Dinamis

Website dinamis yaitu jenis *website* yang isinya terus diperbaharui secara berkala oleh pengelola web atau pemilik web. *Website* jenis ini banyak dimiliki oleh perusahaan atau perorangan yang aktifitas bisnisnya memang berkaitan dengan internet.

3. *Website* Interaktif

Website interaktif pada dasarnya termasuk dalam kategori *website* dinamis, dimana isi informasinya selalu diperbaharui dari waktu ke waktu. Hanya saja, isi informasi tidak hanya diubah oleh pengelola *website* tetapi lebih banyak dilakukan oleh pengguna *website* itu sendiri.

2.5.3 Unsur-unsur *Website* atau Situs

Untuk menyediakan keberadaan sebuah *website*, maka harus tersedia unsur-unsur penunjangnya, adalah sebagai berikut:

1. Nama *domain* (*Domain name/URL – Uniform Resource Locator*)

Pengertian Nama *domain* atau biasa disebut dengan *Domain Name* atau URL adalah alamat unik di dunia internet yang digunakan untuk mengidentifikasi sebuah *website*, atau dengan kata lain *domain name* adalah alamat yang digunakan untuk menemukan sebuah *website* pada dunia internet. Contoh <http://www.baliorange.net>, <https://www.bunehaba.com> dan <http://www.detik.com>.

Nama *domain* diperjualbelikan secara bebas di internet dengan status sewa tahunan. Nama *domain* sendiri mempunyai identifikasi ekstensi/akhiran sesuai dengan kepentingan dan lokasi keberadaan *website* tersebut. Contoh nama *domain* ber-ekstensi internasional adalah com, net, org, info, biz, name, ws. Contoh nama *domain* ber-ekstensi lokasi Negara Indonesia adalah co.id (untuk nama *domain website* perusahaan), ac.id (nama *domain website* pendidikan), go.id (nama *domain website* instansi pemerintah), or.id (nama *domain website* organisasi).

2. Rumah tempat *website* (Web *Hosting*)

Pengertian Web *Hosting* dapat diartikan sebagai ruangan yang terdapat dalam harddisk tempat menyimpan berbagai data, *file-file*, gambar dan lain sebagainya yang akan ditampilkan di *website*. Besarnya data yang bisa dimasukkan tergantung dari besarnya web *hosting* yang disewa/dipunyai, semakin besar web *hosting* semakin besar pula data yang dapat dimasukkan dan ditampilkan dalam *website*. Web *Hosting* juga diperoleh dengan menyewa. Besarnya *hosting* ditentukan ruangan harddisk dengan ukuran MB(Mega Byte) atau GB(Giga Byte). Lama penyewaan web *hosting* rata-rata dihitung per tahun. Penyewaan *hosting* dilakukan dari perusahaan-perusahaan penyewa web *hosting* yang banyak dijumpai baik di Indonesia maupun Luar Negeri.

3. Bahasa Program (Scripts Program)

Bahasa program adalah bahasa yang digunakan untuk menerjemahkan setiap perintah dalam *website* yang pada saat diakses. Jenis bahasa program sangat menentukan statis, dinamis atau interaktifnya sebuah *website*. Semakin banyak ragam bahasa program yang digunakan maka akan terlihat *website* semakin dinamis, dan interaktif serta terlihat bagus.

Beragam bahasa program saat ini telah hadir untuk mendukung kualitas *website*. Jenis jenis bahasa program yang banyak dipakai para desainer *website* antara lain HTML, ASP, PHP, JSP, Java Scripts, Java applets dsb. Bahasa dasar yang dipakai setiap situs adalah HTML sedangkan PHP, ASP, JSP dan lainnya merupakan bahasa pendukung yang bertindak sebagai pengatur dinamis, dan interaktifnya situs. Bahasa program ASP, PHP, JSP atau lainnya bisa dibuat sendiri. Bahasa program ini biasanya digunakan untuk membangun portal berita, artikel, forum diskusi, buku tamu, anggota organisasi, email, *mailing list* dan lain sebagainya yang memerlukan update setiap saat.

2.6 HTML

Hypertext Markup Language (HTML) adalah bahasa markup yang umum digunakan untuk membuat halaman web. Sebenarnya HTML bukanlah sebuah Bahasa pemrograman. Apabila di tinjau dari namanya, HTML merupakan bahasa markup atau penandaan terhadap sebuah dokumen teks. Tanda tersebut di gunakan untuk menentukan format atau style dari teks yang di tandai.

HTML dibuat oleh Tim Berners-Lee Robert ketika masih bekerja untuk CERN (CERN adalah lembaga penelitian fisika energi tinggi di Jenewa) dan dipopulerkan pertama kali oleh *browser* Mosaic. Selama awal tahun 1990 HTML mengalami perkembangan yang sangat pesat. Setiap pengembangan HTML pasti akan menambahkan kemampuan dan fasilitas yang lebih baik dari versi sebelumnya.

Sebelum suatu HTML disahkan sebagai suatu dokumen HTML standar, ia harus disetujui dulu oleh W3C untuk dievaluasi secara ketat. Setiap terjadi perkembangan suatu versi HTML, maka mau tak mau *browser* pun harus memperbaiki diri agar bias mendukung kode-kode HTML yang baru tersebut. Sebab jika tidak, *browser* tak akan bisa menampilkan HTML tersebut.

Mengetahui singkatannya saja tidak akan cukup. Karena itu saya akan menyajikan arti dari kata kata HTML tersebut.

1. *HyperText* adalah metode dimana kita "berpindah" disekeliling web, dengan mengklik sebuah teks yang bernama *hyperlink*. *Hyperlink* adalah sebuah teks khusus di internet, dimana saat teks tersebut diklik, akan membawa kita ke halaman web selanjutnya/halaman web lain yang telah ditentukan.
2. Markup adalah hal yang dilakukan oleh tag HTML kepada teks yang ada didalamnya. HTML menandai teks yang berada didalamnya sebagai tipe teks tertentu. Misalnya saja jika kita menandai sebuah teks dengan tag html <i>, maka teks tersebut akan berubah menjadi italic (huruf yang miring). Sedangkan jika kita menandainya dengan , maka teks tersebut akan berubah menajadi bold (huruf tebal).
3. Language yang berarti bahasa. HTML adalah sebuah bahasa, yang memiliki kata kata berupa kode dan syntax seperti bahasa yang lain.

2.7 CSS

CSS (*Cascading Style Sheet*) adalah suatu cara untuk membuat format atau *layout* halaman web menjadi lebih menarik dan mudah dikelola. CSS muncul karena sulitnya mengatur *layout* tampilan dokumen yang dibuat dengan HTML murni meskipun telah menggunakan berbagai kombinasi format. CSS bukanlah pengganti kode html, namun merupakan pendukung dari *file* html.

CSS dibuat untuk memisahkan konten utama dengan tampilan dokumen yang meliputi *layout*, warna da font. Pemisahan ini dapat meningkatkan daya akses konten pada web, menyediakan lebih banyak fleksibilitas dan kontrol dalam spesifikasi dari sebuah karakteristik dari sebuah tampilan, memungkinkan untuk membagi halaman untuk sebuah formatting dan mengurangi kerumitan dalam penulisan kode dan struktur dari konten, contohnya teknik *tableless* pada desain web. (Madcoms, 2009)

2.8 PHP

2.8.1 Defenisi PHP

PHP atau *Hypertext* Preprocessor adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memprogram situs web dinamis.

2.8.2 Sejarah PHP

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

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

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

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP: *Hypertext Preprocessing*. (Trimarsiah dan Muhajir, 2017)

2.9 Java Sricpt

Javascript merupakan sebuah bahasa pemrograman populer yang di dukung oleh semua web *browser* dan web tools lainnya.yang memungkinkan fungsi-fungsi interaktif yang bisa ditambahkan ke dalam halaman web yang statis. Dengan javascript dapat membuat interaksi pengguna dan aplikasi lebih interaktif. Javascript berkembang dari bahasa netscape live script.Javascript tertanam kedalam halaman web karena hal itu hanya bisa di eksekusi dengan mengambil sebuah halaman dari semua situs web.Hal ini tidak bisa digunakan untuk penyelidikan sumber komputer. Java adalah sebuah bahasa pemrograman yang bersifat full-blown yang bisa memanipulasikan banyak sumber dalam komputer.bagaimana pun juga kedua javaScript routines dan program java (applet) diaktivasi dari sebuah halaman web yang bersifat sanboxed dan tidak mempunyai pemerintahan penuh dalam mesin (Giffin, 2012).

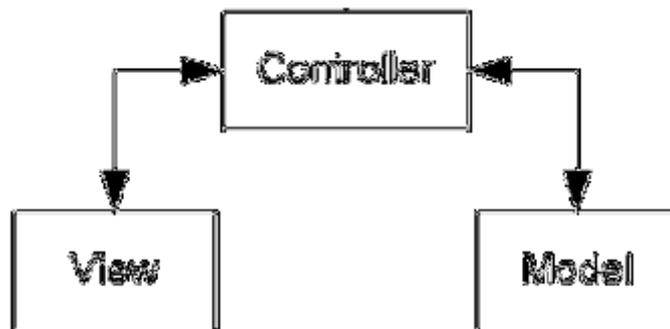
2.10 JQuery

JQuery merupakan tipe dari *library* Javascript (Prototype, Ext Core dan moo.fx merupakan contoh lain) yang dibuat untuk membuat Javascript lebih mudah diakses dan digunakan. JQuery mensesederhanakan sintax Javascript dan memberikan interaksi yang lebih baik antara Javascript dan bahasa pengembangan web yang lainnya. JQuery memberikan akses yang lebih mudah ke DOM (Document Object *Model*) dan memungkinkan pembuatan animasi, widget dan segmen web yang dinamis (AJAX) dibandingkan dengan Javascript saja. JQuery juga merupakan *library* Javascript yang paling populer digunakan sekarang oleh banyak situs web yang terkenal(JSripters.com, 2012).

Fitur lain yang menarik dari JQuery adalah kemampuan untuk mengimplementasikan fungsionalitas AJAX. Dengan AJAX dimungkinkan untuk menyelesaikan beberapa fungsi tanpa meninggalkan atau me-refresh halaman. AJAX saat ini digunakan dimana-dimana di berbagai situs karena lebih cepat dan alternatif yang baik (JScripeters.com, 2012).

2.11 MVC dan CodeIgniter

Pola pemrograman *Model View Controller* (MVC) merupakan pola pemrograman yang memisahkan aplikasi kedalam 3 bagian yang membuat program *logic* dan *presentation logic* tidak tercampur aduk menjadi satu. MVC dikembangkan pada tahun 1970-an oleh Prof. Trygve Reenskaug dari Smalltalk. Alasan sebuah pola MVC ini dikembangkan karena pola arsitektur yang pada saat itu digunakan adalah *Rapid Application Prototyping* yang tidak memisahkan aplikasi dengan presentasi, sehingga design untuk GUI disamakan dengan desain aplikasi (Deacon, 2009 : 1).



Gambar 2.1 Arsitektur *Model View Control*

Model merupakan hal yang bersifat singular di dalam aplikasi. Dalam istilah object oriented *model* terdiri dari *class - class*. *Model* merupakan hal yang berguna untuk menjadi objek untuk memecahkan masalah. *Model* tidak boleh memiliki *programming logic* yang bersangkutan dengan hal diluar *model* itu sendiri (Deacon, 2009 : 2).

View adalah hal yang bersifat plural dalam aplikasi. *View* bisa berupa GUI, CLI, maupun API. *View* tidak boleh memanggil *model*, hal - hal yang diterima dari *view* bergantung pada *controller* (Deacon, 2009 : 2).

Controller merupakan objek yang berguna untuk memanipulasi *view*, singkatnya *controller* untuk menerima dan memproses *input* dan *view* untuk menampilkan hasil dari *controller*. Didalam *controller* terdapat *programming logic* yang menggunakan *model - model* untuk menyelesaikan masalah dan setelah semuanya selesai diproses maka akan dikirim ke *view* untuk dipresentasikan (Deacon, 2009 : 2).



Gambar 2.2 Logo CodeIgniter

(Sumber : Plimbi.com)

CodeIgniter adalah sebuah web *application framework* yang digunakan untuk membangun aplikasi PHP dinamis yang dibangun menggunakan konsep *Model View Controller development pattern*. CodeIgniter menyediakan berbagai macam *library* yang dapat mempermudah dalam pengembangan dan termasuk *framework* tercepat dibandingkan dengan *framework* lainnya. Keuntungannya adalah data dapat dirubah tanpa merubah data lain. Dapat menampilkan banyak macam tampilan dari data yang sama tergantung bagaimana *view* itu dibuat. Kekurangannya adalah diperlukan kode tambahan dan kompleksitas tambahan pada *model* dan interaksi yang simple.

2.12 Database

Database kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut.

Database (basis data) merupakan kumpulan data yang saling berhubungan. Hubungan antara data dapat ditunjukkan dengan adanya *field*/kolom kunci dari tiap *field*/tabel yang ada. Dalam satu *file* atau table terdapat record-record yang sejenis, sama besar, sama bentuk, yang merupakan satu kumpulan entitas yang seragam. Satu record (umumnya digambarkan sebagai baris data) terdiri dari *field* yang saling berhubungan menunjukkan bahwa *field* tersebut dalam satu pengertian yang lengkap dan disimpan dalam satu *record*. Database mempunyai beberapa kriteria penting yaitu:

1. Bersifat data oriented dan bukan program oriented
2. Dapat digunakan oleh beberapa program aplikasi tanpa perlu mengubah basis datanya.
3. Dapat dikembangkan dengan mudah, baik volume maupun strukturnya.
4. Dapat memenuhi kebutuhan sistem-sistem baru secara mudah.
5. Dapat digunakan dengan cara-cara yang berbeda.(Amelia, 2011)

2.13 MySQL

2.13.1 Defenisi MySQL

MySQL (*My Structured Query Language*) adalah sebuah program pembuat database yang bersifat *open source* dan berjalan di semua *platform* baik Windows maupun Linux. Selain itu, MySQL juga merupakan program pengakses database yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *Multi User* (Banyak Pengguna).



Gambar 2.3 Logo MySQL

(Sumber : www.mysql.com)

2.13.2 Sejarah MySQL

Sebagai suatu *software* pengelola database tentunya MySQL ini didirikan oleh suatu lembaga atau perusahaan khusus. Perusahaan itu adalah Oracle yang memang menjadi salah satu perusahaan pengelola *software* terbesar di dunia. Pada awalnya seorang programmer komputer asal negeri Swedia yang bernama Michael “Monty” Widenius mengembangkan suatu sistem database sederhana yang bernama UNIREG. Pada awalnya tools ini menggunakan low level database engine dengan indexing. Michael “Monty” Widenius juga masih bekerja pada salah satu perusahaan di Swedia yang bernama TcX.

2.13.3 Fungsi MySQL

MySQL digunakan untuk membuat dan mengelola suatu database secara terstruktur dan otomatis menggunakan suatu Bahasa khusus. Namun lebih jelasnya lagi MySQL ini memberikan kemudahan bagi para pengguna yang ingin mengelola suatu data yang berisi informasi secara *String (text based)* dan dapat diakses secara pribadi maupun untuk umum dalam suatu web. Hampir semua host atau penyedia server web memberikan fasilitas MySQL untuk para developer web yang menginginkan pengelolaan database di *websitenya*. (Sulastri, 2008)

2.14 XAMPP

XAMPP adalah *software* web server apache yang di dalamnya tertanam server MySQL yang didukung dengan bahasa pemrograman PHP untuk membuat *website* yang dinamis. XAMPP sendiri berbasis *open source*, yang mendukung dua system operasi yaitu windows dan Linux. Untuk linux dalam proses penginstalannya menggunakan command line sedangkan untuk windows dalam proses penginstalannya menggunakan *interface* grafis sehingga lebih mudah dalam penggunaan XAMPP di Windows di banding dengan Linux. Didalam XAMPP ada 3 komponen utama yang di tanam di dalamnya yaitu web server Apache, PHP, dan MySQL.



Gambar 2.4 Logo XAMPP

2.15 Notepad++

Notepad++ adalah sebuah text editor yang sangat berguna dalam membuat program. Notepad++ menggunakan komponen Scintilla untuk menampilkan teks dan berkas kode sumber berbagai bahasa pemrograman yang berjalan diatas sistem operasi Mirosoft Windows. Notepad++ adalah salah satu program yang digunakan untuk melakukan editor seperti HTML, PHP, Java Script, CSS dan lain-lain. Ada beberapa program editor yang dapat dijadikan referensi antara lain Notepad++, Notepad, Komodo, PSPad, PHPEdit, Crimson, ConTXT dan lain-lainnya (Ardhana, 2012).



Gambar 2.5 Logo Notepad++

(Sumber : techcress.com)

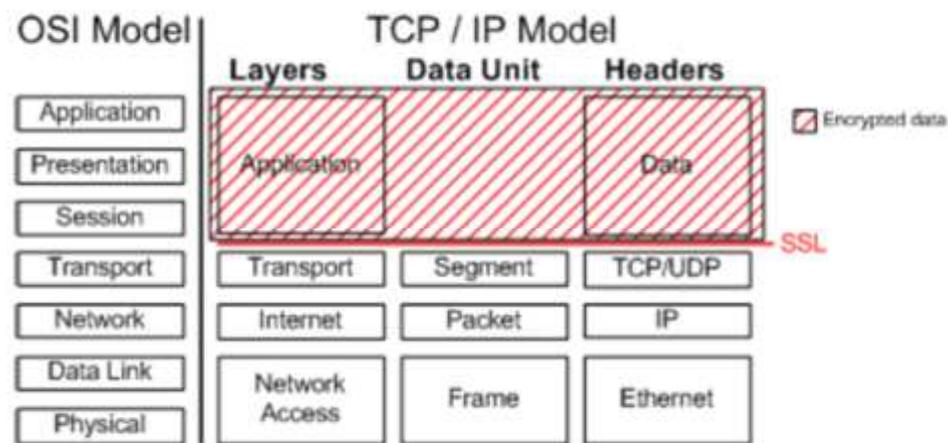
2.16 SMTP

Simple Mail Transfer *Protocol* adalah protokol standar untuk pengiriman electronic mail (email) di internet. SMTP adalah protokol yang cukup sederhana, berbasis teks dimana *protocol* ini menyebutkan satu atau lebih penerima email untuk kemudian diverifikasi. Jika penerima email valid, maka email akan segera dikirim. SMTP menggunakan standart port 25, port 465, dan port 587 yang dapat dihubungi melalui program telnet.

2.17 SSL dan TLS

SSL (*Secure Socket Layer*) yang juga dikenal dengan TLS (*Transport Layer Security*) adalah *protocol* yang awalnya diciptakan oleh *Netscape Communication Corporation* yang menjamin suatu kemudahan, integritas dan autentikasi data ketika internet berkembang pesat pada tahun 1990an. Tiap orang dapat berpartisipasi dalam komunikasi terenkripsi menggunakan *browser* modern. SSI/TLS adalah teknologi yang telah digunakan secara luas pada web untuk perbankan, e-commerce, dan aplikasi-aplikasi lainnya yang dimana privasi dan keamanan dibutuhkan. Teknologi ini telah dikontrol, didebug, dites, dan dikembangkan baik oleh perusahaan *open source* maupun perusahaan berhak cipta.

SSL/TLS berada dibawah *protocol* aplikasi, dan dapat digunakan oleh hampir semua aplikasi. Para pelancar internet mengetahuinya dengan `https://` yang berarti koneksi terenkripsi dari `http://`. Dari *OSI model* standart, *protocol* SSL berada diantara *application layer* dan *transport layer*.



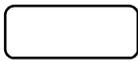
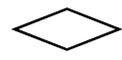
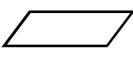
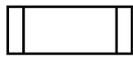
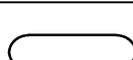
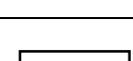
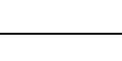
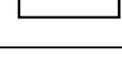
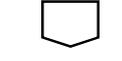
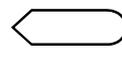
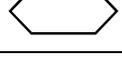
Gambar 2.6 SSL pada OSI Layer

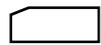
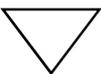
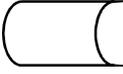
(Sumber : <http://www.openmaniak.com/openvpn.php>)

2.18 Flow Chart

Menurut I Gusti Ngurah Suryantara (2009), badan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi

Tabel 2.1. Simbol Diagram *Flowchart*

No.	Simbol	Nama Simbol	Keterangan
1.		<i>Alternate Process</i>	Menyatakan segala jenis operasi yang diproses dengan menggunakan mesin yang memiliki keyboard
2.		<i>Decision</i>	suatu penyelesaian kondisi dalam program
3.		<i>Data</i>	Mewakilik data <i>input</i> atau <i>output</i>
4.		<i>Predefined Process</i>	Suatu operasi yang rinciannya di tunjukkan di tempat lain
5.		<i>Document</i>	Document <i>input</i> dan <i>output</i> baik untuk proses manual, mekanik atau computer
6.		<i>Terminator</i>	Untuk menunjukkan awal dan akhir dari suatu proses
7.		<i>Process</i>	Kegiatan proses dari operasi program computer
8.		<i>Manual Input</i>	<i>Input</i> yang menggunakan <i>online keyboard</i>
9.		<i>Conector</i>	Penghubung ke halaman yang masih sama
10.		<i>Off-Page Connector</i>	Penghubung ke halaman lain
11.		<i>Display</i>	<i>Output</i> yang ditampilkan di monitor
12.		<i>Delay</i>	Menunjukkan penundaan
13.		<i>Preparation</i>	Memberi nilai awal suatu besaran
14.		<i>Manual Operation</i>	Pekerjaan manual

15.		<i>Card</i>	<i>Input</i> atau <i>output</i> yang menggunakan kartu
16.		<i>Punch Tape</i>	<i>Input</i> atau <i>output</i> menggunakan pita kertas berlubang
17.		<i>Merge</i>	Penggabungan atau penyimpanan beberapa proses atau informasi sebagai salah satu
18.		<i>Direct Access Storage</i>	<i>Input</i> atau <i>output</i> menggunakan drum magnetic
19.		<i>Magnetic Disk</i>	<i>Input</i> atau <i>output</i> menggunakan <i>hard disk</i>
20.		<i>Sequential Access Storage</i>	<i>Input</i> atau <i>output</i> menggunakan pita magnetic
21.		<i>Sort</i>	Proses pengurutan data di luar computer
22.		<i>Stored Data</i>	<i>Input</i> atau <i>output</i> menggunakan <i>diskette</i>
23.		<i>Extract</i>	Proses dalam jalur parallel
24.		<i>Arrow</i>	Menyatakan jalan atau arus suatu proses
25.		<i>Summing Junction</i>	Untuk berkumpul beberapa cabang sebagai proses tunggal
26.		<i>Or</i>	Proses menyimpang dalam dua proses