

BAB II

LANDASAN TEORI

Pada bab ini akan dijelaskan dalam pembuatan dasar tentang teori yang digunakan sebagai acuan dalam analisis dan perancangan "Pembuatan Website Jejaring Sosial untuk Komunitas Jomblo Katholik Indonesia dengan php dan javascript".

II.1 Jomblo

Jomblo adalah suatu istilah yang berarti bujangan. Bujangan dapat diartikan sebagai pria dewasa yang tidak memiliki istri dan dapat juga diartikan sebagai wanita dewasa yang tidak memiliki suami. Akan tetapi, jomblo dapat diartikan dalam konsep pemahaman yang lebih luas dari sekedar arti bujangan. Jomblo tidaklah harus seorang pria dan wanita dewasa yang belum menikah, melainkan dapat diartikan juga sebagai seorang pria dan wanita yang belum memiliki pasangan atau dengan kata lain pacar. Oleh karena itu, seseorang dikatakan jomblo jika belum menikah atau belum memiliki pacar. Komunitas Jomblo Katholik Indonesia adalah sebuah komunitas yang berawal dari keprihatinan terhadap perilaku kaum muda katolik yang mulai meninggalkan iman katoliknya karena menikah dengan pasangan yang berbeda agama dan banyaknya kaum muda katolik sangat sulit menemukan pasangan, walaupun mereka berada dalam satu komunitas gereja. Maka teretuslah ide untuk membuat sebuah komunitas berbasis iman katolik yang menjadi benteng terakhir iman katolik. Komunitas Jomblo

Katolik Indonesia (KJK) mempunyai tujuan sebagai motivator, fasilitator bagi kaum muda katolik untuk menemukan pasangan hidup yang seiman. Oleh karena itu pada bulan Mei tahun 2009 di Salatiga, Jawa Tengah dengan gagasan dan konsep yang jelas yaitu mengumpulkan kaum muda katolik yang belum mempunyai pacar atau pasangan untuk bisa berkumpul, berteman dan memperluas jaringan pertemanan yang seiman, berdirilah Komunitas Jomblo Katholik Indonesia (KJK Indonesia). Pada awal mulanya ide ini kami realisasikan lewat jejaring social facebook yang hanya diketahui dan diminati oleh beberapa orang saja. Namun pada perkembangannya ternyata banyak kaum muda katolik bergabung didalamnya dari hari ke hari. Dalam jangka waktu beberapa bulan kemudian dalam Komunitas Jomblo Katolik (KJK) sudah terbentuk beberapa Regio yang tersebar di berbagai kota besar di seluruh Indonesia. Realisasi pembentukan Komunitas Jomblo Katolik (KJK) di setiap kota ini berdampak positif, karena banyak kaum muda katolik yang menemukan pasangan hidup yang seiman lewat Komunitas Jomblo Katolik (KJK). Hal ini menyebabkan banyak kota-kota lain yang memiliki masalah yang sama ingin mendirikan Komunitas Jomblo Katolik (KJK). Atas dorongan Rm.Aristanto,MSF yang berkantor di Jl. Guntur No.20 Semarang (Keuskupan Agung Semarang), maka eksistensi Komunitas Jomblo Katolik (KJK) semakin berkibar. Adapun anggota Komunitas Jomblo Katolik (KJK) sekarang berjumlah 7.500 anggota di seluruh Indonesia terdiri dari beberapa kota antara lain Medan, Lampung, Jakarta, Bogor, Bandung, Cirebon, Semarang, Salatiga,

Magelang, Jogjakarta, Solo, Surabaya, Malang, Bali, Manado, Makasar, sulawesi, Papua dan beberapa kota lain di Indonesia.

II.2 Komunitas dan Jejaring Sosial

II.2.1 Komunitas

Komunitas adalah sebuah kelompok sosial dari beberapa organisme yang berbagi lingkungan, umumnya memiliki ketertarikan yang sama. Dalam komunitas manusia, individu-individu di dalamnya dapat memiliki maksud, kepercayaan, sumber daya, preferensi, kebutuhan, risiko dan sejumlah kondisi lain yang serupa.

Komunitas berasal dari bahasa Latin *communitas* yang berarti "kesamaan", kemudian dapat diturunkan dari *communis* yang berarti "sama, publik, dibagi oleh semua atau banyak" *communis* berasal dari sebuah kombinasi prefix() Latin *con-* (yang berarti bersama-sama), dan kata *munis* (harus melakukan kinerja pelayanan).

Ahli sosiologi Jerman Ferdinand Tonnies menjelaskan sebuah perbedaan kecil diantara arti komunitas (*Gemeinschaft*) dan masyarakat (*Gesellschaft*). Pada karyanya 1887 *Gemeinschaft and Gesellschaft*, Tonnies berpendapat bahwa komunitas dirasa menjadi bagian yang lebih kecil dan entitas sosial yang lebih menyatu dalam konteks masyarakat yang lebih besar, menjadi satu kesatuan yang memiliki tujuan yang sama. Tonnies menambahkan bahwa keluarga dan hubungan kekeluargaan merupakan ekspresi yang sempurna dalam komunitas, tetapi selain itu

karakteristik seperti tempat atau kepercayaan, dapat juga menghasilkan komunitas (*gemeinschaft*).

II.2.2 Jejaring Sosial

Jejaring sosial adalah struktur sosial yang terdiri dari elemen-elemen individual atau organisasi. Jejaring ini menunjukkan jalan dimana mereka berhubungan karena kesamaan sosialitas, mulai dari mereka yang dikenal sehari-hari sampai dengan keluarga. Istilah ini diperkenalkan oleh profesor J.A. Barnes di tahun 1954. Jejaring sosial adalah suatu struktur sosial yang dibentuk dari simpul-simpul (yang umumnya adalah individu atau organisasi) yang diikat dengan satu atau lebih tipe relasi spesifik seperti nilai, visi, ide, teman, keturunan, dan lainnya.

Jejaring sosial sebenarnya bentuk baru komunitas virtual atau komunitas maya di internet yang saling terhubung dengan cepat. Ini berbeda dengan jejaring sosial lima tahun yang lalu yang mungkin lebih dikenal sebagai forum diskusi, *chat*, *messenger* atau *milis* dimana pola komunikasinya terbatas hanya dalam forum tersebut saja.

Disebut jejaring karena kemampuannya untuk saling terhubung dengan cepat antara satu domain komunitas dengan komunitas lainnya. Misalnya, kalau menggunakan tools status di Plurk.com, maka status tersebut dapat didistribusikan ke facebook, tumblr, twitter, multiply. Bahkan ada yang seolah-olah menjadi konsolidator semua domain komunitas sehingga fungsinya lebih praktis.

Sejak komputer dapat dihubungkan satu dengan lainnya dengan adanya internet banyak upaya awal untuk mendukung jejaring sosial melalui komunikasi antar komputer. Situs jejaring sosial diawali oleh Classmates.com pada tahun 1995 yang berfokus pada hubungan antar mantan teman sekolah dan SixDegrees.com pada tahun 1997 yang membuat ikatan tidak langsung. Dua model berbeda dari jejaring sosial yang lahir sekitar pada tahun 1999 adalah berbasis kepercayaan yang dikembangkan oleh Epinions.com, dan jejaring sosial yang berbasis pertemanan seperti yang dikembangkan oleh Uskup Jonathan yang kemudian dipakai pada beberapa situs UK regional di antara 1999 dan 2001. Inovasi meliputi tidak hanya memperlihatkan siapa berteman dengan siapa, tetapi memberikan pengguna kontrol yang lebih akan isi dan hubungan. Pada tahun 2005, suatu layanan jejaring sosial MySpace, dilaporkan lebih banyak diakses dibandingkan Google dengan Facebook, pesaing yang tumbuh dengan cepat. Jejaring sosial mulai menjadi bagian dari strategi internet bisnis sekitar tahun 2005 ketika Yahoo meluncurkan Yahoo! 360°. Pada bulan juli 2005 News Corporation membeli MySpace, diikuti oleh ITV (UK) membeli Friends Reunited pada Desember 2005. Diperkirakan ada lebih dari 200 situs jejaring sosial menggunakan model jejaring sosial ini.

Komunitas maya / e-komunitas / komunitas online adalah sekelompok orang yang berinteraksi lewat media seperti surat, telepon, email atau *usenet*. Apabila mekanisme yang digunakan adalah jaringan komputer maka

disebut komunitas online. Komunitas maya dan online juga merupakan suatu bentuk komunitas diantara orang yang saling mengenal di dalam kehidupan nyata. Ada begitu banyak sarana yang digunakan dalam perangkat lunak sosial (*social software*) secara terpisah maupun terpadu, termasuk *chatroom* berbasis *text*, dan forum yang menggunakan *audio*, *text*, *video* dan *avatar*. Perkembangan jaringan sosial berbasis internet telah menyebabkan perubahan sosial teknis yang signifikan.

Banyak layanan jejaring sosial berbasiskan web yang menyediakan kumpulan cara yang beragam bagi pengguna untuk dapat berinteraksi seperti *chat*, *messaging*, *email*, *video*, *chat* suara, *share file*, *blog*, diskusi grup, dan lain-lain. Umumnya jejaring sosial memberikan layanan untuk membuat biodata dirinya. Pengguna dapat meng-*upload* foto dirinya dan dapat menjadi teman dengan pengguna lainnya. Beberapa jejaring sosial memiliki fitur tambahan seperti pembuatan grup untuk dapat saling *sharing* didalamnya. Perangkat lunak sosial yang saat ini populer digunakan dalam interaksi jejaring sosial antara lain : *flickr*, *friendster*, *youtube*, *facebook*, *my space*, *skype*, *yahoo messenger*, *mirc*.

II.3 Website

Website / *web* / *situs* / *site* secara terminologi adalah kumpulan dari halaman-halaman, yang biasanya terangkum dalam sebuah domain atau subdomain, yang

tempatnyanya berada di dalam *World Wide Web* (WWW) di Internet. Walaupun istilah *web / site / situs* sudah secara umum dipakai, tetapi untuk *Associated Press Stylebook*, *Reuters*, *Microsoft*, *Academia*, dan kamus-kamus yang ada, penulisan yang dipakai adalah dengan menggunakan dua kata, yaitu *web site*. Hal ini disebabkan *web* bukanlah terminologi umum, tetapi kependekan dari *World Wide Web* (WWW).

Sebuah halaman *website* adalah dokumen yang ditulis dalam format *HTML* (*Hyper Text Markup Language*), yang hampir selalu bisa diakses melalui *HTTP*, yaitu protokol yang menyampaikan informasi dari *server website* untuk ditampilkan kepada para pemakai melalui *web browser*. Semua publikasi dari *website-website* tersebut dapat membentuk sebuah jaringan informasi yang sangat besar. Halaman-halaman dari *website* akan bisa diakses melalui sebuah *URL* yang biasa disebut *Homepage*. *URL* ini mengatur halaman-halaman situs untuk menjadi sebuah hirarki, meskipun, *hyperlink-hyperlink* yang ada di halaman tersebut mengatur para pembaca dan memberitahu mereka susunan keseluruhan dan bagaimana arus informasi ini berjalan.

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

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

2. Rumah tempat website (*Web hosting*)
3. Bahasa Program (*Scripts Program*)
4. Desain website

Pengertian Nama *domain* atau biasa disebut dengan *Domain Name* atau *URL* adalah alamat unik diinternet yang digunakan untuk mengidentifikasi sebuah *website*, atau dengan kata lain domain name adalah alamat yang digunakan untuk menemukan sebuah *website* diinternet. Nama *domain* diperjualbelikan secara bebas diinternet dengan status sewa per-tahun. Nama *domain* sendiri mempunyai identifikasi *ekstensi*/akhiran sesuai dengan kepentingan dan lokasi keberadaan *website* tersebut. Contoh nama domain dengan *ekstensi* internasional adalah com, net, org, info, biz, name, ws. Contoh nama *domain* dengan *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).

Pengertian tentang *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/dimiliki, 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.

Bahasa pemrograman 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 dan lainnya. 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 perubahan informasi setiap saat. Sedangkan desain sangat berpengaruh kepada penilaian pengunjung akan bagus tidaknya sebuah *website*. Semakin baik disain dari sebuah *website*, maka akan dihasilkan situs yang semakin berkualitas dan menarik untuk dikunjungi.

Beberapa *website* membutuhkan *subskripsi* (data masukan) agar para user bisa mengakses sebagian atau keseluruhan isi *website* tersebut. Contohnya, ada beberapa

situs-situs bisnis, situs-situs e-mail gratisan, yang membutuhkan *subkripsi* agar kita bisa mengakses situs tersebut.

Penemu *website* adalah Sir Timothy John "Tim" Berners-Lee, sedangkan *website* yang tersambung dengan jaringan, pertamakali muncul pada tahun 1991. Maksud dari Tim ketika membuat *website* adalah untuk mempermudah tukar menukar dan memperbarui informasi kepada sesama peneliti di tempat dia bekerja. Pada tanggal 30 April 1993, CERN (tempat dimana Tim bekerja) menginformasikan bahwa *WWW* dapat digunakan secara gratis oleh semua orang.

Sebuah *website* bisa berupa hasil kerja dari perorangan atau individu, atau menunjukkan kepemilikan dari sebuah organisasi, perusahaan, dan biasanya *website* itu menunjukkan beberapa topik khusus, atau kepentingan tertentu. Sebuah *website* bisa berisi *hyperlink* (pranala) yang menghubungkan ke *website* lain, jadi, kadangkala perbedaan antara *website* yang dibuat oleh individu perseorangan dengan *website* yang dibuat oleh organisasi bisnis bisa saja tidak kentara.

Website ditulis, atau secara dinamik di konversi menjadi *HTML* dan diakses melalui sebuah program software yang biasa disebut dengan *web browser*, yang dikenal juga dengan *HTTP Client*. Halaman *web* dapat dilihat atau diakses melalui jaringan komputer dan internet, perangkatnya bisa saja berupa komputer pribadi, laptop, PDA ataupun telepon selular.

Sebuah *website* dibuat didalam sebuah sistem komputer yang dikenal dengan *server web*, juga disebut *HTTP Server*, dan pengertian ini juga bisa menunjuk pada *software* yang dipakai untuk menjalankan sistem ini, yang kemudian menerima lalu mengirimkan halaman-halaman yang diperlukan untuk merespon permintaan dari pengguna. Apache adalah piranti lunak yang biasa digunakan dalam sebuah *webserver*, kemudian setelah itu adalah Microsoft Internet Information Services (IIS).

Website dibedakan atas dua macam. *Website statik* dan *website dynamic*. Sebuah *website statik*, adalah salah satu bentuk *website* yang isi didalam *website* tersebut tidak dimaksudkan untuk di *update* secara berkala, dan biasanya di kelola secara manual oleh beberapa orang yang menggunakan *software editor*. Ada 3 tipe kategori *software editor* yang biasa dipakai untuk tujuan mengelola *website* statik :

1. Elemen 1 Penyunting teks. Contohnya adalah Notepad atau *TextEdit*, dimana *HTML* diubah didalam program editor tersebut.
2. Elemen 2 *WYSIWYG editor*. Contohnya Microsoft Frontpage dan Macromedia Dreamweaver, dimana situs di edit menggunakan *GUI* (*Graphical User Interface*) dan format *HTML* ini secara otomatis di *generate* oleh editor ini.
3. Elemen 3 Editor yang sudah memiliki *templat*, contohnya Rapidweaver dan iWeb, dimana, editor ini membolehkan user untuk membuat dan meng-*update*

website-nya langsung ke *server web* secara cepat, tanpa harus mengetahui apapun tentang *HTML*. Mereka dapat memilih *templat* yang sesuai dengan keinginan mereka, menambah gambar atau obyek, mengisinya dengan tulisan, dan dengan sekejap mereka sudah dapat membuat situs *web* tanpa harus melihat sama sekali kode-kode *HTML*.

Sebuah *website dynamic* adalah *website* yang secara berkala, informasi didalamnya berubah, atau *website* ini bisa berhubungan dengan user dengan berbagai macam cara atau metode (*HTTP cookies* atau Variabel *Database*, sejarah kunjungan, variabel sesi dan lain-lain) bisa juga dengan cara interaksi langsung menggunakan *form* dan pergerakan mouse. Ketika *web server* menerima permintaan dari user untuk memberikan halaman tertentu, maka halaman tersebut akan secara otomatis diambil dari media penyimpanan sebagai respon dari permintaan yang diminta oleh user. Sebuah situs dapat menampilkan dialog yang sedang berlangsung diantara dua user, memantau perubahan situasi, atau menyediakan informasi yang berkaitan dengan user tersebut.

Terdapat banyak jenis sistem software yang dapat dipakai untuk membuat *Dynamic Web System* dan *Situs Dynamic*, beberapa diantaranya adalah ColdFusion (CFM), Active Server Pages (ASP), Java Server Pages (JSP) dan PHP. Situs juga bisa termasuk didalamnya berisi informasi yang diambil dari satu atau lebih database atau bisa juga menggunakan teknologi berbasis *XML*, contohnya adalah *RSS*.

Isi situs yang statis juga secara periodik dibuat, atau apabila ada keadaan dimana dibutuhkan untuk dikembalikan kepada keadaan semula, maka akan di *generate*, hal ini untuk menghindari kinerjanya supaya tetap terjaga. *Plugin* tersedia untuk menambah banyaknya *feature* dan kemampuan dari *web browser*, dimana, *plugin* ini dipakai untuk membuka *content* yang biasanya berupa cuplikan dari gambar bergerak (*active content*) contohnya adalah Flash, Shockwave atau applets yang ditulis dalam bahasa JAVA. *Dynamic HTML* juga disediakan supaya dapat secara interaktif dan *realtime*, memperbaharui *web page* tersebut. Halaman yang diubah, tak perlu di-*load* atau *reload* agar perubahannya dapat dilihat, biasanya perubahan yang dilakukan menggunakan DOM dan Javascript yang sudah tersedia pada semua *Web Browser*.

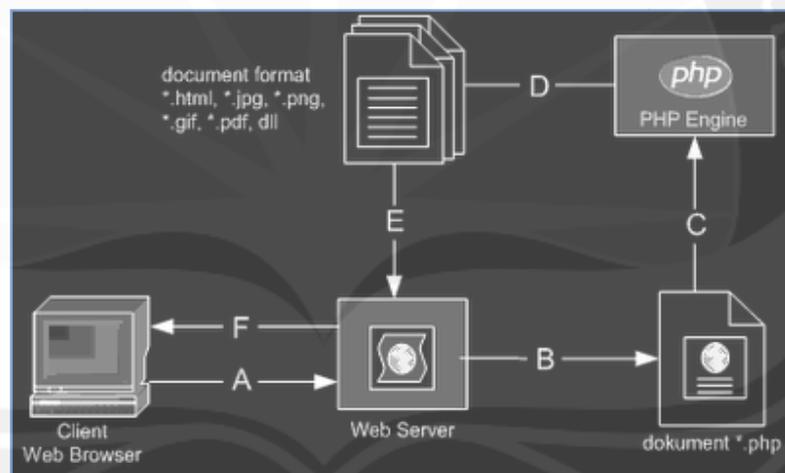
II.4 Server Side Script dan Client Side Script

II.4.1 Server Side Script

Server side scripting merupakan dokumen-dokumen yang digunakan dalam membangun suatu aplikasi internet yang dijalankan pada sisi server dan dikirimkan ke *browser* dalam bentuk *HTML*. Jika yang diinginkan oleh seorang user adalah file yang mengandung perintah *server side* maka *server web* akan menjalankan dahulu program tersebut lalu mengirimkannya kembali ke *browser* dalam bentuk *HTML* sehingga dapat diterjemahkan oleh *browser*. Berikut contoh *server side scripting*: PHP (*PHP pre processor*), ASP (*active server pages*), JSP (*java server pages*). *Server*

Side Scripting digunakan untuk membuat halaman web dinamis. Hal ini disebabkan karena *server side scripting* memiliki kelebihan, diantaranya:

1. Mampu berinteraksi dengan banyak perangkat lunak basis data (database).
2. Mampu mengelola sumber daya yang terdapat di sistem operasi dan perangkat keras komputer.
3. Mampu dijalankan di semua sistem operasi.
4. Aman karena scriptnya di proses di server yang tidak dapat dilihat dari sisi client.



Gambar 2.1 Cara kerja *Server Side Script*

Dari gambar di atas dapat dilihat bagaimana cara kerja *server side script*, yaitu:

- A. Client mengirim request ke *web server* melalui *browser*.
- B. *Web server* menerima request dalam dokumen PHP.
- C. Berkas PHP dikirim ke *PHP Engine* untuk diproses.

- D. *PHP engine* menerjemahkan berkas *PHP* menjadi kode *HTML*
- E. Setelah melalui proses, Berkas kembali dikirim ke *web server*.
- F. *Web server* mengirim berkas ke *client* untuk ditampilkan pada *browser* sehingga bisa dilihat oleh user.

II.4.2 Client Side Script

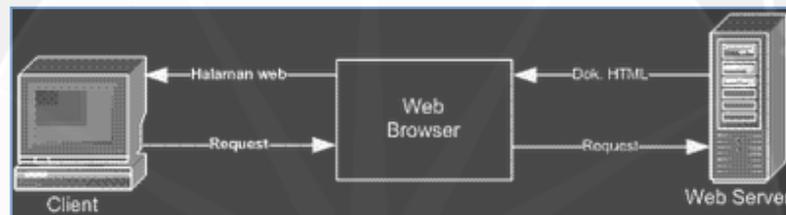
Client side scripting merupakan bahasa pemrograman internet yang akan dieksekusi oleh *browser* dalam format *.html. Biasanya *client side* yang digunakan untuk hal-hal yang membutuhkan interaksi user tetapi data yang ditampilkan tetap sama dengan menggunakan *server side scripting*.

Aplikasi *web* berjalan pada protokol *HTTP*, dan semua protokol di internet selalu melibatkan antara *server* dan *client*. Ketika seseorang mengetikkan suatu alamat di *browser*, maka *browser* akan mengirimkan perintah tersebut ke *web server*. Jika yang diminta oleh *client* adalah file yang mengandung file *client side* maka oleh *server file* tersebut akan langsung dikirimkan ke *browser*.

Client side scripting dikerjakan secara urut dari bagian paling atas *script* sampai bagian paling bawah. Tanpa ada lompatan, perulangan dan sebagainya. *Client side scripting* dapat dilihat melalui *browser* langsung dengan memilih menu *view source*, sehingga keamanan *script* kurang terjaga.

Berikut adalah contoh *client side scripting*: HTML (*hypertext markup language*), java script, XML (*extensible markup language*), CSS (*cascading style sheet*). *Client side scripting* merupakan *script* yang digunakan untuk membuat halaman web statis. Kelebihan *Client-Side-Scripting*

1. Tidak perlu *server* khusus untuk menjalankannya
2. Eksekusi *script* lebih cepat
3. Tidak membebani kinerja *web server*.



Gambar 2.2 Cara Kerja *Client Side Script*

Dari gambar di atas dapat dilihat bagaimana cara kerja *client side script*, yaitu:

1. User me-request halaman web yang berisi *client side scripting* melalui browser.
2. Browser mendapatkan alamat dari web server dan mengidentifikasi alamat yang di-request.
3. Web server mencarikan berkas yang diminta dan memberikan isinya ke browser.
4. Browser mendapatkan isinya segera melakukan proses penerjemahan kode HTML.
5. Halaman web diterima client melalui browser.

II.5 AJAX

Ajax adalah istilah yang diperkenalkan pertama kali oleh Jesse James Garret dari Adaptive Path. Dalam esainya "AJAX : A New Approach to Web Application", Garret menyebutkan bahwa jurang antara aplikasi *desktop* dan aplikasi *web* semakin menipis. Hal tersebut dibuktikan oleh Google dengan munculnya Google Maps dan Google Suggest. Ajax sendiri merupakan akronim dari *Asynchronous Javascript And XML* (Asleson, 2006).

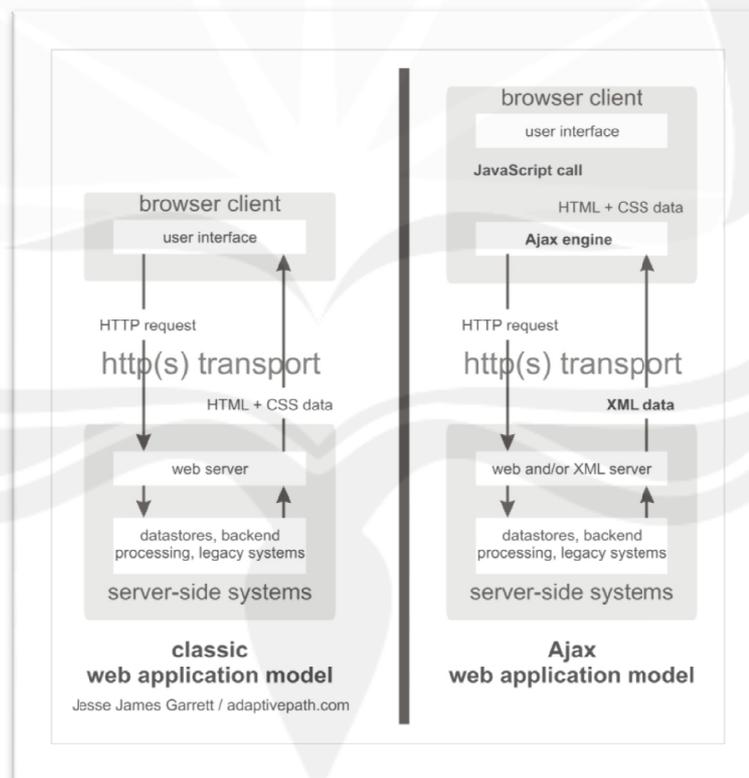
Ajax berbeda dengan aplikasi *web* tradisional yang mengharuskan pengguna menunggu suatu halaman selesai ditampilkan oleh *browser* dan pada saat itu tentu saja pengguna harus menunggu dan tidak bisa melakukan apa-apa, baru kemudian pengguna bisa melakukan interaksi dengan *server*. Sedangkan ajax memiliki cara yang berbeda. Meskipun suatu proses sedang terjadi dalam aplikasi *web*, pengguna tetap dapat melakukan hal yang lain, sebab proses-proses dalam ajax dilakukan secara asinkronus artinya masing-masing proses dapat tidak tergantung oleh proses lain.

Ajax tidak memerlukan perangkat lunak dan perangkat lunak tambahan ataupun berbayar untuk menikmatinya. Sebenarnya, dalam Ajax tidak ada yang baru (Asleson, 2006). Semua yang diperlukan oleh Ajax sudah ada di dalam semua *browser* utama saat ini.

Ajax merupakan kumpulan teknologi-teknologi yang membuat sebuah *browser* bisa berkomunikasi dengan *server*, tanpa harus *me-refresh* semua halaman. Hal ini dimungkinkan karena *browser* melakukan HTTP *request* di

belakang layar secara *asynchronous*. Garret (2005) menyebutkan teknologi-teknologi pembantu Ajax terdiri dari :

1. HTML dan CSS, untuk lapisan presentasi.
2. Document Object Model, untuk tampilan dan interaksi dinamis.
3. XML, sebagai format pertukaran data.
4. XMLHttpRequest, untuk mengambil data secara asynchronous
5. Javascript, untuk menggabungkan semua teknologi diatas.



Gambar 2.3 Perbedaan AJAX Web application dengan Classic Web Application

Dari gambar diatas dapat dilihat bagaimana perbedaan cara kerja antara *classic web application* model dengan *AJAX web application model* yaitu dimana pada *classic web application web browser client* langsung menghubungi *web server* dengan *HTTP request* baru *web server* melaksanakan *request* tersebut dan mengembalikan hasil *request* tersebut ke *client* dalam bentuk *HTML* dan *CSS*. Dengan cara ini akan membutuhkan waktu yang lama karena harus mengirimkan data dalam bentuk *HTML* dan *CSS*, berbeda dengan *Web application* dengan menggunakan *AJAX* dimana digunakan tambahan modul *AJAX* sebagai gerbang dari *web browser client* untuk berinteraksi dengan *web server*. *Web browser client* akan mengeksekusi *javascript* baru kemudian oleh modul *AJAX* akan diteruskan ke *web server* baru di *web server* perintah akan dieksekusi kemudian akan dikembalikan ke *web browser client* dalam bentuk *XML* yang lebih ringan baru kemudian diterjemahkan oleh modul *AJAX* menjadi *HTML* dan *CSS*.

Dengan bekerja sebagai lapisan tambahan diantara *browser* pengguna dan *web server*, *Ajax* menangani komunikasi dengan *server* di belakang layar, melakukan *request* dan memproses data kembalian. Hasilnya diintegrasikan ke halaman yang sedang ditampilkan tanpa memerlukan *refresh* atau *me-load* suatu halaman yang baru (Ballard,2006).

II.6 CSS

Cascading Style Sheets (*CSS*) adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa markup. Penggunaan yang

paling umum dari CSS adalah untuk memformat halaman *web* yang ditulis dengan HTML dan XHTML. Walaupun demikian, bahasanya sendiri dapat dipergunakan untuk semua jenis dokumen XML termasuk SVG dan XUL. Spesifikasi CSS diatur oleh *World Wide Web Consortium* (W3C).

CSS diperkenalkan untuk pengembangan *website* pada tahun 1996. Nama CSS didapat dari fakta bahwa setiap deklarasi *style* yang berbeda dapat diletakkan secara berurutan, yang kemudian akan membentuk hubungan *parent-child* pada setiap *style*. Setelah CSS distandarisasikan, Internet Explorer dan Netscape melepas *browser* terbaru mereka yang telah sesuai atau paling tidak hampir mendekati dengan standar CSS.

CSS digunakan oleh penulis maupun pembaca halaman *web* untuk menentukan warna, jenis huruf, tata letak, dan berbagai aspek tampilan dokumen. CSS digunakan terutama untuk memisahkan antara isi dokumen dengan presentasi dokumen. Pemisahan ini dapat meningkatkan aksesibilitas isi, memberikan lebih banyak keleluasaan dan kontrol terhadap tampilan, dan mengurangi kompleksitas serta pengulangan pada struktur isi.

CSS memungkinkan halaman yang sama untuk ditampilkan dengan cara yang berbeda untuk metode presentasi yang berbeda, seperti melalui layar, cetak, suara (sewaktu dibacakan oleh *browser* basis-suara atau pembaca layar), dan juga alat pembaca braille. Halaman HTML atau XML yang sama juga dapat ditampilkan secara berbeda, baik dari segi gaya tampilan atau skema warna dengan menggunakan CSS.