

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Gerbang adalah tempat keluar atau masuk ke dalam suatu kawasan tertutup yang dikelilingi pagar atau dinding. Gerbang berguna untuk mencegah atau mengendalikan arus keluar-masuknya orang. Gerbang dapat bersifat sederhana hanya berupa bukaan sederhana pada sebuah pagar, maupun dekoratif dan bahkan monumental. Istilah lainnya untuk gerbang adalah pintu dan gapura. Gerbang besar dan kokoh pada sebuah bangunan dapat menjadi sarana pertahanan, misalnya gerbang pada benteng atau kastil. Pintu adalah bagian yang menutup akses lewat melalui rumah gerbang. Kini banyak gerbang modern dioperasikan secara otomatis sehingga dapat membuka dan menutup secara otomatis.

Monitoring (bahasa Indonesia: pemantauan) adalah pemantauan yang dapat dijelaskan sebagai kesadaran (*awareness*) tentang apa yang ingin diketahui, pemantauan berkadar tingkat tinggi dilakukan agar dapat membuat pengukuran melalui waktu yang menunjukkan pergerakan ke arah tujuan atau menjauh dari itu. *Monitoring* akan memberikan informasi tentang status dan kecenderungan bahwa pengukuran dan evaluasi yang diselesaikan berulang dari waktu ke waktu, pemantauan umumnya dilakukan untuk tujuan tertentu, untuk memeriksa terhadap proses berikut objek atau untuk mengevaluasi kondisi atau kemajuan menuju tujuan hasil

manajemen atas efek tindakan dari beberapa jenis antara lain tindakan untuk mempertahankan manajemen yang sedang berjalan.

Sebelumnya, sudah dibuat alat untuk monitoring pintu untuk keamanan sebuah bangunan. Alat tersebut menggunakan *Arduino*, dan LCD sebagai penunjuk status pintu terkunci, pintu tidak terkunci, pintu terbuka atau tertutup yang menggunakan sensor LDR (*Light Dependent Resistor*)^[1]. Sedangkan yang satunya juga system monitoring di gerbang jalan tol untuk mengetahui kepadatan yang terjadi di jalan tol dengan menggunakan Mikrokontroler AT89S51 dan sensor LDR^[2]. Dengan demikian, dari beberapa literatur yang dibutuhkan untuk membangun sistem monitoring yang telah dijelaskan diatas, maka penulis tertarik untuk melakukan perancangan alat yaitu *Rancang Bangun Monitoring Gate Pengunjung Pada Portal Masuk Wahana Permainan Dengan Web Application Berbasis Arduino Mega*.

2.2 Landasan Teori

2.2.1 Web Application

Web application merupakan sebuah aplikasi yang menggunakan teknologi *browser* untuk menjalankan aplikasi dan diakses melalui komputer (Remick, 2011). Sedangkan menurut (Rouse, 2011) aplikasi *web* adalah sebuah program sebuah program yang disimpan di *server* dan dikirim di internet lalu diakses melalui antarmuka *web browser*. Dari pengertian tersebut dapat disimpulkan aplikasi *web* merupakan aplikasi

yang diakses menggunakan *web* browser melalui jaringan internet atau intranet. Aplikasi *web* juga merupakan suatu perangkat lunak komputer yang dikodekan dalam bahasa pemrograman yang mendukung perangkat lunak berbasis *web* seperti *HTML*, *JavaScript*, *CSS*, *Ruby*, *Python*, *PHP*, *Java* dan bahasa pemrograman lain.^[3]

Ada dua bagian pokok dalam aplikasi *web*, yang pertama yaitu sisi *client* dan yang kedua adalah sisi *server*. Sisi *client* dalam hal ini komputer maupun *mobile* seperti *smartphone*, tablet yang terhubung ke jaringan internet, *client* dapat mengakses aplikasi *web* melalui *web browser* seperti *Mozilla Firefox*, *Google Chrome*, *Microsoft Edge*, *Opera*, *Safari*, dll. Sedangkan *server* adalah perangkat komputer dengan spesifikasi yang bagus digunakan untuk menyimpan aplikasi *web* beserta *database server* yang siap untuk diakses oleh *client*. *Client* bertugas meminta halaman *web server* melalui *web browser*, *web browser* akan meneruskannya ke *server* dimana aplikasi *web server* berada. Komputer *server* akan mengolah permintaan dari *client*, ketika halaman web yang diminta ditemukan maka komputer *server* akan mengirimkannya ke komputer *client* dan halaman *web* yang diminta akan ditampilkan pada *web browser* di komputer *client*. Berikut adalah komponen aplikasi *web*.

2.2.1.1 PHP (*Personal Home Page*)

PHP (*Personal Home Page*) pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama

FormInterpreted (FI), yang wujudnya berupa sekumpulan *script* yang digunakan untuk mengolah data formulir dari *web*.^[4]

PHP dapat digunakan secara gratis dan bersifat *open source*. PHP dirilis dalam lisensi *PHP License*, sedikit berbeda dengan lisensi *GNU General Public License (GPL)* yang biasa digunakan oleh proyek *open source*. Hingga saat ini, PHP digunakan pada banyak aplikasi berbasis *web*. Beberapa aplikasi berbasis *web* menggunakan PHP yang cukup familiar yaitu *Facebook, Twitter, Blogspot, Ebay, Wikipedia, Google, Google Map, YouTube, Instagram* dan lain-lain.

PHP disebut bahasa pemrograman *server-side*, karena PHP di proses pada *computer server*. Hal ini berbeda dibandingkan dengan bahasa pemrograman *client-side* seperti *JavaScript* yang diproses pada *web browser (client)*. Intinya, PHP adalah sebuah bahasa pemrograman yang bisa membantu dalam membuat aplikasi apa saja dan bisa diakses oleh siapa saja dengan menggunakan teknologi *server-side*. PHP hanya akan berjalan pada aplikasi berbasis *server*, baik *server* yang berjalan di komputer lokal (*Localhost*) maupun server yang berjalan secara online. Oleh karena PHP berjalan pada server, maka PHP tidak bisa berjalan ketika diakses secara langsung.

2.2.1.2 Web Server dan Database MySQL

Fungsi *Web Server* adalah untuk menempatkan dimana *script* yang kita buat itu berada. Kita bisa menyewa penyedia jasa *Online*

Server(Hosting) atau kiat bisa *install software*-nya di komputer kita sebelum di online-kan agar bisa diakses oleh pengunjung. ^[5]

Jika sudah punya server-nya, *tool* berikutnya yang harus terpasang adalah *Database MySQL*. *Database MySQL* ini berfungsi untuk menyimpan data-data hasil pengolahan *script*. Sebenarnya tanpa database pun PHP bisa berjalan, tetapi hasilnya akan sama seperti kita menuliskan kode HTML. Hasil dari pengolahan *script* PHP menjadi statis. Dengan adanya *Database* berarti pengolahan data yang dihasilkan menjadi statis dan dinamis.

Database adalah suatu aplikasi yang menyimpan sekumpulan data. Setiap *database* mempunyai API tertentu untuk membuat, mengakses, mengatur, mencari, dan menyalin data yang ada di dalamnya. Untuk menampung dan mengatur data yang begitu banyak, dapat menggunakan *Relational Database Management Systems (RDBMS)*. Hal ini disebut *relational database* karena semua data disimpan dalam table-table yang berbeda dan dihubungkan berdasarkan relasinya dengan menggunakan *primary key* dan *foreign key*.

Relational Database Management System (RDBMS) adalah *software* yang:

- a. Memungkinkan pengguna untuk mengimplementasikan sebuah database dengan tabel-tabel, kolom-kolom, dan indeks-indeks.
- b. Menjamin integritas referensi diantara baris-baris pada berbagai tabel.
- c. Mengupdate indeks-indeks secara otomatis.

- d. Menginterpretasikan *query SQL* dan menggabungkan informasi dari berbagai tabel.

MySQL merupakan *database server open source* yang cukup populer keberadaannya. Dengan berbagai keunggulan yang dimiliki, membuat *software database* ini banyak digunakan oleh para praktisi untuk membangun suatu *project*. Adanya fasilitas API (*Application Programming Interface*) yang dimiliki oleh *MySQL*, memungkinkan bermacam-macam aplikasi komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data *MySQL*. Berikut logo *MySQL* pada gambar



2.1.

Gambar 2.1 Logo *MySQL*^[14]

MySQL database server adalah RDBMS (*Relasional Database Management System*) yang dapat menangani data yang bervolume besar. Meskipun begitu, tidak menuntut *resource* yang besar. *MySQL* adalah program *database* yang mampu mengirim dan menerima data dengan sangat cepat dan *multi user*. *MySQL* memiliki 2 bentuk lisensi, yaitu *free software* dan *shareware*. *MySQL free software* sering digunakan karena

bebas menggunakan *database* ini untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensi, yang dapat diunduh pada alamat resminya <http://www.mysql.com>.

a. Kelebihan memakai *MySQL*:

MySQL memiliki beberapa kelebihan dan keuntungan dibanding *database* lain seperti *Oracle*, *PostgreSQL*, *mSQL*, dan *Microsoft SQL Server*. Keuntungan dan kelebihan tersebut antara lain:

- a. Banyak ahli berpendapat *MySQL* merupakan server tercepat
- b. *MySQL* merupakan *system management database* yang *open source* (kode sumbernya terbuka), yaitu *software* ini bersifat bebas digunakan oleh perseorangan atau instansi tanpa harus membeli atau membayar kepada pembuatnya.
- c. *MySQL* memiliki performa yang tinggi tapi *simple*.
- d. *Database MySQL* mengerti bahasa SQL (*Structure Query Language*).
- e. *MySQL* dapat di akses melalui *protocol ODBC (Open Database Connectivity)* buatan *Microsoft*. Hal ini yang menyebabkan *MySQL* dapat diakses oleh banyak *software*.
- f. Semua klien dapat mengakses *server* dalam satu waktu, tanpa harus menunggu yang lain untuk mengakses *database*.
- g. *Database MySQL* dapat diakses dari semua tempat di internet dengan hak akses tertentu.
- h. *MySQL* merupakan *database* yang mampu menyimpan data berkapasitas besar, sampai berukuran *gigabyte*.

- i. *MySQL* dapat berjalan di berbagai operating system seperti *Linux*, *Windows*, *Solaris*, dan lain-lain.

b. Cara Kerja *MySQL*

Keunggulan *MySQL* adalah kemampuannya dalam menyediakan berbagai fasilitas atau fitur-fitur yang dapat digunakan oleh bermacam-macam *user*. *User* ini sendiri termasuk administrator *database*, *programmer* aplikasi, *manager*, sampai *end user*.^[5]

Pada mulanya *MySQL* bekerja pada *platform unix* dan *linux*. Namun dengan perkembangannya, sekarang banyak bermunculan beberapa distro yang mampu berjalan pada beberapa *platform* yang bersifat *share-ware* dan *corporate*.

MySQL adalah sebuah manajemen system *database server* yang mampu menangani beberapa *user*, yaitu mampu menangani beberapa instruksi sekaligus dari beberapa *user* dalam satu waktu. Kemudian *MySQL* merekam semua data *user* di dalam sistem dalam *table user*.

Untuk pengguna yang menjalankan *MySQL* pada *platform Windows* dapat mengakses melalui program *client* seperti *putty*, *talnet*, *psftp*, dan lain-lain. Sedangkan untuk pengguna yang menjalankan *MySQL* pada *platform linux* dapat mengakses melalui program *client* seperti *SSH* maupun *tanlent*.

2.2.1.2 CSS (*Cascading Style Sheet*)

CSS merupakan salah satu kode pemrograman yang bertujuan untuk menghias dan mengatur gaya tampilan/*layout* halaman *web* supaya lebih elegan dan menarik. CSS adalah sebuah teknologi internet yang direkomendasikan oleh *World Wide Web Consortium* atau W3C pada tahun 1996. Awalnya, CSS dikembangkan di SGML pada tahun 1970, dan terus dikembangkan hingga saat ini. CSS telah mendukung banyak bahasa markup seperti HTML, XHTML, XML, SVG (*Scalable Vector Graphics*) dan *Mozilla XUL (XML User Interface Language)*.^[6]

Pada Desember 1996, W3C memperkenalkan Level 1 spesifikasi CSS atau juga dikenal CSS1 yang mendukung format, warna *font* teks, dan lain-lain. Kemudian, Mei 1998, W3C menerbitkan CSS2 yang di dalamnya diatur fungsi peletakan elemen. Dan sekarang, W3C telah memperbaiki dan meningkatkan kemampuan CSS2 ke CSS3.

CSS digunakan oleh *web programmer* dan juga *blogger* untuk menentukan warna, tata letak font, dan semua aspek lain dari presentasi dokumen di situs mereka. Saat ini, hampir tidak ada situs web yang dibangun tanpa kode CSS.

2.2.1.3 HTML (*Hyper Text Markup Language*)

HTML adalah kepanjangan dari *HyperText Markup Language*, merupakan bahasa interpretasi yang digunakan pada sebuah halaman web. HTML mendeskripsikan struktur halaman *web* yang ditulis dengan *element* atau *tag* yang mengapit konten atau teks didalamnya.^[7]

Penjelasan lebih rinci mengenai arti kata-perkata dari HTML adalah sebagai berikut, *HyperText* adalah istilah teks aktif, yang apabila diklik akan meloncat atau menuju halaman lain. Ini merupakan kemampuan dari sebuah halaman *web* yang dapat saling berhubungan antara halaman satu dengan lainnya. *Markup*, merupakan tag-tag yang biasanya diawali dengan *tag* pembuka (*opening tag*) dan tag penutup (*closing tag*) yang memberi kemampuan untuk menata *layout* atau memformat struktur halaman *web* pada sebuah konten teks sederhana didalam *file* HTML itu sendiri. *Language*, yaitu bahasa yang digunakan oleh HTML itu sendiri. Perintah-perintah *tag* menggunakan bahasa yang dapat dimengerti oleh *browser* atau *interpreter* lainnya.

HTML bukanlah sebuah bahasa pemrograman pada umumnya, seperti *Java*, *C*, *C++*, *Visual basic* dan sejenisnya, melainkan bahasa markup yang ditulis dengan perintah *tag-tag* atau element yang menaungi (mengapit) konten didalamnya yang akan ditampilkan pada sebuah halaman web oleh browser atau HTML *interpreter* (penerjemah HTML) lainnya.

Hypertext mengacu pada cara di mana halaman *web* (dokumen HTML) dihubungkan. Jadi, link yang tersedia pada halaman web disebut *Hypertext*. Seperti namanya, HTML adalah bahasa *Markup* yang berarti menggunakan HTML hanya untuk “*mark-up*” dokumen teks dengan tag yang akan memberitahukan *browser* struktur untuk menampilkan sebuah desain *layout web*. Awalnya, HTML dikembangkan

dengan maksud untuk mendefinisikan struktur dokumen seperti judul, paragraf, daftar, dan sebagainya untuk memudahkan berbagi informasi ilmiah antara peneliti.

2.2.1.4 Sublime Text

SublimeText adalah suatu *text editor* yang berjalan pada *Operating System(OS) Windows*. *Sublime text* merupakan sebuah teks editor berbasis *Python* yang mempunyai banyak fitur – fitur. *Sublime text* juga mendukung banyak bahasa pemrograman diantaranya C++, C#, CSS, PHP, HTML, *Javascript*, ASP, dan masih banyak lagi. *Sublime text* mempunyai beberapa keunggulan – keunggulan yang dapat membantu pengguna dalam membuat sebuah *web development*.^[8]

a. Keunggulan Sublime

Berikut adalah kelebihan dari *Sublime Text* :

1) Settings

Di *Notepad++*, pengaturan bisa dilakukan dengan mudah melalui *Settings -> Preferences*, kemudian akan muncul jendela baru yang berisi *interface settings*.

Sedangkan di *Sublime Text*, pengaturan dilakukan dengan membuka *file, sublime, settings* dan mengedit isinya. Perubahan langsung terjadi pada saat *file* tersebut disimpan. Untuk yang bukan

programmer, mungkin akan kesulitan karena harus mengedit data berformat JSON.

2) *Color Scheme/Theme*

Di *Notepad++* kita bisa mengganti *theme* melalui *Settings* -> *Style Configurator*. Di sini ada beberapa pilihan *theme*, dan tiap *theme* bisa diedit dengan mudah. Kita juga bisa menambah *theme* baru dengan cara mengkopi *file xml* ke dalam folder *Notepad++/themes*.

Di *Sublime Text*, kita bisa mengganti *Color Scheme* (untuk area editor) maupun *Theme* (untuk *interface sidebar, tab, console, search, dll*). Untuk mengedit *color scheme*, kita harus membuka *file xml* berekstensi (.tmTheme) dan mengedit isinya.

Sublime Text secara *default* menggunakan warna *background* gelap, sedangkan *Notepad++* menggunakan *background* terang. Tapi tentu saja kita bisa mengubahnya dengan mudah.

3) *Goto Anything*

Dengan *Goto Anything*, kita bisa membuka file di dalam *project* dengan cepat, tinggal tekan Ctrl + P kemudian ketik nama filenya. Untuk mencari nama *file* tidak harus mengetik secara tepat, karena *Sublime Text* menggunakan *algoritma fuzzy* untuk pencariannya. *Notepad++* belum memiliki fitur semacam ini.

4) *Command Palette*

Di *Notepad++*, untuk menjalankan perintah tertentu bisa melalui menu *bar*, *toolbar*, menu konteks atau *shortcut*. Kita bisa mengatur *Shortcut* melalui *Settings* -> *Shortcut Mapper*.

Sublime Text memiliki tampilan yang lebih *simple* dan sangat minim menu, bahkan tidak ada *toolbar* sama sekali. Kebanyakan perintah-perintahnya bisa kita akses menggunakan *Shortcut*, atau kalau belum hafal *shortcut* kita bisa mengakses *Command Palette* (tekan *Ctrl + Shift + P*) kemudian cari perintah yang kita inginkan (menggunakan *fuzzy search*). Sebagai contoh, saya ingin membuat teks menjadi *uppercase*, maka teks tersebut saya blok dulu, kemudian masuk *Command Palette* dan ketik “*case*” atau “*upc*” maka akan muncul perintah untuk *uppercase* beserta keterangan *shortcut*-nya.

5) *Package/Plugin*

Notepad++ memiliki sangat banyak *plugin* yang bisa dipasang melalui *Plugin Manager*. Kita juga bisa membuat *plugin* sendiri menggunakan bahasa *C++*.

Sublime Text juga memiliki sangat banyak *package*. Untuk memasang *package* kita bisa menjalankan *Package Manager* kemudian akan muncul *list package*, kita tinggal mencari (tentu saja menggunakan *fuzzy search* juga) *package* yang diinginkan.

Kelebihan menggunakan *package manager* adalah *package* akan otomatis terbaru jika ada versi baru. *Sublime Text* juga menyediakan

plugin API untuk kita yang ingin membuat *package* sendiri menggunakan bahasa *Python*.

6) *Multiple Selections/Multi-editing*

Ini adalah salah satu fitur unggulan di *Sublime Text*, di mana kita bisa meletakkan *cursor* di beberapa tempat (menggunakan *Ctrl + click*), kemudian mengedit secara bersamaan.

Di *Notepad++* juga ada fitur ini, tapi kita harus mengaktifkan dulu melalui *Settings -> Preferences -> Editing -> Multi-Editing Settings*. Fitur *Multiple Selections* di *Sublime Text* lebih baik karena bisa untuk mempercepat editing teks yang sama. Misal ingin mengedit sebuah kata, letakkan kursor pada kata tersebut kemudian tekan *Ctrl + D* untuk menyeleksi satu persatu semua kata yang sama, atau tekan *Alt + F3* untuk langsung menyeleksi semua kata tersebut.

7) *Column Editing*

Baik *Notepad++* maupun *Sublime Text* memiliki fitur ini. Di *Notepad++* tekan *Alt* kemudian mouse *drag* ke bagian yang ingin diedit. Di *Sublime Text* menggunakan *drag* tombol *mouse* tengah, atau bisa juga dengan *Shift + drag* tombol *mouse* kanan.

8) *Split Editing*

Jika and memiliki monitor yang lebar maka kedua editor ini sangat cocok karena sama-sama memiliki fitur untuk *Split Editing* (membuka 2 atau lebih file secara berdampingan).

9) *Auto Completion*

Sublime Text memiliki *auto complete* untuk beberapa bahasa yang saya pakai seperti *PHP*, *CSS*, *Javascript*. Fitur ini juga mendukung *fuzzy search* sehingga tidak harus mengetik secara tepat. *Notepad++* juga memiliki *auto complete*, namun tidak senyaman *Sublime Text* dan harus diaktifkan dulu melalui *Settings -> Preferences -> Backup/Auto-Completion*.

10) *Minimap/Document Map*

Sublime Text memiliki *Minimap*, semacam versi mini dari file untuk mempermudah melihat file secara keseluruhan. Di *Notepad++* juga ada fitur seperti ini dengan nama *Document Map*. Untuk mengaktifkannya dari *View -> Document Map*.

11) *Goto Definition*

Ini adalah fitur baru *Sublime Text 3* (di *ST2* blm ada). Fitur ini sangat membantu menemukan *function/class* di dalam *project*, caranya dengan meletakkan cursor di nama *function/class* kemudian tekan *F12*, maka file yang berisi definisi *function/class* tersebut akan terbuka. Untuk melihat list semua *function/class* bisa menggunakan *Goto Symbol in Project* (tekan *Ctrl + Shift + R*).

Notepad++ belum memiliki fitur ini. Fitur semacam ini sangat jarang di *text editor* biasa, kebanyakan ada di IDE kelas berat seperti *NetBeans*, *Zend Studio*, dll.

12) *Instant Project Switch*

Untuk mempermudah bekerja di *Sublime Text*, sebaiknya kita membuat *project* terlebih dahulu. Caranya, buka jendela *Sublime Text* baru, kemudian folder tempat *file-file* yang akan kita edit kita drag ke dalam jendela *Sublime Text*, kemudian simpan *project* tersebut melalui *Project -> Save Project As.Sublime Text* akan mengindex semua *file* di dalam folder tersebut supaya fitur *Goto* bisa bekerja. Proses indeks pun berjalan cepat, sebagai contoh, *script-script WordPress* bisa diindex semua dalam waktu sekitar 20 detik saja. Dengan membuat *project*, maka kita bisa berpindah-pindah *project* dengan cepat, dengan menekan *Ctrl + Alt + P*, lalu pilih *project* dari *list project*.

Saat kita berpindah *project* atau saat *Sublime Text* ditutup, kita tidak akan diganggu dengan dialog konfirmasi untuk menyimpan *file*. Semua *file* yang belum disimpan akan kembali lagi (tidak hilang) saat *project* dibuka



lagi. Berikut logo *Sublime Text* pada gambar 2.2.

Gambar 2.2 Logo *Sublime*^[8]

Pembuatan *web application* menggunakan *Sublime Text* untuk memprogram *web*. Membuat “*style.css*” dan “*table.css*” sebagai desain pada *web*. Lampirkan juga *library Javascript*. Kemudian masukkan pada “*index.php*”. Membuat “*index.php*” sebagai halaman utama. Buat file “*koneksi.php*” untuk penyambung dengan *database web server*. Buat “*ambildata.php*” untuk memasukkan data dari *Arduino* ke *database*. Lalu buat “*table.php*” untuk menampilkan data terakhir, lalu buat “*jumlah.php*” untuk menjumlah pengunjung harian dalam tabel. Jadikan *File* dalam 1 satu folder dengan nama “wahana”.

Bila semuanya diatas sudah dilakukan, *upload file* ke *hosting* dengan menggunakan *FileZilla*. Cara menunggah file ke *hosting* menggunakan *FileZilla*. Buka aplikasi *FileZilla*, klik bagian *Site Manager*. Setelah dapat, masukkan *Host*, *User*, dan *Password* yang didapatkan dari layanan *hosting*. Upload folder wahana ke kolom sebelah kanan. Setelah selesai diunggah, maka proses sudah selesai. Untuk memastikan *file* sudah terunggah, maka buka *browser* dengan alamat <http://d3elektro.com/wahana>.

2.2.1.5 JavaScript

Javascript didesain untuk dapat dijalankan pada lingkungan dinamis. Perubahan pada suatu *class* dengan penambahan *property* ataupun *method* dapat dilakukan tanpa mengganggu program pengguna *class* tersebut.^[9]

JavaScript adalah bahasa pemrograman *web* yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah

tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti *Google Chrome* dan *Mozilla Firefox*. Bahasa pemrograman *Client Side* berbeda dengan bahasa pemrograman *Server Side* seperti PHP, dimana untuk *server side* seluruh kode program dijalankan di sisi server. Untuk menjalankan *JavaScript*, kita hanya membutuhkan aplikasi *text editor* dan *web browser*. *JavaScript* memiliki fitur *high-level programming language*, *client-side*, *loosely typed* dan berorientasi objek.

JavaScript pada awal perkembangannya berfungsi untuk membuat interaksi antara *user* dengan situs *web* menjadi lebih cepat tanpa harus menunggu pemrosesan di *web server*. Sebelum *Javascript*, setiap interaksi dari *user* harus diproses oleh *web server*. Bayangkan ketika kita mengisi *form registrasi* untuk pendaftaran sebuah situs web, lalu klik tombol *submit*, menunggu sekitar 20 detik untuk website memproses isian *form* tersebut, dan mendapati halaman yang menyatakan bahwa terdapat kolom *form* yang masih belum diisi.

Untuk keperluan seperti inilah *JavaScript* dikembangkan. Pemrosesan untuk mengecek apakah seluruh form telah terisi atau tidak, bisa dipindahkan dari *web server* ke dalam *web browser*. Dalam perkembangan selanjutnya, *JavaScript* tidak hanya berguna untuk *validasi form*, namun untuk berbagai keperluan yang lebih modern. Berbagai animasi untuk mempercantik halaman *web*, fitur *chatting*, efek-efek modern, *games*, semuanya bisa dibuat menggunakan *JavaScript*.

Akan tetapi karena sifatnya yang dijalankan di sisi *client* yakni di dalam *web browser* yang digunakan oleh pengunjung situs, *user* sepenuhnya dapat mengontrol eksekusi *JavaScript*. Hampir semua *web browser* menyediakan fasilitas untuk mematikan *JavaScript*, atau bahkan mengubah kode *JavaScript* yang ada. Sehingga kita tidak bisa bergantung sepenuhnya kepada *JavaScript*.

2.3 *Wireless Fidelity (Wi-Fi)*

Wi-Fi atau *Wireless Fidelity* adalah sebuah teknologi yang digunakan pada peralatan elektronik untuk bertukar data secara nirkabel (menggunakan gelombang radio) melalui sebuah jaringan komputer. Peralatan dapat terhubung ke jaringan komputermelalui sebuah *access point*. Sebuah *access point* memiliki jangkauan kurang lebih 20 meter di dalam ruangan, dan lebih luas jika di luar ruangan.^[10]

Teknologi *Wi-Fi* memiliki standar, yang ditetapkan oleh sebuah institusi internasional yang bernama *Institute of Electrical and Electronic Engineers* (IEEE), yang secara umum sebagai berikut:

- a. Standar IEEE 802.11a yaitu *Wi-Fi* dengan frekuensi 5 GHz yang memiliki kecepatan 54 Mbps dan jangkauan jaringan 300m.
- b. StandarIEEE 802.11b yaitu *Wi-Fi* dengan frekuensi 2,4 GHz yang memmiliki kecepatan 11 Mbps dan jangkauan jaringan 100m.

c. Standar IEEE 802.11g yaitu *Wi-Fi* dengan frekuensi 2,4 GHz yang memiliki kecepatan 54 Mbps dan jangkauan jaringan 300m.

Teknologi *Wi-Fi* yang akan diimplementasikan adalah standar IEEE 802.11g karena standar tersebut lebih cepat untuk proses transfer data dengan jangkauan jaringan yang lebih jauh serta dukungan vendor (perusahaan pembuat hardware). Perangkat tersebut bekerja di frekuensi 2,4 GHz atau disebut sebagai pita frekuensi ISM (*Industrial, Scientific, and Medical*) yang juga digunakan oleh peralatan lain, seperti *microwave oven* dan *bluetooth*.

2.3.1 Keunggulan dan Kelemahan Jaringan *Wi-Fi*

Keunggulan jaringan *Wi-Fi*.^[10]

- a. Biaya pemeliharaan murah.
- b. Infrastruktur berdimensi kecil.
- c. Pembangunannya cepat.
- d. Mudah dan murah untuk direlokasi.
- e. Mendukung portabilitas.

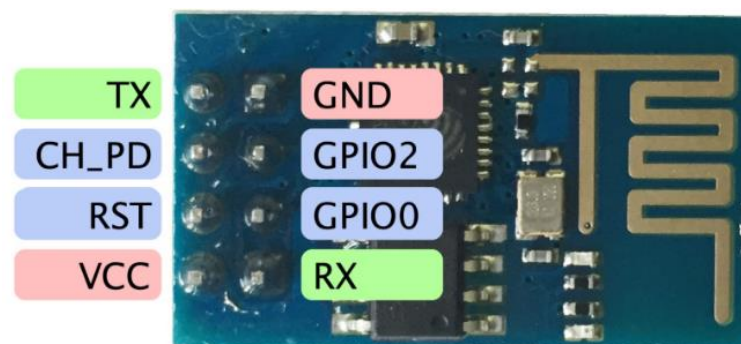
Kelemahan jaringan *Wi-Fi*:

- a. Biaya peralatan mahal.
- b. *Delay* yang sangat besar.
- c. Kesulitan karena masalah propagasi radio.
- d. Mudah untuk interferensi.

- e. Kapasitas jaringan kecil karena keterbatasan spectrum (pita frekuensi yang tidak dapat diperlebar).
- f. Keamanan/kerahasiaan data kurang terjamin.

2.4 ESP8266

Modul ESP8266 merupakan SoC (*System on Chip*) dengan *stack protocol* TCP/IP yang telah terintegrasi, sehingga mudah di akses menggunakan mikrokontroler melalui komunikasi serial 802.11 b/g/n *Wi-Fi Direct (P2P)*. Modul *Wi-Fi* ESP8266 dapat berfungsi sebagai *host* maupun sebagai modul transfer data dalam jaringan *Wi-Fi*. Modul ini memiliki kemampuan pengolahan dan penyimpanan data yang baik sehingga memungkinkan untuk diintegrasikan dengan sensor dan perangkat khusus lainnya melaluiGPIO.Dibawah adalah modul ESP8266 ditunjukkan pada gambar 2.3.^[11]



Gambar 2.3Modul ESP8266^[11]

Berikut ini spesifikasi dari modul ESP8266:

- a. Mendukung protocol 802.11 b/ g/ n.

- b. *Wi-Fi Direct (P2P/ Point to Point), Soft AP (Access Point).*
- c. Protokol TCP/ IP.
- d. Mendukung WEP, TKIP, AES dan WAPI.
- e. *Power Amplifier* 24 dBm.
- f. Rangkaian PLL, pengatur tegangan dan pengelola daya terpadu.
- g. Daya keluaran mencapai +19,5 dBm pada mode 802.11b.
- h. Sensor suhu internal terpadu.
- i. Mendukung berbagai macam antenna.

CPU mikro 32 bit terpadu yang dapat digunakan sebagai pemroses aplikasi melalui antarmuka iBus, dBus, AHB (untuk akses *register*) dan JTAG (untuk *debugging*).

- j. Antarmuka SDIO 2.0, SPI, UART.

Modul ESP8266 membutuhkan *input* tegangan dengan *range* 3.3 volt, namun konsumsi dayanya tinggi. Jika tegangan yang masuk kurang atau lebih dari *range* yang ditentukan maka modul tidak akan aktif atau kondisi yang lebih buruk lagi yaitu menjadi rusak. Arus listrik yang dibutuhkan cukup tinggi, sehingga kita perlu menggunakan arus 1 A.

Pengaturan awal modul ESP8266 dapat menggunakan *AT Command* yang dikirim dari *Arduino* menggunakan komunikasi serial. Penggunaan *AT Command* dapat memberikan kemudahan untuk mengetahui kekuatan sinyal dari terminal, mengirim pesan, menambahkan *item*, mematikan terminal, mendapatkan *IP address*, dan lain-lain. *AT Command* dalam ESP8266 dapat dilihat pada tabel 2.1.

Tabel 2.1 AT Command pada ESP8266 ^[12]

Function	AT Command	Response
Working	AT	OK
Restart	AT+RST	OK [System Ready, Vendor:www.ai-thinker.com]
Firmware version	AT+GMR	AT+GMR 0018000902 OK
List Access Points	AT+CWLAP	AT+CWLAP +CWLAP:(4,"RocheFortSurLac",-38,"70:62:b8:6f:6d:58",1) +CWLAP:(4,"LiliPad2.4",-83,"f8:7b:8c:1e:7c:6d",1) OK
Join Access Point	AT+CWJAP? AT+CWJAP="SSID","Password" "	Query AT+CWJAP? +CWJAP:"RocheFortSurLac" OK
Quit Access Point	AT+CWQAP=? AT+CWQAP	Query OK
Get IP Address	AT+CIFSR	AT+CIFSR 192.168.0.105 OK
Set Parameters of Access Point	AT+ CWSAP? AT+ CWSAP=<ssid>,<pwd>,<chl>,<ecn>	Query ssid, pwd chl = channel, ecn = encryption
WiFi Mode	AT+CWMODE? AT+CWMODE=1 AT+CWMODE=2 AT+CWMODE=3	Query STA AP BOTH
Set up TCP or UDP connection	AT+CIPSTART=? (CIPMUX=0) AT+CIPSTART = <type>,<addr>,<port> (CIPMUX=1) AT+CIPSTART= <id><type>,<addr>,<port>	Query id = 0-4, type = TCP/UDP, addr = IP address, port= port
TCP/UDP Connections	AT+ CIPMUX? AT+ CIPMUX=0 AT+ CIPMUX=1	Query Single Multiple
Check join devices' IP	AT+CWLIF	
TCP/IP Connection Status	AT+CIPSTATUS	AT+CIPSTATUS? no this fun

Send TCP/IP data	(CIPMUX=0) AT+CIPSEND=<length>; (CIPMUX=1) AT+CIPSEND= <id>,<length>	
Close TCP / UDP connection	AT+CIPCLOSE=<id> or AT+CIPCLOSE	
Set as server	AT+ CIPSERVER= <mode>[,<port>]	mode 0 to close server mode; mode 1 to open; port = port
Set the server timeout	AT+CIPSTO? AT+CIPSTO=<time>	Query 0~28800 in seconds
Baud Rate*	AT+CIOBAUD? Supported: 9600, 19200, 38400, 74880, 115200, 230400, 460800, 921600	Query AT+CIOBAUD? +CIOBAUD:9600 OK
Check IP address	AT+CIFSR	AT+CIFSR 192.168.0.106 OK
Firmware Upgrade (from Cloud)	AT+CIUPDATE	1. +CIPUPDATE:1 found server 2. +CIPUPDATE:2 connect server 3. +CIPUPDATE:3 got edition 4. +CIPUPDATE:4 start update
Received data	+IPD	(CIPMUX=0): + IPD, : (CIPMUX=1): + IPD, , :
Watchdog Enable*	AT+CSYSWDTENABLE	Watchdog, auto restart when program errors occur: enable
Watchdog Disable*	AT+CSYSWDTDISABLE	Watchdog, auto restart when program errors occur: disable

Sumber: <http://www.pridopia.co.uk/pi-doc/ESP8266ATCommandsSet.pdf>

2.5 Web Browser

Web Browser adalah suatu program atau *software* yang digunakan untuk menjelajahi internet atau untuk mencari informasi dari suatu *web* yang

tersimpan didalam komputer. Awalnya, *web browser* berorientasi pada teks dan belum dapat menampilkan gambar. Namun, *web browser* sekarang tidak hanya menampilkan gambar dan teks saja, tetapi juga memutar *file* multimedia seperti video dan suara. *Web browser* juga dapat mengirim dan menerima email, mengelola HTML, sebagai input dan menjadikan halaman *web* sebagai hasil *output* yang informatif. Contoh *Web Browser* :^[12]

- a) *Mozilla Firefox* , yaitu dibuat oleh *Mozilla Corporation*, *Firefox* adalah salah satu *web browser open source* yang dibangun dengan *Gecko layout engine*. Tak hanya andal *Firefox* juga didukung oleh sejumlah *add-ons* yang dapat dipasang terpisah yang memungkinkan pengguna melakukan sesuai dengan kegunaan *add-ons* tersebut.
- b) *Internet Explorer*, yaitu *web browser* besutan *Microsoft Corporation* biasanya dikenal dengan nama pendek IE, sejak 1995 IE mulai di masukan sebagai default software pada saat instalasi Sistem Operasi *Windows*, sejak tulisan ini dibuat IE belum lama ini meluncurkan versi IE8. Penerusnya adalah *Microsoft Edge* yang lebih bagus, ringan , dan cepat.
- c) *Safari*, yaitu Dibuat oleh *Apple Inc*, perusahaan yang juga memproduksi komputer *Macintosh*, *iPod*, dan juga *iPhone*, *iPad*. dibangun dengan browser engine *WebKit*, *WebKit* juga adalah *browser engine* pertama yang lulus *test Acid3*.

- d) *Opera*, yaitu *Opera* dikembangkan oleh *Opera Software Company* adalah salah satu *Web Browser* dan juga *Internet Suite*. Jika *Firefox* punya *add-ons*, *Opera* punya “*Opera Widgets*”, sebuah aplikasi *web* kecil yang dijalankan bersamaan dengan *Opera* yang mempunyai kegunaan tertentu, layaknya *add-ons Firefox*. Gambar 2.4 menunjukkan beberapa logo *web browser* terkemuka.



Gambar 2.4Contoh *Web Browser*^[13]