

BAB II

TEORI PENUNJANG

2.1 Python

Python dikenal sebagai bahasa pemrograman interpreter, karena Python dieksekusi dengan sebuah interpreter. Satu hal yang telah kita ketahui bahwa bahasa pemrograman Python adalah bahasa pemrograman yang mudah dibaca dan terstruktur, hal ini karena di gunakannya sistem indentasi. Yaitu memisahkan blok-blok program susunan indentasi. Jadi untuk memasukan sub-sub program dalam suatu blok, sub-sub program tersebut diletakkan satu atau lebih spasi dari kolom suatu blok program[1].

Diakui oleh pembuatnya sendiri, Guido Van Rossum bahwa Python memproses suatu program lebih lambat dibandingkan dengan C, bahasa tingkat tinggi lainnya seperti Perl, C++, Java, Pascal berada satu tingkat dengan Python. Akan tetapi pada kenyataannya python boleh dibilang bahasa pemrograman yang kecepatannya melebihi C[1].

Hal ini dikarenakan karena para pengembang software lebih cenderung memilih kecepatan dalam menyelesaikan suatu proyek dibandingkan dengan kecepatan proses dari program tersebut, mengenai kecepatan program di jawab dengan kecepatan prosesor dan memori yang sangat berkembang saat ini, mengakibatkan tidak terlihatnya kelambatan dari program Python, dalam kata lain kecepatan suatu program Python dapat sebanding dengan program yang dibuat dengan bahasa C[1].

2.2 MAC Address

MAC Address adalah sebuah alamat jaringan yang diimplementasikan pada lapisan data-link dalam tujuh lapisan model OSI, yang merepresentasikan sebuah node tertentu dalam jaringan. Dalam sebuah jaringan berbasis Ethernet, *MAC Address* merupakan alamat yang unik yang memiliki panjang 48-bit (6 byte) yang mengidentifikasi sebuah komputer, interface dalam sebuah router, atau node lainnya dalam jaringan. *MAC Address* juga sering disebut sebagai *Ethernet address*, *physical address*, atau *hardware address*[2].

MAC Address mengizinkan perangkat-perangkat dalam jaringan agar dapat berkomunikasi antara satu dengan yang lainnya. Sebagai contoh, dalam sebuah jaringan berbasis teknologi *Ethernet*, setiap header dalam *frame Ethernet* mengandung informasi mengenai *MAC Address* dari komputer sumber (*source*) dan *MAC Address* dari komputer tujuan (*destination*). Beberapa perangkat, seperti halnya *bridge* dan *switch Layer-2* akan melihat pada informasi *MAC Address* dari komputer sumber dari setiap *frame* yang ia terima dan menggunakan informasi *MAC Address* ini untuk membuat tabel routing internal secara dinamis. Perangkat-perangkat tersebut pun kemudian menggunakan tabel yang baru dibuat itu untuk meneruskan *frame* yang ia terima ke sebuah port atau *segmen* jaringan tertentu di mana komputer atau node yang memiliki *MAC Address* tujuan berada[2].

2.3 Web Server

Web server adalah sebuah software yang memberikan layanan berbasis data dengan menggunakan protokol HTTP atau HTTPS dari client menggunakan aplikasi web browser untuk request data dan server akan mengirim data dalam bentuk halaman web dan pada umumnya berbentuk dokumen HTML. Halaman web yang diminta bisa terdiri dari berkas teks, video, gambar, file dan banyak lagi[3].

Salah satu program dari Web Server adalah Apache. Apache merupakan web server yang paling banyak dipergunakan di Internet. Program ini pertama kali didesain untuk sistem operasi lingkungan UNIX, untuk saat ini telah tersedia apache yang di desai untuk sistem operasi lainnya. Apache mempunyai program pendukung yang cukup banyak. Hal ini memberikan layanan yang cukup lengkap bagi penggunaannya[3].

Aplikasi lain yang memiliki fungsi sebagai web server diantaranya, apache Tomcat, Microsoft windows Server 2003 Internet Information Services (IIS), Lighttpd, Sun Java System Web Server, Xitami Web Server, dan Zeus Web Server Arsitektur request dan response web server[3].

2.4 Raspberry Pi

Raspberry Pi pertama kali diluncurkan pada Februari 2012, tujuannya adalah untuk menciptakan dan membangkitkan minat anak-anak dalam bidang

komputasi. Pemilihan Raspberry Pi pada tugas akhir ini dikarenakan Raspberry Pi lebih kompleks untuk multimedia dan berbasis dasar linux yang banyak dikembangkan oleh siapapun yang bersifat open source. Raspberry Pi menggunakan *Sistem On Chip* (SoC) dari Broadcom BCM2835 dan tidak menggunakan *hard disk*, namun menggunakan *SD Card* untuk proses *booting* dan penyimpanan data jangka panjang. Pada saat awal muncul tersedia dua versi, yaitu model A dan Model B yang harganya US\$ 25 dan US\$ 35. Raspberry Pi ini berjalan diatas sistem operasi yang sifatnya open source seperti Raspbian Wheezy, Pidora, Risc OS, Rasp BMC, Arch Linux, dan lain-lain. Untuk bahasapemrogramannya menggunakan bahasa pemrogramana Phyton[4].

2.5 Radio Frequency Identification (RFID)

RFID ialah suatu metoda identifikasi objek yang menggunakan gelombang radio. Proses identifikasi dilakukan oleh RFID *reader* dan RFID *transponder* (RFID tag). RFID tag dilekatkan pada suatu benda atau suatu objek yang akan diidentifikasi. Tiap-tiap RFID tag memiliki data angka identifikasi (*ID number*) yang unik, sehingga tidak ada RFID tag yang memiliki ID number yang sama[5].

RFID digunakan untuk mendeskripsikan sebuah sistem yang mampu untuk mengirimkan data identitas sebuah objek secara nirkabel dengan menggunakan gelombang radio. RFID termasuk kedalam teknologi *Automatic Identification* (Auto- ID). Saat ini sistem identifikasi otomatis tersebut menjadi sangat populer dalam berbagai macam industri seperti jasa, pembelian, manufaktur dan lain sebagainya. Teknologi lain yang termasuk dalam Auto-ID adalah barcode, pembaca karakter optis dan teknologi *biometric*. Label barcode yang ada dimana-mana merupakan pencetus revolusi sistem identifikasi otomatis. Meskipun barcode sangat murah namun terdapat kelemahan dalam segi kapasistas penyimpanannya yang rendah dan tidak adanya kemampuan untuk diprogram ulang. Solusi optimal secara teknis adalah dengan memanfaatkan sebuah silicon chip sebagai media penyimpanan yang kemudian diadopsi dalam sistem RFID[5].

2.6 Firebase

Layanan pada Firebase meliputi autentikasi pengguna, pengaturan keamanan, dan hosting. Perubahan data pada satu *client* akan disinkronisasi pada semua *client* yang terdaftar ke data tersebut dalam hitungan mili detik. Kelebihan dari Firebase adalah dapat menerima data dari satu juta perangkat secara bersamaan. Penelitian ini akan menggabungkan Firebase dengan AngularJS untuk menyediakan web *realtime*. Firebase dapat melakukan sinkronisasi data dengan data local cache pada perangkat yang ada sehingga aplikasi tetap responsif bahkan saat koneksi jaringan lambat atau tidak tersedia. AngularJS dengan Firebase diklaim menjamin '*no server code*' sehingga memudahkan dalam pembuatan aplikasi (max, 2014). Firebase digunakan untuk menyimpan data yang diterima. Firebase diperlukan untuk menyediakan layanan penyimpanan cloud yang memungkinkan untuk *middleware* dapat digabungkan dengan *interface* aplikasi dari pihak ketiga. Firebase memungkinkan pengembang untuk menciptakan aplikasi *mobile* dan web yang dapat digunakan untuk men-generate data secara *realtime*[6].