BAB II
TINJAUAN PUSTAKA

2.1 Server

Server adalah salah satu infrastruktur yang paling penting dalam organisasi mana pun. Server merupakan induk dari segala komputer yang terhubung pada sebuah jaringan yang berfungsi sebagai pengatur sistem jaringan, misalnya untuk pembatasan akses dan melakukan *control* data. Server berfungsi untuk menyimpan aplikasi dan *database* yang di butuhkan oleh komputer yang terhubung, melindungi semua komputer yang terhubung menggunakan firewall, dan menyediakan IP *Address* untuk mesin komputer terhubung [1]. Komputer server dapat dikategorikan dalam dua kategori yaitu:

1. *Dedicated* Server

   Jenis server yang melakukan fungsi tertentu, seperti *web hosting*. ada berbagai layanan web hosting, yang menggunakan *dedicated serve*r untuk situs web hosting. Perusahaan tertentu juga telah mendedikasikan server komputer untuk menyimpan situs web mereka sendiri. Jenis server ini sangat kuat karena harus menangani lalu lintas web yang mencoba untuk mengakses halaman web yang terkandung di dalamnya.

2. *Non - dedicated* server (Server Bersama)

   Sebuah komputer server bersama adalah server biasa yang digunakan dalam jaringan komputer untuk beberapa pengguna. Sejumlah besar aplikasi, *database* disimpan di dalamnya. Pengguna yang berbeda terhubung ke server, mengakses server tergantung pada kebutuhan mereka. Server ini tidak perlu disesuaikan seperti *dedicated* server. Contoh yang paling umum untuk jenis server ini adalah server aplikasi. Sebuah server aplikasi menyimpan semua informasi yang diperlukan oleh orang-orang dalam jaringan.

2.2 *Database*

Database merupakan sebuah wadah yang dapat menerima, menyimpan dan memberikan suatu data. *Database* sekumpulan data yang terhubung dan tersimpan secara sistematis yang dapat diperiksa, diolah dan dimanipulasi. Dalam pengolahan data, *database* sangat dibutuhkan karena dapat mempermudah dalam melakuakan pengelompokan data, pengidentifikasin suatu data, mengubah data serta fungsi lainya. Suatu *database* pada umumnya terdiri dari banyak data dalam pengolahanya *database* terbagi menjadi beberapa jenis yaitu:

1. *Operation*al *Database*

   Database jenis ini dapat menyimpan data dengan rinci agar dapat dioperasikan dari seluruh organisasi. Dalam hal ini mereka juga dapat disebut *Subject Area Database* (SADB), transaksi *database* atau produksi *database*. Contoh dari *Operational Database* adalah *database* yang digunakan pelanggan, *database* akuntansi dan lain – lain.

2. *Relational Database*

   Jenis *database* yang satu ini bisa dibilang paling popular, hal ini dikarenakan dengan menggunakan *relational database*, *user* dapat mengakses atau mencari informasi dalam tabel yang berbeda-beda. *Query* yang dilakukan juga dapat melibatkan beberapa tabel karena fungsi relasi ini.

3. *Distributed Database*

   Sesuai dengan namanya *database* yang satu ini dapat mendistribusikan data-data secara tersebar namun saling berhubungan serta dapat diakses secara bersama-sama. *Database* jenis ini biasanya digunakan pada suatu lembaga atau perusahaan yang terdiri dari beberapa cabang, agar dapat memudahkan *user* mengakses data dari tempat satu ke tempat lainnya.

4. *External Database*

   Database ini mampu menyediakan akses ke bagian eksternal, data yang disimpan nantinya akan digunakan untuk keperluan komersial. Akses ke dalam *database* ini akan lebih mudah karena diperuntukkan bagi publik dan disamping itu juga lebih efisien karena tidak perlu mencari informasi dari internet.

2.3 *Thingspeak*

Gambar 2.1 Logo *Thingspeak* [2]

Thingspeak adalah *platform database* gratis yang menyediakan layanan untuk kebutuhan IoT. Dengan menggunakan *Thingspeak* pengguna dapat memanfaatkan layanan IoT secara gratis termasuk di dalamnya adalah *free broker, free logging, free API (Application Programming Interface)*. Dalam terminologi IoT dengan protokol MQTT, dikenal tiga macam *network element*, yakni *publisher* (pengirim data), *subscriber* (penerima data) dan *broker* (penghubung *subscriber* dan *publisher*) [3]. Dengan menggunakan *Thingspeak* pengguna dapat membangun infrastruktur IoT. Untuk dapat menggunakan layanan dari *Thingspeak*, maka pengguna harus terlebih dahulu membuat *channel* data pada *account Thingspeak*. Fungsi dari membuat channel ini.

2.4 MIT *App Inventor*

Gambar 2.2 MIT *App inventor* [4]

MIT *App Inventor* merupakan *platform* gratis untuk memudahkan proses pembuatan aplikasi sederhana tanpa harus mempelajari atau menggunakan bahasa pemrograman yang terlalu banyak. Kita dapat mendesain aplikasi android sesuai keinginan dengan menggunakan berbagai macam layout dan komponen yang tersedia. *App Inventor* menggunakan antarmuka grafis, serupa dengan antarmuka

pengguna pada *scratch* dan *starLogo* TNG, yang memungkinkan pengguna untuk men-*drag-and-drop* objek visual untuk menciptakan aplikasi yang bisa dijalankan pada perangkat Android. MIT App Inventor memiliki fitur yaitu:

1. Berbasis *Google Blockly*, Pemrograman visual dengan tujuan untuk mempermudah semua orang untuk melakukan pemrograman tanpa harus mengetahui *syntax* program [5];
2. Komponen dan blok *event* tersedia dengan lengkap dan *user* hanya perlu melakukan *drag and drop*;

2.5 Metode Pengolahan data

Mean square error (MSE) adalah metode uji yang dilakukan untuk mengukur *error* [6]. MSE adalah kuadrat dari rata-rata *error* dengan semakin kecil hasil perhitungan, menunjukkan semakin kecil *error* yang terjadi. Rumus untuk mencari MSE seperti yang ditunjukkan pada Persamaan (1).

$$MSE = \frac{\Sigma(Y_i - \hat{Y}_i)^2}{n} \qquad (1)$$

Keterangan:

Y_i = Data sebenarnya

$\hat{Y}_i$ = Nilai prediksi dari variabel Y

n = Banyaknya observasi

Kesalahan relatif adalah perbandingan antara besaran kesalahan terhadap harga yang sebenarnya. Kesalahan merupakan selisih nilai antara nilai yang didapat dari alat ukur dan nilai sebenarnya/nilai yang dijadikan acuan. Rumus kesalahan relatif seperti yang ditunjukkan pada Persamaan (2).

$$\varepsilon_e = \frac{p - p'}{p} \qquad (2)$$

ε_e = Kesalahan relatif

p = Nilai sebenarnya

p' = Nilai pengukuran

2.6 Metode Pengujian

Black box testing adalah metode pengujian perangkat lunak yang menguji fungsionalitas perangkat tersebut tanpa memperhatikan bagaimana cara kerjanya [7], [8]. Pengujian ini berfokus pada *output* berdasarkan *input* yang diberikan tanpa perlu mengetahui program internal perangkat lunak untuk memperoleh *output*. Untuk melakukan pengujian perlu diketahui desain fungsi spesifik yang diinginkan dari perangkat, sehingga tes dapat dilakukan untuk menunjukkan bahwa tiap fungsi beroperasi penuh dan mencari *error* pada tiap fungsi. Adapun kategori *error* dalam metode *black box testing* [9], yaitu:

1. Fungsi yang hilang atau tidak sesuai
2. *Error* pada *interface*
3. *Error* pada struktur data atau akses *database* eksternal
4. *Error* pada kinerja