

BAB III

LANDASAN TEORI

III.1. Sistem Informasi

Sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen - komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi. Sistem informasi adalah seperangkat komponen yang saling berhubungan yang berfungsi mengumpulkan, memproses, menyimpan dan mendistribusikan informasi untuk mendukung pembuatan keputusan dan pengawasan dalam organisasi (Sugara, 2011). Salah satu komponen dari sistem informasi ini berupa basis data. Menurut Poerwanta et al. (2013), database atau basis data merupakan salah satu komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi, menentukan kualitas informasi (akurat, tepat pada waktunya dan relevan). Informasi dapat dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya serta *database* mampu mengurangi pemborosan tempat simpanan luar. Sistem informasi ini terdiri dari dua kata, yakni sistem dan informasi. Berikut adalah pengertian mengenai sistem dan informasi :

III.1.1. Sistem

Suatu sistem adalah suatu jaringan kerja untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Terdapat dua kelompok pendekatan dalam mendefinisikan sistem, yaitu yang menekankan pada prosedurnya dan yang

menekankan pada komponen atau elemennya. Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu mempunyai komponen-komponen (*components*), sistem data (*boundary*), lingkungan luar sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolahan (proses) dan sasaran (*objectives*) atau tujuan (*goal*).

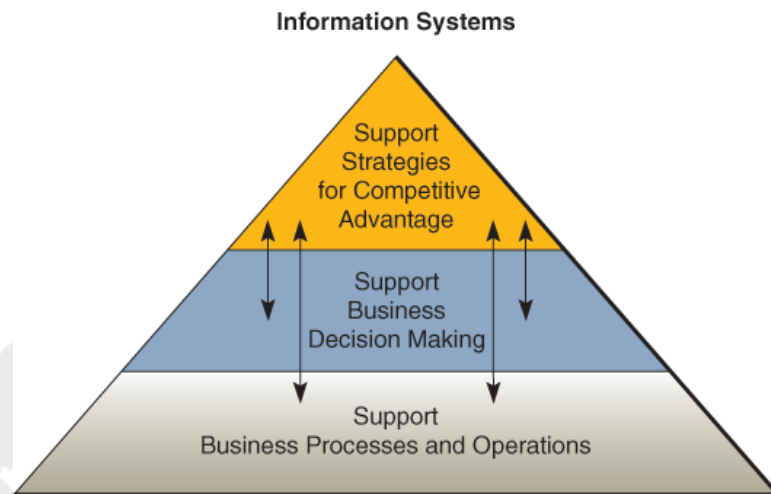
Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Lingkungan luar dari suatu sistem adalah apapun diluar dari batas sistem yang mempengaruhi operasi sistem (Widayanto & Wardati, 2013).

III.1.2. Informasi

Informasi adalah suatu data yang lebih diolah menjadi sebuah bentuk yang penting bagi penerima dan mempunyai nilai yang nyata atau yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau yang akan datang (Widayanto & Wardati, 2013).

III.1.3. Piramida Sistem Informasi

Dalam piramida sistem informasi terdapat 3 level. Level 1 digunakan untuk support proses bisnis dan operator, level 2 digunakan untuk support pengambilan keputusan bisnis, dan level 3 digunakan untuk strategi keunggulan kompetitif. Level 1 memiliki nilai efisiensi, sedangkan level 2 dan 3 memiliki nilai efektivitas.



Gambar 3.2 Sistem Informasi
(O'Brien & Marakas, 2007)

III.2. Sistem Informasi Berbasis Web

Sistem informasi berbasis web secara umum digunakan sebagai sistem informasi yang menjadi standar penyimpanan data, mendapatkan informasi, memformat data, serta menampilkan informasi via *client/server architecture*. Sistem informasi ini menangani seluruh tipe informasi digital meliputi *text, hypermedia, graphics*, dan suara yang sangat mudah digunakan oleh pengguna karena menerapkan *Graphical User Interface (GUI)*.

Sistem informasi berbasis web sama seperti sistem informasi lainnya, tetapi memiliki perbedaan yang terletak pada bahasa pemrograman yang digunakan yaitu bahasa pemrograman web dan juga sistem informasi berbasis web ini bersifat *online*. Sistem informasi berbasis web merupakan media yang digunakan untuk menampilkan informasi mengenai suatu informasi melalui media interaksi seperti media gambar, video, audio, atau gabungan dari

semua media tersebut (Riyadi et al., 2012). Berikut adalah pengertian dari web atau *website* :

III.2.1. Website

Website atau situs juga dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman/*hyperlink* (Wahana, 2006). Informasi dalam *website* pada umumnya ada 2 jenis, yakni informasi statis dan informasi dinamis. Informasi dikatakan statis jika informasi pada *website* tersebut bersifat tetap dan tidak diubah dalam jangka waktu yang lama. Sedangkan informasi dinamis merupakan informasi yang seringkali diubah secara periodik.

III.3. Open Source

Open Source adalah sistem pengembangan yang tidak dikoordinasi oleh suatu individu / lembaga pusat, tetapi oleh para pelaku yang bekerja sama dengan memanfaatkan kode sumber (*source-code*) yang tersebar dan tersedia bebas (biasanya menggunakan fasilitas komunikasi internet). Pola pengembangan ini mengambil model ala *bazaar*, sehingga pola *Open Source* ini memiliki ciri bagi komunitasnya yaitu adanya dorongan yang bersumber dari budaya memberi, yang artinya ketika suatu komunitas menggunakan sebuah program *Open Source* dan telah menerima

sebuah manfaat kemudian akan termotivasi untuk menimbulkan sebuah pertanyaan apa yang bisa pengguna berikan balik kepada orang banyak.

III.3.1. MySQL

MySQL (*My Structured Query Language*) adalah suatu perangkat lunak database relasi (*Relational Database Management System* atau RDBMS). MySQL itu bekerja menggunakan bahasa basis data atau yang sering kita dengar dengan sebutan DBMS (*Database Management System*) (Poerwanta et al., 2013). Pada MySQL, sebuah *database* mengandung satu atau sejumlah tabel. Tabel terdiri dari sejumlah baris dan setiap baris mengandung satu atau beberapa kolom. Fungsi yang dapat dilakukan termasuk membangun objek basis data, memanipulasi objek, mempopulasikan tabel basis data dengan data, memperbarui data yang sudah ada dalam tabel, menghapus data, melakukan *query* basisdata, mengontrol akses basis data dan melakukan administrasi basisdata secara keseluruhan (Arifudzaki et al., 2010).

Data Language ini terbagi dua macam, yaitu:

- a. DDL (*Data Definition Language*), yaitu perintah yang digunakan untuk pendefinisian suatu struktur data. Misalnya menciptakan *database*, *field*, dan sebagainya.
- b. DML (*Database Manipulation Language*), yaitu perintah untuk proses manipulasi data, misalnya *create*, *read*, *update*, dan *delete* (CRUD).

III.3.2. CodeIgniter

CodeIgniter adalah suatu web *application framework* yang bersifat *open-source* yang digunakan untuk membangun aplikasi php dinamis. Tujuan utama pengembangan CodeIgniter adalah untuk membantu developer untuk mengerjakan aplikasi lebih cepat daripada menulis semua kode dari awal. CodeIgniter menyediakan berbagai macam *library* yang dapat mempermudah dalam pengembangan. CodeIgniter sendiri dibangun dengan menggunakan konsep *Model-View-Controller Development Pattern* (Daqiqil, 2011). Dengan adanya model MVC yang digunakan, maka kode yang ada terorganisir dengan baik, serta keamanannya baik. Proses pengembangan dengan *framework* ini lebih cepat dan efisien dibandingkan dengan menggunakan *framework* lainnya (Curteanu, 2010).

III.3.3. Bahasa Pemrograman PHP

PHP merupakan bahasa *open source* yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah system. Hasilnya akan dikirimkan ke klien, tempat pemakai menggunakan *browser*. Secara khusus, PHP dirancang untuk membentuk web dinamis. Artinya ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, dapat ditampilkan isi basis data ke halaman web. Pada prinsipnya, PHP mempunyai fungsi yang sama dengan skrip-skrip

seperti ASP (*Active Server Page*), Cold Fusion, ataupun PERL.

Skrip PHP berkedudukan sebagai *tag* dalam bahasa HTML. Sebagaimana diketahui HTML (*Hyper Text Markup Language*) adalah bahasa standar untuk membuat halaman-halaman web. Model kerja PHP diawali dengan permintaan suatu halaman web oleh *browser*. Berdasarkan URL (*Uniform Resource Locator*) atau dikenal dengan sebutan alamat Internet, *browser* mendapatkan alamat dari *web server*, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh *web server*. Selanjutnya *web server* akan mencarikan berkas yang diminta dan isinya segera dikirim ke mesin PHP dan mesin inilah yang memproses dan memberikan hasilnya (berupa kode HTML) ke *web server* (Arifudzaki et al., 2010).

III.4. Model-View-Control (MVC)

Model View Controller atau MVC adalah sebuah metode yang terbukti dapat membuat aplikasi dengan lebih efektif. Metode MVC ini memecah aplikasi ke dalam tiga modul, yakni model, view, dan controller. Modul model ini berisi kelas-kelas yang dapat digunakan untuk memanipulasi basis data. Keuntungan dari penggunaan MVC ini adalah dapat mengurangi kesalahan syntax di perintah SQL. MVC ini juga membantu mengurangi kompleksitas dari arsitektur dan meningkatkan fleksibilitas dan penggunaan kode dalam pembuatan aplikasi. (Supaartagorn, 2011).

Dengan adanya pemisahan terhadap *model*, *view*, dan *controller* ini maka kerumitan dapat berkurang jika ada penambahan fungsi baru yang diperlukan dalam pembuatan aplikasi (Karim et al., 2011). Selain dapat mengurangi penggunaan kode dalam pembuatan aplikasi, metode MVC ini juga mempermudah dalam proses pemeliharaan serta perbaikan aplikasi (Utpatadevi et al., 2012).

III.5. Arsitektur 3-Tier

Dalam arsitektur 3-Tier *Client-Server*, agen dan server bertindak sebagai klien yang dapat menghasilkan arsitektur yang berbeda dari sistem komputasi terdistribusi. Arsitektur 3-Tier memperluas dasar *client-server* model yang dengan menambahkan tingkat menengah untuk mendukung logika aplikasi dan layanan umum. Berikut adalah 3 tipe komponennya:

- a. Antarmuka pengguna dan pengolahan presentasi
Komponen ini bertanggung jawab untuk menerima inputan dan mempresentasikan hasil. Komponen ini termasuk dalam *client tier*.
- b. Pengolahan fungsi komputasi
Komponen ini bertanggung jawab untuk menyediakan komputasi terdistribusi yang transparan, terpercaya, aman, dan efisien. Komponen ini juga bertanggung jawab untuk melakukan proses yang diperlukan untuk memecahkan masalah aplikasi tertentu. Komponen ini termasuk dalam *application tier*.

c. Pengolahan data akses

Komponen ini bertanggung jawab untuk melakukan akses data yang disimpan pada perangkat penyimpanan eksternal (seperti disk drive). Komponen ini termasuk dalam *back-end tier*.

Dibandingkan dengan arsitektur 2-Tier *Client-Server*, arsitektur 3-Tier memiliki keuntungan, antara lain transparansi yang lebih baik. Server dalam tingkat di arsitektur 3-Tier memungkinkan aplikasi untuk melepaskan user interface dari sumber daya *back-end* nya. Artinya adalah lokasi atau implementasi dari sumber daya *back-end* dapat diubah tanpa mempengaruhi program dalam *client tier*. Keuntungan lainnya adalah skalabilitas yang lebih baik. Arsitektur terpusat dan 2-Tier kurang baik untuk mendukung aplikasi besar (Jia & Zhou, 2005).

III.6. Spare Parts

Klasifikasi dari bermacam-macam barang spare parts sangat penting dalam sistem manajemen inventori. Organisasi yang melakukan penyimpanan spare parts biasanya melakukan klasifikasi dengan kriteria yang berbeda-beda, bergantung pada setiap kategorinya. Kategori yang dapat digunakan untuk manajemen inventori spare parts adalah vital (sangat diperlukan dan penting), penting, dan barang yang diinginkan. Salah satu cara untuk mengurangi penumpukan spare parts di gudang adalah

dengan mempertimbangkan barang yang aktif atau tidak. Diperlukan keputusan yang tepat agar inventori barang terkelola dengan baik (Mesquita & do Rego, 2011).

Manajemen spare parts sendiri termasuk dalam layanan yang berbasis transaksi / Transaction-based-services (Oliva, 2003). Terdapat tiga kebijakan dalam melakukan pengelolaan spare parts dan komponennya, yakni dengan tidak melakukan penyimpanan stok terhadap barang, menyimpan satu barang di stok, dan melakukan penyimpanan dengan jumlah banyak terhadap suatu barang. Pengelolaan ini ditentukan berdasarkan data transaksi yang terjadi (Bošnjaković, 2010).