

BAB 2

TINJAUAN PUSTAKA

2.1. Profil Perusahaan

Pada tahap ini akan dijelaskan mengenai segala aspek mengenai profil perusahaan PT. Alfa Fikrindo Utama.

2.1.1. Sejarah PT. Alfa Fikrindo Utama.

PT. Alfa Fikrindo Utama beralalamat di Jl.Cilemah abang raya blok T1 No.8C Cikarang baru - Bekasi Bekasi 17550 Jawa Barat, Indonesia. Perusahaan ini bergerak dalam bidang pembuatan dan penjualan pemanas elektrik serta komponen sensor suhu untuk menunjang kebutuhan industri otomotif, elektronik, chemical, textil hingga industri makanan.

2.1.2. Visi dan Misi PT. Alfa Fikrindo Utama

Berikut ini adalah visi dan misi PT. Alfa Fikrindo Utama:

2.1.2.1. Visi

Visi PT. Alfa Fikrindo Utama adalah menjadikan PT. Alfa Fikrindo perusahaan penyedia *thermocouple* nomor 1 SEINDONESIA.

2.1.2.2. Misi

Misi PT. Alfa Fikrindo Utama adalah :

1. Meciptakan produk yang inovatif serta berkerja sama dengan mitra usaha.
2. Meningkatkan kualitas produk dan memahami produk secara kontinue.
3. Membuat sistem pemasaran yang memuaskan pelanggan.
4. Menjadikan Costumer pelanggan abadi.

2.1.3. Logo PT. Alfa Fikrindo Utama

Berikut ini adalah logo PT. Alfa Fikrindo Utama:



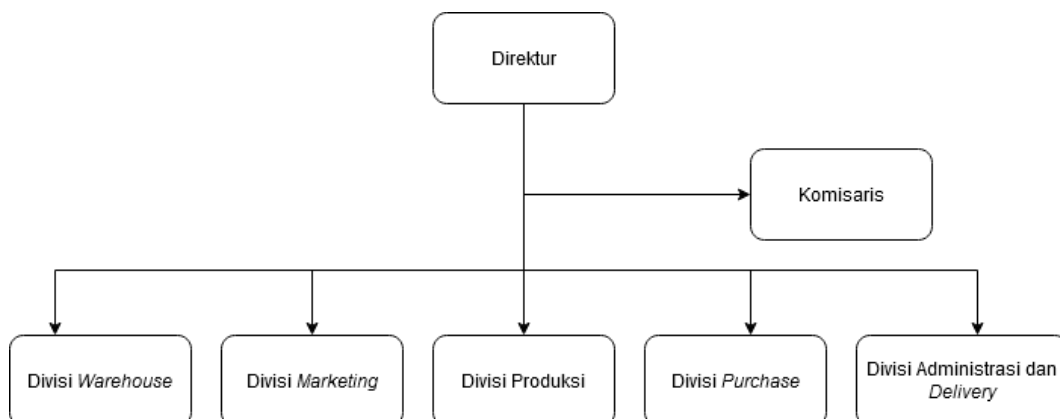
Gambar 2.1 Logo PT. Alfa Fikrindo Utama

Berdasarkan gambar 2.1, Logo PT. Alfa Fikrindo Utama memiliki makna sebagai berikut:

1. Mengerjakan pekerjaan dan mendapatkan hasil yang halal.
2. Mendapatkan hasil pekerjaan yang barokah.
3. Mengerjakan semua pekerjaan dengan istiqomah.

2.1.4. Struktur Organisasi PT. Alfa Fikrindo Utama

Agar semua kegiatan yang terjadi di perusahaan berjalan dengan baik, dibutuhkan struktur organisasi yang bertujuan mempermudah dalam melaksanakan koordinasi dan juga hubungan dalam masing – masing pekerjaan. Berikut ini adalah struktur organisasi pada PT. Alfa Fikrindo Utama yang dapat dilihat pada gambar 2.2.



Gambar 2.2 Struktur Organisasi PT. Alfa Fikrindo Utama

2.1.5. Tugas Dan Wewenang

Berdasarkan struktur organisasi akan diuraikan wewenang dan tanggung jawab masing- masing jabatan yang ada di PT. Alfa Fikrindo.

1. Direktur, mempunyai tugas memimpin pelaksanaan tugas dan fungsi perusahaan.
2. Komisaris, mempunyai tugas pimpinan yang menjalankan proses pemantauan dan pengambilan keputusan di perusahaan.
3. Divisi *Warehouse*, mempunyai tugas mengontrol, dan menyimpan bahan baku untuk digunakan dalam proses produksi.
4. Divisi *Marketing*, mempunyai tugas memimpin dan bertanggung jawab terhadap seluruh proses kegiatan marketing / penjualan agar target perusahaan tercapai.
5. Divisi Produksi, mempunyai tugas melakukan perencanaan, koordinasi dan kontrol dari proses manufaktur dan bertanggung jawab memastikan barang dan jasa diproduksi secara efisien, jumlah produksi yang benar & akurat, diproduksi sesuai dengan anggaran biaya yang tepat dan berkualitas sesuai standar perusahaan.
6. Divisi *Purchase*, mempunyai tugas membeli semua bahan baku yang dibutuhkan dalam proses produksi.
7. Divisi Administrasi dan *Delivery*, mempunyai tugas membuat dan mengelola semua dokumen penjualan dan pengiriman.

2.2. Landasan Teori

Dalam pembangunan sistem informasi manajemen produksi ini terdapat beberapa landasan teori yang digunakan untuk mendukung pembangunan sistem informasi manajemen produksi.

2.2.1. Sistem Informasi

Sistem merupakan sebuah himpunan dari elemen – elemen yang berinteraksi untuk mencapai tujuan tertentu[2]. Sedangkan Informasi adalah kumpulan data yang diolah menjadi bentuk yang lebih berguna dan dapat dimengerti oleh penerimanya[3]. Sehingga dapat disimpulkan bahwa

sistem informasi adalah sebuah komponen-komponen yang terintegrasi yang bertujuan untuk mengumpulkan, menyimpan, memproses data dan memberikan informasi yang telah diolah untuk mencapai tujuan tertentu [4]. Komponen – komponen sistem informasi adalah :

1. Komponen *Input*

Komponen *input* merupakan suatu komponen yang menyalurkan data agar data masuk ke dalam sistem.

2. Komponen Model

Komponen model adalah kombinasi dari prosedur, logika, dan model matematika yang memproses data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan

3. Komponen Output

Komponen output merupakan keluaran dari suatu informasi yang berkualitas dan dokumentasi untuk semua tingkatan dari manajemen hingga pemakai sistem.

4. Komponen Teknologi

Teknologi adalah alat dalam sistem informasi yang digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran (*output*) serta membantu pengendalian sistem secara keseluruhan.

5. Komponen Basis Data

Komponen basis data adalah kumpulan data yang saling berhubungan antara satu dengan yang lainnya yang tersimpan di dalam komputer dengan bantuan perangkat lunak *database*.

6. Komponen Kontrol

Komponen kontrol merupakan pengendalian yang dirancang untuk menanggulangi gangguan atau kesalahan – kesalahan terhadap sistem informasi sehingga kesalahan – kesalahan yang terjadi terhadap sistem informasi bisa diatasi dengan cepat.

2.2.2. Manajemen

Menurut Kreitner[5], manajemen adalah proses penyelesaian masalah untuk mencapai tujuan organisasi secara efektif melalui penggunaan sumber daya secara efisien sesuai dengan perkembangan. Manajemen bertujuan untuk mengatur pemanfaatan sumber daya yang ada secara efektif dan efisien serta mengkoordinasi segala kegiatan agar tujuan organisasi yang ditetapkan tercapai. Manajemen dibutuhkan untuk mengorganisasikan, mengakses dan mengungkit data dan informasi perusahaan untuk pengambilan keputusan. Dalam pengambilan keputusan, eksekutif perusahaan melakukan perencanaan strategis untuk keseluruhan organisasi, area bisnis, dan sumber daya informasi. Sehingga sebuah rencana strategis untuk sumber daya informasi akan mengidentifikasi tujuan-tujuan yang harus dipenuhi oleh perusahaan pada tahun-tahun mendatang dan informasi yang akan diperlukan untuk mencapai tujuan-tujuan tersebut.

2.2.3. Sistem Informasi Manajemen

Sistem Informasi Manajemen adalah sebuah kumpulan prosedur gabungan yang mengumpulkan dan menghasilkan data yang terorganisir dengan baik untuk mendukung proses pengambilan keputusan suatu organisasi. Menurut Jaluanto [6], tugas utama SIM di dalam organisasi adalah sebagai aliran umpan-balik. Sistem Informasi Manajemen sebagai sarana menyelesaikan *planning* dan sekaligus alat *monitoring* dengan membandingkan hasil *planning* dengan standar yang telah ditetapkan. Ketika perusahaan atau organisasi menginginkan umpan-balik yang positif, diperlukanya sebuah pengawasasn organisasi berdasarkan pada prinsip umpan-balik yang negatif lalu dibandingkan dengan standarnya.

Lalu penyimpangan berdasarkan umpan-balik yang negatif akan diterjemahkan ke dalam bentuk tindakan evaluasi yakni dimana umpan-balik tersebut akan dijadikan perbaikan untuk memperbaiki prestasi unit bisnis. Dengan demikian Sistem Informasi Manajemen dapat membantu menemukan penyimpangan dari *planning* yang ditetapkan dan sistem informasi manajemen juga untuk membuat tindakan koreksi yang dibutuhkan oleh organisasi.

2.2.4. Sistem Informasi Manajemen Produksi

Sistem Informasi Manajemen Produksi adalah sebuah kumpulan prosedur gabungan yang mengumpulkan dan menghasilkan data yang terorganisir dengan baik untuk mendukung proses pengambilan keputusan dalam proses produksi. Dengan adanya sistem informasi manajemen produksi, sebuah perusahaan dapat melakukan pengendalian internal baik dari tahapan rencana, *monitoring*, hingga evaluasi terhadap penyimpangan – penyimpangan yang terjadi dalam proses produksi. Dengan mengatur manajemen produksi, maka perusahaan mendapatkan hasil akhir produk sesuai dengan standar dan sesuai dengan rencana produksi. Penggunaan sistem informasi manajemen produksi juga membantu perusahaan untuk menekan biaya yang dikeluarkan oleh perusahaan agar perusahaan mendapatkan laba sebanyak-banyaknya dengan menggunakan biaya yang sedikit.

2.2.5. Earliest Due Date

Metode *Earliest Due Date* adalah salah satu metode yang digunakan dalam penjadwalan produksi. Metode Earliest Due Date adalah metode yang digunakan dalam penjadwalan produksi yang mesin tunggal (*single machine*). Menurut Hamida[7], *earliest due date* adalah pengurutan pekerjaan yang didasarkan pada batas waktu yang tercepat. Dan menurut , metode *earliest due date* adalah metode yang melakukan urutan penyelesaian produksi berdasarkan tanggal dimana harus diselesaikan. Sehingga dapat disimpulkan bahwa *earliest due date* adalah metode

penjadwalan untuk menyelesaikan proses produksi dengan cara mengurutkan batas waktu (*due date*) tercepat. Adapun tahapan – tahapan dalam metode Earliest Due Date, yaitu;

1. Mengurutkan *job* berdasarkan pesanan yang memiliki due date tercepat atau terkecil.
2. Mengelompokkan *job* berdasarkan spesifikasi dan jumlah mesin yang akan digunakan.
3. Menghitung total waktu untuk menyelesaikan pekerjaan (Completion Time) dan menghitung keterlambatan untuk tiap *job* (Lateness). Persamaan untuk Completion Time dan Lateness sebagai berikut:

Completion Time

$$C_i = \sum t_i$$

Lateness

$$L_i = C_i - d_i$$

Keterangan

C_i : Completion Time atau waktu yang dibutuhkan untuk menyelesaikan pekerjaan.

t_i : Waktu yang dibutuhkan untuk mengerjakan suatu pekerjaan.

L_i : Selisih antara completion time dengan due date-nya.

d_i : Batas waktu dimana operasi terakhir dari suatu pekerjaan harus selesai.

4. Menghitung keterlambatan rata-rata, waktu penyelesaian rata-rata, utilitas, dan jumlah *job* rata-rata dengan persamaan sebagai berikut:

Keterlambatan Rata-Rata

$$\frac{\sum L_i}{n \text{ job}}$$

Waktu Penyelesaian Rata-Rata

$$\frac{\sum C_i}{n \text{ job}}$$

Utilitas

$$\sum \frac{t_i}{c_i} * 100\%$$

Jumlah *Job* Rata-Rata

$$\sum \frac{C_i}{t_i}$$

Keterangan:

C_i : Completion Time atau waktu yang dibutuhkan untuk menyelesaikan pekerjaan.

t_i : Waktu yang dibutuhkan untuk mengerjakan suatu pekerjaan.

L_i : Selisih antara completion time dengan due date-nya.

d_i : Batas waktu dimana operasi terakhir dari suatu pekerjaan harus selesai.

$n \text{ job}$: Jumlah pekerjaan yang harus dilakukan.

2.2.6. Tools

Merupakan alat yang digunakan untuk menggambarkan bentuk logika model dari suatu sistem dengan menggunakan simbol-simbol, lambang-lambang, diagram-diagram yang menunjukkan secara tepat arti dan fungsinya. Adapun peralatan pendukung (*tools system*) yang dijelaskan sebagai model sistem yang akan dirancang adalah sebagai berikut

2.2.6.1. Hyper Text Markup Language (HTML)

Sebuah bahasa *markup* yang digunakan untuk membuat sebuah halaman *web* dan menampilkan berbagai informasi di dalam sebuah *browser* Internet. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan

yang disebut dengan SGML (*Standard Generalized Markup Language*), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide Web Consortium (W3C)*.

HTML adalah dasar terbentuknya web. HTML berupa kode-kode tag yang menginstruksikan *browser* untuk menghasilkan tampilan sesuai dengan yang diinginkan. Sebuah file yang merupakan file HTML dapat dibuka dengan menggunakan *browser web* seperti Mozilla Firefox atau Microsoft Internet Explorer. HTML dokumen tersebut mirip dengan dokumen teks biasa, hanya dalam dokumen ini sebuah teks bisa memuat instruksi yang ditandai dengan kode atau lebih dikenal dengan TAG tertentu. Tanda tersebut digunakan untuk menentukan format atau style dari teks yang ditandai.

2.2.6.2. Hypertext Preprocessor (PHP)

PHP (*Hypertext Preprocessor*) merupakan script untuk pemrograman web server-side, script yang membuat dokumen HTML secara *on the fly*, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan teks editor atau editor HTML. Dengan menggunakan PHP, *maintenance* suatu situs web menjadi lebih mudah. Proses update data dapat dilakukan dengan menggunakan aplikasi yang dibuat dengan menggunakan script PHP. PHP/FI merupakan nama awal dari PHP, dibuat pertama kali oleh Rasmus Lerdoff. PHP awalnya merupakan program CGI yang dikhususkan untuk menerima input melalui form yang ditampilkan dalam *browser web*. Software ini disebar dan dilisensikan sebagai perangkat lunak *Open Source*. Adapun keunggulan yang dimiliki oleh PHP adalah:

1. *Life Cycle* yang sangat singkat, sehingga PHP selalu *up to date* mengikuti perkembangan teknologi internet.

2. Cross Platform, yakni PHP dapat dipakai di hampir semua *web server* yang ada di pasaran (terutama Apache dan Microsoft IIS) dan dijalankan pada berbagai sistem operasi (Linux, Windows, FreeBSD).
3. PHP mendukung koneksi ke banyak *database* baik yang gratis maupun komersial, seperti MySQL, mySQLi, Oracle, Microsoft SQL Server, *Interbase*, dan banyak lagi.
4. PHP bersifat *Open Source* dan gratis. Kemudahan dalam mendapatkan dokumentasi di Internet, tidak akan sulit untuk mencari baik itu referensi, kode-kode PHP yang sudah jadi dan juga mengajukan pertanyaan pada grup-grup diskusi yang di dalamnya banyak sekali para master PHP.

2.2.6.3. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*database management system*) atau DBMS yang *multithread* dan *multi-user*, MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Keandalan suatu sistem basis data (DBMS) dapat diketahui dari cara kerja pengoptimal-nya dalam melakukan proses perintah-perintah SQL yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. MySQL mendukung operasi basis data transaksional maupun operasi basis data non-transaksional. Pada modus operasi non- transaksional, MySQL dapat dikatakan unggul dalam hal unjuk kerja dibandingkan perangkat lunak peladen basis data kompetitor lainnya. Namun demikian pada modus non-transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modus

non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi blogging berbasis web (wordpress), CMS, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modus basis data transaksional, hanya saja sebagai konsekuensinya unjuk kerja MySQL pada modus transaksional tidak secepat unjuk kerja pada modus non-transaksional.

2.2.6.4. *Business Process Model And Notation (BPMN)*




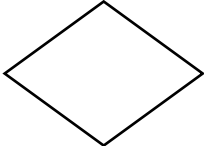
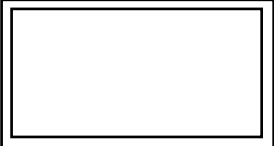
Business Process Model And Notation (BPMN) adalah penggambaran terhadap suatu bisnis dengan menggunakan proses diagram yang didasarkan kepada teknik diagram alur, dirangkai untuk membuat model-model grafis dari operasi-operasi bisnis dimana terdapat aktivitas dan kontrol-kontrol alur yang mendefinisikan urutan kerja. BPMN dirancang bukan hanya mudah digunakan dan dipahami, tetapi juga memiliki kemampuan untuk memodelkan proses bisnis yang kompleks dan dapat dirancang secara spesifik. Tujuan utama dari BPMN adalah memberikan notasi standar yang mudah dipahami oleh semua pengelola yang memiliki kepentingan bisnis. Pengembang teknis bertanggung jawab atas penerapan BPMN yang telah dirancang oleh perancang BPMN yang telah menganalisis proses bisnis dan menyempurnakan semua proses yang dilakukan dalam proses bisnis dan manajer bisnis memantau dan mengelola proses bisnis. Sehingga BPMN berfungsi sebagai bahasa umum untuk memperantarai kesenjangan komunikasi yang sering terjadi antara desain proses bisnis dengan implementasi proses bisnis yang telah dirancang.

2.2.6.5. *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) adalah sebuah diagram struktural yang digunakan untuk merancang sebuah *database*[8]. Sehingga ERD digunakan untuk merancang suatu basis data yang digunakan untuk memperlihatkan suatu relasi antar entitas atau objek yang terlihat beserta

atributnya. Berikut ini adalah simbol – simbol yang digunakan dalam perancangan Entity Relationship Diagram yang dapat dilihat pada tabel :

Table 2.1 Simbol *Entity Relationship Diagram*

| No | Simbol | Keterangan |
|----|---|---|
| 1 |  | Menggambarkan keberadaan sebuah entitas (entitas kuat) |
| 2 |  | Menggambarkan atribut yang dimiliki oleh suatu entitas atau relasi |
| 3 |  | Menggambarkan hubungan entitas dan relasi atau entitas dengan atribut |
| 4 |  | Menggambarkan keterhubungan antar relasi |
| 5 |  | Menunjukkan himpunan entitas lemah |

Dalam penggunaan ERD, relasi dapat terdiri dari sejumlah entitas yang disebut derajat relasi. Derajat relasi maksimum disebut dengan kardinalitas relasi sedangkan derajat minimum disebut dengan modalitas. Jadi kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas lain.

Kardinalitas relasi yang terjadi diantara dua himpunan entitas berbeda dapat berupa :

1. Satu ke satu (one to one)

Setiap entitas pada himpunan entitas A dapat berelasi dengan paling banyak satu entitas pada himpunan entitas B, demikian juga sebaliknya.

2. Satu ke banyak (one to many)

Setiap entitas pada himpunan entitas A dapat berelasi dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya. Atau Setiap entitas pada himpunan entitas A hanya dapat berelasi dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya.

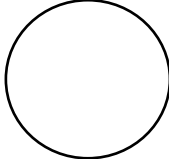
3. Banyak ke banyak (many to many)


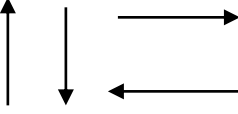

Setiap entitas pada himpunan entitas A dapat berelasi dengan banyak entitas pada himpunan entitas B, demikian juga sebaliknya.

2.2.6.6. Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses yang dikenakan pada data tersebut[9]. Sehingga dapat disimpulkan bahwa DFD adalah alat pemodelan untuk memodelkan alur kerja sistem. Adapun beberapa simbol yang digunakan untuk penggambaran data beserta proses transformasi data dalam DFD dapat dilihat pada tabel 2.2.

Table 2.2 Simbol-Simbol DFD

| No | Simbol | Keterangan |
|----|---|---|
| 1. |  | Menunjukkan kegiatan / kerja yang dilakukan oleh orang, mesin atau komputer |

| | | |
|----|---|--|
| 2. |  | Menunjukkan bagian dari entitas luar |
| 3. |  | Menunjukkan arus atau aliran dari proses |
| 4. |  | Menunjukkan simpanan dari data yang dapat berupa suatu <i>file</i> / database di sistem komputer |

2.2.7. Pengujian Perangkat Lunak

Pengujian perangkat lunak adalah suatu aktivitas-aktivitas yang bertujuan untuk mengevaluasi segala kemampuan dari sebuah program atau sistem dan menentukan apakah program yang dibangun sesuai dengan hasil yang diharapkan. Pengujian perangkat lunak merupakan bagian dari Software Development Life Cycle (SDLC) yang wajib dilakukan untuk mengidentifikasi semua kesalahan dan kelengkapan kebutuhan fungsional / nonfungsional. Meskipun tidak semua kesalahan (error) tidak dapat diidentifikasi, setidaknya dapat mengurangi kesalahan dari semua fungsi dalam sistem. Pengujian perangkat lunak juga dapat dinyatakan sebagai proses validasi dan verifikasi bahwa sistem yang dibangun memenuhi persyaratan teknis dalam desain dan pengembangannya dan dapat bekerja seperti yang diharapkan. Tujuan utama dari pengujian perangkat lunak adalah :

1. Menemukan kesalahan yang dibuat oleh programmer ketika membangun sebuah sistem.
2. Mendapatkan kepercayaan dan memberikan informasi tentang tingkat kualitas.
3. Mencegah kecacatan fungsionalitas dan tampilan.

4. Memastikan bahwa hasil akhir memenuhi proses bisnis dan kebutuhan pengguna.
5. Untuk mendapatkan kepercayaan dari pelanggan dengan menyediakan produk yang berkualitas.

2.2.7.1. Pengujian *Black-Box*

Pengujian *Black-Box* merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program [10]. Pada pengujian *Black-Box*, pengujian ini dilakukan dari kondisi diluar deskripsi yang terdapat pada *software* [11]. Sehingga pengujian black-box hanya meneliti fungsionalitasnya saja tanpa memperhatikan struktur atau cara kerja internalnya. Pengujian kasus dibangun di sekitar spesifikasi dan persyaratan, yakni seputar apa yang harus dilakukan oleh aplikasi yang dibangun. Pengujian pada Black-Box berusaha untuk menemukan kesalahan seperti :

1. Fungsi-fungsi yang tidak benar.
2. Kesalahan Interface
3. Kesalahan dalam struktur data atau akses database
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi

2.2.7.2. User Acceptance Test

User Acceptance Test adalah tahap akhir pada pengujian yang dijalankan untuk mengetahui apakah masih terdapat kecacatan pada aplikasi yang dibangun / dikembangkan. UAT termasuk ke dalam *black-box* testing di mana pengguna tidak mengecek internal sistem, namun membandingkan fungsionalitas sistem dengan requirement yang dispesifikasikan oleh pembuat UAT [12]. Tujuan utama UAT adalah untuk memvalidasi pengembangan sistem / aplikasi yang dibangun terhadap kebutuhan bisnis. UAT terdiri dari beberapa langkah pengujian, yang bertujuan untuk memverifikasi apakah kebutuhan penggunaan

aplikasi yang telah dibangun bisa dimanfaatkan dengan baik dan benar oleh pengguna. Jika pengguna puas dan dapat menjalankan sistem yang dibangun, maka bisa dikatakan sistem yang dibangun itu sudah bagus. User Acceptance Test biasanya dilakukan pada tahapan terakhir sebelum produk dikeluarkan atau dirilis, atau bisa dikatakan sebelum produk diterima oleh calon pengguna sesungguhnya.

Menurut Spillner dkk [13], dalam pelaksanaan UAT direkomendasikan penguji yang akan menguji sistem itu bukan *user* yang ditujukan melainkan salah satu anggota dari perusahaan atau *customer* perusahaan. Dalam pelaksanaan UAT, penguji yang bukan *end user* biasanya memiliki ekspektasi yang berbeda dalam melakukan pengujian sistem yang baru. *User* tersebut biasanya menolak menggunakan sistem karena menurut mereka tidak terbiasa menggunakan sistem tersebut, yang dimana dapat memberikan kesan negatif pada pengenalan sistem. Hal ini dapat terjadi bahkan jika sistem yang dibangun sesuai dengan spesifikasi fungsional. Sehingga, berdasarkan permasalahan diatas diperlukan untuk mengorganisasi UAT kepada setiap hak akses.