

BAB III

LANDASAN TEORI

Landasan teori digunakan untuk menyelesaikan masalah secara sistematis. Pada bab ini akan membahas landasan teori yang meliputi landasan teori yang membahas tentang ilmu yang terkait dalam permasalahan tersebut.

3.1 Konsep Dasar Sistem

Terdapat dua kelompok di dalam mendefinisikan sistem, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponennya atau elemennya.

Pendekatan sistem yang lebih menekankan pada prosedurnya mendefinisikan sistem sebagai berikut:

Suatu sistem merupakan suatu jaringan kerja dari prosedur – prosedur yang saling berhubungan, berkumpul bersama – sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu menurut Ladjamudin (2005,3)

Sistem itu sendiri memiliki karakteristik atau beberapa sifat tertentu, yaitu mempunyai komponen (*components*), batas sistem (*boundry*), lingkungan sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*process*), dan sasaran suatu tujuan (*goal*). Adapun penjelasan dari karakteristik suatu sistem di atas adalah sebagai berikut:

a. Komponen Sistem (*components*)

Bagian sistem yang saling berinteraksi dan membentuk satu kesatuan, komponen atau elemen sistem dapat berupa subsistem atau beberapa bagian sistem.

b. Batas sistem (*Boundary*)

Daerah yang membatasi antara suatu sistem dengan lingkungannya atau dengan sistem lainnya. Batas sistem inilah yang membuat sistem di pandang sebagai satu kesatuan.

c. Lingkungan luar sistem (*environments*)

Segala sesuatu yang berada di luar sistem yang mempengaruhi sistem lingkungan luar sistem dapat bersifat menguntungkan sistem atau merugikan sistem.

d. Penghubung sistem (*interface*)

Merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Penghubung inilah yang menyebabkan beberapa subsistem berintegrasi dan membentuk satu kesatuan.

e. Masukan sistem (*input*)

Sesuatu yang di masukan ke dalam sistem yang berasal dari lingkungan.

f. Keluaran sistem (*output*)

Suatu hasil dari proses pengolahan sistem yang dikeluarkan ke lingkungan.

g. Pengelolaan sistem (*process*)

Bagian dari sistem yang mengubah masukan menjadi keluaran. Sasaran sistem adalah sesuatu yang menyebabkan mengapa sistem itu dibuat atau ada.

Suatu sistem di katakan berhasil bila mengenai sasaran atau tujuannya.

3.2 Pengertian Penggajian

A. Definisi Penggajian

Penggajian berasal dari kata gaji. Gaji mempunyai beberapa definisi. Berikut terdapat berbagai definisi gaji diambil dari berbagai sumber:

1. Hasibuan (2002), menyatakan bahwa “Gaji adalah balas jasa yang dibayar secara periodik kepada karyawan tetap serta mempunyai jaminan yang pasti” (p. 118).
2. Hariandja (2002), yaitu gaji merupakan salah satu unsure yang penting yang dapat mempengaruhi kinerja karyawan, sebab gaji adalah alat untuk memenuhi berbagai kebutuhan pegawai, sehingga dengan gaji yang diberikan pegawai akan termotivasi untuk bekerja lebih giat.
3. Siswanto (1989), yaitu: gaji dapat berperan dalam meningkatkan motivasi karyawan untuk bekerja efektif, meningkatkan produktivitas dalam perusahaan, serta mengimbangi kekurangan dan keterlibatan komitmen yang menjadi ciri angkatan kerja masa kini. Perusahaan yang tergolong modern, saat ini banyak mengaitkan gaji dengan kinerja.

B. Fungsi Penggajian

Menurut Komarudin (1995), fungsi gaji bukan hanya membantu manager personalia dalam menentukan gaji yang adil dan layak saja, tetapi masih ada fungsi-fungsi yang lain, yaitu (p. 164):

1. Untuk menarik pekerja yang mempunyai kemampuan ke dalam organisasi.
2. Untuk mendorong pekerja agar menunjukkan prestasi yang tinggi.
3. Untuk memelihara prestasi pekerja selama periode yang panjang.

C. Tujuan Penggajian

Menurut Hasibuan (2002), tujuan penggajian adalah sebagai berikut:

a. Ikatan Kerja Sama

Dengan pemberian gaji terjalinlah ikatan kerja sama formal antara majikan dengan karyawan. Karyawan harus mengerjakan tugas – tugasnya dengan baik, sedangkan pengusaha atau majikan wajib membayar gaji sesuai dengan perjanjian yang disepakati.

b. Kepuasan Kerja

Dengan balas jasa, karyawan akan dapat memenuhi kebutuhan – kebutuhan fisik, status sosial, dan egoistiknya sehingga memperoleh kepuasan kerja dari jabatannya.

c. Pengadaan Efektif

Jika program gaji ditetapkan cukup besar, pengadaan karyawan yang *qualified* untuk perusahaan akan lebih mudah.

d. Motivasi

Jika balas jasa yang diberikan cukup besar, manager akan mudah memotivasi bawahannya.

e. Stabilitas Karyawan

Dengan program kompensasi atas prinsip adil dan layak serta eksternal konsistensi yang kompentatif maka stabilitas karyawan lebih terjamin karena turnover relatif kecil.

f. Disiplin

Dengan pemberian balas jasa yang cukup besar maka disiplin karyawan semakin baik. Karyawan akan menyadari serta menaati peraturan-peraturan yang berlaku.

3.3 Sistem Penggajian

Sistem Penggajian adalah suatu sistem yang mengorganisasikan serangkaian prosedur dan metode yang dirancang untuk menghasilkan, menganalisa, menyebarkan dan memperoleh informasi guna mendukung pengambilan keputusan mengenai perhitungan gaji karyawan.

Sistem penggajian merupakan fungsi penting yang menjadi tanggung jawab manajemen sumber daya manusia. sistem penggajian bertugas mencatat dan memproses data yang digunakan untuk membayar pegawai atas layanan yang mereka berikan. Penggajian adalah memberikan kompensasi untuk pegawai yang berupa gaji sebagai kembalian *financial* kepada para pegawai sebagai ganti kontribusi mereka terhadap organisasi / instansi.

3.4 Analisa dan Perancangan Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian – bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan, kesempatan, dan hambatan yang terjadi serta kebutuhan yang diharapkan sehingga dapat diusulkan perbaikannya.

Analisis sistem dilakukan setelah tahap perencanaan sistem dan sebelum tahap perancangan sistem. Langkah-langkah dasar dalam melakukan analisa sistem:

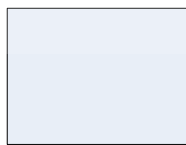
1. *Identify* , yaitu mengidentifikasi masalah.
2. *Understand* , yaitu memahami kinerja dari sistem yang ada.
3. *Analyze*, yaitu menganalisa sistem.

4. *Report*, yaitu membuat laporan hasil analisis

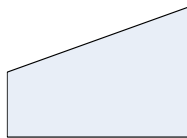
Setelah analisis sistem dilakukan, tahap selanjutnya adalah perancangan sistem. Perancangan sistem mempunyai dua tujuan utama, yaitu untuk memenuhi kebutuhan kepada pemakai serta memberikan gambaran yang jelas dan lengkap kepada pemrogram dan ahli teknik lainnya yang terlibat menurut Jogiyanto (1990).

3.4.1 Sistem *Flow*

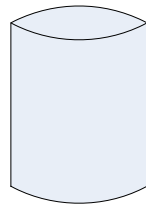
Sistem *Flow* menurut Jogiyanto (2005), adalah bagan yang menunjukkan arus pekerjaan secara menyeluruh dari suatu sistem dimana bagan ini menjelaskan urutan prosedur-prosedur yang ada di dalam sistem. Dalam membuat Sistem *Flow* sebaiknya ditentukan pula fungsi-fungsi yang melaksanakan dan bertanggung jawab terhadap sub-sub sistem. Simbol-simbol yang digunakan dalam sistem *flow* antara lain adalah:



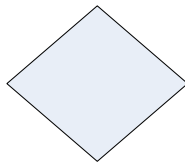
Simbol proses menunjukkan kegiatan proses dari operasi komputer



Simbol inputan *keyboard*, menunjukkan data yang diinputkan melalui *keyboard*



Simbol database

Simbol *decision* (keputusan)

3.4.2 *Data Flow Diagram*

Data flow diagram menggambarkan penyimpanan data dan proses mentransformasikan data. *Data flow diagram* menunjukkan hubungan antara data dan proses pada sistem. *Data Flow Diagram* (DFD) adalah gambaran keseluruhan kerja sistem secara garis besar. *Data flow diagram* merupakan peralatan yang berfungsi untuk menggambarkan secara rinci mengenai sistem sebagai jaringan kerja antar dari dan kemana data mengalir serta penyimpanannya menurut Jogiyanto (2005).

Data flow diagram sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dibuat/dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut disimpan. *Data flow diagram* terdiri dari:

a. *Data Flow Diagram Context Level*

Data flow diagram Context Level merupakan bagian dari *data flow diagram* yang mewakili gambaran keseluruhan sistem. *Data flow diagram Context Level* biasa disebut dengan *context diagram*. *Context diagram* merupakan *data flow diagram* pertama dalam proses bisnis. *Context diagram* juga menunjukkan semua entitas luar yang menerima informasi atau memberikan informasi ke sistem.

b. *Data Flow Diagram Levelled*

Data flow diagram levelled adalah bagian dari *data flow diagram* yang menggambarkan jaringan kerja antara fungsi yang terhubung satu sama lain dengan aliran dan penyimpanan data. *Data flow diagram levelled* terdiri dari beberapa level yaitu:

1. Level 0 Diagram

Menunjukkan semua proses utama yang menyusun keseluruhan sistem. Pada level ini menunjukkan bagaimana proses-proses utama di relasikan menggunakan data flow dan di tunjukkan bagaimana proses-proses utama terhubung dengan entitas eksternal.

2. Level 1 Diagram

Diaram level 1 diciptakan dari proses pada level 0. Level ini menunjukkan proses-proses internal yang menyusun proses utama dalam level 0. Sekaligus menunjukkan bagaimana informasi berpindah dari satu proses ke proses yang lainnya.

3. Level 2 Diagram

Menunjukkan semua proses yang menyusun sebuah proses pada level 1, bisa saja penyusunan DFD tidak mencapai level 2 ini. Atau mungkin harus di lanjutkan ke level berikutnya (level 3, level 4 dan seterusnya).

3.4.3 Entity Relationship Diagram

Entity relationship diagram adalah suatu diagram yang digunakan untuk memodelkan struktur data dan hubungan antar data. *Entity relationship diagram* berfungsi menggambarkan relasi dari dua file atau dua table yang dapat di golongkan dalam tiga macam bentuk relasi yaitu *one to one*, *one to many*, dan *many to many*.

a. *One to one relationship* (Hubungan satu ke satu)

One to one relationship (hubungan satu kesatu) yaitu relasi satu lawan satu yang terjadi apabila satu entity hanya punya relasi dengan file lain. Misalnya satu pekerja hanya boleh memiliki satu jenis pekerjaan. Hubungan ini dinotasikan dengan 1:1.

b. *One to many relationship* (hubungan satu banyak)

One to many relationship yaitu relasi satu lawan banyak yang terjadi apabila satu entity mempunyai banyak relasi dengan file lain. Misalnya satu pekerja boleh mengerjakan banyak jenis pekerjaan. Hubungan ini di notasikan dengan 1:M.

c. *Many to many relationship* (hubungan banyak ke banyak)

Many to many relationship yaitu relasi banyak lawan banyak yang terjadi apabila satu entity mempunyai banyak relasi ke file lain dan file tersebut juga mempunyai banyak relasi ke banyak entity. Misalnya satu pekerja dapat mengerjakan banyak jenis pekerjaan dan satu jenis pekerjaan tersebut dapat juga di kerjakan oleh banyak pekerja. Hubungan ini di notasikan dengan M:M atau M:N. M:N di gunakan apabila jumlah relasi antara kedua file tidak sama.

