

BAB II

LANDASAN TEORI

2.1 Parkir

Parkir adalah keadaan tidak bergerak dari suatu kendaraan yang bersifat sementara (Direktorat Jendral Perhubungan Darat, 1996). Selain Pengertian di atas beberapa ahli memberikan definisinya tentang parkir, yaitu:


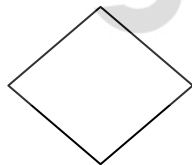

1. Semua kendaraan tidak mungkin bergerak terus, pada suatu saat ia harus berhenti untuk sementara waktu (menurunkan muatan) atau berhenti cukup lama yang disebut parkir (warpani, 1990).
2. Jangka waktu parkir (parking duration) adalah lama parkir suatu kendaraan untuk satu ruang parkir (Morlok, 1995).
3. Parkir adalah memangkalkan/menempatkan dengan memberhentikan kendaraan angkutan orang/barang (bermotor/tidak bermotor) pada suatu tempat parkir dalam jangka waktu tertentu. (Peraturan Pemerintah Daerah Kota Semarang No. 11 tahun 1998).

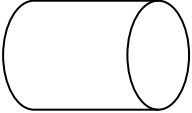


Berdasarkan dari definisi di atas maka dapat ditarik kesimpulan bahwa parkir adalah suatu keadaan tidak bergerak sutau kendaraan bermotor atau tidak bermotor yang dapat merupakan awal dari perjalanan dengan jangka waktu tertentu sesuai dengan keadaan dan kebutuhannya yang membutuhkan suatu area sebagai tempat pemberhentian yang diselenggarakan baik oleh pemerintah maupun pihak lain yang dapat berupa perorangan maupun badan usaha.

2.2 System Flow

Menurut Basuki (2003), *system flow* adalah bagian yang menunjukkan arus pekerjaan secara menyeluruh dari suatu sistem dimana bagan ini menjelaskan urutan prosedur-prosedur yang ada dalam sistem dan biasanya dalam membuat *system flow* sebaiknya ditentukan pada fungsi yang melaksanakan atau bertanggung jawab terhadap sub-sub sistem. Bagan alir sistem menggunakan simbol sebagaimana terdapat pada tabel 2.1

Tabel 2. 1 Simbol Bagan *System Flow*

No	Simbol	Nama Simbol	Keterangan
1		Dokumen	Simbol ini digunakan untuk menunjukkan dokumen input dan output baik untuk proses manual, mekanik, atau komputer.
2		Keputusan	Simbol keputusan digunakan untuk menggambarkan suatu kondisi yang mengharuskan sistem untuk memilih tindakan yang akan dilakukan berdasarkan criteria tertentu.
3		Operasi manual	Simbol ini digunakan untuk menggambarkan proses yang terjadi secara manual yang tidak dapat dihilangkan dari sistem yang ada

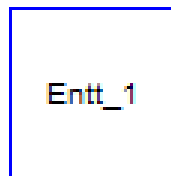
4		Database	Simbol ini digunakan untuk menggambarkan media penyimpanan yang digunakan untuk menyimpan data pada sistem yang akan dibuat.
5		Proses	Simbol proses digunakan untuk menggambarkan proses yang terjadi dalam sistem yang akan dibuat
6		Input manual	Simbol Proses yang digunakan untuk menggambarkan proses yang terjadi dalam sistem yang akan dibuat.

2.3 Data Flow Diagram

Menurut Kendall (2003), *Data Flow Diagram* menggambarkan pandangan sejauh mungkin mengenai masukan, proses dan keluaran sistem, yang berhubungan dengan masukan, proses, dan keluaran dari model sistem yang dibahas. Serangkaian diagram aliran data berlapis juga bisa digunakan untuk merepresentasikan dan menganalisis prosedur-prosedur mendetail dalam sistem. Prosedur-prosedur tersebut yaitu konseptualisasi bagaimana data-data berpindah di dalam organisasi, proses-proses atau transformasi dimana data-data melalui, dan apa keluarannya. Jadi, melalui suatu teknik analisa data terstruktur yang disebut *Data Flow Diagram*, penganalisis sistem dapat merepresentasi proses-proses data di dalam organisasi. Menurut Kendall (2003), dalam memetakan *Data Flow Diagram*, terdapat beberapa simbol yang digunakan antara lain:

1. *External entity*

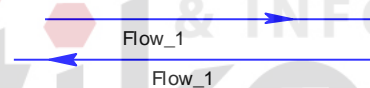
Suatu *external entity* atau entitas luar merupakan orang, kelompok, departemen, atau sistem lain di luar sistem yang dibuat dapat menerima atau memberikan informasi atau data ke dalam sistem yang dibuat.



Gambar 2.1 Simbol *External Entity*

2. *Data Flow*

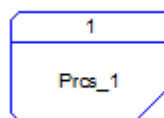
Data Flow atau aliran data disimbolkan dengan data tanda panah. Aliran data menunjukkan arus data atau aliran data yang menghubungkan dua proses atau entitas dengan proses.



Gambar 2.2 Simbol *Data Flow*

3. *Process*

Suatu proses dimana beberapa tindakan atau sekelompok tindakan dijalankan.



Gambar 2.3 Simbol *Process*

4. *Data Store*

Data store adalah simbol yang digunakan untuk melambangkan proses penyimpanan data.

1	Stor_2
---	--------

Gambar 2.4 Simbol *Data Store*

2.4 *User Interface*

Menurut Lastiansah (2012), *user interface* adalah cara program dan pengguna untuk berinteraksi. Istilah *user interface* terkadang digunakan sebagai pengganti istilah *Human Computer Interaction* (HCI) dimana semua aspek dari interaksi pengguna dan komputer. Semua yang terlihat dilayar, membaca dalam dokumentasi dan dimanipulasi dengan *keyboard* (atau *mouse*) juga merupakan bagian dari *user interface*.

User interface memiliki fungsi untuk menghubungkan atau menterjemahkan informasi antara pengguna dengan sistem operasi, sehingga komputer dapat digunakan. Dengan demikian *user interface* bisa juga diartikan sebagai mekanisme *inter-relasi* atau integrasi total dari perangkat keras dan lunak membentuk pengalaman berkomputer. *User interface* dari sisi *software* bisa berbentuk *Graphical User Interface* (GUI) atau *Command Line Interface* (CLI), sedangkan dari sisi *hardware* bisa berbentuk *Apple Desktop Bus* (ADB), USB, dan *fire wire*.

Konsep *user interface* memiliki banyak aspek yang perlu diperhatikan, karena akan mengacu pada beragam aplikasi teknologi seperti *electronic display*, aplikasi *web*, aplikasi *mobile* dan lain-lain.

2.5 *Graphical User Interface* (GUI)

Menurut Lastiansah (2012) *Graphical User Interface* (GUI) adalah tipe antarmuka yang digunakan oleh pengguna untuk berinteraksi dengan sistem

operasi melalui gambar grafik, ikon, dan menggunakan perangkat penunjuk (*pointing device*) seperti *mouse* atau *track ball*.

Sistem operasi memiliki nama tersendiri untuk komponen GUI-nya. Pada Apple Mac OS X, GUI-nya disebut *Aqua*. Microsoft member nama GUI pada Windows XP sebagai *Lunar* dan GUI Windows Vista sebagai *Aero*. Pada Linux, ada dua pengembangan utama *desktop environment*, yang masing-masing menghasilkan produk KDE (*K Desktop Environment*) dan GNOME.

2.6 *User Experience*

(*Creative Business Jakarta, 2013*) *user experience* (UX) merupakan salah satu strategi mendesain produk yang berfokus pada perspektif pengguna. Strategi *User Experience* dibangun dan diimplementasikan bersamaan dengan strategi bisnis dan produk perusahaan untuk melihat produk kita dari perspektif pengguna. Strategi *User Experience* juga dapat digunakan untuk melihat akan seperti apa interaksi pelanggan dengan perusahaan melalui beragam produk yang telah dihasilkan.

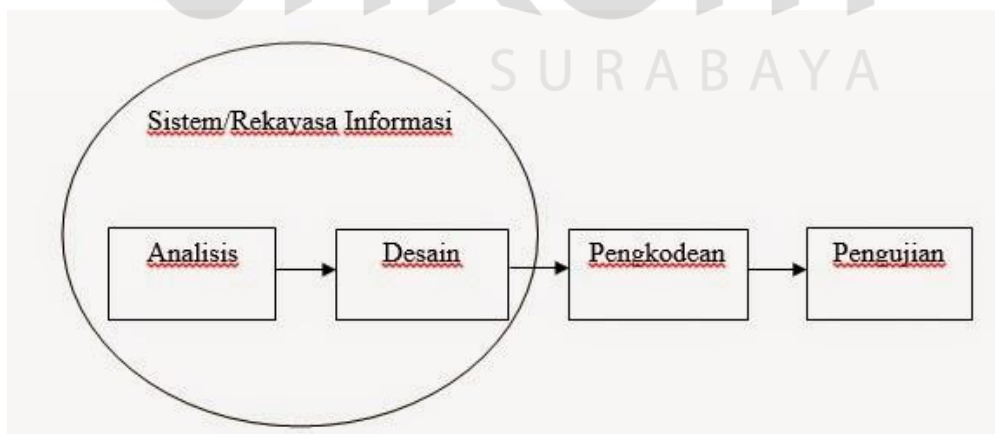
Dalam *user experience* memiliki 4 poin utama, yaitu (1) Mendefinisikan *value* yang dapat diberikan ke pengguna dan mengeksplorasi produk apakah dapat mencapai tujuan bisnis; (2) spesifikasi tujuan, mengidentifikasi setiap kesempatan yang dapat meningkatkan kualitas produk dan mengeksplorasi setiap fase interaksi agar dapat mengidentifikasi sesuai dengan komponen yang sudah ditentukan; (3) merencanakan pengembangan produk dan peningkatannya secara terus-menerus; (4) mengidentifikasi kesuksesan produk dan metode yang digunakan untuk melakukan validasi keberhasilan produk tersebut.

2.7 System Development Life Cycle (SDLC)

Menurut Jogiyanto (1991) metode pengembangan sistem merupakan metode yang digunakan untuk mengembangkan suatu sistem informasi, yaitu suatu proses standar yang diikuti untuk melaksanakan seluruh langkah yang diperlukan untuk menganalisa, merancang, mengimplementasikan, dan memelihara sistem informasi. Daur hidup dari pengembangan sistem ini disebut dengan daur hidup pengembangan sistem (SDLC = *System Development Life Cycle*).

2.8 SDLC Metode Pengembangan Waterfall

Metode pengembangan *system waterfall*. Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linier*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau urut dimulai dari analisis, desain, pengkodean, pengujian dan tahap *support* (Rosa dan Shalahuddin, 2011). Berikut adalah gambar model air terjun :



Gambar 2.5 Ilustrasi Model *Waterfall*

1. Analisis

Analisis sistem dilakukan untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikan. Pada tahap ini dilakukan dalam analisa sistem antara lain :

a. Analisa Identifikasi Masalah

Identifikasi masalah merupakan langkah awal dari analisis sistem. Dalam tahap ini didefinisikan masalah yang harus dipecahkan.

b. Analisa Kebutuhan

Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. (Rosa dan Shalahuddin, 2011).

c. Analisa Kelayakan Sistem

Studi kelayakan digunakan untuk menentukan kemungkinan keberhasilan solusi yang diusulkan. Tahapan ini berguna untuk memastikan bahwa solusi yang diusulkan tersebut benar-benar dapat dicapai (Kadir, 2003).

Dalam tahap ini akan dilakukan analisa kelayakan sebagai berikut :

1) Kelayakan Teknis

Studi mengenai fungsi, kinerja dan batasan yang dapat mempengaruhi kemampuan untuk mencapai sebuah sistem yang dapat diterima (Pressman, 1997).

2) Kelayakan Operasional

Analisa kebutuhan utama yang diperlukan dari suatu sistem baru yang akan dijalankan agar sistem tersebut dapat berjalan sesuai dengan prosedur yang ada. Studi kelayakan operasional

menggunakan kerangka kerja PIECES (*Performance, Information, Economic, Control, Efficiency, Services*).

3) Kelayakan Ekonomi

Evaluasi pengembangan dibobot dengan pemasukan utama atau keuntungan yang didapat dari sistem atau produk yang dikembangkan.

Dalam kelayakan ekonomi ini akan digunakan metode kuantitatif untuk perhitungan biaya.

a) Metode Periode Pengembalian (*Payback Period*)

Metode ini menilai proyek investasi dengan dasar lamanya investasi tersebut dapat tertutup dengan aliran-aliran kas masuk (Jogiyanto, 2005).

$$\text{Payback period} = \text{ x 12 bulan}$$

b) Metode Pengembalian Investasi (*Return Of Investment*)

Metode *return of invesment* digunakan untuk mengukur prosentase manfaat yang dihasilkan oleh proyek dibandingkan dengan biaya yang dikeluarkan (Jogiyanto, 2005). *Return of Investment* dari suatu proyek dapat dihitung dengan rumus :

$$\text{ROI} =$$

Jika $\text{ROI} > 0$, maka dapat diterima.

Jika $\text{ROI} < 0$, maka tidak dapat diterima.

c) Metode Nilai Sekarang Bersih (*Net Present Value*)

Metode nilai sekarang bersih (*net present value*) merupakan metode yang memperhatikan nilai waktu dari uang. metode ini menggunakan suku bunga diskonto yang akan mempengaruhi *proceed* atau arus dari uangnya (Jogiyanto, 2005). Besarnya NPV bila dinyatakan dalam rumus adalah :

$$NPV = -\text{NilaiProyek} + \dots$$

Keterangan :

NPV : *Net Present Value*.

i : Tingkat bunga diskon diperhitungkan.

n : Umur proyek investasi.

Jika $NPV > 0$, maka investasi dapat diterima.

Jika $NPV < 0$, maka investasi tidak dapat diterima.

d) Metode Tingkat Pengembalian Internal (*Internal Rate of Return*)

Internal Rate of Return merupakan metode yang memperhatikan nilai waktu dari uang. Pada metode NPV, tingkat bunga yang diinginkan telah ditetapkan sebelumnya, sedangkan pada metode IRR ini justru tingkat bunga tersebut yang akan dihitung (Jogiyanto, 2005). Rumus perhitungannya adalah sebagai berikut:

NPV1: NPV positif dengan tingkat bunga i_1 .

NPV2: NPV negatif dengan tingkat bunga i_2 .

Keterangan :

i1: Tingkat bunga pertama yang menyebabkan NPV positif.

i2: Tingkat bunga kedua yang menyebabkan NPV negatif.



4) Kelayakan Hukum

Pertimbangan mengenai pelanggaran, kekasaran atau liabilitas yang dihasilkan dari pengembangan sistem.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya (Rosa dan Shalahuddin, 2011).

Pada tahap desain dilakukan perancangan antara lain:

a. Perancangan Proses

Rancangan proses berupa diagram alir data atau *data flow diagram* (DFD). Perancangan Proses yang digunakan adalah model logika yang mana model logika lebih menjelaskan kepada *user* bagaimana nantinya fungsi-fungsi sistem informasi secara logika akan bekerja.

b. Perancangan Basis Data

Perancangan yang dimaksud dalam tahap ini adalah menentukan dan menunjukkan hubungan antara *entity* dan relasinya (*Entity Relationship Diagram*).

c. Perancangan Tabel

Rancangan berupa tabel-tabel yang digunakan dalam pembuatan sistem.

d. Perancangan Antarmuka

Desain aplikasi adalah tahap yang harus dilakukan sebelum mulai membuat aplikasi. Konsep rancangan dalam mendesain halaman aplikasi adalah tampilan pada halaman aplikasi yang akan dipergunakan oleh pengguna.

3. Pengodean

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain (Rosa dan Shalahuddin, 2011).

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai yang diinginkan (Rosa dan Shalahuddin, 2011). Dalam penelitian ini pengujian sistem akan menggunakan pengujian *Black-Box*.

Pengujian *Black-box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian metode ini memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program (Pressman, 1997).

2.9 Website

Website atau situs adalah kumpulan dari halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar (diam/bergerak), animasi, suara, atau gabungan dari keseluruhan. Baik yang memiliki sifat statis atau dinamis yang

akan membentuk suatu rangkaian yang saling berkaitan, yang masing-masing dihubungkan dengan jaringan-jaringan halaman. Beberapa hal yang harus dipersiapkan untuk membangun *website* adalah nama *domain* dan *website hosting*. (Hidayat, 2010)

2.10 *Javascript*

Menurut Sunyoto (2007) “Javascript adalah bahasa *scripting* yang populer di internet dan dapat bekerja di sebagian besar *browser* populer seperti internet explorer, Mozilla Firefox, Netscape, dan Opera”. Kode Javascript dapat disisipkan dalam halaman *web* menggunakan *tag script*. Beberapa hal tentang Javascript sebagai berikut:

1. Javascript didesain untuk menambah interaktif suatu *web*
2. Javascript merupakan sebuah Bahasa *scripting*.
3. Bahasa *scripting* merupakan bahasa pemrograman yang ringan.
4. Javascript berisi kode yang dijalankan di komputer (*web browser*).
5. Javascript biasanya disisipkan (*embedded*) dalam halaman HTML.
6. Javascript adalah bahasa *interprete r*(yang berarti skrip dieksekusi tanpa proses kompilasi).

2.11 *Hyper Text Markup Language (HTML)*

(Anhar, 2010) HTML merupakan bahasa pemrograman *web* yang memiliki sintak atau aturan tertentu dalam menuliskan *script* atau kode-kode, sehingga *browser* dapat menampilkan informasi dengan membaca kode-kode HTML. Pengertian dari HTML sendiri adalah sekumpulan simbol-simbol atau *tag-tag* yang dituliskan dalam sebuah *file* yang digunakan untuk menampilkan

halaman pada *web browser*. *Tag-tag* HTML selalu diawali dengan simbol `<x>` dan diakhiri dengan simbol `</x>` dimana huruf x merupakan *tag* HTML yang akan digunakan.

2.12 *Cascading Style Sheet (CSS)*

(Sya'ban, 2010) CSS adalah suatu bahasa yang dikhususkan untuk mengatur gaya atau *layout* sebuah halaman *web*. CSS digunakan oleh pembuat halaman *web* dan juga pengakses halaman *web*, untuk mendefinisikan warna, huruf *layout*, dan aspek-aspek presentasi dokumen lainnya. CSS memang didesain untuk memisahkan antara isi dokumen (yang ditulis menggunakan HTML atau bahasa lainnya yang sejenis) dengan berbentuk presentasi dokumen (ditulis dalam CSS). Pemisahan ini memberikan keuntungan akan adanya peningkatan dalam aksesibilitas isi, menyediakan fleksibilitas lebih dan pengendalian terhadap spesifikasi karakteristik bentuk presentasi, serta mereduksi kompleksitas dan perulangan-perulangan pada struktur isi.

2.13 *Hypertext Preprocessor*

Menurut Oktavian (2010), PHP adalah akronim dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode-kode yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode HTML.

Menurut Kustiyahningsih (2011), PHP adalah skrip bersifat *server-side* yang ditambahkan ke dalam *Hyper Text Markup Language (HTML)*. Sifat *server-side* berarti pengerjaan skrip dilakukan di *server*, yang kemudian hasilnya dikirim kembali ke *browser*. Cara penulisan skrip PHP dapat dilakukan dengan 2 teknik,

yaitu *Embedded Script* dan *Non Embedded Script*. Seiring dengan perkembangan teknologi maka lahirnya PHP sebagai bahasa pemrograman *open source* yang digunakan secara luas terutama untuk pengembangan *web* dan dapat disimpan dalam bentuk HTML. Sehingga *web* tidak hanya memberikan informasi tetapi terjalin interaksi dan menjadikan *web* bersifat dinamis dan diintegrasikan dengan *web server* Apache, PWS, dan IIS.

Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah skrip PERL yang dapat mengamati siapa yang melihat-lihat daftar riwayat hidupnya pada tahun 1994. Pada tahun 1995, Rasmus menciptakan PHP/FI versi 2, dimana versi tersebut dapat menempelkan kode terstruktur dalam *tag* HTML dan juga PHP dapat digunakan untuk berkomunikasi dengan *database*.

PHP biasanya dipergunakan untuk pemrograman berbasis *web* yang tidak hanya menampilkan halaman secara statis, namun menampilkan *website* berbentuk dinamis dimana data diambil dari dalam *database*. PHP memiliki kelebihan yaitu PHP bersifat sederhana dan memiliki kemampuan untuk menghasilkan berbagai aplikasi *web*, selain itu PHP juga bersifat *multiplatform* (Windows, Linux, dan Mac).

2.14 Web Server

Menurut O'brien (2011) lebih spesifik menyatakan bahwa, "*Server* adalah komputer yang mendukung aplikasi dan telekomunikasi dalam jaringan, serta pembagian peralatan *software*, dan *database* di antara berbagai terminal kerja dalam jaringan".

Menurut Nugroho (2004) mengatakan bahwa yang dimaksud dengan *web server* adalah sebuah bentuk dari *server* yang khusus digunakan untuk menyimpan halaman *website* atau *home page*. Sebuah komputer dapat dikatakan sebagai *web server* apabila komputer tersebut memiliki suatu program *server* yang disebut PWS atau *Personal Web Service*. PWS ini kemudian nantinya difungsikan agar halaman *web* yang ada di dalam sebuah komputer *server* dapat dipanggil oleh komputer klien.

2.15 POAC (*Planning Organizing Actuating Controlling*)

Menurut G. R. Terry dalam Hasibuan (2001) Manajemen mempunyai fungsi-fungsi diantaranya sebagai perencanaan (*planning*), pengorganisasian (*organizing*), pengarahan (*actuating*), pengawasan/pengendalian (*controlling*) atau yang lebih dikenal dengan singkatan POAC.

a. Perencanaan (*Planning*)

Dari semua fungsi di atas, perencanaan adalah fungsi dasar (fundamental) manajemen. Manajemen pasti berangkat dari sebuah perencanaan karena fungsi-fungsi yang lain akan efektif dan efisien apabila terlebih dahulu direncanakan dengan baik.

Menurut Harold Koontz dan Cyril O'Donnel dalam Hasibuan (2001) perencanaan adalah fungsi seorang manajer yang berhubungan dengan memilih tujuan-tujuan, kebijakan-kebijakan, prosedur-prosedur, program-program dari alternatif-alternatif yang ada. Menurut G. R Terry dalam Hasibuan (2001) perencanaan adalah memilih, menghubungkan fakta dan membuat serta menggunakan asumsi-asumsi mengenai masa datang dengan jalan

menggambarkan dan merumuskan kegiatan-kegiatan yang diperlukan untuk mencapai hasil yang diinginkan.

Menurut Hasibuan (2001) perencanaan adalah suatu proses untuk menentukan rencana, sehingga rencana merupakan produk dari perencanaan. Dalam suatu rencana harus ditetapkan tujuan yang ingin dicapai dan pedoman-pedoman untuk mencapai tujuan itu. Jadi, setiap rencana harus mengandung dua unsur yaitu tujuan dan pedoman.

b. Pengorganisasian (*Organizing*)

Pengorganisasian menurut Hasibuan (2001) adalah suatu proses penentuan, pengelompokan, dan pengaturan bermacam-macam aktivitas yang diperlukan untuk mencapai tujuan, menempatkan orang-orang pada setiap aktivitas ini, menyediakan alat-alat yang diperlukan, menetapkan wewenang yang secara relatif didelegasikan kepada setiap individu yang akan melakukan aktivitas-aktivitas tersebut.

Organisasi atau pengorganisasian dapat pula dirumuskan sebagai keseluruhan aktivitas manajemen dalam mengelompokkan orang-orang serta penetapan tugas, fungsi, wewenang serta tanggung jawab masing-masing dengan tujuan tercapainya aktivitas-aktivitas yang berdaya guna dan berhasil guna dalam mencapai tujuan yang telah ditetapkan terlebih dahulu.

c. Pengarahan (*Actuating*)

Menurut G.R Terry dalam Hasibuan (2001) pengarahannya adalah membuat semua anggota kelompok agar mau bekerja sama dan bekerja secara ikhlas serta bergairah untuk mencapai tujuan sesuai dengan perencanaan dan usaha-usaha pengorganisasian.



d. **Pengawasan/Pengendalian (*Controlling*)**

Menurut Earl P. Strong dalam Hasibuan (2001) pengawasan/pengendalian adalah proses pengaturan berbagai faktor dalam suatu perusahaan, agar pelaksanaan sesuai dengan ketepatan-ketepatan dalam rencana.

Menurut G. R. Terry dalam Hasibuan (2001) pengawasan/pengendalian adalah sebagai proses penentuan, apa yang harus dicapai yaitu standar, apa yang sedang dilakukan yaitu pelaksanaan, menilai pelaksanaan sesuai dengan rencana yaitu selaras dengan standar. Menurut Hasibuan (2001) Pengawasan/pengendalian teknis (*technical control*) adalah pengendalian yang ditujukan kepada hal-hal yang bersifat fisik, yang berhubungan dengan tindakan dan teknis pelaksanaan.

