

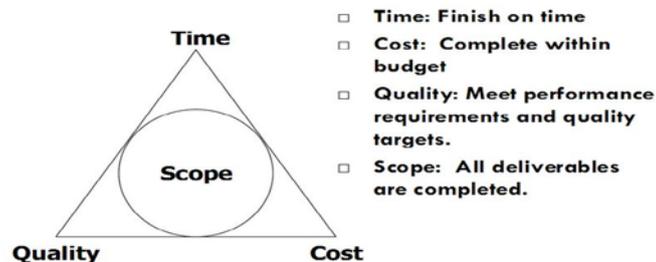
## BAB II

### LANDASAN TEORI

#### 2.1 Manajemen Proyek

Proyek adalah suatu tugas yang perlu didefinisikan dan terarah ke suatu sasaran yang dituturkan secara konkrit serta harus diselesaikan dalam kurun waktu tertentu dengan menggunakan tenaga manusia terbatas dan dengan alat-alat terbatas pula, dan demikian rumit atau barunya, sehingga diperlukan suatu jenis pimpinan dan bentuk kerjasama yang berlainan dari pada yang biasa digunakan (Djojowiriono, Soegeng. 2002).

Menurut Ervianto, WULFRAMI. 2002, tiap proyek memiliki tujuan khusus dan di dalam proses pencapaian tujuan tersebut ada batasan yang harus dipenuhi yaitu besarnya biaya (anggaran) yang dialokasikan, jadwal, serta mutu yang harus dipenuhi. Ketiga hal tersebut merupakan parameter penting bagi penyelenggaraan proyek yang sering diasosiasikan sebagai sasaran proyek. Ketiga batasan di atas disebut tiga kendala (*triple constrain*).



Gambar 2.1 Triple Konstrain Sumber: Djojowiriono, Soegeng. 2002

## 2.2 Jaringan Kerja

Metode jaringan kerja diperkenalkan menjelang akhir dekade 1950-an, oleh suatu tim *engineer* dan ahli matematika dari perusahaan *Du-Pont* bekerja sama dengan *Rand Corporation*, dalam usaha mengembangkan suatu sistem kontrol manajemen. Sistem ini dimaksudkan untuk merencanakan dan mengendalikan sejumlah besar kegiatan yang memiliki hubungan ketergantungan yang kompleks dalam masalah *desain-engineering*, konstruksi, dan pemeliharaan (Husen, A. 2009).

## 2.3 Critical Path Method (CPM)

Pada metode CPM dikenal adanya jalur kritis, yaitu jalur yang memiliki rangkaian komponen-komponen kegiatan dengan total jumlah waktu terlama dan menunjukkan kurun waktu penyelesaian proyek paling cepat. Jadi, jalur kritis terdiri dari rangkaian kegiatan kritis, dimulai dari kegiatan pertama sampai pada kegiatan terakhir proyek (Husen, A. 2009).

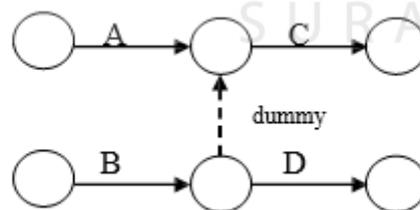
Metode CPM akan menjelaskan beberapa sistematika penyusunan jaringan kerja dan istilah - istilah, *float*, dan jalur kritis yang diperlukan di dalam perhitungan CPM nantinya. Jalur kritis sangat penting bagi pelaksanaan proyek, karena pada jalur /lintasan ini terletak kegiatan-kegiatan yang bila pelaksanaannya terlambat akan menyebabkan keterlambatan pada proyek secara keseluruhan.

Dalam menyusun jaringan kerja dimulai dengan cara mengkaji serta mengidentifikasi lingkup proyek, lalu menguraikanya menjadi beberapa komponen. langkah kedua menyusun jaringan yang sesuai dengan logika ketergantungan yang dilanjutkan dengan memberikan perkiraan waktu masing-

masing kegiatan. Setelah itu mengidentifikasi jalur kritis waktu untuk penyelesaian proyek.

Beberapa langkah yang perlu dilakukan dalam menggambar jaringan kerja adalah sebagai berikut:

1. Buat anak panah dengan garis penuh dari kiri ke kanan dan garis putus-putus untuk *dummy*. *Dummy* adalah kegiatan fiktif yang tidak memerlukan waktu kegiatan dan untuk menunjukkan hubungan ketergantungan kegiatan C bisa dimulai bila kegiatan A, B selesai sedangkan kegiatan D dimulai setelah kegiatan B selesai.
2. Usahakan ada bagian untuk tempat keterangan kegiatan dan kurun waktu.
3. Hindari garis yang saling menyilang. Panjang anak panah tidak ada kaitannya dengan lamanya kurun waktu.
4. Peristiwa atau kejadian dilukis sebagai lingkaran dengan nomor yang bersangkutan, jika mungkin ditulis didalamnya.
5. Nomor peristiwa disebelah kanan lebih besar dari sebelah kiri.



Gambar 2. 2 Kegiatan Dummy

Setelah mengetahui langkah-langkah yang harus dilakukan, harus diketahui pula beberapa istilah yang dipakai dalam penyusunan jaringan kerja yaitu sebagai berikut:

1.  $EET_i$  (*Earliest Event Time*) =  $ES$  (*Earliest Start*) =  $EST$  (*Earliest Start Time*), yaitu waktu mulai paling cepat dari *event* I atau waktu mulai paling awal suatu kegiatan.
2.  $LET_i$  (*Latest Event Time*) =  $LS$  (*Latest Start*) =  $LST$  (*Latest Start Time*), yaitu waktu mulai paling lambat dari *event* I atau waktu mulai paling akhir suatu kegiatan.
3.  $EET_j$  (*Earliest Event Time*) =  $EF$  (*Earliest Finish*) =  $EFT$  (*Earliest Finish Time*), yaitu waktu mulai paling cepat dari *event* j atau waktu selesai paling awal suatu kegiatan.
4.  $LET_j$  (*Latest Event Time*) =  $LF$  (*Latest Finish*) =  $LFT$  (*Latest Finish Time*), yaitu waktu mulai paling lambat dari *event* j atau waktu selesai paling akhir suatu kegiatan.
5.  $D(i-j)$ , yaitu kurun waktu untuk melaksanakan kegiatan antara *event* I dan *event* j.
6. I dan j adalah *Number Event*.

Dalam metode CPM digunakan hitungan maju dan hitungan mundur. Hitungan maju dimaksudkan untuk mengetahui waktu paling awal untuk memulai dan mengakhiri masing-masing kegiatan tanpa penundaan waktu untuk itu diberikan beberapa rumus penyelesaian. Kecuali kegiatan awal, kegiatan baru dapat di mulai bila kegiatan yang mendahului telah selesai. Dimana:

$EF$  : (*Earliest Finish*) waktu selesai paling awal

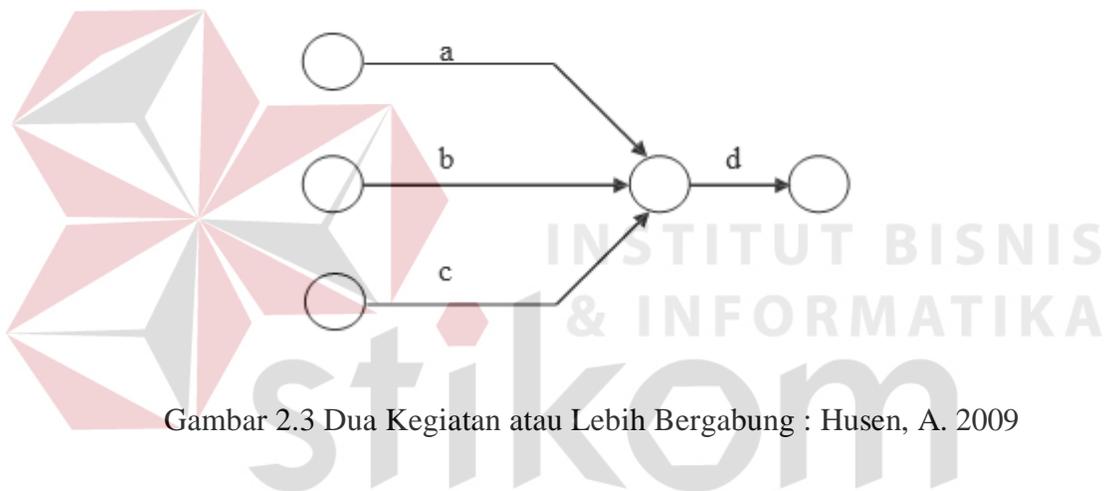
$ES$  : (*Earliest Start*) waktu mulai paling awal

D : kurun waktu kegiatan bersangkutan

I : kegiatan awal atau sebelumnya

J : kegiatan selanjutnya setelah i

Bila kegiatan memiliki dua atau lebih pendahulu yang bergabung, maka waktu mulai paling awal sama dengan waktu selesai paling awal yang terbesar kegiatan pendahulu. Sebagai contoh diberikan jaringan kerja dan alokasi waktu sebagai berikut:



Gambar 2.3 Dua Kegiatan atau Lebih Bergabung : Husen, A. 2009

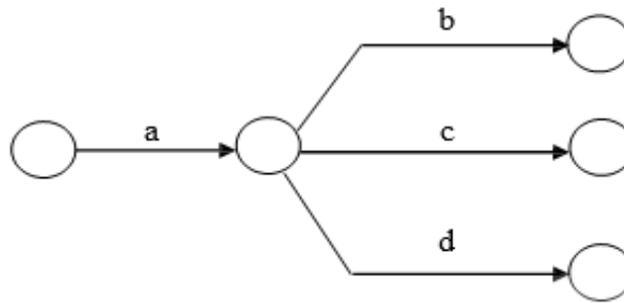
Hitungan mundur dimaksudkan untuk mengetahui waktu paling akhir untuk dapat memulai dan mengakhiri masing-masing kegiatan tanpa menunda kurun waktu penyelesaian proyek secara keseluruhan dari hasil hitungan mundur. Beberapa rumus yang dipakai dalam hitungan mundur yaitu. Dimana

LS : (*Latest Start*) waktu mulai paling akhir suatu kegiatan.

LF : (*Latest Finish*) waktu selesai paling akhir.

D : kurun waktu kegiatan bersangkutan.

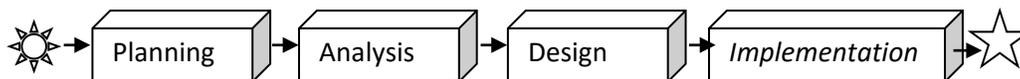
Bila kegiatan pecah menjadi dua kegiatan atau lebih maka waktu selesai paling akhir (LF) kegiatan sama dengan waktu mulai paling akhir (LS) kegiatan berikutnya yang terkecil.



Gambar 2.4 Dua Kegiatan atau Lebih Memecah : Husen, A. 2009

## 2.4 System Development Life Cycle (SDLC)

Menurut Alan Dennis, (2013), dalam membangun sistem dengan menggunakan SDLC memiliki empat fase dasar yaitu perencanaan, analisis, desain, dan implementasi. Setiap fase itu sendiri terdiri atas serangkaian langkah dengan mengandalkan teknik sehingga menghasilkan produk.



Gambar 2.5 Bagan SDLC

a. Perencanaan

Fase perencanaan ini adalah proses dasar dalam memahami mengapa sistem informasi harus dibuat dan menjelaskan bagaimana tim proyek akan melakukannya.

b. Analisis

Fase analisis ini menjelaskan pertanyaan tentang siapa yang akan menggunakan sistem, apa yang akan dilakukan sistem, dimana dan kapan sistem tersebut digunakan. Di dalam fase ini tim proyek melakukan investigasi sistem saat ini, mengidentifikasi adanya perbaikan, dan mengembangkan konsep untuk sistem yang baru.

c. Desain

Fase desain ini menentukan bagaimana sistem akan beroperasi dengan perangkat keras, perangkat lunak, dan infrastruktur jaringan yang ada. Fase ini juga termasuk menentukan tampilan antarmuka, formulir, laporan yang akan digunakan, spesifikasi program, basis data, dan bahan-bahan yang dibutuhkan.

d. Implementasi

Fase akhir di dalam SDLC adalah fase implementasi, dimana sistem ini sudah benar-benar dibangun. Ini adalah fase yang biasanya paling diperhatikan, karena ini adalah bagian yang terpanjang dan termahal di dalam proses pengembangan.

## 2.5 Pengertian Bagan Alir

Menurut Krismiaji (2010), Bagan alir merupakan teknik analitis yang digunakan untuk menjelaskan aspek-aspek sistem informasi secara jelas, tepat dan logis. Bagan alir menggunakan serangkaian simbol standar untuk menguraikan prosedur pengolahan transaksi yang digunakan oleh sebuah perusahaan, sekaligus menguraikan aliran data dalam sebuah sistem.

Terdapat beberapa jenis bagan alir yang biasa digunakan, yaitu sebagai berikut:

### a. Bagan Alir Sistem (*System Flowchart*)

Bagan alir sistem (*system flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan dari prosedur-prosedur yang ada di dalam sistem. Bagan alir sistem menunjukkan apa yang dikerjakan dalam sistem. Bagan alir sistem digambarkan dengan menggunakan simbol-simbol yang telah ditentukan.

### b. Bagan Alir Dokumen (*Document Flowchart*)

Bagan alir dokumen (*document flowchart*) atau disebut dengan bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. Bagan alir dokumen ini menggunakan simbol-simbol yang sama dengan yang digunakan didalam bagan alir sistem.

### c. Bagan Alir Skematik (*Schematic Flowchart*)

Bagan alir skematik (*schematic flowchart*) merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu menggambarkan prosedur dalam sistem. Perbedaannya adalah bagan alir skematik selain menggunakan simbol-simbol bagan alir sistem, juga menggunakan gambar-gambar komputer dan peralatan lainnya yang digunakan. Maksud penggunaan gambar-gambar ini adalah untuk memudahkan dalam menjelaskan simbol-simbol bagan alir kepada orang yang masih awam.

d. Bagan Alir Program (*Program Flowchart*)

Bagan alir program (*program flowchart*) terdiri dari dua macam, yaitu bagan alir logika program (*program logic flowchart*) dan bagan alir program komputer terinci (*detailed computer program flowchart*). Bagan alir logika program digunakan untuk menggambarkan tiap-tiap langkah di dalam program komputer secara logika. Bagan alir logika program ini dipersiapkan oleh analis sistem.

e. Bagan Alir Proses (*Process Flowchart*)

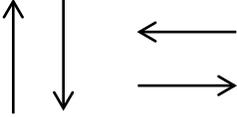
Bagan alir proses (*process flowchart*) merupakan bagan alir yang banyak digunakan di teknik industri. Berguna bagi analis sistem untuk menggambarkan proses dalam suatu prosedur.

Berdasarkan penjelasan di atas, dapat ditarik kesimpulan bahwa bagan alir (*flowchart*) adalah suatu gambaran umum tentang sistem yang berjalan dan berfungsi sebagai alat bantu komunikasi serta untuk mendokumentasikan dan menyajikan kegiatan mulai dari manual, semi manual maupun komputerisasi.

## 2.6 Pengertian Data Flow Diagram

Menurut Hartono (2005), *Data Flow Diagram* (DFD) adalah diagram yang menggunakan notasi simbol untuk menggambarkan arus data sistem. Kita dapat menggunakan DFD untuk dua hal utama, yaitu untuk membuat dokumentasi dari sistem informasi yang ada, atau untuk menyusun dokumentasi untuk sistem informasi yang baru. Empat simbol yang digunakan yaitu:

Tabel 2.1 Simbol *Data Flow Diagram*

Simbol	Keterangan
	<i>External Entity</i> , merupakan kesatuan di lingkungan luars sistem yang bisa berupa orang, organisasi atau sistem lain.
	<i>Process</i> , merupakan proses seperti perhitungan aritmatika penulisan suatu formula atau pembuatan laporan
	<i>Data Store</i> (Simpan Data), dapat berupa suatu file atau database pada sistem komputer atau catatan manual
	<i>Data Flow</i> (Arus Data), arus data ini mengalir di antara proses, simpan data dan kesatuan luar

Menurut Hartono (2005), ada beberapa simbol digunakan pada DFD untuk mewakili:

a. Kesatuan Luar (*External Entity*)

Kesatuan luar (*external entity*) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lain yang berada pada lingkungan luarnya yang memberikan input atau menerima output dari sistem.

b. Proses (*Process*)

Proses (*process*) menunjukkan pada bagian yang mengubah *input* menjadi *output*, yaitu menunjukkan bagaimana satu atau lebih *input* diubah menjadi beberapa *output*. Setiap proses mempunyai nama, nama dari proses ini menunjukkan apa yang dikerjakan proses.

c. Simpanan Data (*Data Store*)

*Data Store* merupakan simpanan dari data yang dapat berupa suatu *file* atau *database* pada sistem komputer.

d. Arus Data (*Data Flow*)

Arus Data (*Data Flow*) di DFD diberi simbol suatu panah. Arus data ini mengalir di antara proses, simpan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.