

BAB III

LANDASAN TEORI

3.1 Penjadwalan

Penjadwalan adalah kegiatan pengalokasian sumber-sumber atau mesin-mesin yang ada untuk menjalankan sekumpulan tugas dalam jangka waktu tertentu (Baker & Kenneth, 1974). Penjadwalan produk adalah suatu kegiatan memasukkan sejumlah produk yang telah direncanakan ke dalam proses penggerjaannya (Biegel, 1992). Dari dua definisi tersebut, maka dapat ditarik sebuah definisi bahwa penjadwalan adalah suatu kegiatan perencanaan berupa pengalokasian sumber daya baik mesin maupun tenaga kerja untuk menjalankan sekumpulan tugas sesuai prosesnya dalam jangka waktu tertentu.

3.2 Tabu Search

Tabu search merupakan sebuah metode optimasi matematis yang termasuk ke dalam kelas *local search*. *Tabu search* memperbaiki performansi *local search* dengan memanfaatkan penggunaan struktur memori. Sebagian solusi yang pernah dibangkitkan ditandai sebagai “tabu”/”taboo” yang berarti sesuatu yang terlarang, sehingga algoritma tabu tidak akan mengunjungi solusi tersebut secara berulang-ulang.

Tabu search diperkenalkan pertama kali oleh Glover pada tahun 1970-an. Ide dasar *tabu search* juga diperkenalkan oleh Hansen. Banyak percobaan yang menunjukkan bahwa *tabu search* saat ini telah menjadi sebuah teknik optimasi yang dapat diadu dengan hampir semua teknik optimasi yang telah dikenal.

Ada tiga komponen penting dalam *tabu search*, yaitu:

1. Penggunaan struktur memori yang fleksibel, yang memungkinkan evaluasi kriteria dan informasi historis bisa dieksplorasi lebih baik dibandingkan dengan struktur memori yang statis.
2. Penggunaan mekanisme control yang didasarkan interaksi antara kondisi yang membatasi dan kondisi yang mendukung proses pencarian.
3. Penggabungan fungsi memori yang memiliki rentang waktu yang berbeda, dari memori jangka pendek sampai dengan memori jangka panjang, untuk menerapkan strategi indensifikasi dan diversifikasi.

Algoritma *tabu search* bisa menerima solusi yang lebih buruk daripada solusi saat ini. Untuk menjaga agar solusi terbaik tidak hilang, *tabu search* menyimpan solusi terbaik dan terus mencari berdasarkan solusi terakhir. Selain itu, metode ini mengingat sebagian solusi yang pernah ditemui dan melarang untuk menggunakan solusi yang telah ditelusuri untuk menghindari pengulangan yang sia-sia. Hal ini yang membuat *tabu search* menjadi lebih efisien dalam hal usaha dan waktu.

Dalam praktiknya *tabu search* menggunakan struktur memori yang disebut *tabu list* untuk menyimpan atribut dari sebagian *move* (langkah transisi dari satu solusi ke solusi yang lain) yang telah diterapkan pada iterasi-iterasi sebelumnya. *Tabu list* digunakan untuk menolak solusi-solusi yang memenuhi atribut tertentu agar proses pencarian tidak berulang-ulang pada daerah solusi yang sama dan untuk menuntun proses pencarian menelusuri solusi-solusi yang belum pernah dikunjungi.

Untuk efisiensi memori dan waktu proses, *tabu list* hanya menyimpan langkah transisi (*move*) yang merupakan kebalikan dari langkah yang telah digunakan pada iterasi-iterasi sebelumnya. Dengan kata lain, *tabu list* hanya berisi langkah-langkah yang mengembalikan solusi yang baru ke solusi yang lama. *Tabu list* menggunakan prinsip *first in first out* (FIFO) dengan panjang *list* tertentu yang dibatasi oleh user. Dengan menggunakan *tabu list*, *tabu search* dapat menerima solusi yang tidak memberikan peningkatan kualitas, sehingga *tabu search* bisa keluar dari *local optimum*. Tetapi, terdapat suatu pengecualian yang sangat penting pada *tabu list*. Jika terdapat *move* yang sudah berada di dalam *tabu list* (terlarang untuk dipilih) tetapi memberikan solusi yang lebih baik dibandingkan semua solusi terbaik yang pernah dibangkitkan maka *move* tersebut bisa diterima dan *move* tersebut harus dikeluarkan dari *tabu list* (dibebaskan dari larangan). Hal ini merupakan prioritas khususnya pada *tabu list* yang disebut kriteria aspirasi atau kondisi aspirasi (*aspiration conditions*).

3.3 *System Development Life Cycle*

SDLC merupakan tahapan-tahapan pekerjaan yang dilakukan oleh analis sistem dan programmer dalam membangun sistem informasi. Ada beberapa tahapan yang digunakan dalam SDLC

1. *Requirement Specification*

Disebut juga sebagai tahap spesifikasi kebutuhan pengguna, dimana desainer sistem mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan mana yang harus dipenuhi oleh program yg akan dibangun.

Pada tahap ini, desainer sistem harus berkomunikasi dengan klien. Desainer sistem atau sistem analis harus melakukan pemeriksaan terhadap kebijakan dan prosedur pengolahan data dan sistem informasi yang berlaku saat ini atau disebut dengan istilah *present system*. Dengan mengetahui sasaran sistem yang sebenarnya, dan memahami bagaimana sistem yang lama bekerja, maka seorang sistem analis dengan mudah bisa membuat sebuah konsep tentang sistem baru yang akan dikerjakan.

2. *Architectural Design*

Pada tahap desain, sistem analis berfokus pada bagaimana sistem dibangun, dengan memperhatikan langkah-langkah berikut:

- Mendefinisikan tujuan sistem, tidak hanya berdasarkan informasi dari user, tetapi juga berupa analisa dari abstraksi dan karakteristik keseluruhan kebutuhan informasi sistem.
- Membangun sebuah model konseptual, berupa gambaran sistem secara keseluruhan yang menggambarkan satuan fungsional sebagai unit sistem.
- Menerapkan kendala-kendala organisasi
- Mendefinisikan aktivitas pemrosesan data
- Menyiapkan proposal sistem desain

3. *Coding* (pengkodean)

Desain program diterjemahkan ke dalam kode-kode dengan menggunakan bahasa pemrograman yang sudah ditentukan. Setelah *coding*, setiap komponen diuji untuk memverifikasi apakah sudah berjalan dengan benar.

4. Integrasi dan *testing*

Dilakukan dengan mengoperasikan program dengan memproses data sehingga kesalahan dapat diketahui seawal mungkin. Pengujian dilakukan dengan teliti, mula-mula perunit sampai berbagai unit secara komprehensif, kemudian dilakukan pengujian tes penerimaan dengan client untuk memastikan sistem yang dibuat memenuhi kebutuhan mereka.

5. *Training* & implementasi

Karena tujuan sistem yang baru adalah untuk mengganti prosedur - prosedur lama, maka pelatihan kepada user yang akan menggunakan sistem merupakan hal penting. Setelah *training* selesai, dilakukan konversi (peralihan) dari sistem lama ke sistem yang baru, mungkin perlu menulis program khusus untuk menukar fail-fail yang ada menjadi fail-fail yang baru atau membuat fail-fail dari catatan manual.

Ada beberapa cara konversi ke sistem yang baru:

- Konversi langsung yaitu sistem yang lama secara sekaligus diganti dengan sistem yang baru.
- Konversi pararel dengan cara sistem baru dan lama dijalankan secara bersamaan untuk beberapa waktu, sehingga jika sistem baru mengalami gangguan sistem lama dapat mengkompensasi.
- Konversi bertahap adalah peralihan ke sistem yang baru dilakukan bagian per bagian.
- Konversi pilot studi: mirip konversi bertahap, sistem baru diimplementasikan dibidang tertentu dalam organisasi, setelah berhasil baru diimplementasikan dibidang yang lain.

Akhirnya bila seluruh tahap diatas selesai sistem baru mulai dipasang atau diimplementasikan.

6. Operasi & *maintenance*

Setelah pemasangan dan organisasi disesuaikan dengan perubahan-perubahan yang ditimbulkan oleh sistem baru, maka tahap operasional dimulai. Pada tahap ini perlu dilakukan pemeliharaan terhadap sistem serta peningkatan mutu sistem agar sesuai dengan kebutuhan organisasi. Sehingga perlu adanya perubahan dan peningkatan terhadap sistem, tidak masuk akal untuk mengatakan bahwa sebuah sistem informasi berbasis komputer telah selesai, sistem tersebut akan terus berkembang selama daur hidupnya, jika pada kenyataannya ia berhasil.

3.4 Pengertian Sistem

Suatu sistem merupakan cara tertentu dan bersifat representatif untuk melaksanakan suatu atau sekelompok aktivitas yang berupa rangkaian langkah-langkah yang terkoordinasi, dan berulang yang dimaksudkan untuk mencapai suatu tujuan tertentu (Anthony & Govindarajan , 2007).

“A *system* is an asset of interdependent elements that together accomplish specific objectives. A system must have organization, interrelationship, integration, and central objectives”. (Gelinas & Dull, 2008). Berdasarkan kutipan tersebut, maka dapat disimpulkan bahwa sistem merupakan suatu kesatuan yang terbentuk dari komponen-komponen dan unsur yang saling berinteraksi, terkoordinasi dan berhubungan erat satu dengan lainnya untuk mencapai tujuan bersama.

3.5 Pengertian Informasi

Informasi adalah data yang telah diatur dan diproses sehingga menjadi lebih berarti, sehingga dapat dijadikan dasar untuk mengambil keputusan yang tepat (Marshal & Steinbart, 2004).

Kualitas informasi tergantung pada tiga hal, yaitu :

1. Akurat, informasi bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan, tetapi harus mencerminkan tujuan yang jelas.
2. Tepat waktu, informasi yang datang pada penerima tidak terlambat, informasi yang telah usang tidak akan bernilai lagi karena tidak dapat dijadikan landasan dalam pengambilan keputusan.
3. Relevan, informasi tersebut dapat memberikan manfaat bagi pemakainya.

3.6 Pengertian Sistem 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. SIM adalah sistem informasi yang diterapkan bagi kepentingan manajemen, dan secara sederhana manajemen dapat diartikan “*getting things done through people*” (Koontz & O'Donnell, 1995).

Manfaat adanya sistem informasi dalam suatu instansi yaitu:

1. Perangkat keras (*hardware*) dan perangkat lunak (*software*) yang berfungsi sebagai mesin.
2. Manusia (*people*) dan prosedur (*procedures*) yang merupakan manusia dan tata cara menggunakan mesin.
3. Data merupakan jembatan penghubung antara manusia dan mesin agar terjadi suatu proses pengolahan data.

3.7 Analisa dan Perancangan Sistem

Penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

Tahap analisis sistem dilakukan setelah tahap perencanaan sistem (*system planning*) dan sebelum tahap desain sistem (*system design*). Tahap analisis merupakan tahap yang kritis dan sangat penting, karena kesalahan di dalam tahap ini juga akan menyebabkan kesalahan di tahap selanjutnya.

Dalam tahap analisis sistem terdapat langkah-langkah dasar yang harus dilakukan oleh analis sistem sebagai berikut:

1. *Identify*, yaitu mengidentifikasi masalah.
2. *Understand*, yaitu memahami kerja dari sistem yang ada.
3. *Analyze*, yaitu menganalisis sistem.
4. *Report*, yaitu membuat laporan hasil analisis.

Setelah tahap analisis sistem selesai dilakukan, maka analis sistem telah mendapatkan gambaran dengan jelas apa yang harus dikerjakan. Tiba waktunya sekarang bagi analis sistem untuk memikirkan bagaimana membentuk sistem tersebut. Tahap ini disebut dengan desain sistem.

Analisa dan perancangan sistem dipergunakan untuk menganalisis, merancang, dan mengimplementasikan peningkatan-peningkatan fungsi bisnis yang dapat dicapai melalui penggunaan sistem informasi terkomputerisasi (Kendall & Kendall, 2003).

3.8 *Data Flow Diagram*

DFD sering digunakan untuk menggambarkan suatu system yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir. DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur dan dapat mengembangkan arus data di dalam sistem dengan terstruktur dan jelas.

3.8.1 Simbol-Simbol yang digunakan dalam DFD

A. *External Entity atau Boundary*

External entity atau kesatuan luar merupakan kesatuan di lingkungan luar system yang dapat berupa orang, organisasi atau system lainnya yang berada di lingkungan luarnya yang akan memberikan masukan atau menerima keluaran dari sistem. *External entity* disimbolkan dengan notasi kotak.

B. Arus Data

Arus data (*data flow*) di DFD diberi symbol panah. Arus data ini mengalir di antara proses, simpanan data (*data store*) dan kesatuan luar (*external entity*). Arus data ini menunjukkan arus data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem.

C. Proses

Suatu proses adalah kegiatan yang dilakukan oleh orang, mesin, atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk menghasilkan arus data yang akan keluar dari proses. Simbol proses berupa lingkaran atau persegi panjang bersudut tumpul.

D. Simpanan Data

Simpanan data merupakan simpanan dari data yang dapat berupa hal-hal sebagai berikut, sebagai gambaran:

1. Suatu fail atau basis data di system komputer.
2. Suatu arsip atau catatan manual.
3. Suatu kotak tempat data di meja seseorang.
4. Suatu tabel acuan manual.

Simpanan data di DFD disimbolkan dengan sepasang garis horizontal paralel yang tertutup disalah satu ujungnya.

3.8.2 *Context Diagram*

Context Diagram merupakan langkah pertama dalam desain proses. Pada *Context Diagram* dijelaskan sistem apa yang dibuat dan entitas eksternal apa saja yang terlibat. Dalam *Context Diagram* harus ada arus data yang masuk dan arus data yang keluar (Kendall, 2003).

3.8.3 *Data Flow Diagram*

DFD level 0 adalah langkah selanjutnya setelah *Context Diagram*. Pada langkah ini, digambarkan proses-proses yang terjadi dalam sistem informasi. Setelah selesai maka akan di dekomposisi menjadi DFD Level 1 yang merupakan penjelasan dari DFD level 0. Pada proses ini dijelaskan proses apa saja yang dilakukan pada setiap proses yang terdapat di DFD level 0 (Kendall, 2003).

3.8.4 *ER-Model*

ER-Model merupakan penggambaran hubungan antara beberapa entitas yang digunakan untuk merancang basis data yang diperlukan.

3.9 Konsep Dasar Basis Data

Database (basis data) merupakan sekumpulan data yang berisi informasi yang saling berhubungan. Pengertian ini sangat berbeda antara basis data relasional dan non-relasional. Pada basis data non-relasional, sebuah basis data hanya merupakan sebuah fail (Yuswanto & Subari, 2005).

Database adalah suatu susunan/kumpulan data operasional lengkap dari suatu organisasi/perusahaan yang diorganisir atau dikelola dan disimpan secara terintegrasi dengan menggunakan metode tertentu menggunakan komputer

sehingga mampu menyediakan informasi optimal yang diperlukan pemakainya (Marlinda, 2004).

Penyusunan satu basis data digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu redundansi dan inkonsistensi data, kesulitan pengaksesan data, isolasi data untuk standarisasi, *multiple user* (banyak pemakai), masalah keamanan (*security*), masalah integrasi (kesatuan), dan masalah *data independence* (ketidak-tergantungan data).

3.9.1 Sistem Basis Data

Sistem basis data adalah suatu sistem menyusun dan mengelola *record-record* menggunakan komputer untuk menyimpan atau merekam serta memelihara dan operasional lengkap sebuah organisasi/perusahaan sehingga mampu menyediakan informasi optimal yang diperlukan pemakai untuk proses mengambil keputusan (Marlinda, 2004).

Pada sebuah sistem basis data terdapat komponen-komponen utama yaitu perangkat keras (*Hardware*), sistem operasi (*Operating System*), basis data (*database*), sistem (aplikasi atau perangkat lunak) pengelola basis data (DBMS), pemakai (*User*), dan aplikasi (perangkat lunak) lain (bersifat opsional).

Kelebihan Sistem Basis Data:

1. Mengurangi kerakangan data, yaitu data yang sama disimpan dalam berkas data yang berbeda-beda sehingga update dilakukan berulang-ulang.
2. Mencegah ketidak-konsistensi.
3. Keamanan data dapat terjaga, yaitu data dapat dilindungi dari pemakai yang tidak berwenang.
4. Integritas dapat dipertahankan.

5. Data dapat dipergunakan bersama-sama.
6. Menyediakan recovery.
7. Memudahkan penerapan standarisasi.
8. Data bersifat mandiri (data independence).
9. Keterpaduan data terjaga, memelihara keterpaduan data berarti data harus akurat. Hal ini sangat erat hubungannya dengan pengontrolan kerangkapan data dan pemeliharaan keselarasan data.

Kekurangan Sistem Basis Data

1. Diperlukan tempat penyimpanan yang besar.
2. Diperlukan tenaga yang terampil dalam mengolah data.
3. Kerusakan sistem basis data dapat mempengaruhi departemen yang terkait.

3.9.2 *Database Management System*

Database Management System (DBMS) merupakan kumpulan fail yang saling berkaitan dan program untuk pengelolanya. Basis data adalah kumpulan datanya, sedang program pengelolanya berdiri sendiri dalam suatu paket program yang komersial untuk membaca data, menghapus data, dan melaporkan data dalam basis data (Marlinda, 2004).

Bahasa-bahasa yang terdapat dalam DBMS:

1. *Data Definition Language* (DDL)

Pola skema basis data dispesifikasi dengan satu set definisi yang diekspresikan dengan satu bahasa khusus yang disebut DDL. Hasil kompilasi perintah DDL adalah satu set tabel yang disimpan di dalam file khusus yang disebut *data dictionary/directory*.

2. *Data Manipulation Language (DML)*

Bahasa yang memperbolehkan pemakai mengakses atau memanipulasi data sebagai yang diorganisasikan sebelumnya model data yang tepat.

3. *Query*

Pernyataan yang diajukan untuk mengambil informasi. *Query* merupakan bagian DML yang digunakan untuk pengambilan informasi.

Fungsi DBMS

4. *Data Definition*

DBMS harus dapat mengolah *data definition* atau pendefinisian data.

5. *Data Manipulation*

DBMS harus dapat menangani permintaan-permintaan dari pemakai untuk mengakses data.

6. *Data Security and Integrity*

DBMS dapat memeriksa keamanan dan integritas data yang didefinisikan oleh DBA.

7. *Data Recovery and Concurrency*

DBMS harus dapat menangani kegagalan-kegagalan pengaksesan basis data yang dapat disebabkan oleh kesalahan sistem, kerusakan disk, dan sebagainya.

DBMS harus dapat mengontrol pengaksesan data yang konkuren yaitu bila satu data diakses secara bersama-sama oleh lebih dari satu pemakai pada saat yang bersamaan.