

BAB II

LANDASAN TEORI

2.1. Sistem Informasi Perpustakaan

2.1.1. Pengertian Perpustakaan

Perpustakaan berasal dari kata dasar pustaka. Menurut *Kamus Besar Bahasa Indonesia*, pustaka artinya kitab atau buku. Dalam bahasa Inggris dikenal dengan *library*. Istilah ini berasal dari kata *librer* atau *libri*, yang artinya buku. Dari kata latin tersebut terbentuklah istilah *librarius* yang berarti tentang buku.

Dalam buku Wiji Suwarno (2010:31), mengutip pendapat Sulistyio Basuki (1991:3), istilah perpustakaan adalah sebuah ruangan, bagian sebuah gedung, ataupun gedung itu sendiri yang digunakan untuk menyimpan buku dan terbitan lainnya yang biasa disimpan menurut tata susunan tertentu untuk digunakan pembaca, bukan untuk dijual.

2.1.2. Pengertian Sistem

Dalam buku Hanif Al Fatta (2007:3), mengutip pendapat Murdick dan Ross (1993), definisi sistem adalah seperangkat elemen yang digabungkan satu dengan yang lainnya untuk suatu tujuan bersama. Sementara dalam kamus *Webster's Unbringed*, sistem adalah elemen-elemen yang saling berhubungan dan membentuk satu kesatuan atau organisasi. Dengan demikian, secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung satu sama lain.

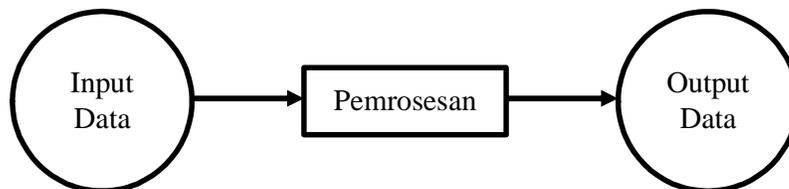
2.1.3. Pengertian Informasi

Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih bermanfaat bagi yang menerima (Andi Kristanto, 2003:6). Jadi ada suatu proses transformasi dari data menjadi

suatu informasi, yaitu input proses output. Informasi yang baik adalah informasi yang berkualitas, yaitu informasi harus akurat (bebas dari kesalahan dan tidak menyesatkan), tepat pada waktunya (*up to date*), dan relevan (mempunyai manfaat bagi penerima informasi).

2.1.4. Pengertian Sistem Informasi

Dalam buku Hanif Al Fatta (2007:9), mengutip pendapat Murdick dan Ross (1993), sistem informasi adalah suatu alat untuk menyajikan informasi dengan cara sedemikian rupa sehingga bermanfaat bagi penerimanya. Tujuannya adalah untuk menyajikan informasi guna pengambilan keputusan pada perencanaan, pemrakarsaan, pengorganisasian, pengendalian kegiatan operasi subsistem suatu perusahaan, dan menyajikan sinergi organisasi pada proses. Dengan demikian, sistem informasi berdasarkan konsep (*input, processing, output* - IPO) dapat dilihat pada Gambar 2.1. berikut :



Gambar 2.1. Konsep Sistem Informasi

Sistem informasi berbasis komputer (CBIS) dalam suatu organisasi terdiri dari komponen-komponen berikut :

1. *Perangkat Keras*, yaitu perangkat keras komponen untuk melengkapi kegiatan memasukkan data, memproses data, dan keluaran data.
2. *Perangkat Lunak*, yaitu program dan instruksi yang diberikan ke computer.
3. *Database*, yaitu kumpulan data dan informasi yang diorganisasikan sedemikian rupa sehingga mudah diakses pengguna sistem informasi.

4. *Telekomunikasi*, yaitu komunikasi yang menghubungkan antara pengguna sistem dengan sistem computer secara bersama-sama ke dalam suatu jaringan kerja yang efektif.
5. *Manusia*, yaitu personal dari sistem informasi, meliputi manajer, analis, programmer, dan operator, serta bertanggung jawab terhadap perawatan sistem.

2.1.5. Pengertian Sistem Informasi Perpustakaan

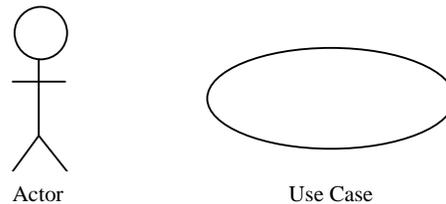
Sistem informasi mencakup sejumlah komponen (manusia, komputer, teknologi informasi, dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan (Abdul Kadir, 2003:10). Dari pernyataan tersebut, maka sistem informasi perpustakaan dapat didefinisikan sebuah sistem terintegrasi, sistem manusia dengan mesin, untuk menyediakan informasi yang mendukung operasi, manajemen, dan fungsi pengambilan keputusan dalam sebuah perpustakaan.

2.2. Analisis dan Perancangan Berorientasi Objek

Pemrograman berorientasi objek adalah suatu cara baru dalam berpikir serta berlogika dalam menghadapi masalah-masalah yang akan dicoba diatasi dengan bantuan komputer. Tidak seperti pendahulunya (pemrograman terstruktur), pemrograman berorientasi objek mencoba melihat pemmasalahan lewat pengamatan dunia nyata dimana setiap objek adalah entitas tunggal yang memiliki kombinasi struktur data dan fungsi tertentu (Adi Nugroho, 2002:1). Dalam analisis dan perancangan metodologi berorientasi objek menggunakan diagram UML (*Unified Modeling Language*). UML adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (Martin Fowler, 2005:1).

2.2.1. Use Case Diagram

Use Case diagram adalah teknik untuk merekam persyaratan fungsional sebuah sistem. *Use case* mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan (Martin Fowler, 2005:141). *Use Case Diagram* dibuat untuk memvisualisasikan atau menggambarkan hubungan antara *Actor* dan *Use Case*.



Gambar 2.2. Actor dan Use Case

Actor mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. Sebuah *actor* mungkin hanya memberikan informasi inputan pada sistem, hanya menerima informasi dari sistem atau keduanya menerima, dan memberi informasi pada sistem. *Actor* hanya berinteraksi dengan *use case*, tetapi tidak memiliki kontrol atas *use case*.

Sedangkan *Use Case* adalah gambaran fungsionalitas dari suatu sistem, sehingga *customer* atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.

2.2.2. Class Diagram

Class Diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka. *Class Diagram* juga menunjukkan properti dan operasi sebuah class dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut (Martin Fowler, 2005:53).

2.2.3. *Activity Diagram*

Activity Diagram adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja. Dalam beberapa hal, diagram ini memainkan peran mirip sebuah diagram alir, tetapi perbedaan prinsip antara diagram ini dan notasi diagram alir adalah diagram ini mendukung behavior paralel (Martin Fowler, 2005:163).

2.2.4. *Statechart Diagram*

Statechart Diagram adalah teknik yang umum digunakan untuk menggambarkan behavior sebuah sistem (Martin Fowler, 2005:151). *Statechart Diagram* menggambarkan semua *state* atau kondisi yang dimiliki oleh suatu *object* dari suatu *class* dan kejadian yang menyebabkan *state* berubah. Kejadian dapat berupa object lain yang mengirim pesan.

2.2.5. *Sequence Diagram*

Sequence diagram adalah penjabaran behavior sebuah skenario tunggal. *Sequence diagram* menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek-objek ini di dalam use case (Martin Fowler, 2005:81). *Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah skenario. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara object juga interaksi antara object, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

2.2.6. *Component Diagram*

Component merupakan bagian fisik dari sebuah sistem, karena menetap di komputer tidak berada di analisis. *Component* terhubung melalui antarmuka yang digunakan dan dibutuhkan (Martin Fowler, 2005:189). *Component* merupakan implementasi *software* dari sebuah atau lebih *class*. *Component* dapat berupa *source code*, komponen biner, atau *executable component*. Sebuah komponen berisi informasi tentang *logic class* atau *class* yang diimplementasikan sehingga membuat

pemetaan dari *logical view* ke *component view*. Sehingga *component diagram* merepresentasikan dunia riil yaitu *component software* yang mengandung *component*, *interface* dan *relationship*.

2.2.7. Deployment Diagram

Deployment Diagram menunjukkan susunan fisik sebuah sistem, menunjukkan bagian perangkat lunak mana yang berjalan pada perangkat keras mana (Martin Fowler, 2005:137). *Deployment Diagram* juga menggambarkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian-bagian hardware, menunjukkan hubungan komputer dengan perangkat (*nodes*) satu sama lain dan jenis hubungannya.

2.3. Basis Data (Database)

Menurut Fathansyah (2007:2), basis data atau database terdiri dari dua kata, yaitu Basis dan Data. Basis dikatakan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek yang direkam dalam bentuk angka, simbol, huruf, teks, gambar, bunyi dan kombinasinya. Sehingga Basis Data dapat didefinisikan dalam sejumlah sudut pandang sebagai berikut :

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file, tabel, atau arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik

2.4. Metode Penentuan Kualitas Perangkat Lunak

Metode penentuan kualitas perangkat lunak dalam penelitian ini menggunakan metode McCall. Menurut taksonomi McCall, atribut tersusun secara hirarkis, dimana level atas (*high-level attribute*) disebut faktor (*factor*) dan level bawah (*low-level attribute*) disebut dengan kriteria (*criteria*). Faktor menunjukkan atribut kualitas produk dilihat dari sudut pandang pengguna. Sedangkan kriteria adalah parameter kualitas produk dilihat dari sudut pandang perangkat lunaknya sendiri (Romi Satrio Wahono, 2006). Faktor dan kriteria ini memiliki hubungan sebab akibat (*cause-effect*), seperti ditunjukkan Tabel 2.1.

Tabel 2.1 Faktor Kualitas Berdasarkan Metode McCall

No	Faktor	Kriteria Kualitas
1	Ketepatan (Correctness)	Kelengkapan, konsistensi, ketertelusuran
2	Keandalan (Reliability)	Akurasi, toleransi kesalahan, konsistensi, kesederhaan
3	Efisiensi (Efficiency)	Efisiensi eksekusi, efisiensi penyimpanan
4	Integritas (Integrity)	Kontrol akses, akses audit
5	Kegunaan (Usability)	Komunikasi, pengoperasian, training
6	Perbaikan (Maintainability)	Konsistensi, singkat, sederhana, teratur, dokumentasi diri
7	Pengetesan (Testability)	Kesederhanaan, teratur, instrumentasi, dokumentasi diri
8	Fleksibilitas (Flexibility)	upgrade, umum, modularitas, dokumentasi diri
9	Portabilitas (Portability)	Sistem kebebasan Software, Kebebasan Hardware, Dokumentasi Diri, modularitas
10	Penggunaan Kembali (Reusability)	Umum, Sistem kebebasan Software, Kebebasan Hardware, Dokumentasi Diri, modularitas
11	Interoperabilitas (Interoperability)	Komunikasi Commonality, Commonality data, modularitas

Dari sebelas faktor kualitas menurut taksonomi McCall seperti ditunjukkan pada Tabel 2.1, untuk menentukan kualitas perangkat lunak cukup dengan lima faktor. Lima faktor untuk menentukan kualitas perangkat lunak tersebut adalah faktor *Ketepatan (Correctness)*, *Keandalan (Reliability)*, *Efisiensi (Efficiency)*, *Kegunaan (Usability)*, dan *Perbaikan (Maintainability)* (Romi Satrio Wahono, 2006).