

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Aktivitas yang terjadi pada perusahaan, instansi atau organisasi tidak terlepas dari sistem sebagai pengelola data dari serangkaian aktivitas agar terlaksana secara optimal. Sistem ini digunakan untuk mengumpulkan, mengolah dan menyediakan informasi. Berikut penjelasan mengenai konsep dasar sistem.

A. Pengertian Sistem

“Sistem bisa diartikan sebagai sekumpulan sub sistem, komponen ataupun element yang saling bekerja sama dengan tujuan yang sama untuk menghasilkan *output* yang sudah ditentukan sebelumnya. Dia juga berpendapat bahwa perusahaan adalah sebuah sistem yang terdiri dari beberapa departemen yang bertindak sebagai subsistem yang membentuk sistem perusahaan tersebut.” (Mulyani & Sistematika, 2017)

Sedangkan pendapat lain mengatakan sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.

B. Karakteristik dan Klasifikasi Sistem

Sistem memiliki ciri yang dapat diklasifikasikan sebagai berikut :

1. Komponen Sistem (*Component System*)

Sistem terdiri dari sejumlah komponen yang saling berinteraksi / bekerjasama membentuk satu kesatuan. Komponen = elemen = bagian = subsistem, (berupa alat, manusia, dsb)

2. Batas Sistem (*Boundary*)

Batas sistem adalah daerah yang membatasi antara satu sistem dengan sistem lainnya atau dengan lingkungan luarnya. Batas sistem memungkinkan dipandang sebagai satu kesatuan serta menunjukkan *scope* sistem.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan adalah apapun di luar batas sistem yang mempengaruhi operasi sistem. Lingkungan ini dapat bersifat menguntungkan / merugikan.

4. Penghubung Sistem (*Interface*)

Penghubung sistem merupakan media penghubung antara satu sistem dengan subsistem lainnya. Melalui *interface*;

- Sumber daya mengalir dari satu sistem ke sistem lainnya,
- *Output* satu subsistem menjadi *input* bagi lainnya,
- Satu subsistem berintegrasi dengan lainnya membentuk satu kesatuan.

5. Masukan Sistem (*input*)

Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan ini dapat berupa *maintenance input* & *signal input*;

a. *Maintenance input* : energi yang dimasukkan agar sistem dapat beroperasi, contoh : program komputer.

b. *Signal input* : energi yang diproses menjadi *output*, contoh : data.

6. Keluaran Sistem (*Output*)

Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna (contoh : informasi), dan sisa pembuangan (contoh : panas dari komputer). Keluaran sistem dapat menjadi *input* bagi subsistem lainnya atau kepada supra sistem.

7. Pengolah Sistem (*Process*)

Pengolah sistem ini merubah *input* melalui proses menjadi *output* (transformasi / proses), juga memperhitungkan batas-batas sistem / sub-sistem dan pengaruh dari lingkungan baik yang bersifat menguntungkan atau merugikan.

8. Sasaran (*Objective*)

Sistem pasti memiliki tujuan (*goal*) atau sasaran (*objective*), sasaran sistem ini menentukan *input* yang dibutuhkan dan *output* yang dihasilkan.

(Nafiudin, 2019)

Klasifikasi sistem dapat digolongkan sebagai berikut :

1. *Abstract System* adalah sistem yang tidak tampak secara fisik, karena hanya berupa pemikiran atau ide-ide. Contoh, Sistem Teologi yang merupakan suatu sistem yang menggambarkan hubungan manusia dengan Tuhan.
2. Sistem Fisik (*Physical System*) adalah sistem yang tampak secara fisik, Contoh Sistem Komputer.
3. Sistem Alamiah (*Natural System*) adalah sistem yang terjadi dari proses-proses alam. Contoh Sistem Geologi.
4. Sistem Buatan Manusia (*Human Made System*) adalah suatu sistem yang dirancang atau didisain oleh manusia, Contoh Sistem Informasi.
5. Sistem Deterministik (*Deterministic System*), adalah sistem yang beroperasi dengan tingkah laku yang dapat diramalkan. Interaksi antar elemen-elemen dapat dideteksi, sehingga *outputnya* juga dapat diramalkan. Contoh sistem komputer.
6. Sistem Probabilitas (*Probabilistic System*), adalah sistem yang tidak bisa diramalkan. Contohnya Sistem manusia.

7. Sistem Tertutup (*Closed System*), adalah sistem yang tidak berhubungan dengan lingkungan luarnya.
8. Sistem Terbuka (*Open System*), adalah sistem yang berhubungan atau dipengaruhi oleh lingkungan luarnya. (Ahmad & Wali, 2018)

C. Definisi Informasi

“Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Sumber informasi adalah data. Data kenyataan yang menggambarkan suatu kejadian – kejadian dan kesatuan nyata. Kejadian – kejadian (*event*) adalah kejadian yang terjadi pada saat tertentu.” (Tyoso, 2016). “Data adalah sekumpulan fakta, bilangan, angka dan simbol yang dibentuk dan diolah menjadi informasi (*Scott*). Data adalah Gambaran / Fakta secara relatif belum berarti bagi si penerima (*Davis*).” (Ahmad & Wali, 2018)

Maka Jika disederhanakan menjadi, Informasi adalah hasil dari pemrosesan data dan data adalah fakta-fakta yang belum diproses.

D. Informasi yang Berkualitas

Ciri – ciri informasi yang berkualitas adalah sebagai berikut :

1. *Accessibility* : Informasi harus mudah diakses, ada / tersedia, semakin mudah dan semakin banyak informasi maka akan semakin baik.
2. *Timelines*: Tepat waktu, informasi yang terlambat akan berakibat tidak baik, informasi yang baik harus cepat.
3. *Relevance*: Informasi yang dihasilkan relevan dan sesuai dengan kebutuhan organisasi / perusahaan atau orang yang membutuhkannya.

4. *Accuracy*: Informasi harus tepat, akurat, bebas dari kesalahan.

5. *Precisian*: Informasi harus presisi atau terperinci / detail.

6. *Useful* : Informasi yang bermanfaat, memiliki nilai kegunaan. (Ahmad & Wali, 2018)

E. Pengertian Sistem Informasi

Sistem memiliki tujuan untuk menghasilkan informasi yang nantinya informasi tersebut akan digunakan sebagai dasar pertimbangan untuk pencapaian didalam suatu Organisasi atau Perusahaan. “Sistem informasi adalah kumpulan komponen yang saling berhubungan dalam mengumpulkan, memproses, menyimpan, menyediakan dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan pengendalian di dalam organisasi” (Fridayanthie, 2016)

F. Komponen – Komponen Sistem Informasi Komponen

- komponen dari sistem informasi adalah :

1. Perangkat keras (*hardware*) : mencakup peranti-peranti fisik seperti komputer dan printer.
2. Perangkat lunak (*software*) atau program : sekumpulan instruksi yang memungkinkan perangkat keras untuk dapat memproses data.
3. Prosedur (*procedure*) : sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki.
4. Orang (*brainware*) : semua pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan dan penggunaan keluaran sistem informasi.
5. Basis data (*database*) : sekumpulan table, hubungan dan lain-lain yang berkaitan dengan penyimpanan data.

6. Jaringan komputer dan komunikasi data : sistem penghubung yang memungkinkan sumber (*resources*) dipakai secara bersama atau diakses oleh sejumlah pemakai. (Swara & Pebriadi, 2016)

G. Pengertian Rumah Sakit Ibu dan Anak

Rumah Sakit Ibu dan Anak merupakan suatu wadah untuk melayani dan memenuhi kebutuhan pasien (ibu, ibu hamil, bayi dan anak umur 0-14th) pada masa pra kehamilan, kehamilan, persalinan, perawatan ibu dan bayi, tumbuh kembang anak, imunisasi, KB dan masalah-masalah yang berhubungan dengan obstetric dan ginekologi (kandungan dan kebidanan) dan juga melayani konsultasi kesehatan terkait dengan masalahmasalah reproduksi ibu dimana semua pelayanan kesehatan tersebut harus memenuhi standar pelayanan kesehatan. (KRISTIANTO, 2018)

Untuk mendukung operasional rumah sakit diperlukannya sarana dan prasarana sebagai penunjang. Sarana dan prasarana rumah sakit itu sendiri dapat diartikan sebagai bangunan, alat, jaringan dan sistem yang digunakan untuk penyelenggaraan pelayanan kesehatan

H. Basis Data

Basis data atau Database merupakan kumpulan file - file yang saling berkaitan dan berinteraksi, relasi tersebut bila ditunjukkan dengan kunci dari tiap - tiap file yang ada. Satu database menunjukkan suatu kumpulan data yang dipakai dalam suatu lingkup perusahaan, instansi. Pengolahan database merupakan suatu cara yang dilakukan terhadap file-file yang berada di suatu instansi yang mana file tersebut dapat disusun, diurut, diambil sewaktu-waktu serta dapat ditampilkan dalam bentuk suatu laporan sehingga dapat mengolah file-file yang berisikan informasi tersebut secara rapi. (Fridayanthie, 2016)

I. Komponen Basis Data

Terdapat 6 komponen dari basis data (*database*) yaitu sebagai berikut

: 1. *Hardware*

Hardware atau perangkat keras yang dimaksud disini adalah perangkat komputer standar sebagai pendukung operasi pengolahan data yang terdiri dari CPU, disk, terminal, memori, dan lain – lain.

2. *Software* Sistem Operasi (*Operating System*)

Software sistem operasi (OS) merupakan perangkat lunak yang bersifat opsional untuk memfungsikan, mengendalikan seluruh sumber daya, serta melakukan operasi dasar pada sistem operasi komputer. Contoh *software* OS yakni *Windows 9x, Linux, Unix, Windows 2000/XP*.

3. *Software* Pengelola Basis Bata (DBMS)

Untuk memasukan data ke dalam *database* atau sebaliknya, mengambil data dari *database*, diperlukan perangkat lunak yang disebut dengan *Database Management System* (DBMS) atau Sistem Manajemen Basis Data. Dengan DBMS, *user* dapat mengontrol, memelihara, dan mengakses data secara praktis dan efisien.

DBMS adalah perantara untuk *user* dengan *database*, sehingga interaksi antara *user* dengan *database*, sehingga interaksi antara *user* dengan *database* memerlukan bahasa pemrograman tertentu. Adapun bahasa pemrograman tersebut terdiri dari dua macam yaitu :

- a. *Data Definition Language* (DDL) digunakan untuk membuat tabel baru, memuat indeks, ataupun mengubah tabel.
- b. *Data Manipulation Language* (DML) digunakan untuk memanipulasi dan pengambilan data dari *database*, seperti menambahkan data baru, menghapus data, atau mengubah data.

4. *Software* Program Aplikasi

Perangkat lunak program aplikasi *database* yang umum digunakan dalam pemrograman dan dianggap sebagai perangkat *database high level*, antara lain :

- a. *MySQL*, kelebihanannya stabil dan cukup tangguh, gratis, security yang baik, mendukung transaksi, dan fleksibel dengan berbagai program. Namun perangkat lunak ini kurang mendukung koneksi bahasa pemrograman *Visual Basic, Delphi, atau Foxpro*. Selain itu tidak mampu menangani data dalam jumlah besar dan belum mendukung *windowing function*.
- b. *Microsoft SQL Server*, DBMS ini mampu mengelola data benar, keamanan baik, mampu melakukan *back up, recovery, dan rollback data*, serta bisa membuat *database mirroring* dan *clustering*. Tetapi untuk mendapatkannya harus dengan biaya mahal karena *software* ini berlisensi, dan perangkat ini hanya bisa dijalankan di *Microsoft Windows*.
- c. *Oracle*, Jenis ini termasuk rumit sehingga memerlukan DBA yang handal, dan membutuhkan biaya yang mahal. Sesuai dengan harganya, *software* ini bisa digunakan oleh organisasi atau perusahaan besar karena memiliki fitur – fitur yang lengkap.

5. Data

Data mengandung pengertian sebagai kumpulan dari sejumlah file atau tabel yang membentuk sebuah *database*. Data merupakan komponen terakhir dari bangunan *database* yang tersusun secara berjenjang dari karakter, *field, record, file*, dan data.

Di dalam *database* dikenal istilah entitas (*entity*) dan atribut. Entitas bisa berupa tempat, kejadian, konsep, atau orang yang informasinya direkam.

Setiap entitas memiliki sebutan untuk mewakili suatu entitas, yang disebut atribut. Sebagai contoh, entitas mahasiswa pada suatu perguruan tinggi memiliki atribut nama, nomor induk, tempat dan tanggal lahir, jenis kelamin, dan sebagainya. Informasi tentang mahasiswa bersangkutan selanjutnya diproses menjadi *file* tertentu, digabungkan dengan *file-file* mahasiswa lainnya, sehingga menjadi data tentang kemahasiswaan.

6. Pengguna (*User*)

User adalah orang-orang yang memiliki otoritas untuk berinteraksi dengan sistem *database*. *User* diklasifikasikan menjadi:

- a. *Database Administrator*, yaitu *user* yang menjadi pusat pengendali seluruh sistem *database*. Berwenang untuk menentukan pola struktur *database*, memodifikasinya, membagi tugas pengolahan, dan sebagainya.
- a. *Application Programmer*, yaitu *user* yang membuat program aplikasi untuk berinteraksi dengan sistem atau mengakses *database* dengan cara pemanggilan bahasa pemrograman tertentu.
- b. *Sophisticated User*, yaitu *user* yang melakukan *request* (permintaan) untuk berinteraksi dengan sistem dalam bentuk bahasa *query database*, seperti menggunakan *MS-Access*, *SQL*, dan sebagainya.
- c. *Specialized User*, yaitu *user* yang menulis aplikasi *database* tidak untuk pemrosesan data secara tradisional melainkan untuk keperluan khusus seperti pengolahan citra, sistem pakar, atau aplikasi *artificial intelegent*.
- d. *Naive User*, yaitu *user* umum yang tidak memerlukan keahlian khusus dalam pemrograman dan cukup menjalankan aplikasi yang tersedia untuk berinteraksi dengan sistem untuk mengakses data. (Rachmadi, 2020)

2.2 Teori Pendukung

2.2.1 Unified Modeling Language (UML)

“*Unified Modeling Language (UML)* adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem.” (Ade Hendini, 2016)

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

1. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.

2. Diagram Aktivitas (*Activity Diagram*)

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis.

3. Diagram Urutan (*Sequence Diagram*)

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

4. Diagram Kelas (*Class Diagram*)

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan

dan

tanggung jawab
entitas yang
menentukan
perilaku sistem.
Class

Diagram juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan.

Class Diagram secara khas meliputi : Kelas (*Class*), Relasi *Associations*, *Generalisation* dan *Aggregation*, atribut (*Attributes*), operasi (*operation/method*) dan *visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality*.

Tabel.II.1
Multiplicity Class Diagram

<i>Multiplicity</i>	<i>Penjelasan</i>
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimal 4

5. *Deployment Diagram*

Deployment Diagram digunakan untuk menggambarkan detail bagaimana komponen disusun di infrastruktur sistem.(Ade Hendini, 2016)

2.2.2 ERD (Entity Relationship Diagram)

1. Pengertian ERD

“ERD (*Entity Relationship Diagram*) adalah model teknik pendekatan yang menyatakan atau menggambarkan hubungan suatu model. Didalam hubungan ini tersebut dinyatakan yang utama dari ERD adalah menunjukkan objek data (*Entity*) dan hubungan (*Relationship*), yang ada pada *Entity* berikutnya.” (Fridayanthie, 2016)

Simbol-simbol dalam ERD (*Entity Relationship Diagram*) adalah sebagai berikut:

- a. Entitas : suatu yang nyata atau abstrak yang mempunyai karakteristik dimana kita akan menyimpan data.
- b. Atribut: ciri umum semua atau sebagian besar instansi pada entitas tertentu.
- c. Relasi: hubungan alamiah yang terjadi antara satu atau lebih entitas.
- d. *Link*: garis penghubung atribut dengan kumpulan entitas dan kumpulan entitas dengan relasi.(Fridayanthie, 2016)

2. Kardinalitas Relasi

Kardinalitas relasi terdiri dari :

- a. Satu ke satu (*One to One*) Setiap elemen dari Entitas A berhubungan paling banyak dengan elemen pada Entitas B. Demikian juga sebaliknya setiap elemen B berhubungan paling banyak satu elemen pada Entitas A.
- b. Satu ke banyak (*One to Many*) Setiap elemen dari Entitas A berhubungan dengan maksimal banyak elemen pada Entitas B. Dan sebaliknya setiap elemen dari Entitas B berhubungan dengan paling banyak satu elemen di Entitas A.
- c. Banyak ke satu (*Many to One*) Setiap elemen dari Entitas A berhubungan paling banyak dengan satu elemen pada Entitas B. Dan sebaliknya setiap elemen dari Entitas B berhubungan dengan maksimal banyak elemen di entitas A.
- d. Banyak ke banyak (*Many to Many*) Setiap elemen dari Entitas A berhubungan maksimal banyak elemen pada Entitas B demikian sebaliknya.(Fridayanthie, 2016)

2.2.3 LRS (Logical Record Structure)

“*Logical Record Structured* (LRS) adalah representasi dari struktur *record-record* pada tabel-tabel yang terbentuk dari hasil relasi antar himpunan entitas. Menentukan kardinalitas, jumlah tabel, dan *Foreign Key* (FK).” (Fridayanthie, 2016)

Berikut adalah cara membentuk skema *database* atau LRS (*Logical Record Structured*) berdasarkan *Entity Relationship Diagram* :

- a. Jika relasinya satu-ke-satu, maka *foreign key* diletakan pada salah satu dari dua entitas yang ada tau menyatukan kedua entitas tersebut.
- b. Jika relasinya satu-ke-banyak, maka *foreign key* diletakan pada entitas Many.
- c. Jika relasinya banyak-ke-banyak, maka dibuat “file konektor” yang berisi dua *foreign key* yang berasal dari kedua entitas. (Fridayanthie, 2016)

