

BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

A. Konsep Dasar Sistem Informasi

Pengertian dan definisi sistem pada berbagai bidang berbeda – beda, tetapi meskipun istilah sistem yang digunakan bervariasi, semua sistem pada bidang – bidang tersebut mempunyai beberapa persyaratan umum, yaitu sistem harus mempunyai elemen, lingkungan, interaksi antar elemen, interaksi antar elemen dengan lingkungannya, dan yang terpenting adalah sistem harus mempunyai tujuan yang akan dicapai.

Berdasarkan persyaratan ini, sistem dapat didefinisikan sebagai seperangkat elemen yang digabungkan satu dengan lainnya untuk suatu tujuan bersama. Kumpulan elemen terdiri dari manusia, mesin, prosedur, dokumen, data atau elemen lain yang terorganisir dari elemen – elemen tersebut. Elemen sistem disamping berhubungan satu dengan yang lain, juga berhubungan dengan lingkungannya untuk mencapai tujuan yang telah ditentukan sebelumnya. Menurut (Jogiyanto, 2019) analisis dan desain sistem informasi adalah Sistem dapat di definisikan dengan pendekatan prosedur dan dengan pendekatan komponen.

B. Definisi Informasi

Menurut (Jogiyanto, 2019) Informasi adalah data yang telah diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya”. Siklus Informasi Data merupakan bentuk yang masih mentah, belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model untuk dihasilkan informasi. Data yang diolah untuk menghasilkan informasi menggunakan suatu model proses tertentu. Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti

menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus. Siklus ini disebut dengan siklus informasi (*information cycle*) atau disebut juga siklus pengolahan data (*data processing cycle*).

C. Kualitas Informasi

1. Kualitas dari suatu informasi (*quality of information*) tergantung dari tiga hal, yaitu informasi harus akurat (*accurate*), tepat pada waktunya (*timeliness*), dan relevan (*relevance*). John Burch dan Grudnitski menggambarkan kualitas dari informasi dengan bentuk bangunan yang ditunjang oleh tiga buah pilar. Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak informasi tersebut.
2. Tepat pada waktunya, berarti informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan di dalam pengambilan 11 keputusan. Bila pengambilan keputusan terlambat, maka dapat berakibat fatal untuk organisasi.
3. Relevan, berarti informasi tersebut mempunyai manfaat untuk pemakainya

D. Definisi Sistem Informasi

Menurut (Ladjamudin, 2018) Sistem Informasi adalah Suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi. Dan juga Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan/atau untuk mengendalikan organisasi. Menurut (Jogyanto, 2019) Sistem Informasi adalah

suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan yang diperlukan.

Komponen Sistem Informasi Menurut (John Burch, 2019) mengemukakan bahwa sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*Build block*), yaitu blok masukan (*input block*), blok model (*model block*), blok keluaran (*output block*), blok teknologi (*technology block*), blok basis data (*database block*) dan blok kendali (*control block*)”.

E. Definisi Pengembangan Sistem

Menurut Jogiyanto dalam (Yakub, 2017) menyimpulkan bahwa: pengembangan sistem berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada karena adanya permasalahan, kesempatan, dan intruksi, maka sistem yang baru perlu dikembangkan untuk memecahkan, permasalahan yang timbul, meraih kesempatan yang ada, dan memenuhi intruksi yang diberikan.

F. Definisi Stock Opname

Menurut (Tamodia, 2018) dalam jurnalnya mengutip “persediaan merupakan barang-barang yang dimiliki untuk kemudian dijual atau digunakan dalam proses produksi atau dipakai untuk keperluan non produksi dalam siklus kegiatan yang normal”.

G. Barcode

Menurut (Yulianto, 2018) mengutip dalam jurnalnya bahwa: Terdapat beberapa kategori barcode berdasarkan kegunaannya, yaitu :

1. Barcode untuk keperluan retail. Barcode untuk keperluan retail, salah satu contohnya adalah UPC (*Universal Price Codes*)
2. Barcode untuk keperluan packaging. Barcode untuk packaging biasanya digunakan untuk pengiriman barang, dan salah satunya adalah barcode tipe ITF.
3. Barcode untuk penerbitan. Barcode untuk keperluan penerbitan, sering digunakan pada penerbitan produk, misalkan barcode yang menunjukkan ISSN suatu buku.

H. Basis Data (Database)

Menurut (Connolly, 2017), database adalah sekumpulan data logikal dan deskripsinya yang saling berhubungan, yang didesain untuk memenuhi kebutuhan informasi sebuah organisasi. Basis data merupakan aspek yang sangat penting dalam sistem informasi karena berfungsi sebagai gudang penyimpanan data yang akan diolah lebih lanjut. 16 Basis data menjadi penting karena dapat mengorganisasi data, menghindari duplikasi data, menghindari hubungan antar data yang tidak jelas dan juga update yang rumit. Beberapa komponen utama database :

1. Perangkat Keras (*Hardware*) Merupakan perangkat keras yang dibutuhkan dalam pengelolaan database, berupa komputer beserta seluruh kelengkapan yang dibutuhkan, seperti prosesor, memori, harddisk sebagai media penyimpanan datanya, dan lain sebagainya.
2. Data Merupakan komponen yang akan diolah sehingga bernilai informasi dan dapat dijadikan sebagai bahan pengambilan keputusan.
3. Perangkat Lunak (*Software*) Perangkat lunak, dalam hal ini DBMS (*Database Management System*), berfungsi sebagai aplikasi untuk mengelola database.4

4. Pengguna (*User*) Dapat diklasifikasikan menjadi tiga yaitu; pengguna akhir, pemrograman aplikasi dan administrator data.

2.2. Peralatan Pendukung (*Tools System*)

Di dalam penulisan laporan tugas akhir ini, penulis memperoleh data-data yang digunakan untuk menggambarkan bentuk logika model dari suatu sistem dengan menggunakan simbol-simbol, lambang-lambang, diagram-diagram yang menunjukkan secara tepat arti dan fungsinya. beberapa peralatan pendukung yang digunakan untuk menggambarkan perancangan sistem informasi yang dibahas.

A. *Unified Modeling Language (UML)*

Menurut (Nugroho, 2020), *UML (Unified Modeling Language)* adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sehingga mudah dipahami, *UML* merupakan diagram yang saling terkait oleh karena itu perlu adanya ke konsistenan diagram yang ada dalam *UML* menurut (Shalahuddin, 2018) terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori:

1. *Structure Diagrams*

Kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan yang termasuk *Structure Diagrams* yaitu: *class diagram*, *Object Diagram*, *Componen Diagram*, *Composite struktur Diagram*.

2. *Behavior Diagrams*

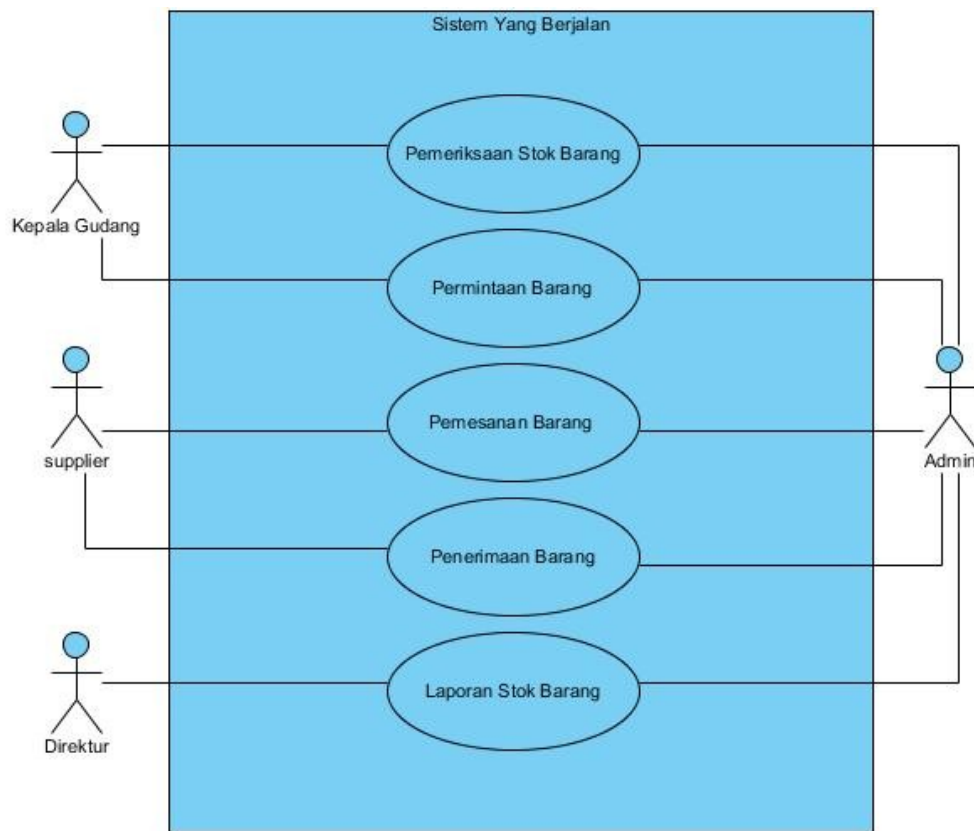
Kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau perubahan yang terjadi pada sebuah sistem yang termasuk *behavior Diagrams* yaitu: *Use case Diagram*, *Activify Diagram*, *State Machine Diagram*

3. *Interaction Diagrams*

Kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem yang termasuk kumpulan diagram interaction diagram yaitu: *Sequence Diagram*, *Communication Diagram*, *Timing Diagram*, *Interaction Diagram*.

a. *Use Case Diagram*

Menurut (Schermuller, 2019) *Use Case Diagram* adalah sebuah gambar dan fungsi sistem yang dipandang dari sudut pandang pemakai, *Actor* adalah segala sesuatu yang perlu berinteraksi dengan sistem untuk pertukaran informasi. Secara grafis menggambarkan, interaksi secara sistem, sistem eksternal dan penggunaan. Dengan kata lain *use case* digunakan untuk mengetahui fungsi apa saja yang ada dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi fungsi tersebut. Fungsi sistem dari sudut pandang sistem *user* secara naratif digunakan untuk secara tekstual menggambarkan sekuensi langkah-langkah dari tiap interaksi.



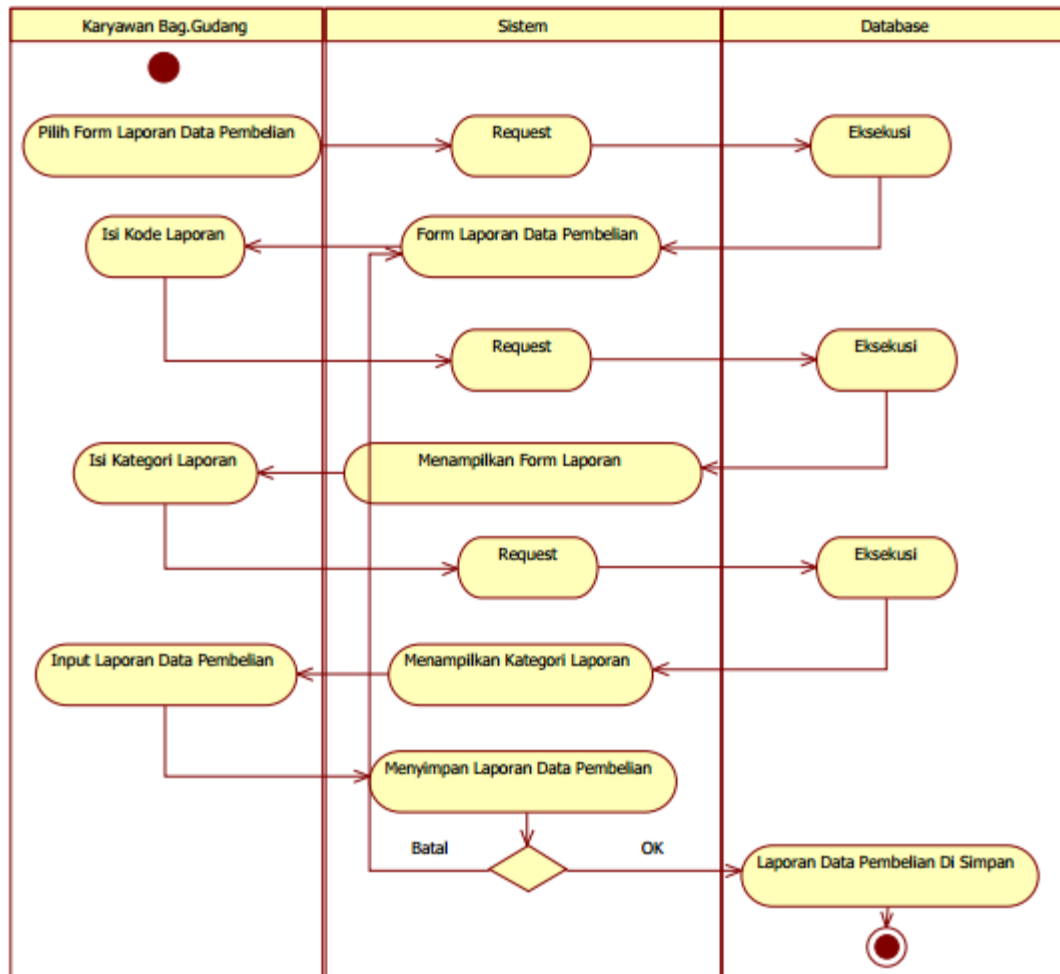
Sumber : Schermuller 2019

Gambar II.1. *Use Case Diagram*

b. Activity Diagram

Diagram Aktivitas (*Activity Diagram*) (Sukamto, 2018), Diagram aktivitas atau activity diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Kegunaan Activity Diagram (Sukamto, 2018). Diagram Aktivitas banyak digunakan untuk mendefinisikan hal-hal berikut:

1. Rancangan proses bisnis.
2. Urutan atau pengelompokkan tampilan dari sistem atau *user interface*.
3. Rancangan pengujian.

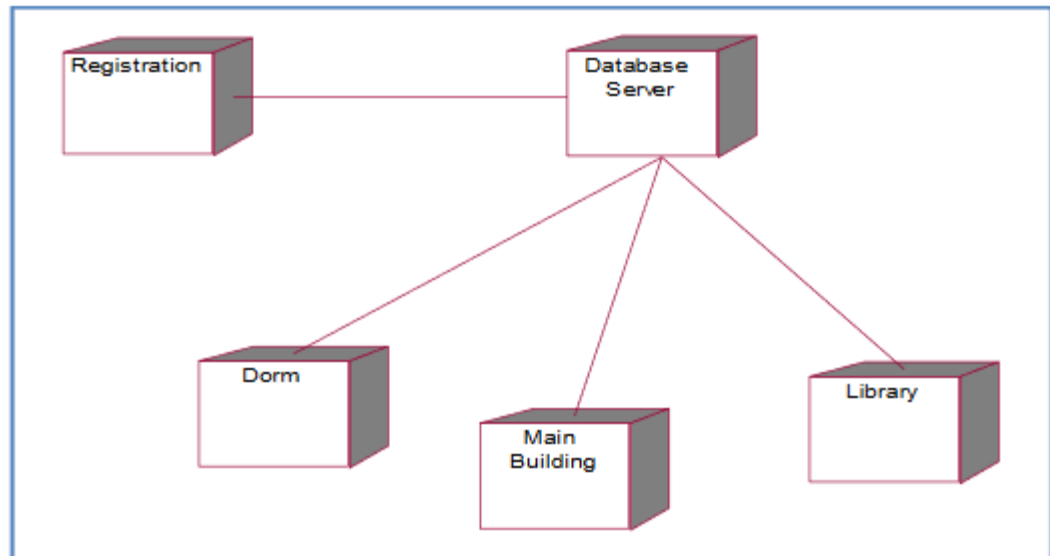


Sumber : Sukamto 2019

Gambar II.2. Activity Diagram

c. Deployment Diagram

Menurut (Satzinger, 2020) *deployment diagram* adalah “Merupakan perangkat keras, sistem perangkat lunak dan lingkungan jaringan dimana sistem akan beroperasi”. *Deployment diagram* hanya ada satu dalam setiap sistem sehingga hanya ada satu dalam setiap model menampilkan semua titik (*node*) suatu jaringan, hubungan antar proses yang dijalankan pada node.



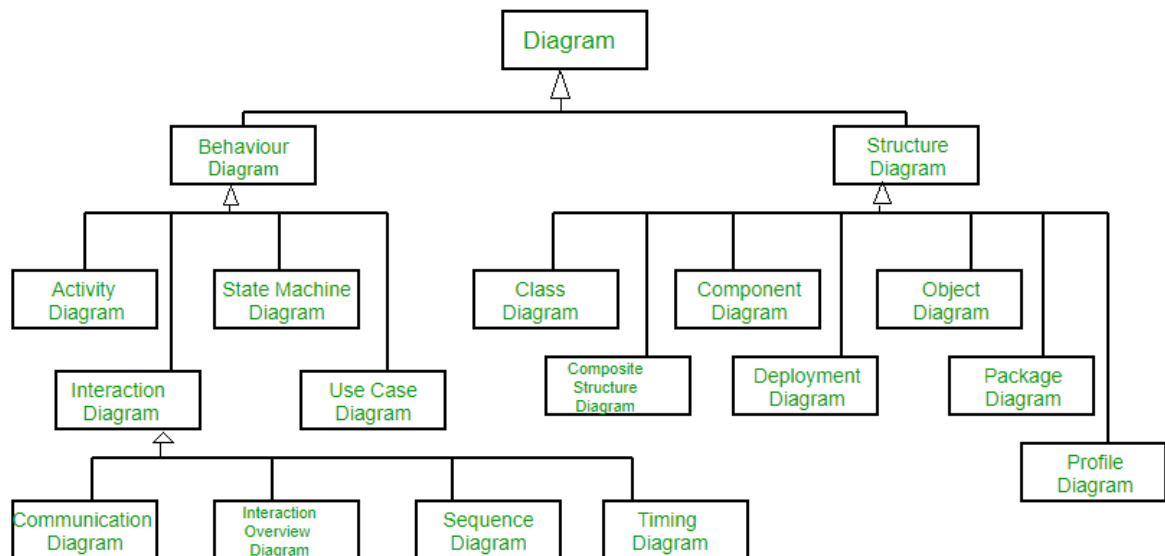
Sumber : Satzinger 2020

Gambar II.3. *Deployment Diagram*

d. *Sequence Diagram*

Sequence Diagram Menurut (Indrajani, 2020), “*Sequence diagram* merupakan suatu diagram interaksi yang menggambarkan bagaimana objek-objek berpartisipasi dalam bagian interaksi dan pesan yang ditukar.

Sequence Diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa pesan yang digambarkan terhadap waktu. suatu diagram interaksi yang menggambarkan bagaimana objek-objek berpartisipasi dalam bagian interaksi dan pesan yang ditukar dalam urutan waktu. *Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa pesan yang digambarkan terhadap waktu. suatu diagram interaksi yang menggambarkan bagaimana objek-objek berpartisipasi dalam bagian interaksi dan pesan yang ditukar dalam urutan waktu.



Sumber : Indrajani 2020

Gambar II.4. *Sequance Diagram*

B. Entity Relation Diagram (ERD)

1. Enterprise Relationship Diagram (ERD)

Menurut (Sutanta, 2019) “*Entity Relationship Diagram (ERD)* merupakan suatu model data yang dikembangkan berdasarkan obyek”. *Entity Relationship Diagram (ERD)* digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis. *Entity Relationship Diagram (ERD)* didasarkan pada suatu persepsi bahwa *real world* terdiri atas obyek-obyek dasar tersebut. Penggunaan *Entity Relationship Diagram (ERD)* relatif mudah dipahami, bahkan oleh para pengguna yang awam. Bagi analis sistem, *Entity Relationship Diagram (ERD)* berguna untuk memodelkan sistem yang nantinya, basis data akan di kembangkan. Model ini juga membantu perancang atau analis sistem pada saat melakukan analis dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasian antardata didalamnya.

2. Komponen *Entity Relationship Diagram* (ERD)

Komponen *ERD* menurut (Sutanta, 2018)sebagai berikut :

- a. Entitas merupakan suatu obyek yang dapat dibedakan dari lainnya yang dapat diwujudkan dalam basis data. Obyek dasar dapat berupa orang, benda, atau hal yang keterangannya perlu disimpan didalam basis data. Untuk menggambarkan sebuah entitas digunakan aturan sebagai berikut :
 - 1) Entitas dinyatakan dengan simbol persegi panjang.
 - 2) Nama entitas dituliskan didalam simbol persegi panjang.
 - 3) Nama entitas berupa kata benda, tunggal.
 - 4) Nama entitas dapat menyatakan maknanya dengan jelas.
- b. Atribut merupakan keterangan-keterangan yang terkait pada sebuah entitas yang perlu disimpan dalam basis data. Atribut berfungsi sebagai penjelas pada sebuah entitas. Untuk menggambarkan atribut digunakan aturan sebagai berikut :
 - 1) Atribut digambarkan dengan simbol ellips.
 - 2) Nama atribut dituliskan didalam simbol ellips.
 - 3) Nama atribut merupakan kata benda, tunggal.
 - 4) Nama atribut sedapat mungkin menggunakan nama yang mudah
- c. Relasi merupakan hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Aturan penggambaran relasi adalah
 - 1) Relasi dinyatakan dengan simbol belah ketupat.
 - 2) Nama relasi dituliskan didalam simbol belah ketupat
 - 3) Nama relasi berupa kata kerja aktif.
 - 4) Nama relasi menyatakan maknanya dengan jelas

C. *Logikal Relation Structure (LRS)*

Menurut (Hasugian, 2019) “*Logical relation structure (LRS)* adalah sebuah model sistem yang digambarkan dengan sebuah diagram-ER akan *LRS*”. Dua metode yang dapat digunakan, dimulai dengan hubungan kedua model yang dapat dikonversikan ke *LRS*. Metode yang lain dimulai dengan Entity Relationship diagram dan langsung dikonversikan ke *LRS*.

1. Konversi ERD ke LRS

Diagram entity relationship diagram harus diubah ke bentuk LRS (struktur record secara logik). Dari bentuk LRS inilah yang nantinya dapat ditransformasikan ke bentuk relasi (tabel). Sebuah model sistem yang digambarkan dengan sebuah ERD akan mengikuti pola pemodelan tertentu. Dalam kaitannya dengan konversi ke LRS, untuk perubahan yang terjadi adalah mengikuti aturan-aturan berikut:

- a. Setiap entitas diubah kebentuk kotak dengan nama entitas, berada diluar kotak dan atribut berada didalam kotak.
- b. Sebuah *relationship* kadang disatukan, dalam sebuah kotak bersama entitas, kadang sebuah kotak bersama-sama dengan entitas, kadang disatukan dalam sebuah kotak tersendiri.

2. Konversi LRS ke relasi (tabel)

Relasi atau tabel adalah bentuk pernyataan data secara grafis 2 (dua) dimensi, yang terdiri dari kolom dan baris. Relasi adalah bentuk visual dari sebuah file, dan tiap tuple dalam sebuah field, atau yang dalam bentuk lingkaran *Diagram entity relationship* dikenal dengan sebutan atribut. Konversi dari *logical record structure*, dilakukan dengan cara :

- a. Nama *logical record structure* menjadi nama relasi.
- b. Tiap atribut menjadi sebuah kolom didalam relasi.

D. Pengkodean

Menurut (Kendall, 2018) menyimpulkan bahwa”Pengkodean adalah proses dari meletakkan data yang berarti dua macam atau data yang sulit dipakai dengan segera”.

1. Kode yang digunakan untuk mempermudah akses data maupun proses data.

a. Kode *Numeric (Numeric Code)*

Kode yang menggunakan angka sebagai simbolnya, mempunyai 10 macam kombinasi angka di dalam kode.

b. Kode *Alphabetic (Alphabetic Code)*

Kode menggunakan simbol huruf, mempunyai 26 kombinasi huruf.

c. Kode *Alphanumeric (Alphanumeric Code)*

Kode dapat dibentuk dari kumpulan angka, huruf dan karakter-karakter khusus (misalnya %, /, - , \$, #, &, ; dan lain sebagainya). Angka merupakan simbol yang banyak digunakan pada sistem kode.

2. Jenis Kode

Jenis kode yang banyak digunakan dalam sistem informasi sebagai kode yang menggunakan gabungan angka, huruf dan karakter khusus.

3. Petunjuk Pembuatan Kode

Beberapa kemungkinan susunan digit (angka), huruf dan karakter-karakter khusus dapat dirancang ke dalam bentuk kode. Didalam merancang suatu kode harus diperhatikan beberapa hal, yaitu :

a. Harus mudah diingat

Supaya kode mudah diingat, maka dapat dilakukan dengan cara menghubungkan kode tersebut dengan objek yang diwakili dengan kodenya.

b. Harus unik

Kode harus unik untuk masing-masing *item* yang diwakilinya. Unik berarti tidak ada kode yang kembar.

c. Harus fleksibel

Kode harus fleksibel sehingga memungkinkan perubahan-perubahan atau penambahan item baru dapat tetap diwakili oleh kode.

d. Harus efisien

Kode harus sependek mungkin, selain mudah diingat juga akan efisien bila direkam atau disimpan di luar komputer. Harus konsisten. Bilamana mungkin, kode harus konsisten dengan kode yang telah dipergunakan.

e. Harus distandarisasi

Kode harus distandarisasi untuk seluruh tingkatan dan departemen dalam organisasi. Kode yang tidak standar akan mengakibatkan kebingungan, salah pengertian dan dapat cenderung terjadi kesalahan pemakaian bagi yang menggunakan kode tersebut.

f. Spasi harus dihindari

Spasi didalam kode sebaiknya dihindari, karena dapat menyebabkan kesalahan di dalam penggunaannya.

g. Hindari karakter yang mirip

Karakter-karakter yang hampir serupa bentuk dan bunyi pengucapannya sebaiknya tidak digunakan dalam kode.

h. Panjang kode harus sama

Masing-masing kode yang sejenis harus mempunyai panjang yang sama.

