

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

Suatu sistem dapat didefinisikan sebagai suatu kesatuan yang terdiri dari dua atau lebih komponen atau sub sistem yang berinteraksi untuk mencapai suatu tujuan. Suatu sistem dapat terdiri dari sistem – sistem bagian (subsistem). Definisi lain dari sistem adalah suatu jaringan kerja dalam prosedur yang salingberhubungan satu sama lain untuk melakukan kegiatan atau melakukan sesuatu dengan tujuan tertentu.

“Sistem adalah sekelompok elemen-elemen yang saling terintegrasi. Dengan maksud yang sama untuk mencapai suatu tujuan. Informasi adalah data yang telah diproses, atau data yang memiliki manfaat/kegunaan. Sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi (Alter,1992), Sebuah sistem informasi mengumpulkan, memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk tujuan yang spesifik (Turban,Mc clean, dan Wetherbhe 1999)”.

Semua definisi sistem diatas tidak bertentangan, yang membedakannya adalah pendekatannya. Dalam mempelajari sistem akan lebih baik mengetahui terlebih dahulu apakah sistem tersebut lebih luas pengertiannya. Sistem mempunyai peranan sangat penting dalam pendekatan untuk mempelajari suatu sistem, pendekatan merupakan kumpulan dari elemen – elemen atau komponen – komponen.Subsistem dari kumpulan sistem tersebut tidak dapat berdiri sendiri tetapi salingberinteraksi dan saling berhubungan membentuk satu kesatuan sehingga tujuan atau sasaran sistem tersebut dapat dicapai.

Model umum sebuah sistem adalah masukan, proses dan keluaran.Hal ini merupakan sistem yang sangat sederhana, sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran.Sistem merupakan segi yang menentukan dan

melukiskan sebuah sistem yang membentuk batasan, sistem terdapat didalam batasan, lingkungan berada di luar batas.

2.2. Konsep Dasar Informasi

Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (*event*) nyata yang digunakan untuk mengambil keputusan.

Sumber dari informasi adalah data. Yang dimaksud dengan data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata. Kejadian-kejadian (*event*) adalah sesuatu yang terjadi pada saat tertentu. Kesatuan nyata (*fact*) adalah berupa suatu objek nyata seperti tempat, benda, dan orang yang betul-betul ada dan terjadi.

Ada beberapa syarat-syarat tentang informasi yang baik dan lengkap. Adapun syarat-syaratnya adalah sebagai berikut :

a. Ketersediaan

Sudah tentu syarat yang mendasar bagi suatu informasi adalah tersedianya informasi itu sendiri. Informasi harus dapat diperoleh bagi orang yang hendak memanfaatkannya.

b. Mudah Dipahami

Informasi harus mudah dipahami oleh pembuat keputusan, baik keputusan yang menyangkut pekerjaan rutin maupun keputusan yang bersifat strategis. Informasi yang rumit dan berbelit-belit hanya akan membuat kurang efektifnya keputusan manajemen.

c. Relevan

Dalam konteks organisasi, informasi yang diperlukan adalah yang benar-benar relevan dalam permasalahan, misi dan tujuan organisasi.

d. Bermanfaat

Sebagai konsekuensi dari syarat-syarat relevansi, informasi juga harus bermanfaat bagi organisasi, karena itu informasi harus dapat tersaji

kedalambentuk-bentuk yang memungkinkan pemanfaatan oleh organisasi yang bersangkutan.

e. **Tepat Waktu**

Informasi harus tersedia pada waktunya. Syarat ini terutama sangat penting pada saat organisasi membutuhkan informasi ketika manajer hendak membuat keputusan.

f. **Keandalan**

Informasi harus diperoleh dari sumber-sumber yang dapat diandalkan kebenarannya. Pengolahan data atau pemberi informasi harus dapat menjamin tingkat kepercayaan yang tinggi atas informasi yang disajikan.

g. **Akurat**

Syarat ini mengharuskan bahwa informasi bersih dari kesalahan dan kekeliruan, ini dapat berarti bahwa informasi harus jelas dan secara akurat mencerminkan makna yang terkandung dari data pendukungnya.

h. **Konsisten**

Informasi tidak boleh mengandung kontradiksi didalam penyajiannya karena konsistensi merupakan syarat penting sebagai dasar pengambilan keputusan.

2.3. Sistem Informasi

Sistem Informasi dapat didefinisikan sebagai suatu susunan dari orang, data, proses, dan teknologi informasi yang saling berhubungan untuk mengumpulkan, memproses, menyimpan, dan menyediakan keluaran informasi yang diperlukan untuk mendukung suatu organisasi.

Sistem informasi dapat digolongkan menurut fungsinya, antara lain adalah sebagai berikut ini:

- a. *Transaction Processing System* (TPS), suatu sistem informasi yang menangkap dan memproses data tentang transaksi bisnis. Seperti pesanan (order), kartu catatan waktu, pembayaran, reservasi, dan sebagainya.

- b. *Management Information System(MIS)*, suatu sistem informasi yang disediakan untuk menghasilkan laporan yang berorientasi pada manajemen yang berdasarkan pada proses transaksi dan operasi dari organisasi. Atau dengan kata lain menggunakan data transaksi untuk menghasilkan informasi yang dibutuhkan oleh manajer untuk menjalankan bisnis.
- c. *Decision Support System(DSS)*, suatu sistem informasi yang membantu mengidentifikasi pengambilan keputusan yang mungkin atau menyediakan informasi untuk membantu pengambilan keputusan manajemen.
- d. *Executive Information System(EIS)*, suatu sistem informasi yang mendukung perencanaan dan kebutuhan penilaian dari manajer eksekutif. EIS dikhususkan untuk kebutuhan informasi yang unik dari para eksekutif yang merencanakan bisnis dan menilai pencapaian rencana bisnis tersebut.
- e. *Expert System(ES)*, suatu sistem informasi yang menangkap keahlian dari para pekerja dan kemudian menirukan keahlian tersebut untuk dimanfaatkan oleh orang yang tidak ahli.
- f. *Communications and Collaboration System*, suatu sistem informasi yang memberikan peluang komunikasi yang lebih efektif antara para pekerja, mitra, pelanggan, dan para penyalur untuk meningkatkan kemampuan mereka untuk bekerja sama.
- g. *Office Automation System*, suatu sistem informasi yang mendukung cakupan luas dari aktivitas kantor yang disediakan untuk meningkatkan alur kerja(work flow) antara para pekerja dan membantu karyawan membuat dan membagi dokumen yang dapat mendukung aktivitas kantor sehari-hari.

Komponen Sistem Informasi adalah sebagai berikut:

- a. **Input**, merupakan data yang masuk ke dalam sistem informasi. Komponen ini perlu ada karena merupakan bahan dasar dalam pengolahan informasi.
- b. **Proses**, menggambarkan bagaimana suatu data diproses untuk menghasilkan suatu informasi bernilai tambah.
- c. **Output**, berupa informasi yang berguna bagi para pemakainya. Output merupakan komponen yang harus ada di sistem informasi.

- d. Basis Data, adalah kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.
- e. Model, Informasi yang dihasilkan oleh sistem informasi berasal dari data yang diambil dari basis data yang diolah lewat suatu model-model tertentu. Model-model yang digunakan di sistem informasi dapat berupa model logika yang menunjukkan suatu proses perbandingan logika atau model matematik yang menunjukkan proses perhitungan matematika.
- f. Teknologi, teknologi merupakan komponen yang penting di sistem informasi. Tanpa adanya teknologi yang mendukung, maka sistem informasi tidak akan dapat menghasilkan informasi yang tepat waktunya.
- g. Penyimpanan, suatu kegiatan untuk memelihara dan penyimpanan data.
- h. Kontrol, digunakan untuk menjamin bahwa informasi yang dihasilkan oleh sistem informasi merupakan informasi yang akurat.

Fokus utama sistem informasi berbasis komputer adalah untuk aplikasi perkantoran (*Office Automation*), dimana komputer memiliki porsi semakin berarti untuk aplikasi Sistem Informasi Akuntansi (SIA), Sistem Informasi Manajemen (SIM), dan *Decision Support System* (DSS).

2.4. Pengembangan Sistem Berorientasi Obyek dengan UML

Menurut Munawar (2005) mendefinisikan obyek sebagai berikut: Obyek, baik yang konkret maupun konseptual, selalu ada di sekeliling kita. Sebuah obyek memiliki keadaan sesaat (*state*) dan perilaku (*behavior*). State suatu obyek adalah kondisi obyek tersebut yang dinyatakan dalam *attribute/properties*. Sedangkan perilaku sebuah obyek mendefinisikan bagaimana sebuah obyek bertindak/beraksi dan memberikan reaksi.

2.4.1. Unified Modelling Language (UML)

UML adalah bahasa nyata (grafis) untuk menggambarkan, menetapkan, membangun, dan mendokumentasikan sesuatu (benda) pada sebuah perangkat lunak secara intensif. Dengan UML dapat membuat model untuk semua

jenis aplikasi software, dimana aplikasi tersebut dapat berjalan pada hardware, system operasi, dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. UML lebih cocok untuk penulisan software dalam bahasa-bahasa pemrograman berorientasi objek seperti C++ (Cbuilder), Java (Jbuidler).

UML mendefinisikan notasi dan sintaksis/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram software, dan sintaksis UML mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya, yaitu:

- a. OOD (*Object Oriented Design*), dikembangkan oleh Grady Booch. Keunggulan metode Booch adalah pada detail dan karyanya dengan notasi dan elemen.
- b. OMT (*Object Modeling Technique*), dikembangkan oleh James Rumbaugh. Tahapan utama dalam metode ini adalah analisis, design sistem, design obyek, dan implementasi. Keunggulan metode ini adalah dalam penotasian yang mendukung semua konsep OO (*Object Oriented*).
- c. OOSE (*Object Oriented Software Engineering*), dikembangkan oleh Ivar Jacobson. Metode OOSE lebih memberi penekanan pada usecase. Keunggulan metode ini mudah dipelajari karena memiliki notasi yang sederhana namun mencakup seluruh tahapan dalam rekayasa perangkat lunak.

UML merupakan hasil kerja dari konsorsium berbagai organisasi yang berhasil dijadikan sebagai standar baku dalam OOAD (*Object Oriented Analysis Design*). Untuk membuat suatu model, UML mendefinisikan diagram-digram grafis berikut ini:

- a. Usecase diagram
- b. Class diagram
- c. Behaviour diagram:
 - 1) Statechart diagram
 - 2) Activity diagram
 - 3) Interaction diagram:

- a) Sequence diagram
- b) Collaboration diagram
- d. Component diagram
- e. Deployment diagram

2.4.2. Analisa Sistem Berorientasi Obyek

Analisa berorientasi obyek(Object-Oriented Analysis) adalah sebuah pendekatan yang digunakan untuk: (Whitten, 2004 :430)

- a. Mempelajari obyek-obyek yang ada untuk mengetahui apakah obyek tersebut dapat digunakan berulang kali atau dapat disesuaikan untuk keperluan yang baru.
- b. Menggambarkan obyek yang baru atau memodifikasi obyek, yang akan dikombinasi dengan obyek-obyek yang sudah ada ke dalam sebuah aplikasi bisnis komputer yang bermanfaat.

Konsep dasar analisa dan perancangan berorientasi obyek adalah sebagai berikut:



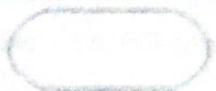


- a. *Object*, sesuatu yang dapat dilihat, disentuh, atau dirasakan. (Jeffery L. Whitten, 2004)
- b. *Class*, sekumpulan objek yang sejenis, yang memiliki perilaku dan attribute yang sejenis. (Jeffery L. Whitten et al, 2004)
- c. *Attribute*, sebuah data yang mewakili karakteristik yang dimiliki obyek. (Jeffery L. Whitten et al, 2004)
- d. *Behavior* (Perilaku), kumpulan sesuatu yang dapat dilakukan oleh obyek. Disebut juga sebagai method, operation atau service. (Jeffery L. Whitten, 2004)
- e. *Inheritance* (Turunan), sebuah konsep dimana method dan/atau attribute yang dimiliki oleh sebuah obyek dapat diturunkan atau digunakan(reused) oleh objek lain. (Jeffery L. Whitten et al, 2004)
- f. *Polymorphisme*, sebuah konsep dimana objek lain dapat merespon operasi yang sama, tetapi dengan implementasi yang berbeda. (Jeffery L. Whitten et al, 2004)

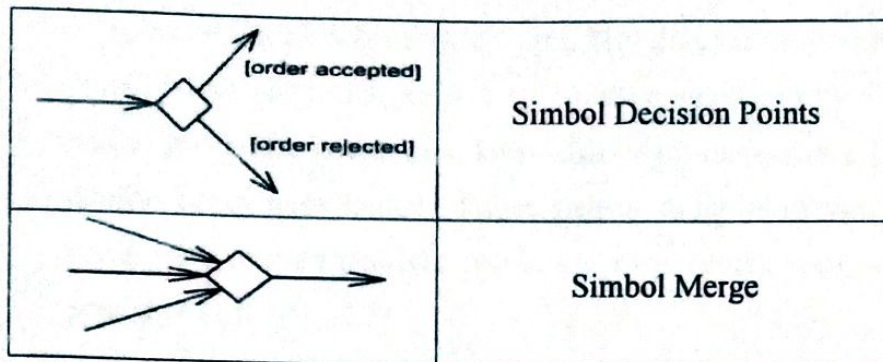
- g. *Message Sending*, suatu obyek mengirim sebuah pesan (message) kepada obyek lain untuk menjalankan sebuah operation dan obyek yang menerima akan memberikan respon untuk menjalankan operasi tersebut. (Munawar:31)
- h. *Encapsulation*, penggabungan dari beberapa atribut dan perilaku(items) menjadi satu unit. (Jeffery L. Whitten et al, 2004)
- i. *Aggregation*, suatu hubungan dimana satu kelas yang lebih besar berisi satu atau lebih bagian kelas yang lebih kecil. (Jeffery L. Whitten et al, 2004)
- j. *Association*, hubungan antar obyek yang saling membutuhkan. Hubungan ini bisa satu arah ataupun lebih dari satu arah. (Munawar, 2004)
- k. *Generalization*, suatu konsep dimana perilaku dan atribut yang umum dibagi ke dalam beberapa jenis kelas obyek yang dikelompokkan (diringkas) kedalam kelas mereka sendiri. (Jeffery L. Whitten et al, 2004)

Alat bantu yang digunakan dalam analisa berorientasi obyek dengan UML antara lain adalah:

a. Activity Diagram

Activity Diagram adalah teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. Activity diagram mempunyai peran seperti halnya flowchart, akan tetapi perbedaannya dengan flowchart adalah activity diagram bisa mendukung perilaku paralel sedangkan flowchart tidak bisa. (Munawar, 2005)

| | |
|---|--------------------|
|  | Simbol Start Point |
|  | Simbol End Point |
|  | Simbol Activities |
|  | Simbol Fork |
|  | Simbol Join |



Simbol – simbol Activity Diagram

b. Analisa Dokumen Keluaran

Analisa keluaran adalah analisa mengenai dokumen–dokumen keluaran yang dihasilkan dari sebuah sistem.

c. Analisa Dokumen Masukan

Analisa masukan adalah bagian dari pengumpulan informasi tentang sistem yang sedang berjalan. Tujuan analisa masukan adalah memahami prosedur berjalan.

d. Usecase Diagram

Use Case Diagram menggambarkan sebuah fungsionalitas yang diharapkan dari sebuah sistem dan bagaimana sistem berinteraksi dengan dunia luar. Yang ditekankan dalam Use Case Diagram adalah “apa” yang diperbuat sistem, dan bukan “bagaimana” sistem itu melakukannya.

Sebuah Use Case merepresentasikan sebuah interaksi antara actor dengan sistem. Use Case Diagram juga menjelaskan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (actor). Use Case merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, mengcreate sebuah daftar belanja, dan sebagainya. Secara umum Use Case Diagram terdiri dari :

- 1) Actor

Actor adalah sebuah peran yang bisa dimainkan oleh pengguna dalam interaksinya dengan sistem. Untuk mengidentifikasi actor harus ditentukan pembagian kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Actor dilukiskan dengan peran yang mereka mainkan dalam use case, seperti staff penjualan, pelanggan, dll.



Simbol Actor

2) Use Case

Use case menggambarkan perilaku, termasuk didalamnya interaksi antara actor dengan sistem.

Use case dibuat berdasarkan keperluan actor, merupakan “apa” yang dikerjakan sistem bukan “bagaimana” system mengerjakannya. Setiap use case harus diberi nama yang menyatakan apa hal yang dicapai dari hasil interaksinya dengan actor. Nama use case boleh terdiri dari beberapa kata dan tidak boleh ada dua use case yang memiliki nama yang sama.



Simbol Use Case

3) Relationship (Relasi) / Association (Asosiasi)

Asosiasi menggambarkan aliran data / informasi. Asosiasi/ relasi juga digunakan untuk menggambarkan bagaimana actor terlibat dalam use case. Relasi (*relationship*) digambarkan sebagai bentuk garis antara dua simbol dalam use case diagram.

Ada dua jenis relasi / asosiasi yang dapat timbul pada use case diagram, yaitu :

- a) Asosiasi antara Actor dan Use Case

Ujung panah pada association antara actor dan use case mengindikasikan siapa / apa yang meminta interaksi dan bukannya mengindikasikan aliran data. Sebaiknya gunakan garis tanpa panah untuk association antara actor dan use case. Association antar actor dan use case yang menggunakan panah terbuka untuk mengindikasikan bila actor berinteraksi secara pasif dengan sistem.

—————
Simbol Asosiasi

b) Asosiasi antara Use Case

Relasi antara use case dengan use case :

- (1) *Include*, menggambarkan suatu use case termasuk di dalam use case lain (diharuskan). Contohnya adalah pemanggilan sebuah fungsi program. Digambarkan dengan garis lurus berpanah dengan tulisan <<include>>.

—————→
Simbol Association Include

- (2) *Extend*, digunakan ketika hendak menggambarkan variasi pada kondisi perilaku normal dan menggunakan lebih banyak control form dan mendeklarasikan ekstension pada use case utama atau dengan kata lain adalah perluasan dari use case lain jika syarat atau kondisi terpenuhi. Digambarkan dengan garis lurus berpanah dengan tulisan <<extend>>.

↑
Simbol Association Extend

- (3) *Generalization / Inheritance* antar Use Case *Generalization* dipakai ketika ada sebuah perlakuan khusus (*single condition*) dan merupakan pola hubungan base-parent use case. Digambarkan dengan *generalization / inheritance* antar use case secara vertikal dengan inheriting use case dibawah base / parent use case.

e. **Deskripsi Usecase**

Deskripsi usecase digunakan untuk menjelaskan urutan langkah-langkah yang menerangkan antara pengguna dengan sistem yang mendeskripsikan urutan kejadian, dan menjelaskan secara rinci mengenai usecase diagram. Urutan langkah-langkah yang menerangkan antara penggunaan sistem disebut juga skenario. Setiap skenario mendeskripsikan urutan kejadian. Setiap urutan diinisialisasi oleh orang, sistem yang lain, perangkat keras atau urutan waktu. Dengan demikian secara singkat bisa dikatakan deskripsi usecase adalah serangkaian skenario yang digabungkan bersama-sama oleh tujuan umum pengguna (aktor).

2.4.3. Perancangan Sistem Berorientasi Obyek

Perancangan berorientasi obyek merupakan tahap lanjutan setelah analisa berorientasi obyek, perancangan berorientasi obyek adalah suatu pendekatan yang digunakan untuk menspesifikasi kebutuhan-kebutuhan sistem dengan mengkolaborasikan obyek-obyek, atribut-atribut, dan method-method yang ada. (Ariesto Hadi Sutopo, 2002).


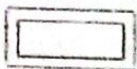



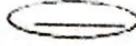

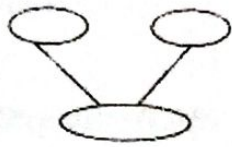

Alat-alat yang digunakan dalam tahap perancangan sistem berorientasi obyek adalah sebagai berikut:

a. **ERD**

ERD adalah suatu strategi untuk membangun sebuah model data konseptual yang menggunakan diagram yang fokus kepada kelas entitas, relasi, dan atribut. (Riccardi, 2001)

ERD terdiri dari :

- 1) *Entity Class* (kelas entitas), karakteristik umum yang mewakili suatu kumpulan dari entitas. (Riccardi, 2001)
- 2) *Relationship*, suatu penghubung antara dua atau lebih entitas. Simbol yang digambarkan dengan bentuk wajik disebut dengan *relationship type* dimana sebuah kata yang ada di dalam gambar wajik tersebut mengartikan bagaimana dua entitas dapat saling berhubungan. (Riccardi, 2001)
- 3) *Attribute*, sebuah karakteristik yang menggambarkan sebuah entitas. Atribut yang digaris bawah menunjukkan bahwa atribut tersebut adalah sebuah *key*. (Riccardi, 2001)

| <u>Notasi</u> | <u>Arti</u> |
|--|-----------------------------|
| 1.  | 1. Entity |
| 2.  | 2. Weak Entity |
| 3.  | 3. Relationship |
| 4.  | 4. Identifying Relationship |
| 5.  | 5. Atribut |
| 6.  | 6. Atribut Primary Key |
| 7.  | 7. Atribut Multivalued |
| 8.  | 8. Atribut Composite |
| 9.  | 9. Atribut Derivatif |

Simbol - simbol diagram ERD

- 4) Setiap entitas harus memiliki atribut yang unik untuk pengidentifikasian, atribut yang dimaksud disebut dengan key, Jenis-jenis key adalah:
 - a) *Primary Key*, yaitu key yang paling umum digunakan untuk mengidentifikasi secara unik setiap instansi dari entitas. (Whitten, 2004)
 - b) *Secondary Key*, yaitu suatu key yang tidak terpilih untuk dijadikan primary key. Juga disebut sebagai alternate key. (Whitten, 2004)
 - c) *Composite key*, suatu kelompok atribut yang dapat dengan unik mengidentifikasi suatu instansi dari suatu entitas. (Whitten, 2004)
 - d) *Foreign key*, suatu *primary key* dari suatu entitas yang digunakan di entitas yang lain untuk mengidentifikasi instansi dari suatu hubungan (relationship). (Whitten, 2004)
- 5) *Cardinality Constraint*, suatu pembatasan di dalam cardinalitas dari suatu hubungan. batasan yang khas adalah to-one, di mana suatu entitas dapat dihubungkan tidak lebih kepada satu instansi pada entitas yang terhubung, dan to-many, di mana suatu entitas dapat dihubungkan dengan jumlah tak terbatas kepada instansi dalam entitas yang terhubung. cardinality constraint dapat menetapkan suatu jumlah kejadian minimum atau maksimum dalam entitas yang saling terkait. (Riccardi, 2001)

b. Logical Record Structure (LRS)

LRS adalah digambarkan oleh kotak persegi panjang dan dengan nama yang unik. File record pada LRS ditempatkan dalam kotak. LRS terdiri dari link-link diantar tipe record lainnya, banyaknya link dari LRS yang diberi nama oleh field-field yang kelihatan pada kedua link tipe record.

Normalisasi merupakan proses pengkelompokan data element menjadi tabel-tabel yang menunjukan entitas dan relasinya. Normalisasi digunakan untuk menghindarkan kemungkinan kesulitan penyajian data.

Berikut ini adalah tahapan-tahapan dalam normalisasi:

- 1) Normalisasi bentuk pertama (1NF)
Menghilangkan beberapa group elemen yang berulang agar menjadi satu harga tunggal yang berinteraksi diantara setiap baris dan kolom pada suatu tabel.
- 2) Normalisasi bentuk kedua (2NF)
Menghilangkan beberapa bagian ketergantungan fungsional atau dengan kata lain apa bila sudah berada pada 1NF dan setiap atribut yang bukan key bergantung pada primary key.
- 3) Normalisasi bentuk ketiga (3NF)
Menghilangkan beberapa ketergantungan transitif (*transitiv dependence*) atau dengan kata lain apabila sudah berada 2NF dan setiap atribut yang berupa key tidak tergantung atribut lain kecuali primary key.
 - a) Normalisasi BCNF (*Boice Codd Normal Form*)
Pada keadaan ini sudah tidak ada lagi primary key yang tergantung pada primary key.
 - b) Normalisasi bentuk keempat
Menghilangkan beberapa ketergantungan pada banyak harga.
 - c) Normalisasi bentuk kelima
Menghilangkan anomaly yang terjadi akibat dekomposisi relasi yang tidak terpakai untuk membentuk kembali relasi semula.

c. Tabel/Relasi

Tabel adalah koleksi objek yang terdiri dari sekumpulan elemen yang diorganisasi secara kontinue, artinya memori yang dialokasi antara satu elemen dengan elemen yang lainnya mempunyai adress yang berurutan.

Pada tabel, pengertian perlu dipahami adalah:

- 1) Keseluruhan tabel (sebagai koleksi) adalah kontainer yang menampung seluruh elemen.
- 2) Indek tabel, yang menunjukan *adres* dari sebuah element.
- 3) Element tabel, yang dapat dipacu melalui indeknya, bertipe tertentu yang sudah terdefinisi.
- 4) Seluruh element tabel bertipe: "sama". Dengan catatan: beberapa bahasa pemograman memungkinkan pendefinisian tabel dengan elemen generik, tapi pada saat diinstansiasi, harus diinstansiasi dengan tipe sama.

d. Spesifikasi Basis Data

Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lain dan tersimpan diluar komputer serta digunakan perangkat lunak (*software*) tertentu untuk memanipulasinya.

Sedangkan sistem berbasis data adalah suatu sistem penyusunan dan pengelolaan record-record dengan menggunakan komputer dengan tujuan untuk menyimpan atau merekam serta melihat data operasional lengkap pada sebuah organisasi, sehingga mampu menyediakan informasi yang diperlukan untuk kepentingan proses pengambilan keputusan.

e. Rancangan Dokumen Keluaran

Rancangan keluaran merupakan informasi yang akan dihasilkan dari keluaran sistem yang dirancang.

f. Rancangan Dokumen Masukan

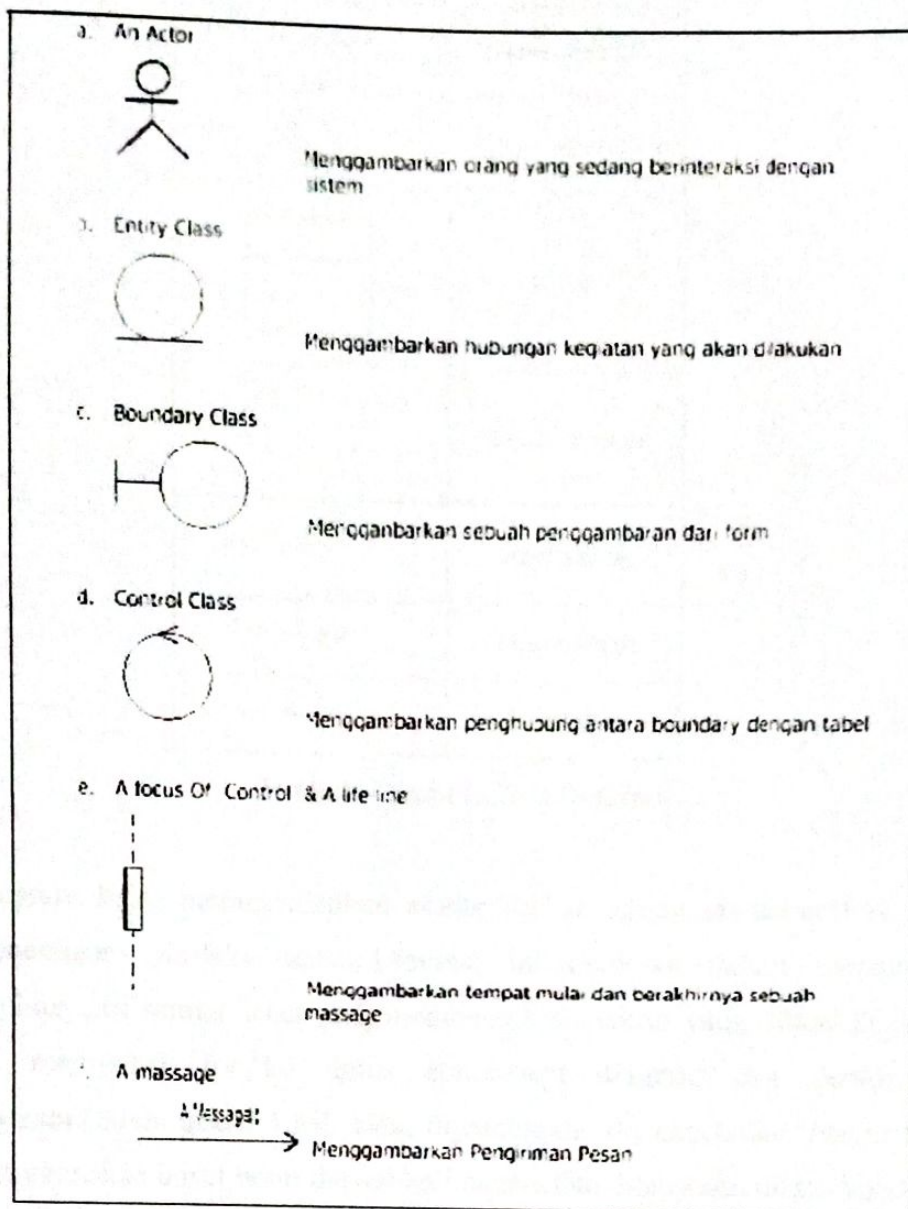
Rancangan masukan merupakan data yang dibutuhkan untuk menjadi masukan sistem yang dirancang.

g. Rancangan Layar Program

Rancangan tampilan merupakan bentuk tampilan sistem layar komputer sebagai antar muka dengan pemakai yang akan dihasilkan dari sistem yang dirancang.

h. Sequence Diagram








Sequence diagram menerangkan objek yang disusun dalam urutan tertentu. Urutan waktu yang dimaksud adalah urutan yang dilakukan seseorang aktor dalam menjalankan sistem. Diagram ini secara khusus berasosiasi dengan use case sequence diagram memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu dalam use case diagram ini sebaiknya digunakan diawal tahap perancangan atau analisis karena kesederhanaannya dan mudah dimengerti. Sequence diagram menunjukkan bagaimana data operasi dilakukan pesan apa yang dikirim dan kapan sequence diagram didasarkan atas kelas diagram yang sudah dibuat hanya saja kelas diagram yang sudah dibuat belum menyertakan kelas boundary dan kelas control, maka sebelum membuat sequence diagram perlu dibuat class boundary dan kelas countrol terlebih dahulu.



Simbol – simbol Squence Diagram

i. Class Diagram

Class diagram sangat membantu dalam visualisasi struktur kelas dari suatu sistem. Hal ini disebabkan karena class adalah diskripsi kelompok obyek-obyek dengan properti, perilaku (operasi) dan relasi yang sama. Disamping itu class diagram bisa memberikan pandangan global atas sebuah sistem. Hal tersebut tercermin dari class-class yang ada yang relasinya satu dengan yang lainnya. Itulah sebabnya class diagram menjadi diagram paling populer di UML.

| | |
|--|-------------------------|
|  | <i>Generalization</i> |
|  | <i>Nary Association</i> |
|  | <i>Class</i> |
|  | <i>Collaboration</i> |
|  | <i>Realization</i> |
|  | <i>Dependency</i> |
|  | <i>Association</i> |

Simbol – simbol Class Diagram

Diagram kelas memperlihatkan aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Diagram ini berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur yang dibuat. Diagram ini merupakan fondasi untuk component diagram dan deployment diagram. Dalam notasi UML class digambarkan dengan kotak. Nama class menggunakan huruf besar diawal kalimatnya dan diletakkan diatas kotak.

1) *Association*

Association/asosiasi adalah class-class yang terhubungkan satu sama lain secara konseptual. Setiap association mempunyai dua association end. Sebuah association end juga memiliki "*multiplicity*". *Multiplicity* menunjukkan beberapa banyak obyek yang berpartisipasi dalam suatu relasi. Secara umum, multiplicity menunjukkan batasan terendah dan tertinggi untuk obyek-obyek yang berpartisipasi. Multiplicity yang paling umum digunakan adalah 1, *, dan 0..1.

Langkah-langkah transformasi dari conceptual data model ke tabel relasi adalah sebagai berikut :

- a) Jika hubungan yang terjadi antar class adalah 1 to 1 (*one to one*) maka atribut dari relationship set diambil dan dimasukkan ke entitas yang lebih membutuhkan.
- b) Jika hubungan yang terjadi antar class adalah 1 to 0..1 (*one to zero one*) maka atribut dari relationship set digabung ke entitas yang memiliki multiplicity 0..1.
- c) Jika hubungan yang terjadi antar class adalah 1 to * (*one to many*) maka atribut dari relationship set digabung dengan set entitas yang memiliki multiplicity banyak (*many*).

2) Atribut (Attribute)

Attribute adalah properti dari sebuah class. Attribute ini melukiskan batas nilai yang mungkin ada pada obyek dari class.

3) Operasi

Operasi adalah sesuatu yang bisa dilakukan oleh sebuah class atau yang ada (atau class lain) dapat dilakukan untuk sebuah class.

2.5. Teori Pendukung

Terdapat beberapa teori pendukung seperti dibawah ini, yaitu

2.5.1. Teori Pengelola Proyek

Proyek adalah upaya temporer untuk menghasilkan produk, jasa, atau hasil yang tertentu/unik. Proyek bersifat temporer artinya waktu berlangsungnya dibatasi, ada awal dan ada akhir untuk pekerjaan yang dilakukan dan tim yang dibentuk. Proyek menghasilkan yang unik, berarti hasil dari proyek merupakan suatu entitas baru yang memiliki karakteristik yang berbeda dengan hasil yang sudah ada.

- a. Definisi Manajemen Proyek menurut PMBOK (*Project Management Body of Knowledge*) adalah aplikasi dari pengetahuan, keahlian, alat-alat, dan teknik untuk melaksanakan aktivitas sesuai dengan kebutuhan proyek.
- b. *Project Execution Plan* (PEP), Sebuah rencana eksekusi suatu proyek sangat erat kaitannya dengan estimasi biaya, dimana keduanya saling bergantung dan tidak akan terpenuhi keduanya secara total jika satu diantara keduanya tidak terselesaikan.
- c. *Objective proyek* (tujuan proyek),
- d. *Stakeholder* dapat diartikan sebagai segenap pihak yang terkait dengan isu dan permasalahan yang sedang diangkat. Misalnya bilamana isu perikanan, maka stakeholder dalam hal ini adalah pihak-pihak yang terkait dengan isu perikanan, seperti nelayan, masyarakat pesisir, pemilik kapal, anak buah kapal, pedagang ikan, pengolah ikan, pembudidaya ikan, pemerintah, pihak swasta di bidang perikanan, dan sebagainya.
- e. *Deliverable* adalah produk yang diproduksi sebagai bagian dari proyek seperti perangkat keras, dokumen perencanaan, atau rapat.
- f. Jadwal proyek, Penjadwalan proyek adalah kegiatan menetapkan jangka waktu kegiatan proyek yang harus diselesaikan, bahan baku, tenaga kerja serta waktu yang dibutuhkan oleh setiap aktivitas.
- g. *Work Breakdown Structure* (WBS), adalah cara pengorganisasian proyek menjadi bagan/struktur pelaporan yang bersifat hirarkis. WBS berfungsi untuk melakukan breakdown atau memecahkan masalah setiap proses pekerjaan menjadi lebih baik dan sempurna. Prinsip dasar dari *Work Breakdown Structure* (WBS) adalah pemecahan atau pembagian pekerjaan ke dalam bagian-bagian yang lebih kecil.
- h. *Milestone*, adalah suatu bagian item pekerjaan yang dibuat seolah-olah menjadi temporary finish atau selesai sementara atas sekelompok atau serangkaian pekerjaan-pekerjaan yang menjadi bagian dari *schedule* besar. Item pekerjaan yang dijadikan *milestone* haruslah item pekerjaan yang dianggap menjadi bagian penting sebelum melanjutkan pekerjaan berikutnya atau berpengaruh atas kelangsungan pekerjaan berikutnya.

- i. Rencana Anggaran Biaya (RAB), adalah perhitungan banyaknya biaya yang diperlukan untuk bahan dan upah, serta biaya-biaya lain yang berhubungan dengan pelaksanaan bangunan atau proyek. Anggaran biaya merupakan harga dari bahan bangunan yang dihitung dengan teliti, cermat dan memenuhi syarat. Anggaran biaya pada bangunan yang sama akan berbeda-beda di masing-masing daerah, disebabkan karena perbedaan harga bahan dan upah tenaga kerja.
- j. Organisasi matriks adalah suatu usaha untuk menggabungkan keuntungan dari struktur fungsional murni dan struktur produk organisasi. Dalam organisasi matriks, setiap manajer proyek melaporkan secara langsung kepada wakil presiden dan manajer umum. Karena setiap proyek merupakan profit center yang potensial, kekuasaan dan otoritas yang digunakan oleh manajer proyek datang langsung dari manajer umum.
- k. Analisa resiko (*Project risk*), menjelaskan proses-proses yang berhubungan dengan pengidentifikasian resiko, kuantifikasi resiko, penyusunan penanggulangan resiko dan pengendalian penanggulangan resiko.

2.5.2. Teori Sistem Informasi Pengarsipan

a. Tujuan Pendataan

Menurut Lamuddin (2009:5), Tujuan dari pendataan adalah untuk memperoleh data yang akurat mengenai catatan keluar dan masuknya surat.

Data ini digunakan Dinas Sosial dan Tenaga Kerja untuk :

- 1) Pencatatan secara akurat dan detail tentang masuk dan keluarnya surat di lingkungan Dinas sosial Dan Tenaga Kerja.
- 2) Langkah awal untuk menindak lanjuti sebuah kebijakan.

b. Pengertian Surat

Menurut Lamuddin (2009:5), Secara Umum, Surat adalah alat untuk menyampaikan suatu maksud secara tertulis jika dilihat dari sisi

pemakainnya surat dapat digolongkan menjadi 4 macam yaitu surat dinas, surat niaga, surat sosial dan surat pribadi.

1) Surat Masuk

Surat masuk adalah semua surat dinas yang diterima oleh suatu instansi pemerintah. Untuk memudahkan pengawasan dan pengendalian, penerimaan, pengelolaan surat masuk dipusatkan di bagian tata usaha atau sekretariat pimpinan.

2) Surat Keluar

Seperti halnya surat masuk maka pengelolaan surat keluar dilakukan oleh bagian tata usaha atau sekretariat. Artinya, bagian tata usaha yang berhak mengolah, memeriksa, menyerahkan pada pimpinan untuk ditandatangani, sekaligus mengirimkannya kepada instansi lain.

3) Pendistribusian Surat

Dalam suatu instansi besar pengiriman surat juga biasanya disentralisir yaitu dilakukan oleh staf bagian pengiriman surat pada bagian Tata Usaha. Setelah surat ditandatangani oleh pimpinan, surat dicatat dalam sebuah buku agenda surat keluar.

c. Pengarsipan Surat

1) Pengertian Arsip

Arsip ialah suatu kumpulan catatan yang disimpan secara sistematis karena mempunyai suatu kegunaan agar setiap kali diperlukan dapat secara mudah dan cepat ditemukan. Istilah lain dari arsip ialah segala sesuatu yang berkenaan dengan penyimpanan segala surat yang meliputi korespondensi, surat-surat instruksi, surat edaran, akte. Atau singkatnya arsip adalah pembendaharaan surat-surat termasuk di dalamnya surat-surat yang telah dijadikan buku atau kitab, baik mendani benda-benda surat maupun mengenai tempat penyimpanan surat-surat.

Kata arsip berasal dari bahasa Belanda "*Archief*" yang artinya simpanan dan dalam bahasa Inggris "*File*". Dalam bahasa

latin "*Archivum*" yang berarti penyimpanan (gedung penyimpanan) surat-surat, peraturan-peraturan, undang-undang, dll.

2) Sistem Kearsipan

Ada dua macam sistem kearsipan

- (a). Berdasarkan "nomor surat" ialah dimana surat disimpan berdasarkan urusannya masing-masing seperti: surat-surat yang berhubungan dengan kependudukan diarsipkan sesuai dengan arsip kependudukan
- (b). Berdasarkan perihal ialah kearsipan dimana penyimpanan surat-surat itu berdasarkan perihal surat.

3) Bagian – bagian Arsip

- (a). *Arsip Permanent* ialah arsip surat-surat penting yang ada hubungannya dengan kantor atau perusahaan yang bersangkutan.
- (b). *Arsip Active* ialah surat-surat yang setiap waktu masih dibutuhkan.
- (c). *Arsip inactive* ialah arsip dari surat yang sudah jarang dipakai, umumnya surat-surat yang lebih dari satu tahun.
- (d). *Arsip mati* ialah arsip dari surat yang sudah tidak pernah dipakai lagi (umumnya surat-surat yang lebih dari 5 tahun).

Masalah kearsipan sangat penting sebab pada kantor besar umumnya diadakan suatu tempat khusus yang merupakan pusat penyimpanan arsip dengan staff khusus, dimana surat seluruh bagian kantor ditetapkan menjadi satu.

Keuntungan pemusatan arsip sebagai berikut:

- (a). Pertanggungjawaban seluruh arsip dipikulkan kepada seorang pegawai atau staf
- (b). Arsip dapat dipakai oleh semua bagian pada tempat yang tertentu.

- (c). Surat-surat itu tidak tersebar pada masing-masing bagian yang kadang-kadang bisa hilang terlupakan dan lain sebagainya.
- (d). Segala rahasia kantor, perusahaan atau suatu organisasi dapat **terjamin.**