

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

Untuk mendefinisikan suatu sistem pakar harus mempunyai konsep dasar untuk memperkuat definisinya. Sistem mempunyai suatu konsep yang mendasari sebuah pengertian-pengertian yang dikemukakan oleh berbagai pakar untuk mendefinisikan dari suatu sistem itu sendiri.

2.1.1. Definisi Sistem

Pengertian tentang sistem menurut Mulyani (2016:2) “Sistem merupakan sekumpulan sub sistem, komponen ataupun element yang saling bekerja sama dengan tujuan yang sama untuk menghasilkan hasil output yang sudah ditentukan sebelumnya”.

Menurut Djahir dan Pratita (2015:7) menyatakan bahwa “Sistem dikelompokkan menjadi dua bagian yang menekankan pada prosedurnya dan ada yang menekankan pada elemennya. Kedua kelompok ini adalah benar dan tidak bertentangan, yang berbeda adalah cara pendekatannya”.

Menurut Tyoso (2016:1) “Sistem merupakan suatu kumpulan dari komponen-komponen yang membentuk suatu kesatuan”.

Sedangkan menurut Hutahaean (2015:2), “Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran tertentu”.

Dari keempat pandangan diatas penulis dapat menyimpulkan bahwa sistem bisa diartikan sebagai suatu kumpulan sub sistem dari komponen atau element unsur-unsur yang saling berkaitan dan mempunyai tujuan yang sama.

2.1.2. Karakteristik Sistem

Menurut Hutahaean (2015:3), suatu sistem memiliki karakteristik atau sifat-sifat tertentu yaitu sebagai berikut:

1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama untuk membentuk suatu kesatuan. Terdiri dari komponen yang berupa subsistem atau bagian-bagian dari sistem.

2. Batasan Sistem (*boundary*)

Merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*environment*)

Diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan dapat bersifat menguntungkan yang harus tetap dijaga dan yang merugikan harus dijaga dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung Sistem (*interface*)

Media penghubung antara satu subsistem dengan subsistem lainnya. Yang memungkinkan sumber-sumber daya mengalir dari subsistem ke subsistem lain. Keluaran dari satu subsistem akan menjadi masukan bagi subsistem lainnya.

5. Masukan Sistem (*input*)

Energi yang dimasukkan kedalam sistem, yang berupa perawatan (*maintenance input*), dan masuknya sinyal (*signal input*). Maintenance input adalah energi yang dimasukkan agar sistem dapat beroperasi. Signal input adalah energi yang diproses untuk didapatkan keluaran.

6. Keluaran Sistem (*output*)

Energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

7. Pengolah Sistem

Suatu sistem menjadi bagian pengolah yang akan merubah masukan menjadi keluaran.

8. Sasaran Sistem

Setiap sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Sasaran dari sistem sangat menentukan input yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.

2.1.3. Klasifikasi Sistem

Beberapa aspek dari sistem ini membuat pengguna sistem dapat mengklasifikasikan sistem yang relevan sesuai dengan arah pandang pengguna sistem. Klasifikasi Sistem Menurut Tyoso (2016:5) terdiri dari:

1. Sistem Alamiah

Sistem alamiah (*natural system*) muncul secara alamiah tanpa campur tangan manusia.

2. Sistem Tiruan

Sistem tiruan (*artificial system*) diciptakan untuk mendukung tujuan tertentu.

3. Sistem Deterministik

Sistem deterministik (*deterministic system*), bekerjanya sistem ini dapat diramalkan sebelumnya. Masukan untuk sistem ini secara pasti menentukan jenis keluarannya.

4. Sistem Probabilistik

Sistem probabilistik (*probabilistic system*) dapat dilacak hanya dengan menggunakan nilai distribusi probabilitas, selalu ada nilai ketidakpastian yang sesungguhnya pada sembarang waktu.

5. Sistem Tertutup

Sistem tertutup (*closed system*), pada sistem ini tidak terjadi pertukaran atau penggunaan sumber daya dengan atau dari lingkungannya, mengingat sistem ini tidak menggunakan input dari lingkungannya, maka output dari sistem ini tidak berkaitan dengan lingkungannya pula.

6. Sistem Terbuka

Sistem terbuka (*opened system*) menggunakan sumber daya dari lingkungannya sehingga keluarannya berkaitan dengan lingkungannya juga.

2.1.4. Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building block*). Adapun penjelasan dari blok bangunan Hutahaean (2015:13) yaitu:

1. Blok Masukkan (*input block*)

Blok masukan merupakan blok yang bertugas dalam input data agar masuk ke dalam sistem informasi. Blok masukan bertugas dalam merekam data yang akan dimasukkan, biasanya berupa dokumen-dokumen dasar.

2. Blok Model (*model block*)

Blok model terbentuk dari kombinasi prosedur, logika dan model matematik yang memproses data input dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran (*output block*)

Sistem informasi menghasilkan keluaran (*output*) yaitu informasi yang berkualitas dan berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi (*technology block*)

Teknologi digunakan merupakan kotak alat dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran berupa informasi dan membantu pengendalian dari sistem secara menyeluruh. Blok teknologi perangkat lunak (*software*) dan perangkat keras (*hardware*) yang dioperasikan oleh teknisi (*brainware*).

5. Blok Basis Data (*database block*)

Basis data (*database*) merupakan media untuk menyimpan data yang saling berhubungan satu sama lainnya, tersimpan di perangkat keras komputer dan dapat dipergunakan kembali, diperlukan perangkat lunak untuk memanipulasinya.

6. Blok Kendali (*control block*)

Sistem informasi memiliki kontrol kendali untuk menanggulangi gangguan-gangguan terhadap sistem apabila terlanjur terjadi kesalahan maka dapat langsung diantisipasi atau diatasi.

2.1.5. Pengertian Informasi

Pengertian informasi menurut Djahir dan Pratita (2015:10), “Informasi adalah hasil dari pengolahan data yang akan menjadi bentuk lebih berguna bagi penerimanya yang menggambarkan suatu kejadian-kejadian nyata untuk mengambil sebuah keputusan”.

Sedangkan menurut Mulyani (2016:14), “Informasi merupakan data yang sudah diolah dan ditunjukkan kepada seseorang, organisasi ataupun siapa saja yang membutuhkan”.

Dari kesimpulan diatas penulis menyimpulkan dapat diartikan informasi adalah data yang diperoleh dari peristiwa-peristiwa tertentu yang telah dikumpulkan atau diolah menjadi bentuk yang penting bagi pengambilan keputusan dan diberikan kepada penerimannya.

2.1.6. Pengertian Sistem Informasi

Menurut Djahir dan Pratita (2015:13), “Sistem informasi memberikan sebuah informasi yang diterima dan diperlukan oleh manajemen sesuai dengan posisinya”.

Menurut Mulyani (2016:19) menyatakan bahwa sistem informasi manajemen “merupakan sistem informasi yang sudah terkomputerisasi yang bekerja karena adanya interaksi manusia dan komputer”.

Dari kesimpulan diatas penulis menyimpulkan sistem informasi bisa dikatakan sebagai suatu sistem kerja yang kegiatannya ditunjukan untuk pengolahan dari sub sistem yang saling berhubungan satu sama lain untuk mencapai suatu tujuan bersama mengolah data menjadi informasi yang berguna.

2.1.7. Pengertian Persediaan

Menurut Prasetyo (2006:65) dalam Tamodia (2013:23) mengemukakan bahwa “Persediaan adalah suatu aktivitas yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam satu periode usaha yang normal, termasuk barang yang dalam pengerjaan/proses produksi menunggu masa penggunaannya pada proses produksi”.

Menurut Horngren (2008:44) dalam Sambuaga (2013:1699) persediaan merupakan perusahaan sektor perdagangan membeli produk berwujud dan kemudian menjualnya tanpa melakukan perubahan bentuk dasar perusahaan jenis ini hanya memiliki 1 jenis persediaan, yang merupakan produk dalam bentuk asli ketika dibeli yang disebut persediaan barang dagang.

Menurut Stice dan Skousen (2009:571) dalam Anwar dan Karamoy (2014:1298) persediaan adalah istilah yang diberikan untuk aktiva yang akan dijual dalam kegiatan normal perusahaan atau aktiva yang dimasukan secara langsung atau tidak langsung ke dalam barang yang diproduksi dan kemudian dijual.

Dari kesimpulan diatas penulis menyimpulkan bahwa pengertian persediaan adalah barang-barang yang dimiliki dan akan dijual kembali atau digunakan pada proses produksi yang dipakai untuk keperluan non produksi dalam siklus kegiatan yang normal.

2.1.8. Pengertian Data

Menurut Lubis (2016:1) menyimpulkan data adalah “Fakta-fakta yang menggambarkan suatu kejadian yang sebenarnya terjadi pada waktu tertentu”.

Menurut Jogiyanto (2017:1.29) mengemukakan bahwa “data merupakan bentuk jamak dari bentuk tunggal datum atau data-item. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian (*event*) adalah sesuatu yang terjadi pada saat tertentu”.

Dari kesimpulan diatas penulis menyimpulkan bahwa data adalah suatu file yang berupa karakter , tulisan dan gambar atau fakta yang dapat diolah menjadi sebuah informasi.

2.1.9. Pengertian Basis Data (*Database*)

Menurut Rosa dan Shalahuddin (2016:43) mengemukakan bahwa basis data adalah “sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan yang pada intinya basisdata adalah media yang menyimpan data agar dapat diakses dengan mudah dan cepat”.

Menurut Pahlevi (2013:1) basis data adalah “sekumpulan data yang saling berhubungan secara logis beserta deskripsinya, yang digunakan secara bersama-sama dan dirancang untuk memenuhi kebutuhan informasi disuatu tempat”.

Menurut Lubis (2016:3) basis data adalah “Suatu sistem penyusunan dan pengelolaan record-record dengan menggunakan komputer, dengan tujuan untuk menyimpan atau merekam serta memelihara data secara lengkap pada sebuah organisasi/perusahaan, sehingga mampu menyediakan informasi yang optimal yang diperlukan oleh pemakai untuk kepentingan proses dalam pengambilan suatu keputusan”.

Dari kesimpulan diatas penulis menyimpulkan BasisData adalah sekumpulan file yang saling berhubungan yang menyimpan data dan tersimpan dalam sebuah media penyimpanan.

2.1.10. Pengertian Java

Menurut Ashari (2014:66) mengemukakan bahwa java adalah bahasa pemrograman serbaguna. Java dapat digunakan untuk membuat suatu program sebagaimana kita membuatnya dengan bahasa seperti pascal atau C++ . Yang lebih menarik java dapat mendukung sumber daya internet yang saat ini populer, yaitu World Wide Web atau yang sering disebut web saja. Java juga mendukung aplikasi client/server baik dalam jaringan lokal (LAN) maupun berskala luas (WAN).

Menurut Rosa dan Shalahuddin (2016:101) mengemukakan bahwa java “merupakan bahasa pemrograman yang paling konsisten dalam mengimplementasikan paradigma pemrograman berorientasi objek”.

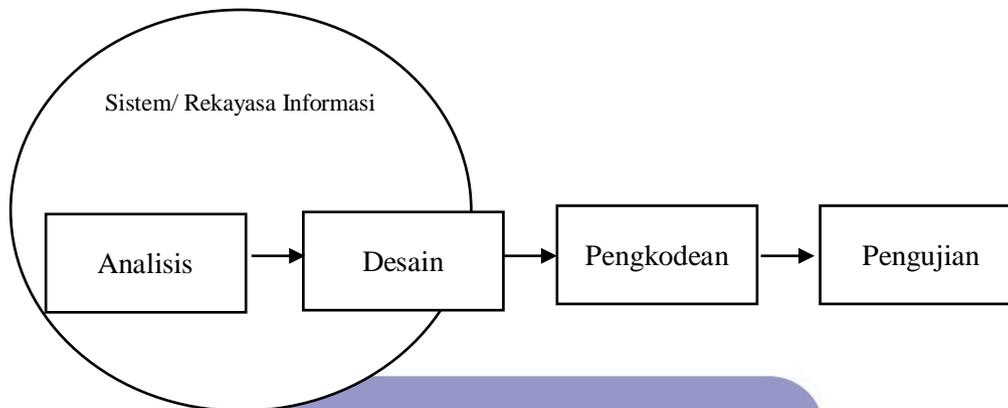
Menurut Rusmayanti (2014:36) mengemukakan bahwa “java merupakan bahasa pemrograman berorientasi objek dan bebas platform, dikembangkan oleh SUN Micro System dengan jumlah keunggulan yang memungkinkan java dijadikan sebagai bahasa pengembang enterprise. Java merupakan bahasa yang powerfull yang bisa digunakan dalam hampir semua bentuk pengembangan software. Anda dapat menggunakan java dalam membuat game, aplikasi desktop, aplikasi web, aplikasi enterprise, aplikasi jaringan, dan lain-lain. Yang menarik adalah bahwa java bisa digunakan untuk membuat laporan yang dapat berjalan di atas HP, PDA, dan peralatan lain yang dilengkapi dengan java Virtual Machine JVM”.

Dari kesimpulan diatas penulis menyimpulkan Java adalah bahasa pemrograman yang bisa dijalankan di komputer maupun telepon genggam yang bersifat terstruktur dan berorientasi objek (*oop*).

2.1.11. Model Pengembangan Perangkat Lunak

Menurut Rosa Dan Shalahuddin (2016:28), “model SDLC air terjun sering disebut juga model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*)”. Model air terjun ini menyediakan pendekatan alur hidup dengan tahapan-tahapan waterfall Rossa dan Shalahuddin meliputi analisis, desain,

pengodean, pengujian, dan tahap pendukung (*support*) atau pemeliharaan (*maintenance*).”



Sumber: Rosa dan Shalahuddin (2016:28)

Gambar II.1. Model Waterfall

1. Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intersif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk di dokumentasikan.

2. Desain

Tahapan berikutnya adalah desain perangkat lunak yang meliputi proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi

sebuah program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan Kode Program

Proses ini ditranslasikan dari desain ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau Pemeliharaan (*maintenance*)

Pada tahap ini tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.2. Teori Pendukung

Dalam penulisan Tugas Akhir ini penulis menggunakan peralatan pendukung (*Tools System*) sebagai alat bantu dalam menyediakan Tugas Akhir, adapun peralatan yang digunakan adalah:

2.2.1. Pengertian *Entity Relationship Diagram (ERD)*

Menurut Rosa dan Shalahuddin (2016:53) "*Entity Relationship Diagram* atau ERD merupakan bentuk paling awal dalam melakukan perancangan basis data relasional". Jika menggunakan OODBMS maka perancangan ERD tidak perlu dilakukan.

Menurut Mulyati dkk (2015:37) "*Entity Relationship Diagram (ERD)* adalah perancangan database yang menggambarkan hubungan antara *entity* yang terdapat dalam sistem". ERD juga digunakan untuk merancang pembangunan database. Database tersebut digunakan untuk menyimpan data yang menunjang program sistem.

Dari kesimpulan diatas penulis menyimpulkan bahwa *Entity Relationship Diagram (ERD)* adalah hubungan antar data dalam basis data berdasarkan objek-objek data yang mempunyai hubungan antar relasi.

2.2.2. Pengertian *Logical Record Structure* (LRS)

Menurut Tabrani (2014:33-41) dalam Kuryanti (2016) “*Logical Record Structure* (LRS) dibentuk dengan nomor dari tipe record”. Beberapa tipe record digambarkan oleh kotak persegi panjang dan dengan nama yang unik. Perbedaan LRS dengan E-R diagram adalah nama tipe record berada diluar kotak field tipe record ditempatkan. LRS terdiri dari link-link diantara tipe record. Link ini menunjukkan arah dari satu tipe record field-field yang kelihatan pada kedua link tipe record. Penggambaran LRS mulai dengan menggunakan model yang dimengerti. Dua metode yang dapat digunakan, dimulai dengan hubungan kedua model yang dapat dikonversikan ke LRS, metode yang lain dimulai dengan ER-Diagram dan langsung dikonversikan ke LRS.

Menurut Marini (2015:107) “LRS digambarkan oleh kotak persegi panjang dan dengan nama yang unik. *File record* pada LRS ditempatkan dalam kotak. LRS terdiri dari *link-link* diantara *tipe record* lainnya, banyak *link* dan LRS yang diberi nama oleh *field-field* yang kelihatan pada kedua *link tipe record*”.

Dari kesimpulan diatas penulis menyimpulkan bahwa LRS (*Logical Record Structure*) terdiri dari tipe-tipe record yang berbentuk kotak persegi panjang dan dengan nama yang unik.

2.2.3. *Unified Modeling Language* (UML)

Menurut Mulyani (2016:42) “UML (*Unified Modeling Language*) adalah sebuah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem”.

Menurut Rosa dan Shalahuddin (2016:133) “UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”.

Secara konsep dasar UML memiliki banyak diagram yang digunakan untuk melakukan pemodelan sistem kedalam notasi grafis. Akan tetapi dalam hal ini yang akan dibahas menurut Rossa dan Shalahuddin sebanyak empat diagram yaitu :

1. Activity Diagram

Diagram aktivitas atau *activity diagram* adalah diagram yang digunakan untuk menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

- a. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
- b. Urutan atau pengelompokan tampilan dari sistem atau user interface dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan
- c. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya
- d. Rancangan menu yang ditampilkan pada perangkat lunak

2. Use Case Diagram

Diagram use case merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case

digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan use case.

- a. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
- b. Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

3. Class Diagram

Diagram kelas atau *class diagram* yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode operasi.

- a. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas
- b. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas

Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan didalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sehingga pembuat perangkat lunak atau *programmer* dapat membuat kelas-kelas didalam program perangkat lunak sesuai

dengan perancangan diagram kelas. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

1) Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

2) Kelas yang menangani tampilan sistem (*view*)

Kelas yang mendefinisikan dan mengatur tampilan ke pemakai.

3) Kelas yang diambil dari pendefinisian *use case* (*controller*)

Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

4) Kelas yang diambil dari pendefinisian data (*model*)

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan kedalam basis data. Semua tabel yang dibuat di basis data dapat dijadikan kelas, namun untuk tabel dari hasil relasi atau atribut multivalued pada ERD dapat dijadikan kelas tersendiri dapat juga tidak asalkan pengaksesannya dapat dipertanggung jawabkan atau tetap ada didalam perancangan kelas.

4. Sequence Diagram

Diagram sekuen atau *Sequence Diagram* yang menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sequence maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek

itu. Membuat sekuen diagram juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuan yang harus dibuat juga semakin banyak.

