

BAB II

TINJAUAN PUSTAKA

II.1. Sistem

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk melakukan sasaran tertentu. Pendekatan sistem yang merupakan jaringan kerja dari prosedur lebih menekankan urutan-urutan operasi di dalam sistem (Jeperson Hutahean ; 2014 :2).

Karakteristik sistem terdiri dari:

1. Komponen (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen dapat berupa suatu subsistem atau bagian-bagian sistem.

2. Batasan Sistem (*Boundary*)

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batasan suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar sistem (environment) adalah diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4. Penghubung Sistem (*Interface*)

Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya.

5. Masukan Sistem (*Input*)

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*Maintenance Input*) dan masukan sinyal (*Signal Input*).

6. Keluaran Sistem (*Output*)

Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

7. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah akan mengubah masukan menjadi keluaran.

8. Sasaran Sistem

Suatu sistem mempunyai tujuan(*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak ada gunanya.

II.2. Sistem Pendukung Keputusan

Sistem pendukung keputusan (*decision support systems* disingkat DSS) adalah bagian dari sistem informasi berbasis komputer termasuk sistem berbasis pengetahuan (manajemen pengetahuan) yang dipakai untuk mendukung pengambilan keputusan dalam suatu organisasi perusahaan atau lembaga pendidikan. Dapat juga dikatakan sebagai sistem komputer yang mengolah data menjadi informasi untuk mengambil keputusan dari masalah yang spesifik.

Menurut Moore and Chang, Sistem Pendukung keputusan dapat digambarkan sebagai sistem yang berkemampuan mendukung analisis data, dan pemodelan keputusan, berorientasi keputusan, orientasi perencanaan masa depan, dan digunakan pada saat-saat yang tidak biasa.

Dari pengertian di atas dapat di jelaskan bahwa sistem pendukung keputusan merupakan sistem yang membantu pengambilan keputusan yang dilengkapi dengan informasi dari data yang telah diolah dengan relevan dan di perlukan untuk membuat keputusan tentang suatu masalah dengan lebih cepat dan akurat. (Asep Abdul Wahit, Andri Ikhwana, Partono. 2012)

Sedangkan menurut Kusri (2007 : 15) DSS merupakan sistem informasi interaktif yang menyediakan informasi, pemodelan, dan pemanipulasian data. Sistem itu digunakan untuk membantu pengambilan keputusan dalam situasi yang semiterstruktur dan situasi tidak terstruktur, dimana tak seorang pun tahu secara pasti bagaimana keputusan seharusnya dibuat.

DSS biasanya dibangun untuk mendukung solusi atas suatu masalah atau untuk mengevaluasi suatu peluang. DSS yang seperti itu disebut aplikasi DSS.

Aplikasi DSS digunakan untuk pengambil keputusan. Aplikasi DSS menggunakan CBIS (*Computer Based Information Systems*) yang fleksibel, interaktif, dan dapat diadaptasi, yang dikembangkan untuk mendukung solusi atas masalah manajemen spesifik yang tidak terstruktur.

II.3. Alat KB atau Alat Kontrasepsi

Kontrasepsi berasal dari kata kontra berarti 'mencegah' atau 'melawan' dan konsepsi berarti pertemuan antara sel telur yang matang dan sel sperma yang mengakibatkan kehamilan. Maksud dari kontrasepsi adalah menghindari / mencegah terjadinya kehamilan sebagai akibat dari pertemuan antara sel telur yang matang dengan sperma tersebut.

Alat kontrasepsi memang sangat berguna sekali dalam program keluarga berencana (KB) namun perlu diketahuibahwa tidak semua alat kontrasepsi cocok dengan kondisi setiap orang. Untuk itu setiap pribadi harus bisa memilih alat kontrasepsi yang cocok bagi dirinya. (Kusumaningrum, 2009). Berdasarkan lama efektivitasnya, kontrasepsi dapat dibagi menjadi :

1. MJKP (Metode Kontrasepsi Jangka Panjang), yang termasuk dalam kategori ini adalah jenis susuk/implant, IUD, MOP, dan MOW.
2. Non MJKP (Non Metode Kontrasepsi Jangka Panjang), yang termasuk dalam kategori ini adalah kondom, pil, suntik, dan metode lain selain yang termasuk dalam MJKP.

Tidak ada satupun metode yang aman dan efektif bagi semua orang karena masing-masing mempunyai kesesuaian dan kecocokan individual bagi setiap orang. Berbagai jenis metode atau alat kontrasepsi dibagi menjadi

1. Kontrasepsi Sterilisasi

yaitu pencegahan kehamilan dengan mengikat sel indung telur pada wanita (*tubektomi*) atau testis pada pria (*vasektomi*). Proses sterilisasi ini harus dilakukan oleh ginekolog (dokter kandungan). Efektif bila memang ingin melakukan pencegahan kehamilan secara permanen, misalnya karena faktor usia.

2. Kontrasepsi Teknik, dibagi menjadi :

- a. *Coitus Interruptus (senggama terputus):*

ejakulasi dilakukan di luar vagina. Faktor kegagalan biasanya terjadi karena ada sperma yang sudah keluar sebelum ejakulasi, orgasme berulang atau terlambat menarik penis keluar.

- b. *Sistem kalender (pantang berkala):* tidak melakukan senggama pada masa subur, perlu kedisiplinan dan pengertian antara suami istri karena sperma maupun sel telur (ovum) mampu bertahan hidup sampai dengan 48 jam setelah ejakulasi. Faktor kegagalan karena salah menghitung masa subur (saat ovulasi) atau siklus haid tidak teratur sehingga perhitungan tidak akurat.

- c. *Prolonged lactation* atau menyusui, selama tiga bulan setelah melahirkan saat bayi hanya minum ASI dan menstruasi belum terjadi, otomatis tidak akan terjadi kehamilan. Tapi jika ibu hanya menyusui

kurang dari enam jam per hari, kemungkinan terjadi kehamilan cukup besar.

3. Kontrasepsi Mekanik, terdiri dari:

a. *Kondom*: Terbuat dari latex. Terdapat

kondom untuk pria maupun wanita serta berfungsi sebagai pemblokir sperma. Kegagalan pada umumnya karena kondom tidak dipasang sejak permulaan senggama atau terlambat menarik penis setelah ejakulasi sehingga kondom terlepas dan cairan sperma tumpah di dalam vagina.

b. *Spermatisida*: bahan kimia aktif untuk membunuh sperma, berbentuk cairan, krim atau tisu vagina yang harus dimasukkan ke dalam vagina lima menit sebelum senggama. Kegagalan sering terjadi karena waktu larut yang belum cukup, jumlah spermatisida yang digunakan terlalu sedikit atau vagina sudah dibilas dalam waktu kurang dari enam jam setelah senggama.

c. *Vaginal diafragma*: lingkaran cincin dilapisi karet fleksibel ini akan menutup mulut rahim bila dipasang dalam liang vagina enam jam sebelum senggama. Efektivitasnya sangat kecil, karena itu harus digunakan bersama spermatisida untuk mencapai efektivitas 80%.

d. *IUD (Intra Uterine Device)* atau spiral: terbuat dari bahan *polyethylene* yang diberi lilitan logam, umumnya tembaga (Cu) dan dipasang di mulut rahim. Kelemahan alat ini yaitu bisa menimbulkan rasa nyeri di perut, infeksi panggul, pendarahan di luar masa menstruasi atau darah menstruasi lebih banyak dari biasanya.

4. Kontrasepsi Hormonal Kontrasepsi hormonal bisa berupa pil KB yang diminum sesuai petunjuk hitungan hari yang ada pada setiap blisternya, suntikan, susuk (*Implan*) yang ditanam untuk periode tertentu, koyo KB atau spiral berhormon.

Kontrasepsi hormonal terdiri dari:

a. Pil Kombinasi *Oral Contraception (OC)*

Pil kombinasi merupakan kombinasi dosis rendah *estrogen* dan *progesteron*. Penggunaan kontrasepsi pil kombinasi *estrogen* dan *progesteron* atau yang hanya terdiri dari *progesteron* saja merupakan penggunaan kontrasepsi terbanyak.

b. Suntik KB Kontrasepsi suntikan mengandung hormon sintetik.

Cara pemakaiannya dengan menyuntikkan zat hormonal ke dalam tubuh. Zat hormonal yang terkandung dalam cairan suntikan dapat mencegah kehamilan dalam waktu tertentu. Biasanya penyuntikan ini dilakukan 2-3 kali dalam sebulan.

c. Susuk KB (*Implan*)

Implan terdiri dari 6 kapsul *silastik*, setiap kapsulnya berisi *levonorgestrel* sebanyak 36 miligram dengan panjang 3,4 cm dan diameter 2,4 cm. Kemasan *Implan* dirancang agar isinya tetap steril selama masa yang ditetapkan asalkan kemasannya tidak rusak atau terbuka. Kapsul yang dipasang harus dicabut menjelang akhir masa 5 tahun. Pemasangan implan hanya dilakukan petugas klinik yang terlatih secara khusus (dokter, bidan dan paramedik) yang dapat melakukan

pemasangan dan pencabutan *Implan*. Terdapat dua jenis *Implan* yaitu *Norplant* dan *Implanon*.

d. Koyo KB

Digunakan dengan ditempelkan di kulit setiap minggu. Kekurangannya adalah menimbulkan reaksi alergi bagi yang memiliki kulit sensitif dan kurang cocok untuk digunakan pada daerah beriklim tropis.

II.4. Visual Basic .Net

Visual basic 2010 merupakan salah satu bagian dari produk pemrograman terbaru yang dikeluarkan oleh Microsoft, yaitu microsoft Visual Studio 2010. Sebagai produk lingkungan pengembangan terintegrasi atau IDE andalan yang di keluarkan oleh microsoft, visual studio 2010 menambahkan perbaikan-perbaikan fitur dan fitur baru yang lebih lengkap visual studio pendahulunya, yaitu mirosoft visual studio 2008. (Wahana Komputer;2010;2)

Sedangkan menurut Aswan (2012 : 1) Visual basic 2010 adalah salah satu bagian dari microsoft visual studio 2010. Sebuat alat yang digunakan oleh pengembang windows dari berbagai level untuk mengembangkan dan membangun aplikasi yang bergerak diatas sistem .NET Framework, dengan menggunakan bahasa BASIC. Visual Basic menyediakan cara cepat dan mudah untuk membuat aplikasi.

Setiap generasi baru dari perangkat lunak bahasa pemrograman datang karena adanya keterbatasan dari generasi sebelumnya. Teknologi device, hardware, network dan internet baru yang muncul menyebabkan bahasa pemrograman yang

ada tidak lagi menjadi alat yang ideal untuk mengembangkan perangkat lunak yang dapat bekerja dengan teknologi baru tersebut (WAH[12]). Sekarang untuk pertama kalinya, platform pengembang perangkat lunak yang lengkap, Microsoft .NET telah didesain dari dasar dengan internet sebagai fokus utamanya (walaupun tidak secara eksklusif hanya untuk pengembang internet saja). Banyak inovasi baru yang berada dalam platform ini akan mengatasi keterbatasan dari tool-tool dan teknologi lama. Visual Basic .NET adalah pengembangan dari Visual basic sebelumnya. Kelebihan VB .NET 2010 terletak pada tampilannya yang lebih canggih dibandingkan dengan edisi Visual Basic sebelumnya. Selain memiliki kelebihan, VB .NET 2005 memiliki kekurangan. Kekurangan VB .NET 2005 yang terlihat jelas adalah beratnya aplikasi ini apabila dijalankan pada komputer yang memiliki spesifikasi sederhana.

II.5. SQL Server

SQL Server 2008 adalah sebuah terobosan baru dari Microsoft dalam bidang database. SQL Server adalah sebuah DBMS (Database Management System) yang yang dibuat Microsoft untuk berkecimpung dalam persaingan dunia pengolahan data menyusul pendahulunya seperti IBM dan Oracle. SQL Server 2008 dibuat pada saat kemajuan dalam bidang hardware sedemikian pesat. Oleh karena itu sudah dapat dipastikan bahwa SQL Server 2008 membawa beberapa terobosan dalam bidang pengolahan dan penyimpanan data. (Wahana Komputer:2010;2)

II.6. Metode SAW

Metode SAW (Simple Additive Weighting) sering juga dikenal istilah metode penjumlahan berbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut. Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada. (Heri Sulistiyo:2010)

Metode SAW dikenal sebagai istilah penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut. Metode SAW membutuhkan proses normalisasi matriks keputusan (X) ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada. Formula untuk melakukan normalisasi tersebut adalah sebagai berikut: dengan r adalah rating kinerja ternormalisasi dari alternatif A_i pada atribut C_j ; $i=1,2,...,m$ dan $j=1,2,...,n$. (Yohana Dewi : 2010)

$$r_{ij} = \begin{cases} \frac{x_{ij}}{\max_i x_{ij}} & \text{jika } j \text{ adalah atribut keuntungan} \\ \frac{\min_i x_{ij}}{x_{ij}} & \text{jika } j \text{ adalah atribut biaya (cost)} \end{cases}$$

Nilai preferensi untuk setiap alternatif (V_i) diberikan sebagai :

$$V_i = \sum_{j=1}^n w_j r_{ij} \dots\dots\dots(1)$$

$$V = w \times r$$

dengan:

V = Nilai Matriks

w = Matriks rating kepentingan (bobot)

r = rating

Nilai V_i yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih. Setelah tujuan dan alternatif keputusan telah didapatkan, langkah selanjutnya adalah mengidentifikasi kumpulan kriteria.

II.7. UML (*Unified Modeling Language*)

UML(*Unified Modeling Language*) yang merupakan metodologi kolaborasi antara metoda booch, OMT (*Object Modeling Technique*), serta OOSE (*Oriented Software Engineering*) dan beberapa metoda lainnya, merupakan metodologi yang paling sering digunakan saat ini untuk mengadaptasi maraknya penggunaan bahasa “pemrograman berorientasi objek” (OOP). (Adi Nugroho;2009;4)

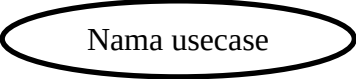
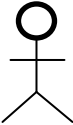
UML (*Unified Modeling Language*) adalah sebuah ”bahasa” yang telah menjadi standar dalam industry untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan *syntax/semantic*. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk – bentuk tersebut dapat dikombinasikan. *Unified Modeling Language* biasa digunakan untuk :

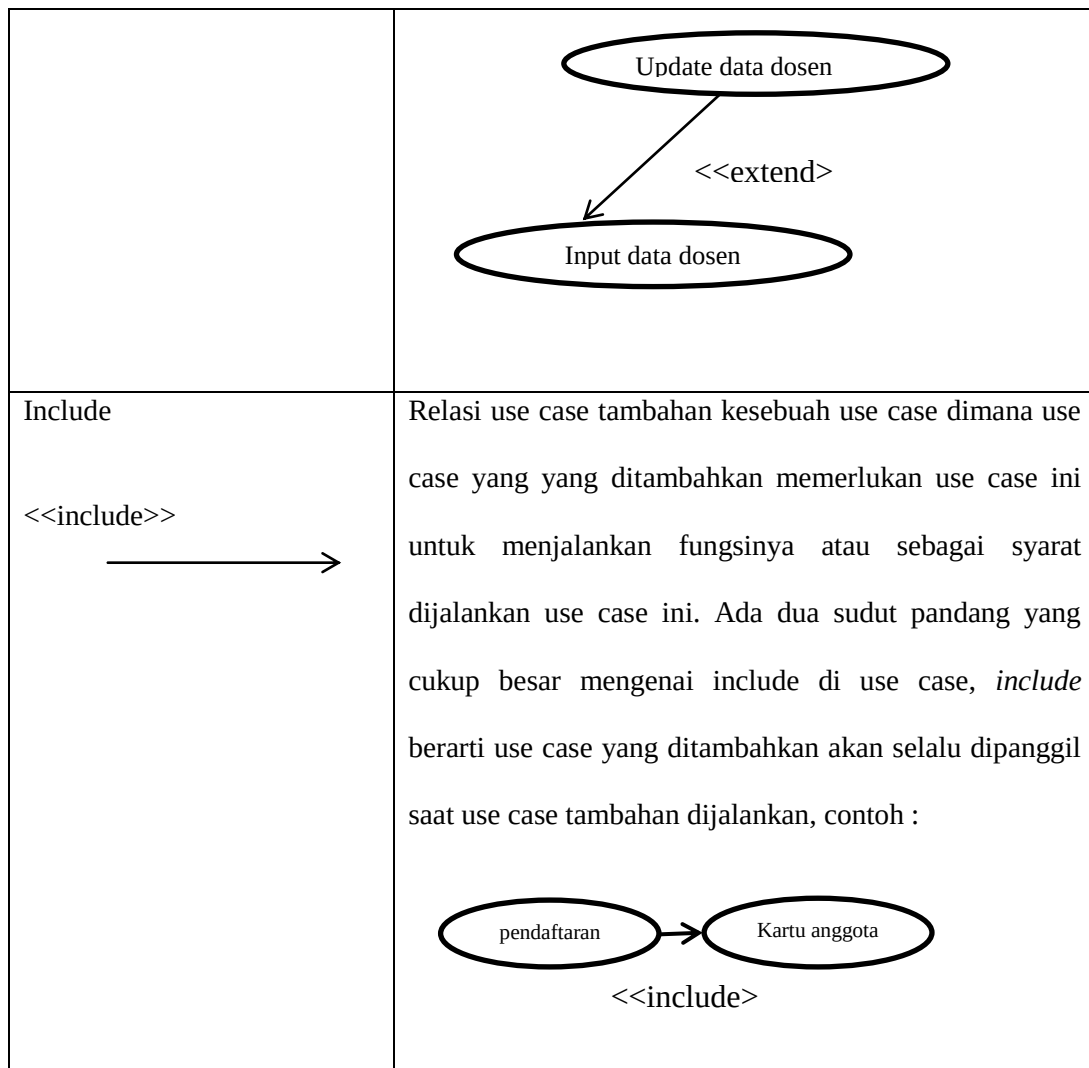
1. Menggambarkan batasan sistem dan fungsi – fungsi sistem secara umum, di buat dengan *use case* dan *actor*.
2. Menggambarkan kegiatan atau proses bisnis yang di laksanakan secara umum, di buat dengan *interaction diagrams*.
3. Menggambarkan representasi struktur *static* sebuah sistem dalam bentuk *class diagrams*.
4. Membuat model behavior “yang menggambarkan kebiasaan atau sifat sebuah sistem” dengan *state transition diagrams*.
5. Menyatakan arsitektur implementasi fisik menggunakan *component and development diagrams*.
6. Menyampaikan atau memperluas *fungsi* dengan *stereotypes*. (Yuni Sugiarti; 2013 :36)

II.8. Use Case Diagram

Use case diagram merupakan pemodelan untuk menggambarkan kelakuan (*behavior*) sistem yang akan dibuat. Diagram *use case* mendeskripsikan sebuah interaksi antara satu atau lebih *actor* dengan sistem yang akan dibuat. Dengan pengertian yang cepat, diagram *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi – fungsi tersebut. Terdapat beberapa simbol dalam menggambarkan diagram *use case*, yaitu *use case*, *actor* dan relasi. Berikut adalah simbol – simbol yang ada pada diagram *use case*. (Yuni Sugiarti; 2013: 42)

Tabel II.1 Simbol – simbol pada Use Case Diagram

Simbol	Deskripsi
Use case 	Fungsionalitas yang disediakan sistem sebagai unit – unit yang saling bertukar pesan antar unit atau <i>actor</i> ; biasanya ditanyakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> .
Aktor  nama aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari <i>actor</i> adalah gambar orang, tapi <i>actor</i> belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama <i>actor</i> .
Asosiasi/ <i>association</i> 	Komunikasi antara actor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
Extend  <<extend>>	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu; mirip dengan prinsip inheritance pada pemrograman berorientasi objek; biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan, arah panah menunjuk pada use case yang dituju. Contoh :



Sumber: (Yuni Sugiarti; 2013)

II.9. Class Diagram

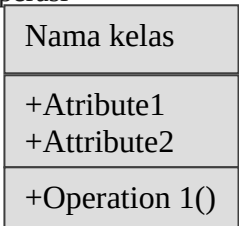
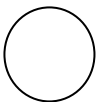
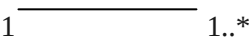
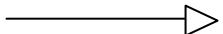
Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan di buat untuk membangun sistem. Kelas memiliki apa yang di sebut atribut dan metode atau operasi.

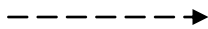
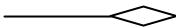
1. Atribut merupakan variabel- variabel yang di miliki oleh suatu kelas.
2. Atribut mendeskripsikan properti dengan sebaris teks di dalam kotak kelas tersebut.

3. Operasi atau metode adalah fungsi – fungsi yang di miliki oleh suatu kelas.

Diagram kelas mendeskripsikan jenis – jenis objek dalam sistem dan berbagai hubungan statis yang terdapat di antara mereka. Diagram kelas juga menunjukkan properti dan operasi sebuah kelas dan batasan – batasan yang terdapat dalam hubungan – hubungan objek tersebut. (Yuni Sugiarti; 2013: 57)

Tabel II.2 Simbol – simbol Class Diagram

Simbol	Deskripsi
Package 	Package merupakan sebuah bungkus dari satu atau lebih kelas
Operasi 	Kelas pada struktur sistem
Antarmuka / interface 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
Asosiasi berarah/directed asosiasi 	Relasi antar kelas dengan makna kelas yang satu di gunakan oleh kelas yang lain, asosiasi biasanya juga di sertai dengan <i>multiplicity</i> .
Generalisasi 	Relasi antar kelas dengan makna generalisasi – spesialisasi (umum khusus).

Kebergantungan / defedency 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi 	Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>)

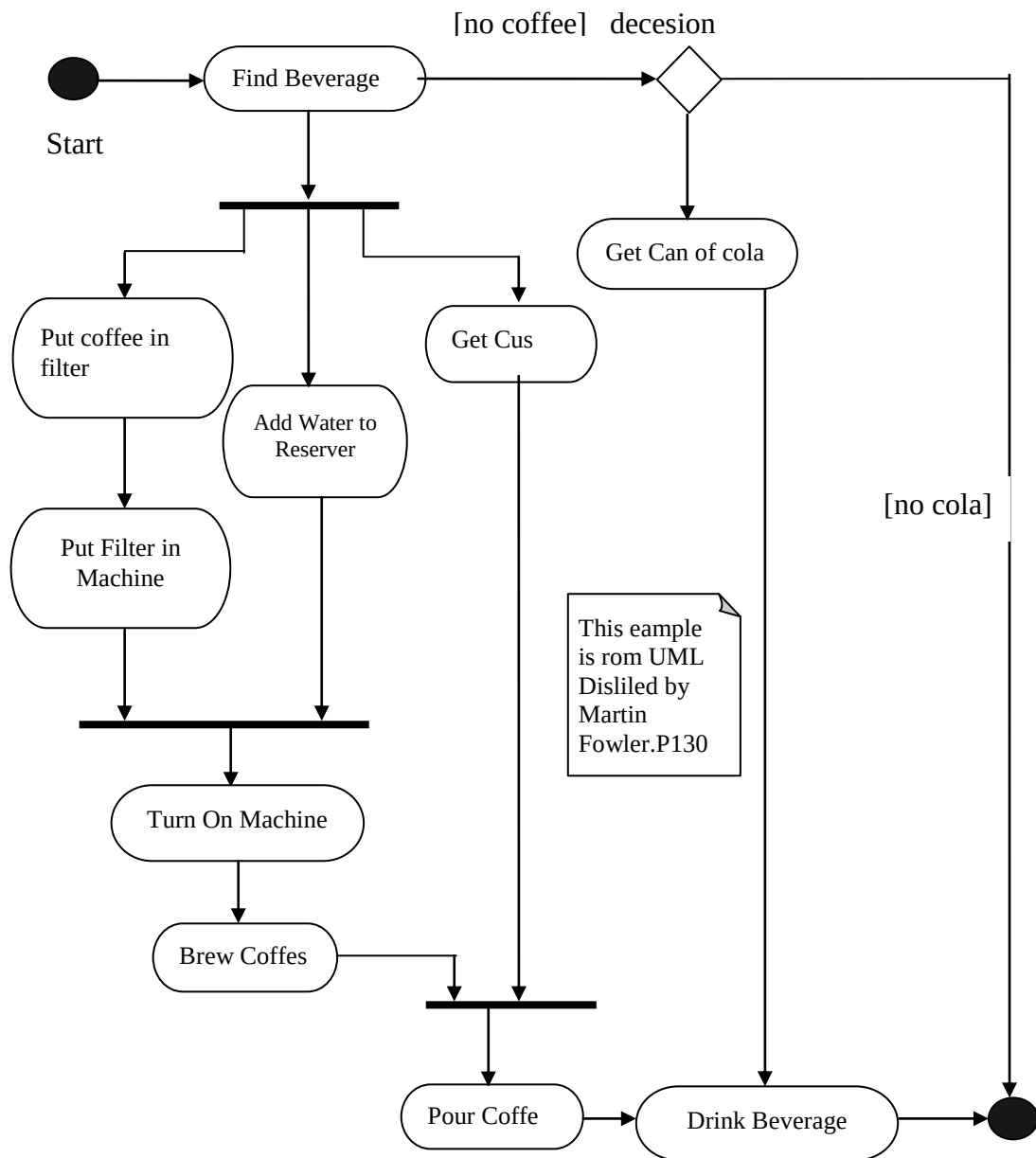
Sumber : (Yuni Sugiarti ; 2013)

II.10. Activity Diagram

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

Activity diagram merupakan *state* diagram khusus, di mana sebagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (*internal processing*). Oleh karena itu *activity* diagram tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Sebuah aktivitas dapat direalisasikan oleh satu use case atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara use case menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. (Yuni Sugiarti; 2013: 75)



Gambar II.1 Activity Diagram

Sumber : (Yuni Sugiarti ; 2013)

II.11. Sequence Diagram

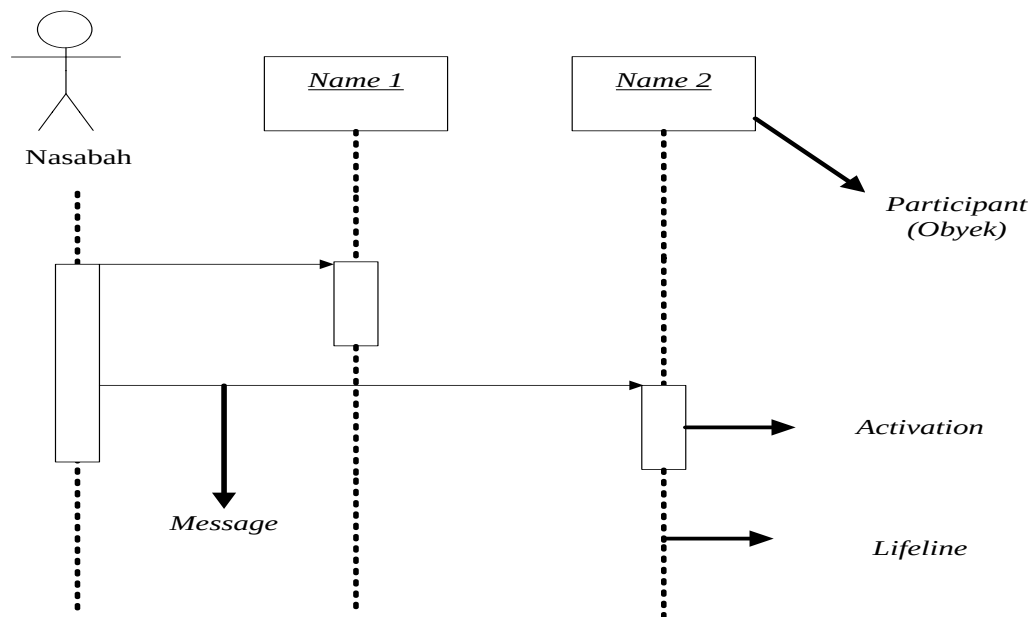
Diagram sekuence menggambarkan kelakuan/ pelaku objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sequence

maka harus diketahui objek – objek yang terlibat dalam sebuah use case beserta metode – metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Diagram sequence memiliki ciri yang berbeda dengan diagram interaksi pada diagram kolaborasi sebagai berikut :

1. Pada diagram sequence terdapat garis hidup objek. Garis hidup objek adalah garis vertical yang mencerminkan eksistensi sebuah objek sepanjang periode waktu. Sebagian besar objek – objek yang tercakup dalam diagram interaksi akan eksis sepanjang durasi tertentu dari interaksi, sehingga objek – objek itu diletakkan dibagian atas diagram dengan garis hidup tergambar dari atas hingga bagian bawah diagram. Suatu objek lain dapat saja diciptakan, dalam hal ini garis hidup dimulai saat pesan *destroy*,
2. jika kasus ini terjadi, maka garis hidupnya juga berakhir.
3. Terdapat focus kendali (*Focus Of Control*), berupa empat persegi panjang ramping dan tinggi yang menampilkan aksi suatu objek secara langsung atau sepanjang sub ordinat. Puncak dari empat persegi panjang adalah permulaan aksi, bagian dasar adalah akhir dari suatu aksi. Pada diagram ini mungkin juga memperhatikan penyaringan (*nesting*) dan *focus* kendali yang disebabkan oleh proses rekursif dengan menumpuk *focus* kendali yang lain pada induknya. (Yuni Sugiarti; 2013: 70)

Berikut simbol – simbol yang ada pada sequence diagram.



Gambar II.2 Simbol Sequence

Sumber : (Yuni Sugiarti ; 2013)

II.12. Normalisasi

Normalisasi adalah teknik perancangan yang banyak digunakan sebagai pemandu dalam merancang basisdata relasional. Pada dasarnya, normalisasi adalah proses dua langkah yang meletakkan data dalam bentuk tabulasi dengan menghilangkan kelompok berulang lalu menghilangkan data yang terduplikasi dari tabel relasional (www.utexas.edu). (Janner Simarmata & dkk; 2010 : 77)

Normalisasi adalah bagian perancangan basis data. Tanpa normalisasi, sistem basis data menjadi tidak akurat, lambat, tidak efisien, serta tidak memberikan data yang diharapkan. Pada waktu menormalisasikan basis data, ada empat tujuan yang harus dicapai, yaitu:

1. Mengatur data dalam kelompok – kelompok sehingga masing – masing kelompok hanya mengenai bagian kecil sistem.
2. Meminimalkan jumlah data berulang dalam basis data.
3. Membuat basis data yang datanya diakses dan dimanipulasi secara cepat dan efisien tanpa melupakan integritas data.
4. Mengatur data sedemikian rupa sehingga ketika memodifikasi data, anda hanya mengubah pada satu tempat.

II.12.1. Bentuk – Bentuk Normalisasi

1. Bentuk normal pertama (1NF)

Tahap ini dilakukan penghilangan beberapa group elemen yang berulang agar menjadi satu harga tunggal yang berinteraksi di antara setiap baris pada suatu tabel, dan setiap *atribut* harus mempunyai nilai data yang *atomic* (bersifat *atomic value*).

2. Bentuk normal kedua (2NF)

Normal kedua didasari atas konsep *full functionanl dependency* (ketergantungan fungsional sepenuhnya) yang dapat didefinisikan sebagai berikut.

Jika A dan B adalah *atribut-atribut* dari suatu relasi, B dikatakan *full function dependency* (miliki ketergantungan fungsional sepenuhnya) terhadap A, jika B adalah tergantung fungsioal terhadap A, tetapi tidak secara tepat memiliki ketergantungan fungsional dari *subset* (himpunan bagian) dari A.

3. Bentuk normal ketiga (3NF)

Jika kita hanya mengupdate satu baris saja, sementara baris yang lainnya tidak, maka data di dalam *database* tersebut akan *inkonsisten*/tidak teratur. Anomali *update* ini disebabkan oleh suatu ketergantungan transitif (*transitive dependency*). Kita harus menghilangkan ketergantungan tersebut dengan melakukan normalisasi ketiga (3-NF).

4. Bentuk normal *boyce-code* (BCNF)

Suatu relasi dalam basis data harus dirancang sedemikian rupa sehingga mereka memiliki ketergantungan sebagian (*partial dependency*), maupun ketergantungan transitif.

II.13. Kamus Data

Kamus data (*data dictionary*) dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan). Kamus data biasanya berisi:

1. Nama - nama dari data
2. Digunakan pada – merupakan proses-proses yang terkait data
3. Deskripsi – merupakan deskripsi data
4. Informasi tambahan – seperti tipe data, nilai data, batas nilai data dan komponen yang membentuk data. (Rosa A.S & M Shalauddin; 2011 : 67)