

BAB II

TINJAUAN PUSTAKA

II.1. Sistem

Definisi sistem adalah “kumpulan dari bagian-bagian yang bekerja sama untuk mencapai tujuan yang sama.” Definisi sistem adalah “sekumpulan objek-objek yang saling berhubungan dan berinteraksi serta hubungan antar objek bisa dilihat sebagai suatu kesatuan yang dirancang untuk mencapai suatu tujuan. Sistem adalah penggabungan dari bagian-bagian atau komponen-komponen yang terpisah-pisah dan disatukan menjadi satu rangkaian dan menjadi suatu fungsi yang baru (Aris : 2015).

II.1.1. Karakteristik Sistem

Untuk memahami atau mengembangkan suatu sistem, maka perlu membedakan unsur-unsur dari sistem yang membentuknya. Berikut adalah karakteristik sistem yang dapat membedakan suatu sistem dengan sistem lainnya yaitu:

1. Batasan (*boundary*) : Penggambaran dari suatu elemen atau unsur mana yang termasuk di dalam sistem dan mana yang di luar sistem.
2. Lingkungan (*environment*) : Segala sesuatu di luar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.
3. Masukan (*input*) : Sumber daya (data, bahan baku, peralatan, energi) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.

4. Keluaran (*output*) : Sumber daya atau produk (informasi, laporan, dokumen, tampilan layar Komputer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.
5. Komponen (*component*) : Kegiatan-kegiatan atau proses dalam sistem yang mentransformasikan input menjadi bentuk setengah jadi (*output*). Komponen ini bisa merupakan subsistem dari sebuah sistem.
6. Penghubung (*Interface*) : Tempat di mana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.
7. Penyimpanan (*storage*): Area yang dikuasai dan digunakan untuk penyimpanan sementara dan tetap dari informasi, energi, bahan baku, dan sebagainya. Penyimpanan merupakan suatu media penyangga di antara komponen tersebut bekerja dengan berbagai tingkatan yang ada dan memungkinkan komponen yang berbeda dari berbagai data yang sama (Aris : 2015).

II.2. Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. (1) Akurat (*Accurate*) Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. (2) Tepat waktu (*Timelines*) Berarti informasi yang datang pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai logika karena informasi merupakan landasan dalam pengambilan keputusan. (3) Relevan (*Relevance*)

Berarti informasi tersebut bermanfaat bagi pemakainya (Deppi Linda, 2016 : 62-63).

II.3. Sistem Informasi

Sebuah sistem mempunyai tujuan atau sasaran berpendapat sistem adalah sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan, mendefinisikan sistem sebagai seperangkat elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan bersama.

Sistem informasi merupakan kumpulan dari sub-sub sistem baik fisik maupun non fisik yang berhubungan satu sama lain dan bekerjasama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berarti dan berguna. Berdasarkan dari kedua definisi di atas, dapat disimpulkan bahwa sistem informasi adalah kumpulan dari sub sub sistem dalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Mutammim Ula:2014).

II.4. Sistem Informasi Akuntansi

Sistem informasi akuntansi adalah kumpulan sumberdaya, seperti manusia dan peralatan, yang diatur untuk mengubah data menjadi informasi. Informasi ini dikomunikasikan kepada beragam pengambil keputusan. Sistem Informasi Akuntansi mewujudkan perubahan ini secara manual atau terkomputerisasi. Sistem Informasi Akuntansi juga merupakan sistem yang paling penting di

organisasi dan merubah cara menangkap, memproses, menyimpan, dan mendistribusikan informasi. Saat ini, digital dan informasi *online* semakin digunakan dalam Sistem Informasi Akuntansi. Organisasi perlu menempatkan sistem di lini depan, dan mempertimbangkan baik segi sistem ataupun manusia sebagai factor yang terkait ketika mengatur Sistem Informasi Akuntansi. Sistem Informasi Akuntansi pada umumnya meliputi beberapa siklus pemrosesan transaksi :

1. Siklus pendapatan. Berkaitan dengan pendistribusian barang dan jasa ke entitas lain dan pengumpulan pembayaran-pembayaran yang berkaitan.
2. Siklus pengeluaran. Berkaitan dengan perolehan barang jasa dari entitas lain dan pelunasan kewajiban yang berkaitan.
3. Siklus produksi. Berkaitan dengan pengubahan sumber daya menjadi barang dan jasa.
4. Siklus keuangan. Kejadian-kejadian yang berkaitan dengan perolehan dan manajemen dana-dana modal, termasuk kas. (Agustinus Mujilan : 2012 : 3).

II.5. Bagi Hasil Usaha

Prinsip bagi hasil menurut Bank Indonesia adalah Suatu prinsip pembagian laba yang diterapkan dalam kemitraan kerja, dimana porsi bagi hasil ditentukan pada saat akad kerja sama. Jika usaha mendapatkan keuntungan, porsi bagi hasil adalah sesuai kesepakatan namun jika terjadi kerugian maka porsi bagi hasil disesuaikan dengan kontribusi modal masing masing pihak. Dasar yang digunakan dalam perhitungan bagi hasil adalah berupa laba bersih

usaha setelah dikurangi dengan biaya operasional. Profit sharing (bagi hasil), pada dasarnya merupakan pembiayaan dengan prinsip kepercayaan dan kesepakatan murni antara kedua belah pihak atau lebih yaitu, pemilik modal (*investor*) dalam hal ini Bank Syariah dengan pemilik usaha dalam hal ini nasabah adalah pengelola usaha. Prinsip bagi hasil dalam perbankan syariah yang paling banyak dipakai adalah Al-musyarakah dan Al-Mudharabah. Al-Musyarakah adalah akad kerja sama antara dua pihak atau lebih untuk suatu usaha tertentu dimana masing-masing pihak memberikan kontribusi dana dengan kesepakatan bahwa keuntungan dan risiko akan ditanggung bersama sesuai dengan kesepakatan. Al-Mudharabah berasal dari kata dharab, yang berarti berjalan atau memukul.

Secara teknis, Al-Mudharabah adalah kerjasama usaha antara dua orang dimana pihak pertama (*shohibul maal*) menyediakan seluruh modal, sedangkan pihak lainnya menjadi pengelola. Keuntungan usaha dibagi menurut kesepakatan dalam kontrak, sedangkan apabila rugi ditanggung oleh pemilik modal selama kerugian itu bukan akibat kelalaian si pengelola. Seandainya kerugian itu diakibatkan karena kecurangan atau kelalaian pengelola, pengelola harus bertanggungjawab atas kerugian tersebut (Kartika Soetopo:2013).

II.6. Basis Data (*Database*)

Basis data (*database*) adalah kumpulan dari berbagai data yang saling berelasi atau berhubungan satu dengan yang lainnya, data dinyatakan dengan nilai. Basis data tersimpan di perangkat keras, serta dimanipulasi dengan

menggunakan perangkat lunak. Pendefinisian basis data meliputi spesifikasi dari tipe data, struktur dan batasan dari data atau informasi yang akan disimpan.

Istilah-istilah dalam basis data: (Mutammimul Ula, 2014 : 38).

1. Enterprise: suatu bentuk organisasi seperti Bank, Sekolah, Rumah Sakit, Pabrik, Kantor dan sebagainya.
2. Entitas: suatu objek yang dapat dibedakan dari lainnya yang dapat di wujudkan dalam basis data. Kumpulan dari entitas disebut himpunan entitas.
3. Atribut dan elemen data: karakteristik dari suatu entitas.
4. Record data: kumpulan suatu elemen data yang saling berhubungan.
5. Tabel: kumpulan data atau informasi.

Basis data mempunyai peran penting dalam sistem informasi, yaitu sebagai sumber penyedia data utama untuk memenuhi kebutuhan-kebutuhan informasi seluruh pemakai atau informasi bagi para pengambil keputusan (Mutammimul Ula, 2014 : 38).

II.7. Normalisasi

Normalisasi adalah suatu proses untuk membuat data yang tidak normal menjadi data yang normal. Bentuk data yang tidak normal / data mentah biasa disebut juga *unnormalized form*. Masing – masing level normalisasi mempunyai aturan tersendiri.

1. *First Normal Form*

Suatu tabel dikatakan dalam keadaan *First Normal Form* (1NF) jika :

- a. Tidak ada perulangan record data dalam tabel.
- b. Setiap sel memiliki satu nilai saja. Artinya tidak ada perulangan group dan array.
- c. Data yang diinputkan memiliki tipe data yang sama dengan tipe data kolom dalam tabel.

2. *Second Normal Form*

Suatu tabel dikatakan dalam keadaan *Second Normal Form* (2NF) jika tabel tersebut sudah dalam keadaan *First Normal Form* (1NF) dan jika semua atribut yang bukan kunci tabel, baik *Primary Key* maupun *Foreign Key* tergantung pada semua kunci dalam tabel.

3. *Third Normal Form*

Suatu tabel dikatakan dalam keadaan *Third Normal Form* (3NF) jika tabel tersebut sudah dalam keadaan *Second Normal Form* (2NF) dan jika tidak terdapat ketergantungan yang transitif. Artinya, data-data yang mungkin diisi berulang-ulang dapat dibuat sebuah tabel baru.

4. *Boyce-Codd Normal Form* (BCNF)

Tabel dikatakan dalam keadaan *Boyce-Codd Normal Form* (BCNF) jika tabel tersebut dalam keadaan *Third Normal Form* (3NF) dan setiap determinan adalah kunci kandidat.

5. *Fourth Normal Form (4NF)*

Suatu tabel dikatakan dalam keadaan *Fourth Normal Form (4NF)* jika tabel tersebut dalam keadaan *Boyce-Codd Normal Form (BCNF)* dan jika tidak terdapat ketergantungan nilai ganda.

6. *Fiveth Normal Form (5NF)*

Tabel dikatakan dalam keadaan *Fiveth Normal Form (5NF)* jika tabel tersebut dalam keadaan *Fourth Normal Form (4NF)* dan jika setiap ketergantungan dalam join ada pada tabel sudah konsekuen dengan kunci kandidat pada tabel tersebut (Ema Utami ; 2012 : 73-76)

II.8. Pengertian Kamus Data

Kamus data umumnya berguna pada pengembangan model sistem dan dapat digunakan untuk menangani semua informasi dari semua tipe model sistem. Kamus data sederhanya adalah daftar alfabetis dari nama-nama termasuk pada berbagai model sistem. Seperti namanya, kamus harus mencakup deskripsi yang berhubungan dengan entitas bernama tersebut dan, jika nama itu merepresentasikan objek komposit, mungkin saja ada deskripsi mengenai komposisinya. Informasi lain seperti tanggal pembuatan, pembuatnya dan representasi entitas juga dapat dimasukkan, tergantung pada tipe model yang sedang dikembangkan. Keuntungan penggunaan kamus data adalah :

1. Kamus data merupakan mekanisme untuk manajemen nama. Banyak orang yang harus menciptakan nama untuk entitas dan relasi ketika mengembangkan model sistem yang besar. Nama-nama ini harus dipakai secara konsisten dan

tidak boleh bentrok. Perangkat lunak kamus data dapat memeriksa keunikan nama dan memberitahu analisis persyaratan sekiranya terjadi duplikasi nama.

2. Kamus data berfungsi sebagai tempat penyimpanan informasi organisasional yang dapat menghubungkan analisis, desain, implementasi dan evolusi. Sementara sistem dikembangkan, informasi diambil untuk memberitahu perkembangan. Informasi baru ditambahkan pada sistem. Semua informasi mengenai entitas berada pada satu tempat (Ian Sommerville ; 2012 : 151).

Tabel II.1 Entri Kamus Data

Nama	Keterangan	Tipe	Tanggal
Has-labels	Relasi 1 N antara entitas bertipe node atau link dan entitas bertipe label	Relasi	5.10.1998
Label	Berisi informasi terstruktur atau tidak terstruktur mengenai node atau link. Label dipresentasikan oleh ikon (yang bisa berupa kotak transparan) dan teks yang berhubungan	Entitas	8.10.1998
Link	Relasi 1 : 1 antara entitas desain yang dipresentasikan sebagai node. Link diketikkan dan dapat diberi nama	Relasi	8.10.1998
Name (Label)	Setiap label memiliki nama yang mengidentifikasi tipe label. Nama harus unik dalam set tipe label yang dipakai pada desain	Atribut	8.10.1998
Name (Node)	Setiap node harus memiliki nama yang unik di dalam sebuah desain. Panjang nama bisa mencapai 64 karakter.	Atribut	15.10.1998

(Sumber : Ian Sommerville ; 2012 : 151)

II.9. Visual Studio 2010

Microsoft Visual Studio .NET merupakan sebuah perangkat lunak yang digunakan untuk membangun aplikasi *web* berbasis ASP, layanan *web* XML, aplikasi *desktop*, maupun aplikasi *mobile*. *Visual Studio* memiliki kompiler, SDK, *Integrated Development Environment* (IDE), dan dokumen *help* (*MSDN Library*). Kompiler yang terdapat dalam *Visual Studio* antara lain adalah kompiler untuk bahasa pemrograman Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe.

Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun *managed code* (dalam bentuk *Microsoft Intermediate Language* di atas *.NET Framework*). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi Silverlight, aplikasi *Windows Mobile* (yang berjalan di atas *.NET Compact Framework*). Visual Studio kini telah menginjak versi Visual Studio 9.0.21022.08, atau dikenal dengan sebutan Microsoft Visual Studio 2008 yang diluncurkan pada 19 November 2007, yang ditujukan untuk *platform* Microsoft *.NET Framework* 3.5. Versi sebelumnya, Visual Studio 2005 ditujukan untuk *platform* *.NET Framework* 2.0 dan 3.0. Visual Studio 2003 ditujukan untuk *.NET Framework* 1.1, dan Visual Studio 2002 ditujukan untuk *.NET Framework* 1.0. Versi-versi tersebut kinidikenal dengan sebutan Visual Studio .NET, karena memang membutuhkan Microsoft *.NET Framework* (Teddy Marcus : 2012).

II.10. SQL Server 2008

Microsoft SQL Server adalah sebuah sistem manajemen basis data relasional produk *Microsoft*. Bahasa *query* utamanya adalah *Transact-SQL* yang merupakan implementasi dari SQL standar ANSI/ISO yang digunakan oleh *Microsoft* dan *Sybase*. Umumnya SQL Server digunakan di dunia bisnis yang memiliki basis data berskala kecil sampai dengan menengah, tetapi kemudian berkembang dengan digunakannya SQL Server pada basis data besar. *Microsoft SQL Server* dan *Sybase/ASE* dapat berkomunikasi lewat jaringan dengan menggunakan protokol TDS (*Tabular Data Stream*). Selain dari itu, *Microsoft SQL Server* juga mendukung ODBC (*Open Database Connectivity*), dan mempunyai driver JDBC untuk bahasa pemrograman Java. Fitur yang lain dari SQL Server ini adalah kemampuannya untuk membuat basis data *mirroring dan clustering* (Teddy Marcus : 2012).

II.11. UML (*Unified Modelling Language*)

Unified Modelling Language (UML) menyediakan beberapa notasi dan artefak standar yang bisa digunakan sebagai alat komunikasi bagi para pelaku dalam proses analisis dan desain. Artefak dalam UML adalah informasi dalam berbagai bentuk yang digunakan atau dihasilkan dalam proses pengembangan perangkat lunak. Contohnya adalah *source code* yang dihasilkan oleh pemrograman. Yang perlu diperhatikan untuk menjaga konsistensi antar artefak selama proses analisis dan desain adalah bahwa setiap perubahan terjadi pada

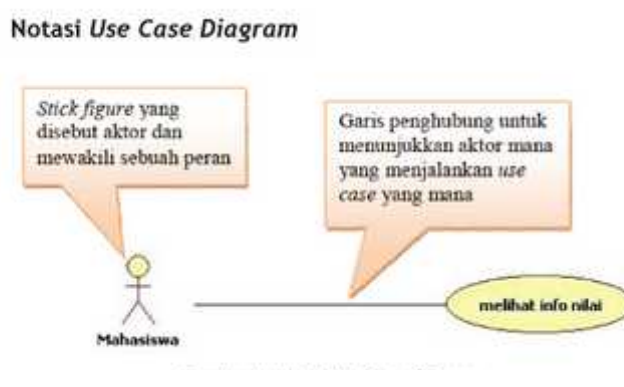
suatu artifak yang harus juga dilakukan pada artifak lainnya. (Sumber: Evi Triandini Dan I Gede Suardika, 2012)

II.11.1. Diagram-Diagram UML

1. Diagram Use Case (Use Case Diagram)

Pengertian use case menurut Jhon Satzinger, 2010 dalam buku *system analysis and design in a changing world* menyatakan bahwa “use case adalah sebuah kegiatan yang dilakukan oleh sistem, biasanya dalam menanggapi permintaan dari pengguna sistem. (Evi Triandini Dan I Gede Suardika, 2012)

Notasi *Use Case Diagram* dapat dilihat pada gambar berikut ini:



Gambar II.1. Notasi Use Case Diagram

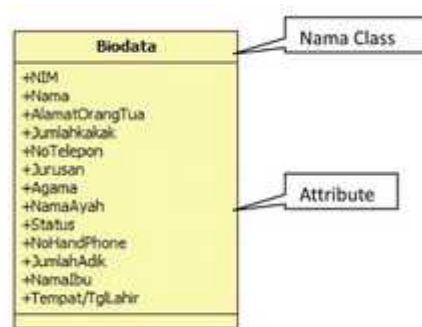
(Sumber : Evi Triandini dan I Gede Suardika)

2. Diagram Kelas (*Class Diagram*)

Menurut John Satzinger, dalam buku *system analysis and design in a changing world* menyatakan bahwa “dalam UML ada dua jenis clas diagram yaitu: *domain class diagram* dan *design class diagram*

1. Domain class diagram

Focus domain class diagram adalah pada sesuatu dalam lingkungan kerja pengguna, bukan pada class perangkat lunak yang nantinya akan dirancang. Berikut ini adalah gambar dari domain class diagram:

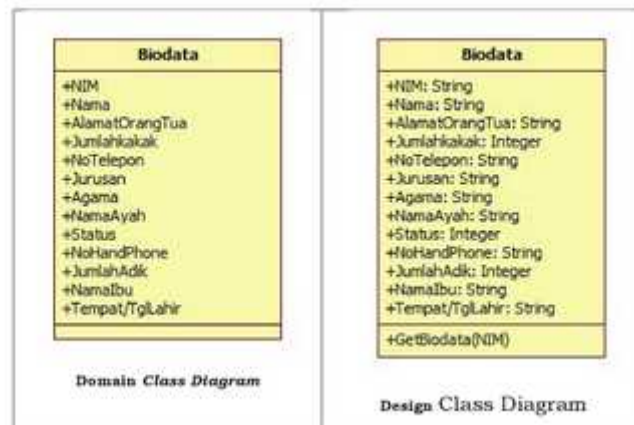


Gambar II.2. Domain Class Diagram

(Sumber : Evi Triandini Dan I Gede Suardika)

2. Design Class Diagram

Tujuan utamanya adalah untuk mendokumentasikan dan menggambarkan kelas-kelas dalam pemrograman yang nantinya akan dibangun. *Design Class Diagram* menggambarkan kelas berorientasi objek yang dibutuhkan dalam pemrograman, navigasi diantara kelas, atribut names dan propertynya serta method names dan propertynya. Gambar dari class diagram dapat dilihat pada gambar di bawah ini:

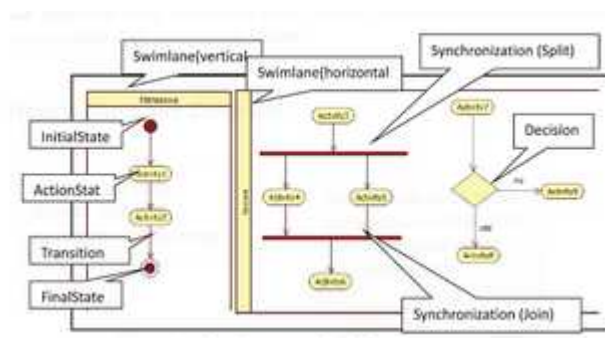


Gambar II.3. Calss Diagram

(Sumber : Evi Triandini Dan I Gede Suardika)

3. Diagram Aktivitas (*Activity Diagram*)

Menurut John Satzinger, dalam buku sistem analysis and design in a changing world menyatakan bahwa” activity diagram adalah sebuah diagram alur kerja yang menjelaskan berbagai kegiatan pengguna (atau pun sistem), orang yang melakukan masing-masing aktivitas, dan aliran sekuensial dari aktivitas-aktivitas tersebut. Notasi *Activity Diagram* adalah sebagai berikut:



Gambar II.4. Notasi Activity Diagram

(Sumber : Evi Triandini dan I Gede Suardika)

Penjelasan dari gambar diatas adalah sebagai berikut:

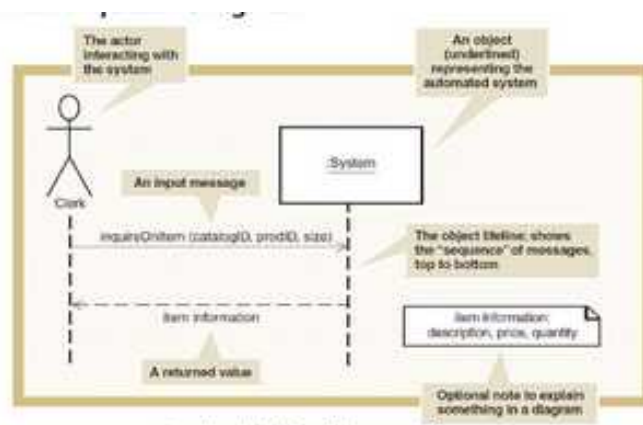
Tabel II.2. Keterangan Notasi Activity Diagram

Notasi	Keterangan
Swimlane	Mewakili agen yang melakukan aktivitas karena dalam alur kerja umumnya mempunyai agen yang berbeda dengan yang melakukan langkah yang berbeda dari proses alur kerja.
Initialstate	Awal dari alur kerja
ActionState	Melambangkan aktivitas tersendiri dalam alur kerja
Transition	Melambangkan dari urutan diantara aktivitas
Final State	Akhir dari alur kerja
Synchronization	Membagi alur kerja menjadi beberapa alur yang berbarengan ataupun menggabungkan lagi alur yang berbarengan
Decision	Titik pengambilan keputusan dimana aliran proses tersebut akan mengikuti satu jalur atau jalur lainnya

(Sumber : Evi Triandini dan I Gede Suardika)

4. Sequence Diagram

Menurut John Satzinger, dalam buku sistem *analysis and design in a changing world* menyatakan bahwa” Sistem Sequence Diagram (SSD) adalah diagram output serta urutan interaksi antara pengguna dan sistem untuk sebuah use case. Berikut ini adalah gambar dari sequence diagram :



Gambar II.5. Sequence Diagram

(Sumber : Evi Triandini dan I Gede Suardika)

Penjelasan dari gambar diatas adalah sebagai berikut:

- a. *Actor*, mewakili seorang aktor (orang atau peran yang berinteraksi dengan sistem)
- b. Kontak berlabel, sistem adalah objek yang mewakili keseluruhan sistem yang terotomatisasi
- c. Garis putus-putus vertikal (lifelines) adalah perpanjangan objek tersebut, baik aktor maupun objek, sepanjang durasi dari sequence diagram
- d. Anak panah antara lifeline mewakili message yang dikirim atau diterima oleh aktor dari sistem
- e. *Message* diberi label untuk menggambarkan maksud message dan input apa pun yang sedang dikirim. Message dipertimbangkan sebagai sebuah inti yang diminta pada tujuan objek, kebanyakan seperti perintah.