

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Adapun penelitian terkait yang akan digunakan sebagai sumber acuan yang relevan dan terkini yaitu:

1. Pada jurnal Isnayati, 2018, yang berjudul “Sistem Peramalan Penjualan Sepeda Motor Menggunakan Metode *Trend Projection* Pada PT. Ud Prima Nusantara”. Tujuan dari penelitian ini yaitu menghasilkan aplikasi sistem peramalan ini hanya meramalkan jumlah penjualan sepeda motor dengan menggunakan Metode *Trend Projection*. Metode *Trend Projection* mampu menghasilkan tingkat kesalahan peramalan RMSE sebesar 247,4524807. Sistem peramalan penjualan sepeda motor menggunakan metode *Trend Projection* dapat dipergunakan untuk meramalkan penjualan honda di periode yang akan datang pada berdasarkan data penjualan tahun sebelumnya serta plikasi sistem peramalan penjualan sepeda motor meramalkan data penjualan peramalan dalam jangka perbulan dengan data 5 tahun.
2. Pada jurnal Billy Eden William Asrul, 2018, yang berjudul “ Implementasi Metode *Trend Projection* Untuk Peramalan Persediaan Ikan Hias Air Tawar Studi Kasus: Batara Indo Aquatic” Dengan adanya sistem Pendukung Keputusan Peramalan Persediaan ini dapat membantu pihak Batara Indo Aquatic dalam menentukan jumlah persediaan ikan untuk periode yang akan

datang. Hal ini dikarenakan nilai penjualan 2017 yang diramalkan hampir mendekati data penjualan sebenarnya. Sistem ini memberikan kemudahan bagi customer untuk mengetahui jenis ikan, harga ikan yang dijual, serta informasi umum tentang Batara Indo Aquatic.

3. Pada jurnal Maftahatul Hakimah, 2017, yang berjudul “ rancang bangun aplikasi peramalan Persediaan barang dengan metode *Trend Projection*”. Tujuan dari penelitian ini yaitu digunakan untuk peramalan dalam jangka pendek biasanya satu bulan kedepan. Data dari hasil perhitungannya berfluktuasi disekitar nilai mean (rata-rata) yang tetap tanpa trend dan pertumbuhan. Dengan dibuatnya sistem aplikasi dari *Single Exponential Smoothing*, diharapkan pemilik toko Qitaz akan lebih mudah untuk menyimpan data penjualan gamisnya, menghitung nilai peramalan untuk periode berikutnya dan membuat laporan hasil perhitungan peramalan secara cepat dan akurat. Sehingga tujuan dan manfaat dari penelitian ini yaitu pendapatan optimal dan efisiensi biaya dapat tercapai

II.2. Uraian Teoritis

II.2.1. Sistem Informasi

Sistem informasi merupakan serangkaian komponen berupa manusia, prosedur data, dan teknologi (seperti komputer) yang digunakan untuk melakukan sebuah proses untuk menghasilkan informasi yang bernilai untuk pengambilan keputusan.

Sistem informasi berfungsi menghasilkan informasi untuk memenuhi kebutuhan aktivitas organisasi. Informasi yang bernilai tinggi dihasilkan oleh suatu sistem informasi yang efektif dan efisien. Untuk itu, sistem informasi yang efektif dan efisien menghendaki intervensi manajemen secara tepat. (Tomi Loveri, 2018)

Definisi Sistem Informasi menurut Azhar Susanto adalah sebagai berikut :
“Sistem informasi adalah kumpulan dari subsistem apapun baik fisik ataupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berarti dan berguna”. (Rini Asmara : 2017)..

Menurut Laudon, *an information system can be defined technically as a set of interrelated components that collect (or retrieve), process, store, and distribute information to support decision making and control in an organization.* Sistem informasi adalah kumpulan komponen yang saling berhubungan dalam mengumpulkan, memproses, menyimpan, menyediakan dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan pengendalian di dalam organisasi. (Rini Asmara : 2017).

II.2.1.1 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen sebagai berikut :

1. Blok Masukan

Input mewakili data yang masuk ke dalam sistem informasi. *Input* disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model

Blok model ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran (*Output Block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi

Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 bagian utama, yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

5. Blok Basis Data

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan DBMS (*Database Management System*).

6. Blok kendali

Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana alam, api, temperature, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk menyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi. (Tomi Loveri, 2018)

II.2.2. Peramalan

Peramalan merupakan suatu keputusan tentang kemungkinan masa yang akan datang yang didasarkan fakta-fakta ekonomi sekarang dan sejarah masa lalu. Sebelum melakukan peramalan harus diketahui lebih dulu persoalan dalam pengambilan keputusan. Dengan adanya peramalan manajemen dapat segera menyiapkan langkah paling baik dalam mengatasi permasalahan dan dapat bersiap-siap memanfaatkan situasi jika terjadi perkembangan yang lebih baik dalam peramalan. Peramalan yang menjadi titik tolak bagi tindakan perusahaan adalah ramalan penjualan. Pengertian ramalan penjualan adalah proyeksi teknis permintaan konsumen yang potensial pada masa yang akan datang dengan menggunakan berbagai pendekatan. Hal ini sangat berhubungan dengan kegiatan produksi. Dalam proses produksi kegiatan yang dilaksanakan dalam peramalan adalah untuk menentukan jumlah permintaan konsumen terhadap suatu produk yang dilakukan pada awal proses perencanaan dan untuk kepentingan proses

pengendalian produksi. Oleh sebab itu perangkat lunak sangat dibutuhkan, untuk mendukung peramalan permintaan konsumen untuk mempercepat, mempermudah dan menghindari hitungan dengan tangan yang kemungkinan terjadi kesalahan. (Salman, Alfarisi, 2017)

II.2.3. Metode *Trend Projection*

Metode peramalan dengan proyeksi trend ini adalah mencocokkan garis *trend* ke rangkaian titik data historis kemudian memproyeksikan garis tersebut ke masa depan dengan horizon waktu menengah dan panjang. Metode *proyeksi* pada penelitian ini melihat trend pada garis lurus (linier). Pendekatan yang tepat untuk trend linier adalah metode kuadrat terkecil (*least square*). Metode kuadrat terkecil adalah langkah untuk menentukan garis lurus yang paling sesuai untuk pola data yang muncul. Tujuan utama metode ini adalah meminimalkan jumlah kuadrat kesalahan atau selisih dari persamaan regresi linier dengan masing-masing data sebenarnya (Maftahatul Hakimah : 2 : 2015)

Metode *Trend Projection* adalah suatu metode peramalan serangkaian waktu yang sesuai dengan garis trend terhadap serangkaian titik-titik data masa lalu, kemudian diproyeksikan ke dalam peramalan masa depan. Metode ini merupakan garis *trend* untuk persamaan matematis. Pada kenyataannya, anggapan bahwa *trend* dapat diwakili oleh beberapa fungsi sederhana seperti garis lurus sepanjang periode untuk waktu berkala yang diamati jarang ditemukan. Ada tiga *trend* yang digunakan untuk meramalkan pergerakan keadaan pada masa yang akan datang, yaitu :

1. *Trend Linier*, yaitu suatu trend yang kenaikan atau penurunan nilai yang akan diramalkan naik atau turun secara linier. Dengan persamaan *trend linear* $Y = a + bX$
2. *Trend Parabolik* (kuadrat) disebut juga *trend* kuadratis, merupakan “salah satu bentuk *trend non linear* yaitu *trend* yang variabel X-nya berpangkat paling tinggi 2 .”. Hubungan variabel dependen dan independen adalah kuadrat, dengan persamaan untuk *trend* kuadrat adalah $Y = a + bX + cX^2$.
3. *Trend Eksponensial*, yaitu sebuah trend yang nilai variabel tak bebasnya naik secara berlipat ganda atau tidak *linier*. Analisis *trend* yang digunakan secara umum untuk model *trend* pertumbuhan eksponensial dengan persamaan *trend eksponensial* adalah $Y' = a(1 + b)^x$. (Aleksius Madu : 2 : 2016)

Metode *Trend Projection* bisa disebut juga metode tren garis lurus.

Adapun persamaan *trend linier* dapat ditulis sebagai berikut :

$$Y' = a + bX \dots \dots \dots (1)$$

Dimana :

Y' = data berkala (time series data),

a dan b = konstanta

X = waktu (hari, minggu, bulan, tahun)

Pada umumnya yang diberi nilai 0 adalah variable waktu yang letaknya ditengah Untuk mencari nilai variable waktu adalah sebagai berikut :

1. Untuk nilai variable waktu (= X) adalah nol (0)

- a. Untuk nilai n ganjil adalah :

$$n = 2k + 1 \quad k = \frac{n-1}{2} \dots \dots \dots (2)$$

$$\frac{Xk+1}{2} = 0 \dots \dots \dots (3)$$

b. Untuk nilai n genap adalah :

$$n = 2k, k = \frac{n}{2} \dots\dots\dots (4)$$

$$\frac{xk + (k+1)}{5} = 0 \dots\dots\dots (5)$$

(Maftahatul Hakimah : 2 : 2017).

Untuk nilai n ganjil adalah :

$$n = 2k + 1, k = \frac{n-1}{2} \dots\dots$$

$$\frac{xk+1}{5} = 0$$

Untuk nilai n genap adalah :

$$n = 2k, k = \frac{n}{2}$$

$$\frac{xk+(k+1)}{2} = 0$$

Dalam mencari hasil x

- (6-1)/2 = 2.5
- (5-2)/2 = 1.5
- (4-3)/2 = 0.5
- (3-4)/2 = -0.5
- (2-5)/2 = -1.5
- (1-6)/2 = -2.5

Untuk nilai x yang ke 7 diperoleh hasil 3,5 dimana periode ke 7 / 2.

Dalam mencari hasil xy = nilai x * nilai y

- 2,5 * 1025 = -2562,5
- 1,5 * 1039 = -1558,5
- 0,5 * 1045 = -522,5
- 0,5 * 1031 = 515,5
- 1,5 * 1075 = 1612,5
- 2,5 * 1088 = 2720

Dalam mencari hasil x² (x pangkat 2) = nilai x²

- 2.5² = 6,25
- 1,5² = 2,25
- 0,5² = 0,25
- 0,5² = 0,25
- 1,5² = 2,25
- 2,5² = 6,25

Sumber (Evi Dewi Sri Mulyani : 2018)

II.2.4. ASP (*Active Server Page*)

ASP.NET adalah arsitektur yang digunakan untuk mengembangkan *website* dinamis. Untuk mengembangkan *website* berbasis arsitektur ini membutuhkan versi *Microsoft .NET Framework 3.5* atau lebih tinggi. Bahkan *website* yang dikembangkan menggunakan ASP.NET MVC memiliki kelebihan seperti SEO (*Search Engine Optimizer*) *friendly*. Artinya arsitektur ini memiliki kelebihan dasar yaitu salah satunya dapat di-*index* dengan mudah oleh *search engine* seperti Google, Yahoo, dan Bing. ASP.NET dikembangkan oleh Microsoft yang di-*release* pertama kali pada Januari 2002 dan berlisensi *open source*. *Framework* ini dibangun menggunakan CLR (*Common Language Runtime*) dan dapat menulis *code* untuk ASP.NET menggunakan bahasa pemrograman berbasis .NET seperti C# dan Visual Basic. (Firyan Aula Juyuspan : 2017)

II.2.5. Database

Menurut Sutabri dalam buku Sistem Informasi Manajemen: 'Database adalah suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data (*controlled redundancy*'. (Fitri Ayu : 2018)

Beberapa model *Database* diantaranya :

1. *Object based* data model merupakan himpunan data dan prosedur atau relasi yang menjelaskan hubungan logis antar data dalam suatu *database* berdasarkan objek datanya.

2. *Record Based* data model. Model ini berdasarkan pada *record* untuk menjelaskan kepada *user* tentang hubungan logis antardata dalam *database*.

(Fitri Ayu : 2018)

II.2.5.1 SQL Server

SQL Server merupakan suatu *Relational Database Management Systems* (RDBMS) yang digunakan untuk menyimpan data. Data yang disimpan pada *database* bisa dalam skala kecil maupun besar. Selain itu, penyajiannya merupakan penyajian pada level fisik karena kita akan menyimpan langsung data pada *database* dengan kondisi yang sebenarnya, yaitu disimpan pada tabel apa, kolom mana, dan menggunakan tipe data saat penyimpanan. (Benardo, 2015).

Database merupakan suatu tempat untuk menyimpan data. Pada sebuah *database* bisa terdapat satu atau lebih tabel dan *query*. Operasi yang biasanya dilakukan *database* berhubungan erat dengan pengaksesan tabel atau *query*. Pada *SQL Server 2008* terdapat fitur-fitur yang dapat mengembangkan performa dari *database* tersebut. Beberapa fitur tersebut, yaitu (Benardo, 2015) :

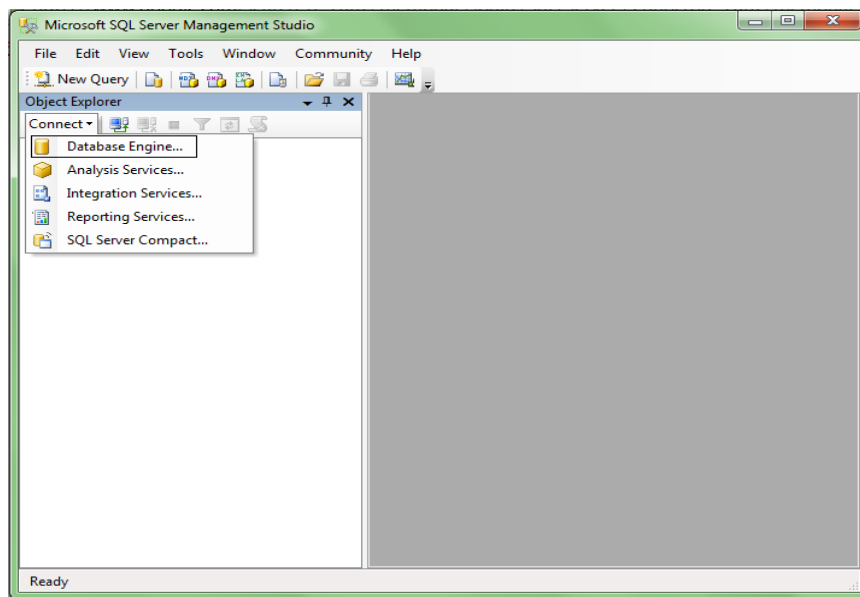
- 1) *Date Data Type*: Digunakan untuk menyimpan data tanggal saja sehingga akan menghemat *space* pada *server*.
- 2) *Data Compression*: Digunakan untuk melakukan compress data sehingga ukuran data yang disimpan dalam hal *space hardisk* akan lebih kecil.
- 3) *Sparse Column*: Digunakan untuk menyimpan data yang memiliki lebih banyak data NULL dengan lebih efisien.

- 4) *Row Constructor*: Digunakan untuk melakukan *insert* beberapa data sekaligus dengan satu perintah *INSERT*.
- 5) *Table-Valued Parameter*: Digunakan untuk melakukan parsing array pada bahasa pemrograman, dimana satu variable diberikan data-data yang akan diproses setelahnya.

Microsoft SQL server 2000 adalah Sistem manajemen basis data yang memakai perintah *Transact-SQL* untuk mengirim perintah dari Komputer client ke Komputer server. *Microsoft SQL Server 2000* dapat dijalankan pada NT 4.0 *Server* atau *Microsoft Windows 2000 Server*, selain itu dapat pula di install pada personal *desktop* di *Windows 2000 Profesional*, *Windows 98 Milenium*. Sistem akan dipergunakan oleh beberapa komputer (*client-server*) maka database yang dibangun adalah merupakan *database* yang berfungsi untuk menunjang hal tersebut untuk itu dalam penulisan ini dipakai database *Microsoft SQL Server 2000*, selain itu *database* ini juga *compatible* dengan bahasa pemrograman yang digunakan. Secara teoritis, program *SQL* yang sudah *terinstall* pada komputer dapat menampung *32.767 database* dan terdapat lebih dari *2 billion objek*. *SQL Server* adalah sistem manajemen *database relasional* (RDBMS) yang dirancang untuk aplikasi dengan arsitektur *client/server*. Istilah *client*, *server*, dan *client/server* dapat digunakan untuk merujuk kepada konsep yang sangat umum atau hal yang spesifik dari perangkat keras atau perangkat lunak. Pada level yang sangat umum, sebuah *client* adalah setiap komponen dari sebuah sistem yang meminta layanan atau sumber daya (*resource*) dari komponen sistem lainnya.

Sedangkan sebuah *server* adalah setiap komponen sistem yang menyediakan layanan atau sumber daya ke komponen sistem lainnya.

SQL Server menggunakan tipe dari *database* yang disebut *database relasional*. *Database relasional* adalah *database* yang digunakan sebuah data untuk mengatur atau mengorganisasikan kedalam tabel. Tabel-tabel adalah alat bantu untuk mengatur atau mengelompokan data mengenai subyek yang sama dan mengandung informasi dan kolom dan baris. Tabel-tabel saling berhubungan dengan mesin *database* ketika dibutuhkan. *SQL Server* mendukung beberapa tipe data yang berbeda, termasuk untuk karakter, angka, tanggal (*datetime*) dan uang (*money*), *SQL Server* digunakan untuk menggambarkan model dan implementasi pada *database*. (Hendra Kurniawan, 2017).



Gambar II.1. Tampilan *SQL Server*
(Sumber : Hendra Kurniawan, 2017)

II.2.7 UML (*Unified Modelling Language*)

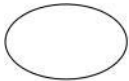

Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem.



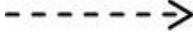

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

II.2.7.1. *Use Case Diagram*

Use case diagram merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *Use Case Diagram* yaitu:

Tabel II.1. Simbol *Use Case Diagram*

Gambar	Keterangan
	<i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukaran pesan antara unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal nama <i>Use Case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mendefinisikan aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem biasa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi

	dengan <i>use case</i> tetapi tidak memiliki <i>control</i> terhadap <i>use case</i> .
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidentifikasi bila aktor berinteraksi secara pasif dengan sistem
	<i>Include</i> , merupakan didalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat dipenuhi.

(Sumber : Ade Hendini, 2016)

II.2.7.2. Class Diagram

Class Diagram merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas dalam model *desain* dari suatu sistem juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

Tabel II.2. Simbol Class Diagram




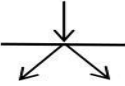
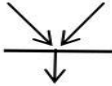
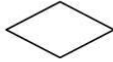

Multiplicity	Penjelasan
1	Satu dan hanya satu
0...*	Boleh tidak ada atau 1 atau lebih
1...*	1 atau lebih
0..1	Boleh tidak ada maksimal 1
n..n	Batasan antara Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Ade Hendini, 2016)

II.2.7.3. Activity Diagram

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram* yaitu :

Tabel II.3. Activity Diagram




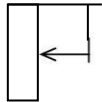


Gambar	Keterangan
	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End Point</i> , akhir aktifitas
	<i>Activities</i> menggambarkan suatu proses atau kegiatan bisnis.
	<i>Fork</i> (Percabangan) digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan parallel menjadi satu.
	<i>Join</i> (Penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision points</i> menggambarkan pilihan untuk pengambilan keputusan, true, false
	<i>Swimlane</i> pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa

(Sumber : Ade Hendini, 2016)

II.2.7.4. Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram* yaitu :

Tabel II.4. Simbol Sequence Diagram

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari system yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal system dan menjadi landasan untuk menyusun basis data
	<i>Boundary Class</i> berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem. Seperti tampilan <i>formentry</i> dan <i>form cetak</i> .
	<i>Control Class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar kelas.
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.

(Sumber : Ade Hendini, 2016)