BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Penelitian terkait bertujuan untuk mengambil beberapa referensi jurnal terkait yang digunakan dalam mendukung penelitian publikasi ilmiah dalam junal local yaitu:

Tabel II.1. Keaslian Penelitian

No	Penulis	Judul	Hasil Penelitian
1	Rezky Rizaldi (2017)	Implementasi Metode *Euclidean Distance* Untuk Rekomendasi Ukuran Pakaian Pada Aplikasi Ruang Ganti Virtual	Sistem memberikan fitur bagi calon pembeli untuk mencoba model pakaian lainnya. Sistem dapat memperlihatkan baju virtual 3 dimensi yang tetap melekat pada badan calon pembeli, ketika melakukan rotasi ke kanan 900, ke kiri 900, balik kanan 1800 dan balik kiri 1800. Hasil uji coba sistem rekomendasi ukuran pakaian ini akan berjalan secara optimal jika pengaturan ketinggian *kinect* sebesar 55 cm dari tanah. Untuk ketinggian *kinect* 55cm, 65cm dan 75 cm dari tanah, sistem ini mampu menyajikan kesesuaian rekomendasi ukuran dibandingkan dengan ukuran asli dari calon pembeli sebesar 70%.
2	Arfinisa Pratidina (2017)	Implementasi Pengolahan Citra Untuk Mendapatkan Informasi Mengenai Citra Baret Tni Dengan Menggunakan Algoritma Euclidean Distance	Proses untuk memunculkan informasi yaitu menggunakan selisih antara nilai histogram dari citra yang di input dengan nilai histogram dari citra database. Untuk proses memunculkan nama dan informasi dari baret TNI membutuhkan waktu lama karena proses yang diambil adalah nilai histogram yaitu terdiri dari nilai Red, nilai Green, dan nilai Blue. Pembuatan aplikasi ini dengan menggunakan microsoft visual studio 2008 dengan bahasa pemograman C#. Aplikasi ini dibuat dalam beberapa tahap, dimulai dengan membuat Flowchart untuk di setiap menu, membuat struktur navigasi dan merancang

			interface setiap halaman dari aplikasi serta melakukan ujicoba terhadap aplikasi.
3	Dwi Nugraheny (2018)	Sistem Informasi Geografis Pemilihan Paket Dengan Metode Eclidean Distance	Metode nilai jarak guna kesamaan atau kemiripan ciri suatu citra (kasus deteksi awan *cumulonimbus* menggunakan *principal component analysis)"* Pada penelitian ini, akan dilakukan perbandingan dari beberapa metode pengukuran jarak diantaranya menggunakan jarak *Euclidean*, *Manhattan/City Block Distance*, *Mahalanobis* yang akan di implementasikan pada deteksi citra awan *Cumulonimbus* menggunakan *Principal Component Analysis* (PCA). Hasil ratarata persentase keakuratan nilai kemiripan citra awan *Cumulonimbus* menggunakan metode jarak Euclidean adalah 93 persen dan jarak *Manhattan/City Block Distance* adalah 90 persen, sedangkan metode jarak *Mahalanobis* adalah 50 persen.

II.2. Uraian Teoritis

Penelitian yang dilakukan dengan cara langsung pada pencarian lokasi distributor pupuk adalah menggunakan beberapa aplikasi atau bahasa program, yaitu :

II.2.1. Sistem Informasi Geografis

SIG adalah sistem komputer yang digunakan untuk mengumpulkan, memeriksa, mengintegrasikan, dan menganalisa informasi-informasi yang berhubungan dengan permukaan bumi. Pada dasarnya, istilah sistem informasi geografi merupakan gabungan dari tiga unsur pokok yaitu sistem, informasi, dan geografi. Dengan demikian, pengertian terhadap ketiga unsur-unsur pokok ini

akan sangat membantu dalam memahami SIG. Dengan melihat unsur-unsur pokoknya, maka jelas SIG merupakan salah satu sistem informasi. SIG merupakan suatu sistem yang menekankan pada unsur informasi geografi. Istilah "geografis" merupakan bagian dari spasial (keruangan). Kedua istilah ini sering digunakan secara bergantian atau tertukar hingga timbul istilah yang ketiga, geospasial. Ketiga istilah ini mengandung pengertian yang sama di dalam konteks SIG. Penggunaan kata "geografis" mengandung pengertian suatu persoalan mengenai bumi: permukaan dua atau tiga dimensi. Istilah "informasi geografis" mengandung pengertian informasi mengenai tempat-tempat yang terletak di permukaan bumi, pengetahuan mengenai posisi dimana suatu objek terletak di permukaan bumi, dan informasi mengenai keterangan-keterangan (atribut) yang terdapat di permukaan bumi yang posisinya diberikan atau diketahui. (Koko Mukti Wibowo : 2015)

II.2.2. *JQuery*

JQuery Mobile adalah framework yang berbasis *HTML/CSS* dan *javascripct* untuk aplikasi *web y*ang dijalankan diprangkat *mobile*. Dalam membangun web biasa menggunakan *framework jquery*, begitu juga apabila ingin membuat aplikasi web untuk perangkat mobile (*smartphone/tablet pc*) bisa digunakan *jquery mobile.* Penggunaan *jquery mobile* akan menghasilkan tampilan yang menyesuaikan dengan lebar layar. *Script jquery mobile* merupakan bawaan dari framework *jquery mobile*, seperti *jqury.mobile-1.orc1.min.css* dan *styles.css.* Untuk menggunakan *jQuery Mobile*, pertama perlu memasukkan tiga *file* yaitu.

1. *JQuery CSS Mobile file (jquery.mobile-1.0a1.min.css)*
2. Perpustakaan *jQuery (jquery1.4.3.min.js)*
3. *Mobile* perpustakaan *jQuery (jquery.mobile-1.0a1.min.js).* (Koko Mukti Wibowo : 2015)

II.2.3. *Euclidean Distance*

Euclidean distance adalah metode yang digunakan untuk mengukur jarak antara 2 titik yang berbeda. Metode ini akan digunakan untuk mengukur lebar punggung calon pembeli dengan memanfaatkan data *skeleton* yang didapatkan dari hasil kalibrasi. Berikut adalah persamaan *Euclidean Distance* untuk titik yang mempunyai ruang 3 dimensi.

d skeleton = $\sqrt{(x1-x2)^2}+(y1-y2)^2\ +(z1-z2)^2$

Keterangan :

d*skeleton* = lebar punggung calon pembeli

x1 = koordinat x pada titik *skeleton_left_shoulder*.

x2 = koordinat x pada titik *skeleton_right_shoulder*. y1 = koordinat y pada titik *skeleton_left_shoulder*.

y2 = koordinat y pada titik *skeleton_right_shoulder*.

z1 = koordinat z pada titik *skeleton_left_shoulder*.

z2 = koordinat z pada titik *skeleton_right_shoulder*. (Rezky Rizaldi : 2017)

III.2. Studi Kasus

Rumus Euclidean distance dapat dilihat pada persamaan III.1 :

$$d_{ij} = \sqrt{\sum_{k=1}^{n}(x_{ik} - x_{jk})^2} \quad \text{..... (III.1)}$$

Dengan :

d(i,j) : jarak dokumen ke-i ke dokumen ke-j

xi(n) : kata ke n di dokumen ke-i.

xj(n) : kata ke n di dokumen ke-j

Rumus Euclidean Distance
d = √(φ1-φ2)² + (λ1-λ2)² . S
S: Satuan derajat (1 derajat bumi = 111.319 km)
1 derajat (lintang/bujur) = 111.319 Km = 111319 meter

Koordinat peta terdiri dari titik (x,y)
dimana sumbu x adalah longitudinal
dan sumbu y adalah latitude.

φ (phi) lambang untuk latitude
λ (lambda) lambang untuk longitudinal.

Implementasi di program:

```
var S = 111.319;
var φ1 = lat1;
var λ1 = lon1;
var φ2 = lat2;
var λ2 = lon2;

var d = Math.sqrt(Math.pow(φ1-φ2,2) + Math.pow(λ1-λ2,2)) * S;
```

Misalkan kumpulan Koordinat dari sebuah rute adalah:
Koordinat Rute [(3.61206,98.67675) (3.61222,98.67674) (3.61264,98.67671) (3.61418,98.67633) (3.61494,98.67612)]

1. Euclidean[(3.612060,98.676750) → (3.612060,98.676740)]
   = √(φ1-φ2)² + (λ1-λ2)² . S
   = √(3.61206-3.61222)² + (98.67675-98.67674)² × 111.319
   = 0.01784579328186677

2. Euclidean[(3.612220,98.676740) → (3.612220,98.676710)]
   = √(φ1-φ2)² + (λ1-λ2)² . S
   = √(3.61222-3.61264)² + (98.67674-98.67671)² × 111.319
   = 0.04687309861336347

3. Euclidean[(3.612640,98.676710) → (3.612640,98.676330)]
   = √(φ1-φ2)² + (λ1-λ2)² . S
   = √(3.61264-3.61418)² + (98.67671-98.67633)² × 111.319
   = 0.17657312966233973

4. Euclidean[(3.614180,98.676330) → (3.614180,98.676120)]
   = √(φ1-φ2)² + (λ1-λ2)² . S
   = √(3.61418-3.61494)² + (98.67633-98.67612)² × 111.319
   = 0.08777275497205393

Jarak total = jarak1+ jarak2+ jarak3 + jarak4 = 0.3291

II.2.4. *Global Positioning System* (GPS)

Google Maps adalah layanan pemetaan berbasis *Android* service yang disediakan oleh Google dan bersifat gratis, yang memiliki kemampuan terhadap banyak layanan pemetaan berbasis *Android*. *Google Maps* juga memiliki sifat server side, yaitu peta yang tersimpan pada server Google dapat dimanfaatkan oleh pengguna. *Google Maps* API adalah suatu library yang berbentuk javascript yang berguna untuk memodifikasi peta yang ada di *Google Maps* sesuai kebutuhan. Untuk membangun aplikasi yang memanfaatkan *Google Maps* di desktop dan mobile device maka akan digunakan *Google Maps* Javascript API v3 yang memiliki keunggulan lebih cepat dari versi sebelumnya. (Koko Mukti Wibowo : 2015)

II.2.5. Basis Data (*Database*)

Pangkalan data atau basis data (*database*) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil kueri (*query*) basis data disebut sistem manajemen basis data (*database management system*, DBMS). Sistem basis data dipelajari dalam ilmu informasi. Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya: penjelasan ini disebut skema. Model yang umum digunakan sekarang adalah model relasional, yang mewakili semua informasi dalam bentuk tabel-tabel yang saling berhubungan dimana setiap tabel terdiri dari baris dan kolom. Model yang lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar tabel (Neni Purwati dan Hendra Kurniawan, 2015 : 50).

II.2.6. Normalisasi

Normalisasi merupakan sebuah teknik dalam logical desain sebuah basis data / database, teknik pengelompokkan atribut dari suatu relasi sehingga membentuk struktur relasi yang baik tanpa redudansi. Tujuan normalisasi adalah mengorganisasikan data kedalam tabel-tabel untuk memenuhi kebutuhan pemakai, menghilangkan kerangkapan data, mengurangi kompleksitas, mempermudah modifikasi data. (Mukhlisulfatih Latief : 2016)

1. Proses Normalisasi

   a. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu kebeberapa tingkat.

   b. Apabila tabel yang diuji belum memenuhi persyaratan tertentu maka tabel tersebut pelu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal.

2. Tahapan Normalisasi :

   1) Bentuk tidak normal : Menghilangkan perulangan grup.

Tabel II.2. Contoh bentuk tidak normal (Unnormal)

No-Mhs	Nama Mhs	Jurusan	Kode-MK	Nama-MK	Kode Dosen	Nama Dosen	Nilai
2683	Welli	MI	M1350	Manajemen DB	B104	Ati	A
			M1465	Analisis Perc. Sistem	B317	Dita	B
5432	Bakti	AK	M1350	Manajemen DV	B104	Ati	C
			Akn201	Akuntansi	D310	Lia	B
			MKT300	Dasar Pemasaran	B212	Lola	A

Sumber : Mukhlisulfatih Latief : 2016

2) Bentuk Normal pertama (1NF) : Menghilangkan ketergantungan sebagian.

Yaitu : suatu relasi dikatakan sudah memenuhi bentuk normal kesatu bila setiap data bersifat atomik yaitu setiap irisan baris dan kolom hanya mempunyai satu nilai data.

Tabel II.3. Contoh Bentuk Normal Pertama (1NF)

No-Mhs	Nama Mhs	Jurusan	Kode-MK	Nama-MK	Kode Dosen	Nama Dosen	Nilai
2683	Welli	MI	M1350	Manajemen DB	B104	Ati	A
2683	Welli	MI	M1465	Analisis Perc. Sistem	B317	Dita	B
5432	Bakti	AK	M1350	Manajemen DV	B104	Ati	C
5432	Bakti	AK	Akn201	Akuntansi	D310	Lia	B
5432	Bakti	AK	MKT300	Dasar Pemasaran	B212	Lola	A

Sumber : Mukhlisulfatih Latief : 2016

3) Bentuk Normal kedua (2NF) : Menghilangkan ketergantungan transitif.

Yaitu : suatu relasi dikatakan sudah memenuhi bentuk normal kedua bila relasi tersebut sudah memenuhi bentuk normal kesatu dan atribut yang bukan key sudah tergantung penuh terhadap key-nya.

Tabel II.4. Contoh Bentuk Normal Kedua (2NF)

Kode-MK	Nama-MK	Kode Dosen	Nama Dosen
M1350	Manajemen DB	B104	Ati
M1465	Analisis Perc. Sistem	B317	Dita
M1350	Manajemen DV	B104	Ati
Akn201	Akuntansi	D310	Lia
MKT300	Dasar Pemasaran	B212	Lola

Sumber : Mukhlisulfatih Latief : 2016

4) Bentuk Normal ketiga (3NF) : Menghilangkan anomali-anomali hasil dari ketergantungan fungsional. Yaitu : suatu relasi dikatakan sudah memenuhi bentuk normal ketiga bila relasi tersebut sudah memenuhi bentuk normal kedua dan atribut yang bukan key tidak tergantung transitif terhadap key-nya.

Tabel II.5. Contoh Tabel Mahasiswa Dan Tabel Kuliah (3NF)

No_Mhs	Nama Mhs	Jurusan
2683	Welli	MI
5432	Bakti	AK

Sumber : Mukhlisulfatih Latief : 2016

II.2.7. MySQL

MySQL (*My Structure Query Language*) merupakan sebuah program pembuat database yang bersifat *Open Source,* artinya semua orang dapat menggunakannya dan dapat dijalankan pada semua *platform* baik *Windows* maupun linux. *MySQL* juga merupakan sebuah perangkat lunak sistem manajemen basis data *SQL* yang bersifat jaringan sehingga dapat digunakan intuk aplikasi multi *user*. *MySQL* juga sering dikenal dengan nama sistem manajemen *database* relasional. Suatu *database* relasional menyimpan data dalam table yang terpisah. Tabel -table tersebut terhubung oleh suatu relasi terdefinisi yang memungkinkan Imemperoleh kombinasi data dari beberapa table dalam suatu permintaan. Untuk administrasi *database,* seperti pembuatan *database,* pembuatan tabel, dan sebagainya dapat digunakan aplikasi berbasis web seperti *PHP MyAdmin* dengan aplikasi *XAMPP*. (Saipul Anwar, 2016**)**

II.2.8. XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama *XAMPP* merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam *GNU General Public License* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis. Untuk mendapatkanya dapat men-*download* langsung dari *web* resminya. (Randi V Palit, 2015 : 2).

II.2.9. PHP

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah skripsi HTML. Bahasa PHP dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, *Java*, dan *Perl* serta mudah untuk dipelajari. PHP merupakan bahasa *scripting server – side*, dimana pemrosesan datanya dilakukan pada sisi *server*. Sederhananya, *server* yang akan menterjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada *client* yang melakukan permintaan. Adapun pengertian lain PHP adalah *akronim* dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasiskan kode – kode *(script)* yang digunakan untuk

mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode HTML". (Astria Firman, 2016 : 30).

II.2.10. *UML* (*Unified Modelling Language*)

Menurut (Windu Gatta, 2013) Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak.

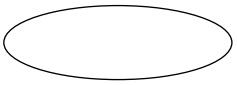
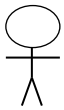
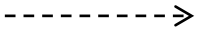
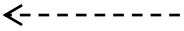
UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem.

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

II.2.8.1. *Use Case Diagram*

Use Case Diagram adalah gambaran fungsionalitas dari suatu sistem, sehingga *customer* atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun. Simbol-simbol yang digunakan dalam *use case* diagram dapat dilihat pada tabel II.6 dibawah ini :

Tabel II.6. Simbol *Use Case*

Gambar	Keterangan	Deskripsi
	*Use case*	Menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama *use case*.
	Aktor	Sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasikan aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem.
	Asosiasi	Penghubung antara aktor dan *use case*, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.
	Asosiasi	Penghubung antara aktor dan *use case* yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.
	*Include*	Merupakan di dalam *use case* lain (*required*) atau pemanggilan *use case* oleh *use case* lain, contohnya adalah pemanggilan sebuah fungsi program.
	*Extend*	Merupakan perluasan dari *use case* lain jika kondisi atau syarat terpenuhi.

***(*Sumber : Ade Hendini, 2016)**

II.2.8.2. *Class Diagram*

Class Diagram menggambarkan struktur dan deskripsi *class*, *package* dan *object* beserta hubungan satu sama lain seperti pewarisan, asosiasi, dan lain-lain Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality* (Ade Hendini, 2016).

Tabel II.7. *Multiplicity 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

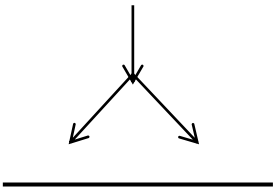
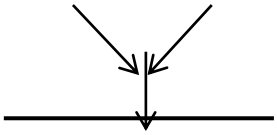
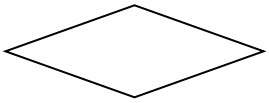
(Sumber : Ade Hendini, 2016)

II.2.8.3. *Activity Diagram*

Activity Diagram menggambarkan sebagai alur aktifitas dalam sistem yang sedang dirancang. Bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. Sebuah aktifitas dapat direalisasikan oleh satu *use case* atau lebih. Aktifitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana *actor* menggunakan sistem untuk melakukan aktivitas. *Decision* digunakan untuk menggambarkan *behaviour* pada kondisi teretentu. Untuk mengilustrasikan proses-proses *parallel* (*fork* dan join) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertical. *Activity* diagram dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktifitas tertentu (I Made Budi Adnyana, 2016).

Simbol-simbol yang digunakan dalam *activity diagram* dapat dilihat pada tabel II.8 dibawah ini:

Tabel II.8. Simbol *Activity Diagram*

Gambar	Keterangan	Deskripsi
	Start point	Diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	End point	Akhir aktifitas.
	Activites	Menggambarkan suatu proses/kegiatan bisnis.
	*Fork* (Percabangan).	Digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.
	*Join* (penggabungan)	Digunakan untuk menunjukkan adanya dekomposisi.
	*Decision Points*	Menggambarkan pilihan untuk pengambilan keputusan, *true, false.*
	Swimlane	Pembagian *activity* diagram untuk menunjukkan siapa melakukan apa.

(Sumber : Ade Hendini, 2016)

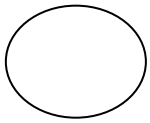
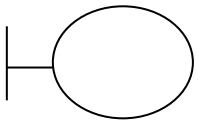
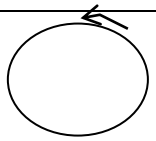
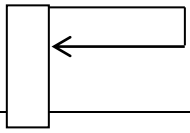
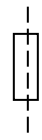
II.2.8.4. *Sequence Diagram*

Sequence Diagram menggambarkan interaksi atar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* terdiri atas dimensi vertical (waktu) dan dimensi horizontal (obyek– obyek yang terkait). *Sequence Diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah

yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu (I Made Budi Adnyana, 2016).

Simbol-simbol yang digunakan dalam *sequence diagram* dapat dilihat pada tabel II.9 dibawah ini :

Tabel II.9. Simbol *Sequence Diagram*

Gambar	Keterangan	Deskripsi
	*Entity Class*	Merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	*Boundary Class*	Berisi kumpulan kelas yang menjadi *interface* atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan *form* cetak.
	*Control class*	Suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	Message	Simbol mengirim pesan antar *class*.
	*Recursive*	Menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	*Activation*	Mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	Lifeline	Garis titik-titik yang terhubung dengan objek, sepanjang *lifeline* terdapat *activation*.

(Sumber : Ade Hendini, 2016)