

BAB II

LANDASAN TEORI

II.1 Penelitian Terkait

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

Saputra, dkk (2019), melakukan penelitian yang berjudul Implementasi Metode *Profile Matching* Untuk Mengetahui Supplier Terbaik Pada PT. Lautan Luas Medan. PT. Lautan Luas Medan kesulitan dalam memilih supplier dengan kelebihanannya masing-masing. Proses pemilihan supplier ini menggunakan metode Profile Matching, dengan tahapan yaitu menentukan kriteria, perhitungan pemetaan gap, melakukan pembobotan, perhitungan core factor yang dan secondary factor, perhitungan nilai total, dan perhitungan untuk menentukan perankingan. Kriteria pemilihan supplier pada PT. Lautan Luas Medan adalah pengiriman, pelayanan, produk, kualitas dan harga. Pemilihan supplier dengan metode *Profile Matching* ini dibangun dengan menggunakan bahasa pemrograman Visual Studio 2010 dan SQL Server 2008. Profile Matching merupakan solusi yang terbaik untuk memecahkan permasalahan dan membantu pimpinan dalam mengambil keputusan pemilihan supplier yang lebih obyektif pada PT. Lautan Luas Medan.

Setiadi, dkk (2020), melakukan penelitian yang berjudul Analisis Kinerja Dalam Pendukung Keputusan Pemilihan Vendor Pada Unit Bisnis Sistem Transportasi Di PT LEN Industri (Persero) Menggunakan Metode *Profile*

Matching. Pada penelitian ini digunakan Metode *Profile Matching* untuk menentukan vendor mana yang memiliki peringkat yang terbaik. Pengambilan keputusan akan dilakukan dengan membandingkan gap antara nilai alternatif dan kriteria yang diinginkan perusahaan. Berdasarkan hasil penelitian terhadap ketiga vendor yang maka vendor Supreme menempati peringkat pertama dengan nilai hasil akhir 5,18, disusul dengan vendor Leoni di peringkat kedua dengan nilai hasil akhir 5,1 dan vendor JLAPP menempati peringkat ketiga dengan nilai hasil akhir 4,97. Hasil pemeringkatan ini dijadikan sebagai masukan bagi perusahaan dalam menentukan mitra utama dan bagi vendor sebagai evaluasi guna meningkatkan kinerjanya sehingga dapat menjadi pilihan utama bagi perusahaan.

Sutinah (2019) melakukan penelitian yang berjudul Sistem Pendukung Keputusan Menggunakan Metode *Profile Matching* Dalam Pemilihan Salesman Terbaik. Pada penelitian ini perusahaan mengalami kesulitan dalam memilih salesman terbaik yang untuk dipromosikan menjadi sales manajer karena saat ini perusahaan dalam menentukan salesman terbaiknya yaitu hanya dengan melihat dari jumlah omset penjualan saja yang dihasilkan oleh masing-masing salesman, namun kurang memperhatikan faktor yang lain yang dapat mendukung penilaian, sehingga terjadi pihak manajemen salah memilih dan menentukan keputusan, dan juga dapat menimbulkan kesan pilih kasih terhadap salah satu salesman serta dapat menimbulkan kecemburuan sosial. Tujuan dari penelitian ini Mengetahui proses pemilihan sales terbaik yang sedang berjalan saat ini, memberikan alternatif pemecahan masalah dengan mengimplementasikan sistem pendukung keputusan metode profile matching dalam memilih salesman terbaik. Untuk

melakukan proses pemilihan *salesman* terbaik dengan menggunakan metode *profile matching* sehingga dapat membantu perusahaan dalam memecahkan masalah yang dihadapi dan mendapatkan suatu keputusan yang tepat. dari beberapa langkah yang telah diterapkan dengan menggunakan metode *profile matching* maka hasilnya didapatkan sebuah keputusan yang tepat dan sesuai harapan yaitu pihak manajemen dapat memilih salesmen terbaik yang nantinya akan dipromosikan menjadi sales manager.

Sudrajat (2019) melakukan penelitian yang berjudul *Pemilihan Pegawai Berprestasi dengan Menggunakan Metode Profile Matching*. Banyaknya pegawai menjadi kesulitan tersendiri dalam memilih pegawai berprestasi, tidak dipungkiri juga bahwa pemilihan pegawai berprestasi pun sering dilakukan tidak objektif. Untuk mengetahui pegawai berprestasi perlu diadakan penilaian kinerja terhadap pegawai. Melakukan suatu penilaian dalam pemberian penghargaan untuk pegawai berprestasi diantaranya menggunakan sistem pendukung keputusan dalam membantu pemecahan suatu masalah. Metode yang digunakan dalam melakukan pemberian penghargaan untuk pegawai berprestasi yaitu metode *profile matching*. Dengan adanya penerapan metode *profile matching* untuk pemilihan pegawai terbaik untuk memecahkan permasalahan yang ada pada saat proses pemilihan pegawai terbaik, agar tidak terjadi kesalahan dalam pengambilan keputusan. Diharapkan dengan upaya ini dapat memberikan nilai secara objektif terhadap pegawai dan membantu pimpinan dalam memberikan penilaian kinerja pegawainya.

Purwanto (2017) melakukan penelitian yang berjudul Penerapan Metode Profile Matching dalam Sistem Pendukung Keputusan Penilaian Kinerja Karyawan Pada PT. Hyundai Mobil Indonesia Cabang Kalimantan. Dalam penelitian ini menggunakan metode Profile Matching menjadi pilihan guna memberikan penilaian dan evaluasi kinerja karyawan untuk dipromosikan menjadi Leader. Ada tiga aspek dalam melakukan penilaian dan evaluasi tersebut, yaitu Aspek Kecerdasan terdiri dari tingkat kreatif, penalaran dan solusi, gagasan ide, konsentrasi, dan antisipasi. Aspek Sikap Kerja memiliki unsur ketelitian, tanggungjawab, pengendalian perasaan, motivasi untuk berprestasi, perencanaan, kehati-hatian. Terakhir Aspek Perilaku meliputi kepatutan, Kesungguhan, mandiri dan dinamis, Pengaruh. Ada perubahan Rangkaian dari kandidat setelah dilakukan pembobotan melalui perhitungan dengan metode Profile Matching dimana karyawan dengan kode K5 memiliki Rangkaian pertama dengan Nilai paling tinggi sebesar 6.55, sebelumnya perhitungan manual sederhana berada di posisi ke dua.

II.2 Landasan Teori

Landasan teori terdiri dari beberapa konsep beserta dengan definisi dan juga referensi yang akan digunakan untuk literatur ilmiah yang sangat relevan, teori yang digunakan untuk studi atau penelitian.

II.2.1. Sistem Pendukung Keputusan

Sistem pendukung keputusan adalah Sebuah aplikasi berupa Sistem Pendukung Keputusan (*Decision Support System*) mulai dikembangkan pada

tahun 1970. *Decision Support Sistem* (DSS) dengan didukung oleh sebuah system informasi berbasis komputer dapat membantu seseorang dalam meningkatkan kinerjanya dalam pengambilan keputusan. (Andani, 2019 : 166)

II.2.2. Distributor

Distributor adalah pihak yang menyalurkan barang dari produsen pada retailer. Umumnya distributor memiliki jaringan distribusi *ritel* yang luas. Jadi, Anda tinggal menitipkan produk Anda pada distributor untuk kemudian distributor lah yang langsung mendistribusikannya ke *retailer*. Harga yang diberikan umumnya lebih rendah 10-20% dari harga khusus (*wholesale price*) (Solihin, 2020 : 2).

II.2.3. Metode Profile Matching

Sistem adalah Suatu kesatuan prosedur atau komponen yang saling berkaitan satu dengan yang lainnya, bekerja bersama-sama sesuai dengan aturan yang diterapkan sehingga membentuk suatu tujuan yang sama, dimana dalam sebuah sistem bila terjadi satu bagian saja yang tidak bekerja atau rusak maka suatu tujuan bisa terjadi kesalahan hasilnya atau outputnya (Sutinah, 2017: 3).

Sistem Pendukung Keputusan merupakan suatu sistem interaktif yang mendukung keputusan dalam proses pengambilan keputusan melalui alternatif-alternatif yang diperoleh dari hasil pengolahan data, informasi dan rancangan model. Sistem Pendukung Keputusan merupakan penggabungan sumber-sumber kecerdasan individu dengan kemampuan komponen untuk memperbaiki kualitas

keputusan. Sistem Pendukung Keputusan juga merupakan sistem informasi berbasis komputer untuk manajemen pengambilan keputusan yang menangani masalah–masalah semi struktur (Angeline, 2018: 2).

Profile matching adalah sebuah mekanisme pengambilan keputusan dengan mengasumsikan bahwa terdapat tingkat minimal yang harus dipenuhi atau dilewati. (Angeline, 2018 : 2).

Langkah-langkah dalam penyelesaian perhitungan dengan menggunakan metode Profile Matching yaitu :

1. Aspek Penilaian.

Langkah pertama yang harus dilakukan yaitu menentukan aspek-aspek penilaian pada *core factor* (faktor utama) dan *secondary factor* (faktor kedua).

2. Pemetaan GAP Kompetensi

GAP kompetensi adalah perbedaan antara kriteria yang dimiliki seseorang dengan kriteria yang diinginkan. Rumus GAP kompetensi dapat dilihat pada persamaan II.1.

$$GAP = \text{Nilai Kriteria} - \text{Nilai Minimal} \dots\dots\dots(\text{II.1})$$

3. Pembobotan

Apabila pemetaan GAP sudah selesai dilakukan, maka hasil dari pemetaan tersebut diberi bobot nilai sesuai dengan patokan tabel bobot nilai GAP, seperti yang terlihat pada Tabel II.1.

Tabel II.1. Pemetaan GAP

No	Selisih	Bobot Nilai	Keterangan
1	0	5	Tidak Ada Selisih (Kriteria sesuai yang dibutuhkan)
2	1	4,5	Kriteria individu kelebihan 1 tingkat/level
3	-1	4	Kriteria individu kekurangan 1 tingkat/level
4	2	3,5	Kriteria individu kelebihan 2 tingkat/level
5	-2	3	Kriteria individu kekurangan 2 tingkat/level
6	3	2,5	Kriteria individu kelebihan 3 tingkat/level
7	-3	2	Kriteria individu kekurangan 3 tingkat/level
8	4	1,5	Kriteria individu kelebihan 4 tingkat/level
9	-4	1	Kriteria individu kekurangan 4 tingkat/level

4. Perhitungan dan pengelompokkan Core Factor dan Secondary Factor.

Setelah bobot nilai GAP ditentukan, maka dibagi menjadi 2 kelompok yaitu *Core Factor* dan *Secondary Factor*.

Rumus untuk menghitung *Core Factor* terlihat pada persamaan II.2.

$$NCF = \frac{\sum NC(Aspek)}{\sum IC} \dots\dots\dots(II.2)$$

dengan:

NCF = nilai rata-rata core factor.

NC (aspek) = jumlah nilai core factor.

IC = jumlah item core factor

Sedangkan rumus untuk menghitung *Secondary Factor* terlihat pada persamaan II.3.

$$NSF = \frac{\sum NS(Aspek)}{\sum IS} \dots\dots\dots(II.3)$$

dengan:

NSF = nilai rata-rata secondary factor.

NS (aspek) = jumlah nilai secondary factor.

IS = jumlah item secondary factor

5. Perhitungan Nilai Total

Untuk menghitung nilai total, rumus yang digunakan pada persamaan II.4.

$$(x)\%NCF(aspek) + (x)\%NSF(aspek) = Ntotal(aspek) \dots\dots\dots (II.4)$$

Keterangan :

$NCF(aspek)$ = nilai rata-rata core factor.

$NSF(aspek)$ = nilai rata-rata secondary factor.

$N(aspek)$ = nilai total dari aspek

$(x)\%$ = nilai persen yang diinputkan

6. Perhitungan Nilai Total

Untuk menentukan perankingan mengacu pada hasil perhitungan menggunakan rumus pada persamaan II.5.

$$Rangking = (x) \% Ns \dots\dots\dots(II.5)$$

Keterangan :

Ns = Nilai aspek

$(x)\%$ = nilai persen yang diinputkan

II.2.4. PHP (*Hypertext Preprocessor*)

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah *web* dan bisa digunakan pada dokumen HTML. PHP dirancang untuk dapat bekerja sama dengan *database server* dan dibuat sedemikian rupa sehingga pembuatan dokumen HTML yang dapat mengakses database menjadi begitu mudah (Anwar, 2016 : 3).

Untuk mencoba PHP, tuliskan *sintaks* Bahasa pemrograman PHP dan berikan nama *pertama.php*, kemudian *file* ini disimpan ke dalam *folder* *C:\xampp\htdocs*. Untuk memudahkan manajemen *file* sebaiknya dibuatkan folder tersendiri. Dalam hal ini misalnya *folder* diberi nama *coba* menjadi *C:\xampp\htdocs\coba*. Untuk melihat hasil program PHP, cobalah buka browser dan ketikkan *http://localhost/coba/pertama.php*.

Berikut sintaks dasar bahasa pemrograman PHP dengan nama *pertama.php*.

File *pertama.php*:

```
<html>
  <head>
    <title>Contoh Sederhana</title>
  </head>
  <body>
    <?php
      echo("Hallo apa kabar? Nama saya PHP script");
    ?>
  </body>
</html>
```

Hasilnya seperti tampak pada halaman <http://localhost/coba/pertama.php> pada Gambar II.1.

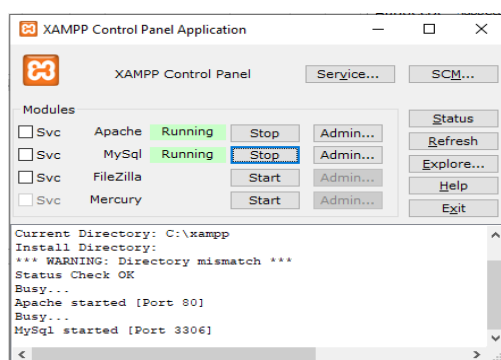


Gambar II.1 Hasil Sintak Dasar PHP

II.2.5. MySQL (Database)

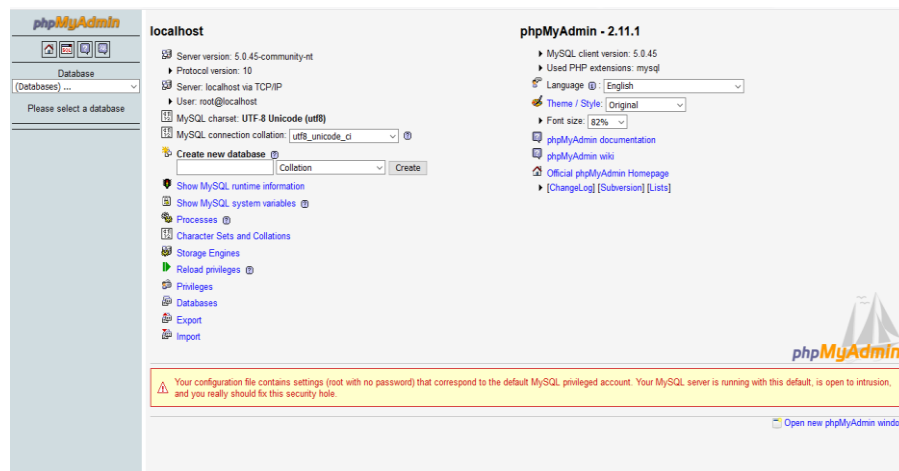
MySQL (*My Structure Query Language*) merupakan sebuah program pembuat *database* yang bersifat *open source*, artinya semua orang dapat menggunakannya dan dapat dijalankan pada semua *platform* baik *windows* maupun *linux*. MySQL juga merupakan sebuah perangkat lunak sistem manajemen basis data *SQL* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *multiuser* (Anwar, 2016 : 3).

Untuk mengakses MySQL buka aplikasi XAMPP, tekan tombol Start pada Apache dan MySQL seperti Gambar II.2.



Gambar II.2 Aplikasi XAMPP

Selanjutnya melalui web browser dengan mengetikkan URL <http://localhost/phpmyadmin/> pada address bar, maka tampilan phpMyAdmin akan seperti Gambar II.3.



Gambar II.3 Tampilan phpMyAdmin

Tampilan phpMyAdmin dibagi menjadi 2, bagian pertama berada pada sebelah kiri berisi daftar database yang telah ada pada Instalasi MySQL. Lalu pada bagian kedua berada pada sebelah kanan tampilan berisi fitur dari aplikasi phpMyAdmin dan informasi dari server MySQL.

Berikut ini adalah penjelasan membuat *database*, *table* dan *field* yaitu:

1. Setelah phpMyAdmin tampil pada web browser, isikan nama *database* pada kolom “*Create new database*”, kemudian klik tombol *Create*. Jika berhasil maka *database* yang dibuat akan tampil pada daftar *database* dan tampil halaman “*Create Table*”.
2. Halaman berikutnya isikan nama tabel pada kolom *Name*, dan jumlah kolom pada kolom *Number of fields*, kemudian klik tombol *Go*.
3. Halaman berikutnya “*Create Field*” untuk membuat kolom dalam *database*. Sebelum memulai pembuatan *field table*, terdapat beberapa hal yang perlu diperhatikan dalam menentukan tipe data yang digunakan.

II.2.6. Normalisasi

Normalisasi adalah salah satu cara untuk meminimalisir pengulangan data (*data redundancy*), normalisasi akan diperlukan jika ada indikasi bahwa tabel yang kita buat tidak baik (terjadi pengulangan informasi, potensi inkonsistensi data pada operasi perubahan, tersembunyinya informasi tertentu dan lain sebagainya) dan diperlukan supaya jika tabel-tabel yang didekomposisi kita gabung kembali dapat menghasilkan tabel awal sebelum didekomposisi, sehingga diperoleh tabel yang baik. Hasil dari normalisasi adalah himpunan-himpunan data (*tabel-tabel*) dalam bentuk normal (*normal form*) (Mulyani, 2016:132)

II.2.7. Unified Modelling Language (UML)

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. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem.


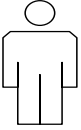

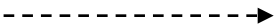

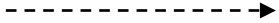
II.2.7.1. Use Case Diagram

Use Case Diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibangun. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibangun. *Use case*

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

Adapun simbol *use case diagram* terdapat pada Tabel.II.2.

Tabel II.2 Simbol Use Case Diagram

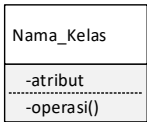


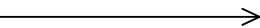
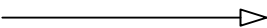
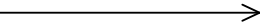
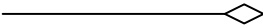
Simbol	Diskripsi
<p><i>Use Case</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor: biasanya dinyatakan dengan menggunakan kata kerja diawal di awal frase nama use case.</p>
<p>Aktor/actor</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.</p>
<p>Asosiasi /association</p> 	<p>Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan actor.</p>
<p>Ekstensi / extend</p> <p><<extends></p> 	<p>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 misal Arah panah mengarah pada use case yang ditambahkan.</p>
<p>Generalisasi / generalization</p> 	<p>Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>
<p>Menggunakan / include</p> 	<p>Include berarti use case yang tambahan akan selalu melakukan pengecekan apakah use case yang ditambahkan telah dijalankan sebelum use case tambahan dijalankan</p>

(Sumber : Julianto Simatupang, 2019)

II.2.5.2. Class Diagram

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Adapun simbol *class diagram* terdapat pada Tabel.II.3.

Tabel II.3 Simbol Class Diagram


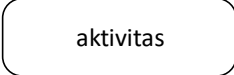


Simbol	Diskripsi
Kelas 	Kelas pada struktur sistem.
Antar muka/interface  nama_interface	Sama dengan konsep interface dalam pemrograman berorientasi objek.
Asosiasi / association 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity.
Asosiasi berarah / directed association 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity.
Generalisasi 	Relasi antar kelas dengan makna generalisasi-generalisasi- spesialisasi (umum khusus).
Kebergantungan / dependency 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi /aggregation 	Relasi antar kelas dengan makna semua- bagian (whole- part).

(Sumber : Julianto Simatupang, 2019)

II.2.5.3. Activity Diagram

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Penekanan pada diagram aktivitas adalah menggambarkan aktivitas sistem atau aktivitas yang dapat dilakukan oleh sistem, bukan apa yang dilakukan actor. Adapun simbol *activity diagram* terdapat pada Tabel.II.4.

Tabel II.4. Simbol Activity Diagram

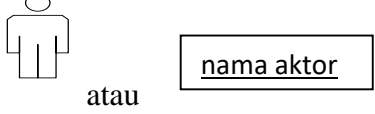

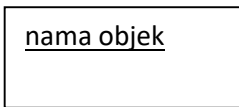

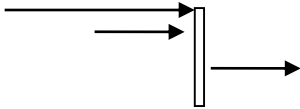
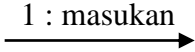
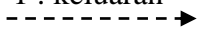
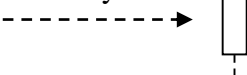
Simbol	Diskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / decision 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari Satu
Penggabungan / Join	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu..
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

(Sumber : Julianto Simatupang, 2019)

II.2.5.4. Sequence Diagram

Sequence Diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Adapun simbol *sequence diagram* terdapat pada Tabel.II.5.

Tabel II.5. Simbol *Sequence Diagram*

Simbol	Diskripsi
Aktor  atau	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
Garis hidup / lifeline : 	Menyatakan kehidupan suatu objek.
Objek 	Menyatakan objek yang berinteraksi pesan.
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya
Pesan tipe create << create >> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.
Pesan tipe send 1 : masukan 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
Pesan tipe return 1 : keluaran 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
Pesan tipe destroy 	Menyatakan suatu objek mengakhiri hidup objek lain, arah panah yang mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy.

(Sumber : Julianto Simatupang, 2019)