

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terdahulu

Dari penelitian yang dilakukan oleh Berutu (2020) mengenai sistem pendukung keputusan pemilihan ibu PKK terbaik dengan metode *fuzzy tahani*, disimpulkan bahwa *fuzzy tahani* dapat digunakan untuk pemilihan ibu PKK terbaik. Penelitian penulis menggunakan *fuzzy tahani* untuk pemilihan *security* terbaik.

Dari penelitian yang dilakukan oleh Matondang dan Tarigan (2020) mengenai sistem pendukung keputusan pemilihan laptop dengan metode *fuzzy tahani* berbasis *web*, disimpulkan bahwa *fuzzy tahani* dapat digunakan untuk pemilihan laptop. Penelitian penulis menggunakan *fuzzy tahani* untuk pemilihan *security* terbaik.

Dari penelitian yang dilakukan oleh Rahmasari, dkk (2020) mengenai perancangan aplikasi pemilihan kualitas keramik *granite* dengan metode *fuzzy tahani*, disimpulkan bahwa *fuzzy tahani* dapat digunakan untuk pemilihan kualitas keramik *granite*. Penelitian penulis menggunakan *fuzzy tahani* untuk pemilihan *security* terbaik.

Dari penelitian yang dilakukan oleh Elizabeth dan Tinaliah (2020) mengenai sistem pendukung keputusan pemilihan asisten dosen menggunakan metode SAW, disimpulkan bahwa sistem berhasil menghitung dan memproses dengan metode SAW dalam penentuan pemilihan asisten dosen yang sesuai dengan kriteria yang telah ditentukan. Penelitian penulis menggunakan *simple additive weighting*

(SAW) untuk pemilihan *security* terbaik dengan menghitung beberapa kriteria terkait.

Dari penelitian yang dilakukan oleh Hutagaol, dkk (2021) mengenai penerapan metode *simple additive weighting* (SAW) dalam pemilihan *handphone* bekas, disimpulkan bahwa sistem berhasil menghitung dan memproses dengan metode *Simple Additive Weighting* (SAW) dalam penentuan pemilihan *handphone* bekas yang sesuai dengan kriteria yang telah ditentukan. Penelitian penulis menggunakan *simple additive weighting* (SAW) untuk pemilihan *security* terbaik dengan menghitung beberapa kriteria terkait.

Dari penelitian yang dilakukan oleh Nurelasari dan Purwaningsih (2021) mengenai sistem pendukung keputusan pemilihan perumahan KPR terbaik menggunakan metode SAW, disimpulkan bahwa sistem berhasil menghitung dan memproses dengan metode SAW dalam penentuan pemilihan perumahan KPR terbaik yang sesuai dengan kriteria yang telah ditentukan. Penelitian penulis menggunakan *simple additive weighting* (SAW) untuk pemilihan *security* terbaik dengan menghitung beberapa kriteria terkait.

II.2. Landasan Teori

Landasan teori peneliti kutip dari beberapa teori yang berkaitan dengan penelitian ini yaitu:

II.2.1. Sistem Pendukung Keputusan (SPK)

Sistem pendukung keputusan (SPK) adalah sebuah sistem berbasis komputer yang membantu dalam proses pengambilan keputusan. SPK sebagai sistem informasi berbasis komputer yang adaptif, interaktif, fleksibel, yang secara khusus dikembangkan untuk mendukung solusi dari permasalahan manajemen yang tidak terstruktur untuk meningkatkan kualitas pengambilan keputusan. Dengan demikian dapat ditarik satu definisi tentang SPK yaitu sebuah sistem berbasis komputer yang adaptif, fleksibel, dan interaktif yang digunakan untuk memecahkan masalah-masalah tidak terstruktur sehingga meningkatkan nilai keputusan yang diambil. (Rahmasari, dkk, 2020: 467).

II.2.1.1. Proses Pengambilan Keputusan

Pengambilan keputusan meliputi beberapa tahap dan melalui beberapa proses, pengambilan keputusan meliputi empat tahap yang saling berhubungan dan berurutan. Empat proses tersebut adalah:

1. Intelligent

Tahap ini merupakan proses penelusuran dan pendekatan dari lingkup problematika serta proses pengenalan masalah. Data masukan diperoleh, diproses, dan diuji dalam rangka mengidentifikasi masalah.

2. Design

Tahap ini merupakan proses menemukan dan mengembangkan alternatif.

Tahap ini meliputi proses untuk mengerti masalah, menurunkan solusi dan menguji kelayakan solusi.

3. *Choice*

Pada tahap ini dilakukan proses pemilihan di antara berbagai alternatif tindakan yang mungkin dijalankan. Tahap ini meliputi pencarian, evaluasi, dan rekomendasi solusi yang sesuai untuk variabel hasil pada alternatif yang dipilih.

4. *Implementation*

Tahap implementasi adalah tahap pelaksanaan dari keputusan yang telah di ambil. Pada tahap ini perlu disusun serangkaian tindakan yang terencana, sehingga hasil keputusan dapat dipantau dan disesuaikan apabila diperlukan perbaikan. Dalam hal ini, model simon juga menggambarkan kontribusi Sistem Informasi Manajemen (SIM) dan Ilmu Manajemen / *Operations Research* (IM / OR) terhadap proses pengambilan keputusan. (Panggabean, 2016: 2).

II.2.1.2. Karakteristik Sistem Pendukung Keputusan

Karakteristik dan kapabilitas kunci dari Sistem Pendukung Keputusan adalah sebagai berikut:

1. Dukungan untuk pengambilan keputusan, terutama pada situasi semiterstruktur dan tak terseteruktur.
2. Dukungan untuk semua level manajerial, dari eksekutif puncak sampai manajer lini.

3. Dukungan untuk individu dan kelompok.
4. Dukungan untuk semua keputusan independen dan sekuensial.
5. Dukungan di semua fase proses pengambilan keputusan: inteligensi, desain, pilihan, dan implementasi.
6. Dukungan pada berbagai proses dan gaya pengambilan keputusan.
7. Kemampuan Sistem beradaptasi dengan cepat dimana pengambilan keputusan dapat menghadapi masalah-masalah dengan cara mengadaptasi sistem terhadap kondisi-kondisi perubahan yang terjadi.
8. Pengguna merasa seperti rumah. *User-friendly*, kapabilitas grafis yang kuat, dan sebuah bahasa interaktif yang alami.
9. Peningkatan terhadap keefektifan pengambilan keputusan (akurasi, *timelines*, kualitas) dari pada efisiensi (biaya).
10. Pengambil keputusan mengontrol penuh semua langkah proses pengambilan keputusan dalam memecahkan masalah.
11. Pengguna akhir dapat mengembangkan dan memodifikasi sistem sederhana.
12. Menggunakan model-model dalam penganalisisan situasi pengambilan keputusan.
13. Disediakkannya akses untuk berbagai sumber data, format, dan tipe, mulai dari Sistem informasi geografis (GIS) sampai sistem berorientasi objek.
14. Dapat dilakukan sebagai alat *standalone* yang digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan di satu

organisasi keseluruhan dan di beberapa organisasi sepanjang rantai persediaan. (Panggabean, 2016: 2).

II.2.1.3. Komponen-Komponen Sistem Pendukung Keputusan

Sistem Pendukung Keputusan terdiri dari empat subsistem, yaitu:

1. Manajemen Data, meliputi basis data yang berisi data-data yang relevan dengan keadaan dan dikelola oleh perangkat lunak yang disebut dengan *Database Management System (DBMS)*.
2. Manajemen Model berupa sebuah paket perangkat lunak yang berisi model-model *financial, statistic, management science*, atau model kuantitatif, yang menyediakan kemampuan analisa dan perangkat lunak manajemen yang sesuai.
3. Subsistem Dialog atau Komunikasi, merupakan subsistem yang dipakai oleh user untuk berkomunikasi dan member perintah (menyediakan *user interface*).
4. Manajemen *knowledge* yang mendukung subsistem lain atau sebagai komponen yang berdiri sendiri. (Panggabean, 2016: 3).

II.2.1.4. Karakteristik Dan Kemampuan Sistem Pendukung Keputusan

Dari pengertian Sistem Pendukung Keputusan maka dapat ditentukan karakteristik antara lain:

1. Mendukung proses pengambilan keputusan, menitik beratkan pada *management by perception*.
2. Adanya tatap muka manusia/mesin di mana manusia (*user*) tetap memegang kendali proses pengambilan keputusan.

3. Mendukung pengambilan keputusan untuk membahas masalah terstruktur, semi terstruktur dan tak struktur.
4. Memiliki kapasitas dialog untuk memperoleh informasi sesuai dengan kebutuhan.
5. Memiliki subsistem-subsistem yang terintegrasi sedemikian rupa sehingga dapat berfungsi sebagai kesatuan item.
6. Membutuhkan struktur data komprehensif yang dapat melayani kebutuhan informasi seluruh tingkatan manajemen. (Panggabean, 2016: 2).

II.2.2. *Security*

Security pada penelitian ini adalah satpam. Satpam adalah satuan pengaman yang dibentuk oleh instansi atau perusahaan untuk melakukan pengamanan diwilayah tersebut. Salah satu profesi yang memiliki tanggung jawab besar adalah satpam. Setiap instansi atau badan usaha pasti mempekerjakan satpam. Tujuan satpam adalah menjaga keamanan dan kestabilan dari segala bentuk gangguan yang terjadi dilingkup area kerjanya. (Bilal dan Abdurrohman, 2019: 1384).

II.2.3. Metode *Fuzzy Tahani*

Metode *Fuzzy Tahani* adalah sebuah metode yang digunakan untuk perangkingan kriteria, dalam melakukan proses perangkingan, metode *Fuzzy Tahani* memiliki beberapa tahapan yang harus dilakukan untuk menghitung metode *Fuzzy Tahani*, yaitu: Adapun langkah-langkah dari metode *Fuzzy Tahani* sebagai berikut:

Langkah 1: Pembentukan *Decision Making Matrix*

$$X = \begin{pmatrix} X_{01} & X_{0j} & X_{0n} \\ X_{i1} & X_{ij} & X_{in} \\ X_{n1} & X_{mj} & X_{mn} \end{pmatrix} \quad (i=0, m; \dots j=1, n) \dots\dots\dots (1)$$

Dimana

m = jumlah alternatif

n = jumlah kriteria

x_{ij} = nilai performa dari alternatif i terhadap kriteria j

x_{0j} = nilai optimum dari kriteria j

Jika nilai optimal kriteria j (X_{0j}) tidak diketahui, maka:

$$X_{0j} = \max_i / i. X_{ij}, \text{ if } \max_i / i. X_{ij} \text{ is preferable} \dots\dots\dots (2)$$

$$X_{0j} = \min_i / i. X_{ij}, \text{ if } \min_i / i. X_{ij} \text{ is preferable} \dots\dots\dots (3)$$

Langkah 2: Penormalisasian matriks keputusan untuk semua kriteria

Jika kriteria Beneficial maka dilakukan normalisasi mengikuti:

$$X_{ij} = \frac{x_{ij}}{\sum_{i=0}^m x_{ij}} \dots\dots\dots (4)$$

Dimana X_{ij}^* adalah nilai normalisasi. Jika kriteria Non-Beneficial maka dilakukan normalisasi mengikuti:

$$\text{Tahap 1: } X_{ij} = \frac{1}{x_{ij}} \dots\dots\dots (5)$$

$$\text{Tahap 2: } R = \frac{x_{ij}}{\sum_{i=0}^m x_{ij}} \dots\dots\dots (6)$$

Langkah 3: Menentukan bobot matriks yang sudah dinormalisasi

$$D = [d_{ij}]_{m \times n} = r_{ij} \cdot W_j \dots\dots\dots (7)$$

Dimana:

W_j = bobot kriteria j

Langkah 4: Menentukan nilai dari fungsi optimalisasi (S_i)

$$S_i = \sum_{j=1}^n d_{ij}; n, j=1 (i = 1, 2, \dots, m; j = 1, 2, \dots, n) \dots\dots\dots (8)$$

Dimana S_i adalah nilai fungsi optimalitas alternatif i . Nilai terbesar adalah yang terbaik, dan nilai yang paling sedikit adalah yang terburuk. Dengan memperhitungkan proses, hubungan proporsional dengan nilai dan bobot kriteria yang diteliti berpengaruh pada hasil akhir.

Langkah 5: Menentukan tingkatan peringkat tertinggi dari alternatif

$$K_i = \frac{S_i}{S_0} \dots\dots\dots (9)$$

Dimana S_i dan S_0 merupakan nilai kriteria optimalitas, diperoleh dari persamaan. Sudah jelas, itu dihitung nilai U_i berada pada interval $[0,1]$ dan merupakan pesanan yang diinginkan didahulukan efisiensi relatif kompleks dari alternatif yang layak bisa ditentukan sesuai dengan nilai fungsi utilitas. (Berutu, 2020: 263).

II.2.4. Metode *Simple Additive Weighting* (SAW)

Simple Additive Weighting (SAW) merupakan metode penjumlahan terbobot. Konsep dasar *Simple Additive Weighting* (SAW) adalah mencari penjumlahan terbobot dari *rating* kinerja pada setiap alternatif pada suatu kriteria. Berikut adalah langkah dan rumus dari metode *Simple Additive Weighting* (SAW):

1. Menentukan kriteria apa saja yang akan dijadikan acuan dalam pengambilan keputusan. $C_1, C_2, C_3, \dots, C_i$.
2. Menentukan kandidat simulasi simple yaitu: $A_1, A_2, A_3, \dots, A_i$.
3. Memberikan nilai preferensi (W) oleh pengambil keputusan untuk masing-masing kriteria yang sudah ditentukan. $W = [W_1, W_2, W_3, \dots, W_i]$.

4. Melakukan proses prangkingan dengan cara mengalikan matriks ternormalisasi (N) dengan nilai bobot preferensi (W).
5. Menentukan nilai preferensi untuk setiap alternative (Vi) dengan cara menjumlahkan hasil kali Antara matriks ternormalisasi (N) dengan nilai bobot preferensi (W).

Menentukan persamaan-persamaan yang ada dalam metode SAW adalah sebagai berikut:

$$R_{ij} = \frac{X_{ij}}{\max X_{ij}} \quad \text{atau} \quad R_{ij} = \frac{\min X_{ij}}{X_{ij}} \dots \dots \dots (6)$$

Keterangan:

Jika I = adalah atribut keuntungan (*benefit*)

Jika j = adalah atribut biaya (*cost*)

R_{ij} = Nilai rating kinerja ternormalisasi

X_{ij} = Nilai atribut yang dimiliki dari setiap kriteria

Max x_{ij} = Nilai terbesar jika yang dicari adalah atribut keuntungan atau nilai tertinggi

Min x_{ij} = Nilai min jika yang dicari adalah biaya atau nilai terendah

Benefit = Jika nilai terbesar adalah terbaik

Cost = Jika nilai terkecil adalah terbaik.

Nilai preferensi untuk setiap alternatif (Vi) diberikan sebagai:

$$V_i = \sum_{j=1}^n W_j \cdot r_{ij} \dots \dots \dots (7)$$

Keterangan:

V_i = Ranking untuk setiap alternatif

W_j = Nilai bobot dari setiap kriteria

R_{ij} = Nilai rating kinerja

Nilai V_i = Nilai yang lebih besar mengindikasikan bahwa alternatif A_i lebih terpilih. (Hutagaol, dkk, 2021: 64).

II.2.5. *Hypertext Preprocessor*

Hypertext Preprocessor atau yang sering disebut dengan PHP merupakan bahasa pemrograman yang bersifat *Open Source*. Bahasa pemrograman ini bersifat *server side*, maksudnya adalah tanpa adanya *server* yang berjalan beriringan dengan program atau aplikasi *script* PHP tidak dapat berjalan. PHP merupakan *script* yang disisipkan dalam *code* HTML untuk membuat halaman pada *website* bekerja secara otomatis dan dapat berfungsi sebagai pengolah data pada *server* dimana *script* tersebut dijalankan. PHP memiliki kemampuan untuk memisahkan diri dari kode HTML saat sebuah *website* dilakukan “*View Page Source*” oleh *user*. (Septyasari, dkk, 2018: 215).

PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML (*Hypertext Markup Language*) untuk membuat halaman *web* yang dinamis. Maksud dari *server-side scriping* adalah sinaks dan perintah-perintah yang diberikan akan sepenuhnya dijalankan di *server* tetapi disertakan pada dokumen HTML sebagai pembangun halaman *web*. (Kosasih, dkk, 2017: 115).

II.2.6. MySQL

MySQL merupakan turunan dari *Structured Query Language* (SQL), SQL merupakan operasional basis data yang dapat dikerjakan dengan mudah dan

otomatis. MySQL bertugas mengatur dan mengelola data-data pada *database*. MySQL merupakan perangkat lunak relasi yang dapat digunakan secara gratis. MySQL adalah perangkat lunak sistem manajemen basisdata pada *Structured Query Language* (SQL) yang *multiuser*. MySQL merupakan manajemen basis data yang paling stabil digunakan sebagai media penyimpanan data. (Oktasari dan Kurniadi, 2019: 150).

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. (Kosasih, dkk, 2017: 115).

II.2.7. Web

Web merupakan kumpulan halaman *web* yang saling berkaitan. *Web* adalah sebuah sistem yang berisi beragam informasi baik berupa teks, gambar, audio maupun video dan dapat diakses melalui perangkat yang biasa disebut *web browser*. *Web* merupakan kumpulan informasi baik yang bersifat statis maupun dinamis yang terdiri dari halaman yang dibuat. (Oktasari dan Kurniadi, 2019: 151).

II.2.8. Unified Modeling Language(UML)

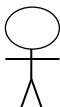
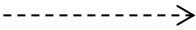
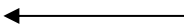
UML yaitu satu kumpulan konvensi permodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. (Andikos, 2019: 39).

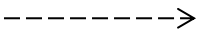
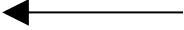
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasiskan UML adalah sebagai berikut:

1. *Use Case Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case diagram* dapat digambarkan dengan sumber-sumber pada Tabel II.1.

Tabel II.1. Simbol *Use Case*

Gambar	Nama	Keterangan
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).

	<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang mnghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

(Sumber: Andikos, 2019: 39)

2. Diagram Aktivitas (*Activity Diagram*)

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dapat digambarkan dengan simbol-simbol seperti pada tabel II.2.

Tabel II.2. Simbol *Activity Diagram*

Gambar	Nama	Keterangan
--------	------	------------

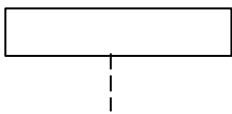
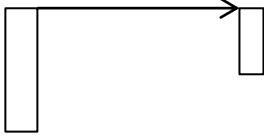
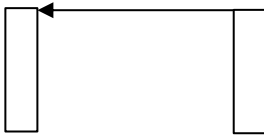
	<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Activity Final</i>	Bagaimana objek dibentuk dan dihancurkan
	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

(Sumber: Andikos, 2019: 39)

3. Diagram Urutan (*Sequence Diagram*)

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.3.

Tabel II.3. Simbol *Sequence Diagram*

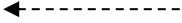
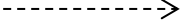
Gambar	Nama	Keterangan
	<i>Lifeline</i>	Objek entity, antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.

(Sumber: Andikos, 2019: 39)

4. *Class Diagram* (Diagram Kelas)

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.4.

Tabel II.4. *Class Diagram*

Gambar	Nama	Keterangan
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Depedency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
	<i>Assocation</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

(Sumber: Andikos, 2019: 39)