



BAB II

TINJAUAN PUSTAKA

2.1 Teori Judul

2.1.1 Penerapan

Menurut Dr. Muhammad Ali, M.Si (2017:51), “Implementasi (penerapan) merupakan penyediaan sarana untuk melaksanakan sesuatu yang menimbulkan dampak atau akibat terhadap sesuatu”.

Sedangkan menurut Browne dan Wildavsky dikutip oleh Arinda Firdianti, M.Pd.I (2018:19), “Implementasi (Penerapan) adalah perluasan aktivitas yang saling menyesuaikan.”

Dari pengertian-pengertian diatas, dapat disimpulkan bahwa penerapan merupakan suatu kegiatan yang meniru suatu teori dalam bentuk gerakan atau aktivitas untuk mencapai suatu hasil.

2.1.2 Pelanggan

Menurut Soedjas Triwibowo (2019:19) “Konsumen atau pelanggan merupakan stake holder bisnis. Bisnis tidak akan berjalan kalau tidak ada konsumen yang menggunakan produk atau jasa yang ditawarkan.”

2.1.2 Sistem Pendukung Keputusan

Menurut Asyriati, dkk. (2018:1) dalam buku Sistem Pendukung Keputusan Teori dan Implementasi. Sistem merupakan kumpulan sub-sub sistem (elemen) yang saling berkorelasi satu dengan yang lainnya untuk mencapai tujuan tertentu. Sebagai contoh: sebuah perusahaan memiliki sistem manajerial yang terdiri dari *bottom management*, *middle management*, dan *top manajemen* yang memiliki tujuan untuk mencapai kemajuan masyarakat. Sistem pendukung keputusan dapat diartikan sebagai suatu sistem yang di rancang yang digunakan untuk mendukung manajemen dalam pengambilan keputusan.

Menurut Nofriansyah dan Defit (2017:2) berpendapat bahwa sistem Pendukung Keputusan adalah suatu sistem informasi spesifik yang ditujukan untuk



membantu manajemen dalam mengambil keputusan yang berkaitan dengan persoalan yang bersifat semi terstruktur.

Konsep Sistem Pendukung Keputusan (SPK) pertama kali diungkapkan pada tahun 1971 oleh Scoot dalam Turban (2001) dengan istilah *Management Decision System*, kemudian sejumlah perusahaan, lembaga penelitian dan perguruan tinggi mulai melakukan penelitian dan membangun Sistem Pendukung Keputusan, sehingga dari produksi yang dihasilkan dapat disimpulkan bahwa sistem ini merupakan suatu sistem berbasis komputer yang ditujukan untuk membantu pengambilan keputusan dalam memanfaatkan data dan model tertentu untuk memecahkan berbagai persoalan yang tidak terstruktur.

2.1.3 *Weighted Product (WP)*

Menurut Asyriati, dkk. (2018:26) “Metode *Weighted Product (WP)* merupakan salah satu metode yang sederhana dengan perkalian untuk menghubungkan rating atribut dimana setiap atribut harus dipangkatkan dengan bobot atribut yang bersangkutan.”

Adapun algoritma penyelesaian metode ini adalah sebagai berikut:

1. Langkah 1 : Mendefinisikan terlebih dahulu kriteria-kriteria yang akan di jadikan sebagai tolak ukur penyelesaian masalah.
2. Langkah 2 : Menormalisasi setiap nilai alternatif (nilai vector).
3. Langkah 3 : Menghitung nilai bobot prefensi pada setiap alternatif.
4. Langkah 4 : Melakukan perangkaian.

Berikut rumus yang digunakan pada metode *Weighted Product* yaitu :

1. Normalisasi bobot menggunakan rumus

$$W_j = \frac{w_j}{\sum w_j}$$

2. Menentukan nilai vektor S_i menggunakan rumus

$$S_i = \prod_{j=1}^n x_{ij} w_j$$

3. Menghitung nilai prefensi V_i untuk setiap alternatif menggunakan rumus

$$V_i = \frac{S_1}{S_1 + S_2 + S_3} \text{ atau } V_i = \frac{\prod_{j=1}^n x_{ij} w_j}{\prod_{j=1}^n (x_{j*}) w_j}$$



2.2 Teori Khusus

2.2.1 Unified Modelling Language (UML)

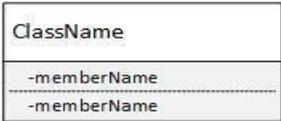
Menurut Sukamto (2018:13), *Unified Modeling Language (UML)* adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

2.2.2 Class Diagram

Sukamto (2018:141), “diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan *method* atau operasi.” Berikut penjelasan atribut dan *method* :

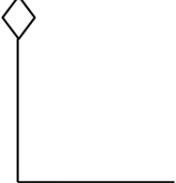
1. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau *method* adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Tabel 2.1 Simbol Class Diagram

No	Simbol	Deskripsi
1.	Kelas 	Kelas pada struktur sistem
2.	Antarmuka/ <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek



Lanjutan Tabel 2.1 Simbol Class Diagram

No	Simbol	Deskripsi
3.	Asosiasi/ <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
4.	Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
5.	Generalisasi 	Relasi antar kelas dengan makna generalisasi – spesialisasi (umum - khusus)
6.	Kebergantungan/ <i>dependensi</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
7.	Agregasi/ <i>aggregation</i> 	Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>)

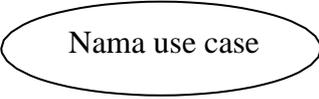
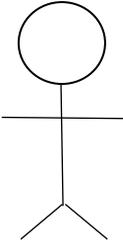
2.2.3 Use Case Diagram

Menurut (Salahuddin, 2014), *Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use*



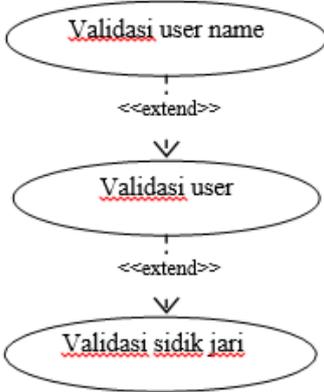
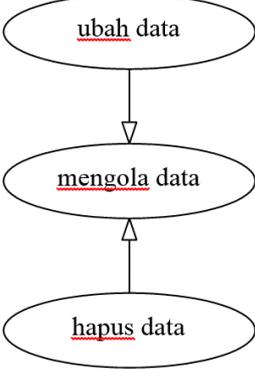
case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Tabel 2.2 Simbol Use Case Diagram

No	Simbol	Deskripsi
1.	<p><i>Use case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama <i>use case</i> .
2.	<p>Aktor/<i>actor</i></p> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3.	<p>Assosiasi/<i>association</i></p> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.	<p>Exstensi/<i>extend</i></p> <p><<extend>></p>	Relasi <i>use case</i> tambahan kesebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu, mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek, biasanya <i>use case</i>

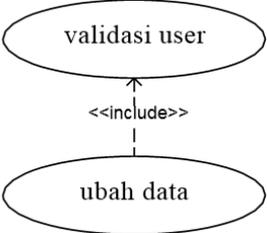


Lanjutan Tabel 2.2 Simbol Use Case Diagram

No	Simbol	Deskripsi
		<p>tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misal</p>  <p>Arah panah mengarah pada <i>use case</i> yang ditambahkan, biasanya <i>use case</i> yang menjadi <i>extend</i>-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p>
5.	<p>Generalisasi/<i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua <i>buah use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya :</p> 



Lanjutan Tabel 2.2 Simbol Use Case Diagram

No	Simbol	Deskripsi
		 <p>Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.</p>

Sumber : Sukamto (2018:156-158)

2.2.4 Activity Diagram

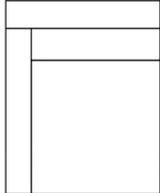
Sukamto (2018:161), “diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.” Berikut adalah simbol-simbol yang ada pada diagram aktivitas :

Tabel 2.3 Simbol Activity Diagram

No	Simbol	Deskripsi
1.		Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.



Lanjutan Tabel 2.3 Simbol Activity Diagram

No	Simbol	Deskripsi
3.	Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.	Status akhir 	Status akhir yang dilakukan oleh sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Sumber : Sukamto (2018:162-163)

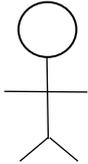
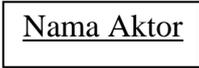
2.2.5 Sequence Diagram

Sukamto (2018:165), diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup dalam diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka



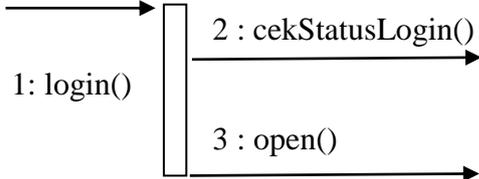
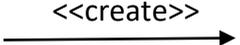
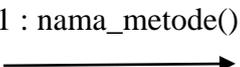
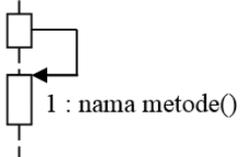
diagram sekuen yang harus dibuat juga semakin banyak. Berikut adalah simbol-simbol yang ada pada diagram sekuen :

Tabel 2.4 Simbol Sequence Diagram

No	Simbol	Deskripsi
1.	<p>Aktor</p>  <p>Atau</p>  <p>Tanpa waktu aktif</p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan dalam menggunakan kata benda diawal frase nama aktor.</p>
2.	<p>Garis hidup/<i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
3.	<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan</p>
4.	<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semuanya yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya</p>

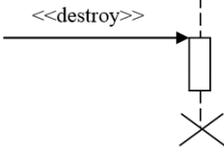


Lanjutan Tabel 2.4 Simbol Sequence Diagram

No	Simbol	Deskripsi
		 <p>Maka cekStatusLogin() dan open() dilakukan didalam metode login(). Aktor tidak memiliki waktu aktif</p>
5.	Pesan tipe <i>create</i> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
6.	Pesan tipe <i>call</i> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,  <p>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</p>
7.	Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirimkan data/masukkan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.



Lanjutan Tabel 2.4 Simbol Sequence Diagram

No	Simbol	Deskripsi
8.	Pesan tipe <i>return</i> 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.
9.	Pesan tipe <i>destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaliknya jika ada <i>create</i> maka ada <i>destroy</i>

Sumber : Sukamto (2018:165-167)

2.3 Teori Program

2.3.1 Pengertian PHP

Menurut Yudhanto dan Prasetyo (2019:9) “PHP atau *Hypertext Processor* adalah Bahasa pemrograman *script server side* yang sengaja dirancang lebih cenderung untuk membuat dan mengembangkan web.”

Sedangkan menurut Sulistiono Heru (2019:5) “PHP (*Hypertext Processor*) adalah Bahasa pemrograman yang digunakan untuk membuat *website* atau situs dinamis dan menangani rangkaian Bahasa pemrograman antara *client side scripting* dan *server side scripting*.”

Dari pengertian-pengertian diatas, dapat disimpulkan PHP atau *Hypertext Processor* adalah Bahasa pemrograman yang digunakan untuk pemrograman *script server side*.

2.3.2 Pengertian Mysql

Menurut Sumarlinda (2015:20) “MySQL adalah *multiuser database* yang menggunakan bahasa *structured query language* (SQL), MySQL dalam operasi



client-server melibatkan *server daemon* MySQL disisi *server* dan berbagai macam program serta *library* yang berjalan disisi *client*.

Menurut Sidik (2017:301) “MySQL merupakan *software database* yang termasuk paling populer dilingkungan *Linux*, kepopuleran ini karena ditunjang karena performansi *query* dari databasenya yang saat itu bisa dikatakan paling cepat dan jarang bermasalah. MySQL telah tersedia juga dilingkungan *Windows*”.

Berdasarkan pengertian diatas, bahwa dapat disimpulkan MySQL adalah sebuah database yang digunakan untuk menyimpan data dalam tabel terpisah, mysql dapat berjalan stabil pada berbagai sistem operasi seperti windows,linux dan lainnya serta dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan.

2.4 Penelitian Terdahulu

Menurut penelitian yang dilakukan oleh Supriyono dan Sari (2015) mengatakan salah satu metode dalam FMADM yang cukup terkenal adalah metode *Weighted Product* (WP). Metode WP cukup banyak digunakan untuk pengambilan keputusan karena metodenya yang sederhana dengan memasukkan semua faktor dan komputasinya cepat.

Rahmawati dan Astuti (2018) melakukan sebuah penelitian yang menyebutkan, metode *Weighted Product* menggunakan perkalian untuk menghubungkan rating atribut, dimana rating setiap atribut harus dipangkatkan dulu dengan bobot atribut yang bersangkutan.

Dalam penelitian yang dilakukan oleh Prasetyo (2017) mengatakan bahwa metode *Weighted Product* (WP) adalah sebuah metode SPK yang digunakan untuk menyelesaikan masalah-masalah MADM (Multi Atribut Decision Making). Metode WP menggunakan perkalian untuk menghubungkan rating atribut, di mana rating setiap atribut harus dipangkatkan dulu dengan bobot atribut yang bersangkutan. Proses ini sama halnya dengan proses normalisasi.

Yoni dan Mustaidah (2016) melakukan sebuah penelitian yang menyebutkan, metode *Weighted Product* (WP) adalah salah satu metode yang digunakan untuk penyelesaian sistem pengambilan keputusan dengan



mempertimbangkan kriteria dan bobot. Penelitian ini menggunakan Metode *Weighted Product* (WP), karena dalam pengambilan keputusan pemilihan mahasiswa lulusan terbaik tidak ada sub kriteria.

Dalam penelitian yang dibuat oleh Septiyana Firdyana, dkk. (2017) mengatakan bahwa metode *Weighted Product* ini lebih efisien karena waktu yang dibutuhkan dalam perhitungan lebih singkat. Metode ini dipilih karena dapat menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perankingan yang akan menghasilkan pilihan penerima bantuan raskin yang sesuai dengan kriteria.