



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Perangkat Lunak

Yurindra (2017:1) menyatakan bahwa “Perangkat Lunak (software) merupakan transformasi informasi yang memproduksi, mengatur, memperoleh, memodifikasi, menampilkan, atau memancarkan informasi sehingga pekerjaan dapat menjadi lebih sederhana”.

Uji Saputro (2017:18) menyatakan bahwa Software merupakan kumpulan beberapa perintah yang dieksekusi oleh mesin computer dalam menjalankan pekerjaannya”.

Sedangkan Subandi & Aulian Akhrian Syahidi (2018:3) menyatakan bahwa “ Software adalah perangkat lunak yang digunakan dalam suatu sistem basis data”.

2.1.2 Pengertian Sistem

Romney (2015: 2) sistem adalah serangkaian dua atau lebih komponen yang saling terkait dan berinteraksi untuk mencapai suatu tujuan.

Definisi sistem menurut Mulyadi (2016:5), Sistem adalah “suatu jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan”.

Sedangkan Mulyadi (2016: 1) Sistem pada dasarnya adalah sekelompok unsur yang erat berhubungan satu dengan lainnya, yang berfungsi bersamasama untuk mencapai tujuan tertentu.

2.1.3 Pengertian Web

Abdullah (2015:1), “Web dapat diartikan sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital baik berupa



text, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet”.

Sedangkan Bekti (2015:35), “Website merupakan kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing- masing masing dihubungkan dengan jaringan-jaringan halaman”.

2.2 Teori Judul

2.2.1 Pengertian Aplikasi

Sanjaya (2015), Aplikasi adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya Microsoft Word, Microsoft Excel. Aplikasi berasal dari kata application yang artinya penerapan lamaran penggunaan

Sedangkan IBISA (2017:93), paket sistem aplikasi adalah “alat bantu untuk mempermudah dan mempercepat proses pekerjaan dan bukan merupakan beban bagi pegawainya.”

Berdasarkan kedua uraian pengertian aplikasi menurut para ahli di atas maka penulis menyimpulkan aplikasi adalah perangkat lunak pembantu yang berguna untuk mempermudah suatu pekerjaan dan mempermudah atau meringkas pekerjaan pembuatan perangkat lunak lainnya.

2.2.2 Bahan Baku

2.2.2.1 Pengertian Bahan Baku

Astyningtyas (2015:12) menyatakan bahwa “bahan baku adalah sejumlah barang-barang yang dibeli dari pemasok (supplier) dan akan dipergunakan atau diolah menjadi produk yang akan dihasilkan oleh perusahaan”.

Sedangkan menurut Hildah 2016:14) menyatakan bahwa “bahan baku mentah adalah bahan-bahan yang belum mengalami jenis pengolahan apapun bagi perusahaan yang bersangkutan”.



Berdasarkan pengertian yang dijelaskan diatas bahwa bahan baku merupakan hal yang sangat mempengaruhi proses produksi, baik pengaruh secara kualitas maupun secara kuantitas bahan baku.

2.2.3 Produksi

2.2.3.1 Pengertian Produksi

Prishardoyo (2017;11) produksi adalah suatu bentuk agenda yang dilakukan dalam upaya memenuhi kebutuhan manusia, agenda ini lebih disempitkan haruslah agenda yang menghasilkan barang atau menambah nilai guna barang atau jasa.

Sedangkan Gaspersz (2018;2) produksi adalah fungsi utama yang di dalamnya mengandung unsur aktivitas meningkatkan nilai jual dalam produk sehingga secara umumnya produksi adalah peningkatkan hasil dari apa yang telah dicapai.

Dari penjelasan diatas, dapat disimpulkan jika produksi akan mampu menggerakkan para pelaku usaha, suatu kawasan dengan pembangunan ekonomi yang merata, kawasan ekonomi yang berdaya saing tinggi serta kawasan yang terintegrasi penuh dengan ekonomi global.

2.2.4 Metode *Economic Order Quantity* (EOQ)

2.2.4.1 Pengertian Metode *Economic Order Quantity* (EOQ)

Metode *Economic Order Quantity* (EOQ) adalah metode yang paling sering digunakan perusahaan karena metode ini dikenal sederhana dan mudah dalam penggunaannya. Seperti yang dikemukakan oleh Irham Fahmi (2016:120) yang menjelaskan bahwa “Model *Economic Order Quantity* (EOQ) merupakan model matematik yang menentukan jumlah barang yang harus dipesan untuk memenuhi permintaan yang diproyeksikan, dengan biaya persediaan yang diminimalkan”.

Sedangkan pengertian metode *Economic Order Quantity* (EOQ) menurut Ricky Virona Martono (2018:142) adalah metode sistem pemesanan yang menyeimbangkan biaya penyimpanan dan biaya pemesanan pada persediaan.



Rumus yang biasa digunakan adalah :

$$EOQ = \frac{\sqrt{2 \times R \times S}}{P \times I}$$

Keterangan:

EOQ = Jumlah persediaan yang ekonomi

R = Kebutuhan bahan baku dalam satu periode (Meter)

S = Biaya pesan (Rupiah)

P = Biaya pembelian perunit (Rupiah)

I = Biaya Penyimpanan (Rupiah)

EOQ merupakan salah satu teknik pengendalian persediaan tertua dan paling terkenal. Teknik ini relatif mudah digunakan, tetapi didasarkan pada beberapa asumsi) sebagai berikut:

1. Tingkat permintaan diketahui dan bersifat konstan.
2. Lead time, yaitu waktu antara pemesanan dan penerimaan pesanan, diketahui, dan bersifat konstan.
3. Persediaan diterima dengan segera. Dengan kata lain, persediaan yang dipesan tiba dalam bentuk kumpulan produk pada satu waktu.
4. Tidak mungkin diberikan diskon.
5. Biaya variable yang muncul hanya biaya pemasangan atau pemesanan dan biaya penahanan atau penyimpanan persediaan sepanjang waktu. Biaya- biaya ini dibahas di bagian sebelumnya.
6. Keadaan kehabisan stok (kekurangan) dapat dihindari sama sekali bila pemesanan dilakukan pada waktu yang tepat.



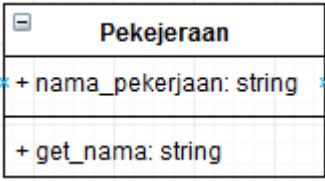
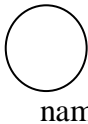
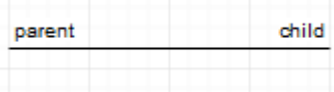

2.3 Teori Khusus

2.3.1. Class Diagram

Menurut 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* :


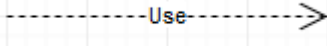

1. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau *method* adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Tabel 2.1 Simbol Class Diagram

No.	Simbol	Deskripsi
1	<p>Kelas</p> 	Kelas pada struktur sistem
2	<p>Antarmuka / <i>interface</i></p> 	sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek.
3	<p>Asosiasi / <i>association</i></p> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4	<p>Asosiasi berarah / <i>directed association</i></p> 	Relasi antarkelas dengan makna kelas yang saat digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .




Lanjutan Tabel 2.1

5	Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
6	Kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antar kelas.
7	Agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>)

2.3.2. Use Case Diagram

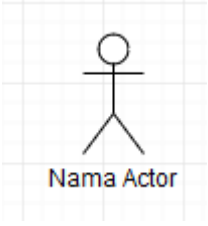

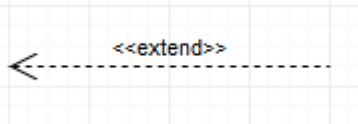
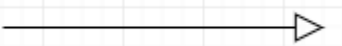
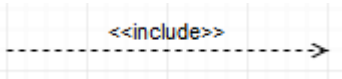
Menurut Sukamto (2018:155), use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut adalah simbol-simbol yang ada pada diagram *use case* :

Tabel 2.2 Simbol Use Case Diagram

No.	Simbol	Deskripsi
1	<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan anta unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal fase nama <i>use case</i>



Lanjutan Tabel 2.2


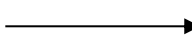
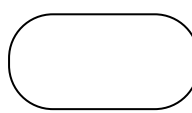
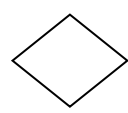
2	<p>Aktor / <i>actor</i></p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal fase nama aktor</p>
3	<p>Asosiasi / <i>association</i></p> 	<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</p>
4	<p>Ekstensi / <i>extend</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <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> tambahan memiliki nama depan yang sama dengan <i>use case</i></p>
5	<p>Generalisasi/ <i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>
6	<p>Menggunakan/ <i>include</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p>



2.3.3. Activity Diagram

Menurut 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

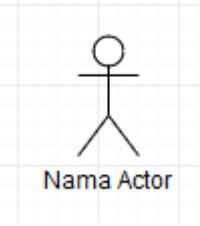
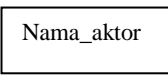

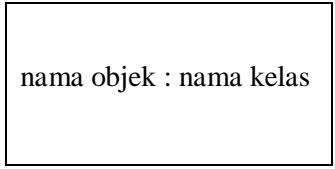
No	Simbol	Deskripsi
1	Start / status awal (<i>Intial State</i>) 	Start atau intial state adalah state atau keadaan awal pada saat sistem mulai hidup
2	End / status akhir (<i>final state</i>) 	End atau final state adalah state keadaan akhir dari daur hidup suatu sistem.
3	Event 	Event adalah kegiatan yang menyebabkan berubahnya status mesin.
4	State 	Sistem pada waktu tertentu. State dapat berubah jika ada event tertentu yang memicu perubahan tersebut.
5	Decision 	Decision menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu.



2.3.4 Sequence Diagram




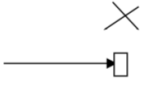
Menurut Sukanto (2018:165), diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Berikut adalah simbol-simbol yang ada pada diagram sekuen :

Tabel 2.4 Simbol Sequence Diagram

No	Simbol	Deskripsi
	<p>Aktor</p>  <p>atau</p> 	<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.</p>
	<p>Garis hidup / <i>life line</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
	<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan</p>



Lanjutan Tabel 2.4

	<p>Waktu aktif</p>	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.</p>
<p>Pesan tipe create</p> <p><<create>></p> 	<p>Pesan tipe create</p> <p><<create>></p>	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe call</p> <p>1 : nama_metode()</p> 	<p>Pesan tipe <i>call</i></p> <p>1 : nama_metode()</p>	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri</p>
<p>Pesan tipe destroy</p> <p><<destory>></p> 	<p>Pesan tipe destroy</p> <p><<destory>></p>	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i></p>



2.4 Teori Program

2.4.1 *Hypertext Preprocessor (PHP)*

2.4.1.1 *Pengertian Hypertext Preprocessor (PHP)*

Supono dan Putratama (2016:3) mengemukakan bahwa "PHP (PHP: Hypertext Preprocessor) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang berbasis server-side yang dapat ditambahkan ke dalam HTML"

Sedangkan, Solichin (2016:11) mengemukakan bahwa "PHP merupakan salah satu bahasa pemrograman berbasis web yang ditulis oleh dan untuk pengembang web".

Ardhana (2012:88), PHP merupakan bahasa pemrograman berbasis *serverside* yang dapat melakukan *parsing script php* menjadi *script web* sehingga dari sisi *client* menghasilkan suatu tampilan yang menarik.

Kumpulan kutipan diatas menerangkan bahwa hypertext preprocessor (PHP) merupakan bahasa pemrograman untuk membuat atau mengembangkan aplikasi berbasis web dan bersifat open soure dan ditanamkan ke dalam script HTML.

2.4.1.2 *Script Hypertext Preprocessor (PHP)*

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 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*.

. Adapun kode berikut contoh kode PHP (*Hypertext Preprocessor*) yang berada dalam kode HTML.

```
<HTML>
<HEAD>
<TITLE>Selamat Datang</TITLE>
</HEAD>
<BODY>
Selamat belajar PHP</BR>
<?php
    printf("Tgl. Sekarang: %s", Date ("d F Y"));
?>
</BODY>
</HTML>
```

Kode *Hypertext Preprocessor* (PHP) diawali dengan `<? PHP` dan diakhiri dengan `?>`. Pasangan kedua kode inilah yang berfungsi sebagai tag kode PHP. Berdasarkan tag inilah, pihak server dapat memahami kode PHP dan kemudian memprosesnya.

2.4.2 MySQL

2.4.2.1 Pengertian MySQL

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*.

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.

2.4.2.2 Fitur MySQL

Sebagai *software DBMS*, *MySQL* memiliki sejumlah fitur seperti yang dijelaskan di bawah ini:

a. *Multiplatform*

MySQL tersedia pada beberapa *platform* (*Windows*, *Linux*, *Unix*, dan lain-lain).

b. Andal, cepat, dan mudah digunakan

MySQL tergolong sebagai *database server* (*server* yang melayani permintaan terhadap *database*) yang handal, dapat menangani *database* yang besar dengan kecepatan tinggi, mendukung banyak sekali fungsi untuk mengakses *database*, dan sekaligus mudah untuk digunakan.

c. Jaminan keamanan akses

MySQL mendukung pengamanan *database* dengan berbagai kriteria penaksesan. Sebagai gambaran, dimungkinkan untuk mengatur *user* tertentu agar bisa mengakses data yang bersifat rahasia (misalnya gaji pegawai), sedangkan *user* lain tidak boleh.

d. Dukungan SQL

Seperti tersirat dalam namanya, MySQL mendukung perintah *SQL* (*Structured Query Language*). Sebagaimana diketahui, SQL merupakan standar pengaksesan *database* relasional. Pengetahuan akan SQL akan memudahkan siapa pun untuk menggunakan MySQL.



2.4.3 Basis Data (*Database*)

Subandi (2018:3) Basis data dapat diungkapkan sebagai suatu pengorganisasian data dengan bantuan computer yang memungkinkan data dapat diakses dengan mudah dan cepat.

Sedangkan Kadir (2019:39) menyatakan bahwa basis data adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi.

Dari pengertian diatas dapat disimpulkan bahwa basis data (*database*) adalah tempat yang digunakan untuk menyimpan data-data agar lebih terorganisir antara satu sama lain.

2.4.4 CSS (*Cascading Style Sheet*)

Bekti (2015:47), “CSS (*cascading style sheet*) merupakan salah satu Bahasa pemrograman web yang digunakan untuk mempercantik halaman web dan mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam”.

Wahana Komputer (2015:2), “CSS atau *cascading style sheet* adalah sekumpulan kode pemrograman web yang berfungsi untuk mengendalikan beberapa komponen di dalam web sehingga menjadi tampak seragam, berstruktur, dan teratur”.

2.4.5 Blackbox Testing

Rosa dan Salahuddin (2015:275) “Blackbox testing yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program”. Rizky (2016:264) “Blackbox testing adalah tipe testing yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya”.