

BAB II TINJAUAN PUSTAKA

1.1 Tinjauan Pustaka

Dalam penelitian ini akan digunakan lima tinjauan pustaka yang nantinya dapat mendukung penelitian, berikut ini merupakan tinjauan pustaka yang diambil yaitu pada tabel 2.1:

Tabel 1.1 Tinjauan Pustaka

| | | |
|----|-------------------|---|
| 1. | Judul | Analisis dan Perancangan Sistem Informasi Manajemen Persediaan Barang pada Advan Service Center Jambi |
| | Penulis | Amelia, Joni Devitra |
| | Tanggal/Tahun | 2018 |
| | Permasalahan | Pada saat ini Advan Service Center Jambi dalam pengolahan data persediaan barang masih dilakukan dengan cara manual. Barang direkap dan diinput ke dalam Microsoft Excel untuk dibuatkan laporan pemakaian |
| | Tujuan Penelitian | Tujuan pada penelitian ini menganalisa dan mempelajari permasalahan sistem informasi manajemen persediaan barang pada Advan Service Center Jambi dan menghasilkan prototipe sistem informasi manajemen persediaan barang pada Advan Service Center Jambi. |
| | Metode Penelitian | Prototipe |
| | Hasil Penelitian | Penelitian ini menghasilkan sebuah prototipe sistem informasi persediaan barang yang dapat mengelola data persediaan barang secara terkomputerisasi dan dapat menyajikan laporan-laporan barang yang dibutuhkan setiap hari atau setiap bulannya |
| 2. | Judul | Optimasi Persediaan Pupuk Non-Subsidi Menggunakan Program Dinamis Model Inventory (UD. Barokah) |
| | Penulis | Reynaldy Teja Kusumah, Anita Ilmaniati |
| | Tanggal/Tahun | 2019 |
| | Permasalahan | Sering kali timbul permasalahan dalam penyaluran pupuk yaitu persediaan yang tidak sesuai dengan jumlah permintaan. |
| | Tujuan Penelitian | Mengoptimalkan persediaan pupuk non subsidi |

Tabel 2.1 Tinjauan Pustaka (Lanjutan)

| | | |
|----|-------------------|---|
| | Metode Penelitian | Metode yang digunakan dalam penelitian ini adalah metode peramalan permintaan dengan regresi linier dan program dinamis untuk menentukan persediaan |
| | Hasil Penelitian | Berdasarkan hasil penelitian, didapatkan bahwa pengaturan persediaan pupuk non-subsidi pada tahun 2018-2019 di UD. Barokah berkisar antara 18.000 kg -26.000 per bulan dengan total biaya adalah Rp.7.741.134.625. |
| 3. | Judul | Rancang Bangun Sistem Informasi Pengolahan Data Persediaan Barang Berbasis Dekstop Dengan Model Waterfall |
| | Penulis | Margareta Nawang, Laela Kurniawati, Dudi Duta |
| | Tanggal/Tahun | 2017 |
| | Permasalahan | Seperti pengelolaan, pencarian, alur keluar masuknya barang, dan laporan data barang ketika dibutuhkan perusahaan. |
| | Tujuan Penelitian | Mempermudah melakukan manajemen persediaan |
| | Metode Penelitian | FIFO (<i>First In First Out</i>) |
| | Hasil Penelitian | adanya system informasi pengelolaan data persediaan barang pada PT. Sakura Yasa Prima maka system yang telah terkomputerisasi; sistem yang terkomputerisasi akan mempermudah segala aktifitas operasional perusahaan. |
| 4. | Judul | Sistem Informasi Manajemen Inventory Secara Online Menggunakan Framework EasyUI |
| | Penulis | Abdul Bari, Kasmawi |
| | Tanggal/Tahun | 2016 |
| | Permasalahan | Lemahnya pengawasan menjadi dampak buruk bagi manajemen sehingga pelaporan penerimaan atau pengeluaran barang dan pengawasan terhadap penggunaan barang menjadi terhambat. |
| | Tujuan Penelitian | Tujuan penelitian ini adalah membuat sistem informasi manajemen inventory menggunakan Framework EasyUI yang dapat diakses dan diawasi secara online |
| | Metode Penelitian | Framework EasyUI |
| | Hasil Penelitian | Penelitian ini menghasilkan sistem informasi manajemen inventory yang memberikan informasi stok secara real-time dan laporan semester penerimaan dan pengeluaran barang, sehingga proses pelaporan dan pengontrolan informasi stok dapat dilakukan dengan baik. |
| 5. | Judul | Implementasi <i>Differential Evolution</i> untuk Optimasi Jadwal Produksi |

Tabel 2.1 Tinjauan Pustaka (Lanjutan)

| | |
|-------------------|--|
| Penulis | Hendry Setiawan, Dewi Fandelina Tan, Kestriana Rega Prilianti |
| Tanggal/Tahun | 2018 |
| Permasalahan | Permasalahan yang sering terjadi adalah masih sering terjadinya keterlambatan dalam pengerjaan pesanan dari batas waktu yang ditentukan dan kerusakan produksi (afal) yang dihasilkan masih sangat tinggi. |
| Tujuan Penelitian | Melakukan optimasi jadwal produksi secara otomatis |
| Metode Penelitian | <i>Differential Evolution</i> |
| Hasil Penelitian | Hasil pengujian yang dilakukan menunjukkan bahwa aplikasi dapat melakukan optimasi jadwal produksi dengan rata-rata tingkat keakuratan mencapai 99,54% |

1.2 Rancang Bangun

Menurut Sari (2017) rancang adalah kegiatan yang memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik.

Menurut Sari (2017) bangun adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian.

Menurut Sari (2017) rancang bangun adalah proses pembangunan sistem untuk menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun hanya sebagian.

1.3 Manajemen Persediaan

Menurut Widodo and Devitra (2018) dalam menerangkan bahwa “Manajemen Persediaan (*Inventory Control*) atau disebut juga Inventory Management atau pengendalian tingkat persediaan adalah kegiatan yang

berhubungan dengan perencanaan, pelaksanaan, dan pengawasan penentuan kebutuhan material sedemikian rupa sehingga disatu pihak kebutuhan operasi dapat dipenuhi pada waktunya dan di lain pihak investasi persediaan material dapat ditekan secara optimal.

Menurut Widodo and Devitra (2018) pengertian persediaan dalam hal ini merupakan suatu aktiva yang meliputi barang-barang perusahaan dengan maksud untuk dijual dalam suatu periode usaha tertentu, atau persediaan barang masih dalam pengerjaan/proses produksi ataupun persediaan barang baku yang masih menunggu penggunaannya suatu proses produksi.

1.4 Pupuk Non Subsidi

Menurut Setiawan (2017) Pupuk non subsidi adalah pupuk yang pengadaan dan penyalurannya di luar program Pemerintah dan tidak mendapat subsidi. Pupuk non subsidi adalah pupuk yang dialokasikan setelah semua kebutuhan pupuk bersubsidi yang telah ditentukan oleh pemerintah terpenuhi.

1.5 Metode FIFO

Menurut Bustami dan Nurlela (2013) Metode FIFO (*first in first out*) menganggap bahwa harga pokok dari barang-barang yang pertama kali dibeli akan merupakan barang yang dijual pertama kali. Dalam metode ini persediaan akhir dinilai dengan harga pokok pembelian yang paling akhir.

Metode ini juga mengasumsikan bahwa barang yang terjual karena pesanan adalah barang yang mereka beli. Oleh karenanya, barang-barang yang dibeli pertama kali adalah barang-barang pertama yang dijual dan barang-barang sisa di

tangan (persediaan akhir) diasumsikan untuk biaya akhir. Karenanya, untuk penentuan pendapatan, biaya-biaya sebelumnya dicocokkan dengan pendapatan dan biaya-biaya yang baru digunakan untuk penilaian laporan neraca. Metode ini konsisten dengan arus biaya aktual, sejak pemilik barang dagang mencoba untuk menjual persediaan lama pertama kali. FIFO merupakan yang paling luas digunakan dalam penilaian persediaan.

Metode FIFO Juga dikenal sebagai metode MPKP (masuk pertama keluar pertama) menurut (Bustami dan Nurlela, 2013) yaitu Persediaan unit dalam proses awal diperlakukan terpisah dengan unit masuk proses dalam periode bersangkutan. Unit dalam proses awal diasumsikan selesai diproses terlebih dahulu.

Berdasarkan penerapan metode tersebut, telah di jelaskan bahwa proses masuk awal akan diselesaikan dan dikeluarkan untuk yang pertama lalu diikuti proses masuk berikutnya, adapun proses MPKM tersebut memiliki rumus untuk melakukan perhitungan yaitu gambar 2.1:

$$\begin{aligned} &(\text{Jumlah Barang Tersedia} - \text{Pemesanan Barang}) = \text{Sisa stok} \\ &\text{Sisa stok} = (\text{Sisa Barang} - \text{Pemesanan Barang}) \\ &\text{Sisa stok} + \text{Jumlah Barang Berikutnya} \end{aligned}$$

Gambar 1.1 Rumus FIFO
Sumber: (Bustami dan Nurlela, 2013)

Tabel 1.2 Kartu Stok

Metode : FIFO

| TGL | KET | MASUK | | | KELUAR | | | SALDO | | |
|------------------------|-----|-------|-------|-------|--------|-------|-------|-------|-------|-------|
| | | QTY | HARGA | TOTAL | QTY | HARGA | TOTAL | QTY | HARGA | TOTAL |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| Nilai Persediaan Akhir | | | | | | | | Unit | | |

1.5.1 Algoritma FIFO

Algoritma FIFO merupakan alur proses pengkodean pada implementasi bahasa pemrograman PHP yaitu sebagai berikut:

```

$cek = $this->session->userdata('logged_in');
if(!empty($cek)){
    $conn = mysqli_connect( 'localhost','root','12345678','bela_tani') or
    die('Could not connect to mysql server. ');
    $id_barang = $_POST['id_barang'];
    $qty = $_POST['jumlah'];

```

1. **Jumlahkan keseluruhan Stok barang yg terpilih**

```

$sql = "SELECT SUM(sisa_stok) AS total FROM barang_masuk
WHERE id_barang = '$id_barang'";
$result = mysqli_query($conn, $sql);
$data = mysqli_fetch_assoc($result);
$stok_all = $data['total'];

```
2. **Lihat Daftar Stok Barang yg terpilih urutkan berdasarkan tanggal ASC (FIFO)**

```

$sql = "SELECT * FROM barang_masuk WHERE id_barang =
'$id_barang' AND sisa_stok > 0 ORDER by tanggal ASC";
$result = mysqli_query($conn, $sql);

```
3. **Bandingkan qty yg masuk dengan stok barang digudang ...**

```

if($qty <= $stok_all) {

```
4. **Lakukan Perulangan pada setiap List Stok Barang hasil (\$result):**

```

while($row = mysqli_fetch_assoc($result)) {
    $tgl = $row['tanggal'];
    $stok = $row['sisa_stok'];

```
5. **Selama Qty > 0 (belum habis) maka => stok pada setiap list akan terus dieksekusi (dikurangi)**

```
if($qty > 0) {
```

6. Buat var \$temp sebagai pengurang

```
$temp = $qty;
```

7. Proses pengurangan

```
$qty = $qty - $stok;
```

8. Jika hasil pengurangan > 0 maka stok pada list pertama kurang, sehingga update stok menjadi 0

```
if($qty > 0) {  
    $stok_update = 0;  
}else{  
    $stok_update = $stok - $temp;  
}  
$sql = "UPDATE barang_masuk SET sisa_stok = $stok_update  
WHERE id_barang = '$id_barang' AND tanggal = '$tgl'";  
mysqli_query($conn, $sql);  
}  
}  
}else{  
    echo " <script>  
    alert('Stok Barang Tidak Cukup, Stok = $stok_all');  
    history.go(-1);  
    </script>";  
    exit();  
}
```

1.6 Metode Pengumpulan Data dan Pengembangan Sistem

Metode pengumpulan data merupakan kegiatan untuk mengumpulkan fakta maupun data terkait permasalahan dan dilakukan pengembangan sistem terhadap permasalahan sebagai bentuk solusi.

1.6.1 Pengumpulan Data

Pengumpulan data merupakan bagian dari kegiatan menemukan sumber dan permasalahan berdasarkan data maupun fakta, pengumpulan data yang dapat dilakukan seperti berikut:

1. Wawancara

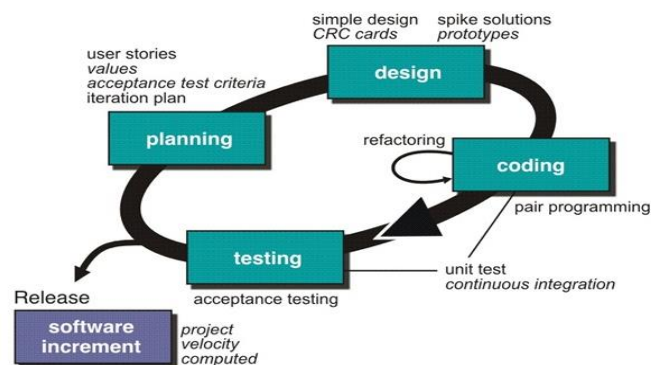
Wawancara digunakan sebagai teknik pengumpulan data apabila peneliti ingin melakukan studi pendahuluan untuk menemukan permasalahan yang harus diteliti.

2. Dokumentasi

Merupakan bentuk pendukung dari pengumpulan data terkait bukti atau fakta terkait permasalahan berupa data tertulis, foto, video hingga dalam bentuk elektronik data.

1.6.2 Pengembangan Sistem

Menurut (Suryantara, 2017) pendekatan *extreme programming* merupakan suatu pendekatan berorientasi objek dan sebagai pengembangan perangkat lunak cepat sedikit lebih rinci dengan tujuan memberikan ulasan secara ringkas. Paradigma yang diinginkan mencakup didalam seperangkat aturan dan praktik-praktik dalam empat konteks kegiatan kerangka kerja yaitu :



Gambar 1.2 *Extreme Programming* (Suryantara, 2017)

1.7 Pemodelan Berorientasi Objek

Menurut (Rosa dan Salahuddin, 2013), pemodelan berorientasi objek merupakan suatu strategi pembangunan perangkat lunak yang mengorganisasikan

perangkat lunak sebagai kumpulan objek yang meliputi rangkaian aktivitas menganalisis, merancang, pemrograman dan pengujian.

1.7.1 Pengertian Pemodelan Berorientasi Objek

Menurut (Rosa dan Salahuddin, 2013), pengertian pemodelan berorientasi objek merupakan tahapan perantara untuk memetakan spesifikasi atau kebutuhan sistem yang akan dibangun dengan konsep berorientasi objek ke pemodelan agar lebih mudah diimplementasikan dengan pemrograman.

1.7.2 Class

Menurut (Rosa dan Salahuddin, 2013), *class* merupakan kumpulan objek-objek dengan karakteristik yang sama. *Class* dapat diartikan himpunan objek yang sama memungkinkan adanya proses kelakuan, hubungan dan atribut. Suatu *class* dapat diturunkan ke *class* yang lain, dimana atribut dan *class* semula dapat diwariskan ke kelas yang baru.

1.7.3 Object

Menurut (Rosa dan Salahuddin, 2013), *object* merupakan bagian-bagian yang memiliki fungsi masing-masing. Objek merupakan suatu entitas yang mampu menyimpan informasi, objek mempunyai siklus hidup yaitu diciptakan, dimanipulasi dan dihancurkan.

1.7.4 Relasi Antar Class

Menurut (Rosa dan Salahuddin, 2013), relasi antar *class* merupakan kumpulan suatu objek yang saling terintegrasi. Umumnya menggambarkan *class*

yang memiliki atribut berupa *class* lain atau hubungan yang menyatakan suatu bagian.

1.7.5 Karakteristik Pemodelan Berorientasi Objek

Menurut (Rosa dan Salahuddin, 2013) karakteristik pemodelan berorientasi objek sebagai berikut:

1. *Encapsulation* (Pembungkusan)

Pembatasan terhadap ruang lingkup program yang ingin dikembangkan

2. *Inheritance* (Pewarisan)

Tehnik yang menyatakan bahwa anak dari objek akan mewarisi data (*Attribute*) dan *method* dari induknya langsung.

3. Generalisasi

Merupakan relasi antar sub *class* dan super *class* di atasnya. *Class* yang rendah akan mewarisi semua atribut dari *class* yang lebih tinggi.

4. *Polymorphisme*

Merupakan aksi yang sama yang dapat dilakukan beberapa objek.

1.8 Pengujian *Black Box*

Menurut (Yakub, 2012) Pengujian sistem adalah proses untuk mengecek apakah suatu perangkat lunak yang dihasilkan sudah dapat dijalankan sesuai standar atau belum. Pengujian sistem dapat menggunakan metode *black box testing* yaitu merupakan pendekatan komplementer dari teknik *white box testing*, karena pengujian *black box testing* mampu mengungkap kesalahan yang lebih luas. *Black box testing* berfokus pada pengujian persyaratan fungsional perangkat

lunak, karena untuk mendapatkan serangkaian kondisi *input* yang sesuai dengan persyaratan fungsional suatu program.