

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Penelitian Terkatit**

##### ***Sri Haryati , Sistem Forecasting Produksi Pada PD. Adi Anugerah Food Industry Tanjung Pinang Dengan Metode Single Exponential Smoothing***

Hasil penelitian ini adalah menghasilkan ramalan produksi bulan berikutnya yang datanya di dapat dari hasil penjualan agar dapat membantu perusahaan meminimalkan kerugian penumpukan stok sisa penjualan, maka dari hasil penelitian dua metode ini dapat dilihat bahwa metode single eksponential smoothing lebih cocok karena hasil ramalan mendekati nilai sebenarnya, yaitu pada data actual SNGL-B adalah 1200/bungkus maka pada SES 1185/bungkus dengan menggunakan nilai alfa 0,8 dan pada Double hasil ramalannya 1257/bungkus dengan alfa 0,9.

##### ***Astriani Agus Setyowati, Penerapan Metode Single Exponential Smoothing dan Double Exponential Smoothing Pada Peramalan Penjualan Pakaian***

Metode *Single Exponential Smoothing* dan *Double Exponential Smoothing* memanfaatkan data *history* penjualan untuk melakukan peramalan , semakin banyak data *history* yang digunakan untuk peramalan maka semakin akurat pula hasil dari peramalan yang dilakukan .Hasil penelitian ini telah berhasil membuat sistem peramalan penjualan pakaian dengan menggunakan metode *Single Exponential Smoothing* dan *Double Exponential Smoothing* sehingga pengguna

dapat lebih mudah untuk menentukan jumlah produksi yang sesuai dengan penjualan sehingga tidak terjadi penumpukan barang.

**Arif Ardiansyah Hasibuan, *Sistem Informasi Pengendalian Persediaan Bahan Baku Obat Tradisional Dengan Metode Single Moving Average dan Single Exponential Smoothing***

Sistem Informasi Pengendalian Persediaan Bahan Baku Obat Tradisional Dengan Metode *Single Moving Average* dan *Single Exponential Smoothing* dibuat agar memberikan informasi peramalan yang berkaitan dengan jumlah obtra yang dibutuhkan pada periode mendatang. Setiap obtra memiliki data time series yang berbeda sehingga masing-masing obtra memiliki panjang periode ramalan terbaik dan alpha terbaik yang berbeda dengan obtra lainnya

Sistem Informasi Pengendalian Persediaan Bahan Baku Obat Tradisional Dengan Metode *Single Moving Average* dan *Single Exponential Smoothing* ini mampu menyajikan laporan sesuai dengan yang diharapkan oleh Laboratorium Penelitian dan Pengembangan Pelayanan Obat Tradisional (LP4OT)

**Sayed Fachrurarzi S.si, M.kom , *Peramalan Penjualan Obat Menggunakan Metode Single Exponential Smoothing pada Toko Obat Bintang Geurugok***

Toko obat bintang sangat memerlukan peramalan penjualan obat untuk meningkatkan keuntungan dan menghindari terjadinya kelebihan maupun kekurangan persediaan jumlah obat. Metode *Single Exponential Smoothing* hanya dapat meramalkan satu periode kedepan. Pada Tahap uji sampel didapat bahwa

peramalan penjualan obat pada obat *Ambeven* bulan maret 2015 berjumlah 49 tablet.

**Kristien Maragi , *Analisa dan Penerapan metode Single Exponential Smoothing untuk memprediksi penjualan pada periode tertentu (Studi kasus : PT. Media Cemara Kreasi )***

Implementasi metode *Single Exponential Smoothing* dapat diterapkan pada sistem prediksi penjualan perbulan. Aplikasi belum dapat digunakan sebagai referensi untuk menghitung prediksi penjualan untuk tiap tahunnya.

**Nendang Kacical , *Sistem Peramalan Dan Monitoring Persediaan Obat Di RSPG CISARUA BOGOR Dengan Menggunakan Metode Single Exponential Smoothing Dan Reorder Point***

Berdasarkan hasil pengujian, dapat disimpulkan bahwa metode *Single Exponential Smoothing* cocok untuk digunakan karena rata-rata akurasi peramalan mencapai 93,4%, selain itu sistem yang dibangun membantu Kepala Instalasi Farmasi dalam menentukan jumlah obat yang harus disediakan dan membantu bagian gudang dalam memantau persediaan stok obat.

## **II.2.Landasan Teori**

### **II.2.1 Sistem**

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi dalam sistem tersebut. (Tata Sutabri, 2012 : 22)

### **II.2.2 Informasi**

Informasi adalah segala sesuatu keterangan yang bermanfaat untuk para pengambil keputusan/manajer dalam rangka mencapai tujuan organisasi yang sudah ditetapkan. (Jimmy, 2008 : 8).

### **II.2.3 Sistem Informasi**

Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan yang diberikan. (Tata Sutabri, 2012 : 46).

### **II.2.4 Peramalan**

Peramalan merupakan gambaran keadaan perusahaan pada masa yang akan datang. Gambaran tersebut sangat penting bagi manajemen perusahaan karena dengan gambaran tersebut maka perusahaan dapat memprediksi langkah-langkah apa saja yang diambil dalam memenuhi permintaan konsumen. Ramalan memang tidak selalu tepat 100%, karena masa depan mengandung masalah ketidakpastian, namun dengan pemilihan metode yang tepat dapat membuat peramalan dengan tingkat kesalahan yang kecil.

Sumayang (2003) mendefinisikan Peramalan adalah perhitungan yang objektif dan dengan menggunakan data-data masa lalu, untuk menentukan sesuatu di masa yang akan datang. Hal ini serupa dengan pendapat Render dan Heizer (2005) Peramalan adalah seni dan ilmu untuk memperkirakan kejadian di masa depan. Menurut Subagyo (2002) Forecasting adalah memperkirakan sesuatu yang akan terjadi. Menurut Gasperz (2005) Aktivitas peramalan merupakan suatu fungsi bisnis yang berusaha memperkirakan penjualan dan penggunaan produk sehingga produk-produk itu dapat dibuat dalam kuantitas yang tepat. Menurut Nasution (2003) Peramalan adalah Proses untuk memperkirakan beberapa kebutuhan dimasa yang akan datang yang meliputi kebutuhan dalam ukuran kuantitas, kualitas, waktu dan lokasi yang dibutuhkan dalam rangka memenuhi permintaan barang dan jasa. Dari kelima pendapat para ahli di atas maka dapat ditarik kesimpulan bahwa peramalan adalah memperkirakan sesuatu yang akan terjadi dengan menggunakan data-data masa lalu (Sayed Fachrurazi 2015 ; 21).

### II.2.5 Metode *Single Exponential Smoothing*

Juga dikenal sebagai *simple exponential smoothing* yang digunakan pada peramalan jangka pendek, biasanya hanya 1 bulan kedepan. Model mengasumsikan bahwa data berfluktuasi di sekitar nilai mean yang tetap, tanpa trend atau pola pertumbuhan konsisten. Rumus untuk *single exponential smoothing* adalah sebagai berikut :

$$F_{t+1} = a * X_t + (1 - a) * F_t \dots\dots\dots (1)$$

Dimana :

$F_t$  = Peramalan untuk periode t

- $X_t$  = Nilai actual time series
- $F_{t+1}$  = Peramalan Pada Waktu  $t+1$
- $a$  = Konstanta perataan antara 0 dan 1 (Astriani Agus Setyowati ; 2017 : 4)

Metode ini membutuhkan nilai alpha ( $\alpha$ ) sebagai nilai parameter pemulusan. Bobot nilai  $\alpha$  lebih tinggi diberikan kepada data yang lebih baru, sehingga nilai parameter  $\alpha$  yang sesuai akan memberikan ramalan yang optimal dengan nilai kesalahan terkecil. Untuk mendapatkan nilai  $\alpha$  yang tepat pada umumnya dilakukan dengan *trial and error* untuk menentukan nilai kesalahan terendah. Nilai  $\alpha$  dilakukan dengan membandingkan menggunakan interval pemulusan antar  $0 < \alpha < 1$ , yaitu  $\alpha$  (0,1 sampai dengan 0,9). Metode ini hanya mampu memberikan ramalan satu periode ke depan dan cocok untuk data yang mengandung unsur stationer. Karena jika diterapkan pada serial data yang memiliki *trend* yang konsisten, ramalan yang dibuat akan selalu berada dibelakang *trend*. Selain itu, metode eksponensial ini juga memberikan bobot yang relatif lebih tinggi pada nilai pengamatan terbaru dibanding nilai-nilai periode sebelumnya.

## II.2.6 PHP

PHP (*Personal Home Page*) adalah pemrograman (*interpreter*) adalah proses penerjemahan baris sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan. Definisi lain mengenai PHP adalah skrip bersifat *server-side* yang ditambahkan ke dalam HTML. PHP sendiri

merupakan singkatan dari *Personal Home Page Tools*. Skrip ini akan membuat suatu aplikasi dapat diintegrasikan ke dalam HTML sehingga suatu halaman web tidak lagi bersifat statis, namun menjadi bersifat dinamis. Berdasarkan beberapa pengertian PHP tersebut dapat disimpulkan bahwa PHP (*Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server-side* yang dapat ditambahkan ke dalam HTML (Supono : 2016 : 3).




### **II.2.7 MySQL**

MySQL merupakan *database* yang sangat populer digunakan bersama PHP. PHP dengan MySQL adalah kombinasi yang baik untuk menampilkan kekuatan sesungguhnya dari *server-side scripting*. Dengan MySQL data dapat disimpan lebih efisien dibandingkan menyimpan data dalam array. Untuk menggunakan MySQL ataupun *database* secara efektif, dibutuhkan pemahaman *structured query language*. (Dian: 2017 : 655).

### **II.2.8 Pengertian ERD**

ERD adalah suatu diagram untuk menggambarkan desain konseptual dari model konseptual suatu basis data relasional. ERD juga merupakan gambaran yang merelasikan antara objek yang satu dengan objek yang lain dari objek di dunia nyata yang sering dikenal dengan hubungan antar entitas. Sebagai contoh jika membuat ERD dari sistem perpustakaan maka bahan sebagai objek ERD bisa berupa anggota, buku, peminjam, pengembalian dan sebagainya. ERD terdiri dari 4 komponen utama, yaitu (Robi Yanto ; 2016 : 32) :

Tabel II.1. Simbol ERD

Notasi	Keterangan
	<b>Entitas</b> , adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
	<b>Relasi</b> , menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
	<b>Atribut</b> , berfungsi mendeskripsikan karakter entitas (atribut yg berfungsi sebagai key diberi garis bawah)

(Sumber : Robi Yanto ; 2016 : 32)

## II.2.9 Pengertian Normalisasi

Normalisasi adalah suatu proses untuk membuat data yang tidak normal menjadi data yang normal. Bentuk data yang tidak normal / data mentah biasa disebut juga *unnormalized form*. Masing – masing level normalisasi mempunyai aturan tersendiri.

### 1. *First Normal Form*

Suatu tabel dikatakan dalam keadaan *first normal form* (1NF) jika :

- a. Tidak ada perulangan record data dalam tabel.
- b. Setiap sel memiliki satu nilai saja. Artinya tidak ada perulangan group dan array.
- c. Data yang diinputkan memiliki tipe data yang sama dengan tipe data kolom dalam tabel.

## 2. *Second Normal Form*

Suatu tabel dikatakan dalam keadaan *Second Normal Form* (2NF) jika tabel tersebut sudah dalam keadaan *First Normal Form* (1NF) dan jika semua atribut yang bukan kunci tabel, baik *primary key* maupun *foreign key* tergantung pada semua kunci dalam tabel.

## 3. *Third Normal Form*

Suatu tabel dikatakan dalam keadaan *third normal form* (3NF) jika tabel tersebut sudah dalam keadaan *second normal form* (2NF) dan jika tidak terdapat ketergantungan yang transitif. Artinya, data-data yang mungkin diisi berulang-ulang dapat dibuat sebuah tabel baru.

## 4. *Boyce-Codd Normal Form* (BCNF)

Tabel dikatakan dalam keadaan *boyce-codd normal form* (BCNF) jika tabel tersebut dalam keadaan *third normal form* (3NF) dan setiap determinan adalah kunci kandidat.

## 5. *Fourth Normal Form* (4NF)

Suatu tabel dikatakan dalam keadaan *fourth normal form* (4NF) jika tabel tersebut dalam keadaan *boyce-codd normal form* (BCNF) dan jika tidak terdapat ketergantungan nilai ganda.

## 6. *Fiveth Normal Form* (5NF)

Tabel dikatakan dalam keadaan *Fiveth Normal Form* (5NF) jika tabel tersebut dalam keadaan *fourth normal form* (4NF) dan jika setiap ketergantungan dalam join ada pada tabel sudah konsekuen dengan kunci kandidat pada tabel tersebut (Ema Utami ; 2012 : 73-76).

### II.2.10 UML (*Unified Modeling Language*)


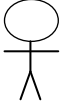


Menurut Gata (2013 : 4) Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak.

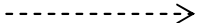
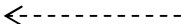
UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

#### 1. *Use case* Diagram

*Use case* diagram 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. Dapat dikatakan *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *use case* diagram, yaitu :

Tabel II.2. Simbol *Use Case*

Gambar	Keterangan
	<p><i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i>.</p>
	<p>Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i>, tetapi tidak memiliki control terhadap <i>use case</i>.</p>
	<p>Asosiasi antara aktor dan <i>use case</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.</p>
	<p>Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.</p>



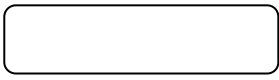
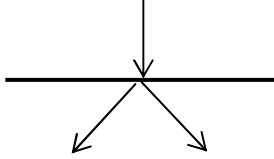
	<i>Include</i> , merupakan di dalam <i>use case</i> lain ( <i>required</i> ) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

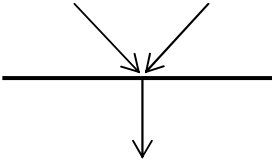
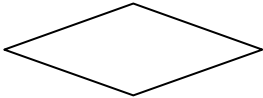

(Sumber : Gata, 2013 : 4)

## 2. Diagram Aktivitas (*Activity Diagram*)

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram*, yaitu :

**Tabel II.3. Simbol *Activity Diagram***

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.

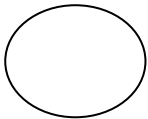
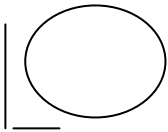
	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

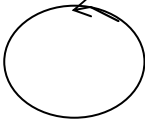

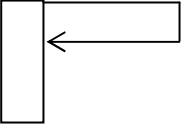

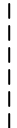
(Sumber : Gata, 2013 : 6)

### 3. Diagram Urutan (*Sequence Diagram*)

*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram*, yaitu :

**Tabel II.4. Simbol *Sequence Diagram***

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.

	<p><i>Control class</i>, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.</p>
	<p><i>Message</i>, simbol mengirim pesan antar <i>class</i>.</p>
	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

(Sumber : Gata, 2013 : 7)

#### 4. *Class Diagram* (Diagram Kelas)

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

*Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/Method*), *Visibility*, tingkat akses objek eksternal kepada suatu

operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau kardinaliti.

**Tabel II.5. *Multiplicity Class Diagram***

<b>Multiplicity</b>	<b>Penjelasan</b>
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

**(Sumber : Gata, 2013 : 9)**