



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

LANDASAN TEORI

2.1 Sistem

Sistem adalah kumpulan dari komponen-komponen yang saling terhubung yang bekerjasama untuk mencapai suatu tujuan yang sama dengan menerima suatu *input* data dan menghasilkan *output* dalam suatu proses yang terorganisasi (O'Brien, 2010).

Sistem adalah sebuah struktur konseptual yang tersusun dari fungsi-fungsi yang saling berhubungan yang bekerja sebagai suatu kesatuan organik untuk mencapai suatu hasil yang diinginkan secara efektif dan efisien (McManama, 1971).

Dengan demikian menurut penulis sistem adalah suatu kesatuan dari beberapa komponen yang saling terhubung dan bekerja dengan tujuan untuk memudahkan suatu kegiatan.

2.2 Teori Pengembangan Web

2.2.1 Web

Website merupakan fasilitas internet yang menghubungkan dokumen dalam lingkup lokal maupun jarak jauh. Dokumen pada website disebut dengan web page dan link dalam website memungkinkan pengguna bisa berpindah dari satu page ke page lain (hyper text), baik diantara page yang disimpan dalam server yang sama maupun server diseluruh dunia. Pages diakses dan dibaca melalui browser seperti Netscape Navigator, Internet Explorer, Mozilla Firefox, Google Chrome dan aplikasi browser lainnya (Hakim, 2004).

2.2.2 PHP

PHP adalah singkatan dari "PHP Hypertext Processor", yang merupakan sebuah bahasa *scripting* yang terpasang pada HTML. Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamik dengan cepat (Peranginangin, 2006).

Bahasa pemrograman PHP dapat digunakan pada berbagai sistem operasi. Selain itu PHP juga kompatibel dengan teknologi yang digunakan pada berbagai *Web Server* masa kini seperti Apache, IIS, dan teknologi *web server* lainnya. Dengan demikian, pembuatan *website* dengan menggunakan bahasa pemrograman PHP mampu memberikan kemudahan kepada penggunanya, baik *end user* maupun *developer* karena dapat berjalan di berbagai sistem operasi dan *web server*.

2.2.3 MySql

MySQL adalah suatu perangkat lunak *database* relasi (Relational *Database Management System* atau DBMS). SQL merupakan singkatan dari *Structure Query Language*, didefinisikan sebagai suatu sintaks perintah-perintah tertentu atau bahasa program yang digunakan untuk mengelola suatu *database*. Jadi MySQL adalah software-nya dan SQL adalah bahasa perintahnya (Anisya, 2013).

2.2.4 AJAX (Asynchronous JavaScript and XML)

AJAX merupakan teknik baru dalam menggunakan standar pemrograman yang sudah ada. AJAX adalah suatu teknik pemrograman

berbasis web untuk menciptakan web interaktif. Tujuannya adalah untuk memindahkan sebagian besar interaksi pada *web surfer*, melakukan pertukaran data dengan *server* dibelakang layar sehingga halaman web tidak harus dibaca ulang secara keseluruhan setiap kali pengguna melakukan perubahan. Hal ini akan meningkatkan interaktivitas, kecepatan, dan *usability*. AJAX merupakan kombinasi dari beberapa teknologi yaitu HTML, CSS, *Javascript*, *jQuery*, dan PHP.

2.3 Tools Analisis Sistem

2.3.1 Konsep Dasar RAD (*Rapid Application Development*)

Rapid Application Development (RAD) adalah strategi siklus hidup yang ditujukan untuk menyediakan pengembangan yang jauh lebih cepat dan mendapatkan hasil dengan kualitas yang lebih baik dibandingkan dengan hasil yang dicapai melalui siklus tradisional (McLeod, 2001).

RAD merupakan gabungan dari bermacam-macam teknik terstruktur dengan teknik *prototyping* dan teknik pengembangan *joint application* untuk mempercepat pengembangan sistem/aplikasi (Whitten L, 2004)).

Dari definisi-definisi konsep RAD ini, dapat dilihat bahwa pengembangan aplikasi dengan menggunakan metode RAD ini dapat dilakukan dalam waktu yang relatif lebih cepat.



Gambar 2.1 Rapid Application Development

Model pengembangan RAD memiliki empat fase, yaitu fase Perencanaan syarat-syarat, fase perancangan, fase konstruksi, dan fase pelaksanaan. Berikut adalah penjelasan masing-masing fase dalam penelitian ini (Bentley, 2007).

1. Fase Perencanaan Syarat-Syarat

Pada tahap ini dilakukan pengidentifikasian tujuan aplikasi atau sistem serta untuk mengidentifikasi syarat-syarat Informasi yang ditimbulkan dari tujuan-tujuan tersebut. Pada tahap ini peneliti melakukan analisis kebutuhan.

2. Fase Perancangan

Pada tahap ini adalah melakukan proses desain dan melakukan perbaikan apabila masih terdapat ketidaksesuaian desain antara *user* dan analisis. Untuk tahap ini keaktifan *user* yang terlibat sangat menentukan untuk mencapai tujuan, karena *user* bisa langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain.

3. Fase Konstruksi

Setelah desain dari sistem yang akan dibuat sudah disetujui baik itu oleh *user* dan analisis, maka pada tahap ini programmer mengembangkan desain menjadi suatu program. Hal terpenting adalah keterlibatan *user* sangat diperlukan supaya sistem yang dikembangkan dapat sesuai dengan permintaan *user*.

4. Fase Pelaksanaan

Pada tahap ini dilakukan pengujian aplikasi, dengan menggunakan pengujian black box testing. Yaitu Sebuah pengujian dengan menjalankan setiap fungsi dari aplikasi.

2.4 *Inventory Management*

Inventory Management adalah Meminimalkan investasi dalam persediaan namun tetap konsisten dengan penyediaan tingkat pelayanan yang diminta ((Deveshwar, 2016).

Manajemen persediaan adalah meminimalkan investasi dalam persediaan namun tetap konsisten dengan penyediaan tingkat pelayanan yang diminta. Dalam manajemen persediaan terdapat 2 hal keputusan persediaan yang perlu diperhatikan, keputusan persediaan yang bersifat umum merupakan keputusan yang menjadi tugas utama dalam penentuan persediaan baik secara kuantitatif maupun kualitatif (Alfredo, 2014).

Keputusan kuantitatif bertujuan untuk mengetahui barang apa yang akan di *stock*, berapa banyak jumlah barang yang akan diproses dan berapa banyak barang yang akan dipesan, kapan pembuatan barang akan dilakukan dan kapan melakukan pemesanan, kapan melakukan pemesanan ulang (*Reorder Point*). Keputusan secara

kualitatif adalah keputusan yang berkaitan dengan teknik pemesanan yang mengarah pada analisis data secara deskriptif. Keputusan secara kualitatif juga bertujuan untuk mengetahui jenis barang apa saja yang tersedia di perusahaan (Alfredo, 2014).

Inventory Management sendiri merupakan model *inventory* tradisional berdasarkan permintaan yang diantisipasi dan bertujuan untuk mencegah terjadinya kehabisan persediaan, menjaga pembentukan persediaan tidak terlalu besar atau berlebihan, dan menjaga agar pembelian kecil dapat dihindari (Charles, 2014).

2.5 Teori Persediaan

Warren yang diterjemahkan Farahmita (2006, h. 452) dalam bukunya yang berjudul *Accounting – Pengantar Akuntansi* mengatakan bahwa “Persediaan (*inventory*) digunakan untuk mengindikasikan (1) barang dagang yang disimpan untuk kemudian dijual dalam bisnis perusahaan dan (2) bahan yang digunakan dalam proses produksi atau yang disimpan untuk tujuan itu”.

2.6 Teori Safety Stock

Safety Stock adalah sebagai persediaan *extra* yang disimpan sebagai jaminan atas fluktuasi permintaan. Bahwa penggunaan maksimal dari suatu barang juga ikut menentukan besarnya *Safety Stock* dari suatu persediaan, apalagi dengan adanya ketidakpastian permintaan dimasa depan.

Safety Stock dapat dihitung dengan rumus:

Rumus 2.1 Rumus *Safety Stock*

$$SS = LT * (MU - AU)$$

Catatan :

SS	= <i>Safety Stock</i>
LT	= <i>Leadtime</i>
Max Usage	= Maksimal permintaan
Average Usage	= Rata-rata permintaan

2.7 Teori ROP

Titik pemesanan kembali (*Reorder Point*) adalah titik waktu dimana sebuahh pemesanan baru harus dilakukan. Tenggang waktu (*leadtime*) adalah waktu yang diperlukan untuk menerima item yang telah dipesan ke *supplier* sampai barang tersebut masuk toko material (Mowen, 2013).

Dengan mengetahui tingkat penggunaan dan tenggang waktu maka dimungkinkan untuk dapat menghitung titik pemesanan kembali (ROP) dengan rumus:

Rumus 2.2 Rumus *Reorder Point*

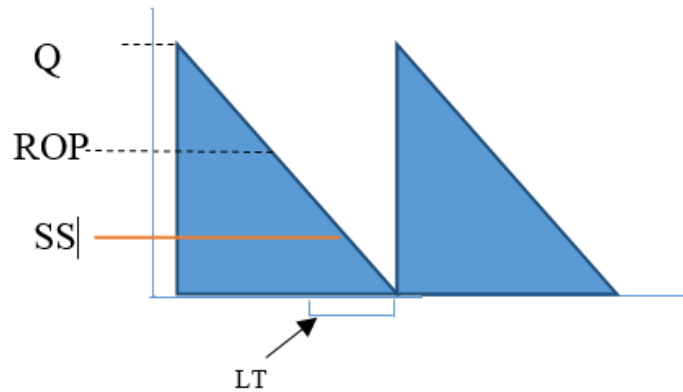
$$Reorder Point = (AU \times LT) + Safety Stock$$

Catatan:

AU = *Average Usage* (rata-rata penggunaan)

LT = *Leadtime* (Jeda waktu barang dipesan hingga barang sampai toko)

Jika digambarkan grafis akan seperti ini:



Catatan :

- Q = Quantity
- ROP = *Reorder Point*
- SS = *Safety Stock*
- LT = *Leadtime*
- ▲ = *Stock*

2.8 Point of Sale

Pengertian *Point of Sale* atau yang biasa yang disingkat POS yaitu, merupakan kegiatan yang berorientasi pada penjualan serta sistem yang membantu proses transaksi. Setiap POS terdiri dari *hardware* dan *software* dimana kedua komponen tersebut digunakan untuk setiap proses transaksi.

Berikut adalah fungsi *Point of Sale* bagi para pemilik usaha (Hendra, 2015):

1. Mendata setiap transaksi secara lengkap dan detail sehingga dapat menjumlahkan hasil penjualan pada saat itu ataupun dapat dipilih dalam periode tertentu.

2. Dapat menggunakan cek persediaan barang dimana pun secara acak. Hal ini akan mengurangi kecurangan atau kelalaian Anda dan para karyawan.

3. Laporan penjualan usaha Anda dapat diketahui secara *online* dan *real-time*. Jurnal Informasi Volume VII No.2 / November / 2015 62

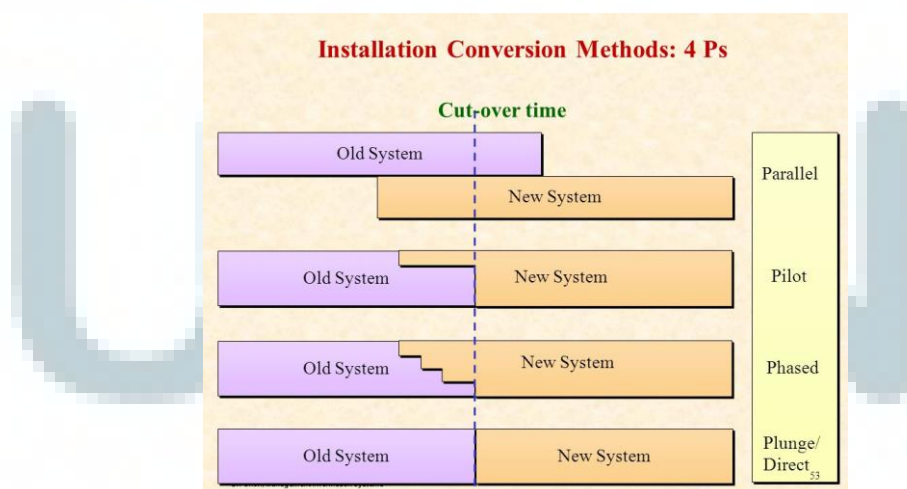
4. Anda dapat mengubah harga jual secara cepat dan mudah. Misalkan apabila barang yang Anda jual memang biasa naik dan turun mengikuti nilai tukar mata uang asing, Anda tidak perlu mengganti satu per satu dan memakan banyak waktu, namun dapat Anda ganti berdasarkan kategori barang.

5. Dapat mengetahui persediaan barang apa saja yang masih memiliki banyak stok ataupun yang mendekati habis sehingga Anda bisa dengan cekatan menyetok ulang barang tersebut.

6. Mempersingkat proses transaksi dan menjaga kenyamanan dan keamanan setiap transaksi yang berlangsung.

2.9 Pendekatan konversi sistem

Di dalam proses implementasi terdapat pendekatan konversi yang bertujuan untuk mengubah *user* yang awalnya menggunakan sistem lama menjadi sistem baru. Berikut ini merupakan 4 jenis pendekatan konversi sistem (cga-pdnet, 2016):



Gambar 2. 2 jenis Pendekatan Konversi Sistem

1. Konversi Langsung (*Direct Conversion*)

Konversi langsung merupakan strategi conversion yang dilakukan dengan cara menghentikan sistem lama kemudian digantikan dengan sistem baru. Dengan menggunakan pendekatan ini maka tidak ada cara untuk kembali menggunakan sistem lama

2. Konversi Paralel (*Parallel Conversion*)

Konversi ini dilakukan dengan cara menjalankan sistem lama dengan sistem baru secara bersamaan. Jika sistem baru sudah mulai bisa diterima oleh *user*, sistem lama segera dihentikan.

3. Konversi Bertahap (*Phase-In Conversion*)

Konversi bertahap dilakukan dengan cara menggantikan suatu bagian dari sistem lama dengan sistem baru. Jika ada hal yang tidak diharapkan terjadi, bagian yang baru diterapkan dapat diganti kembali dengan sistem lama.

4. Konversi Pilot (*Pilot Conversion*)

Konversi Pilot dilakukan dengan cara menerapkan sistem baru hanya pada bagian tertentu untuk menjadi pelopor. Jika bagian yang telah diubah menjadi sistem baru berhasil jalan, maka akan diberlakukan pada bagian lainnya.

2.10 Arsitektur Sistem

Arsitektur *Informasi* adalah desain item komputer secara keseluruhan (termasuk sistem jaringan) untuk memenuhi kebutuhan-kebutuhan organisasi yang spesifik (Azays, 1998).

Arsitektur Informasi adalah bentuk khusus yang menggunakan teknologi Informasi dalam organisasi untuk mencapai tujuan-tujuan atau fungsi-fungsi yang telah dipilih (Loudon, 1998)

Jadi menurut penulis arsitektur adalah sebuah gambaran struktur teknologi dan komponen sistem yang dibutuhkan untuk menjalankan sebuah sistem.

