



## **BAB II**

# **DASAR TEORI**

## BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI

### 2.1 Tinjauan Pustaka

Penelitian mengenai inventori sebelumnya pernah dilakukan pada tahun 2017 mengenai sistem informasi inventori yang berjudul “Sistem Informasi Inventori Gudang untuk Mengontrol Persediaan Barang pada Gudang Studi Kasus : PT.Alaisys Sidoarjo” oleh Hendra Agusvianto dari Program Studi S2 Pendidikan Teknologi dan Kejuruan, Universitas Negeri Surabaya. PT.Alaisys didirikan pada tahun 2010, bergerak dibidang usaha perdagangan, peridustrian, dan jasa. Pada bidang perdagangan usahanya meliputi *import* dan perdagangan peralatan pada bagian multimedia, listrik, elektronik, distributor, agen dan sebagai perwakilan dari badan perusahaan. Metode penelitian yang digunakan yaitu menggunakan studi literatur, kebutuhan sistem, analisis desain sistem. Hasil dari penelitian PT Alaisys ini digunakan untuk membuat suatu aplikasi berupa sistem inventori pada gudang yang berfungsi untuk mencatat keluar masuk barang oleh staf PT.Alaisys, yang dapat dilakukan dengan struktur sehingga memberikan bantuan dalam hal efisiensi waktu kegiatan pencatatan dan penyusunan data pada barang [1].

Penelitian selanjutnya dengan judul “Peranan Sistem Informasi Dalam Membantu Stock Opname Barang di Gudang Koperasi Warga Semen Gresik (Studi Kasus Di Gudang Kwsg Banjarrejo 38 Polos)” dilakukan pada tahun 2018 oleh Anjar Prayogi, Jurusan Ekonomi Syariah Fakultas Ekonomi dan Bisnis Islam, Institut Agama Islam Negeri (IAIN) Metro. Metode yang digunakan dalam penelitian adalah deskriptif kualitatif. Pengumpulan data dengan menggunakan metode wawancara dan dokumentasi. Analisis data menggunakan teknik analisis data kualitatif dengan cara berfikir induktif. Hasil dari penelitian ini dapat bermanfaat bagi peneliti dan karyawan semen gresik serta menambah wawasan keilmuan peneliti khususnya dibidang persediaan barang [2].

Penelitian lainnya dengan judul “Aplikasi Sistem Informasi Pemantauan *Inventory Stock Opname* Berbasis Web pada Pt Makmur Berkat Solusi Logistic” oleh Sugeng Santoso, Ilamsyah, Winda Novita pada tahun 2019. PT. Makmur Berkat Solusi merupakan perusahaan yang bergerak dibidang jasa *supply chain & logistic services* yang

bertujuan untuk merancang sistem pemantauan stok yang dibutuhkan perusahaan dalam melakukan kegiatan *Inbound*, *Outbound*, dan pengiriman stok secara cepat dan efektif untuk dapat membantu dalam proses pekerjaan yang dilakukan. Sistem yang berjalan saat ini masih manual menggunakan *Ms.Excel* untuk laporan stok dan menggunakan *email* untuk laporan stok. Apabila semua kegiatan di dalam gudang selesai, baru laporan stok di kirimkan ke konsumen. Untuk mempermudah, dan membantu meningkatkan pengelolaan barang, terutama mengenai jenis barang, penginputan data keluar masuk barang, pembuatan laporannya bagi bagian administrasi. Baik dalam proses penginputan data, maupun pembuatan laporan sistem sebagai penyimpanan data secara komputerisasi sehingga mudah untuk di *update*. Metode penelitian yang digunakan yaitu metode observasi, studi pustaka dan wawancara, sedangkan metode analisis yang digunakan yaitu *value chain* [3].

Penulis bermaksud untuk mengembangkan suatu sistem informasi inventori barang pada Gudang Koperasi Annisa dengan menggunakan bahasa pemrograman PHP, HTML dan metode pengembangan *prototype*. Dengan menggunakan media *website* maka informasi bisa diakses lebih mudah, karena telah tekomputerisasi dengan baik. Selain itu, mampu memberikan informasi persediaan barang, mengelola proses pendataan barang masuk dan barang keluar, memberikan laporan data hasil distribusi, memberikan laporan permintaan barang yang berupa data pada komputer secara tepat, serta memberikan bukti nyata penerimaan barang distribusi.

Dapat disimpulkan dari hasil penelitian lain yang telah dilakukan sebelumnya, pada penelitian pertama, perusahaan tersebut bergerak pada bidang perdagangan, peridustrian, jasa yang berfokus pada pembelian dan penjualannya, sedangkan sistem yang akan peneliti kembangkan berfokus pada barang masuk dan keluar, serta bukti barang telah sampai pada *customer*. Selanjutnya pada penelitian kedua, sistem tersebut hanya membahas mengenai *stock opname* yang mana sistemnya berhenti pada saat barang masuk lalu dicek menggunakan metode *stock opname* dan dilaporkannya kepada manajerial, sedangkan sistem yang peneliti kembangan membahas mengenai barang masuk dan keluar, tidak hanya membahas metode *stock opname*-nya saja. Pada penelitian ketiga, sistem bergerak pada jasa pemasok yang di dalam sistem tersebut tidak ada bukti pengiriman barang telah diterima, sedangkan sistem yang akan peneliti kembangkan terdapat bukti yang

sah dari admin gudang menggunakan tanda tangan yang di *upload* pada sistem ketika barang telah diterima.

## **2.2 Landasan Teori**

### **2.2.1 Sistem Informasi**

Sistem informasi ialah kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang terorganisir, menjadikan kegiatan dalam dunia usaha menjadi lebih efektif dan efisien, serta menjamin keberhasilan pekerjaan dalam bidang apapun. Sistem informasi adalah sistem dalam organisasi yang dapat menyediakan kebutuhan pengelolaan, mendukung jalannya suatu kegiatan, serta merupakan kegiatan suatu organisasi dengan menyediakan laporan yang dibutuhkan. Sistem informasi memiliki beberapa komponen pendukung, yaitu terdiri dari *hardware*, *software*, telekomunikasi, *database* dan data *warehouses*, serta sumber daya manusia[4].

Sistem Informasi memiliki kualitas bagaimana sistem informasi tersebut layak untuk digunakan, yaitu dengan memperhatikan kualitas sistem informasi yang meliputi akurat, tepat waktu, serta relevan. Akurat yaitu informasi yang didapatkan bebas dari kesalahan atau dengan kata lain tidak menyesatkan. Tepat waktu yaitu informasi harus diberikan tepat waktu atau tidak terlambat, karena informasi yang datang terlambat sudah tidak dapat digunakan lagi untuk membuat keputusan. Relevan yaitu informasi memiliki manfaat sesuai dengan kebutuhan pengguna dengan tingkat relevan berbeda sesuai pemakaian [5].

### **2.2.2 Inventory**

*Inventory* adalah pengelolaan barang oleh perusahaan yang bertujuan untuk dijual dan dapat berupa bahan mentah, ataupun barang jadi. Pengelolaan barang yang dimaksud dapat berupa pengelolaan stok, pendataan barang masuk dan keluar, dan juga pencatatan data *supplier*. *Inventory* merupakan sebuah data yang rinci berisi persediaan barang dengan keterangan jumlah barang yang dimiliki organisasi di waktu tertentu[6].

### 2.2.3 Pendataan Barang

Pendataan barang adalah suatu kegiatan yang dilakukan oleh pelaku usaha untuk mendata atau melakukan manajemen data barang, baik secara manual ataupun komputerisasi (sistem). Pendataan barang terbagi menjadi pendataan barang masuk dan barang keluar, yang mana pendataan barang masuk bertujuan untuk mendata barang pada saat barang masuk ke dalam stok, sedangkan pendataan barang keluar bertujuan untuk mendata barang pada saat barang keluar [7].


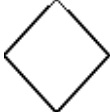


### 2.2.4 Manajemen Stok

Manajemen stok merupakan kegiatan untuk memantau atau melakukan pengontrolan stok barang, baik barang masuk & keluar, ataupun barang digudang. Manajemen stok digunakan untuk memasukkan persediaan barang dalam *database* serta membuat laporan sesuai dengan keadaan yang sebenarnya. Kegiatan ini bertujuan untuk mempermudah dalam melihat data stok barang pada waktu tertentu [8].

### 2.2.5 Basis Data

Basis data adalah kelompok himpunan dari data yang saling terorganisir dan terhubung sedemikian rupa sehingga dapat dimanfaatkan kembali secara cepat dan mudah. Pengertian lainnya adalah berbagai data dalam bentuk arsip/file/tabel yang terhubung dan tersimpan pada penyimpanan elektronik, guna memudahkan pengaturan, pemilahan, pengorganisasian, dan pengelompokan. Basis data memiliki beberapa komponen bahasa, menurut fungsinya, bahasa basis data meliputi DDL atau *Data Definition Language* yang merupakan kumpulan definisi yang tersimpan dalam data *dictionary*. Contoh perintah DDL yaitu *CREATE*, *ALTER*, *RENAME*, dan *DROP*. Selanjutnya ada DML atau *Data Manipulation Language* yang berisi akumulasi pengoperasian manipulasi basis data atau biasa disebut dengan bahasa *query*. Menyediakan fungsi untuk menambah, memodifikasi, menghapus dan mengambil data dari basis data. Contoh dari perintah DML yaitu *INSERT*, *UPDATE*, *DELETE*, dan *SELECT*[9]. Adapun ERD adalah suatu bentuk diagram yang menjelaskan hubungan antara objek – objek data yang memiliki hubungan atau relasi. ERD memiliki fungsi sebagai pemodelan basis data yang dapat dikembangkan menjadi berbagai *project* dalam sistem informasi

Tabel 2. 1 Simbol *Entity Relationship* Diagram

No	Gambar Simbol	Keterangan
1.		<b>Entity</b> , memberikan identitas pada entitas yang memiliki label dan nama.
2.		<b>Relationship</b> , mengetahui jenis hubungan yang ada antara dua file.
3.		<b>Atribut</b> , memperjelas atribut yang dimiliki suatu entitas.
4.		<b>Atribut Key</b> , <i>field</i> data yang butuh disimpan dalam entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan.

### 2.2.6 Rekayasa Web

Rekayasa Web yaitu model rekayasa suatu perangkat lunak yang berguna dalam pengembangan aplikasi berbasis web. Sedangkan menurut Pressman, Rekayasa web merupakan proses yang berguna untuk membuat aplikasi web dengan kualitas tinggi. Rekayasa web dalam prosesnya menekankan pada aktivitas teknis dan manajemen[10].

### 2.2.7 Rekayasa Perangkat Lunak

Rekayasa Perangkat Lunak (RPL) mempelajari pengembangan, pembuatan, pemeliharaan, sampai manajemen organisasi dan manajemen kualitas perangkat - perangkat lunak, yang merupakan mata pelajaran dan juga salah satu bidang profesi. RPL dapat disebut sebagai sebuah perubahan perangkat lunak untuk pengembangan, pemeliharaan, dan pembangunan kembali dengan prinsip rekayasa yang dapat memperoleh perangkat lunak bekerja secara lebih efektif dan efisien pada *user*. Materi yang biasa terkandung dalam rekayasa perangkat lunak ini mengenai bahasa pemrograman, desain *website* [11].

### 2.2.8 Pemrograman Berbasis Objek

Pemrograman berbasis objek adalah pendekatan pemrograman dengan menggunakan class dan objek di dalam nya. Terdapat banyak manfaat menggunakan pemrograman berbasis objek, salah satunya adalah kode yang dapat digunakan kembali (*reusability*), dapat mengganti metode yang sudah ada atau membuat metode baru tanpa membuat ulang kode (*Extensibility*), pemrograman berbasis objek memiliki konsep modularitas yang mana memudahkan apabila program mengalami eror (*Maintainability*), dapat lebih diperluas serta efisien terhadap waktu [12].

### 2.2.9 Pengembangan Perangkat Lunak

Pada tahap pengembangan perangkat lunak seorang *programmer* melakukan pengembangan sistem dengan melakukan beberapa tahap pengumpulan dan pengembangan data [13]:

#### A. Tahap Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data penelitian yang digunakan sebagai data pendukung penelitian. Tahap pengumpulan data ini terbagi menjadi beberapa metode, yaitu :

##### a) Metode Observasi

Tahap pengumpulan data ini dilakukan melalui pengamatan secara sistematis terhadap bahan penelitian, yaitu dengan mendatangi tempat atau objek penelitian secara langsung. Tahap ini direalisasikan dengan mendatangi lokasi Koperasi Annisa Muslimat NU Cilacap, yang berlokasi di Jalan Tentara Pelajar No.42 Rt/Rw 03/14, Kel. Gumilir, Kec. Cilacap Utara.

##### b) Metode Wawancara

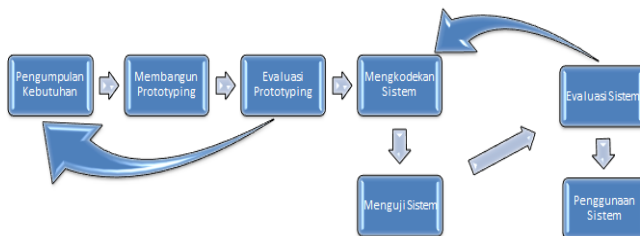
Metode wawancara dilakukan dengan mengajukan beberapa pertanyaan yang telah disiapkan untuk ditanyakan kepada narasumber sebagai responden. Responden pada penelitian ini ialah Ibu Umi Khasanah selaku manager Koperasi Annisa Muslimat NU Cilacap. Data yang dikumpulkan ditulis pada form pertanyaan maupun catatan sebagai tambahan, serta didokumentasikan melalui foto.

##### c) Metode Studi Pustaka

Metode studi pustaka merupakan metode pengumpulan data penelitian melalui buku, jurnal, *website*, maupun sumber sejenis lainnya yang dapat digunakan sebagai bahan penelitian. Sumber – sumber yang digunakan sebagai bahan pendukung penelitian ini akan diteliti sehingga sumber yang digunakan valid dan dapat dipertanggungjawabkan.

## 1. Tahap Pengembangan Sistem

Metode yang digunakan menggunakan metode *Prototype*. *Prototype* didefinisikan sebagai satu versi dari sebuah sistem potensial yang memberikan ide bagi para pengembang dan calon pengguna, bagaimana sistem akan berfungsi dalam bentuk yang telah selesai. Dasar dari pemikiran ini adalah membuat prototipe secepat mungkin, bahkan dalam waktu semalam, lalu memperoleh umpan balik dari pengguna yang akan memungkinkan *prototype* tersebut diperbaiki kembali dengan sangat cepat. Semua rancangan diagram atau model yang dibuat tidak diharuskan telah sempurna dan *final* dalam pendekatan *prototype*. Tujuan utama dari penyiapan rancangan adalah sebagai alat bantu dalam memberi gambaran sistem seperti materi dan menu yang perlu dimasukkan dalam *prototype* yang akan dikembangkan. Setelah rancangan terbentuk, dilanjutkan dengan mulai mengembangkan *prototype* [14].



Gambar 2. 1 Metode *Prototype*



Tahap – tahap pada metode ini sebagai berikut :

a) Pengumpulan Kebutuhan

Langkah pertama kali yang harus dilakukan dalam tahapan metode *prototype* adalah mengidentifikasi seluruh perangkat dan permasalahan. Tahapan metode *prototype* yang sangat penting adalah analisis dan identifikasi kebutuhan garis besar dari system. Setelah itu akan diketahui langkah apa dan permasalahan yang akan di buat dan di pecahkan. Pengumpulan kebutuhan sangat penting dalam proses ini.

b) Membangun *Prototype*

Langkah selanjutnya adalah langkah metode *prototype* membangun *prototype* yang berfokus pada penyajian pelanggan. Misalkan membuat *input* dan *output* hasil sistem. Sementara hanya *prototype* saja dulu selanjutnya akan ada tidak lanjut yang harus di kerjakan.

c) Evaluasi *Prototype*

Sebelum melangkah ke langkah selanjutnya diwajibkan memeriksa langkah pertama, karena ini adalah penentu keberhasilan dan proses yang sangat penting. Ketika langkah pertama dan kedua ada yang kurang atau salah kedepannya akan sulit sekali melanjutkan langkah selanjutnya.

d) Mengkode Sistem

Sebelum pengkodean atau biasa kita sebut proses koding, perlu kita ketahui terlebih dahulu pengkodean menggunakan bahasa pemrograman, dengan cara mengaplikasikan kebutuhan dalam bentuk kode program.

e) Menguji Sistem

Setelah pengkodean atau pengkodean tentunya akan di *testing*. Banyak sekali cara untuk *testing*, misalkan menggunakan *white box* atau *black box*. Menggunakan *white box* berarti menguji kodingan, sedangkan *black box* menguji fungsi - fungsi tampilan apakah sudah benar dengan aplikasinya atau tidak.

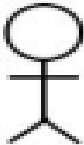

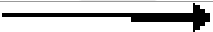




## **B. UML (Unified Modeling Language)**

UML merupakan bahasa permodelan bagi sistem yang digunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML adalah sebuah metodologi sistem yang berorientasi objek dan merupakan alat untuk pengembangan sistem [15].

### 1. Use Case Diagram

*Use case* merupakan teknik dalam pengembangan sebuah sistem informasi untuk mengetahui apa saja kebutuhan fungsional dari sistem yang bersangkutan. *Use case* menjelaskan interaksi antara *user* dengan sistem yang ada. Tujuan dari *use case* antara lain [15]:

**Tabel 2. 2** Simbol *Use Case* Diagram




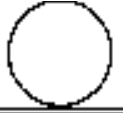
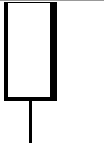


No	Gambar Simbol	Keterangan
1.		<b>Actor</b> , merupakan peran orang, sistem, menjadi alat untuk berhubungan dengan <i>use case</i> .
2.		<b>Use Case</b> , abstraksi penghubung antara aktor dengan <i>use case</i> .
3.		<b>Association</b> , penghubung antara aktor dengan <i>use case</i> .
4.		<b>Include</b> , <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> seluruhnya.
5.		<b>Generalisasi</b> , spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i> .
6.		<b>System</b> , menspesifikasikan paket yang menampilkan sistem secara terbatas.
7.		<b>Extend</b> , <i>use case</i> adalah tambahan fungsionalitas dari <i>use case</i> lainnya jika kondisi terpenuhi.

### 3. Sequence Diagram

Diagram *sequence* menggambarkan interaksi objek dan komunikasi atau hubungan antara objek – objek tersebut. *Sequence diagram* masih berkaitan dengan *use case*, yang mana sebuah *use case*

akan menjadi sebuah *sequence diagram*. Adapun tujuan dari *sequence diagram* ialah [15]:

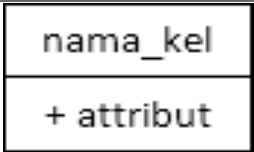




**Tabel 2. 1** Simbol *Sequence Diagram*

No	Gambar Simbol	Keterangan
1.		<b>Actor</b> , merepresentasikan entitas yang berada di luar sistem yang berinteraksi dengan sistem.
2.		<b>Boundary</b> , tepi sistem seperti <i>user interface</i> atau alat interaksi lainnya.
3.		<b>Control</b> , mengatur aliran dari informasi untuk sebuah skenario (Umumnya untuk perilaku bisnis).
4.		<b>Entitas</b> , elemen dalam menyimpan data atau informasi.
5.		<b>Activation</b> , sebuah objek mulai berpartisipasi dalam <i>sequence</i> yang menunjukkan kapan objek mengirim atau menerima.
6.		<b>Message Return</b> , menggambarkan hasil dari pengiriman <i>message</i> dan digambarkan dengan arah dari kanan ke kiri.
7.		<b>Message to Self</b> , menggambarkan pesan atau hubungan objek itu sendiri.

#### 4. Class Diagram

*Class diagram* adalah penghubung antar objek, yang juga dapat memodelkan kelas, atribut, dan operasi. *Class diagram* dapat memberikan gambaran mengenai sebuah sistem yang lebih luas dengan cara memperlihatkan kelas – kelas dan hubungannya[15].

Tabel 2. 2 Simbol *Class Diagram*









No	Gambar Simbol	Keterangan
1.		<b>Kelas</b> , kelas pada struktur sistem.
2.		<b>Interface</b> , sama dengan konsep pemrograman berorientasi objek (OOP).
3.		<b>Association</b> , relasi antar <i>class</i> disertai dengan <i>multiplicity</i> .
4.		<b>Directed Association</b> , relasi antar <i>class</i> dengan arti umum.
5.		<b>Dependency</b> , menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> yang lain.

### 5. Flowchart

*Flowchart* adalah langkah – langkah penyelesaian suatu masalah yang digambarkan dalam bagan – bagan yang memiliki arus. Adapun tujuan dari *flowchart* ialah sebagai penyelesaian masalah yang digambarkan pada suatu tahapan, lebih jelas dan rapi [15].

Tabel 2. 3 Simbol *Flowchart*

No	Gambar Simbol	Keterangan
1.		<b>Connector</b> , sambungan dari proses ke proses lainnya di halaman yang sama.

2.		<b>Input atau Output</b> , proses <i>input</i> atau <i>output</i>
3.		<b>Proses</b> , menyatakan proses yang dilakukan komputer.
4.		<b>Decision</b> , menunjukkan suatu kondisi tertentu yang menghasilkan kemungkinan jawaban ya atau tidak.
5.		<b>Flow atau Arus</b> , menyatakan jalannya arus suatu proses.
6.		<b>Terminal</b> , awal atau akhir suatu program.
7.		<b>Offline Storage</b> , menyatakan data dalam simbol akan disimpan.
8.		<b>Offline Connector</b> , sambungan dari proses ke proses lainnya di halaman berbeda.
9.		<b>Manual</b> , suatu proses yang tidak dilakukan oleh komputer.