# BAB II

**LANDASAN TEORI**



## Tinjauan Pustaka

Dalam penulisan skripsi ini, peneliti menggali informasi dari penelitian penelitian sebelumnya sabagai bahan perbandingan, baik mengenai kekurangan atau kelebihan yang sudah ada.Selain itu, peneliti juga menggali informasi dari jurnal penelitian dalam rangka mendapatkan suatu informasi yang ada sebelumnya tentang teori yang berkaitan dengan judul yang digunakan untuk memperoleh landasan teori ilmiah.

1. Jurnal Denis Firmansyah, mahasiswa Sekolah Tinggi Teknik Elektronika dan Komputer dengan judul “Sistem Informasi Inventory Tembakau Berbasis Web Pada Gudang PT. Djarum Kendal”. Penelitian ini bertujuan untuk:
2. Dengan adanya sistem informasi inventory berbasis web ini dapat membantu petugas dalam mengatasi masalah stok pada gudang PT Djarum Kendal.
3. Memudahkan petugas dalam proses pembuatan laporan pembelian, laporan pengiriman dan laporan stok secara online.
4. Dengan adanya sistem informasi Inventory berbasis web dapat membantu petugas administrasi dalam menerapkan sistem baru yang lebih efektif dan efisien untuk meningkatkan kinerja dan juga memberikan informasi yang akurat dalam pembuatan laporan-laporan yang diperlukan.
5. Memudahkan petugas Bagian Penjualan dan Bagian Pengiriman dalam dalam proses pencarian data
6. Jurnal Junaedi Abdillah, mahasiswa Manajemen Informatika PKN LPKIA Bandung tahun 2017 dengan judul “Perancangan Sistem Informasi Akuntansi Persediaan Bahan Baku Makanan Ternak Pada Bagian Gudang Di KSU Tandangsari Sumedang”. Penelitian ini bertujuan untuk:
7. Dengan adanya sistem yang telah dikomputerisasi ini maka pencatatan persediaan bahan baku di gudang tidak memerlukan waktu yang lama dan juga lebih akurat serta lengkap karena didorong dengan adanya bukti pengeluaran bahan baku bukti penerimaan bahan baku dan bukti permohonan bahan baku.
8. Tidak akan terjadi penomoran ganda dalam pencatatan barang masuk dan keluar.
9. Dengan aplikasi ini makan pengguna lebih terbantu dalam proses pembuatan laporan persediaan bahan baku dan mencetak laporan yang ingin disajikan setiap periodenya secara otomatis.
10. Jurnal Raden Adhiyaksa Indiharto,Atiqah Meutia Hilda,Arry Avorizano

Mahasiswa dan mahasiswi Teknik Informatika Fakultas Teknik Universitas Muhammadiyah Prof. DR. Hamka tahun 2016 dengan judul “Perancangan Sistem Informasi Inventory Barang Berbasis Web Pada Perusahaan Pergudangan”. Penelitian ini bertujuan untuk :

1. Membuat hak akses bagi pengguna sehingga hanya pengguna yang telah terdata dalam basis data saja yang dapat masuk ke dalam sistem.
2. Sistem ini memudahkan pengaksesannya karena berbasis web dan bisa diakses dari mana saja dan kapan saja.
3. Memudahkan penghitungan jumlah
4. barang dan aset pada saat dilakukan stock opname dari barang-barang yang telah ada di gudang, serta pendataan inventaris barang yang dimiliki oleh setiap bagian

## Tinjauan Umum

Dalam penulisan skripsi ini, tinjauan umum yang akan di bahas yaitu tentang konsep dasar aplikasi, perencanaan produksi, sistem informasi dan pengenalan *web*

### Sejarah PT. Sarana Multy Corpora

PT. Sarana Multy Corpora didirikan pada tahun 2008, semuka lahir dan bergerak dibidang percetakan untuk segmen instansi & perkantoran, perdagangan umum (importer plastic dan distributor safety equipment) serta jasa supply peralatan kantor (Work Station & ATK) pada akhirnya kini telah berkembang pesat dan menjadi perusahaan specialist dibidang Contractor, General Supplier, ME dan Interior Design.

PT. Sarana Multy Corpora adalah sebuah perusahaan yang bergerak di bidang distribusi yang mendistribusikan furniture kantor merk ARKADIA. Sebagai distributor, PT Sarana Multy Corpora konsisten untuk terus meningkatkan mutu & kemajuan perusahaan.

a. Visi Perusahaan

Menjadi perusahaan sebagai partner yang dipercaya oleh klien dan mitra usaha untuk mendapatkan solusi terbaikmdalam berbagai layanan sesuai kebutuhan pelanggan sert menjadi perusahaan terkemuka yang kompetitif dan berwawasan lingkungan.

1. Misi Perusahaan
2. Memberikan layanan dan solusi paripurna demi kepuasan pelanggan.
3. Melakukan inovasi dan terus mengembangkan usaha prospektif secara seslektif.
4. Melaksanakan proses produksi dengan memperhatikan K3 dan kelestarian lingkungan.

### Konsep Dasar Sistem *Inventory*

Inventory merupakan proses mengelola pengadaan atau persediaan barang di gudang. Inventory menjadi salah satu aktiva lancar perusahaan yang pada dasarnya merupakan suatu pengolahan barang yang meliputi penjualan, pembelian dan control stok gudang.

Inventory adalah bahan baku dan penolong, barang jadi dan barang proses produksi dana barang-barang yang tersedia, yang dimiliki dalam perjalanan dalam tempat penyimpanan kepada pihak lain pada akhir periode (Koher & Eric, 2009). Sedangkan sistem informasi inventory adalah suatu sistem *software* yang akan membantu proses inventarisasi dengan menerapkan tertib administrasi inventory yang ketat pencatatan dari barang masuk, penyimpanan, sampai dengan barang keluar. Dilengkapi dengan sistem pelaporan yang sistematis dan akurat. Sistem informasi inventory ini dibuat dengan sistem *multi user* yang memungkinkan pengaksenan sistem informasi oleh beberapa *user* yang memungkinkan pengaksenan sistem informasi oleh beberapa *user* yang berbeda dalam suatu waktu

### Sistem Informasi

Visi dan misi organisasi umumnya dinyatakan dalam bentuk sasaran dan tujuan yang akan dicapai oleh organisasi tersebut dan kemudian ditentukan strategi yang diperlukan serta tolak ukur bagi pelaksanaan dan pengukuran hasil-hasilnya. (Surendro, 2009) Hubungan antara strategi organisasi dengan *system* informasi dinyatakan sebagai berikut:

1. Peran system informasi, sebagai pendukung terwujudnya tujuan organisasi, dapat diterjemahkan sebagai misi system informasi dalam organisasi.
2. Misi sistem informasi mengarahkan pembangunan dan implementasi arsitektur organisasi melalui sekumpulan tujuan. Tujuan adalah pernyataan bagian dari misi sistem informasi (Surendro, 2009).
3. Tingkat kebersihan pencapaian tujuan pembangunan dan implelmantasi aplikasi duterjemahkan kedalam tolak ukur pencapaian tujuan atau yang dikenal sebagai indicator tujuan kunci *(Key Goal Indicator, KGI).* Indikator ini sering disebut sebagai indikator tunda (*lag indicator*) karena nilai indikatornya baru akan diketahui setelah akitivitas pencapaian tujuan tersebut selesai dilaksanakan.
4. Untuk meningkatkan kemungkinan kesuksesan implementasi aplikasi, agar sesuai dengan arah dari misi sistem informasi , diperlukan faktor kritis sukses (*Critical Success Factor, CSF*).

Misi utama system informasi yang sangat penting dalam berbagai ketertarikannya dengan faktor kritis sukses tujuan bisnis, sesuai dengan yang disarankan oleh *Spewak* (1992), sebagai berikut:

1. Menyediakan akses yang efektif terhadap data, dalam berbagai *format*, yang berguna pada waktu dan lokasi dibutuhkannya data tersebut.
2. Memiliki kemampuan untuk beradaptasi dengan perubahaan kebutuhan bisnis serta mudah dan *effisien* dalam pemeliharannya.
3. Mengelola data sehingga memiliki intergritas, kosistensi dan kesesuaian dengan standarr untuk skala seluruh organisasi.
4. Mengintegrasikan data dan aplikasi *system* informasi seluruh organisasi, sehingga baik data maupun aplikasi *system* dapat digunakan oleh seluruh pihak terkait dalam organisasi.
5. Memiliki aspek pembiayaan yang efektif, memberikan penambahan nilai dan *return on investment(ROI)* yang jelas dan terukur.

### Pengenalan *Web*

*Web* adalah salah satu aplikasi yang berisikan dokumen-dokumen *multimedia* (teks, gambar, suara, animasi, video) di dalamnya menggunakan *protocol HTTP (hypertext transfer protocol)* dan untuk mengaksesnya menggunakan perangkat lunak yang disebut *browser*. Beberapa jenis *browser* yang popular saat ini diantaranya adalah: *Internet Explorer* yang diproduksi oleh *Microsoft. Mozila Firefox, Opera*, dan *Safari* yang diproduksi oleh *Apple*(Rudyanto, 2011).

## Teori Perancangan Basis Data

Basis data (*database*) merupakan suatu kumpulan data yang disusun dalam bentuk tabel-tabel yang saling berkaitan maupun berdiri sendiri dan disimpan secara bersama-sama pada suatu media. Basis data dapat digunakan oleh satu atau lebih program aplikasi secara optimal, data disimpan tanpa mengalami ketergantungan pada program yang akan menggunakannya.

### Entity Relationship Diagram ( ERD )

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol.

### Transformasi ERD ke Logical Record Structure (LRS)

*Transformasi*ERD ke LRS yaitu proses mengubah data dalam bentuk model data menjadi tabel-tabel basis data.Dalam merancang sebuah sistem dari ERD ke LRS di perlukan transformasi terlebih dahulu. Adapun aturan-aturan untuk transformasi tersebut yaitu :

1. Setiap himpunan *entitas* ditransformasikan sebagai sebuah tabel
2. Relasi dengan derajat satu ke satu yang menghubungkan 2 himpunan *entitas* akan ditransformasikan kedalam bentuk penyertaan *atribut-atribut relasi* ke salah satu himpunan *entitas*
3. Relasi dengan derajat satu ke banyak yang menghubungkan 2 himpunan entitas akan ditransformasikan kedalam bentuk penyertaan *atribut-atribut relasi* ke himpunan *entitas* yang derajat relasinya banyak (*many*)
4. Relasi dengan derajat banyak ke banyak yang menghubungkan dua himpunan entitas , maka atribut-atribut relasi akan ditransformasikan menjadi sebuah tabel
5. Implementasi *himpunan entitas* lemah
6. *ImplementasiSpesialisasi*
7. *ImplementasiGeneralisasi*
8. *ImplementasiGeneralisasi*

### Logical Record Structure (LRS)

*Logical Record Structure* (LRS)yaitu representasi dari struktur *record-record* pada tebel-tabel yang terbentuk dari hasil antar himpunan entitas.menentukan kardinalitas, jumlah table dan *foreign key* (FK) sebagai berikut :

1. *One-to-one* yaitu satu *entitas* berhubungan dengan paling banyak satu entitas lain.
2. *One-to-many*yaitusatu *entitas* dapat berhubungan dengan lebih dari satu entitas lain.
3. *Many-to-many*yaitubeberapa entitas dapat berhubungan dengan beberapa entitas lain.

### *Spesifikasi* Basis Data

*Spesifikasi* basis data yaitu perancangangan yang dilakukan untuk dapat menampung data, masukan dan membantu pengolahan data.Perancangna ini digunakan untuk mempermudah dalam perancangan sistem.

## Definisi Unified Modeling Language(UML)

*UML* adalah bahasa standar untuk penulisan cetak biru perangkat lunak (Roger S. Pressman, 2012).*UML* dapat digunakan untuk memvisualisasikan, menetukan, mengonstruksi, dan mendokumentasikan *artifak-artifak* suatu sistem *software-intensive.*

### Class Diagram

*Class* merupakan sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah obyek dan merupakan inti dari pengembangan dan desain berorientasi obyek.*Class* menggambarkan keadaan (atribut / properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode / fungsi). Atribut merupakan sebuah nilai data yang dimiliki oleh obyek sebuah kelas. Nama, umur, berat badan, adalah atribut dari obyek manusia atau orang. *Metode*merupakan implementasi dari sebuah operasi ke dalam sebuah kelas. Sebagai contoh, sebuah kelas “file” mungkin mempunyai operasi *print*.

### Use Case Diagram

*Use Case* atau *diagram use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat (Shalahuddin & Rosa, 2011). *Use case* mendekripsikan 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 didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Syarat penamaan pada use case adalah nama didefinisikan sesimple mungkin dan dapat dipahami. Berikut ini adalah komponen-komponen Use case diagram.

1. *Aktor*

*Aktor* merupakan 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.

1. *Use Case*

*Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

1. *Relationship*

*Association* antara *actor* dan *use case* digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi dan bukannya mengidentifikasikan aliran data.

1. *Include*

*Include*, digunakan ketika dalam penulisan yang berbeda-beda terdapat deskripsi-deskripsi yang sama, maka relasi ini dapat digunakan untuk menghindari penulisan deskripsi yang berulang-ulang. Seperti digambarkan pada halaman selanjutnya.

1. *Extend*

*Extend,* merupakan perluasan dari *use case* lain jika kondisi atau syarat terpenuhi. Extends ini agak mirip dengan *include.* Namun pada *extend*, tidak harus terjadi apa yang diharapkan.

### Activity Diagram

*Activity Diagram* digunakan untuk memodelkan aspek dinamis dari sistem yang dalam kebanyakan hal terjadi langkah-langkah berurutan, juga memodelkan aliran-aliran dari objek dalam pergerakannya dari suatu state ke state lainnya dalam suatu aliran kendali.

1. *Start Point*

*Start Point (initial node)*, menggambarkan permulaan dari sebuah sistem yang akan dikerjakan, biasanya diletakkan pada pojok kiri atas.

1. *End Point*

*End Point (activity fial node),*menggambarkan akhir dari sebuah sistem.

1. *Activities*

*Activities*, menggambarkan proses bisnisdan dikenal sebagai *activity state*.

1. *Black Hole Activities*

*Black Hole Activities* menggambarkan ada masukan dan keluaran.

1. *Miracle Activities*

*Miracle Activities,* tidak ada masukan dan ada keluaran dan dipakaipada waktu *start point*.

1. *Parallel Activities*

*Parallel Activities, activity* yang berjalan secara bersamaan terdiri dari:

1. *Fork* (percabangan), mempunyai 1 transisi masuk dan 2 atau lebih transisi keluar.
2. *Join* (penggabungan), mempunyai 2 atau lebih transisi masuk dan hanya 1 transisi keluar.
3. *Decision Points*

*Decision Points*, tidak ada keterangan (pertanyaan) pada tengah belah ketupat seperti pada *flowchart* dan harus mempunyai *Guards*(kunci).

1. *Swimlane*

*Swimlane*, sebuah cara untuk mengelempokan *activity*berdasarkan *actor*.

### Sequence Diagram

*DiagramSequence* menggambarkan kelakuan objek pada *use case* dengan mendekripsikan waktu hidup objek dan *message* yang dikirmkan dan diterima antar objek. Oleh karena itu untuk menggambarkan digaram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Banyaknya diagram *sequence* yang harus digambarkan adalah sebanyak pendefinisian *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram *sequence* sehingga semakin *use case* yang didefinisikan maka diagram *sequence* yang harus dibuat juga semakin banyak. Berikut komponen-komponen dalam *sequence* diagram.

1. Aktor

Aktor Menggambarkan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem.

1. *Boundary*

Menggambarkan interaksi antara satu atau lebih *actor* dengan sistem, memodelkan bagian dari sistem yang bergantung pada pihak lain disekitarnya dan merupakan pembatas sistem dengan dunia luar.

1. *Control*

Menggambarkan “perilaku mengatur”, mengkoordinasikan perilaku sistem dan dinamika dari suatu sistem, menangani tugas utama dan mengontrol alur kerja suatu sistem.

1. *Entity*

Menggambarkan informasi yang harus disimpan oleh sistem (struktur data dari sebuah sistem).

1. *Message*

*Message* digambarkan dengan garis berpanah, yangmenunjukkan arah *message.*

## Aplikasi Pendukung

Adapun aplikasi pendukung dalam penulisan skripsi ini mnerangkan tentang *PHP*, *MySQL, Report Pdf*dan*Xampp* :

### PHP

*PHP (PHP;Hypertext Preprocesor)* adalah sebuah bahasa pemrograman yang berbentuk *Scripting*(Bunafit, 2008), sistem kerja dari program ini adalah sebagai *interpreter* bukan sebagai *compailer*.

### MySQL

*MySQL* merupakan *database server* yang awalnya hanya berjalan pada system *Unix* dan *Linux*. Seiring berjalannya waktu dan banyaknya peminat yang menggunakan database ini, *MySQL* merilis *versi* yang dapat diinstal pada hampir semua *platform*, termasuk *windows*.*Lisensi* dari *MySQL* adalah *freemware*.

*SQL* merupkan kependekan dari “*Structured Query La-nguage*”.*SQL* merupakan suatu bahasa permintaan yang tersetruktur, karena *SQL* memiliki beberapa aturan yang telah distandarkan oleh asosiasi yang bernama *ANSI.SQL* adalah bahasa permintaan yang melekat pada suatu database atau *SMBD* tertentu (Nugroho, 2007), sedangkan *MYSQL* merupakan database servernya. Dengan kata lain, *MySQL* merupakan *SMBD*-nya dan *SQL* adalah perintah atau bahasa yang melekat di dalam *SMSBD* tersebut.

### Report PDF

Dengan adanya *dokument PDF* yang universal dan terintergrasi di hampir semua sistem operasi, program berbasis web dapat menghasilkan laporan yang tidak kalah dengan program desktop.Dokumen *PDF* dianggap cukup *universal* karena telah tersedia aplikasi pembacanya (*PDF reader*) diberbagai sistem operasi, baik *Windows* maupun *Linux*. Pada sistem operasi *Windows*, biasanya terintegrasi *software Acrobat Reader* yang geratis dan dapat di *download* di *internet*. Dengan menggunakan *browser webinternet Explorer*, laporan *PDF* yang dihasilkan oleh pemrograman *web*akan dapat dibaca secara langsung melalui *browser* dan dicetak pada saat itu. Jika dikhendaki, dokumen tersebut juga dapat disimpan ke dalam *harddisk* untuk di cetak.

### Xampp

Menurut Yogi wicaksono (2008:7) *XAMPP* adalah sebuah *software* yang berfungsi untuk menjalankan*website* berbasis PHP dan menggunakan pengolah data *MySQL* dikomputer *local*“. *XAMPP* berperan sebagai *server web* pada komputer.XAMPP juga dapat disebut sebuah *CPanel server virtual*, yang dapat membantu melakukan *preview* sehingga dapat memodifikasi *website* tanpa harus *online* atau terakses dengan internet.

## Pengujian Sistem

Pengujian *software* adalah proses verifikasi dan validasi apakah sebuah aplikasi *software* atau program memenuhi persyaratan bisnis dan persyaratan teknis yang mengarahkan desain dan pengembangan dan cara kerjanya seperti yang diharapkan dan juga mengidentifikasi kesalahan yang penting yang digolongkan berdasarkan tingkat *severity* pada aplikasi yang harus diperbaiki (Bunafit, 2008).

### Tujuan Pengujian

Adapun tujuan dari pengujian sistem yaitu:

1. Untuk melakukan *verifikasi, validasi* , dan *deteksi eror*
2. Untuk menemukan sebuah masalah /*eror* kemudian dibenahi.
3. Mencari *eror* dan kelemahan/keterbatasan dari sistem.
4. Mencari sejauh apa kemampuan dari sistem.
5. Menyediakan informasi untuk kualitas dari *product software*.

Dari tujuan pengujian sistem yang sudah dijabarkan tersebut terdapat misi dari tim pengujian sistem (*tester*), yaitu :

1. Sebagai sarana untuk membantu meminimalkan resiko kegagalan proyek.
2. *Tester* hanya menginformasikan, akan tetapi tidak komplain.
3. *Tester* sering digabungkan dengan *user* (RAD)
4. Seorang *tester* harus paham tentang *software quality* supaya dapat mengerti tentang *product* yang ia uji.
5. *Skill* komunikasi juga diperlukan untuk seorang *tester* untuk dapat menginformasikan kepada *debugger* dengan baik, biasanya alat komunikasinya menggunakan *template*.

### Prinsip Pengujian

Sebelum menetapkan *metode* pengujian, seorang ahli pada bidang *software* harus mengerti betul atau memahami prinsip dasar yang menuntun pengujian perangkat lunak. Prinsip-prinsip pengujian secara umum yang banyak dianut oleh para ahli perangkat lunak, antara lain  :

1. Seorang *Programmer* seharusnya tidak menguji programnya sendiri.
2. Sebaiknya satu pengujian tidak hanya mengerjakan program yang dianggap benar, tetapi tidak mengerjakan yang dianggap salah.
3. Tujuan dari pengujian adalah untuk menemukan kesalahan, bukan untuk menunjukkan bahwa program tersebut salah.
4. Tidak ada sejumlah pengujian yang dapat menjamin bahwa program bebas dari kesalahan.
5. Bagian-bagian dari program di mana terdapat banyak kesalahan yang telah ditemukan adalah suatu tempat yang baik untuk menemukan kesalahan yang lebih banyak.
6. Tujuannya adalah bukan untuk mempermalukan *programmer*.

### Pengujian *Black Box*

Black box adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Jadi dianalogikan seperti kita melihat suatu kotak hitam, yang hanya bisa melihat penampilan luarnya saja, tanpa tau ada apa dibalik bungkus hitam nya (Bunafit, 2008).

### Pengujian *White Box*

White box testing adalah cara pengujian untuk meneliti kode-kode program yang ada, dan menganalisis apakah ada kesalahan atau tidak. Jika ada modul yang menghasilkan output yang tidak sesuai dengan proses yang dilakukan, maka baris-baris program, variabel dan parameter pada unit tersebut akan dicek satu persatu dan diperbaiki kemudian di compile ulang(Bunafit, 2008).