

## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

Pada bab ini dibahas tentang identifikasi permasalahan, analisis permasalahan, solusi permasalahan, dan perancangan sistem dalam Sistem Informasi Penjualan pada Toko Sepatu Ivory. Tahapan awal adalah pengumpulan data dengan teknik wawancara dan observasi. Tahapan selanjutnya adalah melakukan identifikasi permasalahan dan analisis permasalahan.

#### **3.1 Analisis Sistem**

Dalam melakukan analisis sistem, ada beberapa hal yang dilakukan seperti mencari informasi mengenai Toko Ivory, mengidentifikasi permasalahan, dan menentukan hasil dari analisis. Berikut penjelasan lebih detailnya:

##### **3.1.1 Komunikasi**

Tahap komunikasi dilakukan wawancara dan observasi langsung ke Toko Sepatu Ivory. Tahap ini dilakukan dengan cara melakukan pengamatan terhadap aktifitas penjualan yang ada pada Toko Sepatu Ivory. Wawancara dilakukan dengan melakukan Tanya jawab dengan pihak manajemen Toko Sepatu Ivory untuk memvalidasi hasil observasi dengan manajemen. Setelah tahap observasi dan wawancara selesai maka dapat disusun taha analisis bisnis, analisis kebutuhan pengguna, analisis kebutuhan data, dan analisis kebutuhan fungsional.

## A Analisis Bisnis

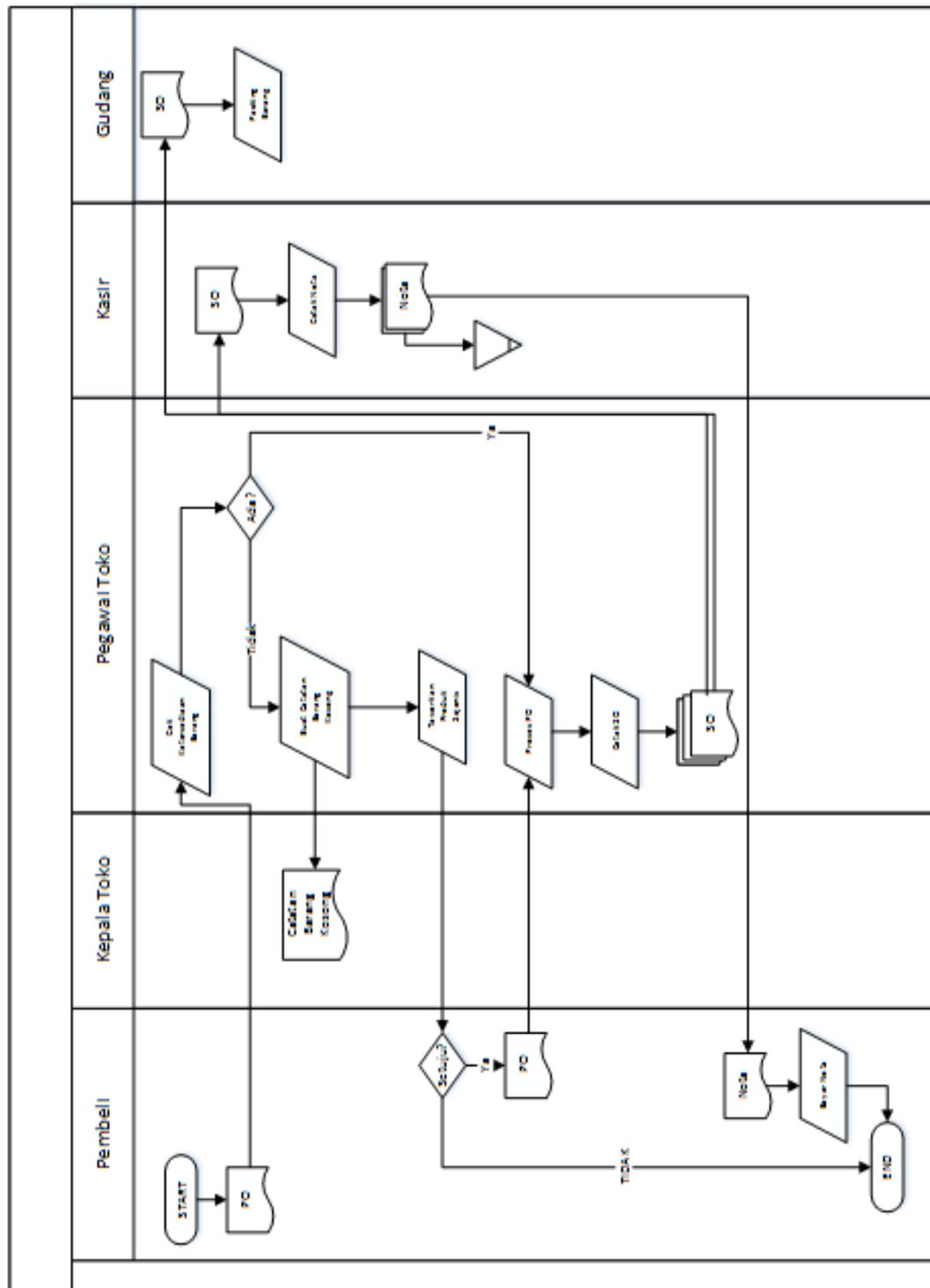
Setelah dilakukan tahap komunikasi, tahapan selanjutnya adalah melakukan analisis bisnis yang meliputi identifikasi masalah, identifikasi pengguna, identifikasi data, dan identifikasi fungsi.

### 1. Identifikasi Masalah

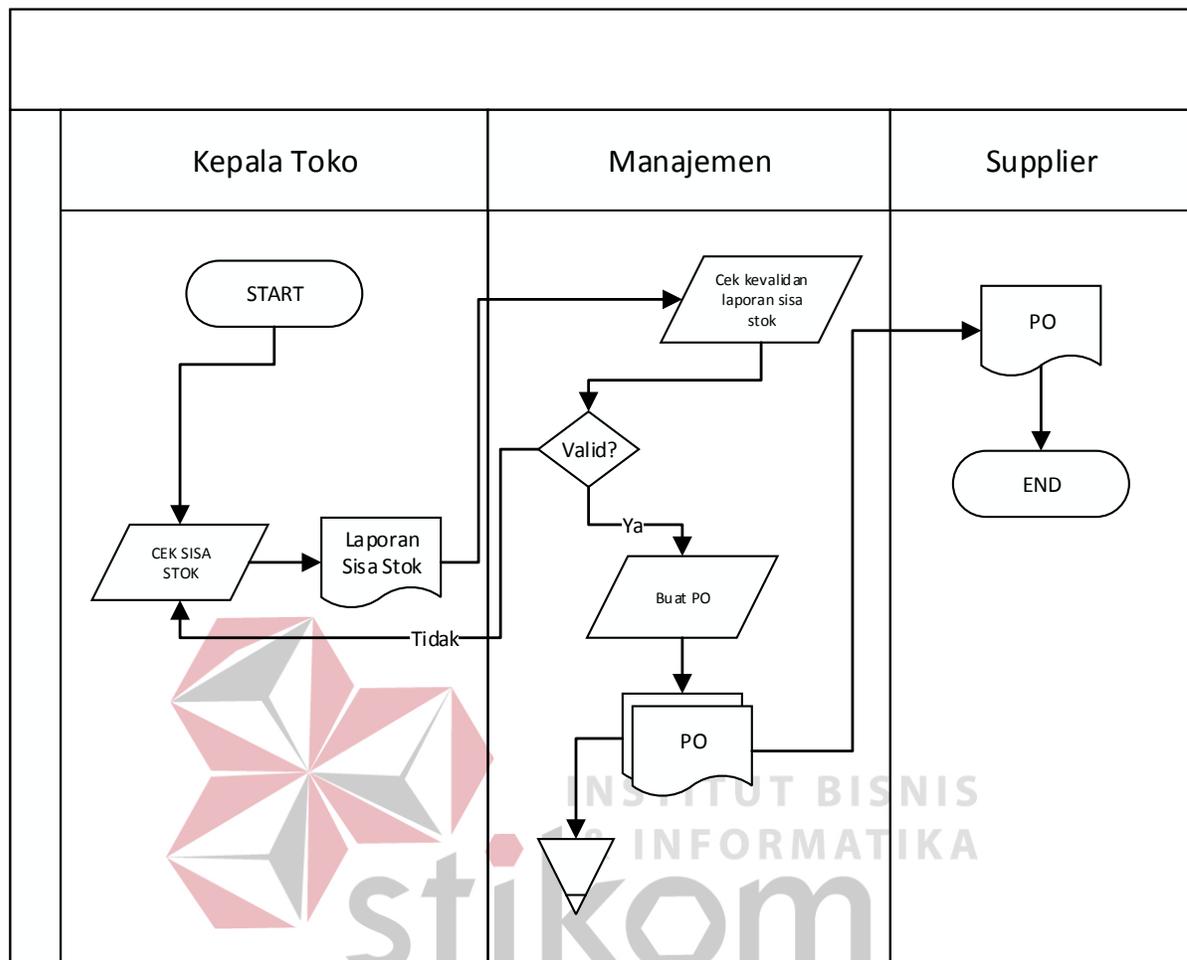
Seperti yang terlihat pada gambar 3.1 terlihat bagaimana Toko Ivory dalam menjalankan proses bisnisnya. Toko Ivory membutuhkan suatu sistem agar mampu bersaing dengan toko *retail* sejenis. Sistem yang dibutuhkan adalah sistem yang dapat memberikan informasi mengenai aktifitas penjualan produk, sehingga permasalahan pengawasan yang selama ini timbul dapat terselesaikan. Sistem yang ada belum mampu memberikan informasi mengenai aktifitas penjualan, sehingga sistem yang ada belum dapat melakukan tindakan terhadap hal-hal atau sesuatu yang perlu direpson secara cepat. Berikut adalah contoh kondisi kegiatan penjualan produk yang sering terjadi pada Toko Sepatu Ivory:

1. Kesulitan dalam pengawasan terhadap kegiatan penjualan masing-masing gerai.
2. Tidak dapat melihat aktifitas penjualan masing-masing gerai secara *realtime*.
3. Harus melakukan pengawasan aktifitas penjualan dengan mendatangi satu persatu gerai.

Hal-hal tersebut dapat membuat Toko Ivory kehilangan potensi pemasukan dan sering kali tidak dapat memenuhi target penjualan yang telah ditentukan.



Gambar 3.1 *Workflow* Aktifitas Pejualan Toko Sepatu Ivory



Gambar 3.2 *Workflow* Aktifitas Pembelian Toko Sepatu Ivory

Seperti yang di gambarkan pada gambar 3.2 diatas, aktifitas pembelian pada Toko Sepatu Ivory dimulai dengan Kepala Toko melakukan cek sisa stok tiap hari. Setelah melakukan cek sisa stok toko, maka kepala toko akan membuat laporan sisa stok yang nanti akan dilaporkan kepada pihak manajemen. Pihak manajemen toko akan mendatangi satu persatu gerai untuk melakukan cek validasi laporan sisa stok. Ketika laporan tersebut valid maka manajemen akan membuat *Purchasing Order* untuk dilakukan pemesanan kepada *supplier*.

Dengan mendatangi satu persatu gerai, pihak manajemen tentunya akan kesulitan untuk melakukan pengawas terhadap ketiga gerai sekaligus, sehingga pasti akan ada gerai yang mengalami stok kosong sebelum di lakukan cek kevalidan data sisa stok. Kekosongan stok ini sangat beresiko terhadap tingkat kepercayaan pelanggan terhadap stok yang ada kepada gerai toko ivory. Hasil identifikasi dari aktifitas penjualan pada Toko Sepatu Ivory ini dapat di lihat pada tabel 3.1 dibawah ini.

Tabel 3.1 Hasil Identifikasi Masalah

<b>Identifikasi Permasalahan</b>	<b>Penyebab</b>	<b>Alternatif Solusi</b>
Pihak manajemen tidak dapat melakukan pengawasan aktifitas penjualan ketiga gerai secara langsung.	Tidak adanya sistem yang menghubungkan ketiga gerai toko.	Sistem yang dapat menghubungkan ketiga toko.
Untuk melakukan aktifitas pengawasan pihak manajemen harus mendatangi satu persatu toko.	Toko tidak mempunyai sistem pengawasan dimana manajemen dapat melakukan aktifitas pengawasan dari manapun.	Sistem yang dapat membantu manajemen untuk melakukan aktifitas pengawasan secara online.

## 2. Identifikasi Pengguna

Berdasarkan hasil wawancara terdapat beberapa pengguna yang akan terlibat dalam sistem informasi penjualan ini yaitu bagian kasir, kepala toko, dan pihak manajemen.

## 3. Identifikasi Data

Setelah dilakukan proses identifikasi permasalahan dan pengguna, maka dapat dilakukan identifikasi data. Pada sistem informasi ini memerlukan data

member, data aktifitas penjualan, data aktifitas pembelian, data aktifitas penerimaan barang, dan data kebijakan-kebijakan.

#### 4. Identifikasi Fungsi

Setelah dilakukan proses identifikasi permasalahan, pengguna, dan data, maka dapat diidentifikasi fungsi dari sistem informasi yang akan dibangun sebagai berikut: penjualan, pembelian, penerimaan barang, dan laporan pendapatan

### B Analisis Kebutuhan Pengguna

Berdasarkan hasil wawancara, maka dapat dibuat kebutuhan pengguna yang akan dianalisis untuk mengetahui kebutuhan dari masing-masing pengguna yang berhubungan langsung dengan aplikasi yang dibangun dapat sesuai dengan apa yang dibutuhkan. Peran dan tanggung jawab pengguna dapat dilihat pada tabel 3.2.

Tabel 3.2 Peran dan Tanggung Jawab

Aktor	Peran	Tanggung Jawab
Kasir	Mengelola transaksi penjualan Mengelola data member	Mencatat data penjualan Mencatat data pembayaran Membuat laporan penjualan Membuat laporan pendapatan
Kepala Toko	Mengelola pembelian Mengelola data penerimaan barang	Mencatat data pembelian Mencatat data penerimaan barang Membuat Laporan Sisa Stok Membuat Laporan Penerimaan Barang
Pihak Manajemen	Melakukan pengawasan aktifitas penjualan Melakukan pengawasan aktifitas penerimaan barang	Membuat Sales Order

Dalam membangun sebuah sistem informasi dibutuhkan sebuah perancangan perangkat lunak yang sesuai dengan kebutuhan pengguna. Hal ini bertujuan untuk memudahkan pengguna dalam mengaplikasikan sistem tersebut sesuai dengan fungsi yang telah ditentukan. Fungsi-fungsi tersebut dikelompokkan berdasarkan entitas dan dapat dilihat pada table 3.3 dibawah ini.

Tabel 3.3 Fungsi-Fungsi Entitas

Entitas	Deskripsi Fungsi yang Diperlukan
Kasir	<ul style="list-style-type: none"> <li>a. Melakukan transaksi penjualan.</li> <li>b. Melakukan pencatatan data pembayaran dan detail data pembayaran.</li> <li>c. Mencetak laporan penjualan.</li> <li>d. Mencetak laporan pendapatan.</li> </ul>
Kepala Toko	<ul style="list-style-type: none"> <li>a. Melakukan pencatatan data sisa stok barang.</li> <li>b. Melakukan pencatatan data penerimaan barang.</li> <li>c. Mencetak laporan penerimaan barang.</li> </ul>
Manajemen	<ul style="list-style-type: none"> <li>a. Melakukan pengawasan terhadap aktifitas penjualan.</li> <li>b. Melakukan pembelian barang.</li> </ul>

### 3.1.2 Perencanaan Kebutuhan

Sistem informasi penjualan ini akan membantu pihak manajemen dalam melakukan pengawasan aktifitas penjualan yang terjadi pada toko ivory. Pada gambar 3.3 terdapat bagan *input*, proses, *output* (IPO) yang menggambarkan garis besar *input*, proses, *output* dari sistem yang di rancang. Penjelasan dari bagan IPO dibawah adalah sebagai berikut:

#### 1. *Input*

Pada bagian ini terdapat apa saja yang menjadi masukan dari sistem.

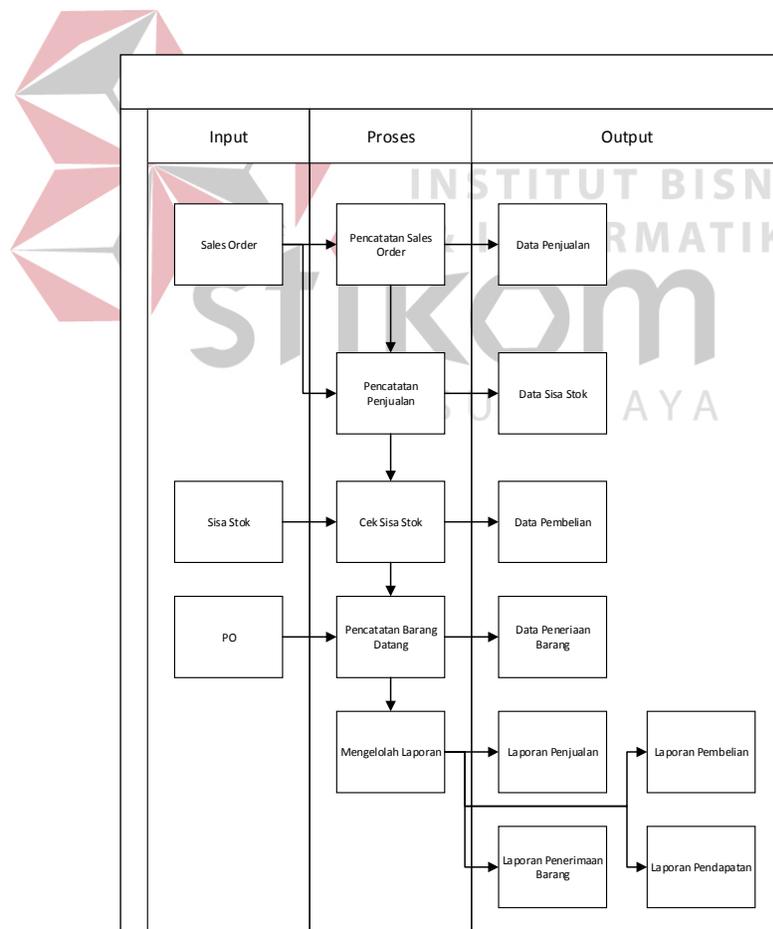
Terdapat tiga buah masukan yaitu, *sales order*, sisa stok, dan PO.

## 2. Proses

Pada kategori ini berisi tentang proses utama dari sistem yang dibuat. Ada lima buah proses yaitu, pencatatan *sales order*, pencatatan penjualan, cek sisa stok, pencatatan barang datang, mengelolah laporan.

## 3. Output

Pada kategori ini berisi tentang *output* dari sistem informasi ini. Ada delapan *output* yaitu, data penjualan, data sisa stok, data pembelian, data penerimaan barang, laporan penjualan, laporan pembelian, laporan penerimaan barang, laporan pendapatan.



Gambar 3.3 *Input, Proses, Output* Sistem Informasi Penjualan pada Toko Ivory

### 3.2 Perancangan Sistem

Dalam tahap perancangan sistem terdapat arsitektur sistem, merancang *context diagram*, *data flow diagram* (DFD), *entity relationship diagram* (ERD), dan tampilan perancangan antarmuka dari sistem.

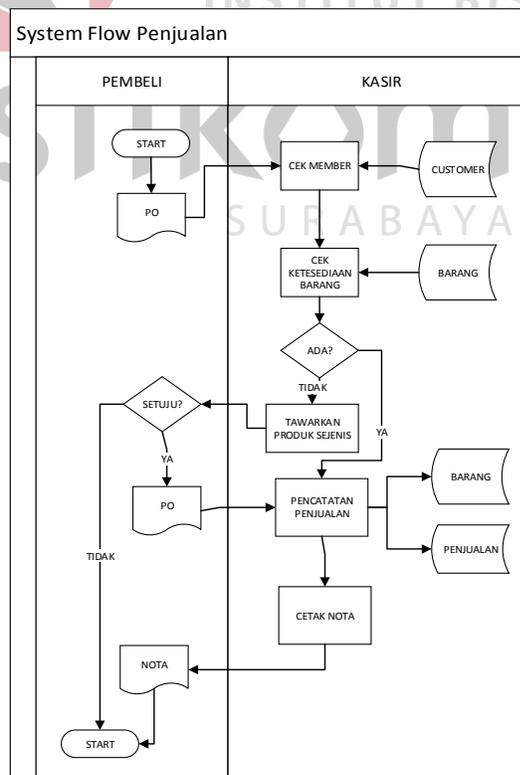
#### 3.2.1 Perancangan Proses

Untuk menggambarkan alur jalannya proses dari sistem yang dibuat maka dibutuhkan *system flow*, *context diagram*, dan DFD sebagai alat bantu.

##### A *System Flow*

*System flow* menggambarkan alur kerja dari sistem aplikasi yang dibuat dan menjelaskan urutan prosedur yang ada pada sistem.

##### 1. *System flow* penjualan

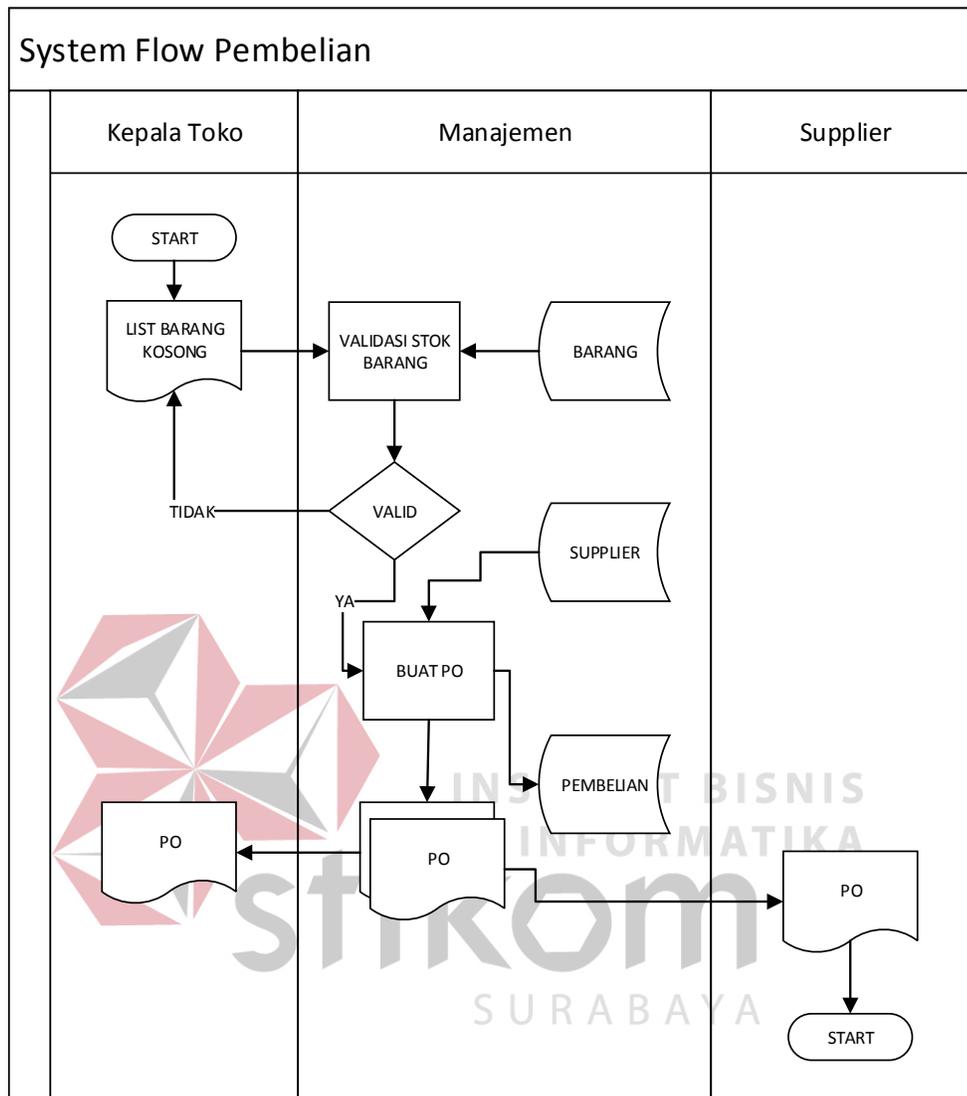


Gambar 3.4 *System Flow* Penjualan

Gambar 3.4 diatas adalah *system flow* Penjualan. Pada aktifitas tersebut terdapat dua actor, yaitu kasir dan pembeli. Aktifitas bermula ketika pembeli memberikan *purchasing order* kepada kasir. Kasir kemudian akan melakukan cek ketersediaan barang, bila barang tersebut tidak ada maka kasir akan memberikan dua saran, yaitu menawarkan barang yang sejenis atau menyarankan untuk mengambil pada gerai lain yang memiliki sepatu tersebut. Bila barang yang diminta pembeli ada maka kasir akan memproses *purchasing order* tersebut dan secara otomatis sistem akan langsung melakukan penyimpanan kedalam table penjualan dan jg asistem akan langsung melakukan *update* data pada tabel barang. Keluaran dari proses ini adalah nota ganda, satu akan diberikan ke pembeli sedangkan satunya akan di simpan untuk rekap data.

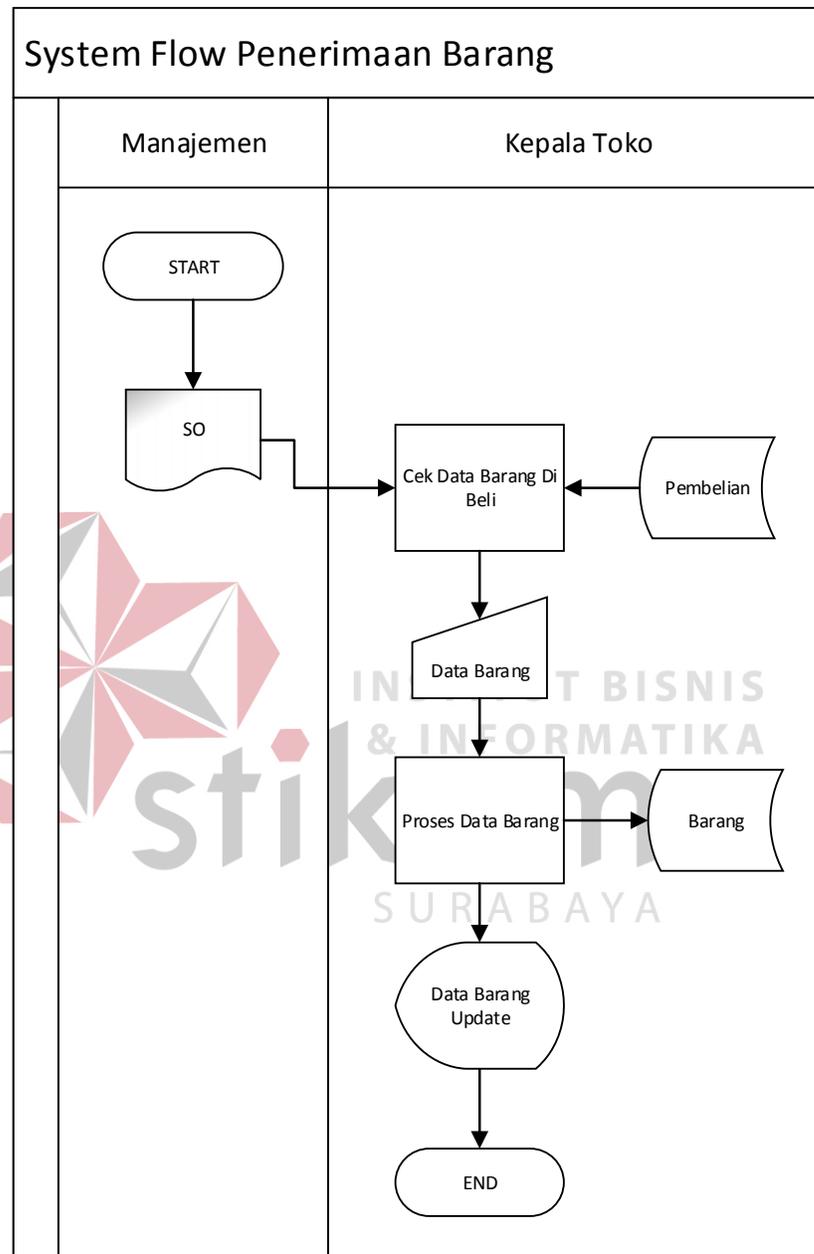
## 2. *System flow* Pembelian

Gambar 3.5 dibawah ini adalah *system flow* dari aktifitas pembelian. Pada aktifitas pembelian terdapat tiga aktor yang akan menjalankan, yaitu Manajemen, *Supplier*, dan Kepala Toko. Aktifitas dimulai dengan kepala toko memberikan daftar barang kosong yang ada pada tokonya. Manajemen akan melakukan pengecekan daftar barang kosong yang diberikan oleh pihak kepala toko. Manajemen akan membuat *sales order* dan akan diberikan kepada pihak *supplier*. Aktifitas pembelian ini akan disimpan ke tabel pembelian. *Output* dari aktifitas ini adalah *sales order* yang akan diberikan ke pihak *supplier*.



Gambar 3.5 *System Flow Pembelian*

*System flow* Penerimaan Barang



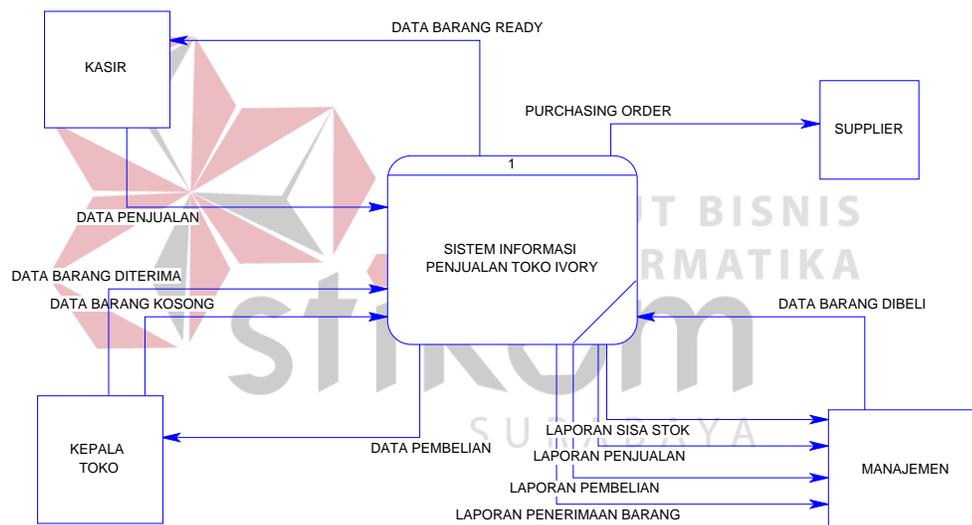
Gambar 3.6 *System Flow* Penerimaan Barang

Gambar 3.6 adalah *system flow* aktifitas penerimaan barang. Pada aktifitas penerimaan barang aktornya adalah manajemen dan kepala toko. Alur

dimulai dengan kepala toko melakukan pengecekan barang yang datang dengan barang yang di beli oleh pihak manajemen. Kemudian kepala toko akan melakukan *update* jumlah stok barang yang telah diterima. Aktifitas ini akan di simpan pada tabel barang.

## B Context Diagram

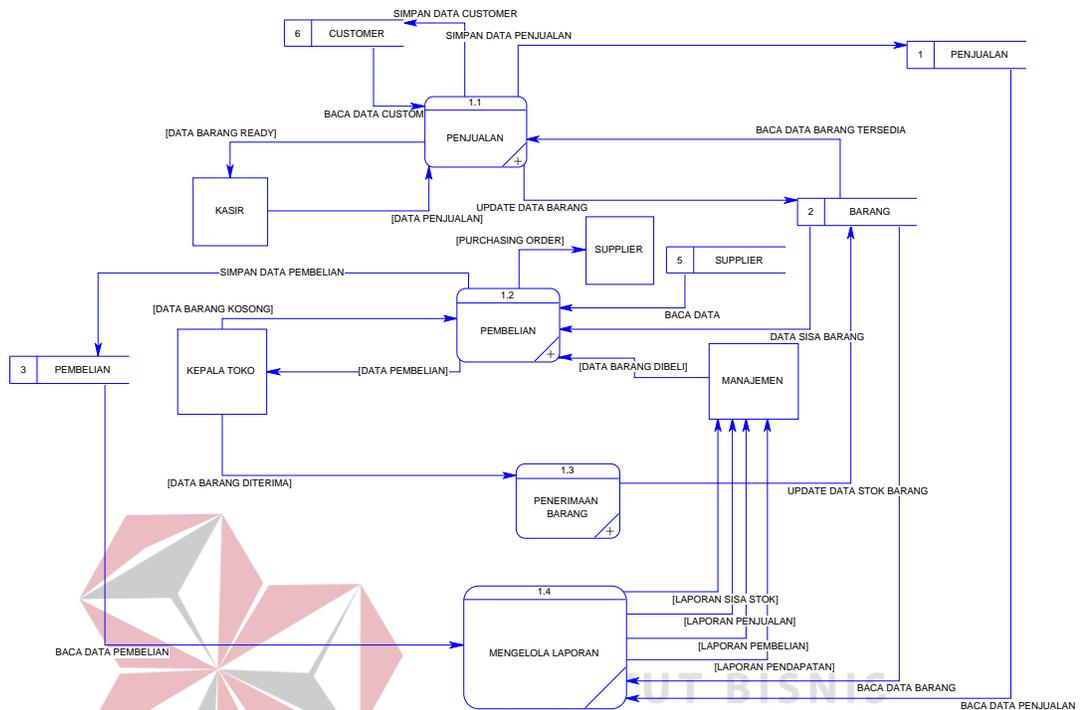
Tahapan selanjutnya setelah menggambarkan alur kerja dengan *system flow* adalah membuat *context diagram*.



Gambar 3.7 *Context Diagram* Sistem Informasi Penjualan pada Toko Ivory

*Context diagram* menggambarkan entitas yang berhubungan langsung dengan sistem dan aliran data secara umum seperti pada gambar 3.7. Terdapat empat entitas dalam aplikasi ini yaitu kepala toko, kasir, supplier, dan manajemen.

## 1. DFD level 0

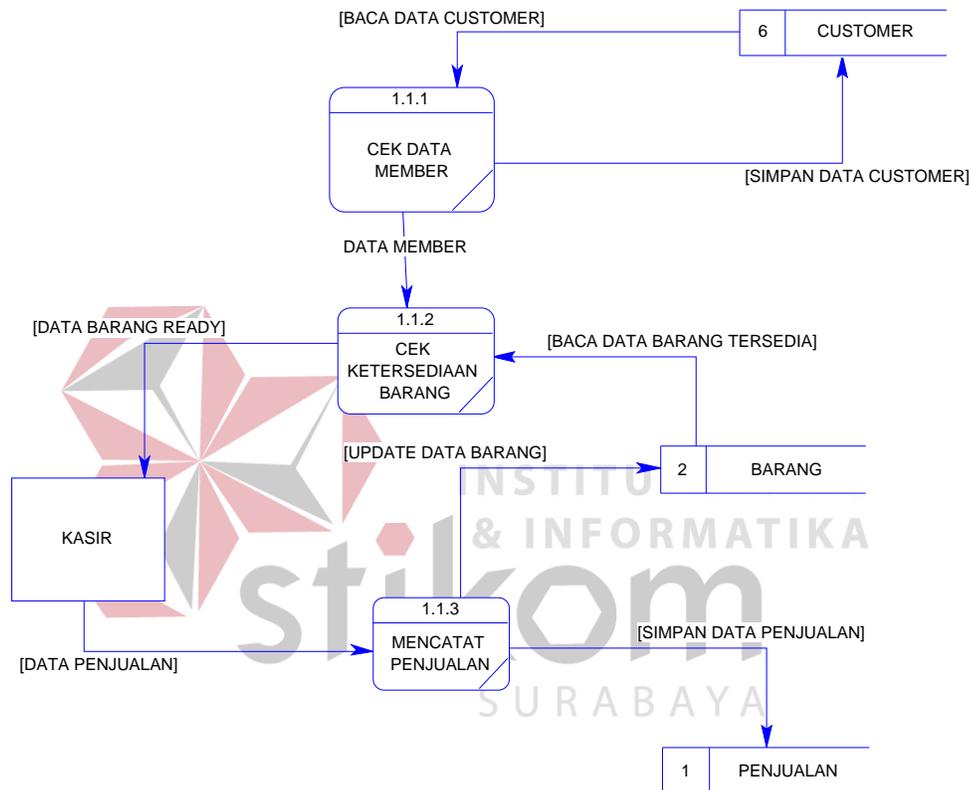


Gambar 3.8 DFD Level 0 Sistem Informasi Penjualan Toko Ivory

Pada gambar 3.8 pada merupakan DFD level 0 dari hasil *decompose context diagram* sehingga memperlihatkan proses-proses utama yang ada dari sistem informasi yang dibuat. Ada 4 proses utama, yaitu penjualan, pembelian, penerimaan barang, dan mengelola laporan. Ada 4 tabel pada DFD level 0 ini, yaitu penjualan, pembelian, barang, pembelian, dan penerimaan barang.

## 2. DFD Level 1 Penjualan

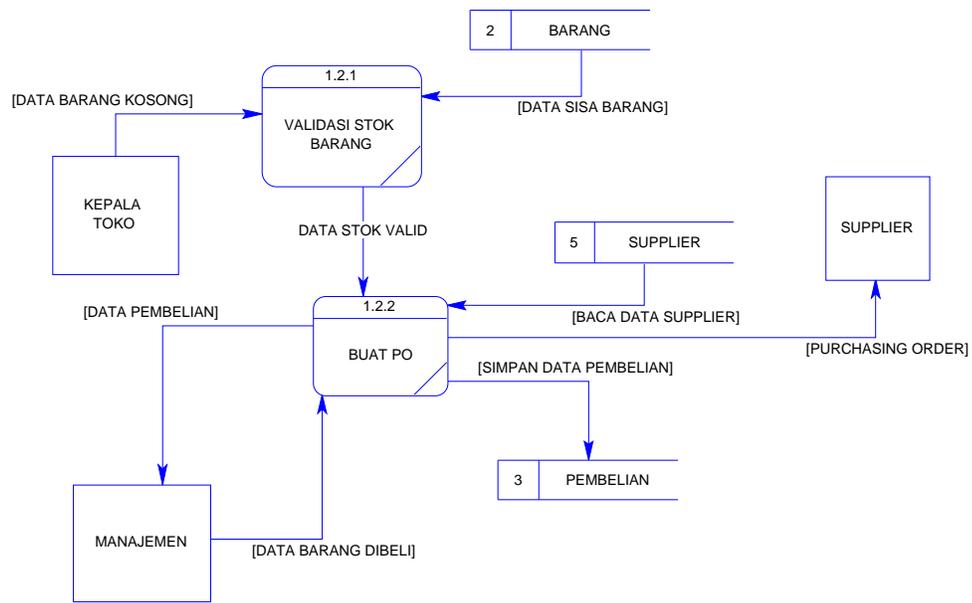
Gambar 3.9 dibawah ini merupakan hasil *decompose* dari proses penjualan. Terdapat tiga proses, yaitu cek data member, cek ketersediaan barang dan mencatat penjualan.



Gambar 3.9 DFD Level 1 Penjualan

## 3. DFD Level 1 Pembelian

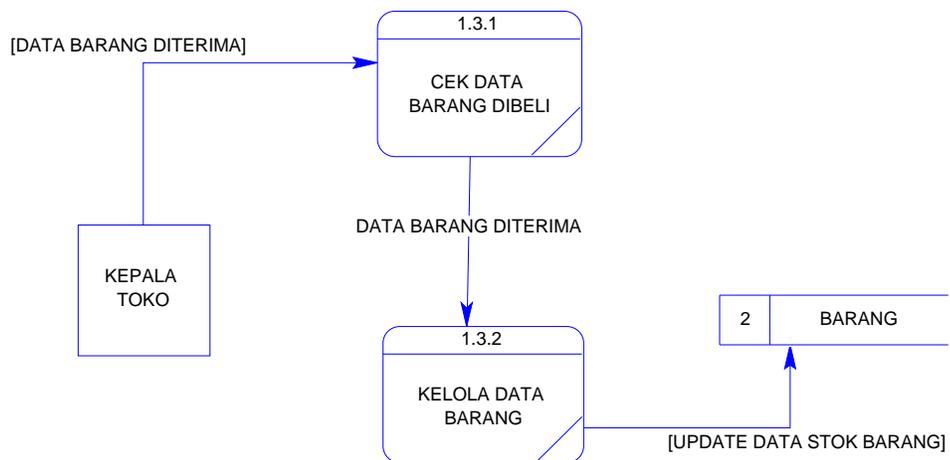
Proses akan diawali dengan melakukan validasi terhadap stok barang yang akan dilakukan oleh kepala toko. Setelah itu manajemen akan membuat PO berdasarkan barang kosong yang kemudian PO tersebut diserahkan kepada supplier seperti yang tergambar pada gambar 3.10



Gambar 3.10 DFD Level 1 Pembelian

#### 4. DFD Level 1 Penerimaan barang

Gambar 3.11 dibawah ini adalah DFD level 1 dari proses penerimaan barang. Pada DFD ini terdapat dua buah proses, yaitu proses cek data barang dibeli dan kelola data barang.



Gambar 3.11 DFD Level 1 Penerimaan Barang

### 3.2.2 Perancangan Basis Data

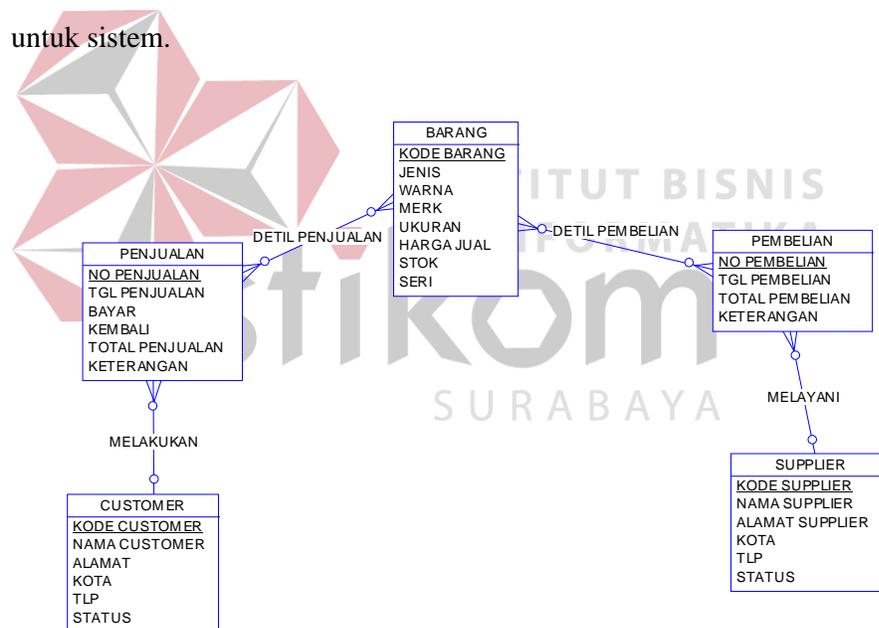
Setelah merancang desain proses menggunakan bantuan *context diagram* dan DFD, maka tahapan selanjutnya adalah merancang skema *database*.

#### A *Entity Relationship Diagram (ERD)*

ERD merupakan rancangan struktur basis data dari aplikasi yang akan dibangun. ERD dibagi menjadi dua yaitu *conceptual data model (CDM)* dan *physical data model (PDM)*.

##### 1. CDM

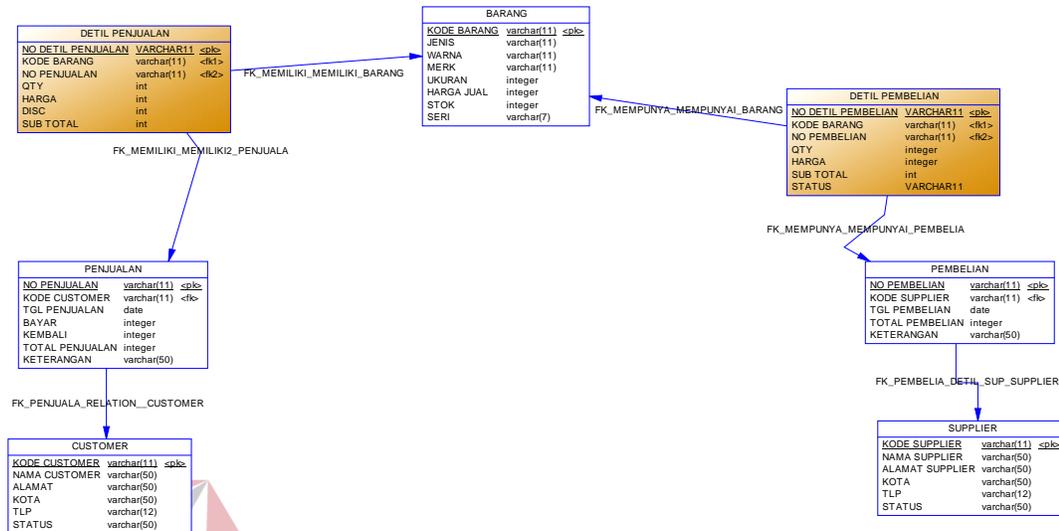
CDM menggambarkan keseluruhan konsep struktur basis data yang dirancang untuk sistem.



Gambar 3.12 CDM Sistem Informasi Penjualan Pada Toko Sepatu Ivory

Pada gambar 3.12 di atas terlihat CDM yang dirancang memiliki lima entitas yang terdiri dari entitas barang, penjualan, pembelian, *customer*, dan *supplier*.

## 2. PDM



Gambar 3.16 PDM Sistem Informasi Penjualan Pada Toko Sepatu Ivory

PDM menggambarkan secara detail konsep rancangan untuk struktur basis data sistem yang dibuat. Pada gambar diatas adalah PDM hasil dari CDM yang telah diolah sehingga entitas-entitas yang ada menjadi saling berhubungan satu sama lain.

### B Struktur Database

Berikut ini adalah struktur tabel yang akan digunakan pada pembuatan sistem. Terdapat tujuh tabel yang digunakan yaitu tabel penjualan, pembelian, barang, dan penerimaan barang. Penjelasan dari tabel-tabel tersebut dapat dilihat dibawah ini.

#### 1. Tabel Penjualan

*Primary Key* : NO\_PENJUALAN

*Foreign Key* : -

Tabel 3.4 Penjualan

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>
<u>NO PENJUALAN</u>	Varchar	5
TANGGAL PENJUALAN	DATE	-
KODE_CUSTOMER	Varchar	11
BAYAR	Integer	-
KEMBALI	Integer	-
TOTAL PENJUALAN	Integer	-
KETERANGAN	Varchar	50

## 2. Tabel Detil Penjualan

*Primary Key* : NO\_DETIL\_PENJUALAN

*Foreign Key* : 1. NO\_PENJUALAN

2. KODE\_BARANG

Tabel 3.5 Tabel Detil Penjualan

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>
<u>NO PENJUALAN</u>	Varchar	5
KODE BARANG	Varchar	11
NO_DETIL_PENJUALAN	Integer	-
QTY	Integer	-
HARGA	Integer	-
DISKON	Integer	-
SUB_TOTAL	Integer	-

## 3. Tabel Barang

*Primary Key* : KODE\_BARANG

*Foreign Key* : -

Tabel 3.6 Barang

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>
<u>KODE_BARANG</u>	Varchar	11
JENIS	Varchar	10
WARNA	Varchar	10
MERK	Varchar	10
UKURAN	Integer	-
SERI	Varchar	5
<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>
HARGA_JUAL	Integer	-
STOK	Integer	-
KETERANGAN	Varchar	50

4. Tabel *Customer*

*Primary Key* : KODE\_CUSTOMER

*Foreign Key* : -

Tabel 3.7 *CUSTOMER*

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>
<u>KODE_SUPPLIER</u>	Varchar	11
NAMA_CUSTOMER	Varchar	50
ALAMAT_CUSTOMER	Varchar	150
KOTA	Varchar	50
TELEPON	Varchar	50
STATUS	Varchar	50

5. Tabel *SUPPLIER*

*Primary Key* : KODE\_SUPPLIER

*Foreign Key* : -

Tabel 3.8 *SUPPLIER*

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>
<u>KODE_SUPPLIER</u>	Varchar	11
NAMA_SUPPLIER	Varchar	150
ALAMAT_SUPPLIER	Varchar	150
KOTA	Varchar	50

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>
TELEPON	Varchar	11
STATUS	Varchar	11

## 6. Tabel Detil Pembelian

*Primary Key* : NO\_DETIL\_PEMBELIAN

*Foreign Key* : 1. NO\_PEMBELIAN

2. KODE\_BARANG

Tabel 3.9 Detil Pembelian

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>
NO_PEMBELIAN	Integer	-
KODE_BARANG	Varchar	11
HARGA	Integer	-
QTY	Integer	-
SUB_TOTAL	Integer	-
NO_DETIL_PEMBELIAN	Varchar	5
STATUS	Varchar	11

## 7. Tabel Pembelian

*Primary Key* : NO\_PEMBELIAN

*Foreign Key* : -

Tabel 3.10 Pembelian

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>
NO_PEMBELIAN	Integer	-
KODE_SUPPLIER	Varchar	11
TANGGAL_PEMBELIAN	Date	-
TOTAL_PEMBELIAN	Integer	-
KETERANGAN	Varchar	50

### 3.2.3 Perancangan Antarmuka Pengguna

Setelah melakukan perancangan struktur tabel basis data, tahap selanjutnya adalah membuat perancangan antarmuka pengguna. Perancangan ini digunakan untuk membantu interaksi pengguna dengan sistem yang dibuat. Terdapat beberapa antarmuka pengguna yang dapat dilihat dibawah ini.

#### A. Desain Menu *Login*

Desain tampilan halaman *login* ini dibuat sama untuk semua pengguna dan akan tampil saat pengguna mengakses Sistem Informasi Penjualan pada Toko Ivory. Pengguna harus memasukkan *username* dan *password* yang telah ada di dalam *database akun* untuk bisa masuk ke halaman berikutnya. Tampilan desainnya bisa dilihat pada Gambar 3.14.

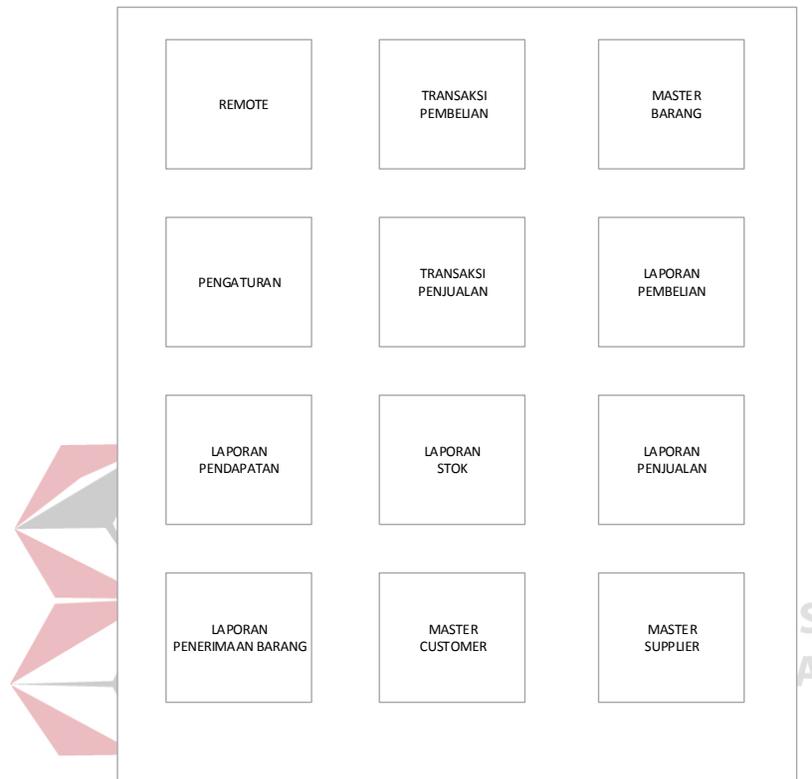


The image shows a login form titled "Form Login". It contains two text input fields: "User Name" and "Password", each with a placeholder text "Enter Text". Below the input fields are two buttons: "LOGIN" and "CANCEL". The form is overlaid on a background featuring a red and white geometric logo and the text "INSTITUT BISNIS" and "INFORMATIKA".

Gambar 3.14 Desain Menu *Login*

## 1. Desain Menu Utama

Gambar 3.15 merupakan desain menu utama sistem.



Gambar 3.15 Desain Menu Utama

## 2. Desain Menu Pembelian

Gambar 3.16 merupakan desain menu untuk melakukan aktifitas pembelian.

Dapat dilihat bahwa *user* harus mengisikan beberapa data mulai dari data sepatu yang akan dibeli, supplier mana yang dipilih, warna, jenis, dan juga ukuran. Setelah itu *user* memasukkan semua data barang yang akan dibeli maka *user* dapat menambahkan data barang yang akan dibeli tersebut kedalam tabel.

FORM PEMBELIAN

NO PEMBELIAN

TANGGAL

SUPPLIER

MERK BARANG	WARNA	JENIS	SERI	HARGA	QTY	
<input type="text" value="Text"/>	<input type="button" value="TAMBAH"/> <input type="button" value="BERSIH"/>					

KETERANGAN

**TOTAL: Rp. 0,-**

Gambar 3.16 Desain Transaksi Pembelian

### 3. Desain Penerimaan Barang

Gambar 3.17 merupakan desain menu untuk melakukan aktifitas penerimaan barang.

LAPORAN PENERIMAAN BARANG

**LAPORAN PENERIMAAN BARANG**

KODE BARANG	MERK	JENIS	WARNA	NAMA SUPPLIER	JUMLAH	KETERANGAN	AKSI
xxx	xxx	xxx	xxx	xxx	xxx	xxx	<input checked="" type="checkbox"/>

Gambar 3.17 Desain Transaksi Penerimaan Barang

#### 4. Desain Menu Penjualan

Gambar 3.18 merupakan desain menu untuk melakukan aktifitas penjualan

FORM PENJUALAN

NO PENJUALAN

TANGGAL

CUSTOMER

BAYAR : Rp.

KEMBALI: Rp. 0-,

MERK BARANG	WARNA	JENS	UKURAN	HARGA	QTY
<input type="text" value="Text"/>					

TAMBAH BERSIH

SIMPAN

KETERANGAN

Gambar 3.18 Desain Transaksi Penjualan

#### 5. Desain Laporan Penjualan

Gambar 3.19 merupakan desain untuk laporan penjualan.

LAPORAN PENJUALAN

PILIH TOKO

KODE BARANG	MERK	JENS	WARNA	HARGA JUAL	JUMLAH	SISA STOK	TOTAL
XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX

Gambar 3.19 Desain Laporan Penjualan

## 6. Desain Laporan Pembelian

Gambar 3.20 merupakan desain untuk laporan pembelian

LAPORAN PEMBELIAN

LAPORAN PEMBELIAN

PLH TOKO

KODE BARANG	MERK	JENS	WARNA	HARGA BELI	JUMLAH	JUMLAH STOK	TOTAL
XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX

Gambar 3.20 Desain Laporan Pembelian

## 3.3 Perancangan Pengujian

Pengembangan perangkat lunak merupakan tahapan pembuatan sistem dengan melakukan proses *coding*. Tahapan ini merubah hasil desain kebutuhan perangkat lunak menjadi bahasa yang dapat dimengerti oleh komputer. Setelah itu dilakukan pengujian terhadap sistem yang telah dibuat.

### 3.3.1 Perancangan Uji Coba Aplikasi

Perancangan ujicoba bertujuan untuk memastikan sistem yang telah dibuat sesuai dengan kebutuhan dan tujuan yang diharapkan. Kekurangan sistem

pada tahap ini akan dilakukan evaluasi sebelum tahap implementasi. Metode yang digunakan adalah pengujian *black box* yaitu pengujian yang dilakukan dengan hanya mengamati hasil eksekusi melalui uji data dan memeriksa fungsi dari perangkat lunak.

a. *Form Login*

*Form* ini digunakan untuk melakukan autentifikasi pengguna untuk memberikan hak akses pada pengguna dalam mengoperasikan sistem. Perancangan uji coba dapat dilihat pada tabel 3.11 berikut ini.

Tabel 3.11 Uji Coba *Form Login*

<b>Tujuan</b>	<b>Input</b>	<b>Output Diharapkan</b>
Autentifikasi Pengguna	Memasukkan <i>username</i> dan <i>password</i> yang benar dan tekan tombol “Masuk” pada form.	Tampil pesan pemberitahuan. Muncul menu utama sesuai hak akses pengguna.

b. *Form Pembelian*

*Form* ini digunakan untuk menambah, mengubah, dan menghapus data pembelian. Perancangan uji coba dapat dilihat pada tabel 3.12 berikut ini.

Tabel 3.12 Uji Coba *Form Pembelian*

<b>Tujuan</b>	<b>Input</b>	<b>Output Diharapkan</b>
Menambah data pembelian	Mengisi parameter yang diminta oleh aplikasi dan klik tombol “Tambah”	Data pembelian muncul di dalam <i>gridview</i>
Simpan Data Pembelian	Mengisi parameter yang diminta oleh aplikasi dan klik tombol “Simpan”	Data pembelian tersimpan
Menghapus data pembelian	Memilih tombol bersih	Daftar data barang yang akan dibeli terhapus

c. *Form* Master Penerimaan Barang

*Form* ini digunakan untuk menambah data penerimaan barang yang dimiliki.

Perancangan uji coba dapat dilihat pada tabel 3.13 berikut ini.

Tabel 3.13 Uji Coba *Form* Penerimaan Barang

<b>Tujuan</b>	<b>Input</b>	<b>Output Diharapkan</b>
Merubah status pembelian dari <i>waiting</i> menjadi <i>approved</i>	Menekan tombol aksi pada <i>form</i> lapooran penerimaan barang	Item penerimaan hilang dari laporan penerimaan barang dan status barang berubah dari <i>waiting</i> menjadi <i>approved</i>

d. *Form* Penjualan

*Form* ini digunakan untuk menambah, mengubah, dan menghapus data penjualan. Perancangan uji coba dapat dilihat pada tabel 3.14.

Tabel 3.14 Uji Coba *Form* Penjualan

<b>Tujuan</b>	<b>Input</b>	<b>Output Diharapkan</b>
Menambah data penjualan.	Mengisi parameter yang diminta oleh aplikasi dan klik tombol “Simpan”	Data penjualan tersimpan.
Menghapus data penjualan.	Memilih tombol hapus	Data penjualan terhapus.

e. *Form* Pengaturan

*Form* ini digunakan untuk menambah, mengubah, dan menghapus data penjualan. Perancangan uji coba dapat dilihat pada tabel 3.15

Tabel 3.15 Uji Coba *Form* Pengaturan

<b>Tujuan</b>	<b>Input</b>	<b>Output Diharapkan</b>
Menambah komputer <i>client</i> .	Mengisi parameter yang diminta oleh aplikasi dan klik tombol “Simpan”	Data komputer <i>client</i> tersimpan.

<b>Tujuan</b>	<b>Input</b>	<b>Output Diharapkan</b>
Update komputer <i>client</i>	Memilih tombol aksi lalu isi data komputer <i>client</i> yang baru	Data komputer <i>client</i> update
Menghapus komputer <i>client</i> .	Memilih tombol hapus	Data komputer <i>client</i> terhapus.

f. *Form* Master Barang

*Form* ini digunakan untuk menambah, mengubah, dan menghapus data barang. Perancangan uji coba dapat dilihat pada tabel 3.16 berikut ini.

Tabel 3.16 Uji Coba *Form* Master Barang

<b>Tujuan</b>	<b>Input</b>	<b>Output Diharapkan</b>
Menambah data barang	Mengisi parameter yang diminta oleh aplikasi dan klik tombol “Simpan”.	Data barang tersimpan.
Mengubah data barang.	Memilih data barang yang akan di ubah.	
	Mengisi parameter yang diminta oleh aplikasi dan klik tombol “Simpan”	Data barang tersimpan.
Menghapus data barang.	Memilih data barang yang akan di ubah.	
	Memilih tombol aksi	Data barang terhapus.

g. *Form Master Customer*

*Form* ini digunakan untuk menambah, mengubah, dan menghapus data *customer*. Perancangan uji coba dapat dilihat pada tabel 3.17 berikut ini.

Tabel 3.17 Uji Coba *Form Master Customer*

<b>Tujuan</b>	<b>Input</b>	<b>Output Diharapkan</b>
Menambah data <i>customer</i>	Mengisi parameter yang diminta oleh aplikasi dan klik tombol “Simpan”.	Data <i>customer</i> tersimpan.
Mengubah data <i>customer</i> .	Memilih data <i>customer</i> yang akan di ubah.	
	Mengisi parameter yang diminta oleh aplikasi dan klik tombol “Simpan”	Data <i>customer</i> tersimpan.
Menghapus data <i>customer</i> .	Memilih data <i>customer</i> yang akan di ubah.	
	Memilih tombol aksi	Data <i>customer</i> terhapus.

h. *Form Master Supplier*

*Form* ini digunakan untuk menambah, mengubah, dan menghapus data *supplier*. Perancangan uji coba dapat dilihat pada tabel 3.18 berikut ini.

Tabel 3.18 Uji Coba *Form Master Supplier*

<b>Tujuan</b>	<b>Input</b>	<b>Output Diharapkan</b>
Menambah data <i>supplier</i>	Mengisi parameter yang diminta oleh aplikasi dan klik tombol “Simpan”.	Data <i>supplier</i> tersimpan.
Mengubah data <i>supplier</i> .	Memilih data <i>supplier</i> yang akan di ubah.	
	Mengisi parameter yang diminta oleh aplikasi dan klik tombol “Simpan”	Data <i>supplier</i> tersimpan.
Menghapus data <i>supplier</i> .	Memilih data <i>supplier</i> yang akan di ubah.	
	Memilih tombol aksi	Data <i>supplier</i> terhapus.

i. *Form Remote*

*Form* ini digunakan untuk melakukan pengecekan stok barang ke toko lain yang terintegrasi. Perancangan uji coba dapat dilihat pada tabel 3.19 berikut ini.

Tabel 3.19 Uji Coba *Form Remote*

<b>Tujuan</b>	<b>Input</b>	<b>Output Diharapkan</b>
Pengecekan sambungan dengan server database.	Nama Server database yang digunakan.	Tampil pesan pemberitahuan.

