

## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

Pada bab ini membahas tentang identifikasi permasalahan, analisis permasalahan, solusi permasalahan dan perencanaan sistem dalam Rancang Bangun Aplikasi Penjualan Obat pada Apotek Ita Farma. Tahap awal adalah pengumpulan data dengan teknik wawancara dan observasi. Tahap selanjutnya adalah melakukan identifikasi permasalahan dan analisis permasalahan.

#### **1.1      Analisis Sistem**

Pada tahapan ini menjelaskan mengenai bagaimana kondisi saat ini pada suatu organisasi bisnis. *Survey* dilakukan untuk mengevaluasi permasalahan yang sedang terjadi saat ini, sehingga dapat diusulkan mengenai suatu perbaikan. Pengumpulan data dilakukan untuk mengetahui kondisi atau informasi mengenai organisasi bisnis. Adapun langkah-langkah pengumpulan data diantaranya :

##### **1. Observasi**

Dilakukan secara langsung pada Apotek Ita Farma untuk mengetahui permasalahan yang terjadi saat ini.

##### **2. Wawancara**

Mengajukan beberapa pertanyaan kepada pihak yang bersangkutan yaitu pemilik Apotek Ita Farma untuk memperoleh data-data yang dibutuhkan yang nantinya dapat digunakan untuk kebutuhan sistem yang akan dibuat.

##### **3. Studi Pustaka**

Kumpulan beberapa informasi seperti buku serta jurnal mengenai penjualan, apotek dan sistem informasi serta penunjang lainnya yang berkaitan dengan

topik yang dibahas oleh penulis yang nantinya akan dijadikan sebagai referensi dalam membuat tugas akhir ini.

## 1.2 Analisis Bisnis

### 1. Identifikasi Masalah

Dalam proses bisnisnya Apotek Ita Farma memiliki beberapa kendala yang menghambat pelayanan, dari hasil observasi dan wawancara yang dilakukan pada apotek ita farma terdapat 2 permasalahan, permasalahan pertama terkait dengan pencatatan transaksi penjualan dimana setiap terjadi penjualan obat bagian kasir akan melakukan pencatatan dalam buku transaksi penjualan yang berisi tentang nama obat, jumlah dan harga. Adapun aktifitas yang terkait dengan transaksi penjualan, antara lain proses rekapitulasi penjualan perhari dilakukan setiap hari oleh bagian kasir setelah apotek tutup di mana proses rekapitulasi ini masih dilakukan dengan menggunakan kalkulator dalam proses perhitungannya serta dimungkinkan terjadi kesalahan dalam perekapan. Saat data itu diolah kembali menggunakan *excel* terdapat selisih pendapatan pada periode tertentu dapat dilihat pada tabel 3.1 dan hal ini menyebabkan kesalahan informasi mengenai laporan pendapatan perperiode. Kemudian permasalahan berikutnya yaitu terkait dengan persediaan obat, saat ini Apotek Ita Farma melakukan pencatatan persediaan obat masuk dengan cara menuliskannya dalam sebuah buku tetapi saat obat keluar pada catatan buku persediaan tidak sesuai dengan kondisi saat ini yang menyebabkan perbedaan antara pencatatan pada buku persediaan dengan kondisi jumlah rill obat.

Berdasarkan dari permasalahan di atas maka didapatkan solusi untuk menyelesaikan permasalahan tersebut dengan membuatkan sistem informasi

penjualan pada Apotek Ita Farma. Yang menghasilkan informasi mengenai pencatatan transaksi penjualan dan informasi laporan stok obat.

Tabel 3. 1 Pendapatan tahun 2015

| BULAN     | Data Asli<br>(Rp) | Data Diolah dengan Excel<br>(Rp) | Selisih<br>(Rp) |
|-----------|-------------------|----------------------------------|-----------------|
| JANUARI   | 11.932.200        | 12.560.000                       | -627.800        |
| FEBRUARI  | 4.339.200         | 4.339.800                        | -600            |
| MARET     | 5.783.700         | 5.783.700                        | 0               |
| APRIL     | 3.612.000         | 3.612.000                        | 0               |
| MEI       | 5.133.600         | 4.958.100                        | -175.500        |
| JUNI      | 4.464.400         | 4.196.400                        | -268.000        |
| JULI      | 5.176.000         | 5.176.000                        | 0               |
| AGUSTUS   | 6.619.000         | 6.619.000                        | 0               |
| SEPTEMBER | 8.143.000         | 8.143.000                        | 0               |
| OKTOBER   | 8.986.000         | 8.986.000                        | 0               |
| NOPEMBER  | 9.517.500         | 9.511.300                        | -6.200          |
| DESEMBER  | 11.435.000        | 11.435.000                       | 0               |

Sumber : Apotek Ita Farma

Pada tabel 3.2 merupakan hasil identifikasi dari pencatatan transaksi penjualan dan pencatatan stok obat, berikut hasil identifikasi masalah.

Tabel 3. 2 Hasil Identifikasi Masalah

| Identifikasi Permasalahan   | Penyebab   | Alternatif Solusi  |
|---|--|--|
| Kesalahan pada pencatatan transaksi penjualan.  | Melakukan pencatatan dalam buku transaksi penjualan secara manual dan membutuhkan waktu yang lama. | Sistem yang dapat membantu melakukan perekapan data penjualan dengan benar.    |
| Perbedaan pencatatan antara buku stok dengan jumlah riil. Sehingga terjadi kesalahan pada pencatatan stok obat. | Pencatatan stok obat yang tidak lengkap.   | Sistem yang dapat membantu memberikan informasi mengenai stok barang saat ini. |

## 2. Identifikasi Pengguna

Berdasarkan hasil wawancara dengan pemilik Apotek Ita Farma terdapat beberapa pengguna yang memiliki peran dalam mengoperasikan aplikasi yang dibangun yaitu bagian Kasir dan Apoteker.

## 3. Identifikasi Data

Setelah dilakukan identifikasi permasalahan dan pengguna, maka dapat melakukan identifikasi data pendukung yang diperlukan dalam aplikasi ini adalah data karyawan, data resep, data obat, data stok obat dan data penjualan.

## 4. Identifikasi Fungsi

Setelah dilakukan proses identifikasi permasalahan, pengguna, dan data, maka dapat diidentifikasi fungsi dari proses penjualan sebagai berikut : Mengelola data master, melakukan penjualan, membuat laporan.

## 5. Identifikasi Hasil Keluaran

Setelah melakukan proses identifikasi masalah, identifikasi pengguna, identifikasi data dan identifikasi fungsi, maka dapat menghasilkan keluaran dari aplikasi berupa laporan stok obat, laporan penjualan, laporan obat terlaris dan laporan transaksi dokter.

### 1.3 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. Berikut peran dan tanggung jawab pengguna dapat dilihat pada Tabel 3.3.

Tabel 3. 3 Peran dan tanggung jawab pengguna

| Aktor          | Peran  | Tanggung Jawab                                    |
|----------------|--|---|
| Kasir          | Mengelola data resep.<br>Mengelola pembayaran. | Mencatat data resep.<br>Mencatat pembayaran obat. |
| Apoteker       | Mengelola data obat.                           | Mencatat data obat.<br>Mengupdate data obat.      |
| Pemilik Apotek | Mengecek Laporan                               | Menerima Laporan                                  |

Dalam membangun sebuah aplikasi diperlukan perancangan perangkat lunak yang sesuai dengan kebutuhan pengguna. Hal ini bertujuan untuk memudahkan pengguna dalam mengoperasikan aplikasi yang sesuai dengan kebutuhan fungsional. Fungsi-fungsi tersebut dikelompokkan berdasarkan entitas dan dapat dilihat pada Tabel 3.4.

Tabel 3.4 Fungsi-fungsi Entitas

| Entitas  | Deskripsi Fungsi yang Diperlukan  |
|----------|---|
| Kasir    | a. Melakukan Transaksi penjualan.<br>b. Mencetak Nota Penjualan.  |
| Apoteker | a. Melakukan pencatatan data obat<br>b. Melakukan pencatatan stok obat<br>c. Mencetak laporan stok obat<br>d. Mencetak laporan penjualan<br>e. Mencetak laporan obat terlaris<br>f. Mencetak laporan transaksi dokter |

#### 1.4 Analisa Kebutuhan Data

Dari analisis kebutuhan pengguna yang telah disusun sebelumnya, maka dibutuhkan beberapa data untuk menunjang aplikasi yang dibangun. Terdapat beberapa data yang diperlukan dalam membangun aplikasi, data tersebut meliputi:

##### 1. Data Karyawan

Data karyawan digunakan sebagai data tambahan untuk pembuatan aplikasi.

Data karyawan yang diperlukan adalah Nama, Alamat, No.Tlp, Jabatan, *Password*.

## 2. Data Resep

Data resep di dapat dari pembeli, pada data resep terdapat beberapa informasi yaitu nama dokter, no. ijin praktek, alamat praktek, nama pasien, umur, alamat, nama obat.

## 3. Data Obat

Merupakan data yang digunakan untuk menyimpan data obat yang berisi mengenai nama obat, jenis obat, satuan, produsen, persentase, harga beli serta harga jual.

## 4. Data Stok Obat

Data yang digunakan untuk menyimpan stok obat yang terdiri dari Nama obat, satuan, stok, tanggal kadaluarsa, nama *supplier*, tanggal masuk obat.

## 5. Data Penjualan

Merupakan data yang dicatat ketika ada penjualan, data tersebut meliputi nama obat, harga obat, jumlah obat, total penjualan, bayar, kembali. Jika menggunakan resep maka akan ada nama dokter dan nama pasien.

## 1.5 Analisa Kebutuhan Fungsi

Berdasarkan kebutuhan pengguna yang telah dibuat sebelumnya, maka dapat diimplementasikan dengan membuat kebutuhan fungsional dari aplikasi yang dibangun. Pada tahapan ini kebutuhan fungsi digunakan untuk mengimplementasikan seluruh fungsi yang didapat dari hasil analisis kebutuhan pengguna. Fungsi-fungsi tersebut adalah sebagai berikut:

## 1. Fungsi Pencatatan Data Obat

Tabel 3.5 Kebutuhan Fungsi Pencatatan Data Obat

|                |   |
|----------------|---|
| Fungsi         | Melakukan pencatatan data obat  |
| Deskripsi      | Fungsi ini digunakan untuk melakukan pencatatan data obat baru untuk disimpan.  |
| Pemicu         | -   |
| Awal           | Autentifikasi oleh Apoteker   |
| Alur           | <ol style="list-style-type: none"> <li>1. <i>User</i> membuka <i>form</i> Data Obat pada aplikasi.</li> <li>2. Aplikasi menampilkan parameter data yang harus dilengkapi.</li> <li>3. <i>User</i> melengkapi data obat yang diminta oleh aplikasi.</li> <li>4. <i>User</i> memilih menu “Simpan”           <ol style="list-style-type: none"> <li>4.1 Data tersimpan pada <i>Database</i>.</li> </ol> </li> <li>5. <i>User</i> menutup form Data obat pada aplikasi.</li> </ol> |
| Error Handling | <ol style="list-style-type: none"> <li>1. Jika <i>user</i> tidak melengkapi isi data yang diminta oleh aplikasi akan muncul peringatan bahwa ada data yang belum diisi.</li> </ol>  |

A. Proses pencatatan obat terdapat tiga jenis obat diantaranya :

1. Jenis obat bebas ini merupakan jenis obat yang dapat dijual bebas kepada pembeli tanpa harus menggunakan resep dokter.
2. Jenis obat bebas terbatas ini terdapat beberapa obat yang harus dibeli menggunakan resep dokter, dikarenakan obat bebas terbatas merupakan obat keras yang memiliki batas pada setiap takaran dan kemasan yang digunakan untuk mengobati penyakit ringan yang dapat dikenali oleh penderita sendiri seperti batuk, flu, dan demam.
3. Jenis obat keras ini merupakan semua jenis obat yang memiliki takaran perdosis maksimum atau yang tercantum pada daftar obat keras. Penjualan obat keras ini harus dengan resep dokter.

Jadi pada saat penjualan obat dapat diketahui mana obat yang harus dijual menggunakan resep dokter dan mana obat yang dapat dijual tanpa menggunakan resep dokter.

- B. Perhitungan konversi satuan Obat memiliki cara untuk melakukan pengkonversian obat seperti contoh dibawah ini :

1 box berisi 5 strip, 1 strip berisi 10 tablet, untuk mengetahui cara perhitungan jumlah obat seluruhnya adalah 5 strip x 10 tablet = 50 tablet, jadi dapat diketahui per box terdapat 50 tablet, hasil ini merupakan hasil satuan pengkonversian terkecil. Sedangkan dalam menentukan penjualan obat secara ecer, Apotek Ita Farma memiliki ketentuan jika harga obat 1 strip harganya >Rp 50.000 dengan jumlah obat 10 tablet, maka dikategorikan obat mahal karena harga persatuannya mencapai Rp 5.000.

- C. Perhitungan persentase keuntungan ditentukan oleh pihak apotek dengan ketentuan antara 5% sampai 15% ini bisa ditentukan berdasarkan jenis obat, ketentuan itu akan dikalikan dengan harga beli kemudian hasilnya dijadikan sebagai harga jual. Contoh 10% dikali harga beli Rp. 3000 maka didapat hasil yang dijadikan harga jual sebesar Rp. 3300, jadi keuntungan yang didapat sebanyak Rp. 300.

Tabel 3.6 Kebutuhan Fungsi Pencatatan Stok

|                |   |
|----------------|---|
| Fungsi         | Melakukan pencatatan stok   |
| Deskripsi      | Fungsi ini digunakan untuk melakukan pencatatan stok obat.  |
| Pemicu         | -   |
| Awal           | Autentifikasi oleh Apoteker   |
| Alur           | <ol style="list-style-type: none"> <li>1. <i>User</i> membuka <i>form</i> Stok</li> <li>2. Aplikasi akan menampilkan parameter data yang harus dilengkapi.</li> <li>3. <i>User</i> melengkapi data yang diminta oleh aplikasi.</li> <li>4. <i>User</i> memilih menu “Simpan”           <ol style="list-style-type: none"> <li>4.1 Data tersimpan pada <i>Database</i>.</li> </ol> </li> <li>5. <i>User</i> menutup <i>form</i> Stok pada aplikasi.</li> </ol> |
| Error Handling | <ol style="list-style-type: none"> <li>1. Jika <i>user</i> tidak melengkapi isi data yang diminta oleh aplikasi akan muncul peringatan bahwa ada data yang belum diisi.</li> </ol>  |

Pada proses pencatatan stok obat terdapat beberapa proses yang harus dilakukan yaitu menginputkan nama obat, jumlah stok, nama *supplier*, tanggal kadaluarsa dan tanggal masuk. Pada setiap obat memiliki tanggal kadaluarsa yang tidak sama, contohnya persediaan stok awal obat Konidin dengan tanggal kadaluarsa 5 Juli 2017 sebanyak 20 pcs kemudian stok bertambah dengan tanggal kadaluarsa yang berbeda yaitu tanggal 18 Februari 2018 sebanyak 100 pcs. Jadi stok obat konidin memiliki 120 pcs dengan tanggal kadaluarsa yang berbeda.

## 2. Fungsi Transaksi Penjualan

Tabel 3.7 Kebutuhan Fungsi Transaksi Penjualan

|           |  |
|-----------|--|
| Fungsi    | Melakukan Transaksi penjualan  |
| Deskripsi | Fungsi ini digunakan untuk melakukan pencatatan transaksi penjualan.   |
| Pemicu    | -  |
| Awal      | <p>Autentifikasi oleh Kasir</p> <ol style="list-style-type: none"> <li>1. <i>User</i> membuka form penjualan.<br/>1.1 Klik “button” Baru</li> <li>2. Aplikasi akan menampilkan parameter data yang harus dilengkapi.             <ol style="list-style-type: none"> <li>2.1 <i>User</i> akan meng-input-kan data dokter dengan menekan button “Tambah dokter”.</li> <li>2.2 Kemudian user akan memilih data dokter pada tabel dokter.</li> </ol> </li> <li>3. <i>User</i> akan meng-input-kan data pasien.             <ol style="list-style-type: none"> <li>3.1 <i>User</i> akan meng-input-kan data dokter dengan menekan button “Tambah Pasien”</li> <li>3.2 Kemudian <i>user</i> akan memilih data pasien pada tabel pasien.</li> </ol> </li> <li>4. <i>User</i> akan meng-input-kan daftar penjualan.             <ol style="list-style-type: none"> <li>4.1 <i>User</i> akan memilih memilih data obat yang diinginkan pada tabel data obat.</li> <li>4.2 Setelah memilih data obat nanti akan mucul satuan serta harga obat</li> <li>4.3 Kemudian <i>user</i> akan meng-input-kan <i>Qty</i> obat yang diinginkan.</li> <li>4.4 Kemudian secara otomatis mengetahui jumlah, yang didapat dari harga jual di kali <i>Qty</i>.</li> <li>4.5 Kemudian <i>user</i> memilih button “Tambah”</li> </ol> </li> <li>5. <i>User</i> akan meng-input-kan pembayaran.</li> <li>6. <i>User</i> akan memilih button Simpan.             <ol style="list-style-type: none"> <li>6.1 Data tersimpan pada Database.</li> </ol> </li> </ol> |
| Alur      |  |

|                |  |
|----------------|--|
|                | 6.2 Aplikasi akan menampilkan nota.  |
| Error Handling | <p>1. Jika <i>user</i> tidak melengkapi isi data yang diminta oleh aplikasi akan muncul peringatan bahwa ada data yang belum diisi.</p> <p>2. Jika <i>user</i> meng-input-kan jenis obat yang harus dibeli menggunakan resep, tetapi <i>user</i> tidak memilih kolom penjualan obat dengan resep maka akan muncul peringatan jika penjualan tidak dapat dilakukan.</p> |

Berikut ini merupakan alur proses penjualan yang akan digunakan pada pembuatan aplikasi ini. Proses penjualan yang dilakukan oleh bagian kasir, terdapat 2 proses penjualan yaitu proses penjualan menggunakan resep dokter dan proses penjualan tanpa resep dokter.

#### 1. Penjualan menggunakan resep

Pada penjualan dengan menggunakan resep ini dilakukan dengan cara menambahkan data yang ada pada resep seperti nama dokter, nama pasien, umur pasien dan alamat pasien, kemudian *input* data obat sesuai dengan resep yang diterima dari pembeli, lalu kasir memproses pembayaran.

#### 2. Penjualan *non* resep

Pada penjualan *non* resep obat bisa dibeli tanpa harus menunjukkan resep. Proses penjualan ini hanya menjual dua jenis obat, yaitu jenis obat bebas dan obat bebas terbatas. Jika pembelian obat dengan jenis obat keras maka transaksi penjualan tidak dapat diproses.

### 1.6 Analisa Kebutuhan Keluaran

Dari Keempat analisa di atas yang dilakukan selanjutnya adalah analisa kebutuhan keluaran. Analisa kebutuhan keluaran menjelaskan tentang hasil keluaran dari aplikasi yang dibuat. Keluaran yang akan dihasilkan, antara lain :

### 1. Laporan Stok Obat

Merupakan hasil rekapitulasi data penjualan obat, laporan ini berisi tentang jumlah stok obat saat ini.

### 2. Laporan Penjualan

Merupakan hasil rekapitulasi penjualan obat perhari, laporan ini berisi tentang pendapatan apotek perhari.

### 3. Laporan Obat Terlaris

Laporan mengenai obat yang sering laku terjual pada apotek.

### 4. Laporan Transaksi Dokter

Merupakan hasil rekapitulasi penjualan obat yang dilakukan oleh dokter pada apotek.

### 1.7 Perancangan Sistem

Perancangan sistem ini bertujuan untuk mendefinisikan kebutuhan-kebutuhan fungsional, menggambarkan aliran data dan alur sistem, dan sebagai tahap awal sebelum implementasi sistem. Langkah-langkah dalam perancangan sistem adalah sebagai berikut:

1. Diagram *Input Process Output* ( IPO).
2. *Document Flow*.
3. *System Flow*.
4. *Data Flow Diagram* (DFD), yang meliputi: *context diagram*, DFD level 0 dan DFD level 1.
5. *Entity Relationship Diagram* (ERD), yang meliputi: *Conceptual Data Model* (CDM), dan *physical data model* (PDM).
6. Struktur Data.

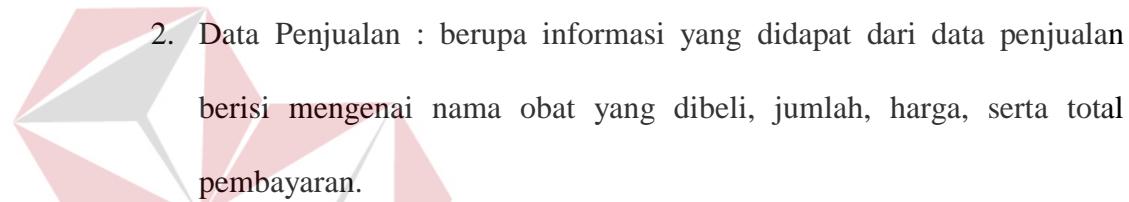
7. Desain Antar Muka.

### **1.7.1    *Input Proses Output***

Adapun penjelasan mengenai *Input*, *Proses*, *Output* Berdasarkan gambar Blok diagram diatas :

a. *Input*

1. Data Obat : berupa informasi data obat masuk yang berisi nama obat, jenis obat, bentuk obat, satuan obat, produsen, persentase, harga beli, harga jual.



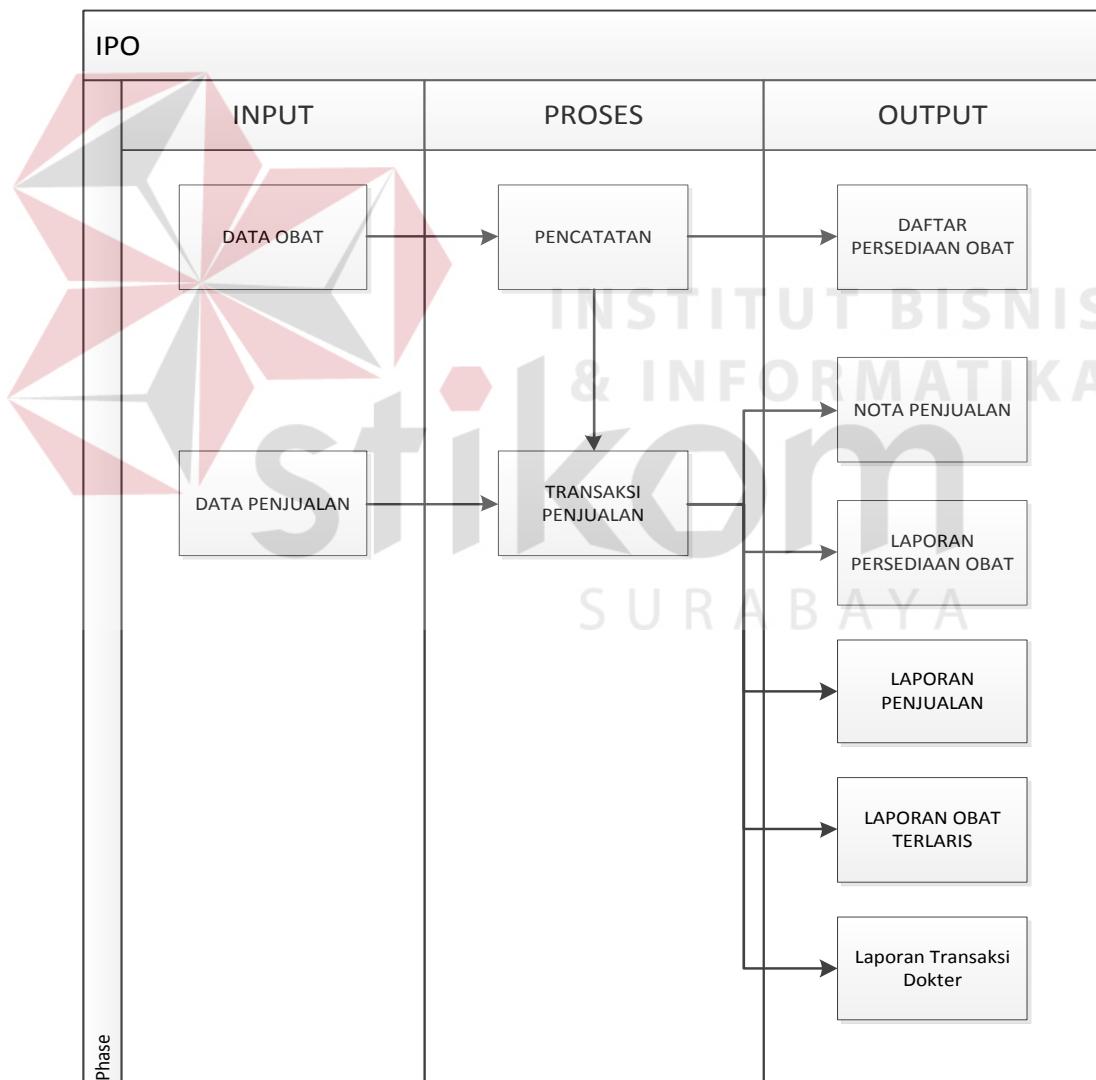
b. *Proses*

1. Pencatatan persediaan : proses ini mencatat obat masuk yang akan disimpan sebagai persediaan obat. Pada proses ini obat akan didata yang terdiri dari Id\_obat, nama obat, jenis obat, ukuran, golongan, produsen, tanggal kadaluarsa dan harga.
2. Proses penjualan : ini merupakan transaksi penjualan yang dapat melakukan transaksi-transaksi penjualan baik pembelian dengan resep atau *non* resep.

c. *Output*

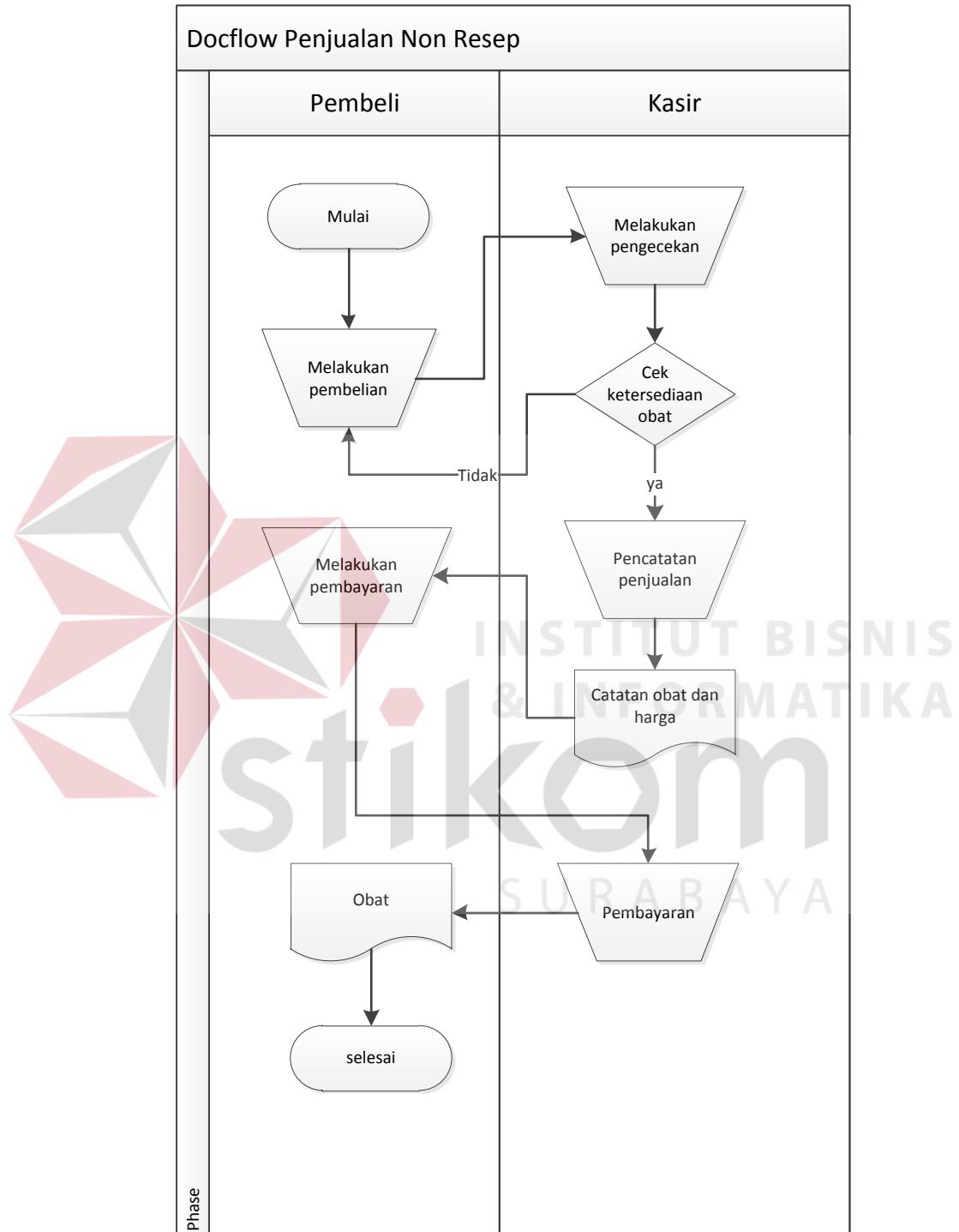
1. Daftar persediaan obat : laporan yang berisi data persediaan obat.
2. Nota penjualan : laporan yang berisi transaksi penjualan yang akan diberikan kepada pembeli
3. Laporan stok obat : laporan persediaan obat per-periode.

4. Laporan penjualan : laporan penjualan per-periode yang nanti akan di cek oleh *manager* untuk mengetahui perkembangan penjualan pada apotek.
5. Laporan obat terlaris : merupakan daftar laporan obat apa saja yang sering laku terjual pada Apotek Ita Farma.
6. Laporan Transaksi dokter : laporan yang dapat mengetahui jumlah transaksi setiap dokter.



Gambar 3. 1 Input Proses Output

### 1.7.2 Document Flow Penjualan Non Resep

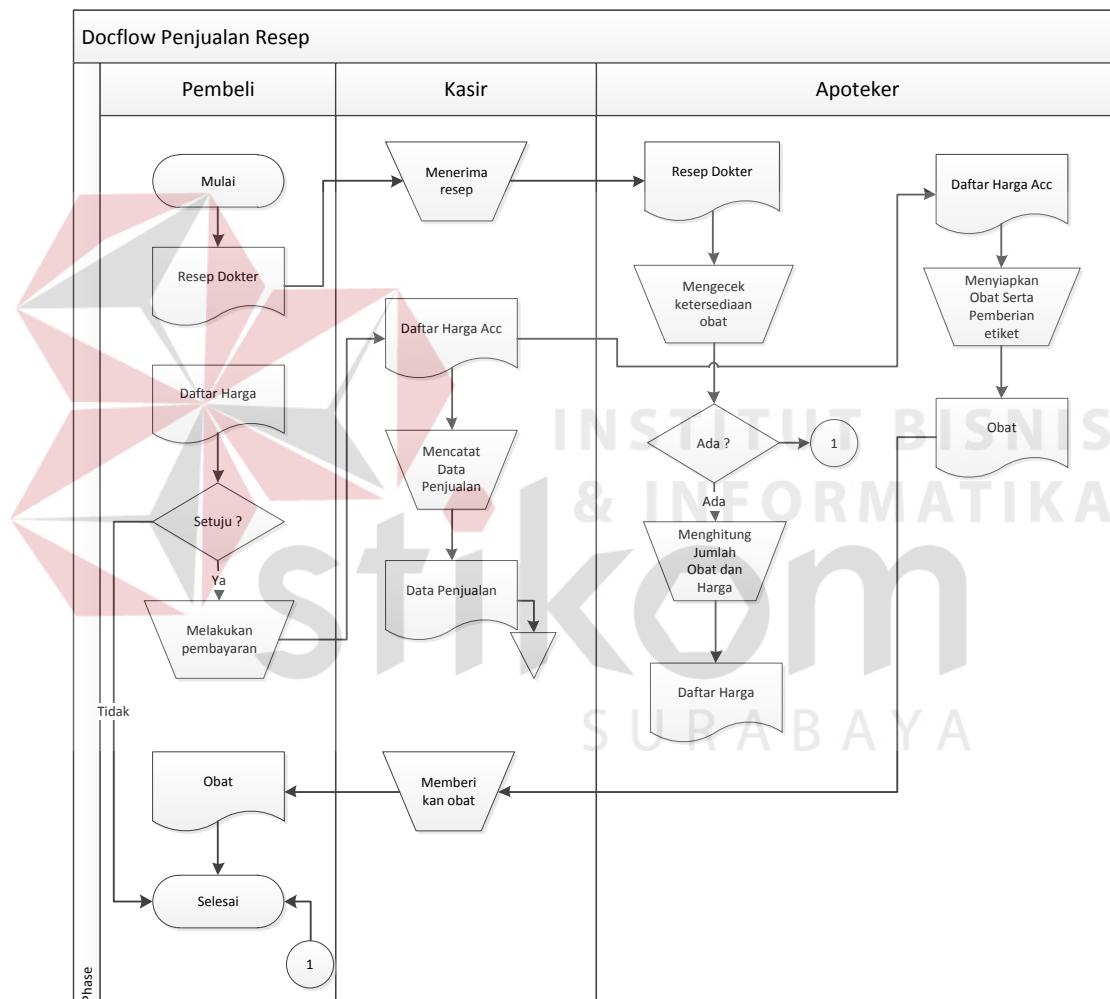


Gambar 3.2 Document Flow Penjualan Non Resep

Pada gambar di atas dapat dilihat proses penjualan *non resep* dimulai dari pelanggan memberitahu obat yang akan dibeli, kemudian bagian kasir akan

melakukan pengecekan, jika tidak ada maka akan ditawarkan dengan obat dengan *merk* berbeda. Jika obat ada maka akan dicatat dalam buku penjualan, pembeli melakukan pembayaran, kemudian obat diserahkan kepada pembeli setelah itu proses selesai.

### 1.7.3 Document Flow Penjualan Resep

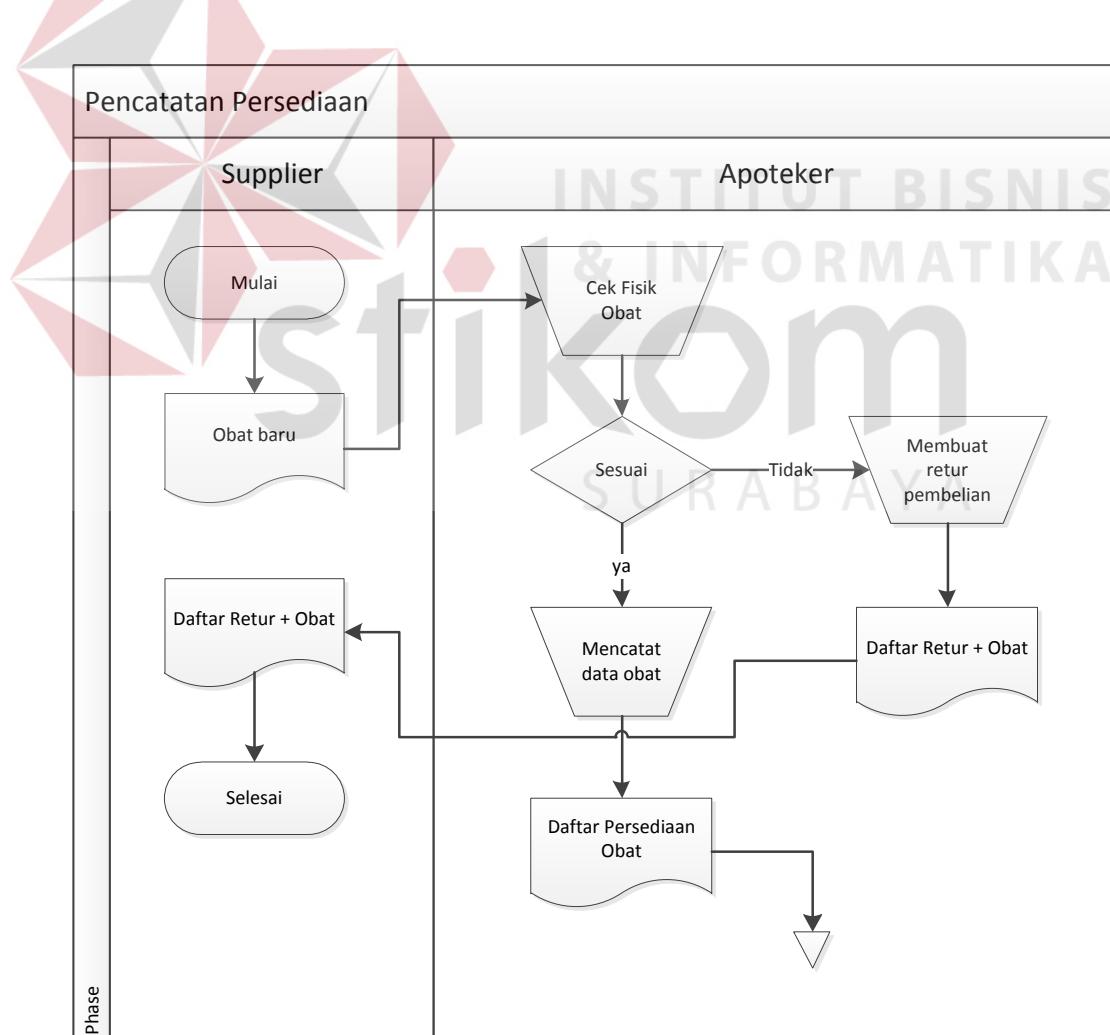


Gambar 3.3 Document Flow Penjualan Resep

Pada gambar di atas dimulai dari pelanggan menyerahkan resep kepada bagian kasir yang akan dicek oleh apoteker. Apoteker memeriksa kelengkapan dan keabsahan resep setelah dianggap benar maka apoteker akan memeriksa

ketersediaan obat, tahap selanjutnya apoteker akan menghitung jumlah obat yang diminta kemudian menghitung harganya. Kemudian harga tersebut akan diinformasikan kepada pelanggan, jika pelanggan tidak dapat menebus semua resep maka pelanggan akan diberi *copy* resep yang dapat digunakan untuk membeli obat kembali. Jika pembeli setuju dengan harga yang ditawarkan maka dilakukan proses pembayaran kepada kasir untuk dicatat dalam buku penjualan. Selanjutnya obat disiapkan, setelah selesai obat akan diberi e-tiket (cara penggunaan obat) kemudian diserahkan kepada pelanggan.

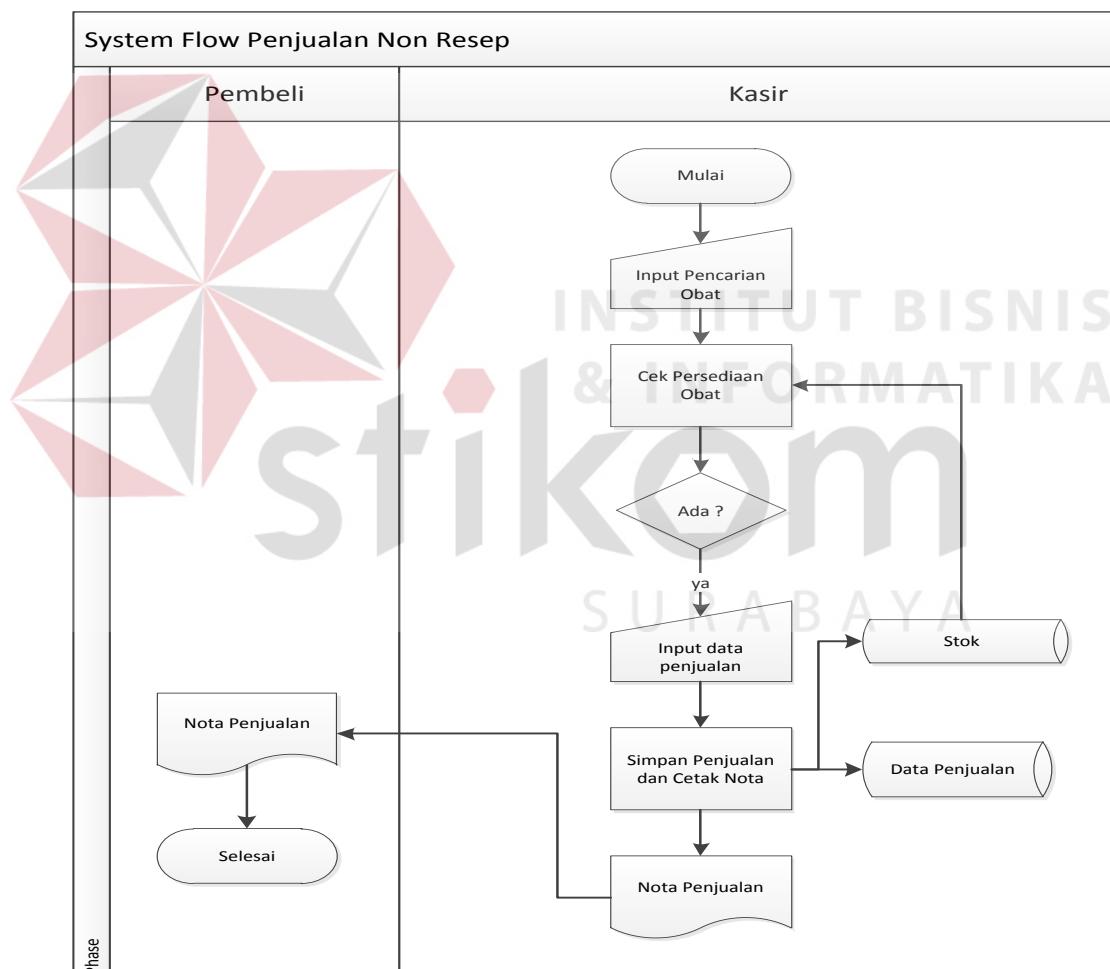
#### 1.7.4 Document Flow Pencatatan Persediaan



Gambar 3.4 Document Flow Pencatatan persediaan

Pada gambar di atas merupakan proses pencatatan persediaan barang dimulai dari *supplier* memberikan obat kemudian apoteker akan mengecek keadaan fisik dari obat tersebut apakah ada kerusakan fisik atau tidak, jika tidak ada maka obat itu akan dicatat kedalam buku persediaan yang nantinya akan digunakan untuk melihat stok saat ada penjualan. Jika ada kerusakan maka akan dibuatkan retur pembelian yang akan diberikan kepada *supplier* kembali.

### 1.7.5 System Flow Penjualan Non Resep

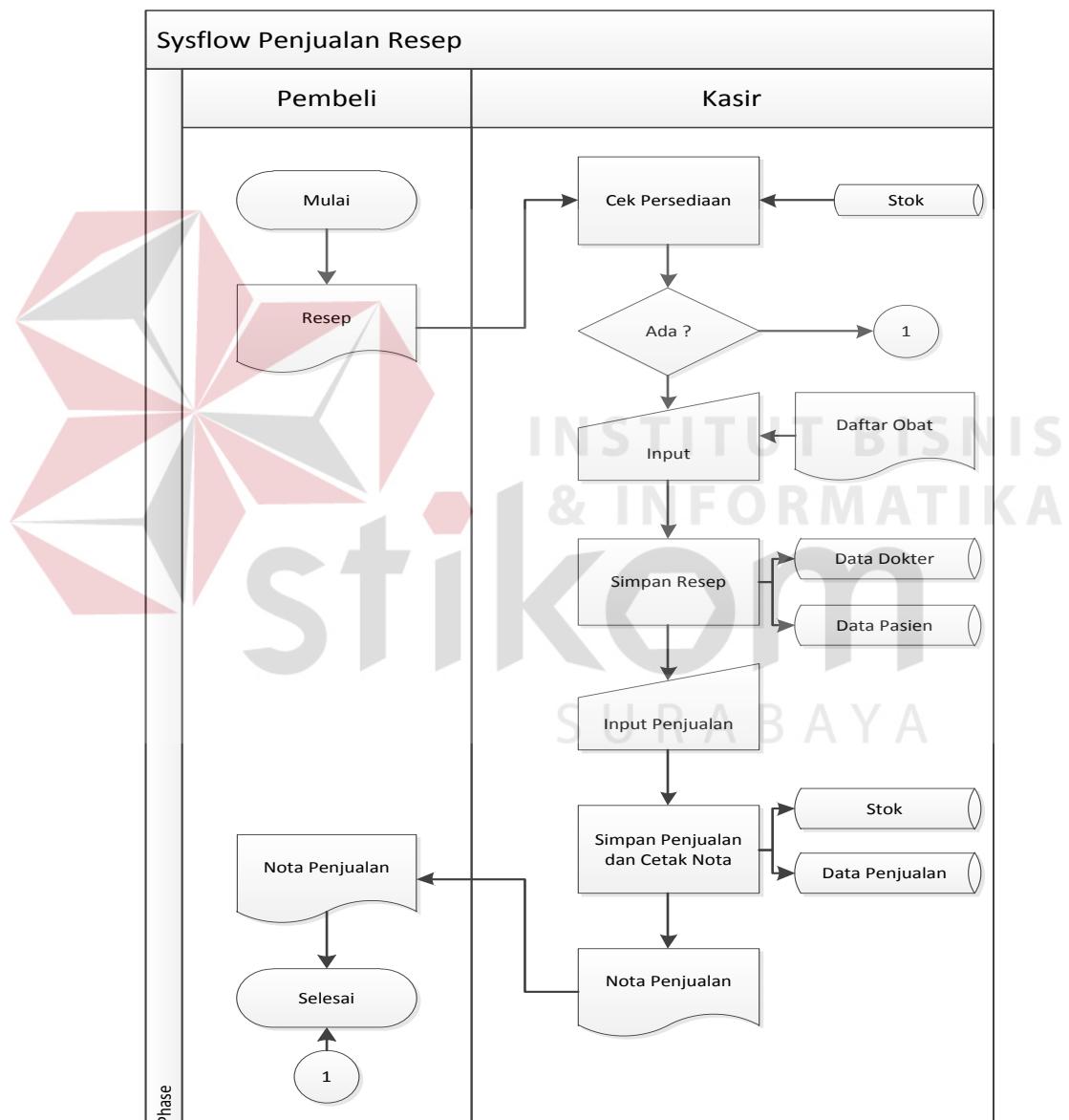


Gambar 3.5 System Flow Penjualan Non Resep

Pada gambar di atas dimulai dari pelanggan memberitahu obat yang akan dibeli, kemudian bagian kasir akan mengecek persediaan obat. Apabila obat

tersedia maka dilakukan *input* data penjualan selanjutnya proses pembayaran yang terjadi akan disimpan di *Database* penjualan lalu akan mengeluarkan nota penjualan rangkap 2 yang 1 akan diberikan kepada pelanggan yang satunya lagi akan disimpan kemudian proses selesai.

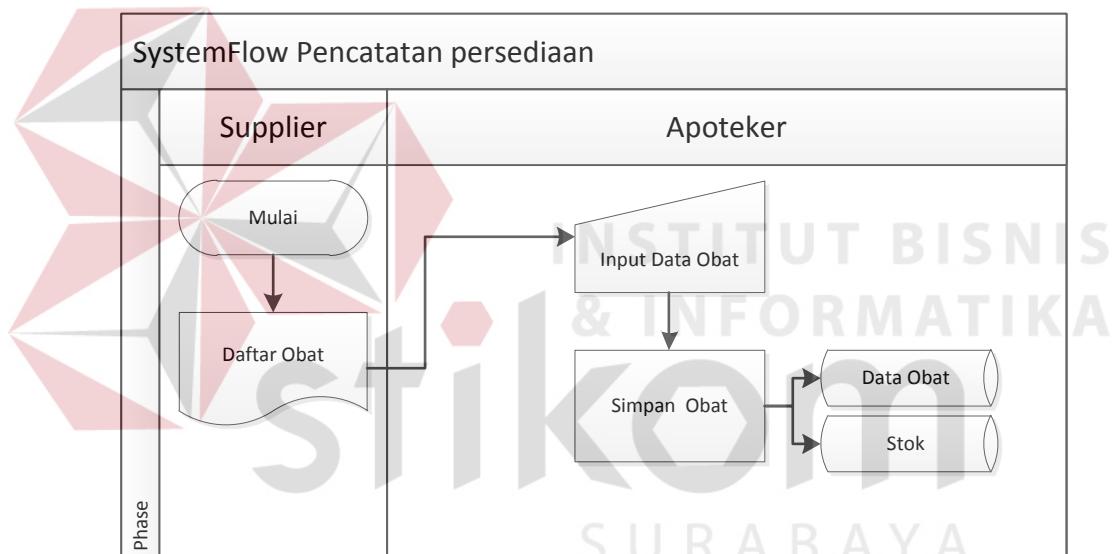
### 1.7.6 System Flow Penjualan Resep



Gambar 3.6 System Flow Penjualan Resep

Proses penjualan menggunakan resep dimulai dari pembeli memberikan data resep kepada kasir lalu akan dicek persediaan obatnya jika ada maka akan langsung diinputkan nama obat, jumlah obat, jumlah pembayarannya, nama dokter serta nama pasien. Pembayaran yang akan diberikan kepada kasir untuk di inputkan pada proses penjualan yang nantinya akan menghasilkan nota penjualan rangkap 2 satu untuk pembeli satu untuk disimpan. Setelah itu obat akan diberikan kepada pembeli beserta Nota.

#### 1.7.7 *System Flow Pencatatan Persediaan*



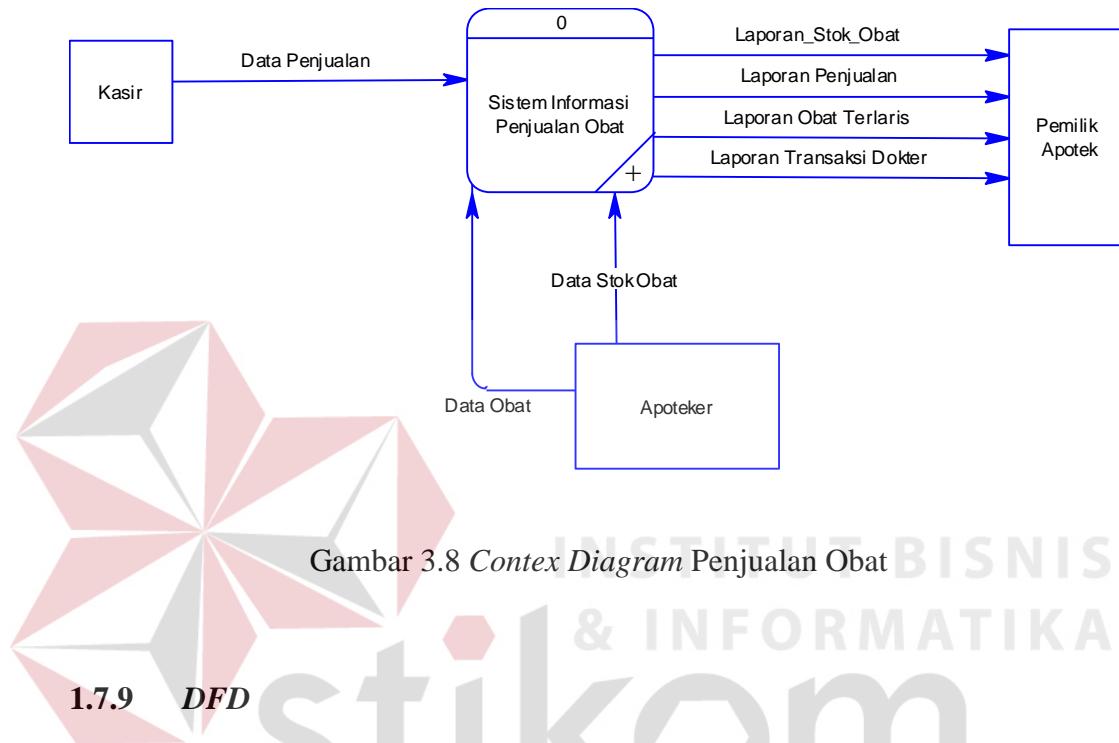
Gambar 3. 7 *System Flow Persediaan*

Pada gambar di atas proses pencatatan persedian yang tersistem dimulai dari *supplier* yang memberikan obat baru, lalu bagian apoteker akan langsung meng-input-kan data obat yang akan di simpan pada *Database* data obat dan stok obat dari proses *input* data obat akan mengeluarkan daftar persediaan obat.

### 1.7.8 Context Diagram

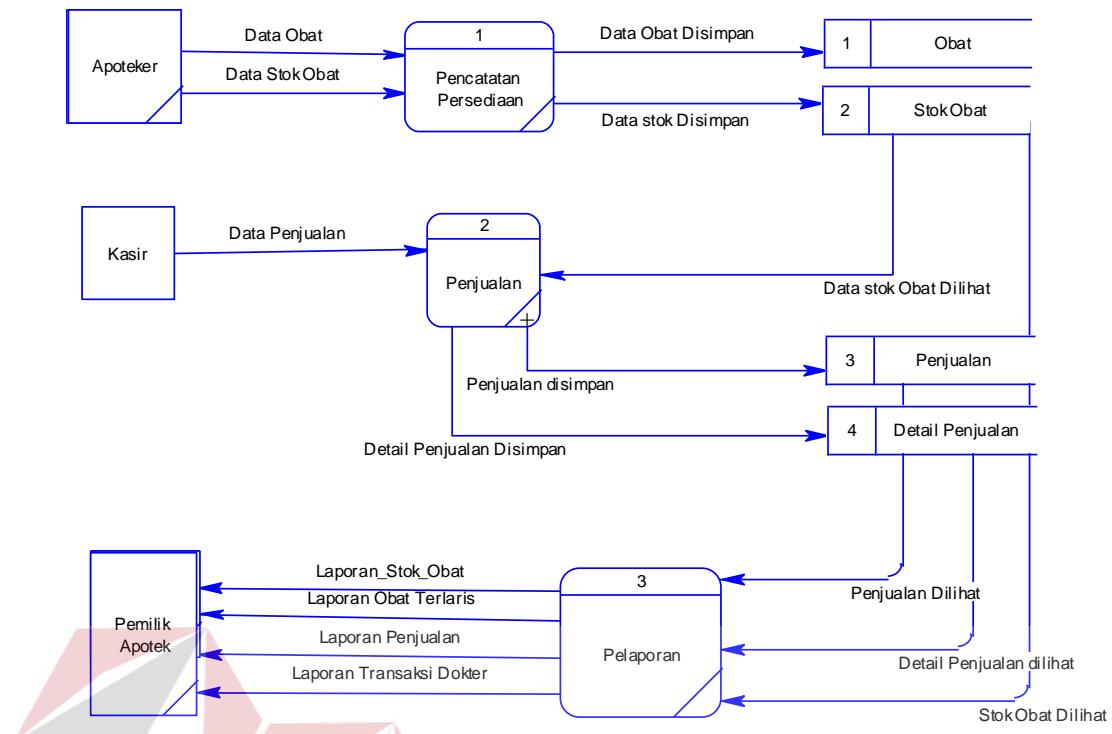
*Context diagram* merupakan gambaran dari DFD secara menyeluruh.

Pada *Context diagram* terdapat 3 eksternal yaitu kasir, apoteker, serta pemilik apotek. Tampilan *Contex diagram* dapat dilihat pada gambar 3.8.

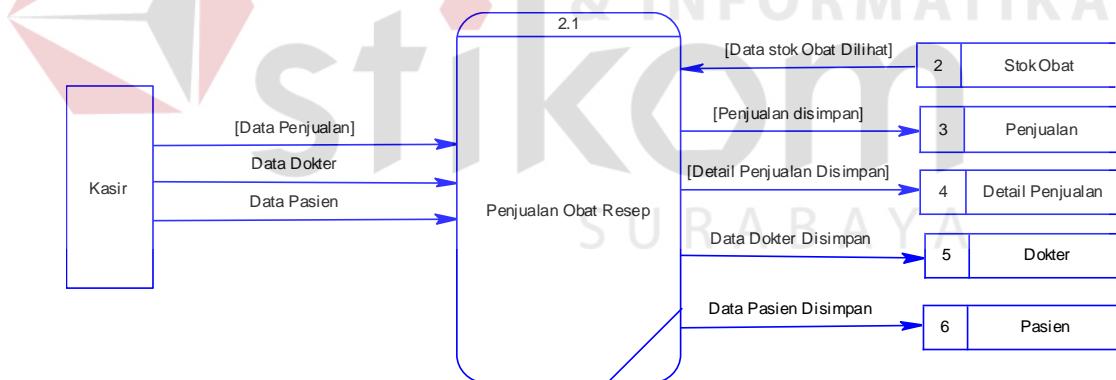


DFD Level 0 merupakan hasil penguraian dari *Contex diagram* yang memperlihatkan proses-proses utama yang ada dari sistem informasi yang dibuat.

### 1.7.9 DFD



Gambar 3.9 DFD Level 0 Penjualan Obat

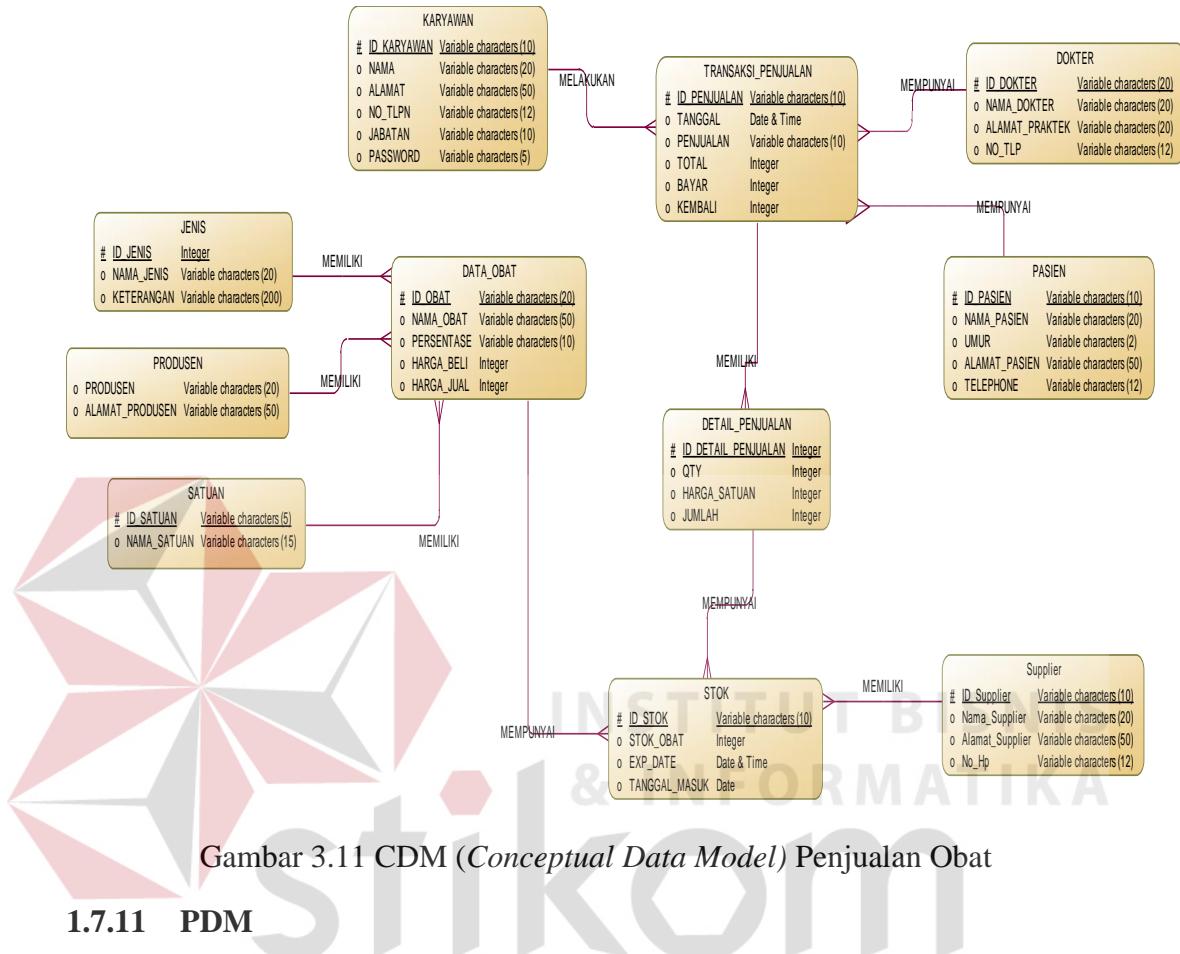


Gambar 3.10 DFD Level 1 Penjualan Obat Menggunakan Resep

### 1.7.10 CDM

CDM menggambarkan keseluruhan konsep struktur basis data yang dirancang untuk sistem informasi. Pada gambar 3.10 terdapat 7 entitas yang telah

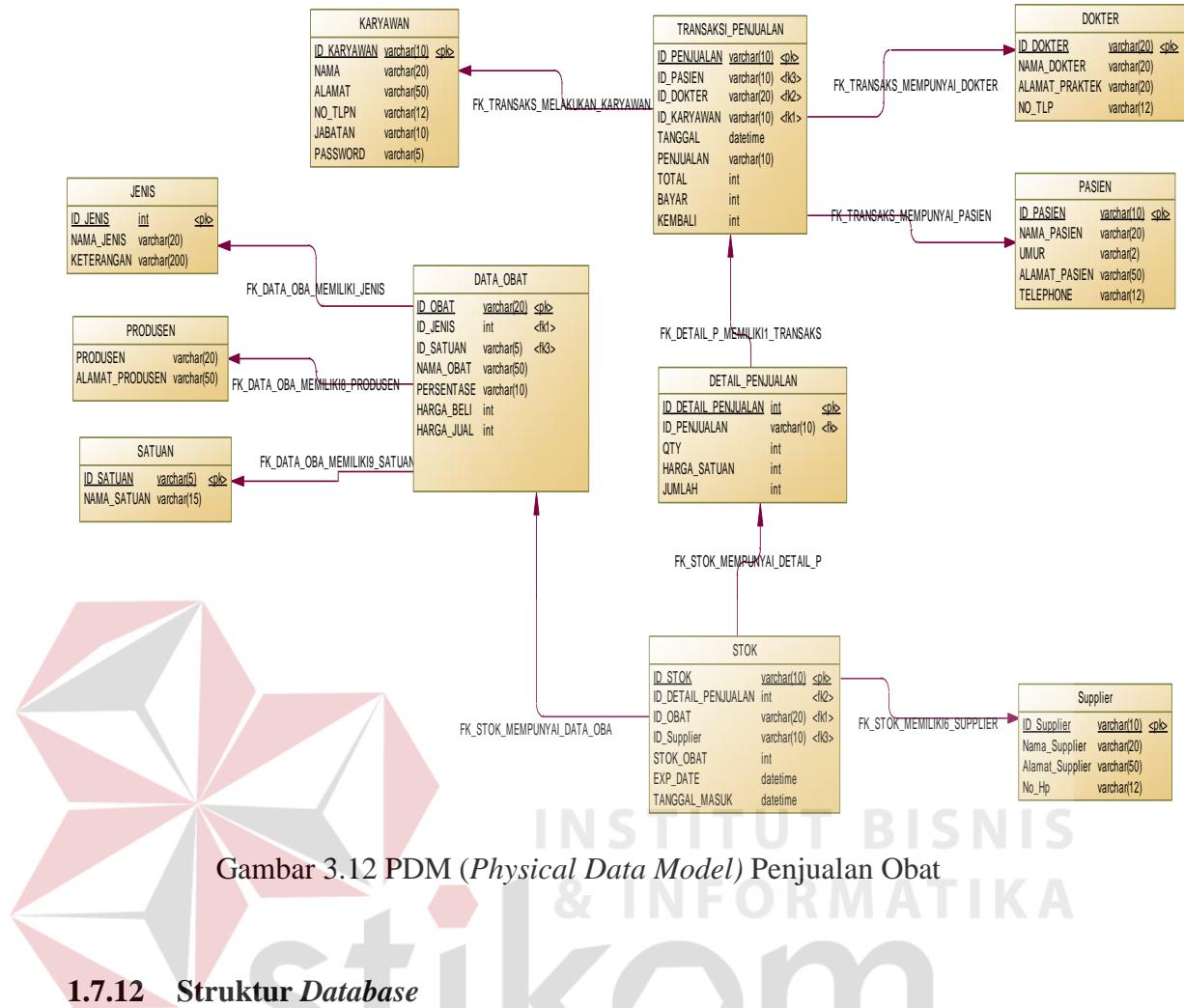
dirancang diantaranya yaitu data karyawan, data jenis obat, data obat, data dokter, transaksi penjualan, data detail penjualan dan data stok.



Gambar 3.11 CDM (Conceptual Data Model) Penjualan Obat

### 1.7.11 PDM

PDM menggambarkan secara *detail* konsep rancangan untuk struktur basis data pada sistem informasi yang dibuat. PDM merupakan hasil *generate* dari CDM sehingga tabel-tabel yang ada saling berhubungan antara satu dengan yang lain, tipe data pada setiap atribut yang dimiliki menjadi nampak begitu juga dengan *foreign key* (FK) yang menunjukkan antara entitas induk dengan entitas yang menggunakan turunannya. Hasil PDM dapat dilihat pada gambar 3.12 di bawah ini.



### 1.7.12 Struktur Database

Berikut adalah struktur tabel yang akan digunakan pada pembuatan aplikasi. Terdapat 7 tabel yaitu Karyawan, Obat, Jenis Obat, Dokter, Stok, Penjualan, Detail Penjualan. Penjelasan tabel-tabel tersebut dapat dilihat di bawah ini :

#### 1. Tabel Karyawan

*Primary Key* : ID\_KARYAWAN

*Foreign Key* : -

Fungsi : Menyimpan data karyawan

Tabel 3.8 Tabel Karyawan

| Nama Kolom  | Tipe Data | Panjang |
|-------------|-----------|---------|
| ID_KARYAWAN | Varchar   | 10      |
| NAMA        | Varchar   | 20      |
| ALAMAT      | Varchar   | 50      |
| NO_TLPN     | Varchar   | 12      |
| JABATAN     | Varchar   | 10      |
| PASSWORD    | Varchar   | 5       |

## 2. Tabel Obat

*Primary Key* : ID\_OBAT

*Foreign Key* : ID\_JENIS, ID\_SATUAN, PRODUSEN

Fungsi : Menyimpan data obat

Tabel 3.9 Tabel Obat

| Nama Kolom | Tipe Data | Panjang |
|------------|-----------|---------|
| ID_OBAT    | Varchar   | 20      |
| ID_JENIS   | Integer   | -       |
| NAMA_OBAT  | Varchar   | 20      |
| ID_SATUAN  | Varchar   | 5       |
| PRODUSEN   | Varchar   | 20      |
| PERSENTASE | Varchar   | 10      |
| HARGA_BELI | Integer   | -       |
| HARGA_JUAL | Integer   | -       |

## 3. Tabel Jenis Obat

*Primary Key* : ID\_JENIS

*Foreign Key* : -

Fungsi : Menyimpan data jenis obat

Tabel 3.10 Tabel Jenis Obat

| Nama Kolom | Tipe Data | Panjang |
|------------|-----------|---------|
| ID_JENIS   | Integer   | -       |
| JENIS_OBAT | Varchar   | 10      |
| KETERANGAN | Varchar   | 200     |

#### 4. Tabel Satuan

*Primary Key* : ID\_SATUAN

*Foreign Key* : -

Fungsi : Menyimpan data satuan

Tabel 3.11 Tabel Satuan

| Nama Kolom  | Tipe Data | Panjang |
|-------------|-----------|---------|
| ID_SATUAN   | Varchar   | 5       |
| NAMA_SATUAN | Varchar   | 10      |

#### 5. Tabel Produsen

*Primary Key* : PRODUSEN

*Foreign Key* : -

Fungsi : Menyimpan data produsen

Tabel 3.12 Tabel Produsen

| Nama Kolom      | Tipe Data | Panjang |
|-----------------|-----------|---------|
| PRODUSEN        | Varchar   | 20      |
| ALAMAT_PRODUSEN | Varchar   | 50      |

#### 6. Tabel Dokter

*Primary Key* : ID\_DOKTER

*Foreign Key* : -

Fungsi : Menyimpan data dokter

Tabel 3.13 Tabel Dokter

| Nama Kolom     | Tipe Data | Panjang |
|----------------|-----------|---------|
| ID_DOKTER      | Varchar   | 50      |
| NAMA_DOKTER    | Varchar   | 20      |
| ALAMAT_PRAKTEK | Varchar   | 20      |
| NO_TLP         | Varchar   | 12      |

### 7. Tabel *Supplier*

*Primary Key* : ID\_SUPPLIER

*Foreign Key* : -

Fungsi : Menyimpan data *supplier*

Tabel 3.14 Tabel *Supplier*

| Nama Kolom      | Tipe Data | Panjang |
|-----------------|-----------|---------|
| ID_SUPPLIER     | Varchar   | 10      |
| NAMA_SUPPLIER   | Varchar   | 20      |
| ALAMAT_SUPPLIER | Varchar   | 50      |
| NO_HP           | Varchar   | 12      |

### 8. Tabel Pasien

*Primary Key* : ID\_PASIEN

*Foreign Key* : -

Fungsi : Menyimpan data pasien

Tabel 3.15 Tabel Pasien

| Nama Kolom    | Tipe Data | Panjang |
|---------------|-----------|---------|
| ID_PASIEN     | Varchar   | 10      |
| NAMA_PASIEN   | Varchar   | 20      |
| UMUR          | Varchar   | 2       |
| ALAMAT_PASIEN | Varchar   | 50      |
| TELEPHONE     | Varchar   | 12      |

### 9. Tabel Stok

*Primary Key* : ID\_STOK

*Foreign Key* : ID\_OBAT

Fungsi : Menyimpan data stok obat

Tabel 3.16 Tabel Stok

| Nama Kolom | Tipe Data       | Panjang |
|------------|-----------------|---------|
| ID_STOK    | <i>Integer</i>  | -       |
| ID_OBAT    | <i>Varchar</i>  | 20      |
| STOK_OBAT  | <i>Integer</i>  | -       |
| EXP_DATE   | <i>Datetime</i> | -       |

## 10. Tabel Penjualan

*Primary Key* : ID\_PENJUALAN

*Foreign Key* : ID\_DOKTER, ID\_KARYAWAN

Fungsi : Menyimpan data penjualan

Tabel 3. 17 Tabel Penjualan

| Nama Kolom   | Tipe Data       | Panjang |
|--------------|-----------------|---------|
| ID_PENJUALAN | <i>Integer</i>  | -       |
| ID_DOKTER    | <i>Varchar</i>  | 50      |
| ID_KARYAWAN  | <i>Varchar</i>  | 10      |
| TANGGAL      | <i>Datetime</i> | -       |
| KETERANGAN   | <i>Varchar</i>  | 10      |
| PASIEN       | <i>Varchar</i>  | 10      |
| UMUR         | <i>Varchar</i>  | 2       |
| ALAMAT       | <i>Varchar</i>  | 50      |
| TPL          | <i>Varchar</i>  | 12      |
| TOTAL        | <i>Integer</i>  | -       |
| BAYAR        | <i>Integer</i>  | -       |
| KEMBALI      | <i>Integer</i>  | -       |

## 11. Tabel Detail Penjualan

*Primary Key* : ID\_DETAIL\_PENJUALAN

*Foreign Key* : ID\_STOK, ID\_PENJUALAN

Fungsi : Menyimpan *detail* penjualan

Tabel 3.18 Tabel Detail\_Penjualan

| Nama Kolom          | Tipe Data      | Panjang |
|---------------------|----------------|---------|
| ID_DETAIL_PENJUALAN | <i>Integer</i> | -       |
| ID_STOK             | <i>Integer</i> | -       |

| Nama Kolom   | Tipe Data      | Panjang |
|--------------|----------------|---------|
| ID_PENJUALAN | <i>Integer</i> | -       |
| QTY          | <i>Integer</i> | -       |
| HARGA_SATUAN | <i>Integer</i> | -       |
| JUMLAH       | <i>Integer</i> | -       |

### 1.7.13 Desain I/O

#### a. Desain form Login

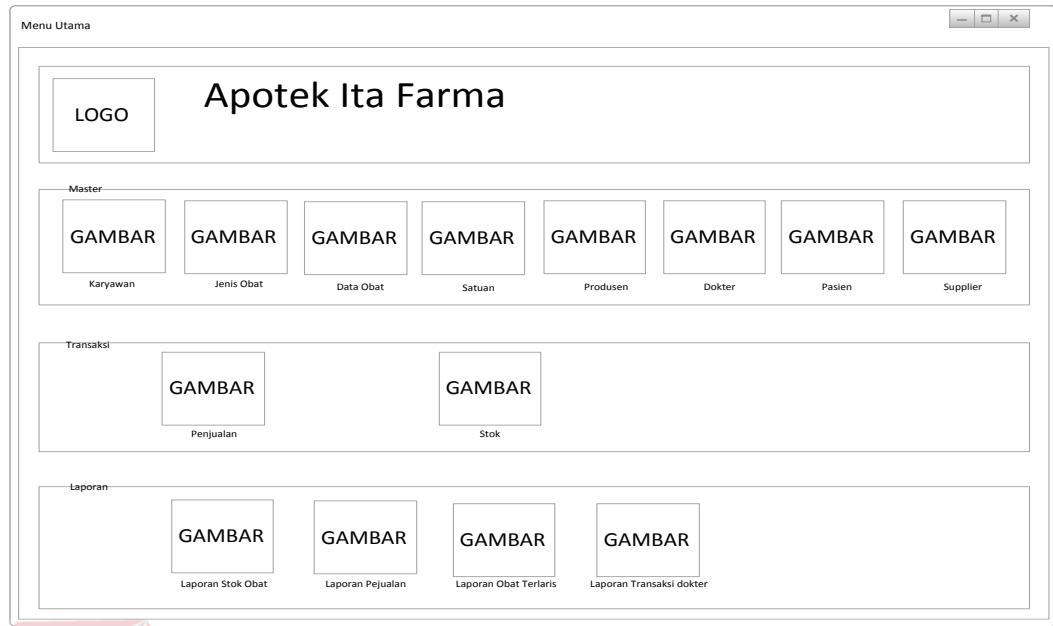
Berikut ini merupakan desain *form Login*, yang berfungsi sebagai validasi pengguna. Pengguna dapat mengakses sistem setelah melalui tahap *Login* dan telah terdaftar pada sistem.



Gambar 3.13 Desain *form Login*

#### b. Desain *form* Menu Utama

Desain *form* menu utama berfungsi sebagai tampilan utama yang nantinya digunakan sesuai hak akses masing-masing pengguna yang telah ditentukan sistem.



Gambar 3.14 Desain *form* Menu Utama

### c. Desain *form* Karyawan

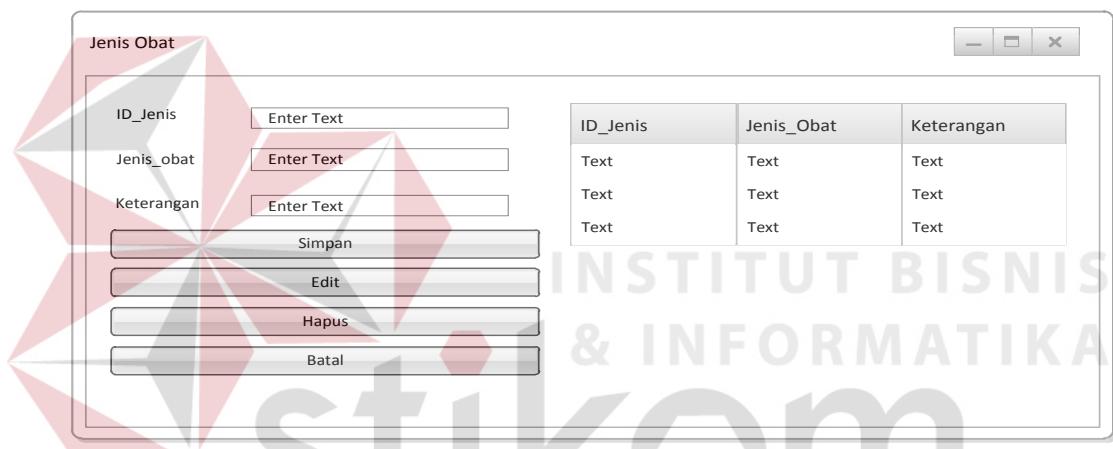
Desain *form* karyawan berfungsi sebagai menyimpan data karyawan yang terdiri dari ID\_Karyawan, Nama, Alamat, No. Tlp, Jabatan, *Password*. *Form* ini juga memiliki beberapa *button* yaitu simpan, *Edit*, Hapus, Batal. *Button* simpan digunakan untuk menyimpan data, *button edit* digunakan untuk mengedit data yang sudah ada, *button hapus* digunakan untuk menghapus data, *button batal* untuk membatalkan *inputan* atau diubah. *Datagridview* digunakan untuk menampilkan seluruh data karyawan yang telah tersimpan di dalam *database*.

The image shows a wireframe design of a 'Master Karyawan' form. On the left, there is a vertical stack of input fields for 'ID\_Karyawan', 'Nama', 'Alamat', 'No.Tlfn', 'Jabatan', and 'Password', each with an associated 'Enter Text' placeholder. To the right of these input fields is a 'Datagridview' table with columns labeled 'ID\_Karyawan', 'Nama', 'Alamat', 'No\_Tlfn', 'Jabatan', and 'Password'. Below the input fields and the table are four horizontal buttons: 'Simpan', 'Edit', 'Hapus', and 'Batal'.

Gambar 3.15 Desain *Form* Karyawan

d. Desain *form* Jenis Obat

Desain *form* jenis obat berfungsi sebagai menyimpan jenis obat yang terdiri dari ID\_jenis, Nama\_jenis, Keterangan. *Form* ini juga memiliki beberapa *button* yaitu Simpan, Edit, Hapus, Batal. *Button* Simpan digunakan untuk menyimpan data, *button* Edit digunakan untuk mengedit data yang sudah ada, *button* Hapus digunakan untuk menghapus data, *button* Batal untuk membatalkan *inputan* atau diubah. *Datagridview* digunakan untuk menampilkan seluruh data jenis obat yang telah tersimpan di dalam *database*.



Gambar 3.16 Desain *form* Jenis obat

e. Desain *form* Data Obat

Desain *form* data obat berfungsi sebagai menyimpan data obat yang terdiri dari ID\_Obat, Nama\_Obat, Bentuk, Satuan, Produsen, Persentase, Harga\_beli, Harga\_jual. *Form* ini juga memiliki beberapa *button* yaitu simpan, Edit, Hapus, Batal. *Button* simpan digunakan untuk menyimpan data, *button* Edit digunakan untuk mengedit data yang sudah ada, *button* hapus digunakan untuk menghapus data, *button* batal untuk membatalkan *inputan* atau diubah. *Datagridview* digunakan untuk menampilkan seluruh data obat yang telah tersimpan di dalam *database*.

| ID_Obat | Jenis_Obat | Nama_Obat | Ukuran | Produsen | Harga_Beli | Harga_Jual |
|---------|------------|-----------|--------|----------|------------|------------|
| Text    | Text       | Text      | Text   | Text     | Text       | Text       |

Gambar 3.17 Desain *form* Data obat

#### f. Desain *form* Satuan

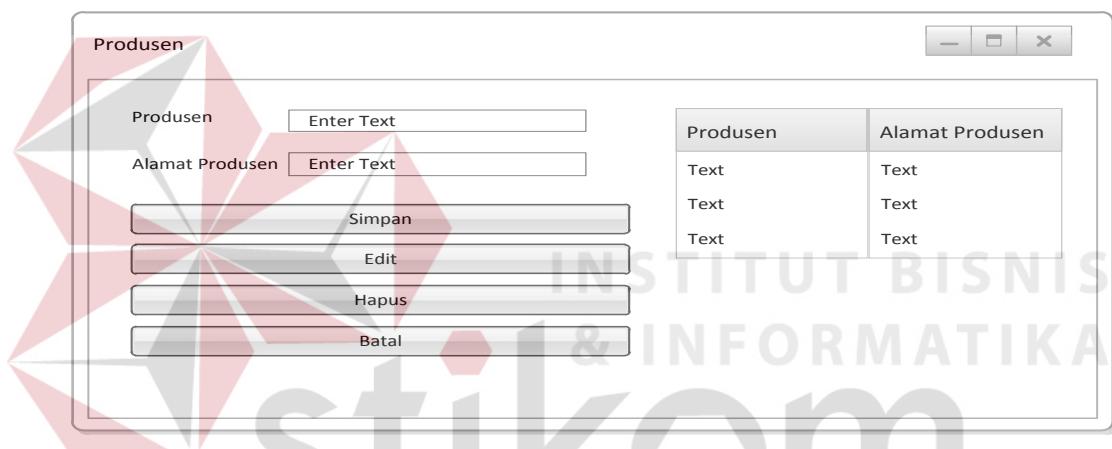
Desain *form* satuan berfungsi sebagai menyimpan data satuan obat yang terdiri dari ID\_Satuan, Nama\_Satuan. *Form* ini juga memiliki beberapa *button* yaitu simpan, *Edit*, Hapus, Batal. *Button* simpan digunakan untuk menyimpan data, *button* *Edit* digunakan untuk mengedit data yang sudah ada, *button* hapus digunakan untuk menghapus data, *button* batal untuk membatalkan *inputan* atau diubah. *Datagridview* digunakan untuk menampilkan seluruh data satuan obat yang telah tersimpan di dalam *database*.

| ID_Satuan | Nama Satuan |
|-----------|-------------|
| Text      | Text        |
| Text      | Text        |
| Text      | Text        |

Gambar 3.18 Desain *form* Satuan

g. Desain *form* Produsen

Desain *form* produsen berfungsi sebagai menyimpan data produsen yang terdiri dari Produsen, Alamat\_produsen. *Form* ini juga memiliki beberapa *button* yaitu simpan, *Edit*, Hapus, Batal. *Button* simpan digunakan untuk menyimpan data, *button Edit* digunakan untuk mengedit data yang sudah ada, *button* hapus digunakan untuk menghapus data, *button* batal untuk membatalkan *inputan* atau diubah. *Datagridview* digunakan untuk menampilkan seluruh data produsen yang telah tersimpan di dalam *database*.



Gambar 3.19 Desain *form* Produsen

h. Desain *form* Supplier

Desain *form supplier* berfungsi sebagai menyimpan data *Supplier* yang terdiri dari ID\_Supplier, Nama\_Supplier, Alamat\_Supplier, No\_Hp. *Form* ini juga memiliki beberapa *button* yaitu simpan, *Edit*, Hapus, Batal. *Button* simpan digunakan untuk menyimpan data, *button Edit* digunakan untuk mengedit data yang sudah ada, *button* hapus digunakan untuk menghapus data, *button* batal untuk membatalkan *inputan* atau diubah. *Datagridview* digunakan untuk menampilkan seluruh data *supplier* yang telah tersimpan di dalam *database*.

Master Supplier

| ID_supplier | Enter Text | ID_Supplier | Nama_supplier | Alamat_supplier | No_Hp |
|-------------|------------|-------------|---------------|-----------------|-------|
| Text        | Text       | Text        | Text          | Text            | Text  |
| Text        | Text       | Text        | Text          | Text            | Text  |
| Text        | Text       | Text        | Text          | Text            | Text  |

**Simpan**

**Edit**

**Hapus**

**Batal**

Gambar 3.20 Desain *form Supplier*

### i. Desain *form* Pasien

Desain *form* Pasien berfungsi sebagai menyimpan data pasien yang terdiri dari ID\_Pasien, Nama\_Pasien, Umur, Alamat\_Pasien, No\_Tlp. *Form* ini juga memiliki beberapa *button* yaitu simpan, Edit, Hapus, Batal. *Button* simpan digunakan untuk menyimpan data, *button* Edit digunakan untuk mengedit data yang sudah ada, *button* hapus digunakan untuk menghapus data, *button* batal untuk membatalkan *inputan* atau diubah. *Datagridview* digunakan untuk menampilkan seluruh data pasien yang telah tersimpan di dalam *database*.

Pasien

| ID Pasien | Enter Text | ID Pasien | Nama Pasien | Umur | Alamat Pasien | No Hp |
|-----------|------------|-----------|-------------|------|---------------|-------|
| Text      | Text       | Text      | Text        | Text | Text          | Text  |
| Text      | Text       | Text      | Text        | Text | Text          | Text  |
| Text      | Text       | Text      | Text        | Text | Text          | Text  |

**Simpan**

**Edit**

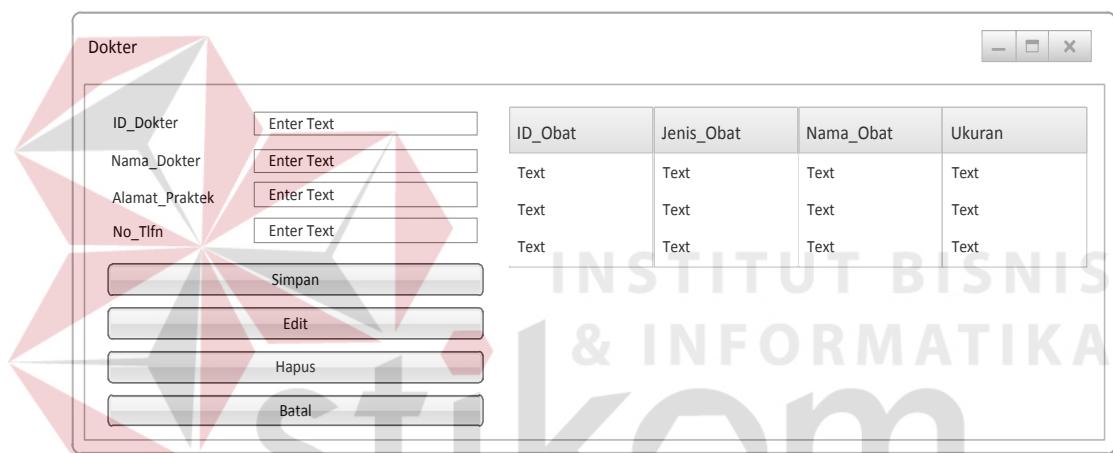
**Hapus**

**Batal**

Gambar 3.21 Desain *form* Pasien

#### j. Desain *form* Dokter

Desain *form* dokter berfungsi sebagai menyimpan data dokter yang terdiri dari ID\_dokter, Nama\_Dokter, Alamat\_Praktek, No\_Tlp. *Form* ini juga memiliki beberapa *button* yaitu simpan, Edit, Hapus, Batal. *Button* simpan digunakan untuk menyimpan data, *button* Edit digunakan untuk mengedit data yang sudah ada, *button* hapus digunakan untuk menghapus data, *button* batal untuk membatalkan *inputan* atau diubah. *Datagridview* digunakan untuk menampilkan seluruh data dokter yang telah tersimpan di dalam *database*.



Gambar 3.22 Desain *form* Dokter

#### k. Desain *form* Penjualan

Desain *form* Penjualan berfungsi sebagai meng-*input*-kan data penjualan yang terdiri dari ID\_Penjualan, Tanggal, Dokter, Pasien, Nama\_Obat, Harga\_Satuan, Qty, Jumlah, Total Bayar Kembali. *Form* ini juga memiliki beberapa *button* yaitu tambah, simpan, batal. *Button* tambah digunakan untuk mengkan detail penjualan, *button* simpan digunakan untuk menyimpan data penjualan, *button* batal untuk membatalkan *inputan*. *Datagridview* digunakan untuk menampilkan seluruh data penjualan yang telah tersimpan di dalam *database*.

Transaksi Penjualan

| ID_Penjualan  | <input type="text"/>                | <b>Rp. 0,</b> |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
|---|-------------------------------------|---------------|--------|--|-----------|--------------|-----|--------|------|------|------|------|------|------|------|------|------|------|------|------|
| Tanggal   | <input type="text"/>                |               |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| Pakai Resep   | <input checked="" type="checkbox"/> |               |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| ID_Dokter   | <input type="text"/>                |               |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| Pasien  | <input type="text"/>                |               |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| Umur  | <input type="text"/>                |               |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| Alamat  | <input type="text"/>                |               |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| Tlp   | <input type="text"/>                |               |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| <table border="1"> <thead> <tr> <th>Nama_Obat</th> <th>Harga Satuan</th> <th>QTY</th> <th>Jumlah</th> </tr> </thead> <tbody> <tr> <td>Text</td> <td>Text</td> <td>Text</td> <td>Text</td> </tr> <tr> <td>Text</td> <td>Text</td> <td>Text</td> <td>Text</td> </tr> <tr> <td>Text</td> <td>Text</td> <td>Text</td> <td>Text</td> </tr> </tbody> </table> |                                     |               |        |  | Nama_Obat | Harga Satuan | QTY | Jumlah | Text |
| Nama_Obat   | Harga Satuan                        | QTY           | Jumlah |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| Text  | Text                                | Text          | Text   |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| Text  | Text                                | Text          | Text   |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| Text  | Text                                | Text          | Text   |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| <input type="button" value="Tambah"/><br><br><input type="button" value="Kembali"/>   |                                     |               |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| Total <input type="text"/><br>Bayar <input type="text"/><br>Kembali <input type="text"/>  |                                     |               |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |
| <input type="button" value="Simpan"/><br><input type="button" value="Batal"/>   |                                     |               |        |  |           |              |     |        |      |      |      |      |      |      |      |      |      |      |      |      |

Gambar 3.23 Desain form Penjualan

### 1. Desain form Stok

Desain *form* Stok berfungsi sebagai menyimpan data stok obat sesuai dengan tanggal kadaluarsa yang terdiri dari ID\_Stok, ID\_Obat, Stok obat, Exp.Date. *Form* ini juga memiliki beberapa *button* yaitu simpan, Edit, Hapus, Batal. *Button* simpan digunakan untuk menyimpan data, *button* Edit digunakan untuk mengedit data yang sudah ada, *button* hapus digunakan untuk menghapus data, *button* batal untuk membatalkan *inputan* atau diubah. *Datagridview* digunakan untuk menampilkan seluruh data stok obat yang telah tersimpan di dalam *database*.

| ID_Stok | Nama Obat | Stok Obat |
|---------|-----------|-----------|
| Text    | Text      | Text      |
| Text    | Text      | Text      |
| Text    | Text      | Text      |

Gambar 3.24 Desain form Stok

## m. Desain laporan Stok

| NO | Id_stock | Nama_obat | Stock | Exp.date |
|----|----------|-----------|-------|----------|
|    |          |           |       |          |
|    |          |           |       |          |
|    |          |           |       |          |
|    |          |           |       |          |

Gambar 3.25 Desain form laporan Stok

## n. Desain Laporan Penjualan

| ID_penjualan | Nama Dokter | Karyawan | Tanggal | Total |
|--------------|-------------|----------|---------|-------|
|              |             |          |         |       |
|              |             |          |         |       |
|              |             |          |         |       |
|              |             |          |         |       |

Gambar 3.26 Desain form laporan Penjualan

o. Desain Laporan Obat Terlaris

|                              |  |           |        |
|------------------------------|--|-----------|--------|
| LOGO APOTEK                  | <b>APOTEK ITA FARMA</b><br>Jl. Raya Dlem - Kwanyar |           |        |
| <b>LAPORAN OBAT TERLARIS</b> |  |           |        |
| No                           | ID_Obat  | Nama Obat | Jumlah |
|                              |  |           |        |

Gambar 3.27 Desain *form* Laporan Obat terlaris

p. Desain Laporan Transaksi Dokter

|                                 |  |       |  |
|---------------------------------|--|-------|--|
| LOGO APOTEK                     | <b>APOTEK ITA FARMA</b><br>Jl. Raya Dlem - Kwanyar |       |  |
| <b>LAPORAN TRANSAKSI DOKTER</b> |  |       |  |
| ID_Dokter                       | Nama Dokter  | Total |  |
|                                 |  |       |  |

Gambar 3.28 Desain *form* Laporan transaksi Dokter

### 1.7.14 Desain Uji Coba

Desain uji coba digunakan untuk menguji sistem dan memastikan kesesuaian hasil rancangan sistem telah sesuai dengan fungsi dan kebutuhan. Metode yang digunakan untuk pengujian adalah metode *black box testing*, berikut

tabel yang digunakan sebagai acuan pada tahap pengujian aplikasi Sistem Informasi Penjualan Obat pada Apotek Ita Farma.

### 1. Desain Uji coba *Form Log In*

Tabel 3.19 Desain Uji coba *Form Log In*

| <b>Objek Pengujian</b> |  | <i>Form Login</i>   |  |
|------------------------|--|---|--|
| <b>Keterangan</b>      |  | Mengetahui tampilan dan fungsi <i>form login</i> dapat berjalan dan menghasilkan <i>output</i> yang diharapkan. |  |
| <b>Test Case ID</b>    | <b>Tujuan</b>  | <b>Input</b>  | <b>Output yang Diharapkan</b>  |
| 1.                     | Menguji <i>textbox username, password, dan button login.</i> | Memasukkan karakter pada <i>textbox username, password, dan button login.</i>                                   | Karakter pada <i>textbox password</i> yang dimasukkan akan tampil dengan simbol.   |
| 2.                     | Validasi <i>username dan password.</i>                       | <i>Button login.</i>  | Jika <i>login</i> sesuai maka akan tampil sesuai dengan hak akses masing-masing pengguna, jika <i>username</i> atau <i>password</i> salah maka akan ada peringatan.<br>Menu utama sesuai dengan hak akses. |

### 2. Desain Uji coba *Form Menu Utama*

Tabel 3.20 Desain Uji coba *Form Menu Utama*

| <b>Objek Pengujian</b> |                                       | <i>Form Menu Utama</i>   |  |
|------------------------|---------------------------------------|--|--|
| <b>Keterangan</b>      |                                       | Mengetahui tampilan dan fungsi <i>form Menu Utama</i> dapat berjalan dan menghasilkan <i>output</i> yang diharapkan. |  |
| <b>Test Case ID</b>    | <b>Tujuan</b>                         | <b>Input</b>   | <b>Output yang Diharapkan</b>            |
| 1.                     | Menguji Fungsi gambar pada menu utama | Gambar Master Karyawan   | Menampilkan <i>form master pelanggan</i> |
|                        |                                       | Gambar master jenis  | Menampilkan <i>form master pegawai</i>   |
|                        |                                       | Gambar master obat   | Menampilkan <i>form master obat</i>      |

### 3. Desain Uji coba *Form Karyawan*

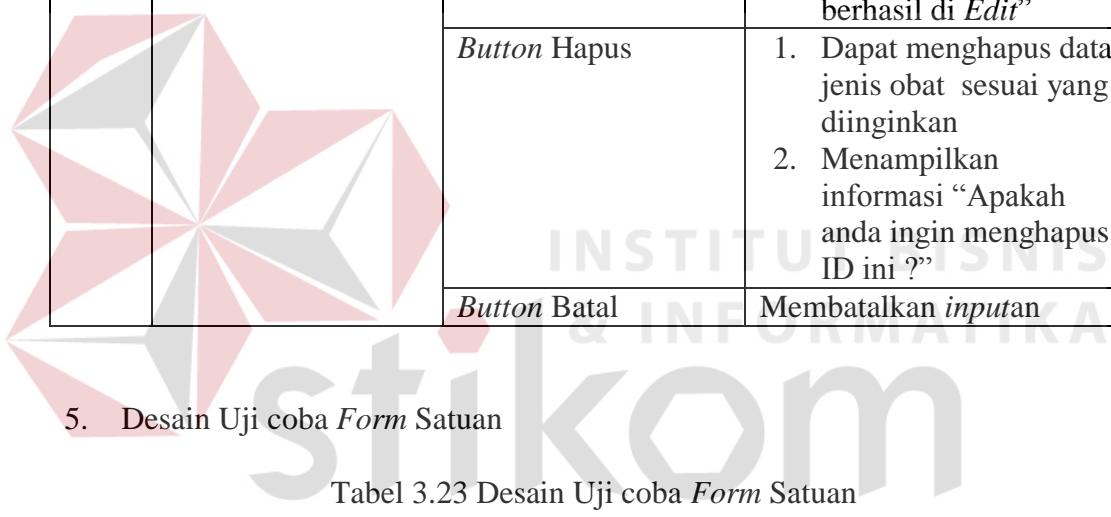
Tabel 3.21 Desain Uji coba *Form Karyawan*

| <b>Objek Pengujian</b> |  | <i>Form Karyawan</i>  |  |
|------------------------|--|---|--|
| <b>Keterangan</b>      |  | Mengetahui tampilan dan fungsi <i>form karyawan</i> dapat berjalan dan menghasilkan <i>output</i> yang diharapkan |  |
| <b>Test Case ID</b>    | <b>Tujuan</b>  | <b>Input</b>  | <b>Output yang Diharapkan</b>  |
| 1.                     | Menguji fungsi <i>button</i> , <i>textbox</i> , <i>datagridview</i> , <i>combobox</i> pada <i>form pegawai</i> | Gambar master karyawan  | Saat <i>form pegawai</i> tampil, <i>button simpan</i> , <i>ubah</i> , <i>hapus</i> , <i>batal</i> berfungsi.   |
|                        |  | <i>datagridview</i>   | Menampilkan data karyawan yang berhasil disimpan   |
|                        |  | <i>Button Simpan</i>  | <ol style="list-style-type: none"> <li>1. Menyimpan data karyawan sesuai yang diinginkan</li> <li>2. Menampilkan informasi “data berhasil disimpan”</li> </ol>                     |
|                        |  | <i>Button Edit</i>  | <ol style="list-style-type: none"> <li>1. Mengubah data karyawan sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Data berhasil di <i>Edit</i>”</li> </ol>                |
|                        |  | <i>Button Hapus</i>   | <ol style="list-style-type: none"> <li>1. Dapat menghapus data karyawan sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Apakah anda ingin menghapus ID ini ?”</li> </ol> |
|                        |  | <i>Button Batal</i>   | Membatalkan <i>inputan</i>   |

### 4. Desain Uji coba *Form Jenis Obat*

Tabel 3.22 Desain Uji coba *Form Jenis Obat*

| <b>Objek Pengujian</b> |               | <i>Form Jenis Obat</i>  |                               |
|------------------------|---------------|---|-------------------------------|
| <b>Keterangan</b>      |               | Mengetahui tampilan dan fungsi <i>form jenis obat</i> dapat berjalan dan menghasilkan <i>output</i> yang dihasilkan |                               |
| <b>Test Case ID</b>    | <b>Tujuan</b> | <b>Input</b>  | <b>Output yang Diharapkan</b> |



|    |   |                          |  |
|----|---|--------------------------|--|
| 1. | Menguji fungsi <i>button</i> , <i>textbox</i> , <i>datagridview</i> , <i>combobox</i> pada <i>form</i> jenis obat | Gambar master jenis obat | Saat <i>form</i> jenis obat tampil, <i>button</i> simpan, ubah, hapus, batal berfungsi.  |
|    |   | <i>datagridview</i>      | Menampilkan data jenis obat yang berhasil disimpan   |
|    |   | <i>Button Simpan</i>     | <ol style="list-style-type: none"> <li>1. Menyimpan data Jenis obat sesuai yang diinginkan</li> <li>2. Menampilkan informasi “data berhasil disimpan”</li> </ol>                     |
|    |   | <i>Button Edit</i>       | <ol style="list-style-type: none"> <li>1. Mengubah data jenis obat sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Data berhasil di <i>Edit</i>”</li> </ol>                |
|    |   | <i>Button Hapus</i>      | <ol style="list-style-type: none"> <li>1. Dapat menghapus data jenis obat sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Apakah anda ingin menghapus ID ini ?”</li> </ol> |
|    |   | <i>Button Batal</i>      | Membatalkan <i>inputan</i>   |

### 5. Desain Uji coba *Form* Satuan

Tabel 3.23 Desain Uji coba *Form* Satuan

| <b>Objek Pengujian</b> |   | <i>Form</i> Satuan  |  |
|------------------------|---|---|--|
| <b>Keterangan</b>      |   | Mengetahui tampilan dan fungsi <i>form</i> satuan dapat berjalan dan menghasilkan <i>output</i> yang dihasilkan |  |
| <b>Test Case ID</b>    | <b>Tujuan</b>   | <b>Input</b>  | <b>Output yang Diharapkan</b>  |
| 1.                     | Menguji fungsi <i>button</i> , <i>textbox</i> , <i>datagridview</i> , <i>combobox</i> pada <i>form</i> satuan | Gambar Satuan   | Saat <i>form</i> satuan tampil, <i>button</i> simpan, ubah, hapus, batal berfungsi.  |
|                        |   | <i>datagridview</i>   | Menampilkan data satuan yang berhasil disimpan   |
|                        |   | <i>Button Simpan</i>  | <ol style="list-style-type: none"> <li>1. Menyimpan data satuan sesuai yang diinginkan</li> <li>2. Menampilkan informasi “data berhasil disimpan”</li> </ol> |
|                        |   | <i>Button Edit</i>  | <ol style="list-style-type: none"> <li>1. Mengubah data satuan</li> </ol>  |
|                        |   |   |  |

| <b>Objek Pengujian</b> |               | <i>Form</i> Satuan  |  |
|------------------------|---------------|---|--|
| <b>Keterangan</b>      |               | Mengetahui tampilan dan fungsi <i>form</i> satuan dapat berjalan dan menghasilkan <i>output</i> yang dihasilkan |  |
| <b>Test Case ID</b>    | <b>Tujuan</b> | <b>Input</b>  | <b>Output yang Diharapkan</b>  |
|                        |               |   | <ul style="list-style-type: none"> <li>sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Data berhasil di <i>Edit</i>”</li> </ul>  |
|                        |               | <i>Button Hapus</i>   | <ul style="list-style-type: none"> <li>1. Dapat menghapus data satuan sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Apakah anda ingin menghapus ID ini ?”</li> </ul> |
|                        |               | <i>Button Batal</i>   | Membatalkan <i>inputan</i>   |

## 6. Desain Uji coba *Form* Produsen

Tabel 3.24 Desain Uji coba *Form* Produsen

| <b>Objek Pengujian</b> |   | <i>Form</i> Produsen  |   |
|------------------------|---|---|---|
| <b>Keterangan</b>      |   | Mengetahui tampilan dan fungsi <i>form</i> produsen dapat berjalan dan menghasilkan <i>output</i> yang dihasilkan |   |
| <b>Test Case ID</b>    | <b>Tujuan</b>   | <b>Input</b>  | <b>Output yang Diharapkan</b>   |
| 1.                     | Menguji fungsi <i>button</i> , <i>textbox</i> , <i>datagridview</i> , <i>combobox</i> pada <i>form</i> produsen | <i>Button Simpan</i>  | <ul style="list-style-type: none"> <li>Saat <i>form</i> produsen tampil, <i>button</i> simpan, ubah, hapus, batal berfungsi.</li> </ul>                             |
|                        |   | <i>Button Edit</i>  | <ul style="list-style-type: none"> <li>Menampilkan data produsen yang berhasil disimpan</li> </ul>  |
|                        |   | <i>Button Batal</i>   | <ul style="list-style-type: none"> <li>1. Menyimpan data produsen sesuai yang diinginkan</li> <li>2. Menampilkan informasi “data berhasil disimpan”</li> </ul>      |
|                        |   | <i>Button Hapus</i>   | <ul style="list-style-type: none"> <li>1. Mengubah data produsen sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Data berhasil di <i>Edit</i>”</li> </ul> |
|                        |   | <i>Button Master</i>  | <ul style="list-style-type: none"> <li>Menampilkan data master produsen</li> </ul>  |

| <b>Objek Pengujian</b> |               | <i>Form Produsen</i>  |  |
|------------------------|---------------|---|--|
| <b>Keterangan</b>      |               | Mengetahui tampilan dan fungsi <i>form</i> produsen dapat berjalan dan menghasilkan <i>output</i> yang dihasilkan |  |
| <b>Test Case ID</b>    | <b>Tujuan</b> | <b>Input</b>  | <b>Output yang Diharapkan</b>  |
|                        |               | <i>Button Hapus</i>   | <ol style="list-style-type: none"> <li>1. Dapat menghapus data produsen sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Apakah anda ingin menghapus ID ini ?”</li> </ol> |
|                        |               | <i>Button Batal</i>   | Membatalkan <i>inputan</i>   |

#### 7. Desain Uji coba *Form Data Obat*

Tabel 3.25 Desain Uji coba *Form Data Obat*

| <b>Objek Pengujian</b> |  | <i>Form Data Obat</i>  |   |
|------------------------|--|--|---|
| <b>Keterangan</b>      |  | Mengetahui tampilan dan fingsi <i>form</i> Data obat dapat berjalan dan menghasilkan <i>output</i> yang diharapkan |   |
| <b>Test Case ID</b>    | <b>Tujuan</b>  | <b>Input</b>   | <b>Output yang Diharapkan</b>   |
| 1.                     | Menguji fungsi <i>button</i> , <i>textbox</i> , <i>datagridview</i> , <i>combobox</i> pada <i>form</i> pegawai | <i>Gambar master data obat</i>   | Saat <i>form</i> data obat tampil, <i>button</i> simpan, ubah, hapus, batal berfungsi.  |
|                        |  | <i>datagridview</i>  | Menampilkan data obat yang berhasil disimpan  |
|                        |  | <i>Button Simpan</i>   | <ol style="list-style-type: none"> <li>1. Menyimpan data obat sesuai yang diinginkan</li> <li>2. Menampilkan informasi “data berhasil disimpan”</li> </ol>      |
|                        |  | <i>Button Edit</i>   | <ol style="list-style-type: none"> <li>1. Mengubah data obat sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Data berhasil di <i>Edit</i>”</li> </ol> |
|                        |  | <i>Button Hapus</i>  | <ol style="list-style-type: none"> <li>1. Dapat menghapus data obat sesuai</li> </ol>   |

| <b>Objek Pengujian</b> |               | <i>Form Data Obat</i>  |  |
|------------------------|---------------|--|--|
| <b>Keterangan</b>      |               | Mengetahui tampilan dan fungsi <i>form</i> Data obat dapat berjalan dan menghasilkan <i>output</i> yang diharapkan |  |
| <b>Test Case ID</b>    | <b>Tujuan</b> | <b>Input</b>   | <b>Output yang Diharapkan</b>  |
|                        |               |  | <ul style="list-style-type: none"> <li>yang diinginkan</li> <li>2. Menampilkan informasi “Apakah anda ingin menghapus ID ini ?”</li> </ul> |
|                        |               | <i>Button Batal</i>  | Membatalkan <i>inputan</i>   |

### 8. Desain Uji coba *Form Dokter*

Tabel 3.26 Desain Uji coba *Form Dokter*

| <b>Objek Pengujian</b> |  | <i>Form Dokter</i>   |  |
|------------------------|--|--|--|
| <b>Keterangan</b>      |  | Mengetahui tampilan dan fungsi <i>form</i> Dokter dapat berjalan dan menghasilkan <i>output</i> yang diharapkan. |  |
| <b>Test Case ID</b>    | <b>Tujuan</b>  | <b>Input</b>   | <b>Output yang Diharapkan</b>  |
| 1.                     | Menguji fungsi <i>button</i> , <i>textbox</i> , <i>datagridview</i> , <i>combobox</i> pada <i>form</i> pegawai | <i>Gambar master Dokter</i>  | Saat <i>form</i> dokter tampil, <i>button</i> simpan, ubah, hapus, batal berfungsi.  |
|                        |  | <i>datagridview</i>  | Menampilkan data dokter yang berhasil disimpan   |
|                        |  | <i>Button Simpan</i>   | <ul style="list-style-type: none"> <li>1. Menyimpan data dokter sesuai yang diinginkan</li> <li>2. Menampilkan informasi “data berhasil disimpan”</li> </ul> |
|                        |  | <i>Button Edit</i>   | <ul style="list-style-type: none"> <li>1. Mengubah data dokter sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Data berhasil di Edit”</li> </ul>   |
|                        |  | <i>Button Hapus</i>  | <ul style="list-style-type: none"> <li>1. Dapat menghapus data dokter sesuai yang diinginkan</li> <li>2. Menampilkan</li> </ul>                              |

| <b>Objek Pengujian</b> | <i>Form Dokter</i>   |              |  |
|------------------------|--|--------------|--|
| <b>Keterangan</b>      | Mengetahui tampilan dan fungsi <i>form Dokter</i> dapat berjalan dan menghasilkan <i>output</i> yang diharapkan. |              |  |
| <b>Test Case ID</b>    | <b>Tujuan</b>  | <b>Input</b> | <b>Output yang Diharapkan</b>                    |
|                        |  |              | informasi “Apakah anda ingin menghapus ID ini ?” |
|                        | <i>Button Batal</i>  |              | Membatalkan <i>inputan</i>                       |

### 9. Desain Uji coba *Form Supplier*

Tabel 3.27 Desain Uji coba *Form Supplier*

| <b>Objek Pengujian</b> | <i>Form Supplier</i>  |                               |   |
|------------------------|---|-------------------------------|---|
| <b>Keterangan</b>      | Mengetahui tampilan dan fungsi <i>form supplier</i> dapat berjalan dan menghasilkan <i>output</i> yang diharapkan |                               |   |
| <b>Test Case ID</b>    | <b>Tujuan</b>   | <b>Input</b>                  | <b>Output yang Diharapkan</b>   |
| 1.                     | Menguji fungsi <i>button, textbox, datagridview, combobox</i> pada <i>form supplier</i>                           | <i>Gambar master supplier</i> | Saat <i>form supplier</i> tampil, <i>button simpan, ubah, hapus, batal</i> berfungsi.   |
|                        |   | <i>datagridview</i>           | Menampilkan data <i>supplier</i> yang berhasil disimpan   |
|                        |   | <i>Button Simpan</i>          | <ol style="list-style-type: none"> <li>1. Menyimpan data <i>supplier</i> sesuai yang diinginkan</li> <li>2. Menampilkan informasi “data berhasil disimpan”</li> </ol>                     |
|                        |   | <i>Button Edit</i>            | <ol style="list-style-type: none"> <li>1. Mengubah data <i>supplier</i> sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Data berhasil di <i>Edit</i>”</li> </ol>                |
|                        |   | <i>Button Hapus</i>           | <ol style="list-style-type: none"> <li>1. Dapat menghapus data <i>supplier</i> sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Apakah anda ingin menghapus ID ini ?”</li> </ol> |
|                        |   | <i>Button Batal</i>           | Membatalkan <i>inputan</i>  |

## 10. Desain Uji coba *Form* Pasien

Tabel 3.28 Desain Uji coba *Form* Pasien

| <b>Objek Pengujian</b> |   | <i>Form</i> Pasien  |  |
|------------------------|---|---|--|
| <b>Keterangan</b>      |   | Mengetahui tampilan dan fungsi <i>form</i> pasien dapat berjalan dan menghasilkan <i>output</i> yang diharapkan |  |
| <b>Test Case ID</b>    | <b>Tujuan</b>   | <b>Input</b>  | <b>Output yang Diharapkan</b>  |
| 1.                     | Menguji fungsi <i>button</i> , <i>textbox</i> , <i>datagridview</i> , <i>combobox</i> pada <i>form</i> pasien | Gambar master pasien  | Saat <i>form</i> pasien tampil, <i>button</i> simpan, ubah, hapus, batal berfungsi.  |
|                        |   | <i>datagridview</i>   | Menampilkan data pasien yang berhasil disimpan   |
|                        |   | <i>Button Simpan</i>  | <ol style="list-style-type: none"> <li>Menyimpan data pasien sesuai yang diinginkan</li> <li>Menampilkan informasi “data berhasil disimpan”</li> </ol>                     |
|                        |   | <i>Button Edit</i>  | <ol style="list-style-type: none"> <li>Mengubah data pasien sesuai yang diinginkan</li> <li>Menampilkan informasi “Data berhasil di <i>Edit</i>”</li> </ol>                |
|                        |   | <i>Button Hapus</i>   | <ol style="list-style-type: none"> <li>Dapat menghapus data pasien sesuai yang diinginkan</li> <li>Menampilkan informasi “Apakah anda ingin menghapus ID ini ?”</li> </ol> |
|                        |   | <i>Button Batal</i>   | Membatalkan <i>inputan</i>   |

## 11. Desain Uji coba *Form* Stok

Tabel 3.29 Desain Uji coba *Form* Stok

| <b>Objek Pengujian</b> |   | <i>Form</i> Stok   |   |
|------------------------|---|--|---|
| <b>Keterangan</b>      |   | Mengetahui tampilan dan fungsi <i>form</i> stok dapat berjalan dan menghasilkan <i>output</i> yang diharapkan. |   |
| <b>Test Case ID</b>    | <b>Tujuan</b>   | <b>Input</b>   | <b>Output yang Diharapkan</b>   |
| 1.                     | Menguji fungsi <i>button</i> , <i>textbox</i> , <i>datagridview</i> , <i>combobox</i> pada <i>form</i> stok | Gambar Stok  | Saat <i>form</i> stok tampil, <i>button</i> simpan, ubah, hapus, batal berfungsi. |
|                        |   | <i>datagridview</i>  | Menampilkan data stok   |

| <b>Objek Pengujian</b> |                  | <i>Form Stok</i>   |  |
|------------------------|------------------|--|--|
| <b>Keterangan</b>      |                  | Mengetahui tampilan dan fungsi <i>form stok</i> dapat berjalan dan menghasilkan <i>output</i> yang diharapkan. |  |
| <b>Test Case ID</b>    | <b>Tujuan</b>    | <b>Input</b>   | <b>Output yang Diharapkan</b>  |
|                        | <i>form Stok</i> |  | yang berhasil disimpan   |
|                        |                  | <i>Button Simpan</i>   | <ol style="list-style-type: none"> <li>1. Menyimpan data stok sesuai yang diinginkan</li> <li>2. Menampilkan informasi “data berhasil disimpan”</li> </ol>                     |
|                        |                  | <i>Button Edit</i>   | <ol style="list-style-type: none"> <li>1. Mengubah data stok sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Data berhasil di <i>Edit</i>”</li> </ol>                |
|                        |                  | <i>Button Hapus</i>  | <ol style="list-style-type: none"> <li>1. Dapat menghapus data stok sesuai yang diinginkan</li> <li>2. Menampilkan informasi “Apakah anda ingin menghapus ID ini ?”</li> </ol> |
|                        |                  | <i>Button Batal</i>  | Membatalkan <i>inputan</i>   |

## 12. Desain Uji coba *Form Penjualan*

Tabel 3.30 Desain Uji coba *Form Penjualan*

| <b>Objek Pengujian</b> |  | <i>Form Penjualan</i>   |  |
|------------------------|--|---|--|
| <b>Keterangan</b>      |  | Mengetahui tampilan dan fungsi <i>form penjualan</i> dapat berjalan dan menghasilkan <i>output</i> yang diharapkan. |  |
| <b>Test Case ID</b>    | <b>Tujuan</b>  | <b>Input</b>  | <b>Output yang Diharapkan</b>  |
| 1.                     | Menguji fungsi <i>button,datagridview</i> , pada <i>form penjualan</i> . | <i>Button Tambah</i>  | Menambahkan data obat pada <i>gridview detail obat</i> . Menampilkan pesan “ data berhasil ditambah” |
|                        |  | <i>Button Simpan</i>  | Menyimpan data penjualan serta menampilkan pesan “data berhasil disimpan”.                           |

### 13. Desain Uji coba *Form* Laporan

Tabel 3.31 Desain Uji coba Form Laporan

| <b>Objek Pengujian</b> | <i>Form</i> Laporan  |                 |                                      |
|------------------------|--|-----------------|--------------------------------------|
| <b>Keterangan</b>      | Menetahui tampilan dan fungsi <i>form</i> laporan dapat berjalan dan menghasilkan <i>output</i> yang diharapkan. |                 |                                      |
| <b>Test Case ID</b>    | <b>Tujuan</b>  | <b>Input</b>    | <b>Output yang Diharapkan</b>        |
| 1.                     | Menguji <i>form</i> Laporan Stok obat  | Tanggal Periode | Menampilkan laporan stok obat        |
| 2.                     | Menguji <i>form</i> Laporan penjualan  | Tanggal Periode | Menampilkan laporan penjualan        |
| 3.                     | Menguji <i>form</i> Laporan Obat terlaris  | Tanggal Periode | Menampilkan laporan obat terlaris    |
| 4.                     | Menguji <i>form</i> Laporan Transaksi Dokter   | Tanggal Periode | Menampilkan laporan Transaksi Dokter |

