

## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

#### **3.1 Analisis Permasalahan**

Transaksi pelayanan pasien pada Praktek Pribadi Dokter Gigi Yoyok dilakukan setiap hari diantaranya cek kesehatan gigi dan penjualan obat. Tidak jarang pasien tidak dapat dilayani dengan baik karena terlalu banyak pasien yang melakukan transaksi pelayanan pasien. Hal ini menyebabkan kepuasan pasien menurun yang pada akhirnya juga dapat mengurangi keuntungan Praktek Pribadi Dokter Gigi Yoyok.

Permasalahan yang dihadapi oleh praktek pribadi dokter gigi yoyok terkendala dalam pengaturan informasi pasien, rekam medis dan catatan administrasi tidak tercatat dengan baik sehingga admin mengalami kesusahan dalam mencari informasi tentang pasien. Dan pemberian obat, biasanya dokter akan memberi obat secara langsung ataupun dengan menggunakan resep..

Pada saat calon pasien datang ke tempat praktek akan diberikan form untuk mengisi data calon pasien, data yang diperoleh akan disalin pada komputer dan pasien mendapat kartu berobat. Dan kartu berobat digunakan untuk berobat. Semua data, transaksi sampai pembuatan laporan dimasukkan dalam buku-buku yang terpisah. Hal tersebut menjelaskan betapa pentingnya data pasien yang diperoleh pada saat melakukan pendaftaran, akan tetapi proses pengolahan data dengan sistem tersebut sangatlah rumit. Petugas harus menyalin berulang-ulang data pasien yang ada. Setelah melakukan pendaftaran pasien dapat melakukan pengobatan sesuai dengan keluhan. Dokter akan bertanya keluhan yang ada dan melakukan tindakan akan keluhan tersebut. Sehingga mendapat hasil data

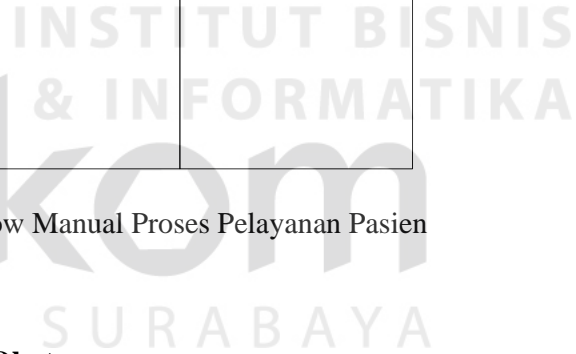
pelayanan pasien. Laporan hasil pemeriksaan dokter didapat dari data pelayanan pasien. Proses pengolahan pelayanan pasien akan dilakukan oleh dokter, dan petugas atau admin akan menyalin data pelayanan pasien pada sistem.

Hal ini dimaksudkan agar memudahkan dokter dalam pencarian data, namun pada kenyataannya malah mempersulit karena data yang begitu banyak harus dicari satu persatu dan akan memakan waktu yang banyak. Pencatatan pada buku-buku terpisah itu memang lebih memudahkan untuk mencari data-data yang diperlukan. Adakalanya pada saat dokter memasukkan data atau melakukan perhitungan biaya tidak sengaja melakukan kesalahan, maka harus melakukan perhitung ulang, Dan tentu saja hal ini memakan waktu yang tidak sebentar.

Berdasarkan uraian di atas maka diperlukan rancang bangun sistem administrasi pasien yang dapat mempermudah kinerja dokter. Dengan adanya sistem administrasi pada praktek pribadi dokter gigi tersebut diharapkan dapat membantu proses input data, perhitungan transaksi dan pembuatan laporan.

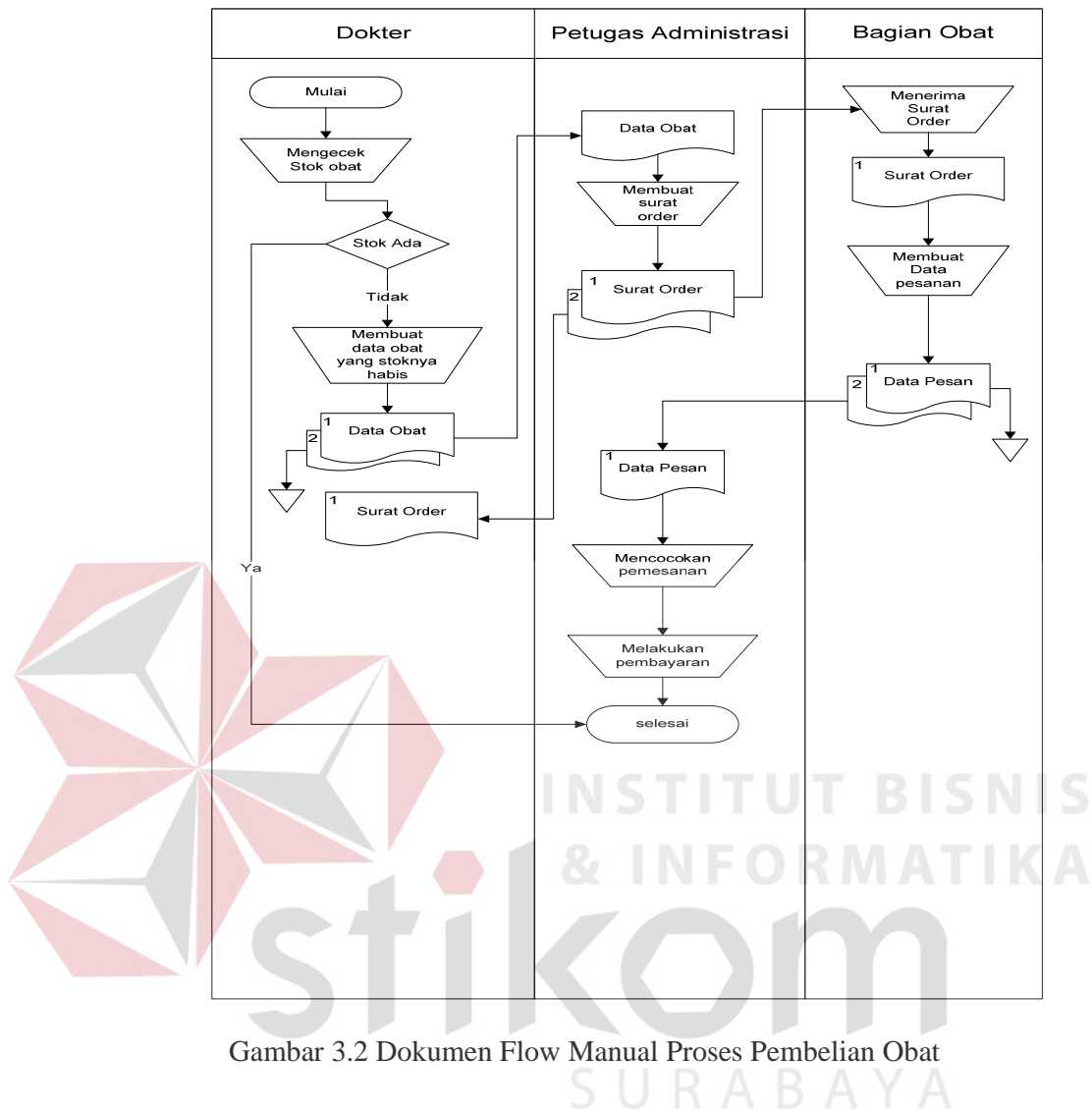
### **3.1.1 Dokumen Flow Pelayanan Pasien**

*Document flow* proses transaksi pelayanan pasien dimulai ketika Pasien melakukan pendaftaran. Kemudian data pasien diberikan pada bagian administrasi untuk cek pasien tersebut baru atau lama dan bagian administrasi mencatat data pasien. Setelah bagian administrasi melakukan cek data pasien maka pasien dapat melakukan pemeriksaan. Jika sudah sesuai maka dilakukan transaksi pembayaran.



### 3.1.2 Document Flow Pembelian Obat

*Document flow* pembelian obat dilakukan ketika dokter membuat surat order pemesanan kemudian memberikan kepada petugas. Pihak dokter membuat daftar pesanan untuk diberikan kepada Petugas. Kemudian petugas melakukan cek data obat dengan surat order, jika sesuai maka petugas melakukan cek kembali surat order dan data pesanan. Jika sesuai maka data tersebut dicatat dalam data pembelian obat.



Gambar 3.2 Dokumen Flow Manual Proses Pembelian Obat

### 3.2 Perancangan Sistem

Dalam pembuatan perancangan dan design digunakan model – model yang telah ada. Model – model tersebut antara lain dokumen flow, data flow diagram satupun perancangan hubungan relasi antara tabel. Tahap – tahap yang digunakan dalam mendesign rancang bangun sistem administrasi pasien pada praktek pribadi dokter gigi. adalah :

1. Membuat Dokumen Flow Pelayanan pasien dan pembelian obat.

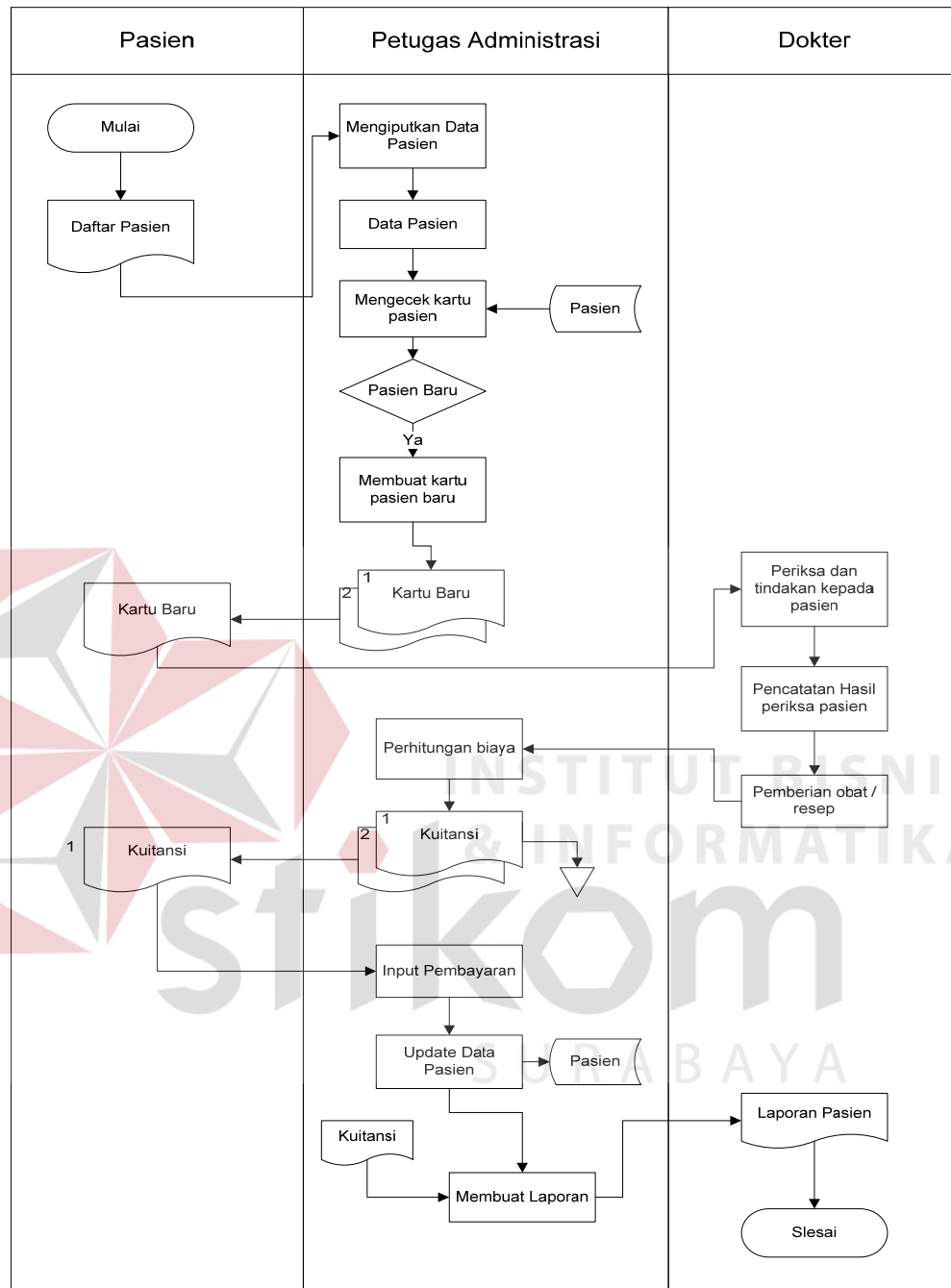
2. Membuat data flow diagram
3. Membuat diagram berjenjang.
4. Membuat rancangan hubungan relasional antara entitas atau ERD (*Entity Relationship Diagram*).
5. Struktur Database

### 3.2.1 Document Flow

*Document flow* ini akan dijelaskan alur proses-proses yang terdapat pada Rancang Bangun Sistem Administrasi Pasien pada Praktek Pribadi Dokter gigi. *Document flow* terkomputerisasi berikut terdapat 2 (dua) macam yaitu *document flow* pelayanan pasien terkomputerisasi dan *document flow* pembelian obat terkomputerisasi. Dimana masing-masing terdapat beberapa proses dari sistem sebagai berikut:

#### A. Dokumen Flow Pelayanan Pasien Terkomputerisasi

Proses transaksi pelayanan pasien yang dilakukan oleh bagian administrasi. Seperti terdapat pada Gambar 3.3 Bagian administrasi menerima data pasien dari pasien. Bagian administrasi dapat melakukan pelayanan pasien, jika data telah lengkap, dan proses transaksi pelayanan pasien dapat dilakukan penyimpanan ke dalam *database*.

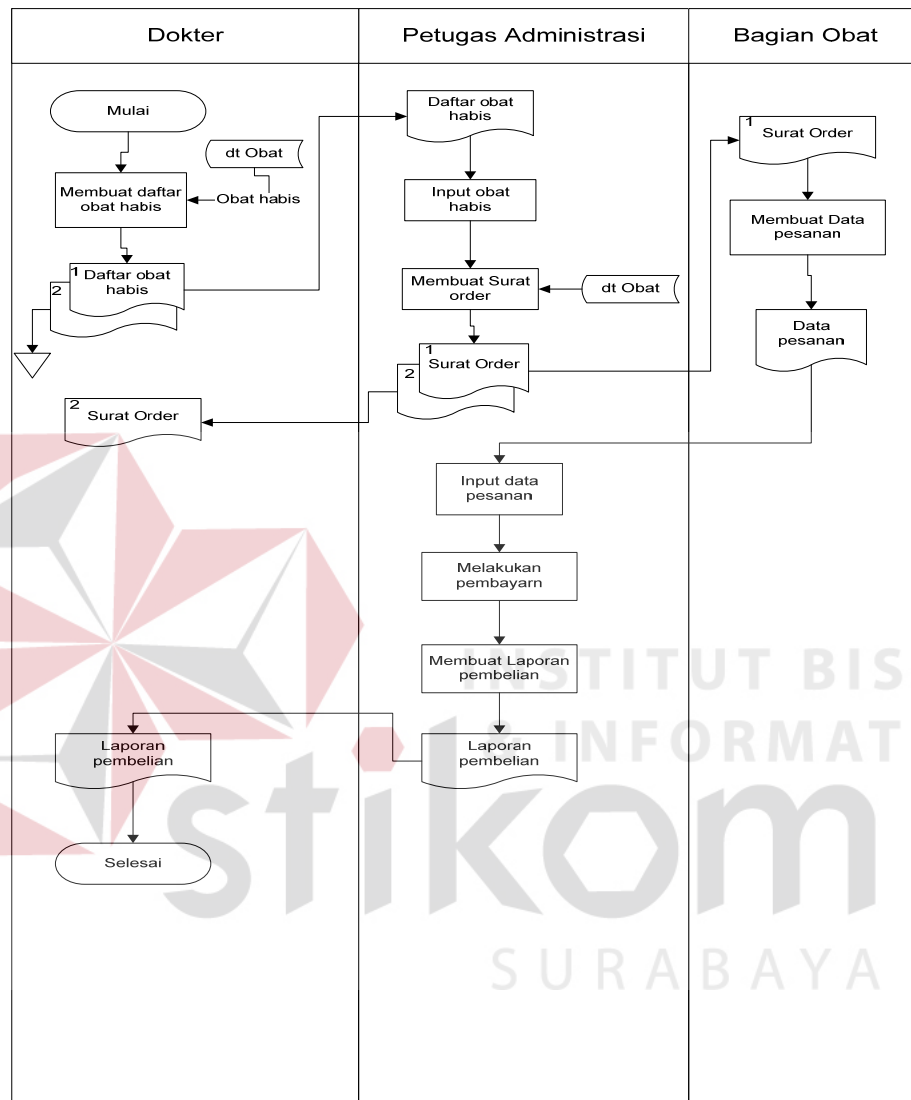


Gambar 3.3 Dokumen Flow Terkomputerisasi Proses Pelayanan Pasien

## B. Dokumen Flow Pembelian Obat Terkomputerisasi

Proses transaksi pembelian obat yang dilakukan oleh petugas. Seperti terdapat pada Gambar 3.4 Petugas menerima *stock* obat yang habis dari dokter.

Petugas dapat melakukan pembelian obat, jika data obat yang tersedia kurang, proses transaksi pembelian obat dapat dilakukan penyimpanan ke dalam *database*.



Gambar 3.4 Dokumen Flow Terkomputerisasi Proses Pembelian Obat

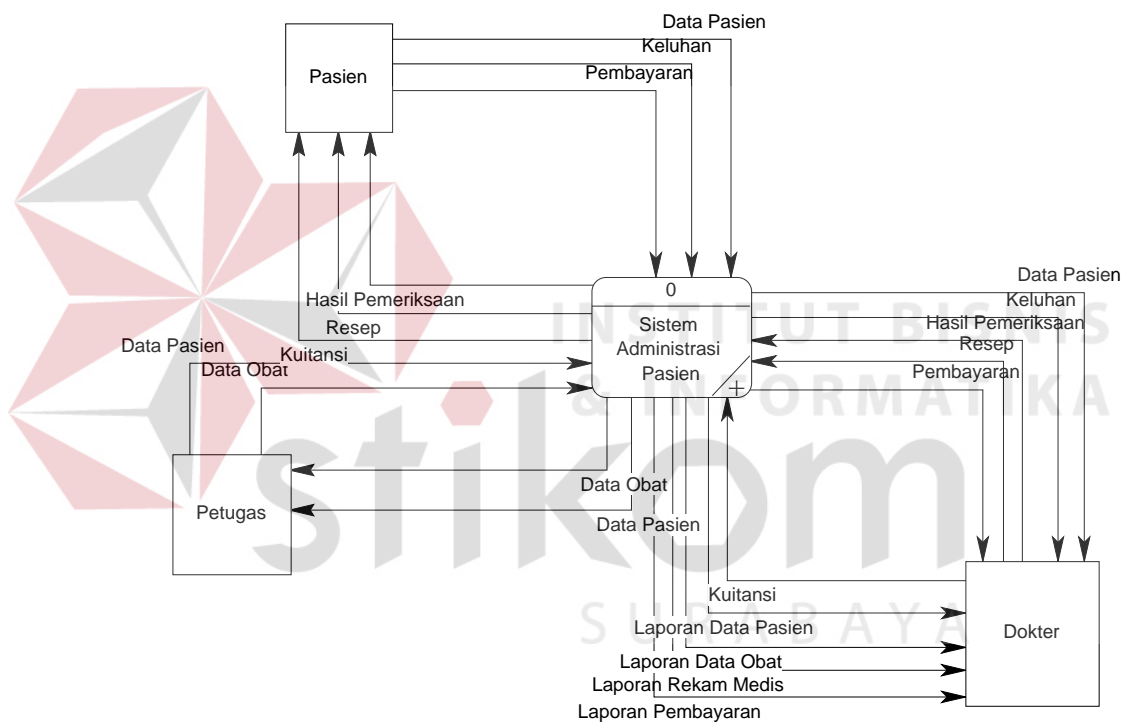
### 3.3 Data Flow Diagram

*Data Flow Diagram* (DFD) adalah gambaran aliran informasi yang terlibat dalam suatu prosedur (event) yang terdapat dalam suatu sistem. Diagram ini

menjelaskan alur data yang terjadi pada setiap proses. Adapun penjelasan dari DFD dapat dilihat sebagai berikut :

### 3.3.1 Context Diagram

*Context Diagram* merupakan diagram pertama dalam rangkaian dari suatu DFD yang menggambarkan entitas – entitas yang berhubungan dengan suatu sistem.



Gambar 3.5 Context Diagram Rancang Bangun Sistem Administrasi Pasien pada Praktek Pribadi Dokter Gigi

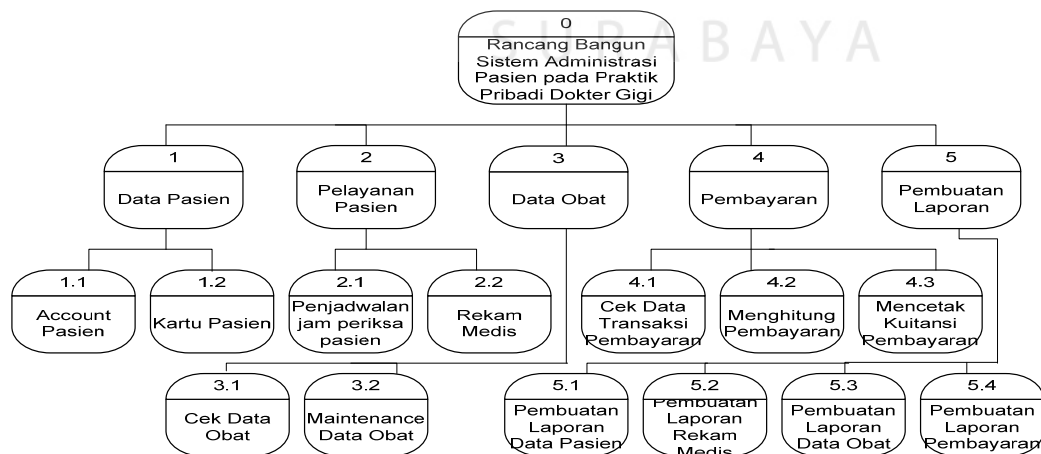
Pada Gambar 3.5 menjelaskan bahawa context diagram sistem informasi penjualan dan pembelian terdapat tiga entitas yaitu: pasien, dokter dan petugas. Masing–masing dari entity tersebut memberikan input dan sistem mengeluarkan output yang berupa laporan atau lainnya. Pada proses rancang bangun sistem



administrasi pasien dimulai dari proses data pasien dari pasien, kemudian sistem melakukan cek data pasien yang mempunyai kartu periksa dan apabila pasien tersebut telah mempunyai kartu periksa maka akan terjadi proses pelayanan pasien. Sedangkan apabila pasien tersebut belum mempunyai kartu pasien maka sistem melakukan proses data pasien kepada pasien baru.

### 3.3.2 Hierarchy Input Proses Output

Diagram berjenjang merupakan alat perancangan sistem yang dapat menampilkan seluruh proses yang terdapat pada suatu aplikasi tertentu dengan jelas dan terstruktur. Pada rancang bangun system administrasi pasien yang terdiri dari 5 proses utama yaitu proses data pasien, proses pelayanan pasien, proses data obat, proses pembayaran dan proses pembuatan laporan. Masing – masing dari proses utama tersebut akan dijabarkan kembali kedalam sub proses. Dari diagram berjenjang berikut ini akan terlihat dengan jelas masing – masing sub level dari *Data Flow Diagram* (DFD). Adapun penjelasan gambar diagram berjenjang dapat dilihat pada Gambar 3.6.



Gambar 3.6 Diagram Berjenjang Rancang Bangun Sistem Administrasi Pasien

### 3.3.3 Data Flow Diagram Level 0

Setelah *Context Diagram* sistem informasi penjualan dan pembelian didekomposisikan maka akan didapat DFD level 0 yang terdiri dari lima subproses, yaitu :

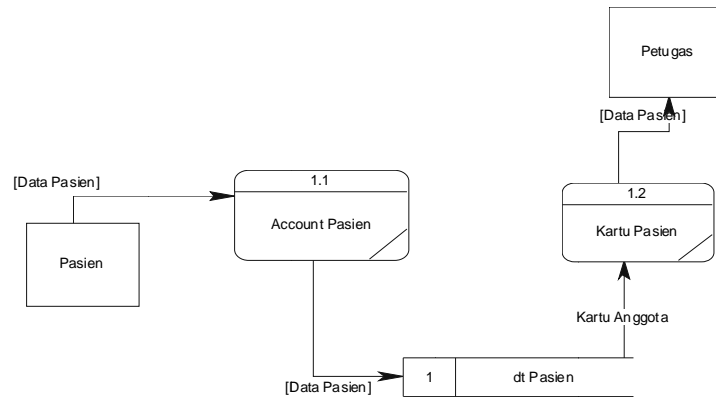
1. Proses Data Pasien. Proses ini digunakan untuk input data pasien. Dimana data-data ini digunakan sebagai masukan pada proses lainnya. Dalam proses ini yang terlibat entitas dokter, dan petugas.
2. Proses Pelayanan Pasien, entitas dalam proses ini adalah Pasien memberikan inputan berupa data pasien yang akan diperiksa. Dari data-data tersebut sistem memproses transaksi rekam medis ke pada dokter. Kemudian sistem memproses rekam medis dari dokter ke petugas, kemudian petugas melakukan pengurangan stock obat dan melakukan penambahan pada data pasien.
3. Proses Data Obat, proses berisi entitas data obat dimana petugas mengawasi keluar masuknya obat. Jika stok obat yang berada di gudang telah habis atau di bawah minimum stok maka petugas menginformasikan kepada dokter untuk melakukan order pembelian obat. Entitas yang lain adalah pasien yang akan menerima resep apabila obat yang diberikan tidak terdapat pada stok data obat.
4. Proses Pembayaran merupakan proses untuk mencetak pembayaran per transaksi pembayaran.
5. Proses Pelaporan Proses Pembuatan Laporan yaitu berupa Proses Laporan Data Pasien, Proses Rekam Medis, Proses Laporan Data Obat, dan Proses Laporan Pembayaran.

Penjelasan lebih lengkap mengenai DFD Level 0 dilihat pada Gambar 3.7



Berikut ini adalah DFD level 1 subproses *maintenance* data. DFD level 1 subproses data pasien terdapat dua proses, yaitu membuat account pasien dan membuat kartu pasien. Dalam proses membuat account pasien yang dimaksud adalah pengolahan data awal pasien untuk mengisi tabel master. Data-data master yang telah terbentuk tetap dapat dilakukan *update* sewaktu-waktu atas isinya melalui proses update data. Penjelasan lebih lanjut mengenai DFD Level 1 Subproses data pasien dapat dilihat pada Gambar 3.8

Berikut ini adalah DFD level 1 subproses *maintenance* data. DFD level 1 subproses data pasien terdapat dua proses, yaitu membuat account pasien dan membuat kartu pasien. Dalam proses membuat account pasien yang dimaksud adalah pengolahan data awal pasien untuk mengisi tabel master. Data-data master yang telah terbentuk tetap dapat dilakukan *update* sewaktu-waktu atas isinya melalui proses update data. Penjelasan lebih lanjut mengenai DFD Level 1 Subproses data pasien dapat dilihat pada Gambar 3.8

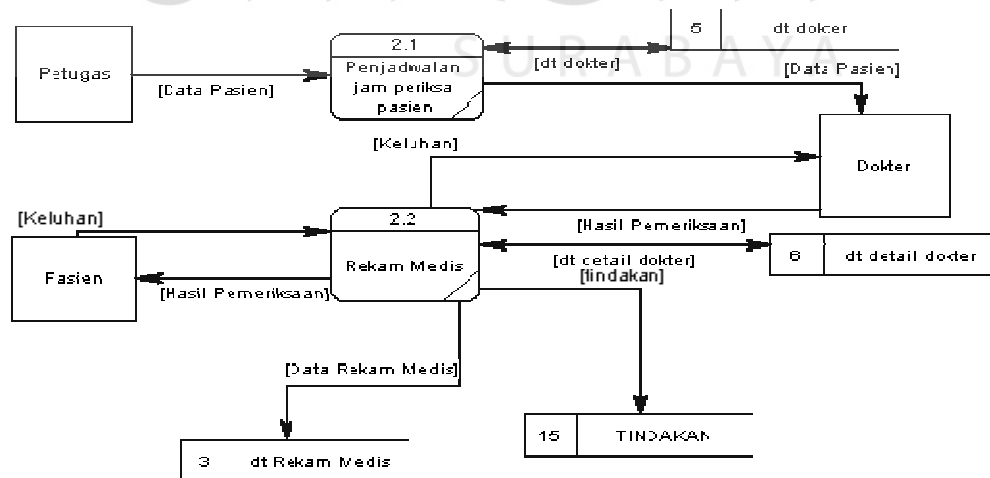


Gambar 3.8 DFD Level 1 Subproses Data Pasien

### 3.3.5 DFD Level 1 Subproses Pelayanan Pasien

Berikut ini merupakan subsistem pelayanan pasien. Terdapat dua proses di dalam subsistem transaksi pelayanan pasien yaitu:

1. Proses penjadwalan jam periksa pasien merupakan proses yang mencatat jadwal pasien periksa kemudian diberikan kepada dokter. Dalam proses ini terdapat proses appointment jadwal pasien.
2. Proses *input* rekam medis merupakan proses yang mengelola data transaksi rekam medis untuk diproses dan disimpan dalam *data store* data rekam medis.

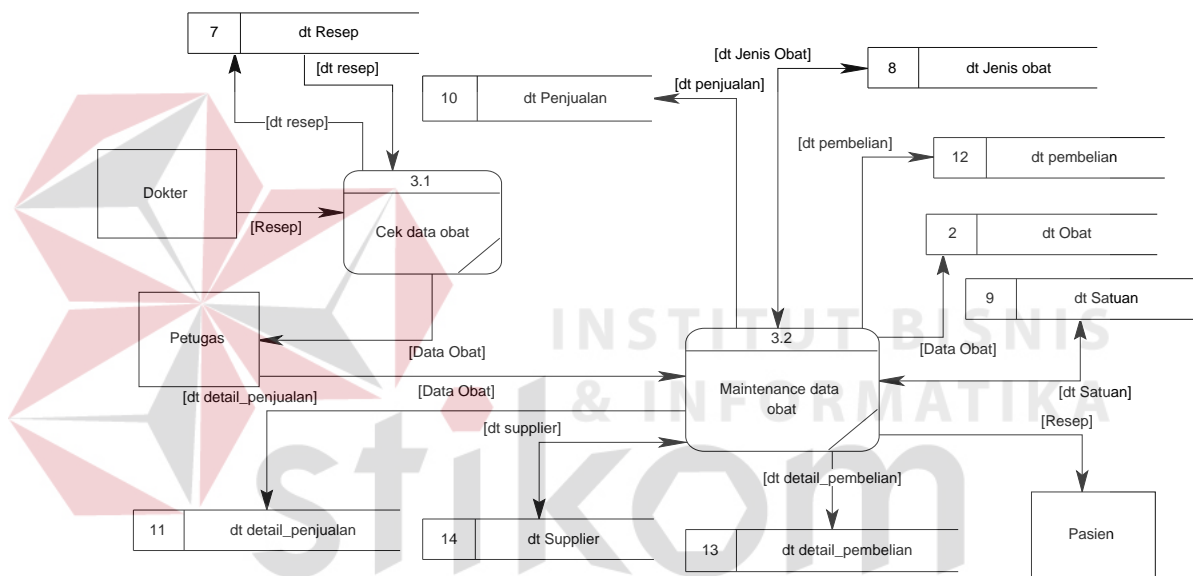


Gambar 3.9 DFD Level 1 Subproses Pelayanan Pasien

### 3.3.6 DFD Level 1 Subproses Data Obat

Berikut merupakan subsistem transaksi pembelian barang. Terdapat dua proses di dalam subsistem Data Obat yaitu:

1. Proses cek data obat merupakan proses yang mengecek kesamaan antara resep yang diberikan dokter dengan database obat.
2. Proses Maintenance data obat merupakan proses yang mengelola data transaksi obat untuk diproses dan disimpan dalam *data store* data obat.



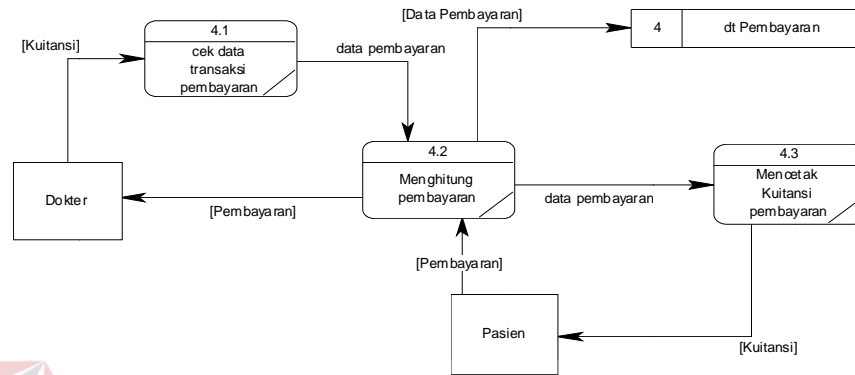
Gambar 3.10 DFD Level 1 Subproses Data Obat

### 3.3.7 DFD Level 1 Subproses Pembayaran

Gambar 3.11 merupakan subsistem pembuatan laporan. Terdapat 3 (tiga) proses di dalam subsistem pembayaran yaitu:

1. Proses cek data transaksi pembayaran merupakan proses cek kesamaan data pembayaran dengan data pesanan atau kuitansi.
2. Proses menghitung pembayaran merupakan proses yang menghitung jumlah hasil pemeriksaan pasien.

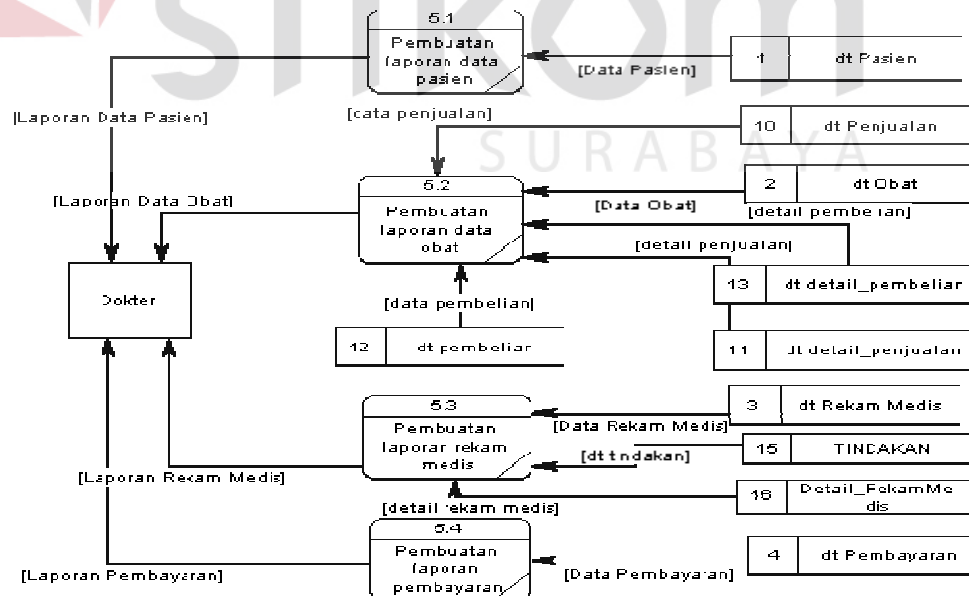
3. Proses mencetak kuitansi pembayaran merupakan proses yang menghitung jumlah pembayaran setelah proses transaksi pelayanan pasien dan data obat.



Gambar 3.11 DFD Level 1 Subproses Pembayaran

### 3.3.8 DFD Level 1 Subproses Pembuatan Laporan

Berikut ini adalah DFD level 1 sub proses pembuatan laporan yang terdiri dari empat proses yaitu pembuatan laporan data pasien, pembuatan laporan rekam medis, pembuatan laporan data obat, dan pembuatan laporan pembayaran.

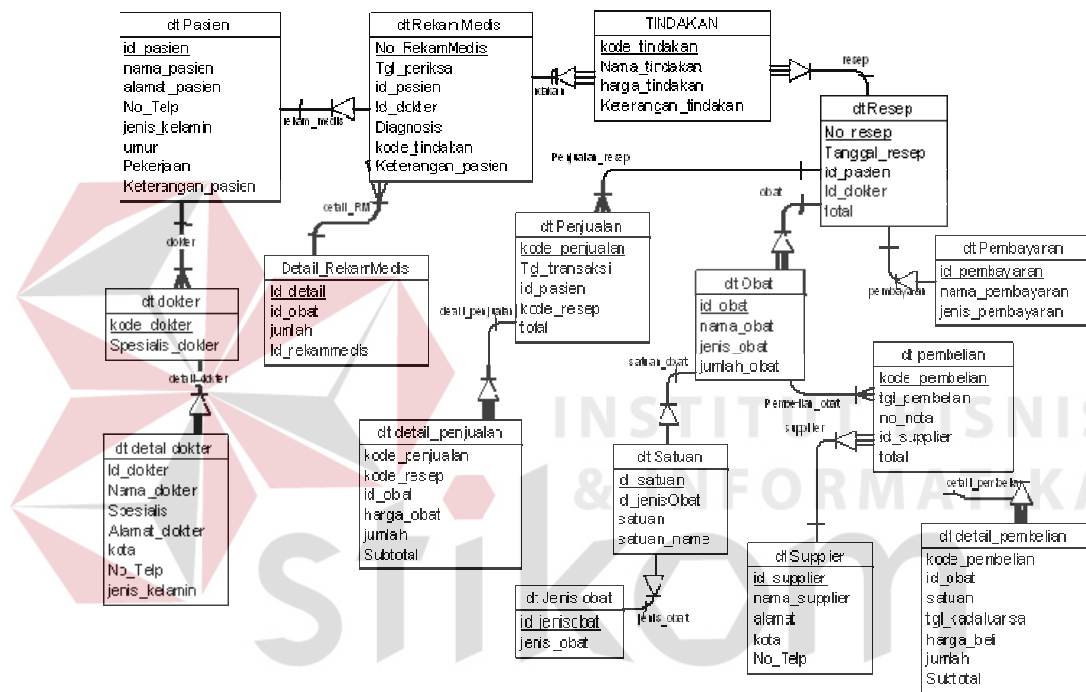


Gambar 3.12 DFD Level 1 Pembuatan Laporan

### 3.4 Pemodelan Basis Data

#### 3.4.1 Conceptual Data Model

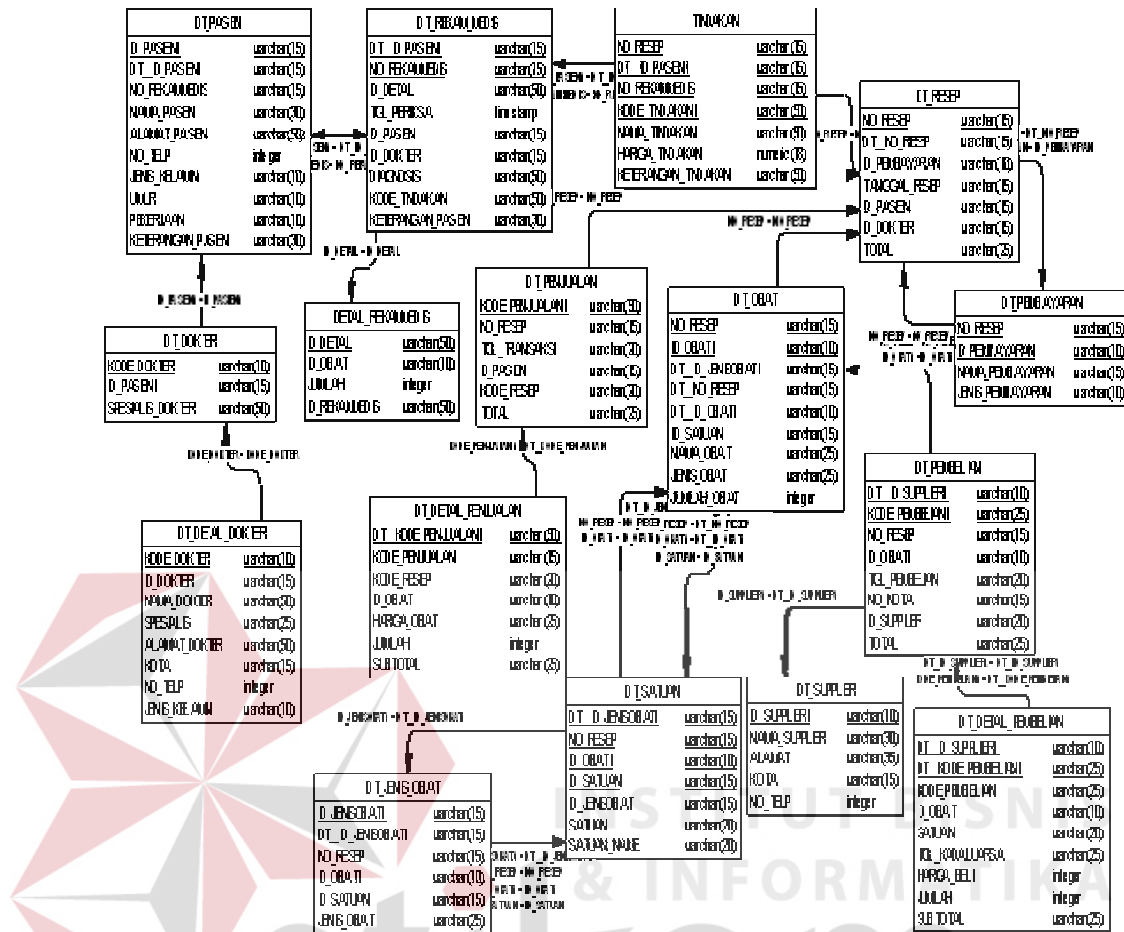
Sebuah Conceptual Data Model (CDM) menggambarkan secara keseluruhan konsep struktur basis data yang di rancang untuk suatu aplikasi sebagaimana terlihat pada gambar dibawah ini:



Gambar 3.13 CDM Rancang Bangun Sistem Administrasi Pasien pada Praktek Pribadi Dokter Gigi

#### 3.4.2 Physical Data Model

Sebuah Physical Data Model (PDM) menggambarkan secara detail konsep rancangan struktur basis data yang dirancang untuk suatu aplikasi. PDM merupakan hasil generate dari CDM. Pada PDM tergambar jelas tabel – tabel penyusun basis data beserta field – field yang terdapat pada setiap tabel sebagaimana terlihat pada gambar di bawah ini:



Gambar 3.14 PDM Rancang Bangun Sistem Administrasi Pasien pada Praktek Pribadi Dokter Gigi

### 3.4.3 Struktur Database

#### 1. Tabel Pasien

Primary\_Key : ID\_Pasien

Foreign\_Key : No\_RekamMedis

Fungsi : Untuk menyimpan Data Pasien

Tabel 3.1 Tabel Pasien

No	Field	Data Type	Lenght	Description
1.	ID_Pasien	Varchar	15	Identitas Pasien



No	Field	Data Type	Lenght	Description
2.	Nama_Pasien	Varchar	30	Nama Pasien
3.	Alamat	Varchar	50	AlamatPasien
4.	No_Telp	Varchar	15	Komunikasi
5.	Jenis_Kelamin	Varchar	15	Jenis Kelamin Pasien
6.	Umur	Varchar	10	Umur Pasien
7.	Pekerjaan	Varchar	10	Pekerjaan Pasien
8.	kota	Varchar	30	Kota pasien

## 2. Tabel Rekam Medis

Primary Key : No\_Rekammedis

Foreign Key : ID\_Pasien

Fungsi : Untuk menyimpan data rekam medis

Tabel 3.2 Tabel Rekam Medis

No	Field	Data Type	Length	Description
1.	ID_RekamMedis	Varchar	15	Identitas Rekam Medis
2.	Tgl_Periksa	DateTime		Tanggal Periksa Pasien
3.	Id_Pasien	Varchar	15	Identitas pasien
4.	Id_Dokter	Varchar	25	Identitas dokter
5.	Diagnosis	Varchar	30	Keterangan Penyakit Pasien
6.	Keterangan Pasien	Varchar	30	Keterangan Pasien Berobat

## 3. Tabel Obat

Primary\_Key : ID\_Obat

Foreign\_Key : ID\_Pasien

Fungsi : Untuk menyimpan Data Obat

Tabel 3.3 Tabel Data Obat

No	Field	Data Type	Lenght	Description
1.	ID_Obat	Varchar	10	Identitas Obat
2.	Nama_Obat	Varchar	25	Nama Obat
3.	Jenis_Obat	Varchar	25	Jenis Obat
4.	Status_Obat	Varchar	30	Status dari obat
5.	Satuan	Varchar	30	Satuan dari obat
6.	Jumlah_Obat	Integer		Jumlah Obat
7.	Harga_Beli	Integer		Harga beli dari obat
8.	Harga_Resep	Integer		Harga resep obat
9.	Harga	Integer		Harga obat
10.	Total_Jumlah	Integer		Total jumlah obat
11.	Safety_Stok	Varchar	25	Stok obat

## 4. Tabel Resep

Primary Key : No\_Resep

Foreign\_Key : -

Fungsi : Untuk Menyimpan data resep obat

Tabel 3.4 Tabel Resep

No	Field	Data Type	Length	Description
1.	No_Resep	Varchar	10	Nomer resep
2.	Tanggal_Resep	Varchar	15	Tanggal penulisan resep
3.	Id_Pasien	Varchar	15	Identitas pasien
4.	Id_Dokter	Varchar	20	Identitas dokter
5.	Total	Integer		Total keseluruhan

## 5. Tabel Detail\_Dokter

Primary Key : ID\_Dokter

Foreign\_Key : -

Fungsi : Untuk Menyimpan data dokter

Tabel 3.5 Tabel Dokter

No	Field	Data Type	Length	Description
1.	ID_Dokter	Varchar	20	Identitas dokter
2.	Nama_Dokter	Varchar	15	Nama dokter
3.	Spesialis	Varchar	15	Spesialis dokter
4.	Alamat	Varchar	50	Alamat dokter
5.	Kota	Varchar	30	Kota dokter
6.	No.Telp	Varchar	15	No telp dokter
7.	Jenis_kelamin	Varchar	15	Jenis kelamin dokter

## 6.Tabel Pembelian

Primary Key : Kode\_Pembelian

Foreign\_Key : -

Fungsi : Untuk Menyimpan data pembelian obat

Tabel 3.6 Tabel Pembelian

No	Field	Data Type	Length	Description
1.	Kode_pembelian	Varchar	10	Kode pembelian
2.	Tanggal_pembelian	Varchar	15	Tanggal pembelian
3.	No_nota	Varchar	15	Nomer nota pembelian
4.	Supplier	Varchar	20	Supplier obat
5.	Total	Integer		Total keseluruhan

## 7.Tabel Detail\_Pembelian

Primary Key : -

Foreign\_Key : -

Fungsi : Untuk Menyimpan data detail pembelian obat

Tabel 3.7 Tabel Detail\_Pembelian

No	Field	Data Type	Length	Description
1.	Kode_pembelian	Varchar	10	Kode pembelian
2.	Kode_barang	Varchar	15	Kode barang
3.	Satuan	Integer	15	Satuan obat

No	Field	Data Type	Length	Description
4.	Tanggal_kadaluarsa	Varchar	20	Tanggal kadaluarsa obat
5.	Harga_Beli	Varchar	20	Harga beli obat
6.	Jumlah	Integer		Jumlah obat
7.	Subtotal	Integer		Jumlah keseluruhan

#### 8. Tabel Penjualan

Primary Key : Kode\_Penjualan

Foreign\_Key : -

Fungsi : Untuk Menyimpan data penjualan obat

Tabel 3.8 Tabel Penjualan

No	Field	Data Type	Length	Description
1.	Kode_penjualan	Varchar	10	Nomer resep
2.	Tanggal_transaksi	Varchar	15	Tanggal transaksi
3.	Id_Pasien	Varchar	15	Identitas pasien
4.	resep	Varchar	20	Resep obat
5.	Total	Integer		Total keseluruhan

#### 9. Tabel Detail\_Penjualan

Primary Key : -

Foreign\_Key : kode\_penjualan

Fungsi : Untuk Menyimpan data detail penjualan obat

Tabel 3.9 Tabel Detail Penjualan

No	Field	Data Type	Length	Description
1.	Kode_penjualan	Varchar	10	Kode penjualan
2.	Kode_Resep	Varchar	15	Kode resep
3.	Kode_barang	Varchar	15	Kode barang
4.	Harga_Barang	Integer		Harga barang
5.	Jumlah	Integer		Jumlah penjualan
6.	Total	Integer		Total keseluruhan

## 10. Tabel Satuan

Primary Key : Id\_Satuan

Foreign\_Key : -

Fungsi : Untuk Menyimpan data satuan obat

Tabel 3.10 Tabel Satuan

No	Field	Data Type	Length	Description
1.	Id_satuan	Varchar	10	Identitas satuan obat
2.	Id_Jenis	Varchar	15	Identitas jenis obat
3.	Satuan	Varchar	15	Satuan dari obat
4.	Satuan_name	Varchar	20	Satuan nama obat

## 11. Tabel Jenis\_Obat

Primary Key : Id\_JenisObat

Foreign\_Key : -

Fungsi : Untuk Menyimpan data jenis obat

Tabel 3.11 Tabel Jenis Obat

No	Field	Data Type	Length	Description
1.	Id_JenisObat	Varchar	10	Identitas jenis obat
2.	Jenis_obat	Varchar	25	Macam-macam jenis obat

## 12. Tabel Supplier

Primary Key : ID\_Supplier

Foreign\_Key : -

Fungsi : Untuk Menyimpan data supplier

Tabel 3.12 Tabel Supplier

No	Field	Data Type	Length	Description
1.	ID_Supplier	Varchar	10	Identitas supplier

No	Field	Data Type	Length	Description
2.	Nama_supplier	Varchar	15	Nama supplier
3.	Alamat	Varchar	15	Alamat supplier
4.	kota	Varchar	20	Kota
5.	No_telp	Varchar	25	No telp supplier

### 3.5 Perancangan Desain Input dan Output

#### 3.5.1 Desain Input

Desain input adalah bagian dari perencanaan *form – form* yang digunakan untuk mendukung rancang bangun sistem persediaan barang. Berikut adalah *desain input* tersebut:

##### A. Form Master Pasien

Desain form master *pasien* berfungsi untuk menyimpan data *pasien* seperti nama, alamat, kota, umur, jenis kelamin, pekerjaan dan nomer telpon. Pasien merupakan aspek yang berpengaruh dalam perkembangan praktek kedokteran sehingga data – data terkait dengan *pasien* harus disimpan. Semua data *pasien* tercatat dalam *form* master *pasien* sehingga pihak administrasi mudah dalam pencarian data *pasien*. Adapun desain dari master *pasien* dapat dilihat pada Gambar 3.15.

Keterangan:

**Id\_Pasien** : *Control* yang digunakan adalah *textbox*. Untuk *id\_pasien* adalah *generate* 1 (satu) huruf pertama dari nama *pasien* kemudian ditambah 000-baris (*row*) beberapa dari tabel itu.

Contoh: Pasien

Maka *id\_pasien* = P-001

*Id\_pasien* bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

Id pasien	Nama pasien	Alamat	No telepon
P-001	Sita ayu	Jl.kendang	031-5673425
P-002	votia	Jl.kendang s	031-5673425
P-003	robert	Jl.taliwarnain	021-3451278
P-004	yuni	Jl.tebet barat	021-2610987

Gambar 3.15 Form Master Pasien

**Nama** : *Control* yang digunakan adalah *textbox*. Nama dari *pasien* yang disimpan pada tabel *Pasien*. Nama *pasien* bertipe data *varchar* dengan panjang 30 (tiga puluh) karakter.

**Alamat** : *Control* yang digunakan adalah *textbox*. Digunakan untuk menyimpan alamat *pasien* pada table *Pasien*. Alamat *pasien* bertipe *varchar* dengan panjang 50 (lima puluh) karakter.

**Kota** : *Control* yang digunakan adalah *combobox*. Apabila *combobox* ini dipilih maka *textbox* menampilkan propinsi dari kota yang dipilih. Kota *pasien* bertipe *varchar* dengan panjang 30 (tiga puluh) karakter.

**No.Telpon** : *Control* yang digunakan adalah *textbox*. *Textbox1* ini merupakan generate dari *combobox* kota. Sedangkan *textbox2* merupakan imputan biasa. No telpon *pasien* bertipe *varchar* dengan panjang 30 (tiga puluh) karakter.

Sedangkan fungsi–fungsi obyek ada pada desain *form* master *pasien* dapat dilihat pada Tabel 3.13.

Tabel 3.13 Fungsi Obyek Desain Form Pasien

No	Nama Obyek	Tipe Obyek	Fungsi
1	Simpan	<i>Button</i>	Menyimpan data <i>pasien</i>
2	Ubah	<i>Button</i>	Mengubah data <i>pasien</i>
3	Batal	<i>Button</i>	Membatalkan input data <i>pasien</i>
4	Keluar	<i>Button</i>	Menutup <i>form</i> <i>pasien</i>

## B. Form Master Obat

Desain *form* master obat berfungsi untuk menyimpan data obat. Adapun

desain master obat dapat dilihat pada Gambar 3.16

Keterangan:

*kode\_obat* : *Control* yang digunakan adalah *textbox*. Untuk *kode\_obat* adalah *generate* 1 (satu) huruf pertama dari nama *obat* kemudian ditambah 000-baris (*row*) seberapa dari tabel itu.

Contoh: Obat

Maka *kode\_obat* = OB-001

*Kode\_obat* bertipe data *varchar* dengan panjang 10 (sepuluh) karakter.

Nama : *Control* yang digunakan adalah *textbox*. Nama dari *obat* yang disimpan pada tabel *Obat*. Nama *obat* bertipe data *varchar* dengan panjang 25 (dua puluh lima) karakter.



The screenshot shows a 'Master Obat' form with the following fields and controls:

- Kode Obat:** Text box containing 'OB-001'.
- Nama\_obat:** Text box containing 'iflaz'.
- Jenis:** Combobox showing 'Obat dalam'.
- Status Obat:** Radio buttons for 'Resep' (selected) and 'Non Resep'.
- Instructions:** '(Resep = Obat harus menggunakan resep)' and '(Non Resep = Obat tanpa menggunakan resep)'.
- Satuan:** Combobox showing 'pack'.
- Safety Stock:** Text box containing '50'.
- Isi:** Text box containing '100'.
- Harga Beli:** Text box containing '2000'.
- Harga Jual bebas:** Text box containing '2500'.
- Harga Jual Resep:** Text box containing '2500'.
- Stock Awal, Stock Keluar, Stock Akhir:** Empty text boxes.
- Buttons:** 'Simpan', 'Ubah', 'Hapus', and 'Keluar'.
- Table:** A table with 4 columns: Kode\_obat, Nama\_obat, Jenis\_obat, and Safety\_stock.
 

Kode_obat	Nama_obat	Jenis_obat	Safety_stock
OB-001	iflaz	Obat dalam	50
OB-002	Voldatex 1%	Obat luar	50
OB-003	panadol	Obat dalam/luar	
OB-004	amoxilin	Obat dalam	30

Gambar 3.16 Form Master Obat

**Jenis Obat :** *Control* yang digunakan adalah *combobox*. Digunakan untuk menyimpan jenis\_obat pada table *Obat*. Jenis obat bertipe *varchar* dengan panjang 25 (dua puluh lima) karakter.

**Status Obat :** *Control* yang digunakan adalah *radiobutton*. Digunakan untuk menyimpan status obat pada table *Obat*. Status obat terdiri dari 2 macam yaitu resep dan non resep. Status obat bertipe *varchar* dengan panjang 30 (tiga puluh) karakter.

**Satuan :** *Control* yang digunakan adalah *combobox*. Digunakan untuk menyimpan satuan pada table *Obat*. Satuan bertipe *varchar* dengan panjang 30 (tiga puluh) karakter.

Isi : *Control* yang digunakan adalah *textbox*. Digunakan untuk menyimpan jumlah *obat* pada table *Obat*. Jumlah *obat* bertipe *integer*.

Harga\_Beli : *Control* yang digunakan adalah *textbox*. Digunakan untuk menyimpan harga beli *obat* pada table *Obat*. Harga beli bertipe *integer*.

Harga\_resep : *Control* yang digunakan adalah *textbox*. Digunakan untuk menyimpan harga resep *obat* pada table *Obat*. Harga resep bertipe *integer*.

Stok\_Masuk : *Control* yang digunakan adalah *textbox*. Digunakan untuk menyimpan stok masuk *obat* pada table *Obat*. Stok masuk *obat* bertipe *integer*.

Safety\_Stok : *Control* yang digunakan adalah *textbox*. Digunakan untuk menyimpan safety stok *obat* pada table *Obat*. Safety stok *obat* bertipe *varchar* dengan panjang 25 (dua puluh lima) karakter.

Tabel 3.13 Fungsi Obyek Desain Form Pasien

No	Nama Obyek	Tipe Obyek	Fungsi
1	Simpan	<i>Button</i>	Menyimpan data <i>pasien</i>
2	Ubah	<i>Button</i>	Mengubah data <i>pasien</i>
3	Batal	<i>Button</i>	Membatalkan input data <i>pasien</i>
4	Keluar	<i>Button</i>	Menutup <i>form pasien</i>

### C. Form Master Dokter

Desain *form* master dokter berfungsi untuk menyimpan data dokter.

Adapun desain master dokter dapat dilihat pada Gambar 3.17.

Keterangan:

Kode\_Dokter: *Control* yang digunakan adalah *textbox*. Untuk *kode\_dokter* adalah *generate* 2 (dua) huruf pertama dari nama *dokter* kemudian ditambah 000-baris (*row*) beberapa dari tabel itu.

Contoh: Detail Dokter

Maka *kode\_dokter* = DK-001

*kode\_dokter* bertipe data *varchar* dengan panjang 20 (dua puluh) karakter.

Nama : *Control* yang digunakan adalah *textbox*. Nama dari *dokter* yang disimpan pada tabel *Detail\_Dokter*. Nama *dokter* bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

Id_dokter	Nama_dokter	Spesialis	Alamat
DK-002	Drg.yoyok	gigi	Jl.Ratu Ayu
DK-001	DR.Didi R.	Umum	JL.CILILITAN
DK-003	DR.PUDJI	Umum	JL.TENGILIS

Gambar 3.17 Form Master Dokter

Spesialis : *Control* yang digunakan adalah *combobox*. Spesialis dari *dokter* yang disimpan pada tabel *Detail\_Dokter*. Spesialis *dokter* bertipe data *varchar* dengan panjang 15 (lima belas) karakter

Alamat : *Control* yang digunakan adalah *textbox*. Digunakan untuk menyimpan alamat *dokter* pada table *Detail\_Dokter*. Alamat *pasien* bertipe *varchar* dengan panjang 50 (lima puluh) karakter.

Kota : *Control* yang digunakan adalah *combobox*. Apabila *combobox* ini dipilih maka *textbox* menampilkan propinsi dari kota yang dipilih. Kota *dokter* bertipe *varchar* dengan panjang 30 (tiga puluh) karakter.

No.Telpon : *Control* yang digunakan adalah *textbox*. *Textbox1* ini merupakan *generate* dari *combobox* kota. Sedangkan *textbox2* merupakan imputan biasa. No telpon *pasien* bertipe *varchar* dengan panjang 15 (lima belas) karakter.

Tabel 3.13 Fungsi Obyek Desain Form Dokter

No	Nama Obyek	Tipe Obyek	Fungsi
1	Simpan	Button	Menyimpan data <i>dokter</i>
2	Ubah	Button	Mengubah data <i>dokter</i>
3	Batal	Button	Membatalkan input data <i>dokter</i>
4	Keluar	Button	Menutup <i>form dokter</i>

#### D. Form Master Satuan

Desain *form* master satuan berfungsi untuk menyimpan data satuan obat.

Adapun desain master satuan dapat dilihat pada Gambar 3.18

Gambar 3.18 Form Master Satuan

Keterangan:

Jenis\_Obat: *Control* yang digunakan adalah *combobox*. *Jenis\_obat* bertipe data *varchar* dengan panjang 10 (sepuluh) karakter.

Satuan : *Control* yang digunakan adalah *textbox*. Satuan dari *obat* yang disimpan pada tabel *Satuan*. Satuan *obat* bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

Tabel 3.13 Fungsi Obyek Desain Form Pasien

No	Nama Obyek	Tipe Obyek	Fungsi
1	Ok	<i>Button</i>	Menyimpan data <i>dokter</i>
2	Cancel	<i>Button</i>	Membatalkan input data <i>dokter</i>

### E. Form Master Spesialis Dokter

Desain *form* master spesialis dokter berfungsi untuk menyimpan data dokter. Adapun desain master spesialis dokter dapat dilihat pada Gambar 3.19

Gambar 3.19 Form Master Spesialis Dokter

Keterangan:

Kode\_Dokter: *Control* yang digunakan adalah *textbox*. Untuk *kode\_dokter* adalah *generate* 2 (dua) huruf pertama dari nama *dokter* kemudian ditambah 000-baris (*row*) beberapa dari tabel itu.

Contoh: Dokter

Maka *kode\_dokter* = DT-001

*kode\_dokter* bertipe data *varchar* dengan panjang 20 (dua puluh) karakter.

Nama : *Control* yang digunakan adalah *textbox*. Nama dari *dokter* yang disimpan pada tabel *Dokter*. Nama *dokter* bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

Tabel 3.13 Fungsi Obyek Desain Form Master Spesialis Dokter

No	Nama Obyek	Tipe Obyek	Fungsi
1	Simpan	<i>Button</i>	Menyimpan data <i>spesialis dokter</i>
2	Keluar	<i>Button</i>	Menutup <i>form spesialis dokter</i>

## F. Form Pelayanan Pasien

Desain *form* pelayanan pasien berfungsi untuk menyimpan data pelayanan pasien. Adapun desain pelayanan pasien dapat dilihat pada Gambar 3.20.

Keterangan:

No\_RekamMedis: *Control* yang digunakan adalah *textbox*. Untuk *No\_RekamMedis* adalah generate 2 (dua) huruf pertama dari nama *Rekam Medis* kemudian ditambah 000-baris (*row*) beberapa dari tabel itu.

Contoh: Rekam Medis

Maka *No\_RekamMedis* = RM-001

*No\_RekamMedis* bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

**RekamMedis**

**Rekam Medis**

Kode\_pasien: P-003  
 Tanggal Transaksi: Friday, July 01, 2011  
 Nama\_Pasien: ROBERT  
 Kode Rekam Medis: RM-001  
 Alamat: JL.JATIWARINGIN II  
 Id\_obat: OB-003  
 Diagnosa: gusi berdarah  
 Nama\_obat: PANADOL  
 tindakan: Gusi berdarah  
 Jumlah: 2  
 Keterangan: disebabkan oleh periodontitis

Kode_Obat	Nama_Obat	Jumlah	Subtotal
OB-001	Oskadon	1	5000
OB-002	Voltadex	2	2000
OB-003	Panadol	3	2000

Simpan    Batal    Keluar    Pembayaran

Id Rekam Medis	Tgl_periksa	Id_pasien	Diagnosa
RM-001	Friday, July 01, 2011	P-003	Gusi berdarah
RM-002	Wednesday, June 29, 2011	P-002	gusi kawat
RM-003	Friday, October 20, 2011	P-001	Penambahan gigi

Gambar 3.20 Form Rekam Medis

**Tanggal** : *Control* yang digunakan adalah *datetimepicker*. Tanggal pemeriksaan bertipe data *datetime*

**Id\_Pasien** : *Control* yang digunakan adalah *combobox*. Apabila *combobox* ini dipilih maka *textbox* menampilkan nama pasien dari id pasien yang dipilih. *Id\_pasien* bertipe *varchar* dengan panjang 15 (lima belas) karakter.

**Id\_Obat** : *Control* yang digunakan adalah *combobox*. Apabila *combobox* ini dipilih maka *textbox* menampilkan nama dokter dari id dokter yang dipilih. *Id\_obat* bertipe *varchar* dengan panjang 15 (lima belas) karakter.

**Diagnosis** : *Control* yang digunakan adalah *textbox*. Diagnosis dari *pelayanan pasien* yang disimpan pada tabel *Rekam Medis*. Diagnosis bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

**Tindakan** : *Control* yang digunakan adalah *textbox*. Tindakan yang dilakukan oleh dokter dari *pelayanan pasien* yang disimpan pada tabel

*Tindakan.* Diagnosis bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

Keterangan : *Control* yang digunakan adalah *textbox*. Keterangan dari *pasien* yang disimpan pada tabel *Rekam Medis*. Keterangan *pasien* bertipe data *varchar* dengan panjang 30 (tiga puluh) karakter.

Tabel 3.13 Fungsi Obyek Desain Form Rekam Medis

No	Nama Obyek	Tipe Obyek	Fungsi
1	Simpan	Button	Menyimpan data <i>rekam medis</i>
2	Batal	Button	Membatalkan input data <i>rekam medis</i>
3	Keluar	Button	Menutup <i>form rekam medis</i>

### G. Form Pembayaran

Desain *form* pembayaran berfungsi untuk menyimpan data pembayaran.

Adapun desain form pembayaran dapat dilihat pada Gambar 3.21

**Pembayaran**

Kode\_tindakan: T-001 ... Tanggal Transaksi: Friday, July 01, 2011 ▾

Nama tindakan: Control

Kode Resep: PB-01201200001 ...

Jumlah Item Obat: 1 + - Total: 15000

Id\_pasien: P-001

Nama\_Pasien: Sita Ayu

Alamat: JL.AMBASADOR 3 NO.23

Kode\_Obat: OB-001 ...

Nama\_Obat: IFLAZ

Spesifikasi: 250/PACK

Harga Obat: 2500

Jumlah: 3 ▾

Total: 7500 + -

Buttons: Simpan, Cetak, Batal, Keluar

Kode Obat	Nama obat	Harga obat	Jumlah
OB-001	IFLAZ	2500	15000
OB-002	Voltadex 1%	5500	30000
OB-003	panadol	2500	25000
OB-004	amoxilin	2300	40000

Gambar 3.21 Form Pembayaran



Keterangan:

Tanggal : *Control* yang digunakan adalah *datetimepicker*. Tanggal pemeriksaan bertipe data *datetime*

Kode\_Resep: *Control* yang digunakan adalah *textbox*. kode\_resep bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

Jumlah Item : *Control* yang digunakan adalah *textbox*. Jumlah item bertipe *varchar* dengan panjang 15 (lima belas) karakter.

Id\_pasien : *Control* yang digunakan adalah *textbox*. id\_pasien bertipe *varchar* dengan panjang 15 (lima belas) karakter.

Nama : *Control* yang digunakan adalah *textbox*. Nama dari *pasien* yang disimpan pada tabel *Pasien*. Nama *pasien* bertipe data *varchar* dengan panjang 30 (tiga puluh) karakter.

Alamat : *Control* yang digunakan adalah *textbox*. Digunakan untuk menyimpan alamat *pasien* pada tabel *Pasien*. Alamat *pasien* bertipe *varchar* dengan panjang 50 (lima puluh) karakter.

Kode\_obat: *Control* yang digunakan adalah *textbox*. Kode obat dapat dicari pada link label. Kode\_obat bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

Spesifikasi : *Control* yang digunakan adalah *textbox*. Spesifikasi dari *obat* yang disimpan pada tabel *Pembayaran*. Spesifikasi bertipe data *varchar* dengan panjang 30 (tiga puluh) karakter.

Harga\_obat : *Control* yang digunakan adalah *textbox*. Harga obat bertipe *varchar* dengan panjang 20 (dua puluh) karakter.

Jumlah : *Control* yang digunakan adalah *numericupdown*. Jumlah dari *obat* yang disimpan pada tabel *Pembayaran*. Jumlah bertipe data *varchar* dengan panjang 20 (dua puluh) karakter.

Total : *Control* yang digunakan adalah *textbox*. Total dari *obat* yang disimpan pada tabel *Pembayaran*. Total bertipe data *varchar* dengan panjang 30 (tiga puluh) karakter.

Tabel 3.14 Fungsi Obyek Desain Form Pembayaran

No	Nama Obyek	Tipe Obyek	Fungsi
1	Simpan	<i>Button</i>	Menyimpan data pembayaran
2	+	<i>Button</i>	Memasukkan data pembayaran
3	-	<i>Button</i>	Membatalkan <i>entry</i> data pembayaran
4	Batal	<i>Button</i>	Membatalkan data pembayaran
5	Keluar	<i>Button</i>	Menutup <i>form</i> penjualan
6	Cari	<i>Button</i>	Mencari data barang

## H. Form pembelian

Desain *form* pembelian berfungsi untuk menyimpan data pembelian.

Adapun desain form pembelian dapat dilihat pada Gambar 3.22

Keterangan:

Kode transaksi: *Control* yang digunakan adalah *textbox*. Kode transaksi bertipe data *varchar*.

Tanggal : *Control* yang digunakan adalah *datetimepicker*. Tanggal pemeriksaan bertipe data *datetime*

No\_nota : *Control* yang digunakan adalah *textbox*. No\_nota bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

**Pembelian**

**Id\_Supplier** : SP-001 ...

**Nama\_supplier** : TONY

**Alamat** : JL.SEMOLO

**Kota** : Surabaya ▾

**No.Telp** : 031 - 8796756

**Kode\_Barang** : OB-005 ...

**Nama\_Barang** : SANMOL

**Satuan** : TABLET

**Harga beli supplier** : 1500

**Jumlah beli** : 1 ▾

**Total** : 1500

**Stock** : 83 + -

Id obat	Nama obat	Sisa stock
OB-005	Sanmol	83
OB-001	iflaz	50
OB-003	Panadol	50
OB-002	Voltadek 1 %	50

**Kode Transaksi** : PB-01201200001

**Tanggal Transaksi** : Friday, July 01, 2011 ▾

**No Nota** : 1

**Data Transaksi Pembelian**

Kode_barang	Nama_barang	satuan	Harga_brg
Ob-005	sanmol	tablet	1500
OB-001	iflaz	pack	2000
OB-002	Voltadek 1 %	pack	2000
OB-003	Panadol	Tablet	2500

**Total** : 3000

**Simpan** **Batal** **Keluar**

**Total Pembelian**

Kode_pem	Nama_pem	No_nota	total
PB-01201200001	tony	1	3000
PB-01201200002	yotta	2	10000
PB-01201200004	dewi	3	25000
PB-01201200004	adi	4	30000

Gambar 3.22 Form Pembelian

**Jumlah Item** : *Control* yang digunakan adalah *textbox*. Jumlah item bertipe *varchar* dengan panjang 15 (lima belas) karakter.

**Id\_supplier** : *Control* yang digunakan adalah *textbox*. Id\_supplier bertipe *varchar* dengan panjang 15 (lima belas) karakter.

**Nama** : *Control* yang digunakan adalah *textbox*. Nama dari *supplier* yang disimpan pada tabel *pembelian*. Nama *supplier* bertipe data *varchar* dengan panjang 30 (tiga puluh) karakter.

**Alamat** : *Control* yang digunakan adalah *textbox*. Digunakan untuk menyimpan alamat *supplier* pada table *pembelian*. Alamat *supplier* bertipe *varchar* dengan panjang 50 (lima puluh) karakter.

Kode\_barang: *Control* yang digunakan adalah *textbox*. Kode barang dapat d cari pada link label. Kode\_barang bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

Spesifikasi : *Control* yang digunakan adalah *textbox*. Spesifikasi dari *obat* yang disimpan pada tabel *Pembayaran*. Spesifikasi bertipe data *varchar* dengan panjang 30 (tiga puluh) karakter.

Harga\_brg : *Control* yang digunakan adalah *textbox*. Harga barang bertipe *varchar* dengan panjang 20 (dua puluh) karakter.

Jumlah : *Control* yang digunakan adalah *numericupdown*. Jumlah dari *obat* yang disimpan pada tabel *Pembelian*. Jumlah bertipe data *varchar* dengan panjang 20 (dua puluh) karakter.

Total : *Control* yang digunakan adalah *textbox*. Total dari *harga* yang disimpan pada tabel *Pembelian*. Total bertipe data *varchar* dengan panjang 30 (tiga puluh) karakter.

Tabel 3.15 Fungsi Obyek Desain Form Pembelian

No	Nama Obyek	Tipe Obyek	Fungsi
1	Simpan	<i>Button</i>	Menyimpan data pembelian
2	+	<i>Button</i>	Memasukkan data pembelian
3	-	<i>Button</i>	Membatalkan <i>entry</i> data pembelian
4	Batal	<i>Button</i>	Membatalkan data pembelian
5	Keluar	<i>Button</i>	Menutup <i>form</i> pembelian
6	Cari	<i>Button</i>	Mencari data barang dan <i>supplier</i>

## I. Form Histori Pasien

Desain *form* histori pasien berfungsi untuk menyimpan data pembelian.

Adapun desain form pembelian dapat dilihat pada Gambar 3.23

Histori Pasien

Id\_pasien P=001 ... Kunjungan 112-02-2012 ...

Histori rekam medis

Kode_RM	Nama_Pasien	diagnosa	tindakan
RM-001	Yotta	Sakit gigi	Penambalan gigi
RM-001	Yotta	Sakit gigi	Penambalan gigi
RM-001	Yotta	Gigi berlubang	Penambalan gigi
RM-001	Yotta	Gigi berlubang	Penambalan gigi

Histroi Obat

Kode_barang	Nama_barang	satuan	Harga_brq
Ob-005	sanmol	tablet	1500
OB-001	iflaz	pack	2000
OB-002	Voltadek 1 %	pack	2000
OB-003	Panadol	Tablet	2500

Keluar

Gambar 3.23 Form Histori Pasien

Keterangan:

Id\_pasien : *Control* yang digunakan adalah *textbox*. id\_pasien bertipe data *varchar*.

kunjungan : *Control* yang digunakan adalah *datetimepicker*. Tanggal pemeriksaan bertipe data *datetime*

cari : *Control* yang digunakan adalah *linklabel*. cari bertipe data *varchar* dengan panjang 15 (lima belas) karakter.

Histori RM : *Control* yang digunakan adalah *gridview*.

Tabel 3.15 Fungsi Obyek Desain Form Pembelian

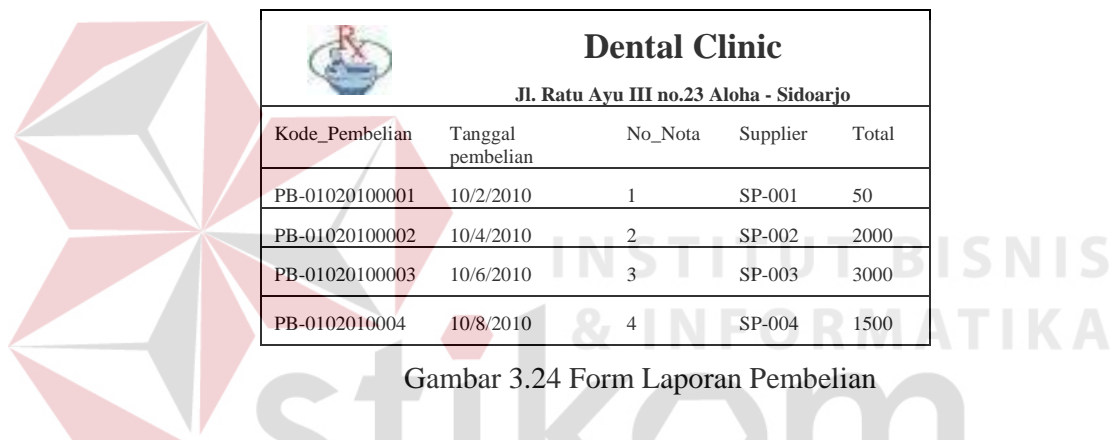
No	Nama Obyek	Tipe Obyek	Fungsi
1	keluar	Button	Menutup form histori pasien
2	Cari	Button	Mencari data pasien dan kunjungan


### 3.5.2 Desain Output

Desain *output* merupakan rancangan berupa laporan hasil pengolahan data yang dilakukan. Desain output juga merupakan acuan dalam pembuatan aplikasi dalam merancang dan membangun sistem.

#### A. Desain Output Laporan Pembelian

Laporan pembelian digunakan untuk menampilkan data barang yang telah dibeli. Desain laporan pembelian barang dapat dilihat pada Gambar 3.24.



 <b>Dental Clinic</b> Jl. Ratu Ayu III no.23 Aloha - Sidoarjo				
Kode_Pembelian	Tanggal pembelian	No_Nota	Supplier	Total
PB-01020100001	10/2/2010	1	SP-001	50
PB-01020100002	10/4/2010	2	SP-002	2000
PB-01020100003	10/6/2010	3	SP-003	3000
PB-0102010004	10/8/2010	4	SP-004	1500

Gambar 3.24 Form Laporan Pembelian


Fungsi-fungsi obyek dalam desain *output* table pembelian sebagai berikut:

Tabel 3.14 Fungsi Obyek Form Laporan Pembelian

Nama Obyek	Tipe Obyek	Fungsi
Lappeembelian	<i>Crystal Reports</i>	Menampilkan data pembelian dan data detail pembelian stok barang.

#### B. Desain Output Laporan Penjualan

Laporan penjualan digunakan untuk menampilkan data barang yang telah dijual. Desain laporan penjualan barang dapat dilihat pada Gambar 3.25. Fungsi-fungsi obyek dalam desain *output* table penjualan sebagai berikut:

 <b>Dental Clinic</b> <b>Jl. Ratu Ayu III no.23 Aloha - Sidoarjo</b>				
Kode_penjualan	Tanggal Transaksi	Id_Pasien	Kode_resep	Total
PJ-01020100001	10/2/2011	P-001	RS-01020100001	15000
PJ-01020100002	10/4/2011	P-002	RS-01020100002	25000
PJ-01020100003	10/6/2011	P-003	RS-01020100003	35000

Gambar 3.25 Form Laporan Penjualan

Fungsi-fungsi obyek dalam desain *output* table penjualan sebagai berikut:

Tabel 3.15 Fungsi Obyek Form Laporan Penjualan


Nama Obyek	Tipe Obyek	Fungsi
LapPenjualan	<i>Crystal Reports</i>	Menampilkan data penjualan dan data detail penjualan.

### C. Desain Output Laporan Data Pasien

Laporan Data Pasien digunakan untuk menampilkan data pasien yang telah terdaftar dan melakukan perawatan. Desain laporan data pasien dapat dilihat pada Gambar 3.26. Fungsi-fungsi obyek dalam desain *output* table data pasien sebagai berikut:

Tabel 3.16 Fungsi Obyek Form Laporan Data Pasien

Nama Obyek	Tipe Obyek	Fungsi
LapDataPasien	<i>Crystal Reports</i>	Menampilkan data pasien.

 <b>Dental Clinic</b> <b>Jl. Ratu Ayu III no.23 Aloha - Sidoarjo</b>						
Id_Pasien	Nama_Pasien	umur	Jenis_kelamin	Alamat	Kota	Pekerjaan
P-001	Sita Ayu	30 tahun	Wanita	Jl.Ambasador 3	Jakarta	Swasta
P-002	Yotta	35 tahun	Pria	Jl.Kendang sari	Surabaya	Swasta
P-003	Robert	18 tahun	Pria	Jl.Jatiwaringin	Jakarta	Pelajar

Gambar 3.26 Form Laporan Data Pasien