

BAB III

PERANCANGAN SISTEM

3.1 Analisis Permasalahan

Permasalahan yang dihadapi dalam proses penyimpanan data yang dilakukan secara manual yaitu pencatatan data yang dilakukan di kertas saja sehingga hal ini menyebabkan kesulitan dalam pencarian data historis untuk penyusunan laporan. Disamping kesulitan dalam pencarian data historis, kesalahan dalam proses pencatatan data pasien dapat menyebabkan adanya kejanggalaan data, hal ini seharusnya dapat diantisipasi dengan adanya pesan peringatan ketika salah menginput data. Dengan sistem yang masih sangat manual, hal ini sangat sulit dilakukan sehingga kesalahan penginputan data pasien, rekam medis, hutang pasien dan yang lainnya masih sering terjadi.

Karena segala transaksi pencatatannya menggunakan kertas, maka klinik membutuhkan jumlah kertas yang sangat banyak untuk setiap bulannya. Hal ini seharusnya dapat diantisipasi jika saja klinik melakukan pencatatan transaksi menggunakan komputer. Karena jumlah kertas yang dibutuhkan klinik sangat banyak setiap bulannya maka klinik harus mengeluarkan biaya yang cukup banyak untuk pembelian kertas, sehingga berpengaruh pada pendapatan klinik. Selain berpengaruh pada pendapatan, penggunaan kertas yang banyak juga membutuhkan ruang untuk penyimpanan berkas-berkas.

Selain itu, klinik belum menyediakan kartu berobat yang dapat digunakan sebagai tanda pengenal untuk pasien-pasien yang sudah pernah berobat. Hal tersebut dapat menyebabkan adanya catatan ganda untuk data pasien

yang sama. Adanya kemungkinan kelalaian dalam menangani hutang pasien yang belum terbayar, hal ini disebabkan tidak ada yang mengingatkan dan hutang pasien hanya dicatat di rekam medis, sehingga ada kemungkinan hutang pasien tidak terbayar dan dapat menyebabkan kerugian bagi pihak klinik. Untuk hak akses dan pencatatan yang masih dilakukan secara manual, menyebabkan data dapat diakses oleh siapa saja karena hanya berupa dokumen dalam kertas dan tidak ada sistem hak akses. Dengan sistem pencatatan yang dilakukan di kertas maka keamanan data tidak terjamin, disamping itu juga dapat menyebabkan kehilangan data dan pencurian data.

Pada saat penyusunan laporan pendapatan dan pengeluaran, membutuhkan waktu yang cukup lama sehingga tidak efisien. Keadaan ini disebabkan karena pencatatan dilakukan masih secara manual dan adanya kemungkinan beberapa data pendapatan dan data pengeluaran yang terlewatkan.

Dengan sistem kerja yang seperti itu, akan menyebabkan gangguan dan masalah pada kegiatan seperti pencatatan, pengecekan, penyusunan laporan dan lain-lain. Gangguan dan masalah tersebut dapat mengurangi kinerja pelayanan kesehatan suatu klinik. Untuk itu, sistem manual tersebut sebaiknya diganti dengan sistem baru yang telah terkomputerisasi, sehingga aktivitas-aktivitas yang dilakukan di klinik dapat berjalan lebih lancar.

Dokumen *flow* pada Rancang Bangun Aplikasi Klinik Gigi dengan Studi Kasus: Klinik Gigi Drg. Basuki adalah sebagai berikut:

1. Dokumen *flow* proses penerimaan pasien, pemeriksaan pasien, dan pembayaran pasien.
2. Dokumen *flow* proses pembelian obat ke *supplier*.

Masing masing dokumen *flow* akan digambarkan pada sub bab ini. Setiap dokumen *flow* menggambarkan aliran sistem dari aplikasi.

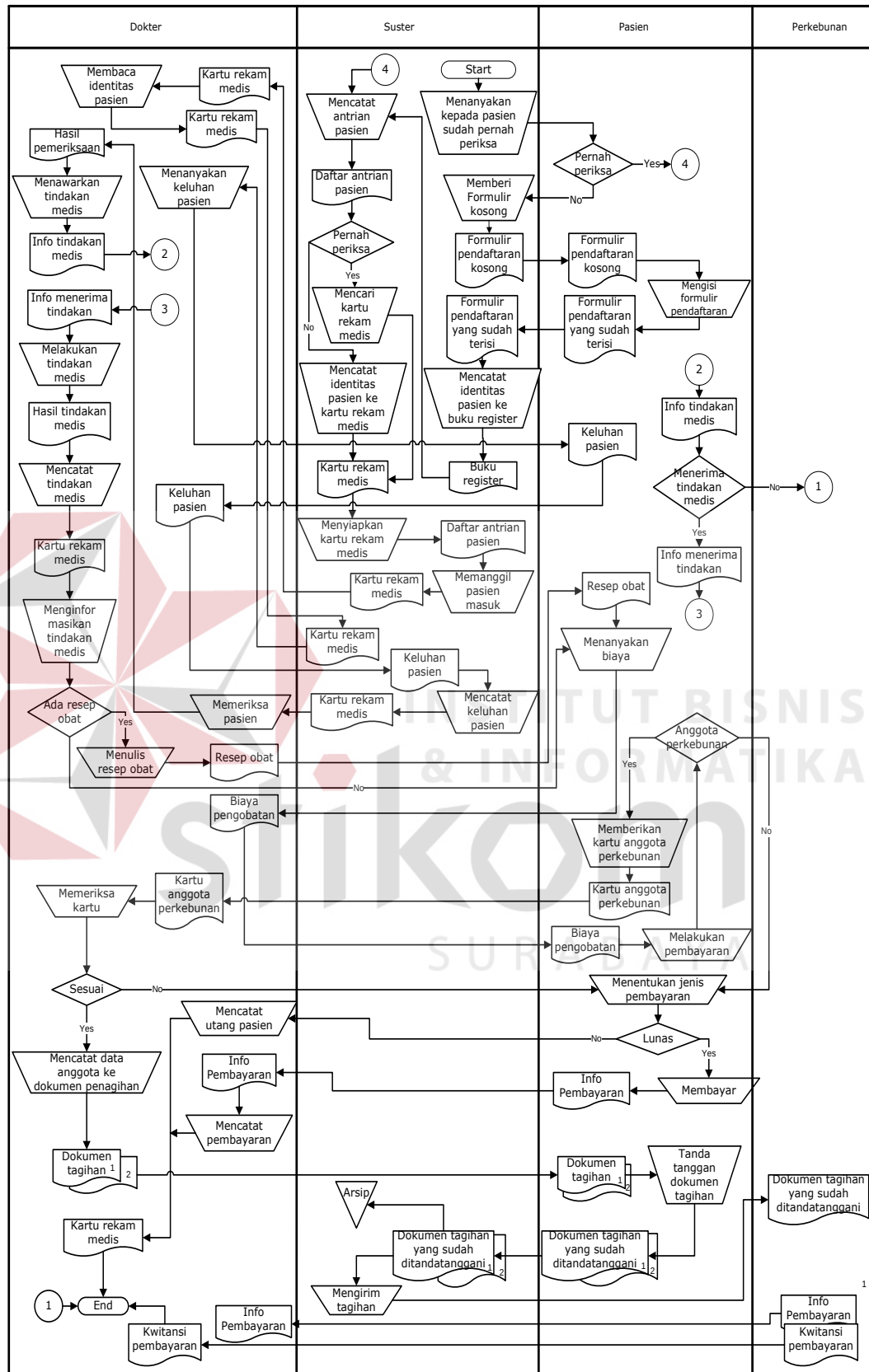
1. Dokumen *Flow* Proses Penerimaan Pasien, Pemeriksaan Pasien, dan Pembayaran Pasien

Dalam dokumen *flow* proses penerimaan pasien, pemeriksaan pasien, dan pembayaran pasien terdapat 4 (empat) *entity* yaitu dokter, suster, pasien, dan perusahaan. Pada dokumen *flow* ini, proses dimulai dari suster yang menanyakan status pasien apakah sudah pernah berobat di klinik tersebut atau belum. Jika sudah pernah maka pasien langsung dicatat namanya pada buku antrian, dan jika belum maka suster akan meminta identitas pasien dan mencatat identitasnya ke buku register dan juga buku antrian. Setelah proses penerimaan pasien, suster kemudian menyiapkan rekam medis pasien yang sudah pernah berobat dan yang baru pertamakali berobat untuk kemudian diserahkan kepada dokter saat giliran pasien diperiksa. Saat pemeriksaan, dokter tidak langsung melakukan tindakan medis tetapi memberikan keputusan kepada pasien apakah akan menerima tindakan medis atau tidak. Pada proses pembayaran, dokter akan memberi tahu total biaya pemeriksaan kepada pasien. Dalam hal ini jika pasien berstatus anggota perusahaan, maka pasien cukup menunjukkan kartu anggota perusahaan dan menandatangani dokumen tagihan. Dokumen tagihan kemudian dikirimkan ke perusahaan yang bersangkutan, dan pengiriman dokumen tersebut biasanya dilakukan tiap akhir bulan. Bagi pasien umum, klinik ini juga memberi kemudahan dalam proses pembayaran. Kemudahannya yaitu pasien dapat membayar biaya medis secara kredit berdasarkan kesepakatan dokter. Dokumen

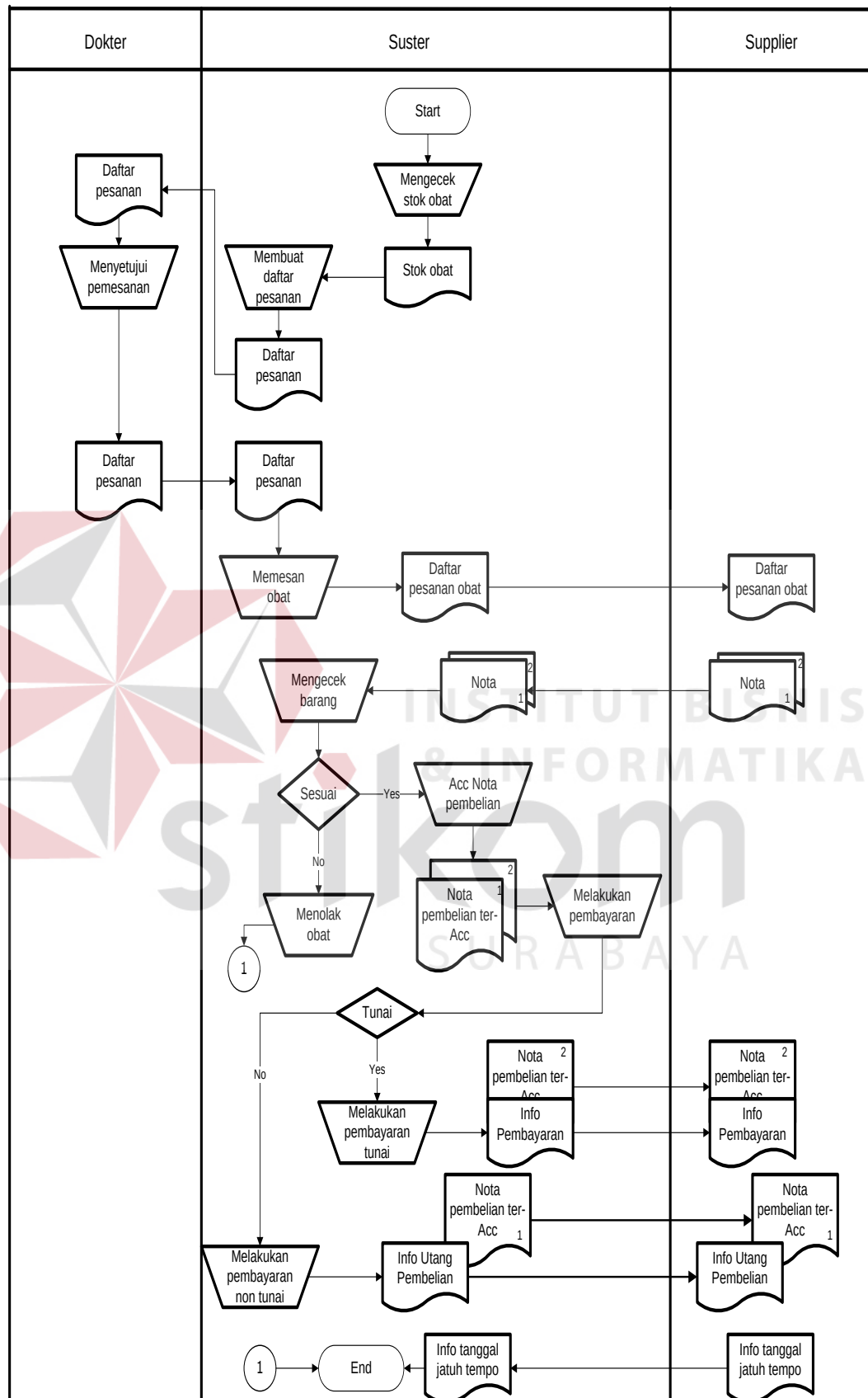
flow proses penerimaan pasien, pemeriksaan pasien, dan pembayaran pasien dapat dilihat pada Gambar 3.1 di halaman 17.

2. Dokumen *Flow* Proses Pembelian Obat ke *Supplier*

Pada dokumen *flow* proses pembelian obat dimulai dari suster yang melakukan pengecekan stok obat. Kemudian membuat daftar obat yang akan dipesan dan kemudian diberikan ke dokter untuk memperoleh persetujuan pemesanan pembelian obat. Jika dokter telah setuju maka suster akan melakukan pemesanan pada *supplier*. Saat barang pesanan sudah sampai ke klinik maka suster yang akan melakukan pemeriksaan kesesuaian barang kiriman jika sesuai maka suster akan menandatangani nota sebagai bukti barang sudah diterima dengan baik. Setelah nota ditandatangani, maka suster menyampaikan informasi pembayaran pembelian akan dilakukan secara tunai atau non tunai. Jika pembayaran dilakukan non tunai maka klinik akan memperoleh informasi jatuh tempo pembayaran dari *supplier*. Dokumen *flow* proses pembelian obat ke *supplier* ini memiliki 3 (tiga) *entity* yaitu dokter, suster dan *supplier*. Dokumen *flow* proses pembelian obat ke *supplier* dapat dilihat pada Gambar 3.2 di halaman 18.



Gambar 3.1 Dokumen Flow Proses Penerimaan Pasien, Pemeriksaan Pasien, dan Pembayaran Pasien

Gambar 3.2 Dokumen *Flow* Proses Pembelian Obat ke *Supplier*

3.2 Analisis Kebutuhan

Berdasarkan permasalahan yang ada maka dapat dilakukan analisis kebutuhan untuk menghasilkan suatu sistem terkomputerisasi yang mampu membantu pencatatan data pasien maupun rekam medis, dan pencatatan data transaksi tidak dilakukan pada kertas sehingga dapat mengatasi kesalahan pada saat menyimpan data dengan memberi pesan kesalahan *input* atau tipe data yang disimpan tidak sesuai dan menghindari penyimpanan data yang tidak lengkap. Dengan berkurangnya penggunaan kertas maka secara otomatis akan meminimalkan penggunaan ruang untuk menyimpan berkas-berkas. Selain itu, sistem yang akan dibuat diharapkan dapat mempermudah pencarian data historis dengan bantuan *input* data yang dicari berdasarkan kategori pencarian karena setiap transaksi tersimpan dalam satu *database*.

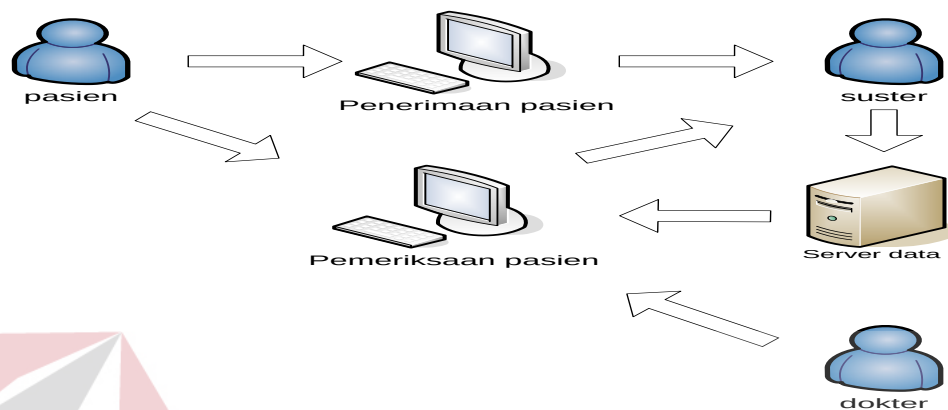
Untuk pembuatan setiap laporan, tidak perlu merekap dan menghitung kembali setiap data transaksi dan laporan dapat dicetak setiap waktu pada saat dibutuhkan.

Dengan analisa kebutuhan diharapkan aplikasi yang akan dibangun dapat mengefisienkan waktu dan membantu permasalahan yang terjadi di klinik gigi drg. Basuki, yaitu mulai pencatatan, pencarian dan sampai pada pembuatan laporan sehingga dapat mempermudah dan mempercepat kegiatan transaksi yang dilakukan.

3.3 Gambaran Umum Sistem

Aplikasi ini dibangun dalam bentuk aplikasi desktop (*desktop application*) dan menggunakan bahasa pemrograman Visual Basic .Net 2005.

Sistem dimulai dari penerimaan pasien. Selanjutnya data pasien yang akan berobat diberikan ke dokter. Hasil pemeriksaan disimpan sebagai rekam medis. Gambaran umum sistem untuk proses penerimaan dan pemeriksaan klinik dapat dilihat pada Gambar 3.3.



Gambar 3.3 Proses Penerimaan dan Pemeriksaan Klinik

3.4 Perancangan Sistem

Berdasarkan dari analisis permasalahan yang ada, tahap berikutnya dari siklus pengembangan sistem adalah perancangan sistem. Pada tahap ini terdapat aktifitas pendefinisian kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun hingga implementasi dari sistem. Pada tahap ini juga akan membahas tentang perancangan sistem yang meliputi *System Flow* yang menunjukkan alur sistem yang akan dibuat hasil analisa, *Data Flow Diagram* (DFD) yang merupakan diagram aliran data pada sistem yang akan dibuat, dan *Entity Relationship Diagram* (ERD) yang menggambarkan diagram relasi antar tabel-tabel. Pada ERD terdapat *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM) yang menggambarkan tabel-tabel dan relasinya secara konsep dan detilnya, juga dilengkapi dengan desain *Input Output* yang menggambarkan tampilan *input* dan *output* program yang akan dibuat.

3.5 *System Flow*

System flow adalah model yang disusun berdasarkan analisa untuk menggambarkan jalannya urutan proses yang terjadi dalam sistem aplikasi klinik gigi. *Sistem flow* pada Rancang Bangun Aplikasi Klinik Gigi dengan Studi Kasus: Klinik Gigi Drg. Basuki adalah sebagai berikut:

1. *System flow* proses penerimaan pasien.
2. *System flow* proses pemeriksaan pasien.
3. *System flow* proses pembayaran biaya pemeriksaan.
4. *System flow* proses pembelian bahan baku dan peralatan ke *supplier*.
5. *System flow* proses pemesanan ke *tehniker*.

Masing-masing *system flow* akan digambarkan pada sub bab ini.

1. *System Flow* Proses Penerimaan Pasien.

Pada Gambar 3.4 *System Flow* Proses Penerimaan Pasien di halaman 23 terdapat 4 (empat) *entity* yaitu pasien, perusahaan, suster, dan dokter. Pada proses penerimaan pasien dimulai dari pasien melakukan pendaftaran antrian dengan memberikan data diri yang diperlukan klinik kepada suster jika pasien adalah baru pertama kali berobat dan belum memiliki kartu berobat. Berdasarkan data diri yang ada, suster akan mencetak kartu berobat yang nantinya dapat digunakan seterusnya oleh pasien, setiap kali pasien datang berobat. Disini pasien digolongkan menjadi 2 (dua) yaitu pasien umum dan jaminan sosial tenaga kerja (jam sostek). Untuk pasien jam sostek akan selalu dimintakan kartu keanggotaan perusahaan tempat pasien bekerja, yang mana kartu tersebut sebagai bukti bahwa pasien bekerja di perusahaan tersebut atau merupakan tanggungan perusahaan yang bekerjasama dengan klinik.

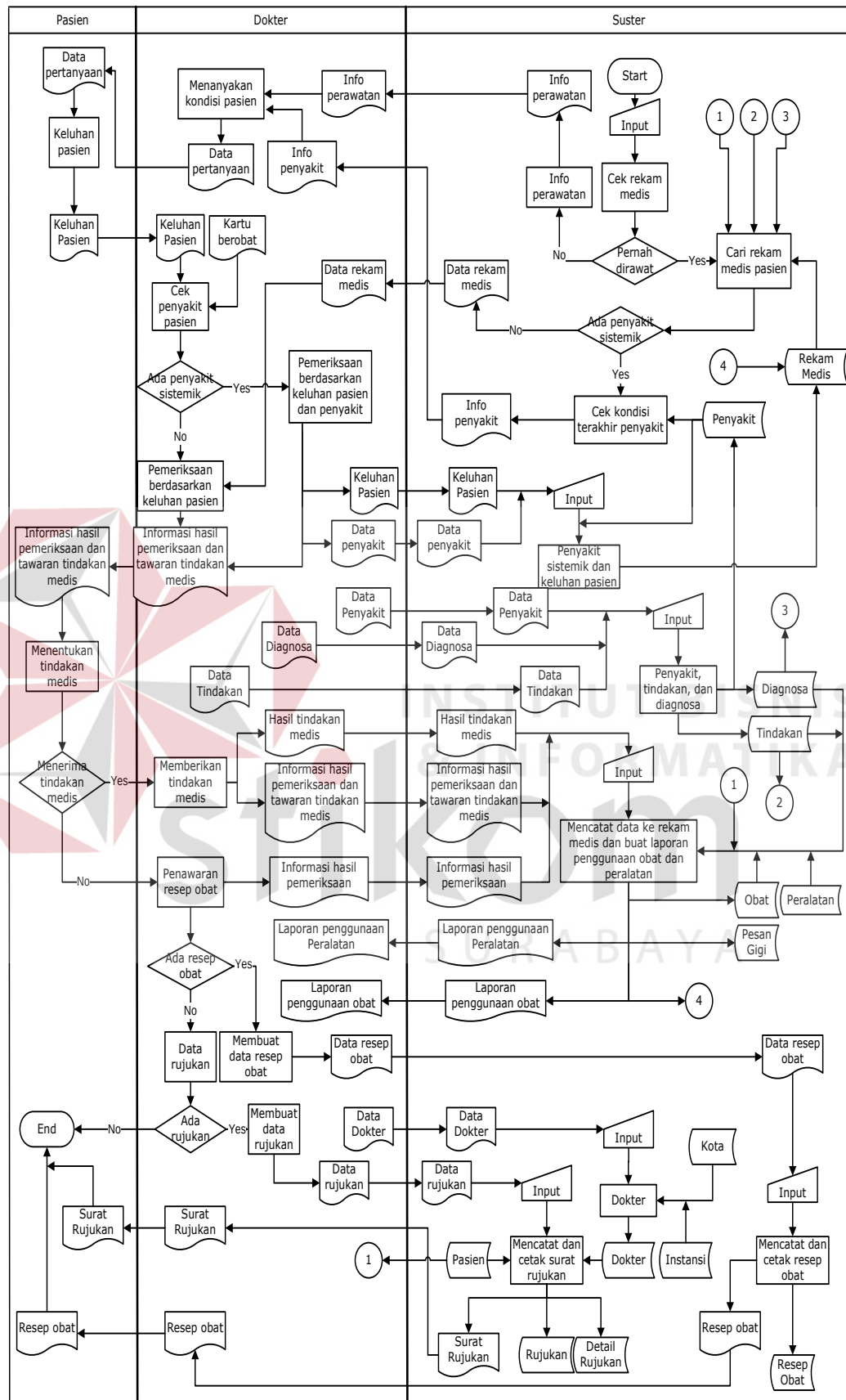
2. *System Flow* Proses Pemeriksaan Pasien.

Untuk Gambar 3.5 *System Flow* Proses Pemeriksaan Pasien di halaman 24 ada 3 (tiga) *entity* yaitu pasien, suster, dan dokter. Pasien menyampaikan keluhan pada dokter dan dokter akan melakukan pemeriksaan. Berdasarkan hasil pemeriksaan, dokter menawarkan tindakan medis, resep obat, atau bahkan surat rujukan kepada pasien jika diperlukan. Jika pasien menyetujui tindakan medis yang ditawarkan maka dokter akan memberi tindakan medis. Untuk keluhan pasien, hasil pemeriksaan, tindakan medis, dan resep obat kemudian akan dicatat oleh suster sebagai catatan rekam medis. Sedangkan rujukan akan disimpan sebagai informasi bahwa pasien pernah ditangani oleh dokter siapa saja. Apabila ada resep obat maka suster akan mencetak resep tersebut demikian juga dengan surat rujukan.

3. *System Flow* Proses Pembayaran Biaya Pemeriksaan.

Gambar 3.6 *System Flow* Proses Pembayaran Biaya Pemeriksaan pada halaman 25 terdapat 4 (empat) *entity* diantaranya yaitu pasien, dokter, suster, dan perusahaan. Pada proses pembayaran, dokter menentukan biaya pemeriksaan berdasarkan tindakan yang diberikan kepada pasien. Biaya pemeriksaan kemudian dicatat oleh suster yang nantinya akan digunakan untuk membuat laporan pendapatan. Biaya pemeriksaan akan diinformasikan ke pasien sehingga pasien dapat melakukan pembayaran. Pasien yang bersetatus jam sostek hanya perlu menandatangani dokumen penagihan yang kemudian akan dikirim ke perusahaan setiap akhir bulan sebagai bukti pemeriksaan. Pembayaran dapat dilakukan secara tunai atau diangsur jika diperlukan.

Gambar 3.4 *System Flow* Proses Penerimaan Pasien.



Gambar 3.5 System Flow Proses Pemeriksaan Pasien.

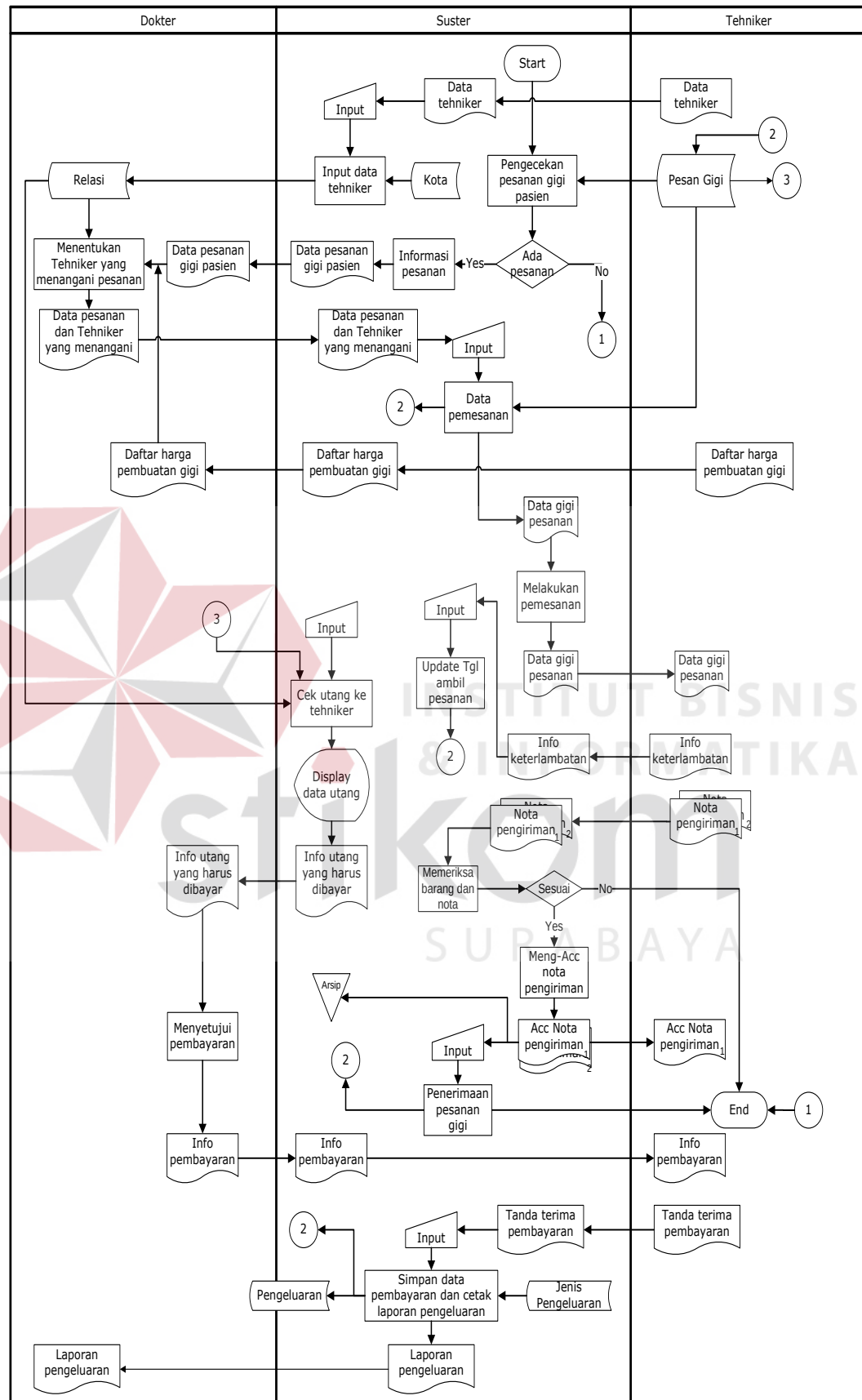
4. *System Flow* Proses Pembelian Bahan Baku dan Peralatan ke *Supplier*.

Penjelasan Gambar 3.7 *System Flow* Proses Pembelian Bahan Baku dan Peralatan ke *Supplier* di halaman 27 memiliki 3 (tiga) *entity* yaitu *supplier*, suster, dan dokter. Pada proses ini, suster melakukan pemesanan apabila bahan baku sudah mencapai batas minimum atau ada bahan baku baru yang diinginkan dokter. Pemesanan peralatan dilakukan jika diperlukan atau ada peralatan baru yang diinginkan dokter. Sebelum melakukan pemesanan, suster meminta persetujuan dokter, apabila data pesanan distujui maka pemesanan dapat dilakukan. Jika ada pemesanan bahan baku dan peralatan baru, suster akan menyimpan data dari bahan baku dan peralatan baru tersebut. Pemeriksaan barang pesanan yang sudah tiba dan pencatatan pembelian dilakukan oleh suster. Pembayaran pembeli bahan baku dan peralatan dapat dilakukan secara tunai ataupun kredit.

5. *System Flow* Proses Pemesanan ke *Tehniker*.

Pada Gambar 3.8 *System Flow* Proses Pemesanan Gigi ke *Tehnikrt* di halaman 28 terdapat 3 (tiga) *entity* yaitu dokter, suster, dan *tehniker*. *System flow* proses pemesanan dimulai saat suster memeriksa rekam medis dan bila ada pasien yang melakukan pemesanan pembuatan gigi tiruan maka suster akan menginformasikannya ke dokter. Setelah menerima informasi pemesanan dari suster, selanjutnya dokter memilih *tehniker* yang akan mengerjakan order gigi tiruan. Suster akan melakukan pemesanan ke *tehniker* apabila telah memperoleh persetujuan dokter. Dan barang pesanan yang sudah selesai akan diterima dan diperiksa oleh suster. Jika barang kiriman sudah sesuai dengan pesanan maka suster akan mencatat tanggal penyerahan barang yang di order.

Gambar 3.7 *System Flow* Proses Pembelian Bahan Baku dan Peralatan ke *Supplier*.



Gambar 3.8 System Flow Proses Pemesanan ke Tehnik.

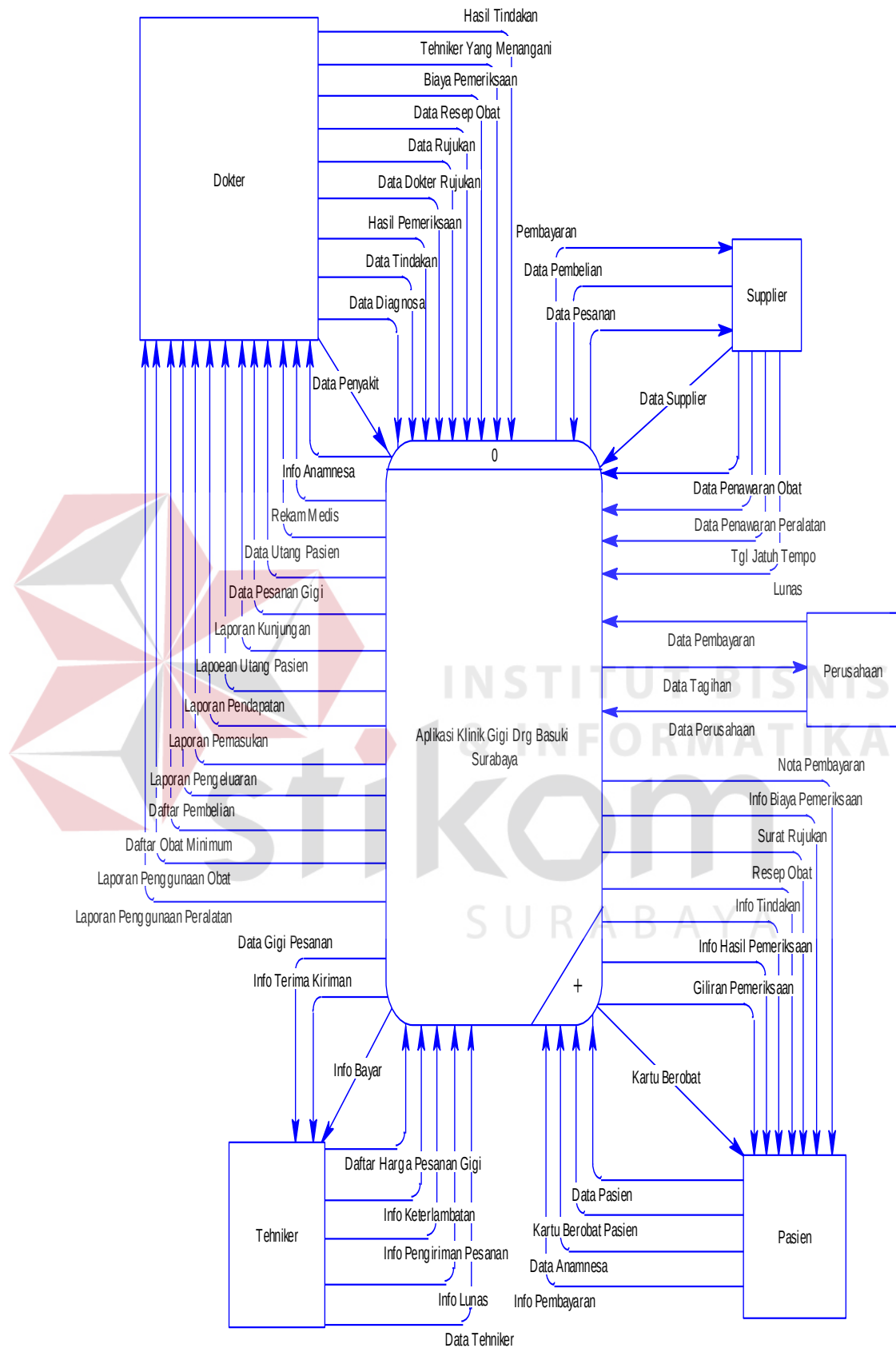
3.6 Data Flow Diagram

Menurut Kristanto.(2004), *Data Flow Diagram* (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

Data Flow Diagram merupakan suatu metode pengembangan sistem yang terstruktur (*structure analysis and design*). Penggunaan notasi dalam *data flow diagram* ini sangat membantu sekali untuk memahami suatu sistem pada semua tingkat kompleksitas. Pada tahap analisis penggunaan notasi ini dapat membantu dalam berkomunikasi dengan pemakai sistem untuk memahami sistem secara logika.

3.6.1 Context Diagram Sistem Informasi Klinik Gigi Drg. Basuki Surabaya

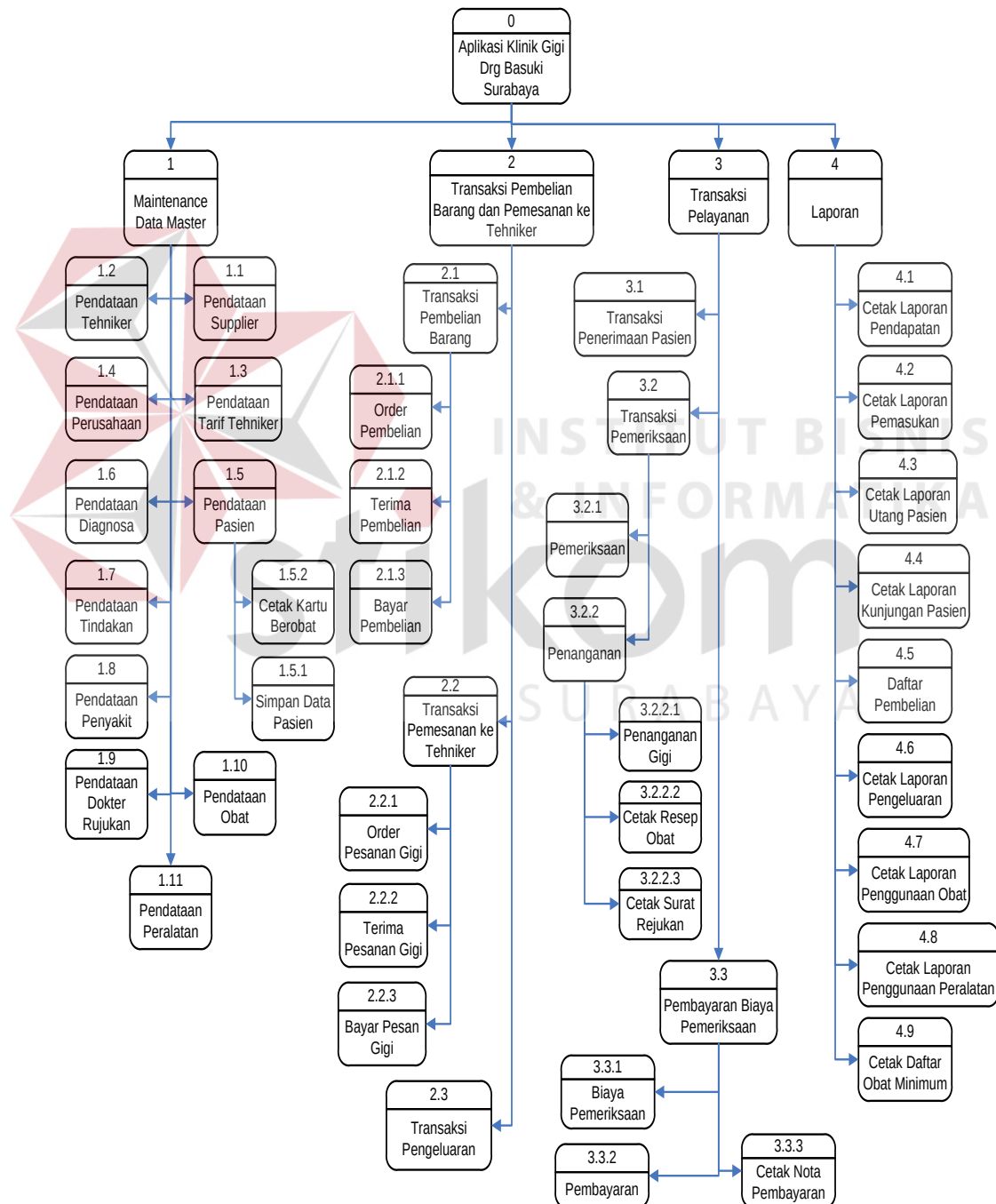
Penggambaran sistem DFD dimulai dari *Context Diagram*. Dari *Context Diagram* dapat didekomposisi lagi menjadi level yang lebih rendah untuk menggambarkan sistem lebih rinci. Pada *Context Diagram*, *entity system* informasi klinik gigi drg. Basuki Surabaya terdiri dari Dokter, Pasien, Perusahaan, *Tekniker*, dan *Supplier*. Dokter nantinya akan menerima laporan data pelayanan dan data transaksi pada periode tertentu yang merupakan keluaran dari *system* ini. *Context Diagram* dapat dilihat pada Gambar 3.9 di halaman 30.



Gambar 3.9 *Context Diagram* Sistem Informasi Klinik Gigi Drg. Basuki Surabaya.

3.6.2 Diagram Berjenjang

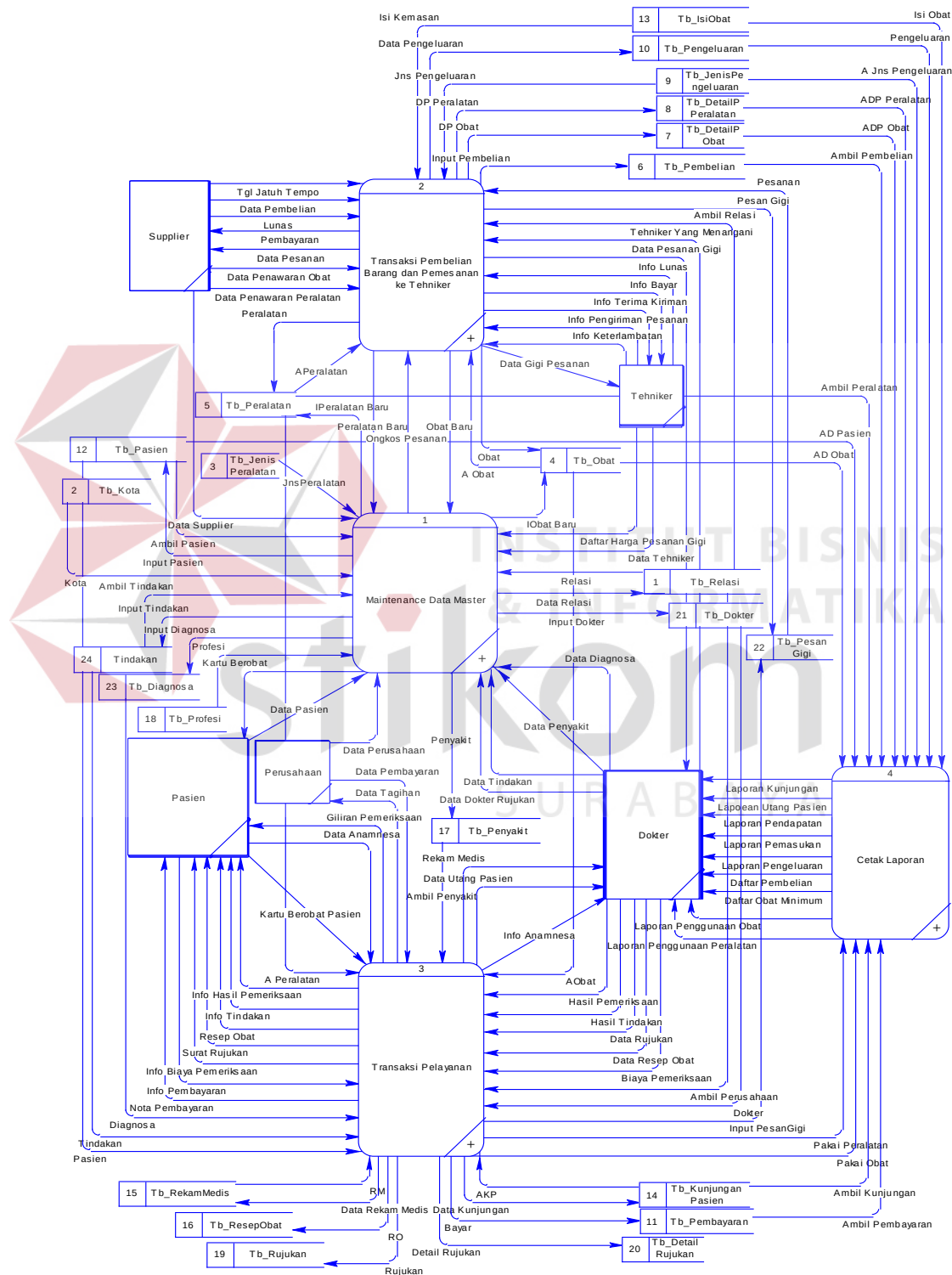
Diagram Berjenjang digunakan untuk menggambarkan proses dalam sistem, dimana dimulai dari proses sistem yang besar atau yang tinggi hingga proses sistem yang kecil atau rendah. Diagram berjenjang Rancang Bangun Aplikasi Klinik Gigi Drg. Basuki dapat dilihat pada gambar 3.10.



Gambar 3.10 Diagram Berjenjang Sistem Klinik Gigi Drg. Basuki

3.6.3 DFD Level 0 Sistem Informasi Klinik Gigi Drg. Basuki Surabaya

Setelah *context diagram* didekomposisi akan didapat DFD *level 0* seperti terlihat pada Gambar 3.11.



Gambar 3.11 DFD Level 0 Sistem Informasi Klinik Gigi Drg. Basuki Surabaya

Setelah *context diagram* didekomposisikan maka akan didapat DFD *level*

0. Pada DFD *level* 0 terdapat 3 (tiga) proses yaitu:

1. Proses Pendataan / *Maintenance Data Master*

Data yang masuk pada proses ini adalah data pasien yang berupa identitas pasien dan data perusahaan dari perusahaan yang bekerjasama dengan klinik. Berikutnya *supplier* dan *tehniker*, dimana *supplier* memasukan data *supplier* dan barang-barang yang ditawarkan ke klinik dan *tehniker* memasukan data *tehniker* dan biaya pembuatan gigi, sedangkan dokter memasukan data tindakan, data diagnosa, data penyakit, dan data dokter untuk rujukan.

2. Proses Transaksi Pembelian Barang dan Pemesanan ke *Tehniker*

Masukan data proses transaksi pembelian barang dan pemesanan ke *tehniker* dari data tersimpan, yaitu *tehniker*, *supplier*, biaya pembuatan gigi, tindakan, obat, peralatan, dan data pesanan gigi. Proses ini digunakan untuk melakukan transaksi yang dilakukan oleh klinik, berupa proses pembelian barang yaitu obat dan peralatan medis, dan proses pemesanan ke *tehniker*.

3. Proses Transaksi Pelayanan

Masukan data proses transaksi pelayanan dari data tersimpan, yaitu pasien, perusahaan, rekam medis, tindakan, diagnosa, obat, peralatan, dan dokter rujukan. Proses ini digunakan untuk melakukan transaksi yang dilakukan oleh klinik, berupa proses penerimaan pasien, pemeriksaan pasien, dan pembayaran biaya pemeriksaan.

4. Proses Pelaporan

Proses ini digunakan untuk membuat pelaporan transaksi yang terjadi di klinik.

3.6.4 DFD Level 1 Subproses Pendataan / Maintenance Data Master

Proses sistem informasi klinik gigi drg. Basuki Surabaya dapat didekomposisi lagi menjadi DFD *level 1* (satu) subproses pendataan / *maintenance* data master. DFD *level 1* (satu) subproses pendataan / *maintenance* data master dapat dilihat pada Gambar 3.12 di halaman 38. Pada DFD *level 1* (satu) ini terdapat 11 (sebelas) proses, yaitu proses pendataan *supplier*, proses pendataan *tehniker*, proses pendataan tarif *tehniker*, proses pendataan perusahaan, proses pendataan pasien, proses pendataan diagnosa, proses pendataan tindakan, proses pendataan penyakit, proses pendataan dokter rujukan, proses pendataan obat, dan proses pendataan peralatan.

Pada Gambar 3.12 di halaman 38 terdapat 11 (sebelas) proses yaitu:

1. Proses Pendataan *Supplier*

Proses pendataan *supplier* berupa penyimpanan data *supplier* yang memasok bahan baku dan peralatan di klinik. Data-data yang dicatat yaitu nama *supplier*, alamat, dan nomor telepon. Data *supplier* disimpan dalam *file* relasi.

2. Proses Pendataan *Tehniker*

Pada proses pendataan *tehniker*, data-data yang dicatat yaitu nama *tehniker*, alamat, dan nomor telepon. Data disimpan dalam *file* relasi.

3. Proses Pendataan Tarif *Tehniker*

Data tarif *tehniker* diperoleh dari *tehniker* yang menerima order pesanan pembuatan gigi. Data tarif *tehniker* disimpan dalam *file* tarif *tehniker*, yang berisikan *tehniker*, tindakan, biaya1, dan biaya2.

4. Proses Pendataan Perusahaan

Perusahaan yang bekerja sama dengan klinik memasukkan data perusahaannya berupa nama, alamat, dan nomor telepon. Data tersebut kemudian disimpan pada *file* relasi.

5. Proses Pendataan Pasien

Pada proses ini pasien memasukkan data pasien berupa nama, alamat, tanggal lahir, jenis kelamin, penyakit sistemik, dan pekerjaan, kemudian data pasien disimpan dalam *file* pasien. Sistem akan memberikan umpan balik kepada pasien berupa kartu berobat.

6. Proses Pendataan Diagnosa

Pada proses pendataan diagnosa data yang dibutuhkan yaitu daftar diagnosa yang ada di klinik. Daftar tersebut diperoleh dari dokter dan kemudian disimpan pada *file* diagnosa.

7. Proses Pendataan Tindakan

Pada proses ini dokter memasukan data-data tindakan yang disediakan oleh klinik. Data tindakan yang dimasukan berupa tindakan dan biaya-biaya yang disimpan di *file* tindakan.

8. Proses Pendataan Penyakit

Data penyakit disimpan pada *file* penyakit, yang mana *file* tersebut berisi daftar penyakit-penyakit sistemik (penyakit “badan”)

9. Proses Pendataan Dokter Rujukan

Pada proses ini dokter memasukkan data dokter rujukan berupa nama, alamat, kota, telepon rumah, hp, nama tempat dokter bekerja, alamat tempat dokter

bekerja, kota tempat dokter bekerja, dan telepon tempat dokter bekerja. Data tersebut kemudian disimpan dalam *file* dokter rujukan.

10. Proses Pendataan Obat

Data obat dapat diperoleh dari dokter atau *supplier* yang berasal dari transaksi dan disimpan pada *file* obat. Data yang menjadi masukan berupa nama obat, isi obat persatuan dan harga pembelian terakhir.

11. Proses Pendataan Peralatan

Data peralatan dapat diperoleh dari dokter atau *supplier* yang berasal dari transaksi dan disimpan pada *file* peralatan. Data yang menjadi masukan berupa jenis peralatan, merek/tipe peralatan dan jumlahnya.

3.6.5 DFD Level 1 Subproses Transaksi Pembelian Barang dan Pemesanan ke *Tehniker*

Proses sistem informasi klinik gigi drg. Basuki Surabaya dapat didekomposisi lagi menjadi DFD *level* 1 (satu) subproses transaksi pembelian barang dan pemesanan ke *tehniker*. Pada DFD *level* 1 (satu) ini terdapat 3 (tiga) proses, yaitu proses transaksi pembelian barang, proses transaksi pesanan ke *tehniker*, dan proses pengeluaran. DFD *level* 1 (satu) subproses transaksi pembelian dan pemesanan ke *tehniker* dapat dilihat pada Gambar 3.13 di halaman 39.

Pada Gambar 3.13 di halaman 39 terdapat 3 (tiga) proses, yaitu:

1. Proses Transaksi Pembelian Barang

Transaksi pembelian barang dapat berupa pembelian obat atau pembelian peralatan. Transaksi pembelian mendapat masukan dari *file* obat berupa stok

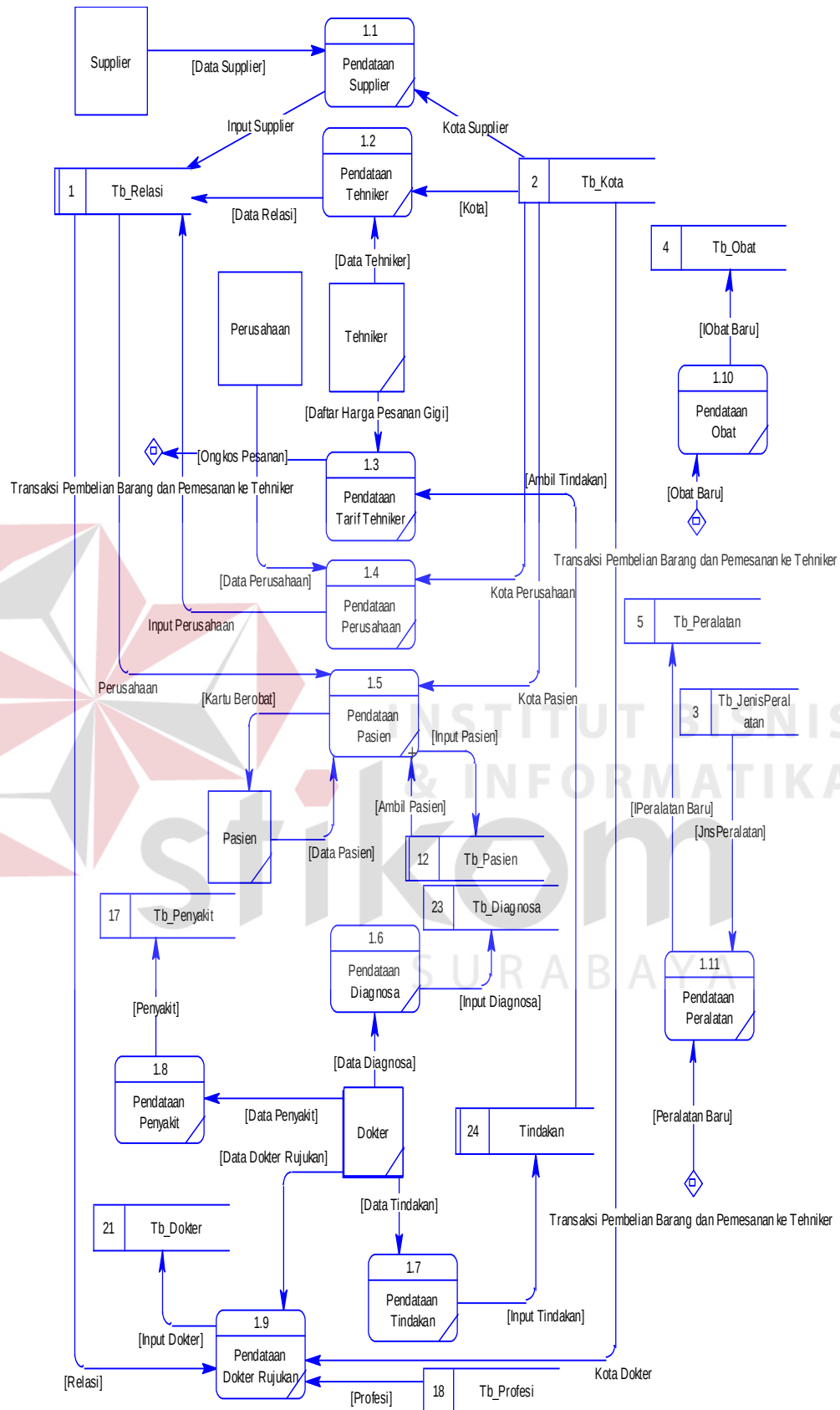
obat minimum, *file* peralatan berupa jumlah peralatan, *file* relasi, dan data pembelian dari *supplier*. Proses ini menghasilkan daftar barang yang akan dipesankan ke *supplier* dan memperoleh umpan balik berupa nota pembelian. Transaksi pembelian akan disimpan di *file* pembelian, *file* detail pembelian peralatan, dan *file* detail pembelian obat.

2. Proses Transaksi Pesanan ke *Tehniker*

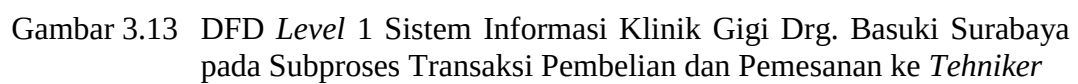
Transaksi pemesanan gigi menggunakan *file* pesan gigi, *file* relasi dan data ongkos pesanan dari proses master / pendataan, serta mendapat masukan info pengiriman pesanan dan informasi keterlambatan dari *tehniker* jika jadwal pengiriman pesanan mengalami penundaan. Proses transaksi ini menghasilkan inputan pesanan gigi ke *tehniker* untuk *file* pesan gigi, dan daftar pesanan gigi dan info terima kiriman untuk *tehniker*.

3. Proses Transaksi Pengeluaran

Proses transaksi pengeluaran menggunakan *file* jenis pengeluaran dan memperoleh masukan dari proses transaksi pembelian barang berupa biaya pembelian dan masukan dari proses transaksi pemesanan ke *tehniker* berupa biaya pemesanan. Transaksi ini menghasilkan data pengeluaran yang disimpan pada *file* pengeluaran.



Gambar 3.12 DFD Level 1 Sistem Informasi Klinik Gigi Drg. Basuki Surabaya pada Subproses Pendataan / Master



Proses sistem informasi klinik gigi drg. Basuki Surabaya dapat didekomposisi lagi menjadi DFD *level* 1 (satu) subproses transaksi pelayanan. Pada DFD *level* 1 (satu) ini terdapat 3 (tiga) proses, yaitu proses transaksi penerimaan pasien, proses pemeriksaan, dan proses pembayaran biaya pemeriksaan. DFD *level* 1 (satu) subproses transaksi pelayanan dapat dilihat pada Gambar 3.14.

Pada Gambar 3.14 di halaman 40 terdapat 3 (tiga) proses, yaitu:

1. Proses Transaksi Penerimaan Pasien

Transaksi penerimaan pasien mendapat masukan dari *file* pasien, *file* relasi, dan pasien. Proses ini menghasilkan informasi giliran pemeriksaan untuk pasien. Transaksi penerimaan pasien akan disimpan di *file* kunjungan pasien.

2. Proses Transaksi Pemeriksaan

Transaksi pemeriksaan menggunakan *file* obat, *file* peralatan, *file* penyakit, *file* rekam medis, *file* dokter, *file* diagnosa, dan *file* tindakan. Proses ini memperoleh masukan dari pasien berupa data anamnesa, selanjutnya masukan dari dokter berupa hasil tindakan, hasil pemeriksaan, data resep obat, dan data rujukan. Proses transaksi ini disimpan pada *file* pesan gigi, *file* rekam medis, *file* resep obat, *file* detail rujukan, dan *file* rujukan.

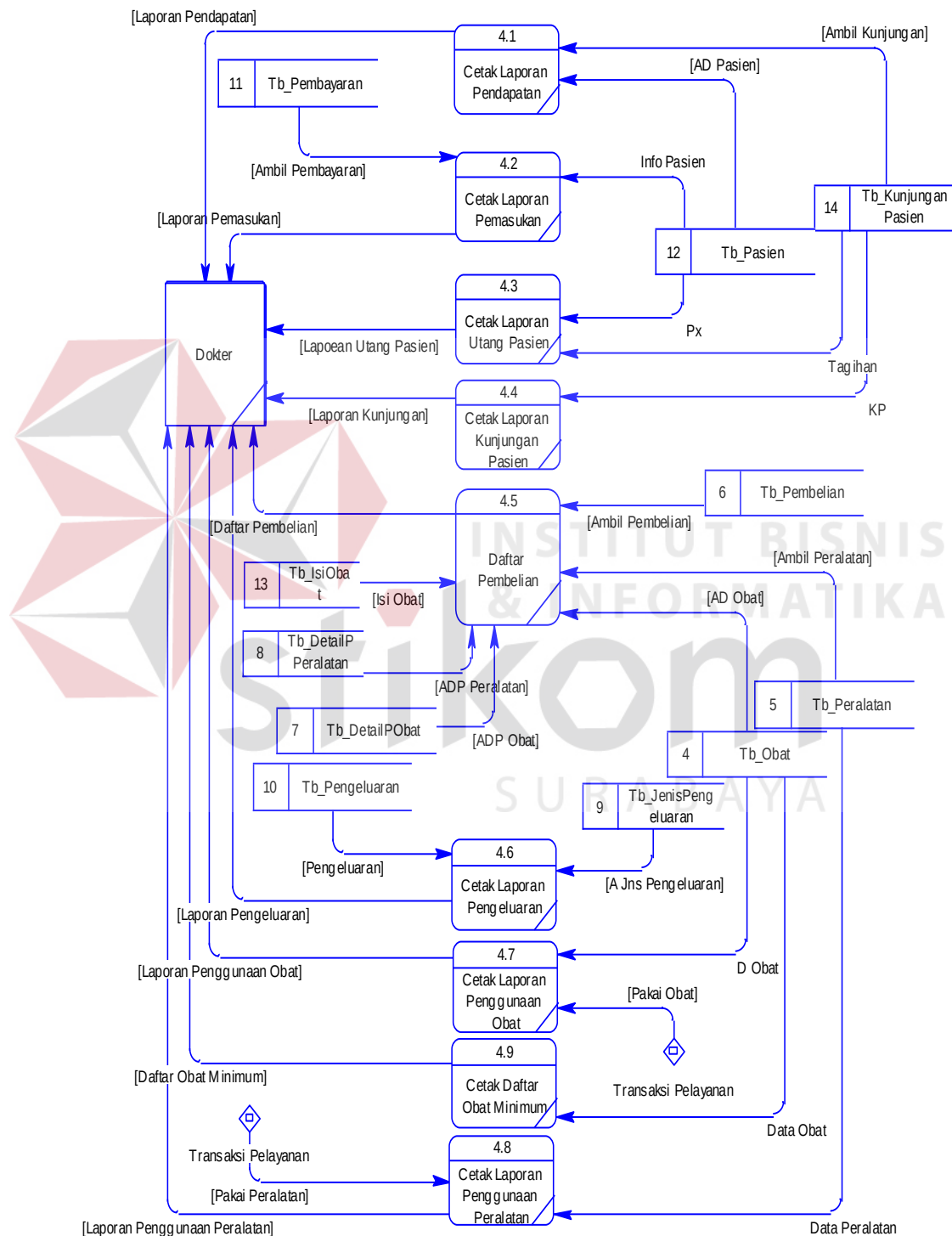
3. Proses Transaksi Pembayaran Biaya Pemeriksaan

Proses transaksi pembayaran biaya pemeriksaan menggunakan *file* kunjungan pasien dan rekam medis. Proses ini mendapat masukan dari transaksi pemeriksaan, dokter, perusahaan dan pasien yang kemudian disimpan pada *file* pembayaran.

3.6.7 DFD Level 1 Subproses Laporan

Proses sistem informasi klinik gigi drg. Basuki Surabaya dapat didekomposisi lagi menjadi DFD *level* 1 (satu) subproses laporan. Pada DFD *level* 1 (satu) ini terdapat 9 (sembilan) proses, yaitu proses cetak laporan pendapatan, cetak laporan pemasukan, cetak laporan utang pasien, cetak laporan kunjungan pasien, daftar pembelian, cetak laporan pengeluaran, cetak laporan penggunaan

obat, cetak laporan penggunaan peralatan, dan cetak daftar obat minimum. Laporan-laporan tersebut diserahkan kepada dokter. DFD *level 1* (satu) subproses laporan dapat dilihat pada Gambar 3.15.



Gambar 3.15 DFD *Level 1* Sistem Informasi Klinik Gigi Drg. Basuki Surabaya pada Subproses Laporan

Pada Gambar 3.15 di halaman 42 terdapat 9 (sembilan) proses, yaitu:

1. Proses cetak laporan pendapatan

Sistem menerima data dari *file* kunjungan pasien berupa data biaya pemeriksaan pasien yang akan menghasilkan laporan pendapatan.

2. Proses cetak laporan pemasukan

Sistem menerima data dari *file* pembayaran berupa data jumlah pembayaran biaya pemeriksaan yang akan menghasilkan laporan pemasukan.

3. Proses cetak laporan utang pasien

Sistem menerima data dari *file* kunjungan pasien berupa data utang yang akan menghasilkan laporan utang pasien.

4. Proses cetak laporan kunjungan pasien

Sistem menerima data kunjungan dari *file* kunjungan untuk diproses, setelah itu akan memberikan laporan kunjungan pasien.

5. Proses cetak laporan pembelian

Sistem menerima data dari *file* pembelian, *file* detail pembelian obat, dan *file* detail pembelian peralatan untuk diproses, setelah itu akan menghasilkan laporan pembelian.

6. Proses cetak laporan pengeluaran

Menerima data dari *file* pengeluaran dan *file* jenis pengeluaran. untuk diproses, setelah itu akan menghasilkan laporan pengeluaran.

7. Proses cetak laporan penggunaan obat

Menerima data dari *file* obat dan data penggunaan obat dari proses transaksi pemeriksaan dan pembayaran yang akan menghasilkan laporan penggunaan obat.

8. Proses cetak laopran penggunaan peralatan

Menerima data dari *file* peralatan dan data penggunaan peralatan dari proses transaksi pemeriksaan dan pembayaran yang akan menghasilkan laporan penggunaan peralatan.

9. Proses cetak daftar obat minimum

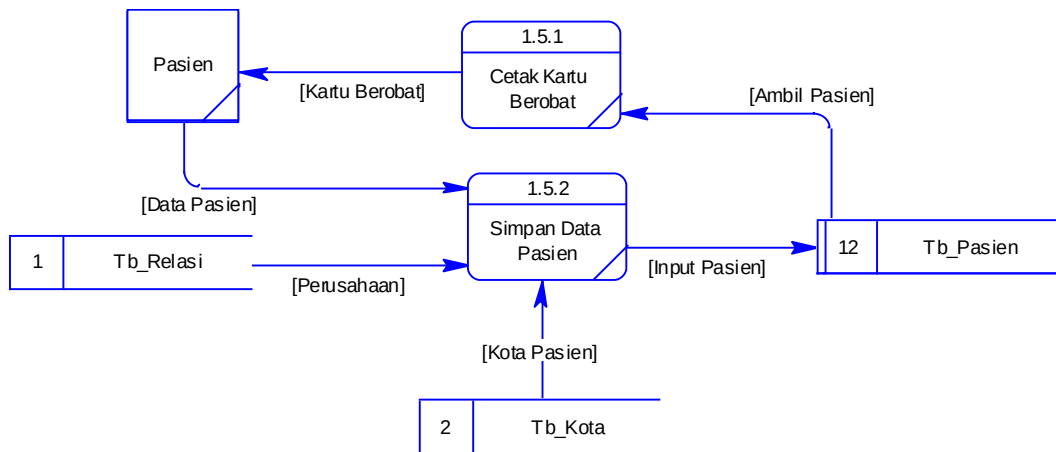
Sistem menerima data dari *file* obat yang akan menghasilkan daftar obat minimum

3.6.8 DFD Level 2 Subproses Pendataan Pasien

Proses sistem informasi klinik gigi drg. Basuki Surabaya pada DFD *level* 1 (satu) proses pendataan / master dapat didekomposisi lagi menjadi DFD *level* 2 (dua) subproses pendataan pasien. DFD *level* 2 (dua) subproses pendataan pasien dapat dilihat pada Gambar 3.16 di halaman 45. Pada DFD *level* 2 (dua) ini terdapat 2 (dua) proses, yaitu proses simpan data pasien dan proses cetak kartu berobat pasien.

Pada Gambar 3.16 di halaman 45 terdapat 2 (dua) proses yaitu:

1. Proses simpan data pasien. Data pasien menjadi masukan yang akan disimpan pada tabel pasien. Penyimpanan data pasien tidak hanya berupa data pasien baru tapi juga dapat berupa penyimpanan perubahan data pada pasien lama.
2. Proses cetak kartu berobat pasien. Data pasien pada tabel pasien dan tabel perusahaan diproses untuk menghasilkan kartu berobat yang kemudian diberikan kepada pasien untuk dibawa setiap kali pasien berkunjung.



Gambar 3.16 DFD Level 2 Sistem Informasi Klinik Gigi Drg. Basuki Surabaya pada Subproses Pendataan Pasien

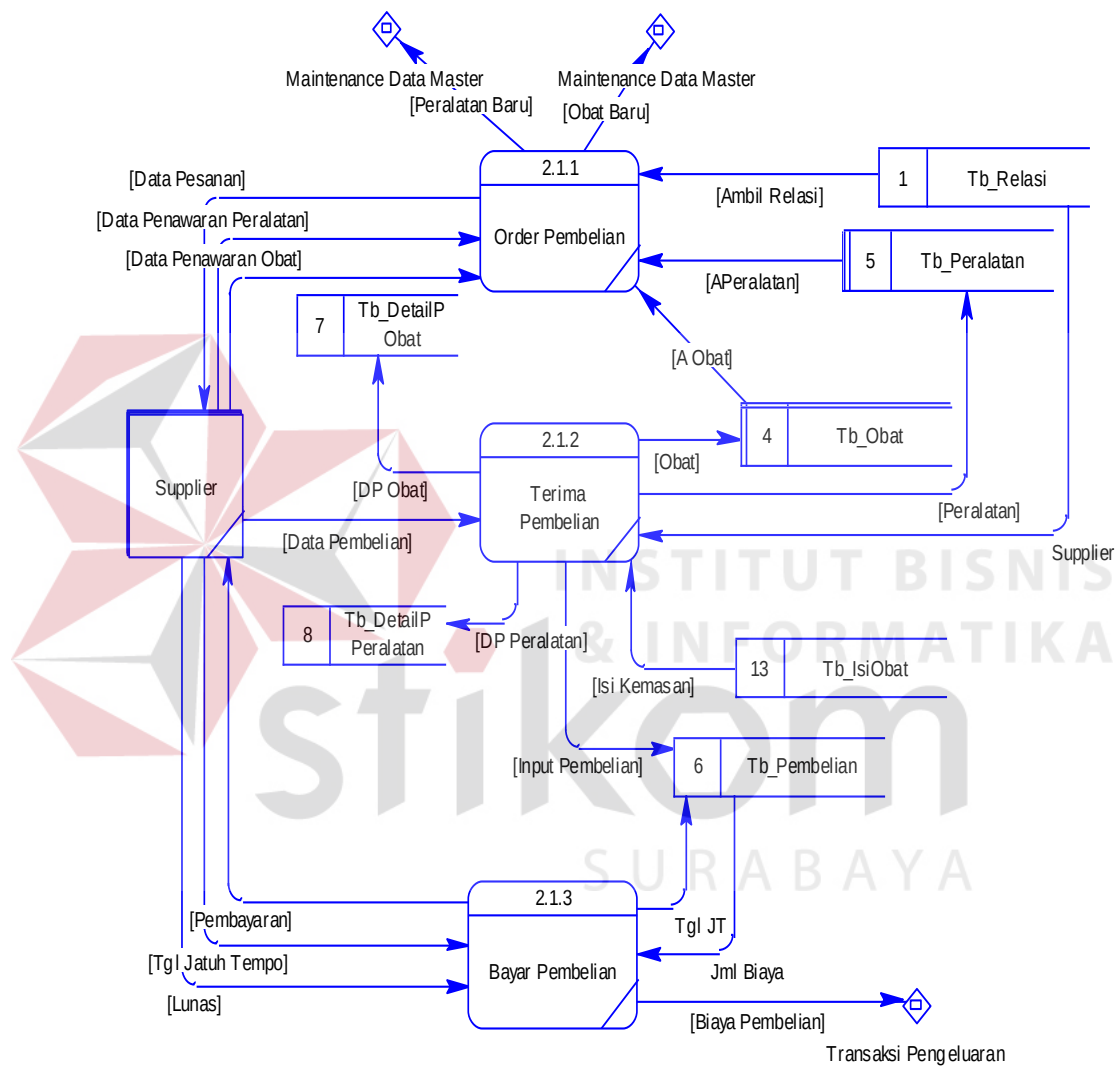
3.6.9 DFD Level 2 Subproses Transaksi Pembelian Barang

Proses sistem informasi klinik gigi drg. Basuki Surabaya pada level 1 (satu) proses transaksi pembelian barang dapat didekomposisi lagi menjadi DFD level 2 (dua) subproses transaksi pembelian barang. DFD level 2 (dua) subproses transaksi pembelian barang dapat dilihat pada Gambar 3.17 di halaman 46. Pada DFD level 2 (dua) ini terdapat 3 (tiga) proses, yaitu proses order pembelian, proses terima pembelian, dan proses bayar pembelian.

Pada Gambar 3.17 di halaman 46 terdapat 3 (tiga) proses yaitu:

1. Proses order pembelian. Data obat dari tabel obat, data peralatan dari tabel peralatan, dan data *supplier* dari tabel relasi diproses untuk menghasilkan daftar pesanan yang kemudian diberikan kepada *supplier*.
2. Proses terima pembelian. Nota pembelian menjadi masukan dari *supplier* untuk disimpan pada tabel pembelian, tabel detail pembelian obat dan tabel detail pembelian peralatan. Jika ada yang tidak sesuai maka *supplier* akan menerima informasi penolakan.

3. Proses bayar pembelian. Tanggal jatuh tempo menjadi masukan dari *supplier* untuk disimpan pada tabel pembelian. *Supplier* mendapat masukan berupa data pembayaran, dan data biaya pembelian akan disimpan melalui proses transaksi pengeluaran.



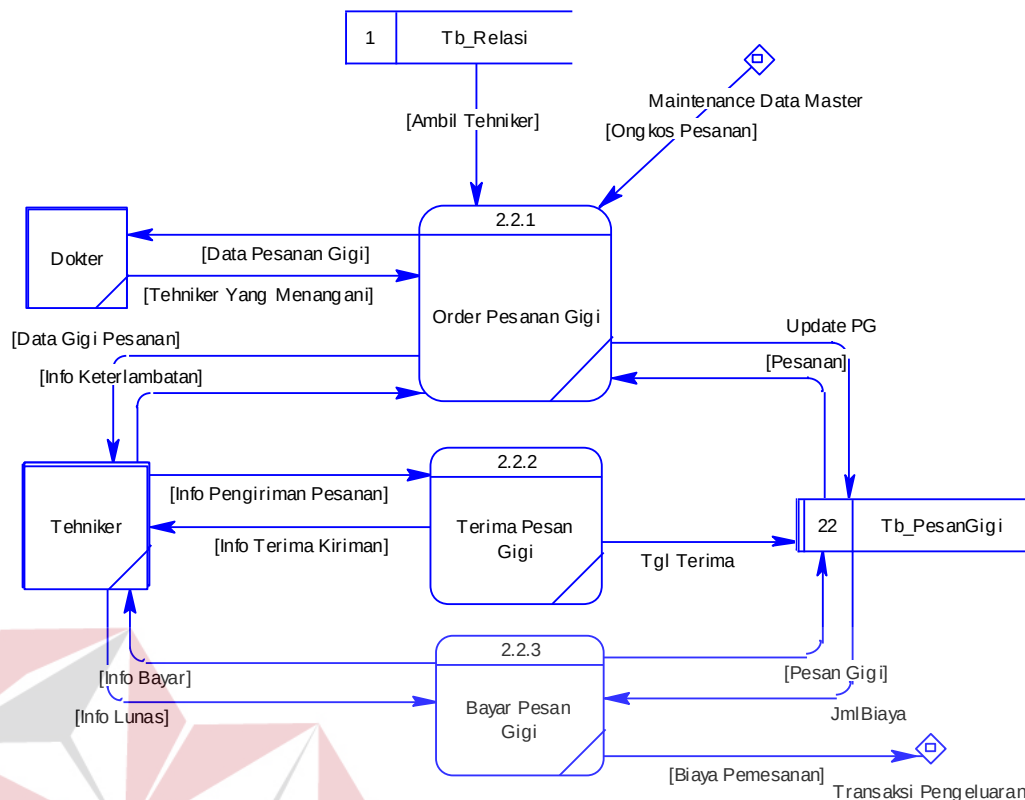
Gambar 3.17 DFD Level 2 Sistem Informasi Klinik Gigi Drg. Basuki Surabaya pada Subproses Transaksi Pembelian Barang

3.6.10 DFD Level 2 Subproses Transaksi Pesanan ke Tehniker

Proses sistem informasi klinik gigi drg. Basuki Surabaya pada DFD *level* 1 (satu) proses transaksi pesanan ke *tehniker* dapat dikomposisi lagi menjadi DFD *level* 2 (dua) subproses transaksi pesanan ke *tehniker*. DFD *level* 2 (dua) subproses transaksi pesanan ke *tehniker* dapat dilihat pada Gambar 3.18 di halaman 48. Pada DFD *level* 2 (dua) ini terdapat 3 (tiga) proses, yaitu proses order pesanan gigi, proses terima pesanan gigi, dan bayar pesanan gigi.

Pada Gambar 3.18 di halaman 48 terdapat 3 (tiga) proses yaitu:

1. Proses order pesanan gigi. Data pesanan gigi dari tabel pesan gigi menjadi masukan untuk diinformasikan ke dokter dan dokter memberi masukan berupa *tehniker* yang menangani dan menghasilkan data gigi pesanan untuk *tehniker*. Data *tehniker* dari tabel relasi dan data ongkos pesanan dari proses pendataan / *maintenance* data master kemudian diproses dan disimpan pada tabel pesan gigi.
2. Proses terima pesanan gigi. Info pengiriman pesanan menjadi inputan dari *tehniker* yang kemudian mendapat umpan balik berupa info terima kiriman. Info pengiriman pesanan diproses dan disimpan pada tabel pesanan.
3. Proses bayar pesan gigi. *Tehniker* memperoleh masukan berupa informasi biaya yang kemudian memberi umpan balik berupa info lunas. Proses ini menghasilkan biaya pemesanan yang akan diteruskan ke transaksi pengeluaran.



Gambar 3.18 DFD Level 2 Sistem Informasi Klinik Gigi Drg. Basuki Surabaya pada Subproses Transaksi Pesanan ke *Tehniker*

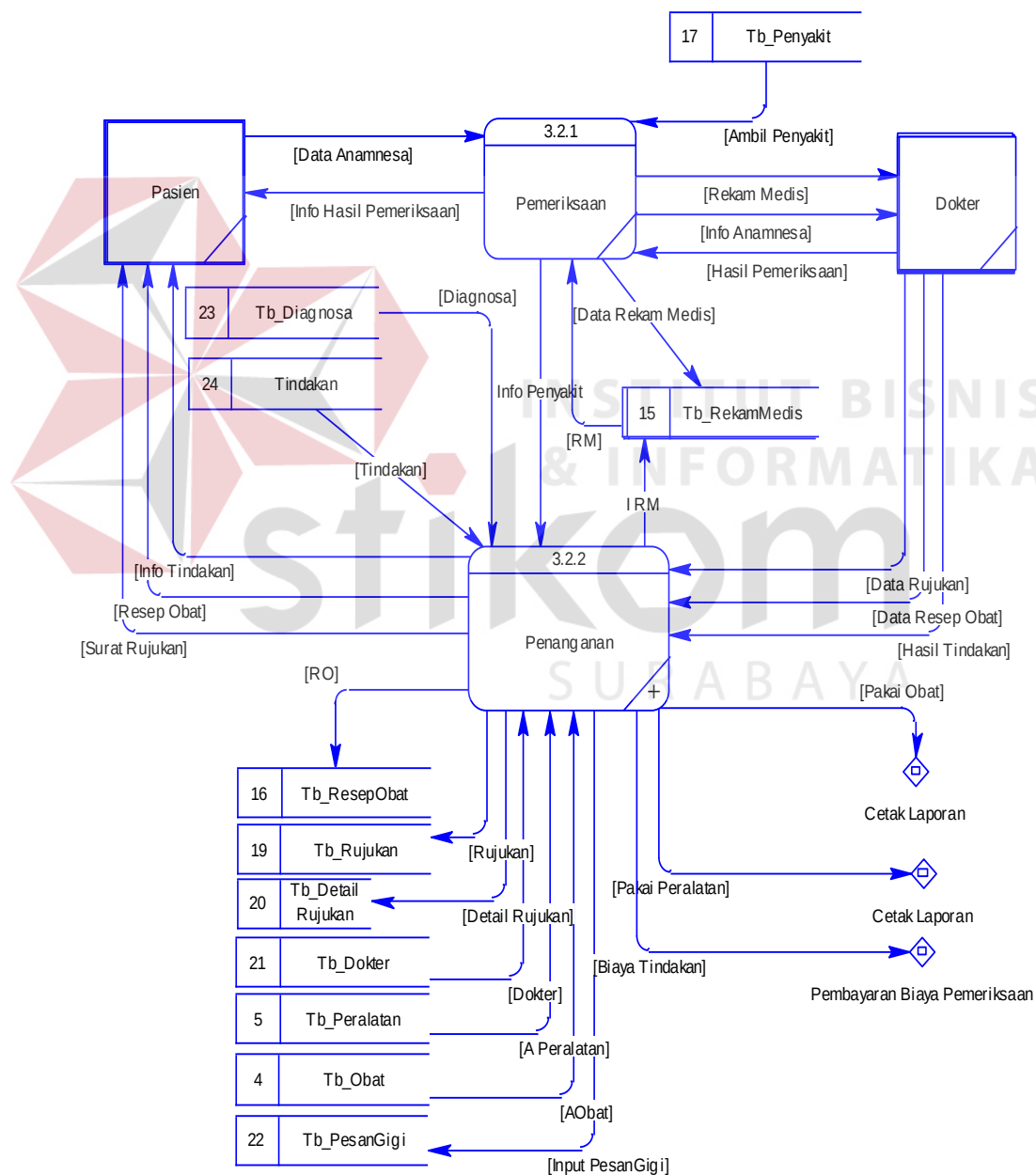
3.6.11 DFD Level 2 Subproses Transaksi Pemeriksaan

Proses sistem informasi klinik gigi drg. Basuki Surabaya pada DFD level 1 (satu) proses transaksi pemeriksaan dapat didekomposisi lagi menjadi DFD level 2 (dua) subproses transaksi pemeriksaan. DFD level 2 (dua) subproses transaksi pemeriksaan dapat dilihat pada Gambar 3.19 di halaman 49. DFD level 2 (dua) ini terdapat 2 (dua) proses, yaitu proses pemeriksaan dan proses penanganan.

Pada Gambar 3.19 di halaman 49 terdapat 2 (dua) proses yaitu:

1. Proses pemeriksaan. Proses ini menggunakan *file* penyakit, *file* rekam medis, data anamnesa dari pasien, dan hasil pemeriksaan dari dokter. Data hasil pemeriksaan akan disimpan di *file* rekam medis.

2. Proses penanganan. Proses ini menggunakan *file* dokter, *file* obat, *file* peralatan, *file* diagnosa, dan *file* tindakan. Selain itu mendapat masukan data rujukan, data resep obat, dan hasil tindakan dari dokter, dan informasi penyakit dari proses pemeriksaan. Hasil dari proses ini akan diberikan kepada pasien berupa surat rujukan, resep obat, dan informasi tindakan kemudian datanya akan disimpan pada masing-masing *file*.



Gambar 3.19 DFD Level 2 Sistem Informasi Klinik Gigi Drg. Basuki Surabaya pada Subproses Pemeriksaan

3.6.12 DFD Level 2 Subproses Pembayaran Biaya Pemeriksaan

Proses sistem informasi klinik gigi drg. Basuki Surabaya pada DFD *level* 1 (dua) proses transaksi pembayaran biaya pemeriksaan dapat didekomposisi lagi menjadi DFD *level* 2 (dua) subproses pembayaran biaya pemeriksaan dapat dilihat pada Gambar 3.20 di halaman 51. DFD *level* 2 (dua) ini terdapat 3 (tiga) proses, yaitu proses biaya pemeriksaan, proses pembayaran, dan proses cetak nota pembayaran.

Pada Gambar 3.20 di halaman 51 terdapat 3 (tiga) proses yaitu:

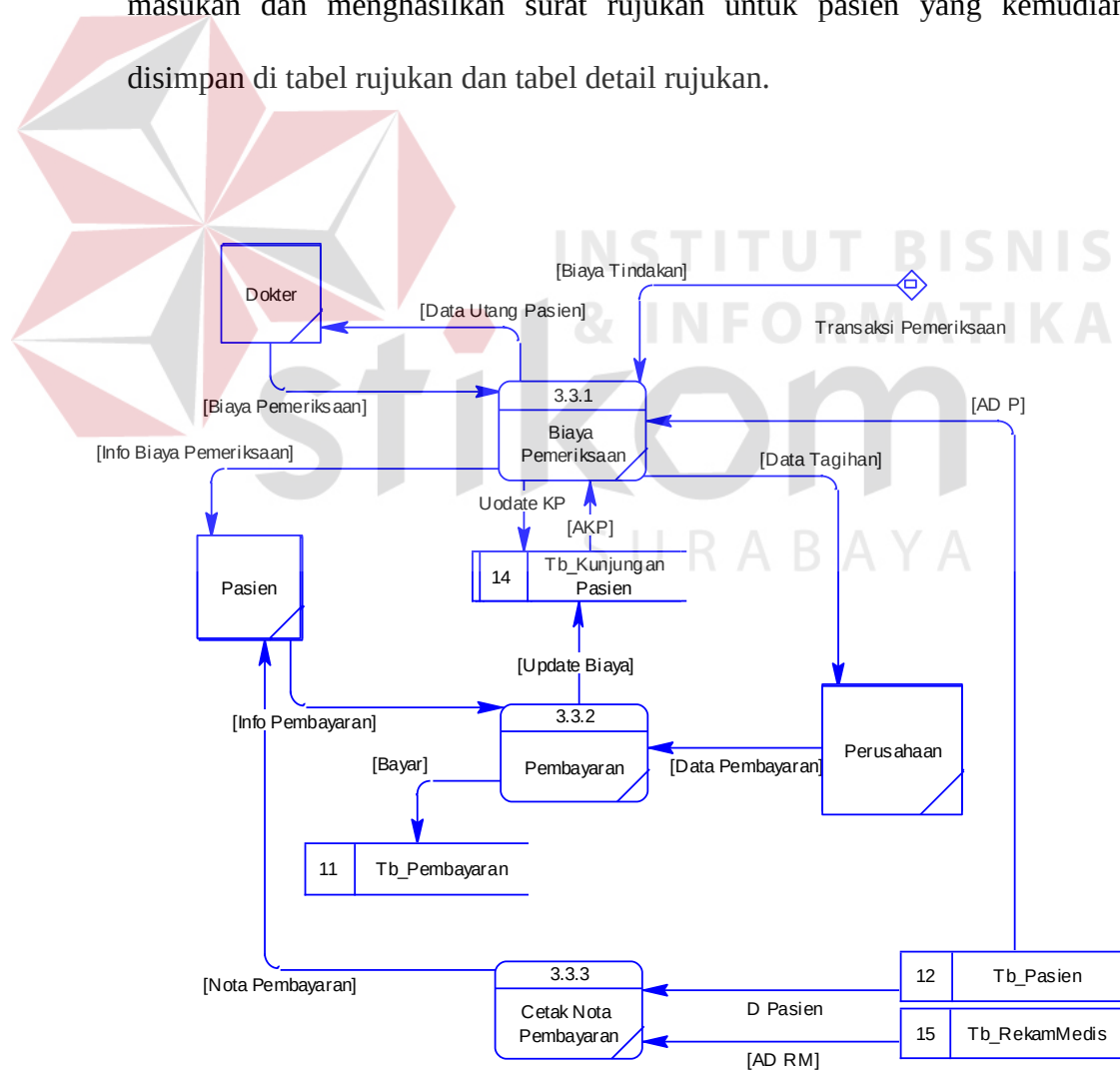
1. Proses biaya pemeriksaan. Biaya tindakan dari proses penanganan, dan biaya pemeriksaan dari dokter menjadi masukan yang kemudian disimpan pada tabel kunjungan pasien. Pasien memperoleh masukan berupa informasi biaya pemeriksaan dan perusahaan mendapat masukan berupa data tagihan.
2. Proses pembayaran. Proses ini mendapat masukan dari pasien dan perusahaan berupa data pembayaran yang kemudian disimpan pada tabel pembayaran.
3. Proses cetak nota pembayaran. Pasien mendapat masukan nota pembayaran yang dihasilkan sistem dengan memproses data pasien, data pembayaran, dan data rekam medis.

3.6.13 DFD Level 3 Subproses Penanganan

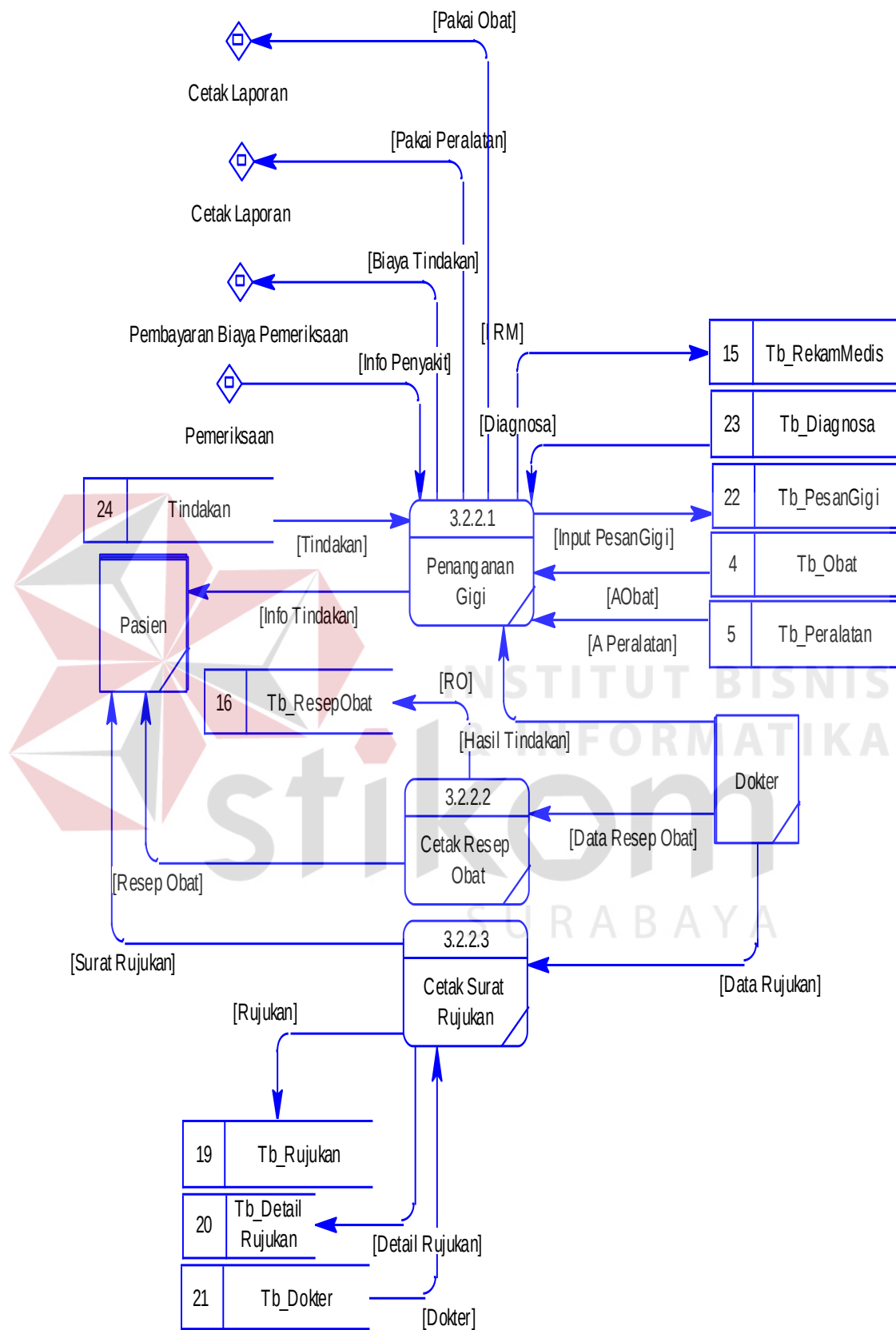
Proses sistem informasi klinik gigi drg. Basuki Surabaya pada DFD *level* 2 (dua) proses penanganan dapat didekomposisi lagi menjadi DFD *level* 3 (tiga) subproses penanganan dapat dilihat pada Gambar 3.21 di halaman 52. DFD *level* 3 (tiga) ini terdapat 3 (tiga) proses, yaitu proses penanganan gigi, proses cetak resep obat, dan proses cetak rujukan.

Pada Gambar 3.21 di halaman 52 terdapat 3 (tiga) proses yaitu:

1. Penanganan Gigi. Proses ini menggunakan tabel diagnosa, tabel tindakan, dan dokter memberi masukan hasil tindakan yang akan diberikan ke pasien dan disimpan pada tabel rekam medis.
2. Cetak Resep Obat. Data resep obat dari dokter menjadi masukan yang kemudian disimpan pada tabel resep obat dan dicetak untuk diberikan ke pasien sebagai masukan berupa cetakan resep obat
3. Cetak Surat Rujukan. Data dokter dan data rujukan dari dokter menjadi masukan dan menghasilkan surat rujukan untuk pasien yang kemudian disimpan di tabel rujukan dan tabel detail rujukan.



Gambar 3.20 DFD Level 2 Sistem Informasi Klinik Gigi Drg. Basuki Surabaya pada Subproses Transaksi Pembayaran Biaya Pemeriksaan



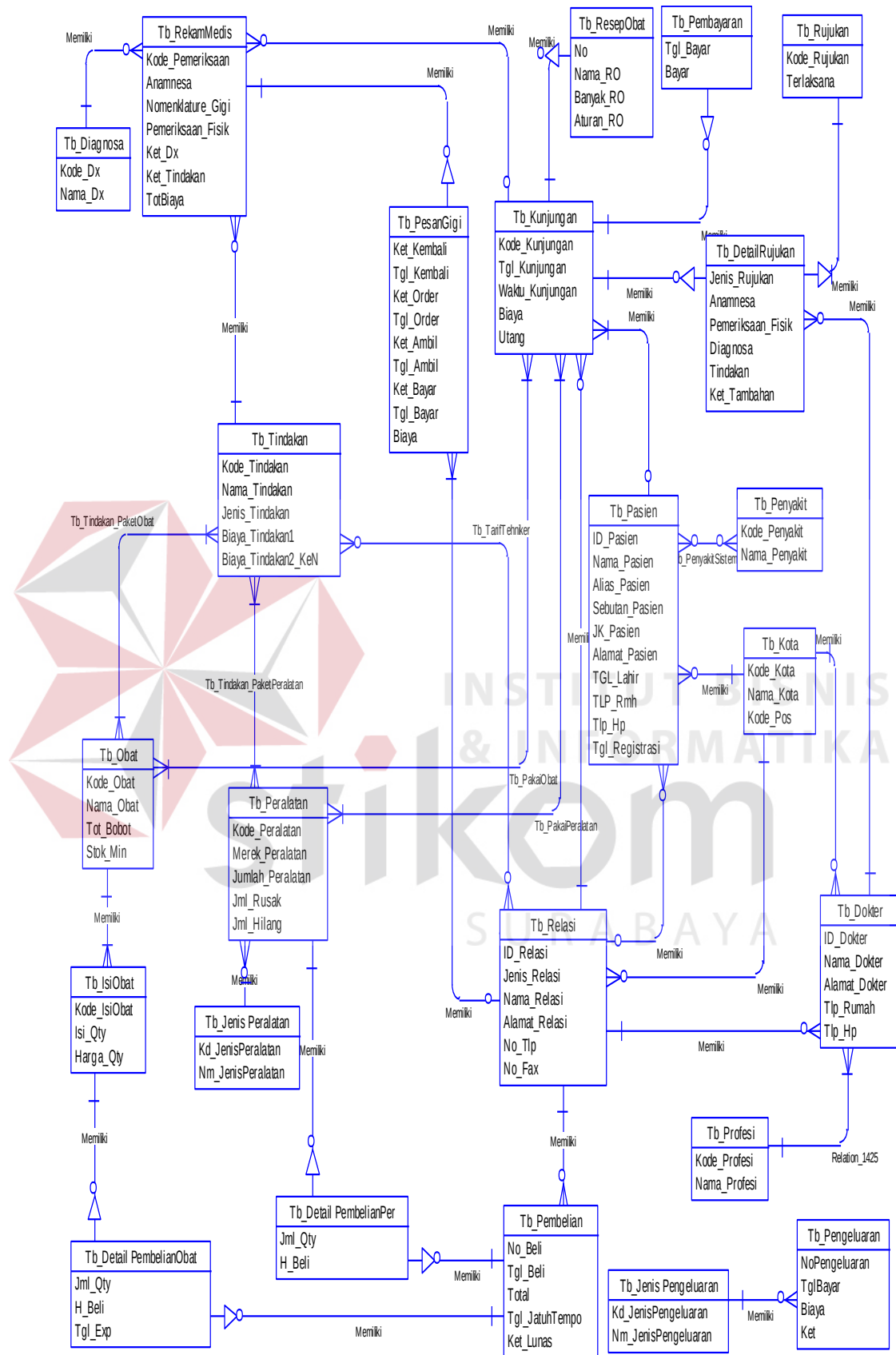
Gambar 3.21 DFD Level 3 Sistem Informasi Klinik Gigi Drg. Basuki Surabaya pada Subproses Penanganan

3.7 Entity Relationship Diagram

Entity Relationship Diagram (ERD) digunakan untuk menginterpretasikan, menentukan, dan mendokumentasikan kebutuhan-kebutuhan untuk sistem pemrosesan *database*. ERD menyediakan bentuk untuk menunjukkan struktur keseluruhan kebutuhan data dari pemakai. Dalam ERD, data tersebut digambarkan dengan menggunakan simbol entitas.

Dalam perancangan sistem ini, ada entitas yang saling terkait untuk menyediakan data yang dibutuhkan oleh sistem yang disajikan dalam bentuk *conceptual data model* (CDM) dan *physical data model* (PDM). Pada CDM Sistem Informasi Klinik Gigi Drg. Basuki Surabaya ini memiliki 24 entitas yaitu Tb_Relasi, Tb_Kota, Tb_JenisPeralatan, Tb_Obat, Tb_Peralatan, Tb_Pembelian, Tb_Detail_P_Obat, Tb_Detail_P_Peralatan, Tb_JenisPengeluaran, Tb_Pengeluaran, Tb_Pembayaran, Tb_Pasien, Tb_IsiObat, Tb_KunjunganPasien, Tb_RekamMedis, Tb_ResepObat, Tb_Penyakit, Tb_Profesi, Tb_Rujukan, Tb_DetailRujukan, Tb_Dokter, Tb_PesanGigi, Tb_Diagnosa, dan Tb_Tindakan. CDM ini akan di *generate* menjadi PDM. CDM dapat dilihat pada Gambar 3.22 di halaman 54.

Dengan meng-*generate* CDM, maka akan didapat PDM. PDM ini merupakan gambaran dari struktur *database* dari sistem informasi klinik gigi drg. Basuki Surabaya. PDM dapat dilihat pada Gambar 3.23 di halaman 55.



Gambar 3.22 CDM Sistem Informasi Klinik Gigi Drg. Basuki Surabaya

Gambar 3.23 PDM Sistem Informasi Klinik Gigi Drg. Basuki Surabaya

3.8 Struktur Database

Berikut ini terdapat Struktur Tabel dari Sistem Informasi Klinik Gigi Drg. Basuki Surabaya.

1. Tabel Tb_Kota

Primary Key : Kode_Kota

Foreign Key : -

Fungsi : Data kota.

Tabel 3.1 Kota

Nama Field	Tipe Data	Size	Keterangan
Kode_Kota	Varchar	10	Kode kota
Nama_Kota	Varchar	80	Nama kota
Kode_Pos	Varchar	15	Kode pos kota

2. Tabel Tb_Relasi

Primary Key : ID_Relasi

Foreign Key : Kode_Kota *references* Tb_Kota.Kode_Kota

Fungsi : Menyimpan data *supplier*, *tehniker* dan perusahaan yang menjadi mitra kerja klinik.

Tabel 3.2 Relasi

Nama Field	Tipe Data	Size	Keterangan
ID_Relasi	Varchar	10	Kode relasi
Jenis_Relasi	Varchar	20	Jenis relasi
Nama_Relasi	Varchar	80	Nama relasi
Kode_Kota	Varchar	10	Kode kota
Alamat_Relasi	Varchar	80	Alamat supplier
No_Tlp	Varchar	12	No telepon supplier
No_Fax	Varchar	12	No fax supplier

3. Tabel Tb_JenisPeralatan

Primary Key : Kd_JenisPeralatan

Foreign Key : -

Fungsi : Menyimpan data jenis peralatan

Tabel 3.3 Jenis Peralatan

Nama Field	Tipe Data	Size	Keterangan
Kd_JenisPeralatan	Varchar	3	Kode jenis peralatan
Nm_JenisPeralatan	Varchar	80	Nama jenis peralatan

4. Tabel Tb_Peralatan

Primary Key : Kode_Peralatan

Foreign Key : Kd_JenisPeralatan *references*

Tb_JenisPeralatan.Kd_JenisPeralatan

Fungsi : Menyimpan data peralatan

Tabel 3.4 Peralatan

Nama Field	Tipe Data	Size	Keterangan
Kode_Peralatan	Varchar	8	Kode peralatan
Kd_JenisPeralatan	Varchar	3	Kode jenis peralatan
Merek_Peralatan	Varchar	80	Merek atau tipe peralatan
Jumlah_Peralatan	Integer	-	Jumlah peralatan
Rusak	Integer	-	Jumlah rusak
Hilang	Integer	-	Jumlah hilang

5. Tabel Tb_Obat

Primary Key : Kode_Obat

Foreign Key : -

Fungsi : Menyimpan data obat

Tabel 3.5 Obat

Nama Field	Tipe Data	Size	Keterangan
Kode_Obat	Varchar	8	Kode obat
Nama_Obat	Varchar	80	Nama obat
Tot_Bobot	Integer	-	Total keseluruhan isi obat
Stok_Min	Integer	-	Stok minimum obat

6. Tabel Tb_Pembelian

Primary Key : No_Beli

Foreign Key : ID_Relasi *references* Tb_Relasi.ID_Relasi

Fungsi : Menyimpan data pembelian

Tabel 3.6 Pembelian

Nama Field	Tipe Data	Size	Keterangan
No_Beli	Varchar	14	Kode pembelian
ID_Relasi	Varchar	10	Kode relasi <i>supplier</i>
Tgl_Beli	DateTime	-	Tanggal pembelian
Total	Integer	-	Total biaya pembelian
Tgl_JatuhTempo	DateTime	-	Tanggal batas pembayaran
Ket_Lunas	Varchar	5	None atau lunas

7. Tabel Tb_Detail_P_Peralatan

Primary Key : -

Foreign Key : No_Beli *references* Tb_Pembelian.No_Beli,
Kode_Peralatan *references* Tb_Peralatan.Kode_Peralatan

Fungsi : Menyimpan data detail pembelian peralatan

Tabel 3.7 Detail Pembelian Peralatan

Nama Field	Tipe Data	Size	Keterangan
No_Beli	Varchar	14	Kode pembelian
Kode_Peralatan	Varchar	8	Kode peralatan
Jml_Qty	Integer	-	Jumlah barang
H_Beli	Integer	-	Harga pembelian peralatan

8. Tabel Tb_Detail_P_Obat

Primary Key : -

Foreign Key : No_Beli *references* Tb_Pembelian.No_Beli,
Kode_IsiObat *references* Tb_IsiObat.Kode_IsiObat

Fungsi : Menyimpan data detail pembelian obat

Tabel 3.8 Detail Pembelian Obat

Nama Field	Tipe Data	Size	Keterangan
No_Beli	Varchar	14	Kode pembelian
Kode_IsiObat	Varchar	10	Kode isi obat
Jml_Qty	Integer	-	Jumlah obat
H_Beli	Integer	-	Harga pembelian obat
Tgl_Exp	DateTime	-	Tanggal kadaluarsa obat

9. Tabel Tb_JenisPengeluaran

Primary Key : Kd_JenisPengeluaran

Foreign Key : -

Fungsi : Menyimpan data jenis pengeluaran

Tabel 3.9 Jenis Pengeluaran

Nama Field	Tipe Data	Size	Keterangan
Kd_JenisPengeluaran	Varchar	5	Kode jenis pengeluaran
Nm_JenisPengeluaran	Varchar	80	Nama jenis pengeluaran

10. Tabel Tb_Pengeluaran

Primary Key : No_Pengeluaran

Foreign Key : Kd_JenisPengeluaran *references*

Tb_JenisPengeluaran.Kd_JenisPengeluaran

Fungsi : Menyimpan data pengeluaran

Tabel 3.10 Pengeluaran

Nama Field	Tipe Data	Size	Keterangan
No_Pengeluaran	Varchar	11	Kode pengeluaran
Kd_JenisPengeluaran	Varchar	5	Kode jenis pengeluaran
Tgl_Bayar	DateTime	-	Tanggal pengeluaran
Biaya	Integer	-	Biaya pengeluaran
Ket	Varchar	Max	Keterangan pengeluaran

11. Tabel Tb_Pembayaran

Primary Key : -

Foreign Key : Kode_Kunjungan references

Tb_KunjunganPasien.Kode_Kunjungan

Fungsi : Menyimpan data pembayaran pasien

Tabel 3.11 Pembayaran

Nama Field	Tipe Data	Size	Keterangan
Kode_Kunjungan	Varchar	16	Kode kunjungan pasien
Tgl_Bayar	DateTime	-	Tanggal pembayaran
Bayar	Integer	-	Jumlah pembayaran

12. Tabel Tb_Pasien

Primary Key : ID_Pasien

Foreign Key : Kode_Kota references Tb_Kota.Kode_Kota, ID_Relasi
references Tb_Relasi.ID_Relasi

Fungsi : Menyimpan data pasien

Tabel 3.12 Pasien

Nama Field	Tipe Data	Size	Keterangan
ID_Pasien	Varchar	7	Kode pasien
ID_Relasi	Varchar	10	Jenis tanggungan pasien
Nama_Pasien	Varchar	80	Nama lengkap pasien
Alias_Pasien	Varchar	50	Nama panggilan
Sebutan_Pasien	Varchar	2	Tuan, nyonya, nona dan anak
JenisKelamin	Varchar	6	Jenis kelamin pasien
Alamat_Pasien	Varchar	80	Alamat pasien
Kode_Kota	Varchar	10	Kode kota pasien
TGL_Lahir	DateTime	-	Tanggal lahir
TLP_Rmh	Varchar	12	No telepon rumah pasien
TLP_Hp	Varchar	12	No Handphone pasien
Tgl_Registrasi	DateTime	-	Tanggal pasien mendaftar

13. Tabel Tb_IsiObat

Primary Key : Kode_IsiObat

Foreign Key : Kode_Obat *references* Tb_Obat.Kode_Obat

Fungsi : Menyimpan data bobot obat per quantity

Tabel 3.13 Isi Obat

Nama Field	Tipe Data	Size	Keterangan
Kode_IsiObat	Varchar	10	Kode isi obat
Kode_Obat	Varchar	8	Kode obat
Isi_Qty	Integer	-	Isi per qty
Harga_Qty	Integer	-	Harga per qty

14. Tabel Tb_KunjunganPasien

Primary Key : Kode_Kunjungan

Foreign Key : ID_Pasien *references* Tb_Pasien.ID_Pasien, ID_Relasi
references Tb_Relasi.ID_Relasi

Fungsi : Menyimpan data kunjungan pasien

Tabel 3.14 Kunjungan Pasien

Nama Field	Tipe Data	Size	Keterangan
Kode_Kunjungan	Varchar	16	Kode kunjungan pasien
ID_Pasien	Varchar	7	Kode pasien
ID_Relasi	Varchar	10	Jenis tanggungan pasien
Tgl_Kunjungan	DateTime	-	Tanggal kunjungan
Waktu_Kunjungan	DateTime	-	Waktu kunjungan
Biaya	Integer	-	Total biaya pemeriksaan
Utang	Integer	-	Utang pasien per kunjungan

15. Tabel Tb_RekamMedis

Primary Key : Kode_Pemeriksaan

Foreign Key : Kode_Dx *references* Tb_Diagnosa.Kode_Dx,
Kode_Tindakan *references* Tb_Tindakan.Kode_Tindakan,
Kode_Kunjungan *references*
Tb_KunjunganPasien.Kode_Kunjungan

Fungsi : Menyimpan data pemeriksaan dan tindakan medis pasien

Tabel 3.15 Rekam Medis

Nama Field	Tipe Data	Size	Keterangan
Kode_Pemeriksaan	Varchar	20	Kode pemeriksaan
Kode_Kunjungan	Varchar	16	Kode kunjungan pasien
Kode_Dx	Varchar	5	Kode diagnosa
Kode_Tindakan	Varchar	5	Kode tindakan
Anamnesa		Max	Keluhan pasien
Nomenklature_Gigi	Varchar	Max	Penomoran kode gigi
Pemeriksaan_Fisik	Varchar	Max	Uraian pemeriksaan fisik
Ket_Dx	Varchar	Max	Keterangan diagnosa
Ket_Tindakan	Varchar	Max	Keterangan tindakan pemeriksaan
TotalBiaya	Integer	-	Biaya per tindakan

16. Tabel Tb_ResepObat

Primary Key : -

Foreign Key : Kode_Kunjungan references

Tb_KunjunganPasien.Kode_Kunjungan

Fungsi : Menyimpan data resep obat pasien

Tabel 3.16 Resep Obat

Nama Field	Tipe Data	Size	Keterangan
Kode_Kunjungan	Varchar	16	Kode kunjungan pasien
Nama_RO	Varchar	80	Nama obat
Banyak_RO	Varchar	50	Dosis yang dibutuhkan
Aturan_RO	Varchar	Max	Aturan pakai atau minum

17. Tabel Tb_Penyakit

Primary Key : Kode_Penyakit

Foreign Key : -

Fungsi : Menyimpan data penyakit

Tabel 3.17 Penyakit

Nama Field	Tipe Data	Size	Keterangan
Kode_Penyakit	Varchar	7	Kode penyakit
Nama_Penyakit	Varchar	80	Nama penyakit

18. Tabel Tb_Profesi

Primary Key : Kode_Profesi

Foreign Key : -

Fungsi : Menyimpan data profesi dokter

Tabel 3.18 Profesi

Nama Field	Tipe Data	Size	Keterangan
Kode_Profesi	Varchar	5	Kode profesi
Nama_Profesi	Varchar	80	Nama profesi

19. Tabel Tb_Rujukan

Primary Key : Kode_Rujukan

Foreign Key : -

Fungsi : Menyimpan data rujukan

Tabel 3.19 Rujukan

Nama Field	Tipe Data	Size	Keterangan
Kode_Rujukan	Varchar	15	Kode rujukan
Terlaksana	Varchar	1	Terlaksana atau none

20. Tabel Tb_Dokter

Primary Key : ID_Dokter

Foreign Key : Kode_Kota *references* Tb_Kota.Kode_Kota,
 Kode_Profesi *references* Tb_Profesi.Kode_Profesi,
 Kode_Relasi *references* Tb_Relasi.Kode_Relasi

Fungsi : Menyimpan data dokter rujukan

Tabel 3.20 Dokter

Nama Field	Tipe Data	Size	Keterangan
ID_Dokter	Varchar	5	Kode dokter rujukan
Nama_Dokter	Varchar	80	Nama dokter
Alamat_Dokter	Varchar	80	Alamat rumah dokter
Kode_Kota	Varchar	10	Kota rumah dokter
Tlp_Rumah	Varchar	12	Telepon rumah dokter
Tlp_Hp	Varchar	12	No Hp
ID_Relasi	Varchar	10	Id relasi
Kode_Profesi	Varchar	5	Kode profesi

21. Tabel Tb_DetailRujukan

Primary Key : -

Foreign Key : Kode_Rujukan *references* Tb_Rujukan.Kode_Rujukan,
 Kode_Kunjungan *references*
 Tb_KunjunganPasien.Kode_Kunjungan, ID_Dokter
references Tb_Dokter.ID_Dokter

Fungsi : Menyimpan data surat rujukan

Tabel 3.21 Detail Rujukan

Nama Field	Tipe Data	Size	Keterangan
Kode_Rujukan	Varchar	15	Kode rujukan
Kode_Kunjungan	Varchar	16	Kode kunjungan pasien
ID_Dokter	Varchar	5	Kode dokter rujukan
Jenis_Rujukan	Varchar	20	Jenis rujukan
Anamnesa	Varchar	Max	Anamnesa pasien
Pemeriksaan_Fisik	Varchar	Max	Hasil pemeriksaan fisik
Diagnosa	Varchar	Max	Diagnosa dokter
Tindakan	Varchar	Max	Tindakan yang dilakukan
Ket_Tambahan	Varchar	Max	Keterangan tambahan

22. Tabel Tb_Diagnosa

Primary Key : Kode_Diagnosa

Foreign Key : -

Fungsi : Menyimpan data diagnosa

Tabel 3.22 Diagnosa

Nama Field	Tipe Data	Size	Keterangan
Kode_Diagnosa	Varchar	5	Kode diagnosa
Nama_Diagnosa	Varchar	80	Nama diagnosa

23. Tabel Tb_PesanGigi

Primary Key : Kode_Pemeriksaan

Foreign Key : Kode_Tindakan *references* Tb_Tindakan.Kode_Tindakan,
ID_Relasi *references* Tb_Relasi.ID_Relasi

Fungsi : Menyimpan data pesanan gigi pasien

Tabel 3.23 Pesan Gigi

Nama Field	Tipe Data	Size	Keterangan
Kode_Pemeriksaan	Varchar	20	Kode pemeriksaan
ID_Relasi	Varchar	10	Kode relasi
Ket_Kembali	Varchar	10	Kembali atau none
Tgl_Kembali	DateTime	-	Tanggal pesanan kembali
Ket_Orde	Varchar	10	Orde atau none
Tgl_Orde	DateTime	-	Tanggal order pesanan
Ket_Ambil	Varchar	10	Ambil atau none
Tgl_Ambil	DateTime	-	Tanggal ambil order
Ket_Bayar	Varchar	10	Bayar atau none
Tgl_Bayar	DateTime	-	Tanggal pesanan dibayar
Biaya	Integer	-	Biaya pemesanan

24. Tabel Tb_Tindakan

Primary Key : Kode_Tindakan

Foreign Key : -

Fungsi : Menyimpan data tindakan

Tabel 3.24 Tindakan

Nama Field	Tipe Data	Size	Keterangan
Kode_Tindakan	Varchar	6	Kode tindakan
Nama_Tindakan	Varchar	80	Nama tindakan
Jenis_Tindakan	Varchar	20	Kesehatan atau tehniker
Biaya_Tindakan1	Integer	-	Biaya tindakan per gigi
Biaya_Tindakan2_KeN	Integer	-	Biaya tindakan setelah gigi pertama

25. Tabel Tb_PenyakitSistemik

Primary Key : -

Foreign Key : ID_Pasien *references* Tb_Pasien.ID_Pasien,
Kode_Penyakit *references* Tb_Penyakit.KodePenyakit

Fungsi : Menyimpan data penyakit sistemik pasien

Tabel 3.25 Penyakit Sistemik

Nama Field	Tipe Data	Size	Keterangan
ID_Pasien	Varchar	7	Kode pasien
Kode_Penyakit	Varchar	7	Kode penyakit
Ket_Sistemik	Varchar	Max	Keterangan penyakit

26. Tabel Tb_TarifTehniker

Primary Key : -

Foreign Key : ID_Relasi *references* Tb_Relasi.ID_Relasi,
Kode_Tindakan *references* Tb_Tindakan.Kode_Tindakan

Fungsi : Menyimpan data tarif pekerjaan *tehniker*

Tabel 3.26 Tarif Tehniker

Nama Field	Tipe Data	Size	Keterangan
ID_Relasi	Varchar	10	Kode Relasi
Kode_Tindakan	Varchar	6	Kode tindakan
Biaya1	Integer	-	Biaya tindakan per gigi
Biaya2_KeN	Integer	-	Biaya tindakan setelah gigi pertama

27. Tabel Tb_Tindakan_PaketObat

Primary Key : -

Foreign Key : Kode_Tindakan *references* Tb_Tindakan.Kode_Tindakan,
Kode_Obat *references* Tb_Obat.Kode_Obat

Fungsi : Menyimpan penggunaan obat pertindakan

Tabel 3.27 Paket Obat

Nama Field	Tipe Data	Size	Keterangan
Kode_Tindakan	Varchar	6	Kode tindakan
Kode_Obat	Varchar	8	Kode obat
Jml_Pemakaian	Integer	-	Jumlah obat yang digunakan

28. Tabel Tb_Tindakan_PaketPeralatan

Primary Key : -

Foreign Key : Kode_Tindakan *references* Tb_Tindakan.Kode_Tindakan,
 Kode_Peralatan *references* Tb_Peralatan.Kode_Peralatan

Fungsi : Menyimpan penggunaan obat pertindakan

Tabel 3.28 Paket Peralatan

Nama Field	Tipe Data	Size	Keterangan
Kode_Tindakan	Varchar	6	Kode tindakan
Kode_Peralatan	Varchar	8	Kode peralatan
Jml_Pemakaian	Integer	-	Jumlah peralatan yang digunakan

29. Tabel Tb_PakaiObat

Primary Key : -

Foreign Key : Kode_Kunjungan *references*
 Tb_KunjunganPasien.Kode_Kunjungan, Kode_Obat
references Tb_Obat.Kode_Obat

Fungsi : Menyimpan penggunaan obat pertindakan

Tabel 3.29 Pakai Obat

Nama Field	Tipe Data	Size	Keterangan
Kode_Kunjungan	Varchar	16	Kode kunjungan pasien
Kode_Obat	Varchar	8	Kode obat
Jml_Pemakaian	Integer	-	Jumlah obat yang digunakan

30. Tabel Tb_PakaiPeralatan

Primary Key : -

Foreign Key : Kode_Kunjungan *references*

Tb_KunjunganPasien.Kode_Kunjungan, Kode_Peralatan
references Tb_Peralatan.Kode_Peralatan

Fungsi : Menyimpan penggunaan obat pertindakan

Tabel 3.30 Pakai Peralatan

Nama Field	Tipe Data	Size	Keterangan
Kode_Kunjungan	Varchar	16	Kode kunjungan pasien
Kode_Peralatan	Varchar	8	Kode peralatan
Jml_Pemakaian	Integer	-	Jumlah peralatan yang digunakan

3.9 Desain Input/Output

Pada tahap ini dilakukan perancangan *input/output* untuk berinteraksi antara *user* dengan system. Berikut adalah gambar desain *input/output* tersebut:

A. Desain Form Login

Tampilan awal saat aplikasi pertama kali dijalankan. Terdapat *textbox* nama dan kata sandi untuk *login*. Desain *form login* ditampilkan pada Gambar 3.24 di bawah.

Login	
Nama	:
Kata Sandi	:
Masuk Keluar	

Gambar 3.24 Desain Form Login

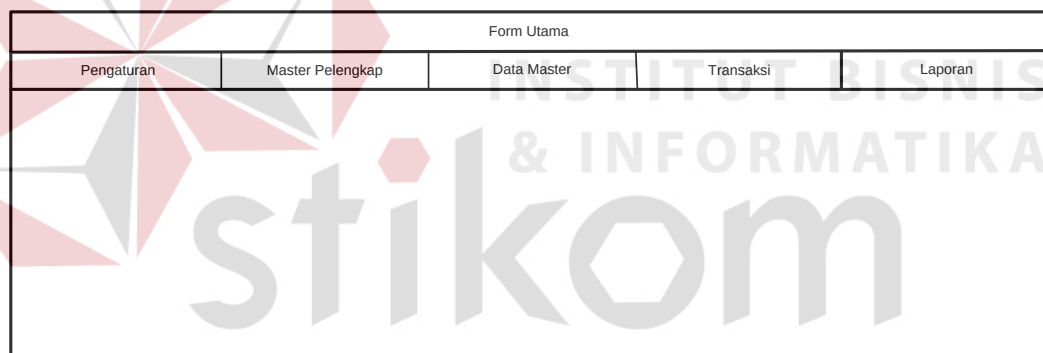
Fungsi-fungsi obyek dalam desain *form login* sebagai berikut:

Tabel 3.31 Fungsi Obyek *Form Login*

Nama Obyek	Tipe Obyek	Fungsi
Nama	<i>TextBox</i>	Diisi dengan nama pengguna yang akan menggunakan aplikasi
Kata Sandi	<i>TextBox</i>	Diisi dengan kata sandi masing-masing pengguna
Masuk	<i>Button</i>	Digunakan untuk konfirmasi Nama dan Kata Sandi
Keluar	<i>Button</i>	Digunakan untuk menutup aplikasi

B. Desain *Form Utama*

Form yang akan muncul setelah *user* berhasil melakukan login. *Form* ini berisikan menu yang dapat dipilih oleh *user*. Desain *form* utama ditampilkan pada Gambar 3.25.



Form Utama				
Pengaturan	Master Pelengkap	Data Master	Transaksi	Laporan

Gambar 3.25 Desain *Form Utama*

Fungsi-fungsi obyek dalam desain *form* utama sebagai berikut:

Tabel 3.32 Fungsi Obyek *Form Utama*

Nama Obyek	Tipe Obyek	Fungsi
Pengaturan	<i>MenuStrip</i>	Digunakan untuk memanggil <i>form</i> login, keluar aplikasi
Master Pelengkap	<i>MenuStrip</i>	Digunakan untuk memanggil <i>form</i> master jenis
Data Master	<i>MenuStrip</i>	Digunakan untuk memanggil <i>form</i> master pasien, perusahaan, dll
Transaksi	<i>MenuStrip</i>	Digunakan untuk memanggil <i>form</i> transaksi seperti <i>form</i> pembelian dan penerimaan pasien

Tabel 3.32 (Lanjutan)

Nama Obyek	Tipe Obyek	Fungsi
Laporan	<i>MenuStrip</i>	Digunakan untuk memanggil laporan kunjungan pasien, pendapatan, dll

C. Desain Form Master Kota

Form untuk melakukan proses *maintenance* kota. Pada *form* ini dapat dilakukan proses *insert* dan *edit* data yang sudah ada. Desain *form* master kota dapat dilihat pada Gambar 3.26.

Gambar 3.26 Desain *Form* Master Kota

Fungsi-fungsi obyek dalam desain *form* master kota sebagai berikut:

Tabel 3.33 Fungsi Obyek *Form* Master Kota

Nama Obyek	Tipe Obyek	Fungsi
Kode Kota	<i>TextBox</i>	Di <i>generate</i> otomatis oleh sistem. Menunjukkan Kode Kota untuk kota baru
Nama Kota	<i>TextBox</i>	Diisi dengan nama kota
Kode Pos	<i>TextBox</i>	Diisi dengan kode pos kota
Baru	<i>Button</i>	Digunakan untuk membersihkan <i>textbox</i>
Simpan	<i>Button</i>	Digunakan untuk menyimpan data kota baru dan menyimpan perubahan data kota
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> Master Kota
Tampilkan Semua	<i>CheckBox</i>	Digunakan untuk mengatur keluaran data di <i>datagridview</i>
Cari Berdasarkan	<i>ComboBox</i>	Digunakan untuk memilih kategori untuk mencari data

Tabel 3.33 (Lanjutan)

Nama Obyek	Tipe Obyek	Fungsi
Cari Berdasarkan	<i>TextBox</i>	Diisi dengan karakter yang akan digunakan untuk mencari data
Proses	<i>Button</i>	Digunakan untuk memproses “Cari Berdasarkan”
Tabel Data Relasi	<i>DataGridView</i>	Untuk menampilkan data-data kota yang ada pada <i>database</i>

D. Desain Form Master Jenis

Desain *form* ini digunakan sebagai acuan untuk membuat *form-form* proses *maintenance* data jenis. *Form-form* yang menggunakan desain *form* ini yaitu, *Form* Penyakit, *Form* Master Jenis Pengeluaran, *Form* Master Jenis Peralatan, dan *Form* Master Diagnosa. *Form* ini digunakan untuk *insert* data dan *edit* data yang sudah ada. Desain *form* master jenis dapat dilihat pada Gambar 3.27.

Gambar 3.27 Desain *Form* Master Jenis

Fungsi-fungsi obyek dalam desain *form* master jenis sebagai berikut:

Tabel 3.34 Fungsi Obyek *Form* Master Jenis

Nama Obyek	Tipe Obyek	Fungsi
Kode Jenis	<i>TextBox</i>	Di <i>generate</i> otomatis oleh sistem. Menunjukan Kode jenis untuk data jenis baru
Nama Jenis	<i>TextBox</i>	Diisi dengan nama jenis
Baru	<i>Button</i>	Digunakan untuk membersihkan <i>textbox</i>

Tabel 3.34 (Lanjutan)

Nama Obyek	Tipe Obyek	Fungsi
Simpan	<i>Button</i>	Digunakan untuk menyimpan data jenis baru dan menyimpan perubahan data jenis
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> Master Jenis
Tampilkan Semua	<i>CheckBox</i>	Digunakan untuk mengatur keluaran data di <i>datagridview</i> data jenis
Cari Berdasarkan	<i>ComboBox</i>	Digunakan untuk memilih kategori untuk mencari data
Cari Berdasarkan	<i>TextBox</i>	Diisi dengan karakter yang akan digunakan untuk mencari data
Proses	<i>Button</i>	Digunakan untuk memproses “Cari Berdasarkan”
Data Kode dan nama	<i>DataGrid View</i>	Menampilkan data jenis yang tersimpan di <i>database</i>

E. Desain Form Master Relasi

Desain *form* ini digunakan sebagai acuan untuk membuat *form-form* proses *maintenance* data para relasi klinik. *Form-form* yang menggunakan desain ini yaitu, *Form* Master Perusahaan, *Form* Master Supplier, dan *Form* Master Tehniker. *Form* ini digunakan untuk *insert* data baru dan *edit* data yang sudah ada. Desain *form* master relasi dapat dilihat pada Gambar 3.28.

Gambar 3.28 Desain Form Master Relasi

Fungsi-fungsi obyek dalam desain *form* master relasi sebagai berikut:

Tabel 3.35 Fungsi Obyek *Form* Master Relasi

Nama Obyek	Tippe Obyek	Fungsi
ID Relasi	<i>TextBox</i>	Di <i>generate</i> otomatis oleh sistem. Menunjukkan Kode Relasi untuk relasi baru
Nama Relasi	<i>TextBox</i>	Diisi dengan nama relasi
Alamat Relasi	<i>TextBox</i>	Diisi dengan alamat relasi
Kota Relasi	<i>ComboBo</i>	Digunakan untuk memilih kota yang ada
No Telepon	<i>TextBox</i>	Diisi dengan nomor telepon relasi
No Fax	<i>TextBox</i>	Diisi dengan nomor fax relasi
Baru	<i>Button</i>	Digunakan untuk membersihkan <i>textbox</i>
Simpan	<i>Button</i>	Digunakan untuk menyimpan data relasi baru dan menyimpan perubahab data relasi
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> Master Relasi
Tampilkan Semua	<i>CheckBox</i>	Digunakan untuk mengatur keluaran data di <i>datagridview</i>
Cari Berdasarkan	<i>ComboBox</i>	Digunakan untuk memilih kategori untuk mencari data
Cari Berdasarkan	<i>TextBox</i>	Diisi dengan karakter yang akan digunakan untuk mencari data
Proses	<i>Button</i>	Digunakan untuk memproses “Cari Berdasarkan”
Tabel Data Relasi	<i>DataGridView</i>	Untuk menampilkan data-data relasi yang ada pada <i>database</i>

F. Desain Form Master Pasien

Form ini untuk melakukan proses *maintenance* pasien. Pada *form* ini dapat dilakukan proses *insert* dan *edit* data yang sudah ada. Daftar Pasien yang ada pada *database* dapat dilihat pada *gridview* yang terdapat pada *form*. *Form* Master Pasien dapat dilihat pada Gambar 3.29 di halaman 74.

Data Master Pasien

ID Pasien : [ID Pasien]

Nama :

Alias :

Sebutan :

Jenis Kelamin : ☒ Pria ☐ Wanita

Alamat :

Kota :

Tanggal Lahir : dd MMMM yyyy

Umur : [Tahun]

Jenis Pasien :

Tlp Rumah :

Tlp HP :

Penyakit Sistemik

Penyakit

[Add](#) [Keterangan](#)

Baru

Simpan

Hapus

Tutup

Cetak Kartu Berobat

Cetak Rekam Medis

Detail Pemeriksaan

☐ Tampilkan Semua Cari Berdasarkan :

ID Pasien	Nama Pasien	Alias	Sebutan	JK	Alamat	Kota	Tanggal

Gambar 3.29 Desain Form Master Pasien

Fungsi-fungsi obyek dalam desain form master pasien sebagai berikut:

Tabel 3.36 Fungsi Obyek Form Master Pasien

Nama Obyek	Tipe Obyek	Fungsi
ID Pasien	Label	Di generate otomatis oleh sistem. Menunjukan Kode pasien untuk pasien baru
Nama	TextBox	Diisi dengan nama lengkap pasien
Alias	TextBox	Diisi dengan nama panggilan atau nama sapaan pasien
Sebutan	ComboBox	Berisi pilihan anak, nona, nyonya dan tuan
Jenis Kelamin	RadioButton	Berisi jenis kelamin pasien
Alamat	TextBox	Diisi dengan alamat pasien
Kota	ComboBo	Digunakan untuk memilih kota yang ada
Tanggal Lahir	DateTimePicker	Diisi dengan tanggal lahir pasien
Umur	Label	Dihasilkan dari perubahan yang terjadi pada tanggal lahir. Hasilnya berupa usia dari pasien
Jenis Pasien	ComboBox	Berisi jenis pasien umum atau jaminan social tenaga kerja
Tlp Rumah	TextBox	Diisi dengan nomor telepon rumah pasien
Tlp HP	TextBox	Diisi dengan nomor Hp pasien
Baru	Button	Digunakan untuk membersihkan textbox
Simpan	Button	Digunakan untuk menyimpan data pasien baru dan menyimpan perubahab data pasien
Hapus	Button	Digunakan untuk menghapus data pasien
Tutup	Button	Digunakan untuk keluar dari form Master Pasien

Tabel 3.36 (Lanjutan)

Nama Obyek	Tipe Obyek	Fungsi
Cetak Kartu Berobat	<i>Button</i>	Digunakan untuk proses cetak kartu berobat
Cetak Rekam Medis	<i>Button</i>	Digunakan untuk proses cetak rekam medis
Detail Pemeriksaan	<i>Button</i>	Digunakan untuk melihat data rekam medis
Tampilkan Semua	<i>CheckBox</i>	Digunakan untuk mengatur keluaran data di <i>datagridview</i> data pasien
Cari Berdasarkan	<i>ComboBox</i>	Digunakan untuk memilih kategori untuk mencari data
Cari Berdasarkan	<i>TextBox</i>	Diisi dengan karakter yang akan digunakan untuk mencari data
Proses	<i>Button</i>	Digunakan untuk proses cari
Penyakit	<i>DataGridView</i>	Untuk menampilkan data penyakit sistemik pasien yang ada pada <i>database</i>
Add	<i>LinkLabel</i>	Digunakan untuk proses tambah data penyakit sistemik pasien
Keterangan	<i>LinkLabel</i>	Digunakan untuk menambah atau mengubah keterangan penyakit sistemik pasien

G. Desain Form Input Pengeluaran

Form Input Pengeluaran digunakan untuk memasukan data transaksi pengeluaran klinik. Desain *form input* pengeluaran dapat dilihat pada Gambar 3.30.

Input Pengeluaran					
No Pengeluaran : [No Pengeluaran]	No Pengeluaran	Pengeluaran	Biaya	Tgl Pengeluaran	Keterangan
Pengeluaran :					
Biaya :					
Tanggal :					
Keterangan :					
Baru Simpan Tutup Daftar Pengeluaran					

Gambar 3.30 Desain *Form Input Pengeluaran*

Fungsi-fungsi obyek dalam desain *form input* pengeluaran sebagai berikut:

Tabel 3.37 Fungsi Obyek *Form Input* Pengeluaran

Nama Obyek	Tipe Obyek	Fungsi
No. Pengeluaran	<i>Label</i>	Di <i>generate</i> otomatis oleh sistem. Menunjukkan Kode No. Pembelian untuk data pembelian baru.
Pengeluaran	<i>ComboBox</i>	Berise jenis-jenis pengeluaran.
Biaya	<i>NumericUpDown</i>	Diisi dengan nilai biaya pengeluaran
Tanggal	<i>DateTimePicker</i>	Diisi dengan tanggal terjadinya pengeluaran
Keterangan	<i>TextBox</i>	Diisi dengan keterangan pengeluaran
Baru	<i>Button</i>	Digunakan untuk membersihkan <i>textbox</i>
Simpan	<i>Button</i>	Digunakan untuk menyimpan data pengeluaran ke <i>database</i>
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form Input</i> Pengeluaran
Daftar Pengeluaran	<i>Button</i>	Digunakan untuk membuka <i>form</i> yang berisikan informasi pengeluaran sebelumnya
Tabel	<i>DataGridView</i>	Menampilkan data pengeluaran per hari yang tersimpan di <i>database</i>

H. *Form Input* Pembelian

Form Input Pembelian digunakan untuk memasukan data transaksi pembelian obat dan peralatan yang terjadi di klinik. Desain *form input* pembelian dapat dilihat pada Gambar 3.31.

Gambar 3.31 Desain *Form Input* Pembelian

Fungsi-fungsi obyek dalam desain *form input* pembelian sebagai berikut:

Tabel 3.38 Fungsi Obyek *Form Input* Pembelian

Nama Obyek	Tipenya	Fungsi
No. Pembelian	<i>Label</i>	Di <i>generate</i> otomatis oleh sistem. Menunjukkan Kode No Pembelian untuk data pembelian baru
Tgl. Pembelian	<i>Label</i>	Mengikuti tanggal sistem
ID Supplier	<i>Label</i>	Diisi dengan ID <i>Supplier</i> yang dipilih melalui Nama <i>Supplier</i>
Nama Supplier	<i>ComboBox</i>	Digunakan untuk memilih <i>supplier</i> yang ada
Alamat Supplier	<i>Label</i>	Diisi dengan Alamat <i>Supplier</i> yang dipilih melalui Nama <i>Supplier</i>
Kode Item	<i>Label</i>	Diisi dengan kode obat yang dipilih melalui Nama
Nama	<i>ComboBox</i>	Digunakan untuk memilih obat yang ada
Isi / Netto	<i>ComboBox</i>	Digunakan untuk memilih isi obat yang ada
Jumlah	<i>NumericUpDown</i>	Diisi dengan banyaknya item yang dibeli
Harga/satuan	<i>NumericUpDown</i>	Diisi dengan harga beli item
Tgl. Kadaluarsa	<i>DateTimePicker</i>	Diisi dengan tanggal kadaluarsa obat
Obat Peralatan	<i>TabPage</i>	Digunakan untuk memilih obat atau peralatan
OK	<i>Button</i>	Digunakan untuk mengisi Detail Pembelian berdasarkan Item yang dipilih
Detail Pembelian	<i>DataGridView</i>	Berisi data pembelian per-transaksi
Total	<i>TextBox</i>	Diisi secara otomatis. Berisi total biaya pembelian per-transaksi
Baru	<i>Button</i>	Digunakan untuk membersihkan <i>text box</i> dan table Detail Pembelian
Batal	<i>Button</i>	Digunakan untuk membersihkan table Detail Pembelian
Simpan	<i>Button</i>	Digunakan untuk menyimpan Detail Pembelian ke <i>database</i>
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form Input</i> Pembelian
Tunai	<i>Button</i>	Digunakan untuk memproses pembayaran secara tunai
Angsuran	<i>Button</i>	Digunakan untuk membuka <i>form</i> untuk proses pembayaran secara angsuran
Lihat Data Pembelian	<i>LinkLabel</i>	Digunakan untuk membuka <i>form</i> Data Pembelian yang berisikan data transaksi pembelian yang pernah terjadi

I. Desain Form Input Kunjungan Pasien

Form Input Kunjungan Pasien ini digunakan untuk memaukan data kunjungan pasien. Desain *form* ini dapat dilihat pada Gambar 3.32.

Gambar 3.32 Desain Form Input Kunjungan Pasien

Fungsi-fungsi obyek dalam desain *form Input* Kunjungan Pasien sebagai berikut:

Tabel 3.39 Fungsi Obyek Form Input Kunjungan Pasien

Nama Obyek	Tipe Obyek	Fungsi
Input Berdasarkan	<i>ComboBox</i>	Berisi pilihan kategori pencarian
Input Berdasarkan	<i>TextBox</i>	Diisi dengan karakter yang dicari
OK	<i>Button</i>	Digunakan untuk menyimpan data pasien yang berkunjung berdasarkan “Input Berdasarkan”
Tanggal	<i>Label</i>	Mengikuti tanggal sistem
Waktu	<i>Label</i>	Mengikuti jam system
NO	<i>Label</i>	Diisi dengan NO urut pasien yang sedang diperiksa
Periksa	<i>Button</i>	Digunakan untuk membuka <i>form</i> Pemeriksaan
Kartu Berobat	<i>Button</i>	Digunakan untuk membuka <i>form</i> cetak kartu berobat
Baru	<i>Button</i>	Digunakan untuk membuka <i>form</i> Master Pasien untuk melakukan proses <i>input</i> data pasien baru
Ubah	<i>Button</i>	Digunakan untuk membuka <i>form</i> Master Pasien untuk melakukan proses ubah data pasien
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form Input</i> Kunjungan Pasien

J. Desain Form Data Pemeriksaan

Form Data Pemeriksaan digunakan untuk menampilkan data pemeriksaan atau rekam medis, *input* resep obat, dan *input* rujukan. Desain form data pemeriksaan dapat dilihat pada Gambar 3.33.

Gambar 3.33 Desain Form Data Pemeriksaan

Fungsi-fungsi obyek dalam desain form data pemeriksaan sebagai berikut:

Tabel 3.40 Fungsi Obyek Form Data Pemeriksaan

Nama Obyek	Tipe Obyek	Fungsi
ID Pasien	Label	Secara otomatis terisi dengan ID Pasien
Nama Pasien	Label	Secara otomatis terisi dengan nama pasien
Jenis Klamin	Label	Terisi dengan “Peria” atau “Wanita”
Jenis Pasien	Label	Secara otomatis terisi dengan jenis tanggungan pasien
Tanggal Lahir	Label	Secara otomatis terisi dengan tanggal lahir
Usia	Label	Diisi dengan usia pasien
Add	LinkLabel	Digunakan untuk proses tambah data penyakit sistemik pasien
Keterangan	LinkLabel	Digunakan untuk menambah atau mengubah keterangan penyakit sistemik pasien
Total Biaya	NumericUpDown	Diisi dengan total biaya pemeriksaan
Utang	NumericUpDown	Diisi dengan utang pasien
Tanggal	ComboBox	Digunakan untuk memilih tanggal pemeriksaan pasien
Tampilkan Semua	CheckBox	Digunakan untuk mengatur keluaran data di <i>datagridview</i> data pemeriksaan
Rekam Medis Resep Obat Rujukan	Tab Page	Digunakan untuk berpindah halaman rekam medis, resep obat, dan rujukan

Tabel 3.40 (Lanjutan)

Nama Obyek	Tipe Obyek	Fungsi
Keluhan	<i>Button</i>	Digunakan untuk membuka <i>form</i> pemeriksaan
Bayar	<i>Button</i>	Digunakan untuk membuka <i>form</i> pembayaran
Cetak Rekam Medis	<i>Button</i>	Digunakan untuk membuka <i>form</i> cetak rekam medis
Cetak Nota Pembayaran	<i>Button</i>	Digunakan untuk membuka <i>form</i> cetak nota pembayaran
Tutup	<i>Button</i>	Digunakan untuk menutup <i>form</i> data pemeriksaan

K. Desain Form Pemeriksaan

Form Pemeriksaan digunakan untuk *insert* dan *delete* data pemeriksaan pada tanggal pemeriksaan. Desain *form* pemeriksaan dapat dilihat pada Gambar 3.34.

The form is titled 'Form Pemeriksaan' and is designed for data entry. It features a sidebar on the left for 'Anamnesa' and a main area on the right for 'Pelayanan & Odonto'. The 'Anamnesa' sidebar contains several input fields: 'Gigi', 'Pemeriksaan Fisik / Penunjang Medis', 'Diagnosa', 'Pengobatan / Tindakan', 'Keterangan Tindakan' (with radio buttons for 'Pesan' and 'Kembali'), and 'Biaya'. The 'Pelayanan & Odonto' section has a 'Perawatan' tab, a 'Nomenklature Gigi' field, a large text area for notes, and a 'Total Biaya (Rp)' field. A 'Tutup' button is located at the bottom right. A watermark for 'STIKOM SURABAYA' is visible across the center of the form.

Gambar 3.34 Desain *Form* Pemeriksaan

Fungsi-fungsi obyek dalam desain *form* pemeriksaan sebagai berikut:

Tabel 3.41 Fungsi Obyek *Form* Pemeriksaan

Nama Obyek	Tipe Obyek	Fungsi
Anamnesa	<i>TextBox</i>	Diisi dengan keluhan pasien
Gigi	<i>TextBox</i>	Terisi secara otomatis saat Nomenklature Gigi dipilih
Pemeriksaan Fisik	<i>TextBox</i>	Diisi dengan pemeriksaan fisik
Diagnosa	<i>ComboBox</i>	Berisi data-data diagnosa
Diagnosa	<i>TextBox</i>	Diisi dengan keterangan diagnosa
Tindakan	<i>ComboBox</i>	Berisi data-data tindakan
Tindakan	<i>TextBox</i>	Diisi dengan keterangan tindakan
Keterangan Tindakan	<i>RadioButton</i>	Memilih “Pesan” atau “Kembali”
Biaya	<i>NumericUpDown</i>	Diisi dengan biaya pemeriksaan
Biaya Ke 2	<i>CheckBox</i>	Digunakan untuk memilih biaya
Baru	<i>Button</i>	Digunakan untuk membersihkan <i>textbox</i>
Tambah	<i>Button</i>	Digunakan untuk menambah data ke tabel pemeriksaan
Batal	<i>Button</i>	Digunakan untuk membatalkan data yang ada di tabel pemeriksaan
Pelayanan Perawatan	<i>Tab Page</i>	Digunakan untuk berpindah halaman pelayanan atau perawatan
Nomenklature Gigi	<i>GroupBox</i>	Berisikan penomoran gigi
Pemeriksaan	<i>DataGridView</i>	Berisi data pemeriksaan
Total Biaya	<i>NumericUpDown</i>	Berisi Total biaya pemeriksaan
Tutup	<i>Button</i>	Digunakan untuk menutup <i>Form</i> Pemeriksaan

L. Desain *Form* Pembayaran Biaya Pemeriksaan

Form Pembayaran Biaya Pemeriksaan digunakan untuk *insert* data pembayaran biaya pemeriksaan. Desain *form* pembayaran biaya pemeriksaan dapat dilihat pada Gambar 3.35 di halaman 82.

Pembayaran

Total Biaya (Rp) : Bayar (Rp) :

Utang (Rp) : Kembali (Rp) :

Daftar Utang

Tanggal Kunjungan	Biaya	Utang

Riwayat Pembayaran

Tanggal Bayar	Bayar

Gambar 3.35 Desain *Form* Pembayaran Biaya Pemeriksaan

Fungsi-fungsi obyek dalam desain *form* pembayaran biaya pemeriksaan sebagai berikut:

Tabel 3.42 Fungsi Obyek *Form* Pembayaran Biaya Pemeriksaan

Nama Obyek	Tippe Obyek	Fungsi
Total Biaya	<i>TextBox</i>	Berisi nilai biaya pemeriksaan
Utang	<i>TextBox</i>	Berisi nilai utang pasien
Bayar	<i>NumericUpDown</i>	Diisi dengan nilai pembayaran
Kembali	<i>TextBox</i>	Berisi nilai kembali dari pembayaran
Bayar	<i>Button</i>	Digunakan untuk menyimpan data pembayaran ke <i>database</i>
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> pembayaran biaya pemeriksaan
Daftar Utang	<i>DataGridView</i>	Berisi data utang
Riwayat Pembayaran	<i>DataGridView</i>	Berisi data pembayaran

M. Desain *Form* Cetak Nota Pembayaran

Form ini digunakan untuk menampilkan data pembayaran biaya pemeriksaan pada tanggal tertentu. Desain *form* cetak nota pembayaran dapat dilihat pada Gambar 3.36 di halaman 83.

Gambar 3.36 Desain *Form* Cetak Nota Pembayaran

Fungsi-fungsi obyek dalam desain *form* cetak nota pembayaran sebagai berikut:

Tabel 3.43 Fungsi Obyek *Form* Cetak Nota Pembayaran

Nama Obyek	Tipe Obyek	Fungsi
Report	<i>CrystalReport</i>	Digunakan untuk menampilkan laporan
Tanggal Pembayaran	<i>ComboBox</i>	Berisi tanggal transaksi pembayaran
Proses	<i>Button</i>	Digunakan untuk menampilkan data jika <i>filter</i> telah dipilih
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> pembayaran biaya pemeriksaan

N. Desain *Form* Pemesanan Ke *Tehniker*

Form Pemesanan ke *Tehniker* digunakan untuk *insert*, *edit*, dan *delete* data pemesanan ke *tehniker*. Desain *form* pesanan ke *tehniker* dapat dilihat pada Gambar 3.37 di halaman 84.

Data Pesanan Pasien								
Daftar Pesanan Belum Order								
Nama Pasien	Gigi	Pelayanan/tindakan	Tgl Kembali	Biaya ke Pasien				
Nama Pasien : [Nama Pasien]		Nama Tehniker :		Simpan				
Tindakan : [Tindakan]		Total Biaya :						
Gigi : [Gigi]								
Tgl Kembali : [Tgl Kembali]								
Daftar Pesanan Sudah Order								
Nama Pasien	Gigi	Pelayanan/tindakan	Tgl Order	Tgl Ambil	Tgl Kembali	Nama Tehniker	Biaya	Batal
								Edit Tgl Ambil
								Pesanan Selesai
								Tutup

Gambar 3.37 Desain Form Pesanan ke Tehniker

Fungsi-fungsi obyek dalam desain form pesanan ke *tehniker* sebagai berikut:

Tabel 3.44 Fungsi Obyek Form Pesanan ke Tehniker

Nama Obyek	Tipe Obyek	Fungsi
Daftar Pesanan Belum Order	<i>DataGridView</i>	Berisi daftar pesanan pasien yang belum diorder ke <i>tehniker</i>
Daftar Pesanan Sudah Order	<i>DataGridView</i>	Berisi daftar pesanan ke <i>tehniker</i> yang belum selesai dikerjakan
Nama Pasien	<i>Label</i>	Diisi dengan nama pasien yang pesannya akan diproses
Tindakan	<i>Label</i>	Diisi dengan tindakan yang dipesan
Gigi	<i>Label</i>	Diisi dengan gigi yang dipesan
Tgl Kembali	<i>Label</i>	Diisi dengan tanggal pesanan dikembalikan ke pasien
Nama Tehniker	<i>ComboBox</i>	Berisi nama-nama <i>tehniker</i>
Total Biaya	<i>NumericUpDown</i>	Diisi dengan biaya pekerjaan ke <i>tehniker</i>
Simpan	<i>Button</i>	Digunakan untuk menyimpan data order ke <i>tehniker</i>
Edit Tgl Ambil	<i>Button</i>	Digunakan untuk mengubah tanggal pengambilan order dari <i>tehniker</i>
Pesanan Selesai	<i>Button</i>	Digunakan untuk menyimpan informasi pesanan ke <i>tehniker</i> sudah selesai
Tutup	<i>Button</i>	Digunakan untuk keluar dari form pesanan ke <i>tehniker</i>

O. Desain *Form* Laporan Kunjungan Pasien

Form ini digunakan untuk menampilkan laporan mengenai jumlah kunjungan pasien pada priode tertentu. Desain *form* laporan kunjungan pasien dapat dilihat pada Gambar 3.38.

Gambar 3.38 Desain *Form* Laporan Kunjungan Pasien

Fungsi-fungsi obyek dalam desain *form* laporan kunjungan pasien sebagai berikut:

Tabel 3.45 Fungsi Obyek *Form* Laporan Kunjungan Pasien

Nama Obyek	Tipe Obyek	Fungsi
Report	<i>CrystalReport</i>	Digunakan untuk menampilkan laporan
Tanggal, Bulan dan Tahun	<i>RadioButton</i>	Digunakan untuk meng <i>enable</i> atau <i>disable</i> Tanggal, Bulah dan Tahun Kunjungan
Tanggal Kunjungan	<i>DateTimePicker</i>	Digunakan untuk mengfilter laporan berdasarkan jangka waktu tertentu
Bulan Kunjungan	<i>ComboBox</i>	Berisikan 12 bulan. Digunakan untuk mengfilter laporan
Tahun Kunjungan	<i>ComboBox</i>	Berisikan tahun saat ini s/d 5 (lima) tahun sebelumnya. Digunakan untuk mengfilter laporan
Proses	<i>Button</i>	Digunakan untuk menampilkan data jika <i>filter</i> telah dipilih
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> laporn kunjungan pasien

P. Desain Form Laporan Grafik Kunjungan Pasien

Form ini digunakan untuk menampilkan laporan mengenai jumlah kunjungan pasien dalam bentuk grafik pada priode tertentu. Desain *form* laporan grafik kunjungan pasien dapat dilihat pada Gambar 3.39.

Gambar 3.39 Desain Form Laporan Grafik Kunjungan Pasien

Fungsi-fungsi obyek dalam desain *form* laporan grafik kunjungan pasien sebagai berikut:

Tabel 3.46 Fungsi Obyek Form Laporan Grafik Kunjungan Pasien

Nama Obyek	Tipe Obyek	Fungsi
Report	<i>CrystalReport</i>	Digunakan untuk menampilkan laporan
Pelayanan, Jenis Pasien, dan Pasien Baru/Lama	<i>RadioButton</i>	Digunakan untuk mengubah pengelompokan data
Tanggal Kunjungan	<i>DateTime Picker</i>	Digunakan untuk mengfilter laporan berdasarkan jangka waktu tertentu
Tahun	<i>CheckBox</i>	Digunakan untuk meng <i>enable</i> atau <i>disable</i> Tanggal Kunjungan dan Tahun
Tahun	<i>ComboBox</i>	Berisikan tahun saat ini s/d 5 (lima) tahun sebelumnya. Digunakan untuk mengfilter laporan
Jumlah Kunjungan(%)	<i>CheckBox</i>	Digunakan untuk mengubah jumlah kunjungan ke dalam persen (%)
Proses	<i>Button</i>	Digunakan untuk menampilkan data jika <i>filter</i> telah dipilih

Tabel 3.46 (Lanjutan)

Nama Obyek	Tipe Obyek	Fungsi
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> laporn grafik kunjungan pasien

Q. Desain *Form* Grafik Jam Kunjungan Terpadat

Form ini digunakan untuk menampilkan laporan mengenai jumlah kunjungan pasien pada jam tertentu dalam bentuk grafik pada priode tertentu. Desain *form* laporan grafik jam kunjungan terpadat dapat dilihat pada Gambar 3.40 di bawah.

Gambar 3.40 Desain *Form* Grafik Jam Kunjungan Terpadat

Fungsi-fungsi obyek dalam desain *form* grafik jam kunjungan terpadat sebagai berikut:

Tabel 3.47 Fungsi Obyek *Form* Grafik Jam Kunjungan Terpadat

Nama Obyek	Tipe Obyek	Fungsi
Report	<i>CrystalReport</i>	Digunakan untuk menampilkan laporan
Tanggal, Bulan dan Tahun	<i>RadioButton</i>	Digunakan untuk meng <i>enable</i> atau <i>disable</i> Tahun, Bulan dan Tanggal Pengeluaran
Tanggal Pengeluaran	<i>DateTimePicker</i>	Digunakan untuk mengfilter laporan berdasarkan jangka waktu tertentu

Tabel 3.47 (Lanjutan)

Nama Obyek	Tipe Obyek	Fungsi
Bulan Pengeluaran	<i>ComboBox</i>	Berisikan 12 bulan. Digunakan untuk mengfilter laporan
Tahun	<i>ComboBox</i>	Berisikan tahun saat ini s/d 5 (lima) tahun sebelumnya. Digunakan untuk mengfilter laporan
Jumlah Kunjungan(%)	<i>CheckBox</i>	Digunakan untuk mengubah jumlah kunjungan ke dalam persen (%)
Proses	<i>Button</i>	Digunakan untuk menampilkan data jika <i>filter</i> telah dipilih
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> grafik jam kunjungan terpadat

R. Desain Form Laporan Pengeluaran

Desain *form* ini digunakan sebagai acuan untuk membuat *form* cetak laporan pengeluaran, *form* cetak laporan pemasukan, dan *form* cetak laporan pendapatan. *Form* ini digunakan untuk menampilkan dan mencetak laporan pengeluaran, pemasukan, dan pendapatan yang terjadi di klinik pada priode tertentu. Desain *form* laporan pengeluaran dapat dilihat pada Gambar 3.41.

Gambar 3.41 Desain *Form* Laporan Pengeluaran

Fungsi-fungsi obyek dalam desain *form* pengeluaran sebagai berikut:

Tabel 3.48 Fungsi Obyek *Form* Laporan Pengeluaran

Nama Obyek	Tipe Obyek	Fungsi
Report	<i>CrystalReport</i>	Digunakan untuk menampilkan laporan
Tanggal, Bulan dan Tahun	<i>RadioButton</i>	Digunakan untuk meng <i>enable</i> atau <i>disable</i> Tahun, Bulan dan Tanggal Pengeluaran
Tanggal Pengeluaran	<i>DateTimePicker</i>	Digunakan untuk mengfilter laporan berdasarkan jangka waktu tertentu
Bulan Pengeluaran	<i>ComboBox</i>	Berisikan 12 bulan. Digunakan untuk mengfilter laporan
Tahun	<i>ComboBox</i>	Berisikan tahun saat ini s/d 5 (lima) tahun sebelumnya. Digunakan untuk mengfilter laporan
Sertakan Jenis Pengeluaran	<i>CheckBox</i>	Digunakan untuk mengubah pengelompokan data
Proses	<i>Button</i>	Digunakan untuk menampilkan data jika <i>filter</i> telah dipilih
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> pengeluaran

S. Desain *Form* Penggunaan Obat dan Peralatan

Form ini digunakan untuk menampilkan laporan mengenai jumlah penggunaan obat dan peralatan dalam bentuk grafik, dan data penggunaan obat pada priode tertentu. Desain *form* penggunaan obat dan peralatan dapat dilihat pada Gambar 3.42 di bawah.

Gambar 3.42 Desain *Form* Penggunaan Obat dan Peralatan

Fungsi-fungsi obyek dalam desain *form* penggunaan obat dan peralatan sebagai berikut:

Tabel 3.49 Fungsi Obyek *Form* Penggunaan Obat dan Peralatan

Nama Obyek	Tipe Obyek	Fungsi
Report	<i>CrystalReport</i>	Digunakan untuk menampilkan laporan
Data penggunaan obat	<i>RadioButton</i>	Digunakan untuk mengubah laporan dalam bentuk data
Grafik penggunaan obat dan grafik penggunaan peralatan	<i>RadioButton</i>	Digunakan untuk mengubah laporan dalam bentuk grafik
Tanggal Pengeluaran	<i>DateTimePicker</i>	Digunakan untuk mengfilter laporan berdasarkan jangka waktu tertentu
Tahun	<i>ComboBox</i>	Berisikan tahun saat ini s/d 5 (lima) tahun sebelumnya. Digunakan untuk mengfilter laporan
Jumlah Kunjungan(%)	<i>CheckBox</i>	Digunakan untuk mengubah jumlah penggunaan ke dalam persen (%)
Proses	<i>Button</i>	Digunakan untuk menampilkan data jika <i>filter</i> telah dipilih
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> penggunaan obat dan peralatan

T. Desain *Form* Trend Penyakit dan Tindakan

Form ini digunakan untuk menampilkan laporan mengenai jumlah penyakit gigi dan mulut yang sering dialami pasien dan tindakan yang sering digunakan untuk merawat masalah gigi dan mulut dalam bentuk grafik pada periode tertentu. Desain *form* trend penyakit dan tindakan dapat dilihat pada Gambar 3.43.

Gambar 3.43 Desain *Form* Trend Penyakit dan Tindakan

Fungsi-fungsi obyek dalam desain *form* trend penyakit dan tindakan sebagai berikut:

Tabel 3.50 Fungsi Obyek *Form* Trend Penyakit dan Tindakan

Nama Obyek	Tipe Obyek	Fungsi
Report	<i>CrystalReport</i>	Digunakan untuk menampilkan laporan
Grafik trend pelayanan dan grafik trend penyakit	<i>RadioButton</i>	Digunakan untuk mengubah grafik pelayanan atau grafik penyakit
Tanggal Pengeluaran	<i>DateTimePicker</i>	Digunakan untuk mengfilter laporan berdasarkan jangka waktu tertentu
Bulan Pengeluaran	<i>ComboBox</i>	Berisikan 12 bulan. Digunakan untuk mengfilter laporan
Tahun	<i>ComboBox</i>	Berisikan tahun saat ini s/d 5 (lima) tahun sebelumnya. Digunakan untuk mengfilter laporan
Jumlah Kunjungan(%)	<i>CheckBox</i>	Digunakan untuk mengubah jumlah penanganan ke dalam persen (%)
Proses	<i>Button</i>	Digunakan untuk menampilkan data jika <i>filter</i> telah dipilih
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> trend penyakit dan tindakan

U. Desain *Form* Cetak Daftar Utang Pasien

Desain *form* ini digunakan sebagai acuan untuk membuat *form* cetak daftar utang pasien. *Form* ini digunakan untuk menampilkan dan mencetak laporan utang pasien yang belum tertagih. Desain *form* cetak daftar utang pasien dapat dilihat pada Gambar 3.44.

Gambar 3.44 Desain *Form* Cetak Daftar Utang Pasien

Fungsi-fungsi obyek dalam desain *form* cetak daftar utang pasien sebagai berikut:

Tabel 3.51 Fungsi Obyek *Form* Cetak Daftar Utang Pasien

Nama Obyek	Tipe Obyek	Fungsi
Report	<i>CrystalReport</i>	Digunakan untuk menampilkan laporan
Semua dan Perusahaan	<i>RadioButton</i>	Digunakan untuk mengfilter laporan berdasarkan jenis pasien atau semua
Perusahaan	<i>ComboBox</i>	Digunakan untuk mengfilter laporan berdasarkan jenis pasien
Bulan	<i>CheckBox</i>	Digunakan untuk meng <i>enable</i> atau <i>disable combo box</i> bulan dan tahun
Bulan	<i>ComboBox</i>	Berisikan 12 bulan. Digunakan untuk mengfilter laporan
Tahun	<i>ComboBox</i>	Berisikan tahun saat ini s/d 5 (lima) tahun sebelumnya. Digunakan untuk mengfilter laporan
Proses	<i>Button</i>	Digunakan untuk menampilkan data jika <i>filter</i> telah dipilih
Tutup	<i>Button</i>	Digunakan untuk keluar dari <i>form</i> cetak daftar utang pasien

3.10 Desain Uji Coba

Desain uji coba bertujuan untuk memastikan bahwa aplikasi telah dibuat sesuai dengan kebutuhan atau tujuan yang diharapkan. Kekurangan atau kelemahan aplikasi pada tahap ini akan dievaluasi sebelum diimplementasikan secara nyata.

A. Desain Uji Coba Fitur Dasar Sistem

Desain uji coba fitur dasar sistem ini dilakukan dengan melakukan berbagai percobaan untuk membuktikan bahwa aplikasi yang telah dibuat telah sesuai dengan tujuan.

A.1. Desain Uji Coba Fitur Login

Proses *login* dilakukan dengan cara menginputkan Nama dan Kata Sandi. Data *login* yang digunakan terlihat pada Tabel 3.52 di bawah untuk masing-masing pengguna.

Tabel 3.52 Data Testing Pengguna

Nama Field	Data 1	Data 2	Data 3
Nama	Admin	Ananda	Ana
Kata Sandi	Admin	12345	Admin

Untuk *test case login* dapat dilihat pada Tabel 3.53.

Tabel 3.53 Test Case Login

Test case ID	Tujuan	Input	Output yang diharapkan
1	Deskripsi Nama, dan Kata Sandi yang valid	Memasukan data 1 dari table testing data pengguna	Form login tertutup dan menu - menu pada form utama keluar
2	Deskripsi Nama, dan Kata Sandi yang tidak ada dalam system	Memasukan data 2 dari table testing data pengguna	Muncul pesan “Nama tidak terdaftar”

Tabel 3.53 (Lanjutan)

Test case ID	Tujuan	Input	Output yang diharapkan
3	Deskripsi Nama yang terdaftar dan Kata Sandi yang tidak valid	Memasukan data 3 dari <i>table testing</i> data pengguna	Muncul pesan “Kata Sandi tidak sesuai”
4	Deskripsi <i>username</i> , <i>password</i> yang kosong	Tidak ada	Muncul <i>error provider</i> dengan <i>tool tip</i> “Nama harus diisi” dan “Sandi harus diisi”

A.2. Desain Uji Coba Manipulasi Data Kota

Proses manipulasi data kota adalah proses penyimpanan untuk data kota baru, dan perubahan data untuk data kota yang telah disimpan sebelumnya, dan membatalkan proses penyimpanan dan perubahan data. Proses ini bertujuan mengetahui dan menentukan keberhasilan dari obyek-obyek yang ada dalam desain *form* kota. Data kota yang digunakan terlihat pada Tabel 3.54 untuk masing-masing kota.

Tabel 3.54 Data Testing Kota

Nama Field	Data 1	Data 2	Data 3
Kode Kota	K000000001	K000000001	K000000002
Nama Kota	Surabaya	Surabaya	Surabaya
Kode Pos	090909	S	001122

Untuk *test case* kota dapat dilihat pada Tabel 3.55.

Tabel 3.55 Test Case Kota

Test case ID	Tujuan	Input	Output yang diharapkan
5	Menghindari <i>input</i> data kosong	Menekan tombol Simpan dengan mengkosongkan semua inputan	Muncul <i>error provider</i> disetiap inputan yang kosong
6	Menghindari data yang bertipe invalid	Memasukan data 2 dari tabel <i>testing</i> data kota	Muncul <i>error provider</i> ditempat yang bertipe invalid

Tabel 3.55 (Lanjutan)

Test case ID	Tujuan	Input	Output yang diharapkan
7	Menambah data baru ke tabel kota dengan data lengkap	Masukan data 1 dari tabel <i>testing</i> data kota	<i>Form</i> mengeluarkan pesan “Data kota sukses tersimpan!”, dan data bertambah pada <i>gridview</i> dan <i>database</i>
8	Menambah data baru ke tabel kota dengan data lengkap	Memasukan data 3 dari tabel <i>testing</i> data kota	<i>Form</i> mengeluarkan pertanyaan berupa <i>message box</i>
9	Menyimpan perubahan data	Mengubah Nama Kota data lain dengan nama kota data 1	<i>Form</i> mengeluarkan pertanyaan berupa <i>message box</i>
10	Menyimpan perubahan data	Mengubah kode pos pada data 1 dengan kode pos data 3	<i>Form</i> mengeluarkan pesan “Perubahan data kota sukses tersimpan!”, dan data yang diubah tersimpan dalam <i>database</i>

A.3. Desain Uji Coba Manipulasi Data Jenis

Proses manipulasi data jenis adalah proses penyimpanan untuk data jenis baru, perubahan data untuk data jenis yang telah disimpan sebelumnya, dan membatalkan proses penyimpanan dan perubahan data. Proses ini bertujuan mengetahui dan menentukan keberhasilan dari obyek-obyek yang ada dalam desain *form* jenis. Data-data yang menggunakan desain *form* jenis yaitu, data penyakit, data jenis pengeluaran, data jenis peralatan, dan data diagnosa. Data jenis yang digunakan terlihat pada Tabel 3.56.

Tabel 3.56 Data Testing Jenis

Nama Field	Data 1	Data 2	Data 3
Kode Jenis	J001	J002	J001
Nama Jenis	Abc	Abc	Xyz

Untuk *test case* jenis dapat dilihat pada Tabel 3.57.

Tabel 3.57 *Test Case* Jenis

Test case ID	Tujuan	Input	Output yang diharapkan
11	Menambah data baru ke tabel jenis dengan data lengkap	Masukan data 1 dari tabel <i>testing</i> data jenis	<i>Form</i> mengeluarkan pesan “Data jenis sukses tersimpan!”, dan data bertambah pada gridview dan database
12	Menghindari <i>input</i> data kosong	Menekan tombol Simpan dengan mengosongkan semua imputan	Muncul <i>error provider</i> disetiap inputan yang kosong
13	Menambah data baru ke tabel jenis dengan data lengkap	Memasukan data 2 dari tabel <i>testing</i> data jenis	<i>Form</i> mengeluarkan pertanyaan berupa <i>message box</i>
14	Menyimpan perubahan data	Mengubah Nama jenis pada data 1 dengan mengambil Nama jenis data 3	<i>Form</i> mengeluarkan pesan “Perubahan data jenis sukses tersimpan!”, dan data yang diubah tersimpan dalam database
15	Menyimpan perubahan data	Mengubah Nama Jenis data lain dengan nama jenis data 1	<i>Form</i> mengeluarkan pertanyaan berupa <i>message box</i>

A.4. Desain Uji Coba Manipulasi Data Relasi

Proses manipulasi data relasi adalah proses penyimpanan untuk data relasi baru, perubahan data untuk data relasi yang telah disimpan sebelumnya, dan membatalkan proses penyimpanan dan perubahan data. Proses ini bertujuan mengetahui dan menentukan keberhasilan dari obyek-obyek yang ada dalam desain *form* relasi. Data-data yang menggunakan desain *form* relasi yaitu, data perusahaan, *supplier*, dan *tehniker* gigi. Data relasi yang digunakan terlihat pada Tabel 3.58 di halaman 97.

Tabel 3.58 Data *Testing* Relasi

Nama Field	Data 1	Data 2	Data 3
ID Relasi	R0001	R0002	R0001
Nama Relasi	PT.Xyz	PT.Xyz	PT.Abc
Alamat Relasi	Jl.Xyt No.01	Jl.Abc No.02	Jl.Abc No.02
Kota Relasi	Xyz	Abc	Abc
No Telepon	031555999	0310000099	Abc
No Fax	031666999	-	-

Untuk *test case* relasi dapat dilihat pada Tabel 3.59.

Tabel 3.59 Test Case Relasi

Test case ID	Tujuan	Input	Output yang diharapkan
16	Menambah data baru ke tabel relasi dengan data lengkap	Masukan data 1 dari tabel <i>testing</i> data relasi	<i>Form</i> mengeluarkan pesan “Data relasi sukses tersimpan!”, dan data bertambah pada <i>gridview</i> dan <i>database</i>
17	Menambah data baru ke tabel relasi dengan data lengkap	Memasukan data 2 dari tabel <i>testing</i> data relasi	<i>Form</i> mengeluarkan pertanyaan berupa <i>message box</i>
18	Menyimpan perubahan data	Mengubah Nama Kota data lain dengan nama kota data 1	<i>Form</i> mengeluarkan pertanyaan berupa <i>message box</i>
19	Menyimpan perubahan data	Mengubah Alamat pada data 1 dengan mengambil alamat data 2	<i>Form</i> mengeluarkan pesan “Perubahan data relasi sukses tersimpan!”, dan data yang diubah tersimpan dalam <i>database</i>
20	Menghindari data yang bertipe invalid	Memasukan data 3 dari tabel <i>testing</i> data relasi	Muncul <i>error provider</i> dengan <i>tool tip</i> “Telepon harus berupa angka”
21	Menghindari <i>input</i> data kosong	Menekan tombol Simpan dengan mengkosongkan semua inputan	Muncul <i>error provider</i> disetiap inputan yang kosong

A.5. Desain Uji Coba Manipulasi Data Pasien

Proses manipulasi data pasien adalah proses penyimpanan untuk data pasien baru, perubahan data untuk data pasien yang telah disimpan sebelumnya, membatalkan proses penyimpanan dan perubahan data, dan menghapus data yang ada. Proses ini bertujuan mengetahui dan menentukan keberhasilan dari obyek-obyek yang ada dalam desain *form* pasien. Data pasien yang digunakan terlihat pada Tabel 3.60 untuk masing-masing pasien.

Tabel 3.60 Data *Testing* Pasien

Nama Field	Data 1	Data 2	Data 3
ID Pasien	A011111	M011111	J011111
Nama Pasien	Amelia Raharjo	Melisa	Joko
Alias	Amel	-	-
Sebutan	Nyonya	Nona	Tuan
Jenis Kelamin	Wanita	Wanita	Pria
Alamat	Jl.Darmo No.110	Jl.Kali No.01	Jl.xyz
Kota	Surabaya	Surabaya	Xyz
Tanggal Lahir	28/02/1981	28/02/1989	28/02/1978
Jenis Pasien	Umum	PT.xyz	Umum
Tlp Rumah	-	-	Xyz
Tlp HP	0852999	-	08993888999

Untuk *test case* pasien dapat dilihat pada Tabel 3.61.

Tabel 3.61 Test Case Pasien

Test case ID	Tujuan	Input	Output yang diharapkan
22	Menambah data baru ke tabel pasien dengan data lengkap	Masukan data 1 dari tabel <i>testing</i> data pasien	<i>Form</i> mengeluarkan pesan “Data pasien sukses tersimpan!”, dan data bertambah pada <i>gridview</i> dan <i>database</i>
23	Menghindari data yang bertipe invalid	Masukan data 3 dari tabel <i>testing</i> data pasien	Muncul <i>error provider</i> dengan <i>tool tip</i> “No telepon rumah dan HP harus diisi dengan angka”
24	Menghindari <i>input</i> data kosong	Menekan tombol Simpan dengan mengkosongkan semua imputan	Muncul <i>error provider</i> disetiap inputan yang kosong

Tabel 3.61 (Lanjutan)

Test case ID	Tujuan	Input	Output yang diharapkan
25	Menyimpan perubahan data	Mengubah Alamat pada data 1 dengan mengambil alamat data 2	<i>Form</i> mengeluarkan pesan “Perubahan data pasien sukses tersimpan!”, dan data yang diubah tersimpan dalam <i>database</i>
26	Menghapus data pasien yang tidak memiliki data kunjungan	Memilih pasien pada data 2	<i>Form</i> mengeluarkan pesan “Data pasien sukses terhapus!”
27	Menghapus data pasien yang memiliki data kunjungan	Memilih pasien pada data 1	<i>Form</i> mengeluarkan pesan data tidak dapat dihapus.

A.6. Desain Uji Coba Manipulasi Data Pengeluaran

Proses manipulasi data pengeluaran adalah proses penyimpanan untuk data pengeluaran baru, perubahan data untuk data pengeluaran yang telah disimpan sebelumnya, dan membatalkan proses penyimpanan data pengeluaran. Proses ini bertujuan mengetahui dan menentukan keberhasilan dari obyek-obyek yang ada dalam desain *form* pengeluaran.

Tabel 3.62 Data Testing Pengeluaran

Nama Field	Data 1	Data 2	Data 3
No Pengeluaran	-	-	-
Pengeluaran	XXX	Xyz	XXX
Biaya	100000	Xxx	100000
Tanggal	10/10/2010	10/10/2010	11/10/2010
Keterangan	-	-	-

Untuk *test case* pengeluaran dapat dilihat pada Tabel 3.63.

Tabel 3.63 Test Case Pengeluaran

Test case ID	Tujuan	Input	Output yang diharapkan
28	Menambah data baru ke tabel pengeluaran dengan data lengkap	Masukan data 1 dari tabel <i>testing</i> data pengeluaran	<i>Form</i> mengeluarkan pesan “Data pengeluaran sukses tersimpan!”

Tabel 3.63 (Lanjutan)

Test case ID	Tujuan	Input	Output yang diharapkan
29	Menghindari <i>input</i> data kosong	Menekan tombol Simpan dengan mengkosongkan semua imputan	Muncul <i>error provider</i> disetiap inputan yang kosong
30	Menghindari <i>inputan</i> data dengan jenis pengeluaran yang sama pada bulan yang sama	Masukan data 3 dari tabel <i>testing</i> data pengeluaran	<i>Form</i> mengeluarkan pesan "Transaksi pengeluaran ini sudah dilakukan pada tanggal 10/10/2010. Apa ingin menyimpan transaksi ini?". Jika menekan tombol "Yes" maka data baru akan disimpan.
31	Menyimpan perubahan data	Mengubah biaya pada data 1 dengan mengambil biaya data 3	<i>Form</i> mengeluarkan pesan "Anda yakin untuk mengubah data pengeluaran ini?" Jika menekan tombol "Yes" maka perubahan data akan disimpan.

A.7. Desain Uji Coba Manipulasi Data Pembelian

Proses manipulasi data pembelian adalah proses penyimpanan untuk data pembelian baru, dan membatalkan proses penyimpanan data pembelian. Proses ini bertujuan mengetahui dan menentukan keberhasilan dari obyek-obyek yang ada dalam desain *form* pembelian.

Tabel 3.64 Data *Testing* Pembelian

Nama Field	Data 1	Data 2	Data 3
No. Pembelian	NB111120110001	NB111120110002	NB111120110003
Tgl. Pembelian	11 Nov 2011	11 Nov 2011	11 Nov 2011
ID Supplier	Sp0001	Sp0001	Sp0001
Nama Supplier	Xyz	Xyz	Xyz
Kode Item	IT0001	IT0001	IT0001
Jenis	Obat	Obat	Obat
Nama	Alkohol	Alkohol	Alkohol
Jumlah	1	0	1
Harga	80000	0	90000
Tgl Kadaluarsa	11/11/2015	11/11/2015	11/11/2011

Untuk *test case* pembelian dapat dilihat pada Tabel 3.65.

Tabel 3.65 *Test Case* Pembelian

Test case ID	Tujuan	Input	Output yang diharapkan
32	Menambah data baru ke tabel pembelian dengan data lengkap	Masukkan data 1 dari tabel <i>testing</i> data pembelian	Data bertambah pada <i>gridview</i> detail pembelian
33	Menghindari <i>input</i> data kosong	Menekan tombol OK dengan mengosongkan semua imputan	Muncul <i>error provider</i> disetiap inputan yang koson
34	Menghindari <i>input</i> Tgl Kadaluarsa yang tidak sesuai	Masukkan data 3 dari tabel <i>testing</i> data pembelian	<i>Form</i> mengeluarkan pesan “Tanggal Kadaluarsa harus lebih dari tanggal hari ini”
35	Menghindari <i>input</i> tanpa memilih item	Menekan tombol Ok tanpa memilih Item	<i>Form</i> mengeluarkan pesan “Pilih Item yang ingin diinputkan”
36	Membatalkan data pembelian yang ada di tabel Detal Pembelian	Pilih data yang ingindibatalkan dan tekan tombol Batal	Data yang dipilih akan dihapus dari <i>gridview</i> detail pembelian
37	Menghindari proses penyimpanan ke <i>database</i> dengan data detail pembelian yang kosong	Menekan tombol Simpan dengan Detail Pembelian yang kosong	<i>Form</i> mengeluarkan pesan “Maaf! Data pembelian Kosong.”
38	Menghindari inputan data yang sama ke detail pembelian	Masukkan data 1 dari tabel <i>testing</i> data pembelian	<i>Form</i> mengeluarkan pesan “Item sudah terdaftar pada detail pembelian! Periksa lagi daftar detail pembelian.”
39	Menyimpan data pembelian ke <i>database</i> tanpa memilih <i>supplier</i>	Menekan tombol simpan dengan tabel detail pembelian terisi dan <i>supplier</i> belum diketahui	<i>Form</i> mengeluarkan pesan “Supplier belum dipilih!”
40	Menyimpan data pembelian ke <i>database</i> dengan data legkap	Menekan tombol simpan dengan tabel detail pembelian terisi	<i>Form</i> mengeluarkan pesan “Data Pembelian sukses tersimpan!”

Tabel 3.65 (Lanjutan)

Test case ID	Tujuan	Input	Output yang diharapkan
41	Menghindari menutup jendela pembelian tanpa melakukan proses pembayaran	Menekan tombol tutup setelah berhasil melakukan penyimpanan transaksi pembelian	<i>Form</i> mengeluarkan pesan "Anda belum melakukan pembayaran Tunai atau Angsuran"
42	Menyimpan data pembayaran tunai	Menekan tombol tunai setelah berhasil melakukan penyimpanan transaksi pembelian	<i>Form</i> mengeluarkan pesan "Transaksi pembayaran tunai telah tersimpan"

A.8. Desain Uji Coba Manipulasi Data Kunjungan Pasien

Proses manipulasi data kunjungan adalah proses penyimpanan untuk data kunjungan pasien, dan membatalkan proses penyimpanan data kunjungan pasien. Proses ini bertujuan mengetahui dan menentukan keberhasilan dari obyek-obyek yang ada dalam desain *form* kunjungan pasien.

Tabel 3.66 Data Testing Kunjungan Pasien

Nama Field	Data 1	Data 2
Input Berdasarkan	Nama	Nama
Inputan	Mindy	Xxx

Untuk *test case* kunjungan pasien dapat dilihat pada Tabel 3.67.

Tabel 3.67 Test Case Kunjungan Pasien

Test case ID	Tujuan	Input	Output yang diharapkan
43	Menambah data baru ke tabel kunjungan dengan data lengkap	Masukkan data 1 dari tabel <i>testing</i> data kunjungan	Data bertambah pada <i>gridview</i> data kunjungan pasien
44	Menghindari <i>input</i> data kosong	Menekan tombol OK dengan mengkosongkan imputan	<i>Form</i> mengeluarkan pesan "Inputan masih kosong"
45	Menghindari <i>input</i> data yang tidak ada dalam sistem	Masukan data 2 dari tabel <i>testing</i> data kunjungan	<i>Form</i> mengeluarkan pesan "Nama pasien yang diinputkan tidak ada dalam data pasien"

Tabel 3.67 (Lanjutan)

Test case ID	Tujuan	Input	Output yang diharapkan
46	Menghindari inputan data yang sama ke tabel kunjungan	Masukkan data 1 dari tabel <i>testing</i> data kunjungan	<i>Form</i> mengeluarkan pesan "Nama pasien yang diinputkan sudah ada dalam daftar penerimaan"
47	Menghindari proses dengan tabel kunjungan yang kosong	Menekan tombol proses pemeriksaan, cetak kartu berobat, atau ubah data pasien	<i>Form</i> mengeluarkan pesan "Pilih data pasien yang ingin diproses"

A.9. Desain Uji Coba Manipulasi Data Resep Obat

Proses manipulasi data resep obat adalah proses penyimpanan data resep obat, dan membatalkan proses penyimpanan. Proses ini bertujuan untuk mengetahui dan menentukan keberhasilan obyek-obyek yang ada dalam desain *form* data pemeriksaan.

Tabel 3.68 Data *Testing* Resep Obat

Nama Field	Data 1	Data 2
Nama Obat	Abc	Abc
Banyak	-	10 tablet
Aturan minum	-	1x1

Untuk *test case* resep obat dapat dilihat pada Tabel 3.69.

Tabel 3.69 *Test Case* Resep Obat

Test case ID	Tujuan	Input	Output yang diharapkan
48	Menghindari input data tidak lengkap	Masukan data 1 dari tabel <i>testing</i> resep obat	<i>Form</i> mengeluarkan pesan berupa <i>message box</i>
49	Menambah data baru ke tabel data resep obat dengan data lengkap	Masukan data 2 dari tabel <i>testing</i> resep obat	<i>Form</i> mengeluarkan pesan berupa <i>message box</i>

A.10. Desain Uji Coba Manipulasi Data Pemeriksaan

Proses manipulasi data pemeriksaan adalah proses penyimpanan untuk data pemeriksaan, dan membatalkan proses penyimpanan data pemeriksaan. Proses ini bertujuan mengetahui dan menentukan keberhasilan dari obyek-obyek yang ada dalam desain *form* pemeriksaan.

Tabel 3.70 Data Testing Pemeriksaan

Nama Field	Data 1	Data 2
Anamnesa	-	Abc
Gigi	-	11
Pemeriksaan Fisik	-	Abc
Diagnosa	-	Torus mandibularis
Pengobatan/tindakan	-	Kuretase
Biaya	-	100000

Untuk *test case* pemeriksaan dapat dilihat pada Tabel 3.71.

Tabel 3.71 Test Case Pemeriksaan

Test case ID	Tujuan	Input	Output yang diharapkan
50	Menghindari <i>input</i> data kosong	Masukan data 1 dari tabel <i>testing</i> pemeriksaan dan tekan tombol tambah	Muncul <i>error provider</i> disetiap inputan yang kosong
51	Menambah data baru ke tabel pemeriksaan dengan data lengkap	Masukkan data 2 dari tabel <i>testing</i> pemeriksaan dan tekan tombol tambah	Data bertambah pada <i>gridview</i> pemeriksaan
52	Menyimpan data pemeriksaan ke <i>database</i> dengan data lengkap	Menekan tombol tutup dengan tabel pemeriksaan terisi	<i>Form</i> mengeluarkan pesan pertanyaan berupa <i>message box</i>

A.11. Desain Uji Coba Manipulasi Data Pembayaran Biaya Pemeriksaan

Proses manipulasi data pembayaran biaya pemeriksaan adalah proses penyimpanan untuk data pembayaran biaya pemeriksaan. Proses ini bertujuan mengetahui dan menentukan keberhasilan dari obyek-obyek yang ada dalam

desain *form* pembayaran biaya pemeriksaan. Data pembayaran biaya pemeriksaan yang digunakan terlihat pada Tabel 3.72.

Tabel 3.72 Data *Testing* Pembayaran Biaya Pemeriksaan

Nama Field	Data 1	Data 2
Bayar	0	50000

Untuk *test case* pembayaran biaya pemeriksaan dapat dilihat pada Tabel 3.73.

Tabel 3.73 Test Case Pembayaran Biaya Pemeriksaan

Test case ID	Tujuan	Input	Output yang diharapkan
53	Menghindari <i>input</i> data bernilai 0 (nol)	Masukan data 1 dari tabel <i>testing</i> pembayaran biaya pemeriksaan dan tekan tombol bayar	<i>Form</i> mengeluarkan pesan berupa <i>message box</i>
54	Menyimpan data pembayaran ke <i>database</i>	Masukan data 2 dari tabel <i>testing</i> pembayaran biaya pemeriksaan dan tekan tombol bayar	Data bertambah pada <i>gridview</i> riwayat pembayaran

A.12. Desain Uji Coba Pesanan ke Tehniker

Proses manipulasi data pesanan ke *tehniker* adalah proses penyimpanan untuk data pesanan ke *tehniker*, mengubah data pesanan, dan membatalkan proses penyimpanan data pesanan. Proses ini bertujuan mengetahui dan menentukan keberhasilan dari obyek-obyek yang ada dalam desain *form* pesanan ke *tehniker*.

Tabel 3.74 Data *Testing* Pesanan ke *Tehniker*

Nama Field	Data 1	Data 2	Data 3	Data 4
Nama Pasien	[Nama Pasien]	Mindy	Mindy	Mindy
Nama Tehniker	Adi	None	Adi	Adi
Total Biaya	90000	98000	0	98000

Untuk *test case* pesanan ke *tehniker* dapat dilihat pada Tabel 3.75.

Tabel 3.75 *Test Case* Pesanan ke *Tehniker*

Test case ID	Tujuan	Input	Output yang diharapkan
55	Menghindari <i>input</i> data dengan data pesanan pasien kosong	Masukkan data 1 dari tabel <i>testing</i> data pesanan ke <i>tehniker</i>	<i>Form</i> mengeluarkan pesan "Pilih data pesanan yang belum order!"
56	Menghindari <i>input</i> data dengan nama <i>tehniker</i> belum dipilih	Masukkan data 2 dari tabel <i>testing</i> data pesanan ke <i>tehniker</i>	<i>Form</i> mengeluarkan pesan "Harus memilih <i>tehniker</i> !"
57	Menghindari <i>input</i> data dengan nilai biaya 0 (nol)	Masukkan data 3 dari tabel <i>testing</i> data pesanan ke <i>tehniker</i>	<i>Form</i> mengeluarkan pesan "Masukan total biaya pekerjaan!"
58	Simpan data pesanan ke <i>tehniker</i> dengan data lengkap	Masukan data 4 dari tabel <i>testing</i> data pesanan ke <i>tehniker</i>	<i>Form</i> mengeluarkan pesan berupa <i>message box</i> dan data pada <i>gridview</i> daftar pesanan sudah order bertambah
59	Menekan tombol batal, tombol <i>edit</i> tanggal ambil, dan pesanan selesai	Daftar pesanan sudah order kosong	<i>Form</i> mengeluarkan pesan "Daftar pesanan sudah order kosong!"
60	Menekan tombol batal	Daftar pesanan sudah order tidak kosong	<i>Form</i> mengeluarkan pesan pertanyaan berupa <i>message box</i>
61	Menekan tombol <i>edit</i> tanggal ambil	Daftar pesanan sudah order tidak kosong	Membuka <i>form</i> ubah tanggal ambil
62	Menekan tombol pesanan selesai	Daftar pesanan sudah order tidak kosong	<i>Form</i> mengeluarkan pesan pertanyaan berupa <i>message box</i>

A.13. Desain Uji Fitur Laporan

Proses laporan untuk menghasilkan laporan yang diambil dari *database* dan ditampilkan dalam *form* melalui *crystal reports*. Laporan yang dihasilkan yaitu laporan kunjungan, laporan grafik kunjungan, grafik jam kunjungan terpadat, laporan pengeluaran, laporan jumlah penggunaan obat dan peralatan, dan

laporan jumlah penyakit dan tindakan. Data *testing* yang digunakan untuk menguji ketepatan pengambilan data oleh sistem untuk ditampilkan di *crystal report* terlihat pada Tabel 3.76.

Tabel 3.76 Data Testing Laporan

Nama Field	Data 1	Data 2
Data Tanggal	01/06/2011	01/05/2011
Data Bulan	06/2011	05/2011
Data Tahun	2011	2010

Untuk *test case* laporan dapat dilihat pada Tabel 3.77.

Tabel 3.77 Test Case Laporan

Test case ID	Tujuan	Input	Output yang diharapkan
63	Menampilkan laporan grafik jumlah kunjungan pasien	Masukan data 2 dari tabel <i>testing</i> data laporan	<i>Form</i> mengeluarkan pesan berupa <i>message box</i>
64	Menampilkan laporan grafik jumlah kunjungan pasien	Masukan data 1 dari tabel <i>testing</i> data laporan	<i>Form</i> menampilkan laporan grafik kunjungan pasien tahun 2011