

SKRIPSI

PENGEMBANGAN SISTEM INFORMASI REKAM MEDIS PADA RUMAH SAKIT UMUM DAERAH DOMPU



Disusun Oleh :
MUHAMAD KHAIRUN
04.12.748

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2009

0000

THE UNITED STATES DEPARTMENT OF JUSTICE
FEDERAL BUREAU OF INVESTIGATION
WASHINGTON, D. C. 20535
MEMORANDUM FOR THE DIRECTOR

04/18/48

MEMORANDUM FOR THE DIRECTOR
SUBJECT: [Illegible]

MEMORANDUM FOR THE DIRECTOR
SUBJECT: [Illegible]

000000

LEMBAR PERSETUJUAN

PENGEMBANGAN SISTEM INFORMASI REKAM MEDIS
PADA RUMAH SAKIT UMUM DAERAH DOMPU

SKRIPSI

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Komputer dan Informatika Strata Satu (S-1)*

Disusun Oleh :

MUHAMAD KHAIRUN


NIM : 04.12.748

Diperiksa dan Disetujui

Dosen Pembimbing I


Dosen Pembimbing II


Ir. F. Yudi Limpraptono, MT
NIP.Y 103 9500 274


Joseph Dedy Irawan, ST, MT.
NIP. 132 315 178

Mengetahui,

Ketua Jurusan Teknik Elektro S-1


Ir. F. Yudi Limpraptono, MT
NIP.Y 103 9500 274

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2009

ABSTRAKSI

PENGEMBANGAN SISTEM INFORMASI REKAM MEDIS PADA RUMAH SAKIT UMUM DAERAH DOMPU

Muhamad Khairun(04.12.748)

Jurusan Teknik Elektro S-1, Konsentrasi Teknik Komputer dan Informatika,

Fakultas Teknologi Industri, Institut Teknologi Nasional Malang

Jl. Raya Karanglo Km 2 Malang

Email : eyoenzaghi_09@yahoo.com

Rumah Sakit sebagai lembaga kesehatan setiap harinya selalu melakukan aktivitas kesehatan baik itu pencatatan pendaftaran pasien, hingga pelaporan berkas-berkas maupun riwayat penyakit pasien. Pencatatan data-data sosial maupun data medis tersebut dinamakan Rekam Medis. Namun dengan semakin berkembangnya ilmu pengetahuan dan teknologi mengakibatkan pencatatan data-data secara manual sangatlah tidak dapat diandalkan karena sering terjadi kesalahan dan kesulitan dalam proses pencarian arsip arsip maupun dalam pembuatan laporan. Hal ini yang sedang dirasakan oleh Rumah Sakit Umum Daerah Dompus. Pencatatan dan pengolahan data-data rekam medis pasien masih manual, sehingga proses pengolahan data sangat lambat dan cenderung banyak terjadi kesalahan. Oleh karena itu, perlu dibangun suatu Sistem Informasi yang dapat memudahkan dan meningkatkan efektifitas kinerja dari pengolahan data Rekam Medis pada Rumah Sakit Umum Daerah Dompus.

Dalam aplikasi ini, user dapat melakukan kegiatan pengolahan data Rekam Medis berupa pendaftaran pasien baru, Register Pasien pada bangsal Instalasi Gawat Darurat, Rawat Jalan, sampai Register Pasien Rawat Inap. Pengolahan data Pemeriksaan pasien, Tindakan medis yang dilakukan terhadap pasien, data rujukan pasien dan pelaporan dari hasil pengolahan data tersebut.

Kata Kunci : Rumah Sakit Umum Daerah Dompus, Sistem Informasi Rekam Medis

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa yang telah memberikan rahmat dan hidayah-Nya, sehingga penyusun dapat menyelesaikan skripsi yang berjudul "PENGEMBANGAN SISTEM INFORMASI REKAM MEDIS PADA RUMAH SAKIT UMUM DAERAH DOMPU" ini dengan lancar. Skripsi ini merupakan persyaratan kelulusan Studi di Jurusan Teknik Elektro S-1 Konsentrasi Teknik Komputer dan Informatika ITN Malang dan untuk mencapai gelar Sarjana Teknik.

Keberhasilan penyelesaian laporan skripsi ini tidak lepas dari dukungan dan bantuan berbagai pihak. Untuk itu penyusun menyampaikan terima kasih kepada :

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE selaku rektor ITN Malang
2. Bapak Ir. H. Sidik Noertjahjono, MT selaku Dekan Fakultas Teknologi Industri.
3. Ir. F Yudi Limpraptono, MT, selaku Ketua Jurusan Teknik Elektro S-1 dan selaku Dosen Pembimbing I.
4. Joseph Dedy Irawan, ST.,MT, selaku Dosen Pembimbing II.
5. Dr. Ahmad Faisal, SpA, selaku Direktur Rumah Sakit Umum Daerah Dompus.
6. Kedua orang tua yang telah memberikan dukungan untuk selalu berdoa dan berusaha.
7. Megasari Widiani, Ronny susetyo, Skom, Agus Santoso, Skom, serta teman-teman seperjuangan terima kasih atas bantuan serta dukungannya.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang membangun sangat diharapkan. Harapan penyusun semoga laporan skripsi ini memberikan manfaat bagi perkembangan ilmu pengetahuan dan pembaca.

Malang, Agustus 2009

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	
LEMBAR PERSETUJUAN	i
ABSTRAKSI.....	ii
KATA PENGANTAR	iii
DAFTAR ISI.....	iv
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xv
BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	3
1.4 Tujuan	3
1.5 Metode Penelitian.....	4
1.5.1 Metode Pengumpulan Data.....	4
1.5.2 Metode Pengembangan Sistem.....	5
1.6 Sistematika Penulisan	7
BAB II LANDASAN TEORI	
2.1 Sistem Informasi	8
2.2 Rekam Medis	12
2.2.1 Pengertian Rekam Medis	12
2.2.2 Tujuan Rekam Medis.....	12

2.2.3	Kegunaan Rekam Medis	13
2.3	Sistem informasi Rekam Medis	13
2.4	Rumah Sakit Umum Daerah Dompu.....	14
2.5	Data Flow Diagram Fisik.....	15
2.5.1	Data Flow Diagram	15
2.5.2	Entity Relational Model(<i>E-R Model</i>)	16
2.5.3	Pengertian Information Oriented Flow Chart	19
2.6	Sistem Database	20
2.6.1	Pengertian Database	20
2.6.2	Keuntungan Sistem Database.....	22
2.7	Borland Delphi 7.0	24
2.7.1	Main Windows.....	25
2.7.2	Code Editor.....	26
2.7.3	Form Designer	27
2.7.4	Object Inspector.....	27
2.8	MySQL 5.0	28

BAB III PERANCANGAN SISTEM

3.1	Sistem Saat Ini.....	30
3.1.1	Pengguna Sistem.....	31
3.2.	Spesifikasi Sistem.....	32
3.3	Perencanaan Sistem.....	33
3.3.1	Diagram Alir Data(DAD).....	33
3.3.1.1	DAD Level 0.....	34
3.3.1.2	DAD Level 1	34

3.3.1.3	DAD Level 2.....	35
3.3.1.3.1	DAD Level 2 Proses 1.....	36
3.3.1.3.2	DAD Level 2 Proses 2.....	36
3.3.1.3.3	DAD Level 2 Proses 3.....	37
3.3.1.3.4	DAD Level 2 Proses 4.....	38
3.3.1.3.5	DAD Level 2 Proses 5.....	40
3.3.1.3.6	DAD Level 2 Proses 6.....	41
3.3.2	Desain Database Rekam Medis	42
3.3.2.1	Relasi Antar Tabel.....	42
3.3.2.2	Struktur Tabel yang digunakan	45
3.3.3	Desain Antarmuka Aplikasi	66
3.3.3.1	Desain Halaman Login	66
3.3.3.2	Desain Halaman Menu Utama	67
3.3.3.3	Desain Halaman Isi Data	68
3.3.3.4	Desain Form Register Pasien	68
3.3.3.5	Desain Form Pemeriksaan Pasien	69
3.3.3.6	Desain Form Manajemen User	70
3.3.3.7	Desain Form Pengisian Data.....	71
3.3.3.8	Desain Form Operasi.....	72
3.3.3.9	Desain Form Laporan	73

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1.	Implementasi Sistem	75
4.1.1	Menu Sistem	76
4.1.2	Menu Pendataan Siswa	77

4.1.2.1	Manajemen User	77
4.1.2.2	Isi Data	78
4.1.2.1	Manajemen Data Pasien	79
4.1.3	Menu Instalasi Gawat Darurat(I.G.D).....	79
4.1.3.1	Pendaftaran Pasien Baru.....	79
4.1.3.2	Daftar I.G.D.....	81
4.1.3.3	Pemeriksaan Pasien I.G.D	82
4.1.3.4	Tindakan Pasien I.G.D	85
4.1.3.5	Rujukan Pasien I.G.D.....	87
4.1.4	Menu Rawat Jalan(RAJAL)	88
4.1.4.1	Pendaftaran Pasien Baru.....	88
4.1.4.2	pendaftaran Pasien Rawat Jalan.....	89
4.1.4.3	Pemeriksaan Pasien Rawat Jalan	90
4.1.4.4	Tindakan Pasien.....	91
4.1.4.5	Rujukan Pasien	92
4.1.5	Menu Rawat Inap(RANAP)	93
4.1.5.1	Pendaftaran Pasien Baru.....	93
4.1.5.2	pendaftaran Pasien Rawat Inap.....	93
4.1.5.3	Pemeriksaan Pasien Rawat Inap	96
4.1.5.4	Tindakan Pasien	97
4.1.5.5	Pindah Kamar Pasien	97
4.1.5.6	Pencarian Kamar Pasien.....	98
4.1.5.7	Rujukan Pasien	99
4.1.5.8	Keadaan Keluar Pasien.....	99

4.1.6	Lain-lain	100
4.1.6.1	Laboratorium	100
4.1.6.2	Radiologi Pasien	102
4.1.6.3	Operasi Pasien.....	103
4.1.6.4	Pencarian Pasien	104
4.1.7	Laporan.....	105
4.1.7.1	Laporan Register Pasien I.G.D	106
4.1.7.2	Laporan Penyakit I.G.D.....	106
4.1.7.3	Laporan Dokter Pemeriksaan	107
4.1.7.4	Laporan Tindakan I.G.D	108
4.1.7.5	Laporan Register Pasien RAJAL	108
4.1.7.6	Laporan Penyakit RAJAL	109
4.1.7.7	Laporan Dokter RAJAL	110
4.1.7.8	Laporan Tindakan RAJAL	111
4.1.7.9	Laporan Register Pasien RANAP	111
4.1.7.10	Laporan Penyakit RANAP	112
4.1.7.11	Laporan Laboratorium.....	113
4.1.7.12	Laporan Operasi.....	114
4.1.7.13	Laporan Radiologi Pasien.....	115
4.1.7.10	Statistik Penyakit Pasien.....	115
4.2.	Pengujian Sistem	116
4.2.1	Pengujian Login.....	116
4.2.2	Pengujian Penginputan Data Pasien Baru	117
4.2.3	Pengujian Registrasi Pada Masing-masing Bangsal	119
4.2.4	Pengujian Pemeriksaan Pasien	120

4.2.5	Pengujian Tindakan Pasien	121
4.2.6	Pengujian Laporan	122

BAB V KESIMPULAN

5.1.	Kesimpulan	124
5.2.	Saran	125

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar :	Halaman :
2.1 Betuk Sistem.....	8
2.2 Siklus Informasi.....	11
2.3 Relasi One to One.....	18
2.4 Relasi One to Many	18
2.5 Relasi Many to Many.....	19
2.6 Jenjang Data	21
2.7 Main Windows	25
2.8 Code Editor.....	26
2.9 Form Designer	27
2.10 Object Inspector.....	28
3.1 Aplikasi yang Digunakan Saat Ini	30
3.2 DAD Level 0	34
3.3 DAD Level 1	35
3.4 DAD Level 2 Proses Login	36
3.5 DAD Level 2 Proses I.G.D.....	37
3.6 DAD Level 2 Proses Rawat Jalan.....	38
3.7 DAD Level 2 Proses Rawat Inap.....	39
3.8 DAD Level 2 Proses Pelaporan	40
3.9 DAD Level 2 Proses Master Data	41
3.10 PDM Data Rekam Medis	43
3.11 CDM Data Rekam Medis.....	44
3.12 Desain Halaman Login.....	67

3.13	Desain Menu Utama	67
3.14	Desain Isi Data Pasien.....	68
3.15	Desain Form Register Pasien.....	69
3.16	Desain Form Pemeriksaan Pasien.....	70
3.17	Desain Form Manajemen User.....	70
3.18	Desain Form Isi Data	71
3.19	Desain Operasi Pasien.....	72
3.20	Desain Laporan Pasien I.G.D	73
3.21	Desain Laporan Penyakit Rawat Jalan.....	73
3.22	Desain Laporan Diagnosa Rawat Inap.....	74
4.1	Menu Utama	76
4.2	Form Login.....	76
4.3	Form Ganti Password.....	77
4.4	Form Manajemen User.....	78
4.5	Isi Data	78
4.6	Hapus Data Pasien	79
4.7	Pendaftaran Pasien Baru	80
4.8	Pencetakan Kartu Berobat.....	80
4.9	Pendaftaran Pasien I.G.D	81
4.10	Pencarian NORM Pasien.....	82
4.11	Pencarian Cara Bayar.....	82
4.12	Pemeriksaan Pasien I.G.D	83
4.13	Pencarian Noregister I.G.D	84
4.14	Pencarian Dokter	84
4.15	Pencarian Diagnosa Penyakit	85

4.16	Tindakan Medis Pasien I.G.D	86
4.17	Pencarian Tindakan Pasien I.G.D.....	87
4.18	Rujukan Pasien I.G.D.....	88
4.19	Pendaftaran Pasien Baru	89
4.20	Registrasi Pasien Rawat Jalan	90
4.21	Pemeriksaan Pasien Rawat Jalan.....	91
4.22	Tindakan Medis Pasien Rawat Jalan.....	92
4.23	Rujukan Pasien Rawat Jalan.....	93
4.24	Pendaftaran Pasien Baru	94
4.25	Pendaftaran Pasien Rawat Inap	95
4.26	Pencarian Kamar Kosong.....	95
4.27	Pemeriksaan Pasien Rawat Inap.....	96
4.28	Tindakan Pasien Rawat Inap	97
4.29	Pindah Kamar Pasien	98
4.30	Pencarian Kamar Pasien.....	98
4.31	Rujukan Ranap	99
4.32	Keadaan Keluar Ranap.....	100
4.33	Pelayanan Laboratorium Pasien	101
4.34	Pencarian Pemeriksaan Pasien.....	101
4.35	Radiologi Pasien	102
4.36	Pencarian Radiologi Pasien	103
4.37	Operasi Pasien	104
4.38	Pencarian Operasi Pasien	104
4.39	Pencarian Pasien	105
4.40	Laporan Pasien	105

4.41	Laporan Register I.G.D	106
4.42	Laporan Penyakit Pasien I.G.D	107
4.43	Laporan Dokter Pasien I.G.D	107
4.44	Laporan Tindakan I.G.D	108
4.45	Laporan Register Pasien RAJAL.....	109
4.46	Laporan Penyakit Pasien RAJAL	110
4.47	Laporan Dokter Pasien RAJAL.....	110
4.48	Laporan Tindakan Pasien RAJAL.....	111
4.49	Laporan Register Pasien RANAP.....	112
4.50	Laporan Laboratorium	114
4.51	Laporan Operasi.....	114
4.52	Laporan Radiologi Pasien	115
4.53	Diagram Statistik Pasien	116
4.54	Pengujian Login.....	117
4.55	Pesan Kesalahan	117
4.56	Pesan Selamat Datang.....	117
4.57	Pengujian Pengisian Data Pasien Baru	118
4.58	Pesan Data Tersimpan.....	118
4.59	Data Pasien Pada Database.....	119
4.60	Pengujian Registrasi Pasien.....	120
4.61	Pengujian Pemeriksaan Pasien	121
4.62	Pengujian Tindakan Pasien	121
4.63	Pengujian Laporan	122
4.64	Pengujian Diagram	123

DAFTAR TABEL

Tabel :	Halaman :
2.1 Simbol Data Flow Diagram	16
2.2 Symbol Information Oriented Flowchart	19
3.1 Struktur Tabel Pasien	45
3.2 Struktur Tabel IGD	46
3.3 Struktur Tabel RAJAL	47
3.4 Struktur Tabel Poli	48
3.5 Struktur Tabel RANAP	48
3.6 Struktur Tabel Cara_bayar	49
3.7 Struktur Tabel Kamar	50
3.8 Struktur Tabel Pemeriksaan IGD	50
3.9 Struktur Tabel Pemeriksaan RAJAL	51
3.10 Struktur Tabel Pemeriksaan RANAP	52
3.11 Struktur Tabel Diagnosa_Komplik	52
3.12 Struktur Tabel Morfologi dan Neoplasma	53
3.13 Struktur Tabel Diagnosa	54
3.14 Struktur Tabel Penyebab_Cidera	54
3.15 Struktur Tabel Penyebab_cidera_dtl	55
3.16 Struktur Tabel Imunisasi	55
3.17 Struktur Tabel Imunisasi_dtl	56
3.18 Struktur Tindakan IGD	56
3.19 Struktur Tabel Tindakan RAJAL	57
3.20 Struktur Tabel Tindakan RANAP	58

3.21	Struktur Tabel Tindakan_Dtl.....	58
3.22	Struktur Tabel Rujukan IGD	59
3.23	Struktur Tabel Rujukan RAJAL	59
3.24	Struktur Tabel Rujukan RANAP	60
3.25	Struktur Tabel Dokter	60
3.26	Struktur Tabel Operasi	61
3.27	Struktur Tabel Operasi_dtl	61
3.28	Struktur Tabel Laboratorium.....	62
3.29	Struktur Tabel Laboratorium_dtl.....	63
3.30	Struktur Tabel Radiologi.....	63
3.31	Struktur Tabel Radiologi_dtl.....	64
3.32	Struktur Tabel Pindah Kamar	64
3.33	Struktur Tabel Keadaan Keluar	65
3.34	Struktur Tabel User.....	66

BAB I

PENDAHULUAN

1.1 Latar Belakang

Rekam medis adalah berkas yang berisikan informasi tentang identitas pasien, anamnese, penentuan fisik laboratorium, diagnosa segala pelayanan dan tindakan medik yang diberikan kepada pasien dan pengobatan baik yang dirawat inap, rawat jalan maupun yang mendapatkan pelayanan gawat darurat.^[1]

Rekam medis setiap pasien harus sudah siap apabila mereka datang berobat kembali karena jika rekam medis tidak ada, petugas kesehatan akan sulit melakukan tindakan yang dilakukan terhadap seorang pasien menurut sejarah penyakit yang terdapat dalam rekam medis tersebut.

Kelengkapan pengisian berkas rekam medis oleh tenaga kesehatan akan memudahkan tenaga kesehatan lain dalam memberikan tindakan atau terapi kepada pasien. Selain itu juga sebagai sumber data pada bagian rekam medis dalam pengolahan data yang kemudian akan menjadi informasi yang berguna bagi pihak manajemen dalam menentukan langkah-langkah strategis untuk pengembangan pelayanan kesehatan.

Penyimpanan berkas rekam medis yang sudah berbasis komputer akan memudahkan pengolahan dan pengaksesan data dari petugas rekam medis dalam pembuatan laporan maupun statistik perkembangan pelayanan kesehatan maupun statistik penyakit.

Rumah Sakit Umum Daerah Dompu juga telah menerapkan sistem informasi rekam medis yang berbasis komputer. Data-data yang tersimpan dalam *database* adalah data sosial maupun data medis pasien rawat jalan maupun rawat inap, imunisasi, tindakan dan diagnosa akhir.

Akan tetapi yang masih menjadi permasalahan adalah sistem informasi ini masih bersifat berdiri sendiri, jadi dalam satu komputer terdapat satu database dan database tersebut belum terhubung satu sama lain sehingga petugas rekam medis masih kesulitan dalam pembuatan laporan dan pengolahan data pasien pada masing-masing bagian. Tampilan sistem informasi rekam medis ini juga masih kurang bagus sehingga para petugas agak kesulitan mengoperasikannya.

Oleh karena itu, dibutuhkan sebuah sistem yang terhubung satu dengan yang lain dan dapat memudahkan pengolahan data rekam medis secara cepat, tepat dan akurat serta memudahkan pengguna. Sehingga diharapkan dengan dibuatnya aplikasi sistem informasi rekam medis ini membuat pengolahan data pasien di Rumah Sakit Umum Daerah Dompu dapat dilakukan dengan mudah dan lebih efisien.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas maka timbul suatu permasalahan bagaimana membangun dan mengembangkan suatu aplikasi Sistem Informasi Rekam Medis untuk meningkatkan kinerja dan efisiensi dalam pengolahan data-data pasien pada Rumah Sakit Umum Daerah Dompu.

1.3 Batasan Masalah

Pembahasan skripsi ini dibatasi oleh hal-hal sebagai berikut:

1. Aplikasi sistem Rumah Sakit ini tidak terhubung dengan internet.
2. Pengembangan aplikasi ini menggunakan Borland Delphi 7, database MySQL 5.0.
3. Aplikasi ini dilengkapi dengan fasilitas hak akses bagi pengguna.
4. Implementasi sistem Informasi Rekam Medis ini adalah di Lingkungan Rumah Sakit Umum Daerah Dompus berbasis *client-server*.
5. Tidak membahas secara rinci mengenai sistem jaringan yang digunakan.

1.4 Tujuan

Tujuan dari skripsi ini adalah menghasilkan Aplikasi Sistem Informasi Rekam medis untuk mempermudah pengolahan dan pengaksesan data pasien oleh tenaga kesehatan baik itu berupa perawat, dokter, maupun petugas administrasi di Rumah Sakit Umum Daerah Dompus.

1.5 Metodologi

1.5.1 Metode Pengumpulan data

Data merupakan sumber atau bahan mentah yang sangat berharga bagi proses menghasilkan informasi. Oleh sebab itu dalam pengambilan data perlu dilakukan penanganan secara cermat dan hati-hati, sehingga data yang diperoleh dapat bermanfaat dan berkualitas.

Dalam pengumpulan data penyusun menggunakan metode sebagai berikut :

1. Studi Lapangan

Dengan metode ini data-data diperoleh langsung dari sumber yang bersangkutan, dimana peneliti berhadapan langsung dengan obyek yang diteliti, yang dilakukan dengan cara :

a. *Survey*

Teknik pengumpulan data dengan cara terjun secara langsung dan mencatat secara sistematis terhadap obyek masalah.

b. *Wawancara / Interview*

Teknik pengumpulan data dengan jalan mengadakan komunikasi langsung dengan pimpinan atau pegawai perusahaan tentang sistem yang diterapkan.

2. Studi Pustaka / *Literature*

Pengumpulan data ini dilakukan dengan cara mencari bahan-bahan kepustakaan sebagai landasan teori yang ada

hubungannya dengan permasalahan yang dijadikan obyek penelitian.

1.5.2 Metode Pengembangan Sistem

Metode yang digunakan dalam pengembangan Sistem Informasi Akademik adalah :

1. **Rekayasa Sistem.**

Tahapan awal yang dilakukan adalah merumuskan sistem yang akan dibuat. Rekayasa sistem ini bertujuan agar pengembangan benar-benar mengerti sistem yang akan dibuat dan langkah-langkah serta kebijaksanaan apa saja yang akan berkaitan dengan pengembangan sistem.

2. **Analisis Sistem.**

Tahapan kedua adalah analisis yang berkaitan dengan proses dan data yang diperlukan oleh sistem serta keterkaitannya. Pemodelan yang digunakan pada analisis ini adalah dengan menggunakan *Data Flow Oriented* dengan *tool Data Flow Diagram (DFD)*.

3. **Desain Sistem.**

Tahapan ketiga adalah desain sistem, dilakukan setelah mendapat gambaran yang jelas dari sistem yang akan dibuat. Tahapan desain sistem ini dilakukan untuk memberikan gambaran umum yang jelas kepada pengguna dan rancangan bangun yang lengkap tentang sistem yang

akan dikembangkan kepada pihak-pihak yang terlibat dalam pengembangan sistem.

4. Implementasi Sistem.

Setelah mendapatkan gambaran yang jelas tentang rancang bangun sistem, kemudian dilakukan implementasi rancangan sistem ke dalam kode-kode dalam bahasa pemrograman. Pada tahap ini dilakukan pembuatan komponen-komponen sistem yang meliputi implementasi modul-modul program, antarmuka dan basis data.

5. Pengujian Sistem.

Tujuan dilakukan tahapan pengujian ini adalah untuk mendapatkan perangkat lunak yang benar-benar valid dan sesuai dengan kebutuhan yang sudah dideskripsikan.

6. Pemeliharaan Sistem.

Setelah dilakukan pengujian dan sistem diyakini benar-benar memenuhi persyaratan, selanjutnya sistem tersebut didistribusikan kepada pengguna. Pada tahap ini juga dilakukan evaluasi terhadap sistem yang baru untuk melihat apakah sistem yang telah memenuhi tujuan yang ingin dicapai. Dari hasil evaluasi ini memungkinkan dilakukan perubahan-perubahan yang perlu terhadap sistem yang ada.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penyusunan skripsi ini adalah sebagai berikut:

BAB I : PENDAHULUAN

Bab ini berisi tentang latar belakang, tujuan, permasalahan, batasan masalah, dan sistematika pembahasan dari skripsi ini.

BAB II : LANDASAN TEORI

Bab ini berupa penjelasan tentang teori – teori yang mendukung dalam perancangan system yang meliputi, teori Borland Delphi 7 dan MySQL 5.0.

BAB III : PERANCANGAN SISTEM

Bab ini berisi tentang membangun Sistem Informasi Rekam Medis.

BAB IV : IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi pembahasan hasil pengujian serta pembahasan dari hasil analisa mengenai cara kerja dari sistem.

BAB V : PENUTUP

Bab ini berisi kesimpulan dan saran dari hasil pembahasan pada skripsi ini.

BAB II

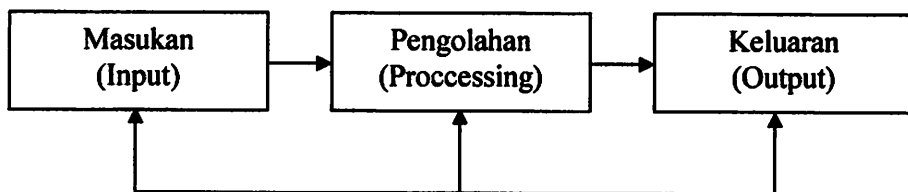
LANDASAN TEORI

2.1 Sistem Informasi

Agar mampu melakukan analisis yang sesuai dengan konsep-konsep yang berkaitan dengan sistem informasi, perlu dijelaskan kembali beberapa definisi tentang sistem, informasi, dan sistem informasi. Selain itu akan dijelaskan pula proses-proses atau aktivitas di dalam sistem informasi.

1. Konsep Dasar Sistem

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai tujuan tertentu. Model umum sebuah sistem terdiri dari masukan, pengolahan dan keluaran. Berikut gambaran umum sebuah sistem:



Gambar 2.1 Bentuk Sistem^[3]

Selain itu suatu sistem mempunyai karakteristik atau sifat-sifat tertentu (Hartono, 1999), yaitu mempunyai komponen-komponen (*components*), batas sistem (*boundary*), lingkungan luar sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran

(*output*), pengolah (*process*), dan tujuan (*goal*). Sistem mempunyai ciri:

- a. Adanya tujuan yang jelas.
- b. Mempunyai struktur tertentu.
- c. Terdiri dari satu kesatuan usaha dari bagian-bagian yang saling tergantung dan berinteraksi satu sama lain.^[3]

2. Kosep Dasar Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Sumber informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata.^[3]

Informasi mempunyai ciri-ciri:

- a. Data yang telah diolah.
- b. Menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima.
- c. Menggambarkan suatu kejadian dan kesatuan nyata.
- d. Digunakan untuk mengambil keputusan.

3. Siklus Informasi

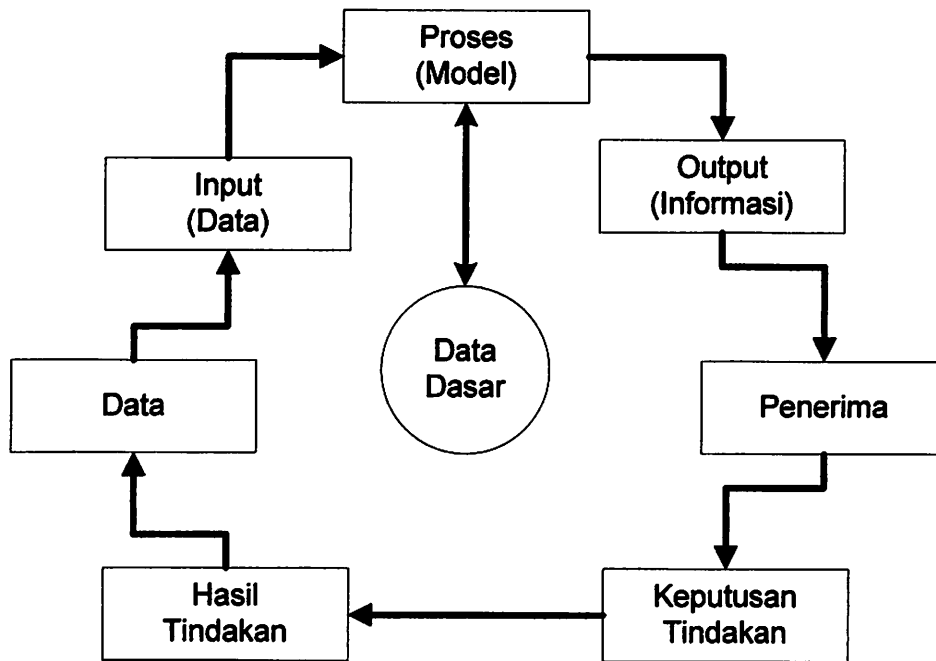
Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model untuk menghasilkan informasi.^[2]

Data yang diolah untuk menghasilkan informasi menggunakan suatu model proses tertentu. Misalnya data temperatur ruangan yang didapat adalah dalam satuan derajat fahrenheit dan data ini masih dalam bentuk yang kurang berarti bagi penerimanya yang terbiasa dengan satuan derajat celcius. Supaya dapat lebih berarti dan berguna dalam bentuk informasi, maka perlu diolah melalui suatu model tertentu.

Dalam hal ini digunakan model matematik yang berupa rumus konversi satuan derajat *fahrenheit* menjadi satuan derajat *celcius*. Data diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan lain yang akan membuat sejumlah data kembali.

Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus informasi.^[2]

Siklus ini oleh John Burch disebut sebagai siklus informasi (*information cycle*). Berikut gambar siklus informasi :



Gambar 2.2 Siklus Informasi^[3]

4. Konsep Dasar Sistem Informasi

Sedangkan sistem informasi sendiri adalah suatu cara yang sudah tertentu untuk menyediakan informasi yang dibutuhkan oleh organisasi untuk beroperasi dengan cara yang sukses dan untuk organisasi bisnis dengan cara yang menguntungkan. Peran sistem informasi adalah menghasilkan informasi dari data yang diproses oleh sistem informasi.

Komponen yang terkait dengan sistem informasi adalah :

- a. Pemakai
- b. Tujuan
- c. Masukan – proses – keluaran
- d. Data

- e. Teknologi
- f. Model
- g. Pengendali ^[3]

2.2 Rekam Medis

2.2.1 Pengertian rekam medis

Rekam medis merupakan hasil aktivitas pencatatan pada suatu rumah sakit atau suatu institusi pelayanan kesehatan yang berupa data. Data tersebut meliputi data sosial maupun data medis pasien rawat jalan dan rawat inap dan diproses oleh seorang tenaga rekam medis ataupun paramedis sehingga menjadi informasi yang berguna bagi rumah sakit.

Adapun pengertian rekam medis adalah himpunan fakta-fakta yang berhubungan dengan riwayat hidup dan kesehatan tentang seorang pasien tersebut yang ditulis oleh professional dibidang kesehatan.^[4]

2.2.2 Tujuan rekam medis

Menurut petunjuk teknis penyelenggaraan rekam medis di rumah sakit Depkes RI Dirjen Yanmed tahun 1991, tujuan terlaksananya rekam medis adalah untuk menunjang tertib administrasi dalam rangka upaya peningkatan pelayanan kesehatan rumah sakit.

2.2.3 Kegunaan rekam medis

Menurut Huffman (1994) menyatakan bahwa kegunaan rekam medis adalah sebagai berikut :

- a. Manajemen pelayanan pasien
- b. Quality Review (tinjauan kualitas)
- c. Financial reimbursement (pengurusan klaim asuransi)
- d. Legal affairs (perkara hukum)
- e. Education (pendidikan)
- f. Research (penelitian)
- g. Public health (kesehatan umum)
- h. Planning and marketing (perencanaan dan pemasaran)

2.3. Sistem Informasi Rekam Medis

Sistem Informasi Rekam Medis merupakan sistem yang mencatat, mengolah, maupun melaporkan hasil aktivitas pencatatan pada suatu rumah sakit atau suatu institusi pelayanan kesehatan yang berupa data baik itu data Sosial maupun data medis pasien rawat jalan dan rawat inap dan diproses oleh seorang tenaga rekam medis ataupun paramedis sehingga menjadi informasi yang berguna bagi rumah sakit. ^[2]

Dan dari data-data inilah nantinya akan diolah sedemikian rupa sehingga didapatkan segala sesuatu yang diperlukan dalam kegiatan yang terdapat di Rumah Sakit baik itu untuk pasien ataupun dokter yang menangani pasien tersebut.

Jadi, data pasien tersimpan dalam database yang terdapat pada *server*. Dan dapat diakses oleh *client* di beberapa tempat seperti petugas yang berada di bagian IGD, bagian rawat inap, maupun pada bagian rawat jalan.

2.4 Rumah Sakit Umum Daerah Dompus

Rumah Sakit Umum Daerah Dompus didirikan pada tahun 1954 dengan nama Balai Pengobatan. Dan pada tahun 1970 hingga sekarang diambil alih pengelolaannya oleh kabupaten Dompus. Dan baru pada tahun 1998 Rumah Sakit Umum Daerah Dompus ditetapkan kelas C dengan jumlah tempat tidur 67 buah dan mempunyai lahan 2 hektar dengan luas bangunan 15,553.80 m².

Tahun 2001 sampai dengan sekarang RSUD Kabupaten Dompus mulai melakukan pengembangan peningkatan baik fisik maupun sarana peralatan kesehatan untuk meningkatkan kualitas pelayanan. Kegiatan tersebut didukung oleh dana dari APBN maupun APBD jumlah tempat tidur sekarang 100 buah.

Saat ini Rumah Sakit Umum Daerah Dompus dipimpin oleh Dr. H. Ahmad Faisal, SpA. Sebagai direktur dan Ketua Komite medik dipimpin oleh Dr. Ketut Suardana, SpOG.

Beberapa prestasi yang diraih Rumah Sakit antara lain :

1. Rumah Sakit Gerakan Sayang Ibu TK. Nasional tahun 2002
2. Rumah Sakit Pelayanan Prima TK Nasional tahun 2003
3. Rumah Sakit Sayang Ibu TK. Propinsi tahun 2005

4. Rumah Sakit Sayang Ibu TK. Nasional tahun 2007
5. Rumah Sakit Lingkungan Bersih Tk. Propinsi 2007
6. Rumah Sakit TerAkreditasi Tahun 2008
7. Rumah Sakit Sayang Ibu TK. Propinsi 2008

Fasilitas yang terdapat pada Rumah Sakit ini antara lain pada bangsal rawat jalan terdapat poliklinik spesialis kandungan, poliklinik anak, penyakit dalam, poli mata serta poli bedah. Pada bangsal rawat inap memiliki Zaal Perawatn Bedah, Zaal Perawatn Penyakit Dalam, KBR, Zaal Perawatan Anak dan Kelas I/VIP. Selain itu terdapat fasilitas penunjang antara lain Radiologi, Laboratorium, Farmasi/Apotik, Gizi dan IPRS. Rumah Sakit ini juga dilengkapi dengan instalasi khusus seperti UGD, ICU, dan Fisioterapi.

Selain pengembangan fasilitas, rumah sakit umum daerah dompu juga mengembangkan sumber daya alam melalui program-program :

1. P2KP (Pusat Pelatihan Klinik Primer)
 - Pendidikan Khusus D.III
 - Pelatihan Ponek, APN
2. Lahan Praktek Mahasiswa AKPER dan Poltekkes





2.5 Data Flow Diagram Fisik

2.5.1 Data Flow Diagram (DFD)

Menurut Kristanto (2003:5), DFD adalah model logika data atau proses yang dibuat untuk menggambarkan dari mana 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. DFD menggambarkan penyimpanan data dan proses yang mentransformasikan data. DFD menunjukkan hubungan antara data pada sistem dan proses pada sistem.

Tabel 2.1. Simbol Data Flow Diagram^[3]

Symbol	Keterangan
	Menunjukkan proses transformasi data atau dapat mewakili suatu pekerjaan atau proses.
	Panah menunjukkan aliran suatu berkas elemen suatu data.
	Menunjukkan tempat penyimpanan data atau file.
	Menunjukkan entity yang ada dalam sistem.

2.5.2 Entity Relational Model (E-R Model)

Menurut William Donald (2000:414), E-R Model : *“Entity Relationship is a system analysis model that representents the requirements of a system”*. (E-R Model merupakan sebuah model analisis sistem yang menunjukkan syarat hubungan data atau kebutuhan dari suatu sistem).

Jadi, E-R Model adalah suatu metode atau cara untuk dapat menunjukkan suatu hubungan antara data dalam suatu sistem.

1. *Entity*

Adalah orang, tempat, kejadian atau konsep yang informasinya direkam.

2. *Atribut*

Adalah merupakan data elemen/data item, data field yang menggambarkan suatu entity. *Atribut* dibagi menjadi dua yaitu:

- a. Simple *atribut*, misalnya kode tamu.
- b. Komposit *atribut*, misalnya nama tamu.

3. *Relationship*

Menggambarkan hubungan dua atau lebih entity. Bila data dinormalisasikan dan informasi di pindah dari satu tabel ke tabel lain harus ada cara menghubungkan kedua tabel tersebut. Hubungan tersebut terbentuk dengan menggunakan kunci data yang bersifat unik. Dalam hubungan antar tabel dikenal ada dua kunci data penghubung, yaitu :

a. Kunci Utama (*Primary key*)

Suatu *atribut/field* atau satu set *atribut* yang mengidentifikasi secara unik suatu kejadian yang spesifik pada *entity*.

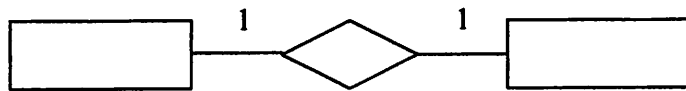
b. Kunci Tamu (*Foreign Key*)

Satu atribut atau satu set *atribut* yang melengkapi suatu hubungan yang menunjukkan ke *entity* induknya. Kunci tamu berada pada *entity* anak.

Berdasarkan hubungannya, hubungan antara *entity* dibedakan menjadi :

a. *Relasi one to one*

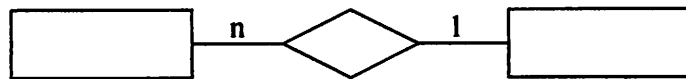
Menggambarkan hubungan satu ke satu, yaitu satu *record* pada *entity* pertama berhubungan dengan satu *record* pada *entity* kedua atau sebaliknya.



Gambar 2.3. Relasi one to one^[5]

b. *Relasi one to many*

Menggambarkan hubungan satu ke banyak, yaitu satu *record* pada *entity* pertama berhubungan dengan banyak *record* pada *entity* kedua atau sebaliknya.



Gambar 2.4. Relasi one to many^[5]

c. *Relasi many to many*

Menggambarkan hubungan banyak ke banyak, yaitu lebih dari satu *record* pada *entity* pertama berhubungan dengan lebih dari satu *record* pada *entity* kedua atau sebaliknya. Dibutuhkan *associative entity* atau *entity* perantara yang berisi *atribut/field* kunci dari masing-masing *entity*.



Gambar 2.5. Relasi many to many^[5]





2.5.3 Pengertian Information Oriented Flow Chart

Flowchart adalah penyajian yang sistematis tentang proses dan logika dari kegiatan penanganan informasi. *Flowchart* menggambarkan setiap langkah yang diperlukan dalam setiap aktifitas atau kegiatan yang dilakukan.

Information Oriented Flow Chart (IOFC) menggambarkan *input* dan *output flowchart* dari data yang akan melewati suatu sistem pada suatu proses *input output* data dengan menggunakan *flowchart*.

Tabel 2.2. Symbol Information Oriented Flowchart^[3]

No	Symbol	Keterangan
1.		Punched Card
2.		Magnetic Tape
3.		Disk
4.		Document
5.		Manual Input/keyboard
6.		Display

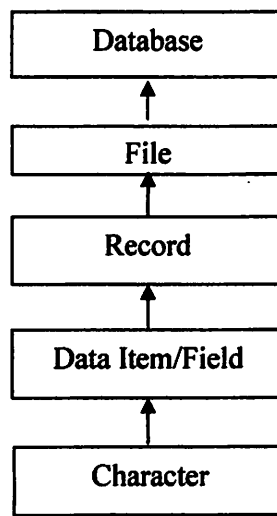
7.		Magnetic Disk
8.		Proses Manual
9.		File/arsip
10.		Murge/Penggabungan data

2.6 Sistem Database

2.6.1. Pengertian Database

Menurut Jogiyanto H.M (2000:265), database adalah “kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer yang digunakan perangkat lunak untuk memanipulasinya”.

Salah satu cara efektif (apabila kita bekerja dengan komputer) dalam mengolah suatu data dan informasi adalah dengan menciptakan suatu sistem Database. Sampai dengan membentuk suatu database, data mempunyai jenjang mulai dari karakter, data item (*field*) file dan kemudian database. Untuk lebih jelasnya jenjang data dapat dilihat gambar dibawah ini.



Gambar 2.6. Jenjang Data^[6]

Keterangan :

- *Character*

Karakter merupakan bagian data yang terkecil, dapat berupa karakter numerik, huruf ataupun karakter-karakter khusus membentuk suatu item data (*field*).

- *Field*

Suatu *field* menggambarkan suatu *atribut* dari record-record yang menunjukkan suatu item dari data misalnya nip, nama, dan lain sebagainya.

- *Record*

Kumpulan dari data *item/field* yang saling berkaitan membentuk suatu data, jadi satu *record* mewakili satu data.

- *File*

Kumpulan dari *record-record* yang sejenis yang mempunyai panjang elemen yang sama, *atribut* yang sama, namun berbeda-beda data volumenya.

- *Database*

Kumpulan file-file yang saling berkaitan satu dengan lainnya yang terpadu (*integrated*), terkontrol dan terpusat.

Sistem database meliputi :

- a. *Data*, yaitu data yang tersimpan didalamnya
- b. *Hardware*, perangkat keras yang bisa meliputi monitor, keyboard, CPU, dan printer sebagai perangkat tambahan.
- c. *Software*, perangkat lunak (biasanya disebut DBMS) yang berfungsi mengatur proses penyimpanan dan pencarian.
- d. *Brainware*, orang (*user*) yang berhubungan langsung dengan komputer, misalnya operator.

2.6.2 Keuntungan Sistem Database

Beberapa keuntungan jika menggunakan sistem database antara lain :

- a. Duplikasi data (*Redudancy*) dapat dikurangi

Pada sistem *non database conventional* masing-masing aplikasi yang berbeda mempertahankan filenya masing-masing. Sehingga seringkali timbul *redudancy* dalam sistem database. *Redudancy* ini dapat dikurangi dengan cara menyatukan file-file yang ada.

- b. *Inconsistency* dapat dihindari

Dengan mengurangi *redudancy* maka *inconsistency* dapat dihindari atau dikurangi.

c. *Shared data* (data dapat dipakai bersama)

Sharing adalah salah satu keuntungan yang paling penting dari sistem database, aplikasi-aplikasi yang ada dapat menunjuk data yang sama, aplikasi-aplikasi yang baru dapat menuju data yang sudah ada.

d. Standarisasi dapat dijalankan

Pengendalian terpusat (*sentralisasi kontrol*) membuat sistem database memungkinkan untuk menyelenggarakan suatu *standarisasi*. Standart ini cukup penting dalam jaringan komputer terpusat dimana terjadi perpindahan data antar sistem.

e. Batasan keamanan dapat diterapkan

Sistem database harus dirancang untuk dapat melindungi data karena data menjadi lebih riskan bila dikumpulkan dilokasi terpusat, daripada diletakkan tersebar secara fisik pada file-file yang berbeda dibanyak lokasi. Dalam hal ini batasan keamanan dapat diterapkan.

f. Integritas dapat dipertahankan

Integritas/keterpaduan data dapat dikendalikan dengan lebih baik dari suatu lingkungan database karena dapat dipakai secara bersama-sama.

2.7 Borland Delphi 7.0

Delphi adalah versi visual dari pascal. Berbagai kemudahan ditawarkan oleh delphi, mulai dari perancangan aplikasi berbasis *form*, kemudahan pemberian komponen visual, rekayasa *property* dan *event* yang tergabung melalui *object inspector*, sampai *code insight*.

Selain itu, dukungan penuh diberikan kepada struktur bahasa pemrograman yang digunakan, dengan memberikan migrasi dari Bahasa Pascal Object (*Object Pascal Language*). Dengan demikian, cara pemrograman sudah lain bila dibandingkan dengan pascal yang lama.

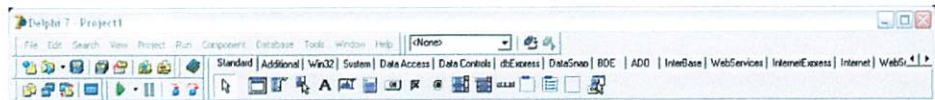
Dengan cara perancangan aplikasi secara visual, maka aplikasi-aplikasi Delphi mudah sekali dikembangkan, karena segala yang akan ingin ditampilkan dapat terlihat secara visual pada saat merancang aplikasi tersebut.

Salah satu lagi kelebihan Delphi adalah kemampuan databasenya yang mendukung hampir semua jenis database, mulai dari yang bersifat local hingga client server. Ditambah lagi dengan kelengkapan koneksi *Active Data Object* (ADO) yang merupakan standar dari microsoft, menambah integrasi aplikasi yang dikembangkan oleh Delphi dengan sistem operasi Microsoft Windows.

Selain itu disediakan pula versi Delphi khusus untuk sistem operasi linux, yaitu Borland Kylix.

2.7.1 Main Window

Window utama (*Main window*) adalah sebuah window yang terdapat pada bagian atas setiap kali Delphi dijalankan. Main window berisi menu-menu utama dari IDE Delphi, selain juga berisi komponen-komponen yang digunakan dalam membuat sebuah *project*.



Gambar 2.7. Main Window

Pada main window masih terdapat beberapa elemen yang digunakan untuk melakukan pemrograman pada Delphi. Beberapa elemen tersebut antara lain *main menu*, *speed buttons*, dan *component pallete*.

- *Main Menu*

Main menu merupakan menu utama yang digunakan untuk memilih perintah-perintah dalam IDE Delphi. Main menu berada pada posisi teratas dalam *main window*.

- *Speed Buttons*

Speed buttons merupakan tombol-tombol yang gunanya sama dengan main menu. Hanya saja bila dengan main menu diperlukan masuk berkali-kali dalam submenu, tetapi untuk beberapa perintah tertentu yang sering dipakai akan dapat

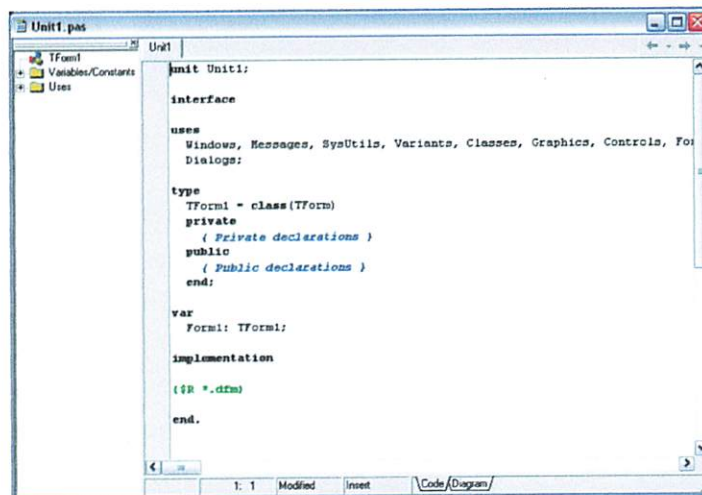
dieksekusi dengan cepat menggunakan speed buttons tersebut, yaitu dengan klik pada salah satunya.

- *Component Palette*

Component palette adalah sebuah palet yang berisi daftar komponen yang bisa digunakan dalam form aplikasi Delphi. Ada beberapa tab pada *component palette*, dan pemrogram harus memilihnya terlebih dahulu sebelum menggunakan komponen yang ada di dalam setiap tab tersebut.

2.7.2 Code Editor

Code editor adalah sebuah window yang digunakan untuk menyunting kode program, dimana pemrogram melakukan *coding*. *Code editor* juga terdapat IDE Turbo Pascal.

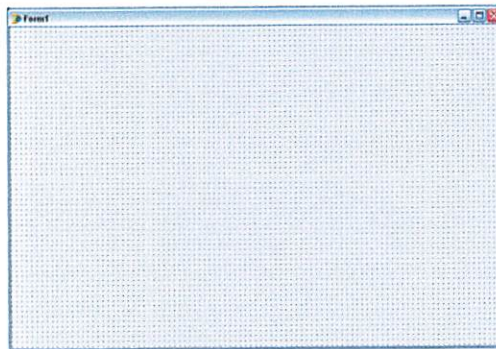


Gambar 2.8. Code Editor

2.7.3 Form Designer

Inti perancangan aplikasi secara visual terdapat pada *form designer*. Dalam *form designer*, pemrogram dapat merancang bagaimana sebuah form nantinya akan tampil dan berinteraksi dengan pemakai.

Sebuah form dapat berisi beberapa komponen yang terdapat pada main window. Beberapa komponen visual dapat langsung ditentukan posisi dan ukurannya hanya dengan melakukan *drag-drop mouse* pada komponen yang bersangkutan.



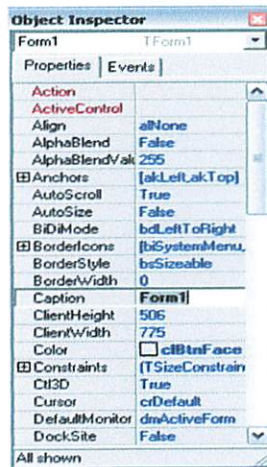
Gambar 2.9. Form Designer

2.7.4 Object Inspector

Dalam Delphi, komponen-komponen yang ada pada form (baik komponen visual maupun non-visual) memiliki beberapa property (*property*) dan *event*.

Properti menggambarkan sifat dari komponen tersebut. Sebagai contoh, properti *width* menunjukkan lebar dari komponen tersebut. Sedangkan *event* merupakan kejadian-kejadian yang terjadi dan

nantinya ditanggapi oleh pemrogram. Kode-kode untuk menanggapi kejadian tersebut dinamakan *event-handler*.



Gambar 2.10. Object Inspector

Object inspector merupakan sebuah window untuk merekayasa properti dan *event* dari beberapa komponen tersebut. Terdapat dua tab pada *object inspector*, yaitu tab *properties* dan tab *events*.

2.8 MySQL 5.0

MySQL merupakan salah satu DBMS (*Database Management System*) yang *free* dan *open source*, dikembangkan, didistribusikan, dan disupport oleh MySQL AB (Widenius, 2004).

Sampai saat ini, MySQL sudah mencapai versi 5.0 yang sudah mendukung *stored procedures* dan *functions*. *Stored procedures* adalah sejumlah perintah SQL yang disimpan dan dijalankan di MySQL server (Kofler, 2005). Kelebihan lain MySQL adalah dalam hal (Pachev, 2003):

1. *Speed* (kecepatan)
2. *Reliability* (dapat dipercaya)

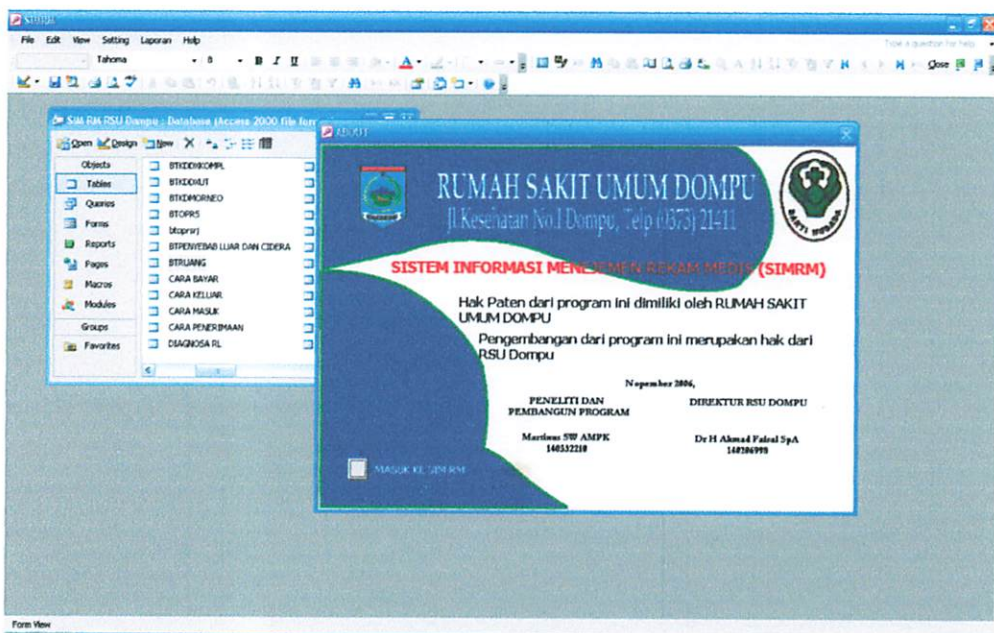
3. *Low system resource requirement* (dapat berjalan dalam sistem dengan spesifikasi rendah)
4. *Scalability* (menyesuaikan dengan sistem yang ada)
5. *Platform diversity* (dukungan terhadap berbagai macam sistem operasi)
6. *Support for a large number of host language* (dukungan terhadap berbagai macam bahasa pemrograman)
7. *ODBC support* (dukungan terhadap Open Database Connectivity)
8. *Free or low-cost licensing* (gratis atau dengan lisensi murah)
9. *Inexpensive commercial support* (dukungan untuk layanan komersial yang murah)
10. *Strong user community backing* (didukung oleh komunitas yang kuat)
11. *Availability of the source code* (source code diberikan secara gratis)

BAB III

PERANCANGAN SISTEM

3.1 Sistem Saat Ini

Sistem Informasi Rekam Medis yang ada saat ini sudah berbasis komputer, tetapi masih belum menerapkan sistem yang terstruktur. Aplikasi yang digunakan adalah aplikasi database *Microsoft Access*.



Gambar 3.1 Aplikasi yang digunakan saat ini

Sistem juga masih berdiri sendiri dan tidak terhubung dalam jaringan komputer sehingga data pasien yang ada pada setiap bagian perawatan tidak terhubung sehingga petugas rekam medis kesulitan dalam pembuatan laporan dan statistik perkembangan pelayanan kesehatan maupun statistik penyakit.

Kekurangan yang dimiliki sistem saat ini antara lain :

1. Pendistribusian data pasien pada sistem masih belum dapat dilakukan dengan cepat karena sistem yang digunakan belum memanfaatkan aplikasi *client-server*.
2. Pembuatan laporan dan statistik masih dilakukan pada masing-masing unit di setiap bagian sehingga petugas masih kesulitan dalam pembuatan laporan gabungan.
3. Data pasien yang terdapat pada sistem harus diinputkan pada setiap unit yang mengakibatkan pasien tetap menginputkan data di setiap unit pada bagian pelayanan di Rumah Sakit.
4. Banyak terjadi kesalahan (*error*) pada data pasien karena sulitnya pengoperasian sistem.

3.1.1 Pengguna Sistem

Pengguna sistem informasi rekam medis ini antara lain :

- a. Administrator, dalam hal ini adalah administrator RSUD Dompu, yaitu pengguna yang memiliki hak akses penuh terhadap aplikasi Sistem Informasi rekam medis di RSUD Dompu.
- b. Pimpinan, yaitu pejabat-pejabat yang memiliki hak akses untuk menggunakan aplikasi.
- c. Perawat, yaitu perawat yang hanya dapat mengakses data-data medis pasien.
- d. Lain-lain, orang-orang yang diijinkan untuk mengakses aplikasi ini.

3.2 Spesifikasi Sistem

Sesuai dengan tujuan dari pengembangan sistem informasi rekam medis, maka spesifikasi dari sistem informasi rekam medis adalah sebagai berikut:

1. Sistem informasi dengan model *client - server*, dimana aplikasi *client* dapat mengakses basis data.
2. Sistem basisdata server menggunakan MySql 5.0 dengan satu buah server.
3. Sebuah modul pelaporan dan pendukung keputusan yang dapat memberikan data-data dan rekapan yang akurat yang sangat berguna untuk pengambilan keputusan bagi *top management* di RSUD Dompnu.
4. Sistem informasi ini, terutama bagian aplikasi client diharapkan dapat dengan mudah dipakai oleh petugas kesehatan, baik itu dokter maupun operator.

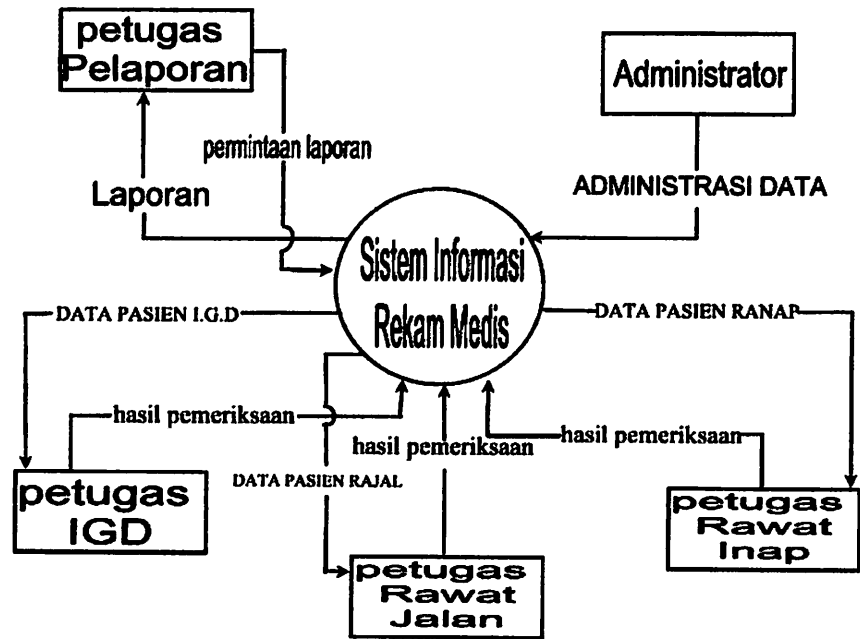
3.3 Perancangan Sistem

3.3.1 Diagram Alir Data (DAD)

DAD merupakan tahap perancangan aplikasi yang menggambarkan aliran dari data. Diagram tersebut memperlihatkan darimana data dimasukkan dan data apa yang akan dihasilkan dari setiap proses. Hal tersebut diperlukan untuk melihat detail proses dari aplikasi.

3.3.1.1 DAD Level 0

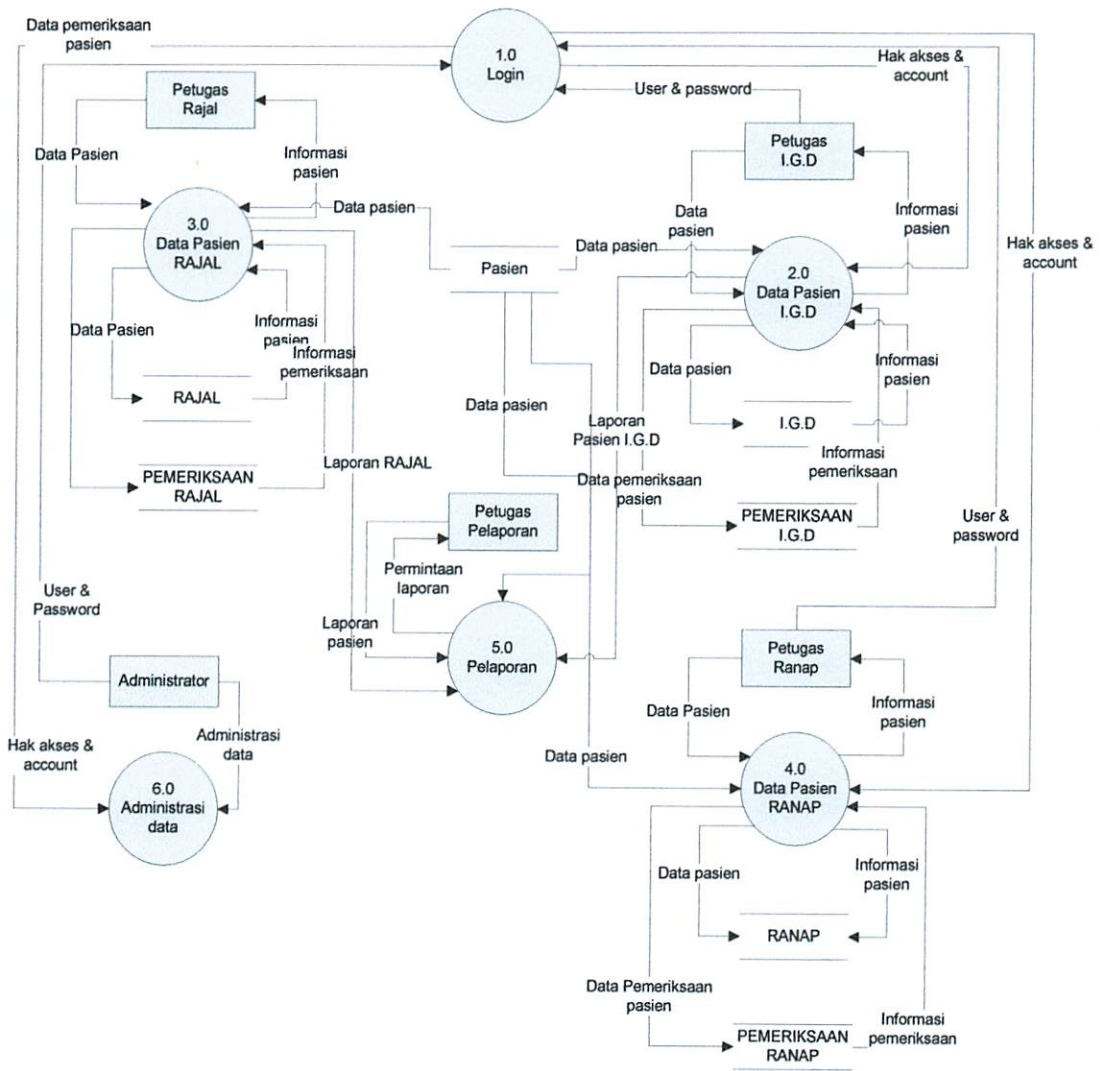
Diagram Alir Data level 0 akan menjelaskan gambaran sistem dan arus data yang digunakan. Gambaran sistem dan arus data ini selanjutnya dapat digunakan untuk mempermudah pembuatan sistem. Penggambaran system dan arus data dibuat dengan menggunakan Diagram Alir Data (DAD). Aliran data dalam sistem informasi rekam medis Rumah Sakit Umum Daerah Dompus memiliki enam entitas luar yang berinteraksi langsung dengan sistem, yaitu : petugas pendaftaran, perawat IGD, perawat poliklinik rawat jalan, perawat bangsal rawat inap, petugas pelaporan, dan administrator. Aliran data sistem informasi rekam medis Rumah Sakit Umum Daerah Dompus dapat diuraikan pada gambar 3.3. dibawah ini.



Gambar 3.3 DAD Level 0

3.3.1.2 DAD Level 1

Dari diagram alir data yang ditunjukkan pada gambar 3.3. dapat dijabarkan menjadi DAD level 1 Sistem Informasi Rekam Medis yang ditunjukkan pada gambar 3.4.



Gambar 3.4 DAD Level 1

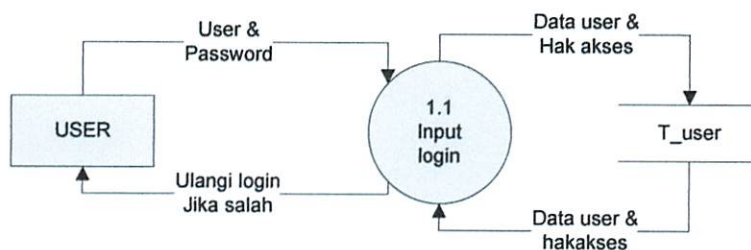
3.3.1.3 DAD Level 2

Dari gambar 3.4. menunjukkan bahwa proses pada sistem informasi rekam medis terdiri dari 6 proses, yaitu proses login user, proses I.G.D, proses RAJAL, proses RANAP, proses pelaporan data dan administrasi data.

3.3.1.3.1 Diagram Alir Data level 2 untuk proses 1

Proses awal pada Sistem Informasi Rekam Medis adalah login user, yaitu mengidentifikasi user yang akan mengakses aplikasi ini, apakah administrator, perawat, petugas rekam medis atau orang yang diijinkan mengakses aplikasi.

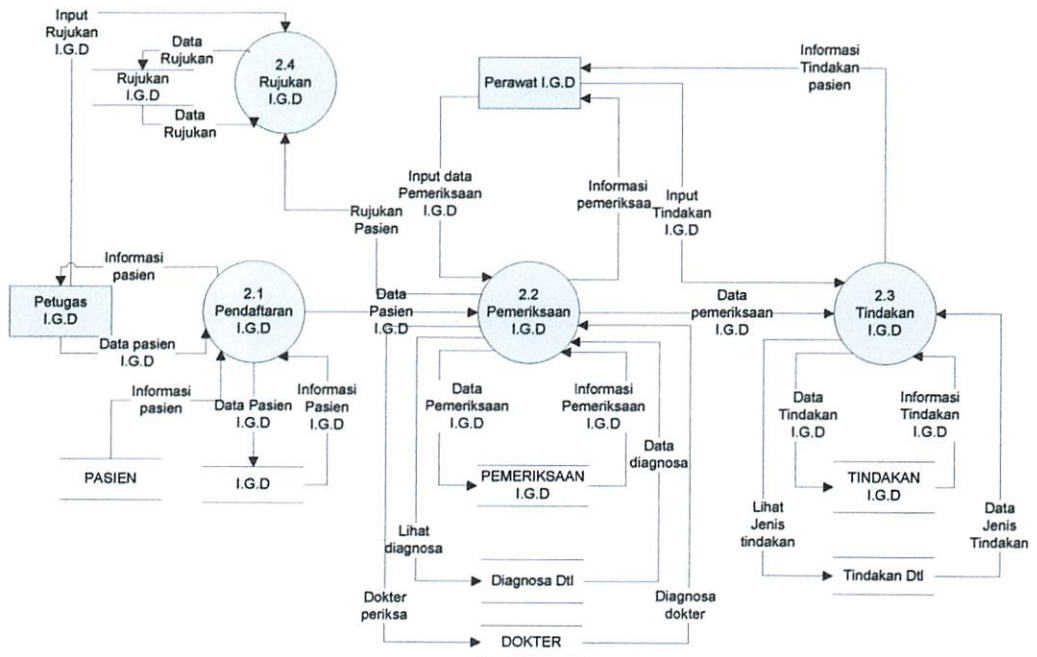
Setelah proses login sukses, maka user dapat melakukan proses selanjutnya sesuai hak akses.



Gambar 3.5 DAD level 2 proses login

3.3.1.3.2 Diagram Alir Data level 2 untuk proses 2

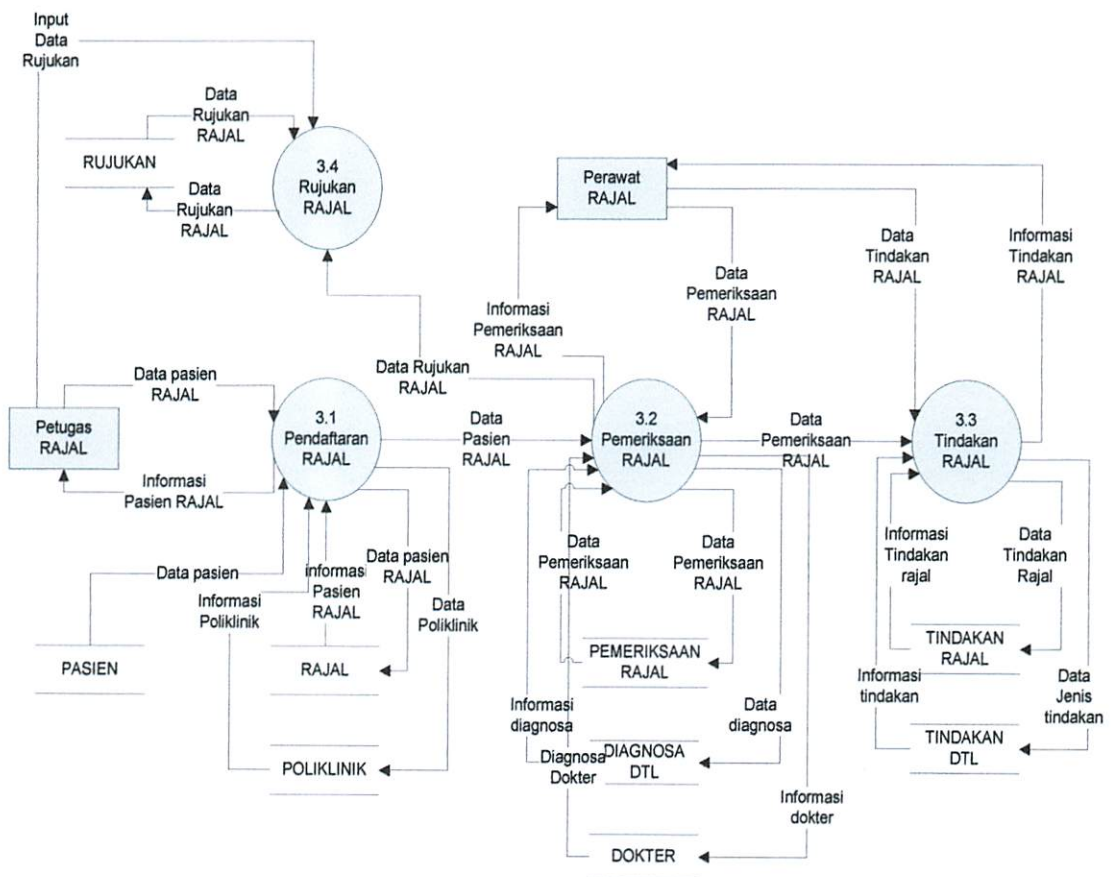
Proses selanjutnya adalah I.G.D, dimana pada proses ini dilakukan ketika pasien dirawat di Istalasi Gawat Darurat(I.G.D), ketika pasien belum memiliki No Rekam medis maka akan melakukan pendaftaran pasien baru, jika pasien sudah memiliki No Rekam Medis, maka pasien dapat langsung dilakukan tindakan medis.pada proses ini terdapat beberapa sub proses antara lain seperti gambar 3.5.



Gambar 3.5. DAD level 2 proses I.G.D

3.3.1.3.3 Diagram Alir Data level 2 untuk proses 3

Proses ke 3 adalah proses pada Bangsal Rawat Jalan, adalah pada saat pasien masuk dan dirawat pada Bangsal Rawat jalan. Sub prosesnya antara lain pendaftaran pasien Rawat Jalan, jika pasien belum mempunyai kartu pasien, maka pasien harus mendaftarkan data sosial baru, jika pasien sudah mempunyai kartu berobat maka dengan menunjukkan kartu dapat langsung mendaftarkan diri pada bangsal Rawat Jalan dan akan menjalani perawatan pada Poliklinik yang ada pada Rumah Sakit. Sub proses pada bangsal Rawat Jalan dapat dilihat pada gambar 3.6:



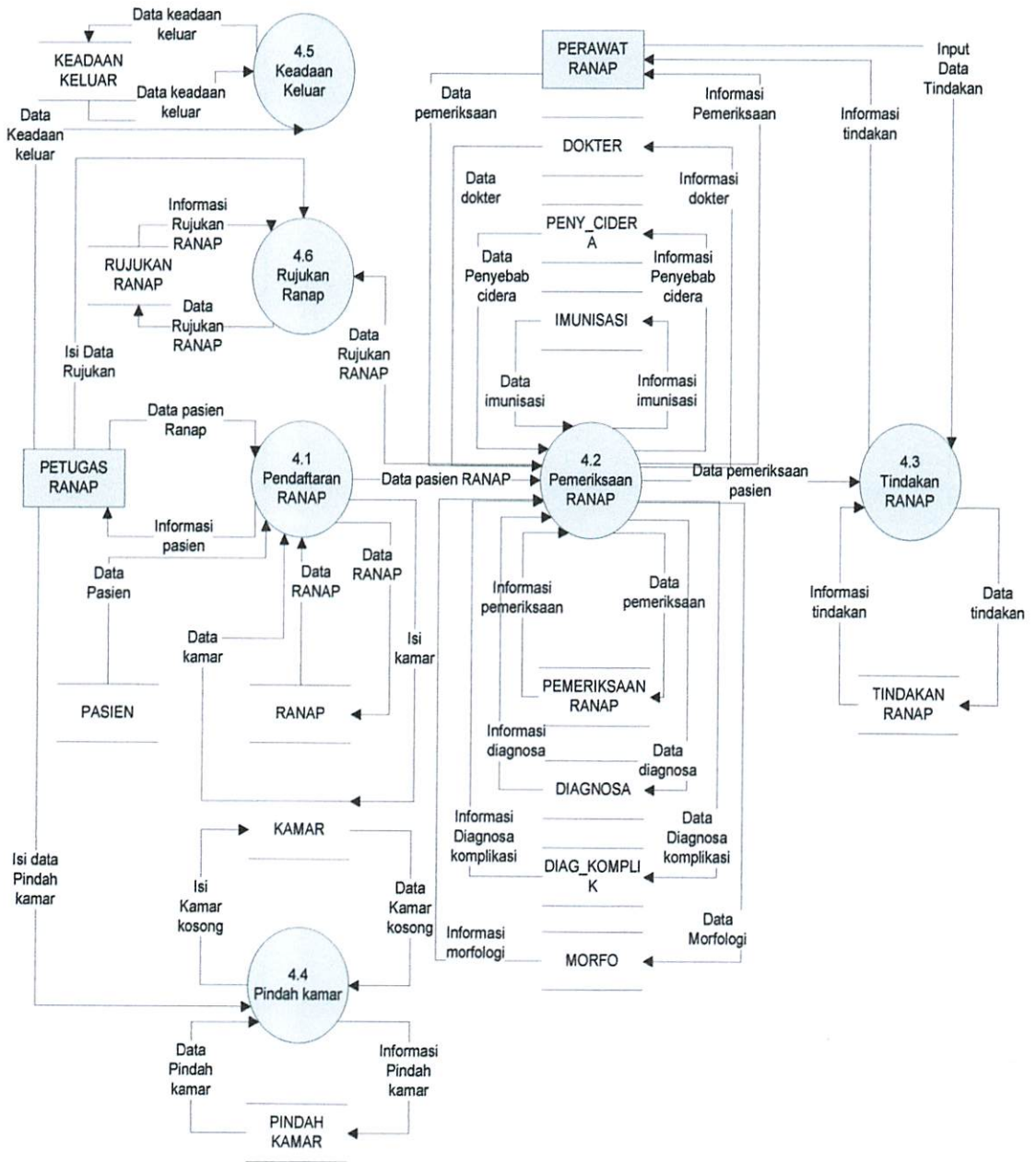
Gambar 3.6 DAD level 2 Proses 3 Rawat Jalan

3.3.1.3.4 Diagram Alir Data level 2 untuk proses 4

Pada proses ke empat yaitu proses pada Bangsal Rawat Inap, sama halnya pada proses Rawat Jalan, jika pasien yang dirawat pada bangsal Rawat Inap belum memiliki kartu maka akan didaftarkan pasien baru dan mendapatkan No Rekam medis, jika pasien sudah memiliki kartu Berobat, maka pasien akan langsung didaftarkan pada bangsal Rawat Inap dan dirawat

sesuai ruangan, untuk lebih jelas tentang sub-sub proses pada bangsal Rawat

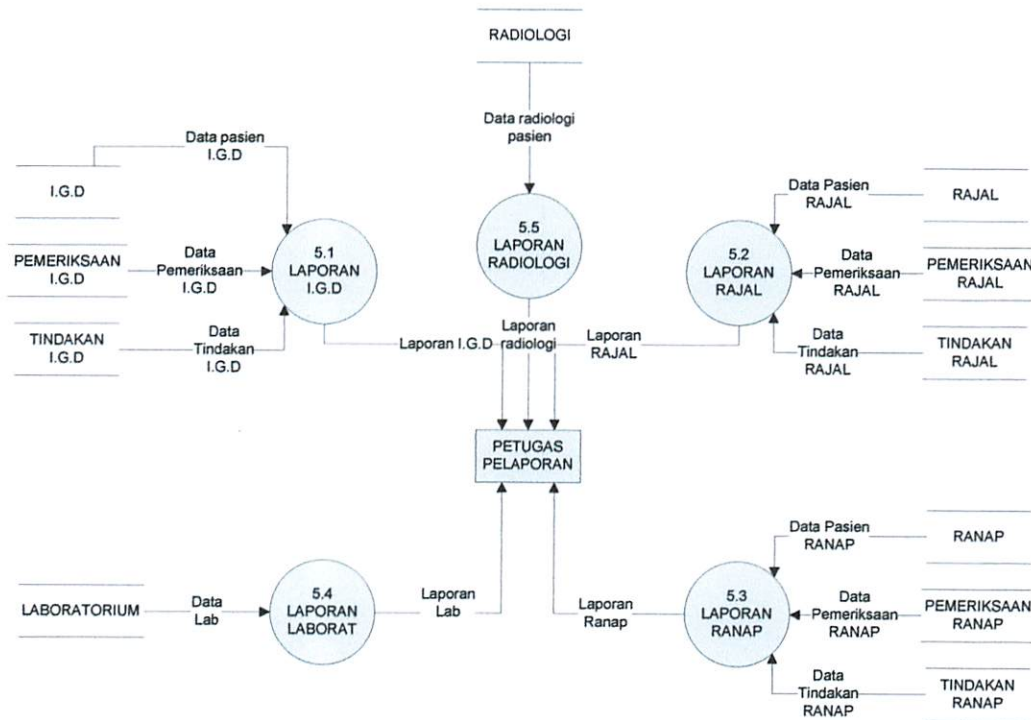
Inap ini dapat dilihat pada gambar 3.7



Gambar 3.7. DAD level 2 proses 4 RANAP

3.3.1.3.5 Diagram Alir Data level 2 untuk proses 5

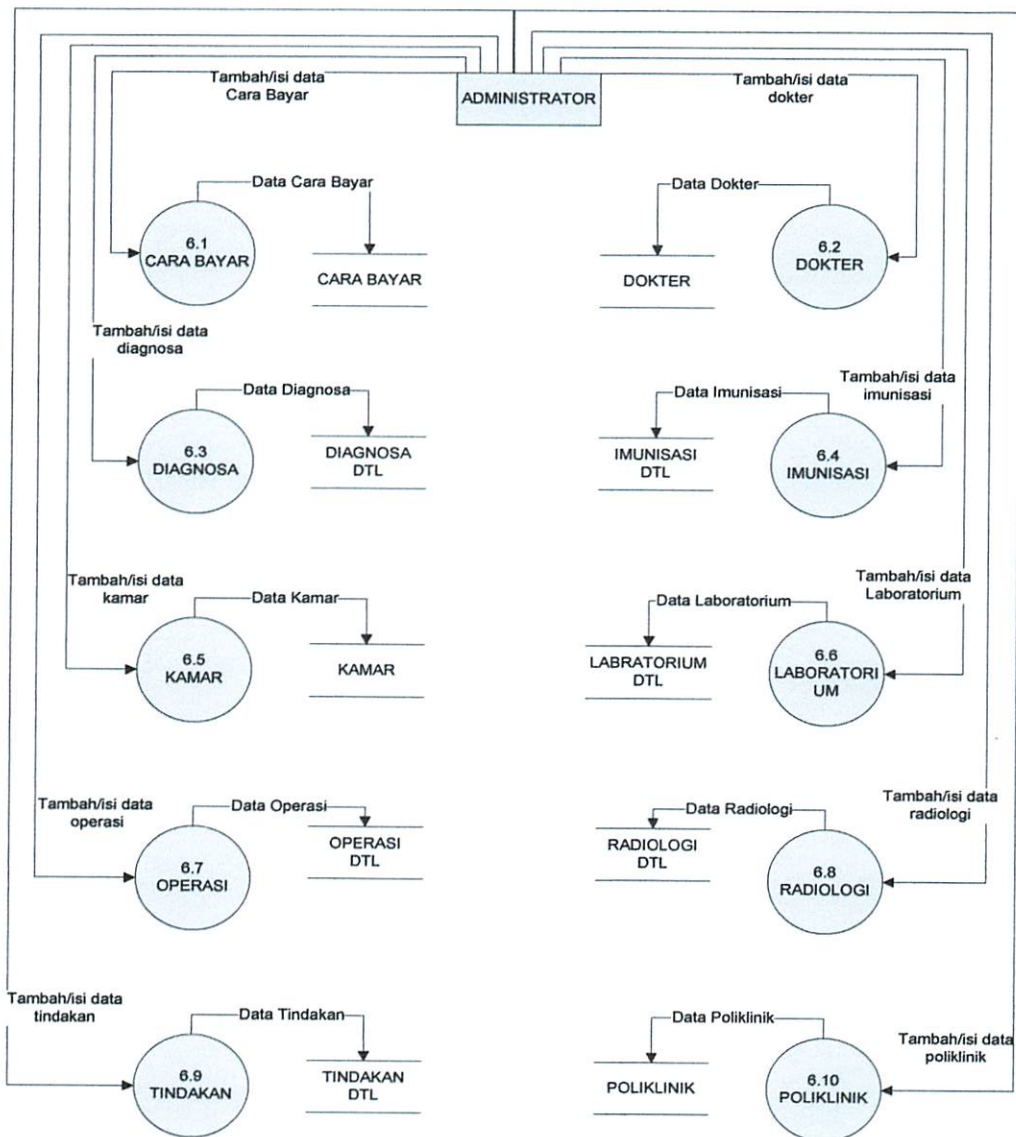
Proses ke 5 adalah proses pelaporan, dimana data-data yang telah di isi dapat dilihat dan di cetak sehingga mempermudah petugas dalam melakukan rekapitulasi maupun membuat statistik-statistik yang dibutuhkan. Sub-sub proses dapat dilihat pada gambar 3.8 .



Gambar 3.8. DAD level 2 proses 5 Pelaporan

3.3.1.3.6 Diagram Alir Data level 2 untuk proses 6

Pada proses terakhir yaitu proses administrasi data dimana proses ini khusus ditujukan bagi administrator untuk penambahan data-data yang dibutuhkan pada Sistem Informasi Rekam Medis seperti data dokter, cara bayar pasien, data Poliklinik, dan lain-lain.



Gambar 3.9. DAD level 2 proses 6

3.3.2 Desain Basisdata Rekam medis

3.3.2.1 Relasi Antar Tabel

Relasi antar tabel-tabel pemebentuk sistem pada basisdata *Rekam Medis* digambarkan dalam bentuk *Conceptual Data Model (CDM)* atau pemodelan data secara konsep dan dalam bentuk *Physical Data Model (PDM)* atau pemodelan data secar fisik, seperti yang ditunjukkan pada gambar berikut :

3.3.2.2 Struktur Tabel – Tabel Yang Digunakan

1. Tabel PASIEN

Tabel ini sebuah tabel untuk penyimpanan data pasien, berisi informasi sosial pasien dan data pasien lainnya.

Tabel 3.1 Struktur Tabel PASIEN

No.	Nama Kolom	Tipe Data	Keterangan
1.	NORM	Char(12)	<i>Primary Key (PK), NORekam Medis Pasien</i>
2.	NAMA	Varchar(30)	<i>Nama Pasien</i>
3.	TL	Varchar(25)	<i>Tempat Lahir Pasien</i>
4.	TTL	DATE	<i>Tanggal Lahir Pasien</i>
5.	PEND	Varchar(25)	<i>Pendidikan terakhir Pasien</i>
6.	GD	Varchar(4)	<i>Golongan darah Pasien</i>
7.	SEX	Varchar(20)	<i>Jenis Kelamin Pasien</i>
8.	AGAMA	Varchar(20)	<i>Agama yang dianut</i>
9.	STATUS_KAWIN	Varchar(20)	<i>Status Perkawinan Pasien</i>
10.	PEKER	Varchar(20)	<i>Pekerjaan Pasien</i>
11.	ALAMAT	Varchar(50)	<i>Alamat Pasien</i>
12.	KOTA	Varchar(20)	<i>Kota alamat Pasien</i>
13.	PROVINSI	Varchar(20)	<i>Provinsi tempat tinggal pasien</i>
14.	KD_POS	Varchar(6)	<i>Kode Pos pasien</i>
15.	TELP	Varchar(15)	<i>No Telpon Pasien</i>

16.	NO_KTP	Varchar(20)	<i>No Kartu Identitas Pasien</i>
17.	SA	Varchar(20)	<i>Status Alergi Pasien</i>
18.	NAMA_AYAH	Varchar(30)	<i>Nama Ayah Pasien</i>
19.	NAMA_IBU	Varchar(30)	<i>Nama Ibu Pasien</i>

2. Tabel IGD

Tabel ini berfungsi menyimpan data pendaftaran pasien IGD.

Noreg_igd pasien akan digunakan sebagai foreign key untuk Pemeriksaan Pasien dan tindakan yang diberikan kepada Pasien.

Tabel 3.2 Struktur Tabel IGD

No.	Nama Kolom	Tipe Data	Keterangan
1.	NOREG_IGD	CHAR (12)	<i>Primery Key (PK) No Registrasi Pasien IGD</i>
2.	NORM	CHAR(12)	<i>No Rekam Medis pasien (FK)</i>
3.	KD_CARA_BAYAR	CHAR (5)	<i>Cara Pembayaran Pasien (FK)</i>
4.	TGL_DAFTAR	DATE	<i>Tanggal Pendaftaran pasien</i>
5.	CARA_MASUK	CHAR (20)	<i>Cara masuk Pasien</i>

3. Tabel RAJAL

Tabel ini berfungsi menyimpan data Pendaftaran pasien Rawat jalan.

Tabel 3.3 Struktur Tabel RAJAL

No.	Nama Kolom	Tipe Data	Keterangan
1.	NOREG_RAJAL	CHAR (12)	<i>Primery Key (PK) No Pendaftaran Pasien RAJAL</i>
2.	NORM	CHAR(12)	<i>No Rekam medis Pasien (FK)</i>
3.	KD_CARA_BAYAR	CHAR (5)	<i>Cara Pembayaran Pasien (FK)</i>
4.	TGL_DAFTAR	DATE	<i>Tanggal Pendaftaran pasien</i>
5.	CARA_MASUK	CHAR (20)	<i>Cara masuk Pasien</i>
6.	POLI	CHAR(5)	<i>Kode Poliklinik Yang akan dimasuki (FK)</i>

4. Tabel POLIKLINIK

Tabel ini digunakan untuk menyimpan dan menampilkan data-data jenis poliklinik yang akan dikunjungi oleh pasien rawat jalan.

Tabel 3.4 Struktur Tabel POLIKLINIK

No.	Nama Kolom	Tipe Data	Keterangan
1.	POLI	CHAR(5)	<i>Primary Key (PK), Kode Poliklinik</i>
2.	NAMA_POLI	VARCHAR(30)	<i>Jenis Poliklinik</i>

5. Tabel RANAP

Tabel ini berfungsi menyimpan data pendaftaran pasien Rawat Inap.

Tabel 3.5 Struktur Tabel RANAP

No	Nama Kolom	Tipe Data	Keterangan
1.	NOREG_RANAP	CHAR (12)	<i>Primery Key (PK) No Pendaftaran Pasien RANAP</i>
2.	NORM	CHAR(12)	<i>No Rekam medis Pasien (FK)</i>
3.	KD_CARA_BAYAR	CHAR (5)	<i>Cara Pembayaran Pasien (FK)</i>
4.	TGL_DAFTAR	DATE	<i>Tgl Pendaftaran pasien</i>
5.	JAM_MSK	TIME	<i>Waktu masuk Pasien</i>
6.	CARA_PENERIMAAN	VARCHAR(50)	<i>Cara Penerimaan pasien pd bangsal</i>

			<i>RANAP</i>
7.	CARA_MASUK	VARCHAR(20)	<i>Cara masuk Pasien</i>
8.	NAMA_KERABAT	VARCHAR(30)	<i>Nama Penanggung Jawab pasien</i>
9.	ALAMAT_KERABAT	VARCHAR(50)	<i>Alamat Penanggung Jawab</i>
10.	TELP	VARCHAR(15)	<i>No Telpon kerabat</i>
11.	NO_KTP	VARCHAR(20)	<i>No identitas Kerabat</i>
12.	TGL_KELUAR	DATE	<i>Tgl Keluar pasien</i>
13.	NO_KAMAR	CHAR(5)	<i>No Kamar Tempat Pasien di Rawat</i>

6. Tabel CARA_BAYAR

Tabel ini adalah tabel yang digunakan sebagai tabel tampilan cara bayar pasien, apakah pasien memiliki asuransi atau tidak.

Tabel 3.6 Struktur Tabel CARA_BAYAR

No.	Nama Kolom	Tipe Data	Keterangan
1.	KD_CARA_BAYAR	CHAR(5)	<i>Primary Key (PK), Kode Cara bayar</i>
2.	CARA_BAYAR	VARCHAR(30)	<i>Cara bayar</i>

7. Tabel KAMAR

Tabel ini berfungsi untuk menyimpan data kamar dan apakah kamar tersebut berisi atau kosong. Jumlah bed dan sisa bed pada kamar.

Tabel 3.7 Struktur Tabel KAMAR

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_KAMAR	CHAR(5)	<i>Primary Key (PK), Kode Kamar</i>
2.	KELAS	VARCHAR(10)	<i>Tipe Kamar</i>
3.	JUMLAH_BED	INTEGER	<i>Jumlah Bed Dalam Kamar</i>
4.	SISA_BED	INTEGER	<i>Sisa Bed Dalam Kamar</i>

8. Tabel PEMERIKSAAN_IGD

Tabel ini digunakan untuk menyimpan data Pemeriksaan pasien bangsal I.G.D. diagnosa Penyakit dan Dokter yang memeriksa.

Tabel 3.8 Struktur Tabel PEMERIKSAAN_IGD

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_PEMERIKSAAN	CHAR(16)	<i>Primary Key (PK), No Pemeriksaan Pasien</i>
2.	NOREG_IGD	CHAR(12)	<i>No Registrasi IGD pasien (FK)</i>
3.	KD_DIAGNOSA	CHAR(10)	<i>Kd Diagnosa hasil Pemeriksaan pasien (FK)</i>

4.	ID_DOKTER	CHAR(5)	<i>Id Dokter yang memeriksa (FK)</i>
5.	TGL_PERIKSA	DATE	<i>Tgl Pemeriksaan Pasien</i>

9. Tabel PEMERIKSAAN RAJAL

Tabel ini berfungsi sebagai penyimpanan data-data pemeriksaan pasien seperti hasil diagnosa dan dokter yang memeriksa.

Tabel 3.9 Struktur Tabel PEMERIKSAAN_RAJAL

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_PEMERIKSAAN	CHAR(16)	<i>Primary Key (PK),No Pemeriksaan Pasien</i>
2.	NOREG_RAJAL	CHAR(12)	<i>No Registrasi RAJAL pasien (FK)</i>
3.	KD_DIAGNOSA	CHAR(10)	<i>Kd Diagnosa hasil Pemeriksaan pasien (FK)</i>
4.	ID_DOKTER	CHAR(5)	<i>Id Dokter yang memeriksa (FK)</i>
5.	TGL_PERIKSA	DATE	<i>Tgl Pemeriksaan Pasien</i>

10. Tabel PEMERIKSAAN_RANAP

Tabel ini digunakan untuk penyimpanan data-data pemeriksaan pasien

Rawat inap.

Tabel 3.10 Struktur Tabel PEMERIKSAAN_RANAP

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_PEMERIKSAAN	CHAR(16)	<i>Primary Key (PK),No Pemeriksaan Pasien</i>
2.	NOREG_RANAP	CHAR(12)	<i>No Registrasi RANAP pasien (FK)</i>
3.	KD_DIAGNOSA	CHAR(10)	<i>Kd Diagnosa hasil Pemeriksaan pasien (FK)</i>
4.	ID_DOKTER	CHAR(5)	<i>Id Dokter yang memeriksa (FK)</i>
5.	TGL_PERIKSA	DATE	<i>Tgl Pemeriksaan Pasien</i>

11. Tabel DIAGNOSA_KOMPLIK

Tabel ini digunakan bila terdapat Komplikasi Penyakit hasil pemeriksaan pasien rawat inap.

Tabel 3.11 Struktur Tabel DIAGNOSA_KOMPLIKASI

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_PEMERIKSAAN	CHAR(16)	<i>Primary Key (PK),No Pemeriksaan Pasien</i>
2.	NOREG_RANAP	CHAR(12)	<i>No Registrasi RANAP pasien (FK)</i>

3.	KD_DIAGNOSA	CHAR(10)	<i>Kd Diagnosa hasil Pemeriksaan pasien (FK)</i>
4.	ID_DOKTER	CHAR(5)	<i>Id Dokter yang memeriksa (FK)</i>
5.	TGL_PERIKSA	DATE	<i>Tgl Pemeriksaan Pasien</i>

12. Tabel MORFOLOGI DAN NEOPLASMA

Tabel ini digunakan untuk menyimpan data-data pemeriksaan Morfologi dan Neoplasma pasien rawat inap.

Tabel 3.12 Struktur Tabel MORFOLOGI DAN NEOPLASMA

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_PEMERIKSAAN	CHAR(16)	<i>Primary Key (PK),No Pemeriksaan Pasien</i>
2.	NOREG_RANAP	CHAR(12)	<i>No Registrasi RANAP pasien (FK)</i>
3.	KD_DIAGNOSA	CHAR(10)	<i>Kd Diagnosa hasil Pemeriksaan pasien (FK)</i>
4.	ID_DOKTER	CHAR(5)	<i>Id Dokter yang memeriksa (FK)</i>
5.	TGL_PERIKSA	DATE	<i>Tgl Pemeriksaan Pasien</i>

13. Tabel DIAGNOSA

Tabel ini digunakan untuk menyimpan dan menampilkan data-data jenis diagnosa yang berhubungan dengan tabel-tabel pemeriksaan, diagnosa komplikasi, dan morfologi pasien.

Tabel 3.13 Struktur Tabel DIAGNOSA

No.	Nama Kolom	Tipe Data	Keterangan
1.	KD_DIAGNOSA	CHAR(10)	<i>Primary Key (PK), Kode Diagnosa</i>
2.	DIAGNOSA	VARCHAR(500)	<i>Jenis Diagnosa</i>

14. Tabel PENYEBAB_CIDERA

Tabel ini digunakan sebagai tempat penyimpanan data-data pasien jika terjadi cedera dan penyebab luar.

Tabel 3.14 Struktur Tabel PENYEBAB_CIDERA

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_PEMERIKSAAN	CHAR(16)	<i>Primary Key (PK), No Pemeriksaan Pasien</i>
2.	NOREG_RANAP	CHAR(12)	<i>No Registrasi RANAP pasien (FK)</i>
3.	KD_PENY_CEDERA	CHAR(10)	<i>Kd Penyebab cedera pasien (FK)</i>
4.	ID_DOKTER	CHAR(5)	<i>Id Dokter yang memeriksa (FK)</i>

5.	TGL_PERIKSA	DATE	<i>Tgl Pemeriksaan Pasien</i>
----	-------------	------	-------------------------------

15. Tabel PENYEBAB_CIDERA_DTL

Tabel ini adalah tabel yang fungsinya menampilkan dan menyimpan data-data Penyebab cedera dan penyebab luar yang dialami pasien yang berrelasi dengan tabel PENYEBAB_CIDERA.

Tabel 3.15 Struktur Tabel PENYEBAB_CIDERA_DTL

No.	Nama Kolom	Tipe Data	Keterangan
1.	KD_PENY_CIDERA	CHAR(10)	<i>Primary Key (PK), Kode Penyebab cedera</i>
2.	PENYEBAB_CIDERA	VARCHAR(500)	<i>Jenis Penyebab cedera</i>

16. Tabel IMUNISASI

Tabel ini digunakan untuk menyimpan data-data Imunisasi yang diberikan kepada pasien.

Tabel 3.16 Struktur Tabel IMUNISASI

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_PEMERIKSAAN	CHAR(16)	<i>Primary Key (PK), No Pemeriksaan Pasien</i>
2.	NOREG_RANAP	CHAR(12)	<i>No Registrasi RANAP pasien (FK)</i>

3.	KD_IMUN	CHAR(10)	<i>Kd Imunisasi Pasien (FK)</i>
4.	ID_DOKTER	CHAR(5)	<i>Id Dokter yang memeriksa (FK)</i>
5.	TGL_PERIKSA	DATE	<i>Tgl Pemeriksaan Pasien</i>

17. Tabel IMUNISASI_DTL

Tabel ini digunakan untuk menampilkan dan menyimpan data-data jenis Imunisasi pasien dan tabel ini direlasikan dengan Tabel IMUNISASI.

Tabel 3.17 Struktur Tabel IMUNISASI_DTL

No.	Nama Kolom	Tipe Data	Keterangan
1.	KD_IMUN	CHAR(10)	<i>Primary Key (PK), Kode Imunisasi</i>
2.	IMUNISASI	VARCHAR(500)	<i>Jenis Imunisasi</i>

18. Tabel TINDAKAN_IGD

Tabel ini digunakan untuk menyimpan data tindakan medis yang dilakukan dokter atau perawat kepada pasien.

Tabel 3.18 Struktur Tabel TINDAKAN_IGD

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_TINDAKAN	CHAR(16)	<i>Primary Key (PK),No Tindakan Pasien</i>

2.	NOREG_IGD	CHAR(12)	<i>No Registrasi IGD pasien (FK)</i>
3.	KD_TINDAKAN	CHAR(10)	<i>Kd Tindakan pasien (FK)</i>
5.	TGL_PERIKSA	DATE	<i>Tgl Pemberian tindakan Pasien</i>

19. Tabel TINDAKAN_RAJAL

Tabel ini digunakan untuk Menyimpan data-data tindakan medis yang diberikan oleh dokter atau petugas medis kepada pasien Rawat Jalan.

Tabel 3.19 Struktur Tabel TINDAKAN_RAJAL

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_TINDAKAN	CHAR(16)	<i>Primary Key (PK),No Tindakan Pasien</i>
2.	NOREG_RAJAL	CHAR(12)	<i>No Registrasi RAJAL pasien (FK)</i>
3.	KD_TINDAKAN	CHAR(10)	<i>Kd Tindakan pasien (FK)</i>
5.	TGL_PERIKSA	DATE	<i>Tgl Pemberian tindakan Pasien</i>

20. Tabel TINDAKAN_RANAP

Tabel ini digunakan untuk menyimpan data-data tindakan medis kepada pasien Rawat Inap.

Tabel 3.20 TINDAKAN_RANAP

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_TINDAKAN	CHAR(16)	<i>Primary Key (PK),No Tindakan Pasien</i>
2.	NOREG_RANAP	CHAR(12)	<i>No Registrasi RANAP pasien (FK)</i>
3.	KD_TINDAKAN	CHAR(10)	<i>Kd Tindakan pasien (FK)</i>
5.	TGL_PERIKSA	DATE	<i>Tgl Pemberian tindakan Pasien</i>

21. Tabel TINDAKAN_DTL

Tabel ini berfungsi untuk menyimpan dan menampilkan jenis tindakan yang diberikan pada pasien pada tabel TINDAKAN_IGD, TINDAKAN_RAJAL, dan TINDAKAN_RANAP.

Tabel 3.21 Struktur Tabel TINDAKAN_DTL

No.	Nama Kolom	Tipe Data	Keterangan
1.	KD_TINDAKAN	CHAR(10)	<i>Primary Key (PK), Kode Tindakan</i>
2.	TINDAKAN	VARCHAR(500)	<i>Jenis Tindakan</i>

22. Tabel RUJUKAN_IGD

Tabel ini digunakan untuk menyimpan data rujukan pasien IGD apakah masuk rawat inap atau dirujuk kerumah sakit lain.

Tabel 3.22 Struktur Tabel RUJUKAN_IGD

No.	Nama Kolom	Tipe Data	Keterangan
1.	ID_RUJUK	CHAR(16)	<i>Primary Key (PK),No Tindakan Pasien</i>
2.	NOREG_IGD	CHAR(12)	<i>No Registrasi IGD pasien (FK)</i>
3.	TGL_RUJUK	DATE	<i>Tgl pasien dirujuk</i>
5.	TUJUAN	VARCHAR(40)	<i>Tujuan Rujuk Pasien</i>

23. Tabel RUJUKAN_RAJAL

Tabel ini digunakan untuk menyimpan data-data rujukan pasien bangsal Rawat Jalan.

Tabel 3.23 Struktur Tabel RUJUKAN_RAJAL

No.	Nama Kolom	Tipe Data	Keterangan
1.	ID_RUJUK	CHAR(16)	<i>Primary Key (PK),No Tindakan Pasien</i>
2.	NOREG_RAJAL	CHAR(12)	<i>No Registrasi RAJAL pasien (FK)</i>
3.	TGL_RUJUK	DATE	<i>Tgl pasien dirujuk</i>
5.	TUJUAN	VARCHAR(40)	<i>Tujuan Rujuk Pasien</i>

24. Tabel RUJUKAN_RANAP

Tabel ini digunakan untuk menyimpan data-data Rujukan Pasien Rawat Inap.

Tabel 3.24 Struktur Tabel RUJUKAN_RANAP

No.	Nama Kolom	Tipe Data	Keterangan
1.	ID_RUJUK	CHAR(16)	<i>Primary Key (PK), No Tindakan Pasien</i>
2.	NOREG_RANAP	CHAR(12)	<i>No Registrasi RANAP pasien (FK)</i>
3.	TGL_RUJUK	DATE	<i>Tgl pasien dirujuk</i>
5.	TUJUAN	VARCHAR(40)	<i>Tujuan Rujuk Pasien</i>

25. Tabel DOKTER

Fungsinya untuk menyimpan data dokter dan berisi informasi yang diperlukan tentang dokter tersebut.

Tabel 3.25 Struktur Tabel DOKTER

No.	Nama Kolom	Tipe Data	Keterangan
1.	ID_DOKTER	Char(10)	<i>Primary Key (PK), Kode Dokter</i>
2.	NAMA	Varchar(30)	<i>Nama Dokter</i>
3.	SPEKIALIST	Varchar(20)	<i>Keahlian Dokter</i>

26. Tabel OPERASI

Tabel ini digunakan untuk menyimpan data operasi pasien.

Tabel 3.26 Struktur Tabel OPERASI

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_OPERASI	CHAR(16)	<i>Primary Key (PK), No Operasi</i>
2.	NORM	VARCHAR(10)	<i>No Rekam medis pasien</i>
3.	KD_OPERASI	CHAR(15)	<i>Kd operasi Pasien</i>
4.	ID_DOKTER	CHAR(5)	<i>Id dokter yang melakukan operasi</i>
5.	TGL_OPERASI	DATE	<i>Tgl operasi pasien</i>
6.	JENIS_ANASTESI	VARCHAR(20)	<i>Jenis anastesi Operasi</i>

27. Tabel OPERASI_DTL

Tabel Operasi_dtl ini digunakan untuk menampilkan serta menyimpan data-data Jenis Operasi yang akan dilakukan Pasien.

Tabel 3.27 Struktur Tabel LABORATORIUM

No.	Nama Kolom	Tipe Data	Keterangan
1.	KD_OPERASI	CHAR(10)	<i>Primary Key (PK), Kode Operasi</i>
2.	OPERASI	VARCHAR(50)	<i>Jenis Operasi</i>

28. Tabel LABORATORIUM

Tabel ini digunakan untuk menyimpan data-data Pemeriksaan Laboratorium Pasien.

Tabel 3.28 Struktur Tabel LABORATORIUM

No.	Nama Kolom	Tipe Data	Keterangan
1.	NO_LAB	CHAR(16)	<i>Primary Key (PK), no pemeriksaan LAB</i>
2.	NORM	VARCHAR(10)	<i>No Rekam medis pasien</i>
3.	KD_LAB	CHAR(15)	<i>Kd Pemeriksaan LAB Pasien</i>
4.	TGL_PERIKSA	DATE	<i>Tgl Periksa LAB pasien</i>
5.	HASIL	VARCHAR(20)	<i>Hasil pemeriksaan</i>
6.	HASIL NORMAL	VARCHAR(20)	<i>Hasil normal pemeriksaan</i>
7.	KET	VARCHAR(40)	<i>Keterangan hasil pemeriksaan</i>

29. Tabel LAB_DTL

Tabel ini berfungsi menyimpan dan menampilkan data-data jenis Pemeriksaan Laboratorium yang dilakukan pasien.

Tabel 3.29 Struktur Tabel LAB_DTL

No.	Nama Kolom	Tipe Data	Keterangan
1.	KD_LAB	CHAR(10)	<i>Primary Key (PK), Kode Tindakan</i>
2.	PEMERIKSAAN	VARCHAR(50)	<i>Jenis Tindakan</i>

30. Tabel RADIOLOGI

Tabel ini digunakan untuk menyimpan data-data pemeriksaan Radilologi pasien.

Tabel 3.30 Struktur Tabel RADIOLOGI

No.	Nama Kolom	Tipe Data	Keterangan
1.	KD_PEMERIKSAAN	CHAR(16)	<i>Primary Key (PK), no pemeriksaan Radiologi</i>
2.	NORM	VARCHAR(10)	<i>No Rekam medis pasien</i>
3.	KD_RADIOLOGI	CHAR(15)	<i>Kd Pemeriksaan radiologi Pasien</i>
4.	TGL_PERIKSA	DATE	<i>Tgl Periksa LAB pasien</i>
5.	KETERANGAN	VARCHAR(20)	<i>Keterangan</i>

			<i>Radiologi pasien</i>
6.	HASIL	VARCHAR(20)	<i>Hasil pemeriksaan radiologi</i>

31. Tabel RADIOLOGI_DTL

Tabel ini berfungsi menyimpan serta menampilkan data-data jenis radiologi serta kode radiologi yang akan dilakukan pasien.

Tabel 3.31 Struktur Tabel LAB_DTL

No.	Nama Kolom	Tipe Data	Keterangan
1.	KD_RADIOLOGI	CHAR(10)	<i>Primary Key (PK), Kode Tindakan</i>
2.	RADIOLOGI	VARCHAR(50)	<i>Jenis Tindakan</i>

32. Tabel PINDAH_KAMAR

Tabel ini digunakan untuk menyimpan data-data jika ada pasien Rawat Inap yang melakukan pindah kamar rawat. Sehingga petugas rawat inap dapat mengetahui selama dirawat pasien pindah kamar berapa kali.

Tabel 3.32 Struktur Tabel PINDAH_KAMAR

No.	Nama Kolom	Tipe Data	Keterangan
1.	ID_PINDAH_KAMAR	CHAR(16)	<i>Primary Key (PK), id pindah kamar</i>
2.	TGL_PINDAH	DATE	<i>Tgl pindah kamar</i>
3.	NOREG_RANAP	CHAR(12)	<i>Noreg rawat inap pasien (FK)</i>

4.	NO_KAMAR	CHAR(5)	No Kamar asal(FK)
5.	NO_KAMAR_TUJUAN	CHAR(5)	No kamar tujuan pindah

33. Tabel KEADAAN_KELUAR

Tabel ini digunakan menyimpan data-data keadaan keluar pasien.

Tabel 3.33 Struktur Tabel KEADAAN_KELUAR

No.	Nama Kolom	Tipe Data	Keterangan
1.	ID_KEADAAN	CHAR(16)	Primary Key (PK), Kode Keadaan
2.	NOREG_RANAP	CHAR(12)	Noreg Rawat inap pasien (FK)
3.	INFEK_NOS	VARCHAR(100)	Infeksi nosokomial pasien
4.	PENY_INFK_NOS	VARCHAR(100)	Penyebab infeksi nosokimial
5.	NUKLIR	VARCHAR(100)	Radiasi nuklir
6.	TRANSFUSI_DARAH	VARCHAR(100)	Transfusi darah
7.	KEADAAN_KELUAR	VARCHAR(100)	Keadaan keluar pasien
8.	CARA_KELUAR	VARCHAR(100)	Cara Keluar Pasien
9.	TGL_KELUAR	DATE	Tgl keluar pasien

34. Tabel TBL_USER

Tabel ini adalah tabel yang menyimpan data-data User yang dapat menggunakan atau menjalankan aplikasi Sistem Informasi Rekam Medis dan juga Hak aksesnya.

Tabel 3.34 Struktur Tabel TBL_USER

No.	Nama Kolom	Tipe Data	Keterangan
1.	USER_ID	VARCHAR(15)	<i>Primary Key (PK), User name untuk login</i>
2.	NAMA	VARCHAR(50)	<i>Nama pengguna</i>
3.	PASSWD	VARCHAR(15)	<i>Password user</i>
4.	GRP	TINYINT	<i>Group User</i>

3.3.3 Desain Antarmuka Aplikasi

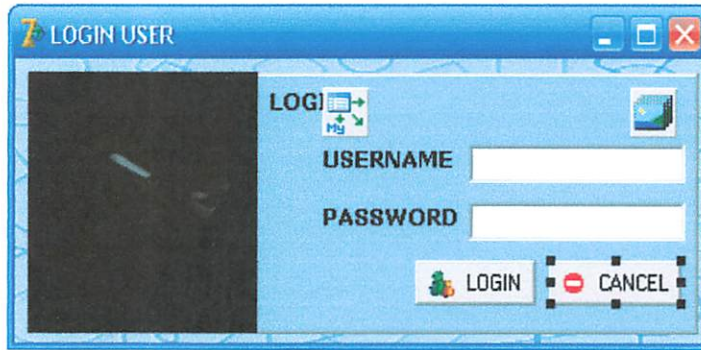
Selain struktur tabel yang sudah normal, tampilan antarmuka yang mudah dimengerti dan dioperasikan juga menentukan apakah suatu aplikasi dapat diterima oleh semua pengguna. Jadi, desain antarmuka aplikasi juga haruslah dapat membantu dan mempercepat pemrosesan data.

Pada aplikasi ini desain antarmuka terdiri dari desain halaman Login, desain halaman menu utama, desain halaman entri data, dan desain halaman laporan.

3.3.3.1 Desain Halaman Login

Pada halaman Login, terdapat dua buah edit text yang fungsinya sebagai input *username* dan *Password* bagi user, terdapat juga dua buah button yang

berfungsi sebagai tombol login dan tombol cancel yang berfungsi untuk menutup form login. Terdapat juga imagelist dan image untuk gambar background serta icon pada tombol dan datasource sebagai komponen untuk koneksi dengan tabel user pada database.

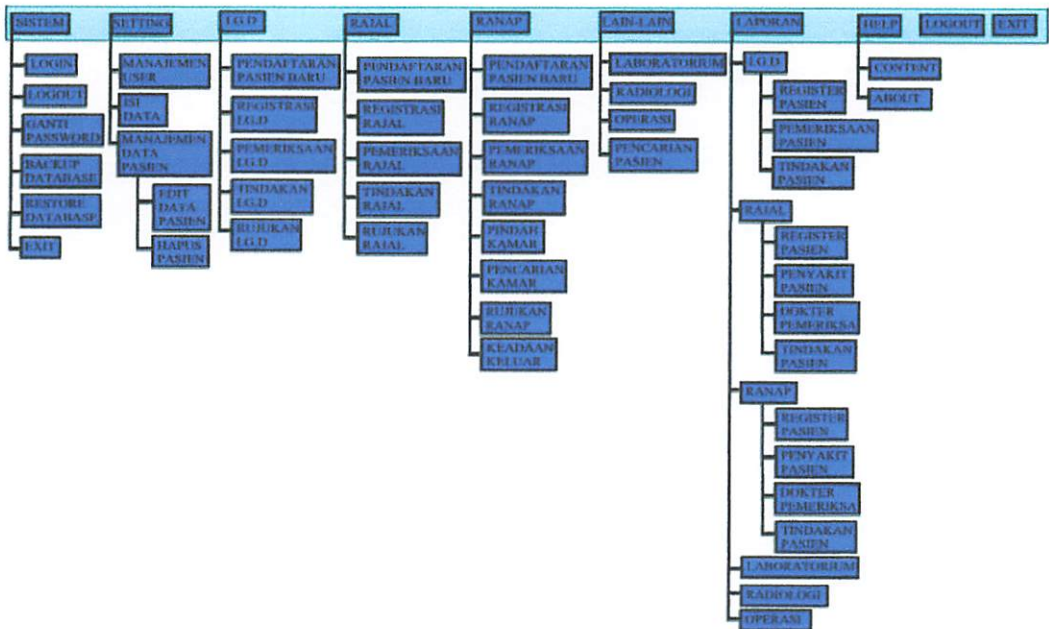


Gambar 3.14 Desain halaman Login

3.3.3.2 Desain Halaman Menu Utama

Halaman menu utama sistem dibuat dengan cara pop-up yaitu menu akan bergeser kebawah dan mengeluarkan sub-menu pada saat diklik oleh user.

Adapun desain dari halaman menu utama sebagai berikut :



Gambar 3.15 Menu Utama

3.3.3.3 Desain Form Isi Data Pasien

Desain form isi data pasien ini dibuat dengan edit-edit text dan combobox sebagai tempat input text dari user tentang data sosial pasien, button, dan Dbgrid sebagai tabel penampil data yang ada pada tabel pasien. Fungsi dari form ini adalah sebagai tempat penginputan data pasien baru, yaitu data pasien yang pertama kali dirawat di rumah sakit.

The image shows a software window titled "PENDAFTARAN PASIEN BARU". The window contains a form with two columns of input fields. The left column includes fields for "NORM", "NAMA", "TEMPAT LAHIR", "TTL", "JENIS KELAMIN", "NO. IDENTITAS", "AGAMA", "GOLONGAN DARAH", "PENDIDIKAN", and "PEKERJAAN". The right column includes fields for "STATUS PERKAWINAN", "ALAMAT", "KOTA", "KD POS", "PROVINSI", "TELP", "STATUS ALERGI", "NAMA_AYAH", and "NAMA_IBU". At the bottom of the form, there is a section labeled "DATA PASIEN" which contains a table with one row and one column. To the right of the table are five buttons: "SAVE", "CETAK", "EDIT", "CANCEL", and "CLOSE".

Gambar 3.16 Desain Isi Data Pasien

3.3.3.4 Desain Form Register pasien

Form ini fungsinya sebagai tempat registrasi pasien pada bangsal rumah sakit, baik itu Instalasi Gawat Darurat, Rawat Jalan, maupun Rawat Inap. Terdiri dari beberapa edit text dan mask edit yang berfungsi sebagai edit yang diisikan sesuai dengan kode yang telah ditentukan sebelumnya. berikut tampilan form register I.G.D.

PENDAFTARAN PASIEN I.G.D

TANGGAL MASUK:

NOREG IGD:

NORM:

CARA MASUK:

CARA BAYAR:

DATA PASIEN :

NORM: DBEdit1

NAMA: DBEdit2

SEX: DBEdit6

GD: DBEdit7

ALAMAT: RzDBMemo1

TELP: DBEdit4

NOKTP: DBEdit5

TABEL DATA PASIEN I.G.D

TGL DAFTAR	NOREG IGD	NORM	NAMA	CARA MASUK	KD CARA BAYAR

DAFTAR PASIEN BARU CANCEL CLOSE

Gambar 3.17 Desain Form Register Pasien

3.3.3.5 Desain Form Pemeriksaan Pasien

Form pemeriksaan pasien digunakan untuk antarmuka penyimpanan data pemeriksaan pasien pada masing-masing bangsal. Pada form ini terdapat *page control* yang digunakan untuk menyimpan jenis pemeriksaan pasien seperti diagnosa penyakit pasien, morfologi dan imunisasi pasien.

Gambar 3.18 Desain Form Pemeriksaan Pasien

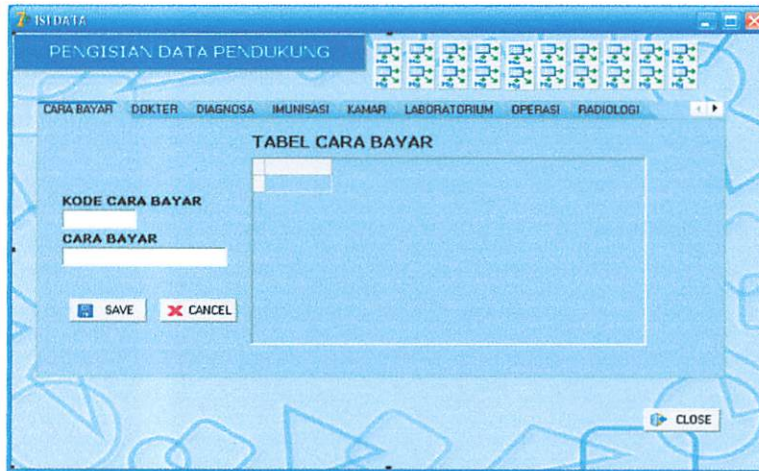
3.3.3.6 Desain Form Manajemen User

Form manajemen user digunakan sebagai tempat pengelolaan data user, baik menambah user baru, mengedit, ataupun menghapus user.

Gambar 3.19 Desain Form Manajemen User

3.3.3.7 Desain Form Pengisian Data

Form ini digunakan sebagai tempat penyimpanan data-data tambahan pada aplikasi seperti halnya data kamar, data dokter, data diagnosa, serta data tindakan medis. Form ini dibangun dengan menggunakan beberapa edit text, button, serta komponen *page control* sebagai *switch* menu.

The image shows a screenshot of a software application window titled 'ISIDATA'. The main window has a blue header with the text 'PENGISIAN DATA PENDUKUNG'. Below the header is a navigation menu with several tabs: 'CARA BAYAR', 'DDKTER', 'DIAGNOSA', 'IMUNISASI', 'KAMAR', 'LABORATORIUM', 'OPERASI', and 'RADIOLOGI'. The 'CARA BAYAR' tab is currently selected. The main content area is titled 'TABEL CARA BAYAR' and contains a large empty table. To the left of the table are two text input fields: 'KODE CARA BAYAR' and 'CARA BAYAR'. Below these fields are two buttons: 'SAVE' and 'CANCEL'. At the bottom right of the window is a 'CLOSE' button. The background of the window has a light blue pattern of overlapping circles and lines.

Gambar 3.20 Desain Form Isi Data

3.3.3.8 Desain Form Operasi

form ini digunakan sebagai antarmuka penyimpanan data operasi pasien. Form ono dibuat dengan beberapa komponen edit text serta combo box, beberapa button dan image list.

The screenshot shows a software window titled "DAFTAR OPERASI PASIEN". It features several input fields and a table. On the left, there are fields for "TGL OPERASI", "NO OPERASI", "KD OPERASI", "NORM", "KD DOKTER", and "JENIS ANESTESI". On the right, under "DATA PASIEN:", there are fields for "NORM" (00000001) and "NAMA" (SODIKIN). Below that, under "DATA DOKTER:", there are fields for "ID_DOKTER" (01), "NAMA_DOKTER" (dr. H.Ahmad Feisol, SpA), and "SPECIALIST". At the bottom, there is a table titled "DAFTAR TABEL OPERASI PASIEN" with columns for "TGL OPERASI", "NO OPERASI", "KD OPERASI", and "OPERASI". The table is currently empty. At the bottom right, there are three buttons: "SAVE", "CANCEL", and "CLOSE".

Gambar 3.21 Desain Operasi Pasien

3.3.3.9 Desain Form Laporan

Form laporan dibuat dengan menggunakan komponen *Quick Report*, laporan-laporan yang terdapat pada aplikasi ini antara lain :

1. Laporan pasien Instalasi Gawat Darurat

Pada laporan ini terdiri dari laporan register pasien I.G.D menurut tanggal pendaftaran pasien, pemeriksaan pasien I.G.D yang berisi Hasil pemeriksaan pasien pada bangsal I.G.D, dan laporan tindakan medis pasien I.G.D.

LAPORAN REGISTRASI I.G.D

MASUKAN TANGGAL AWAL DAN AKHIR

TGL DAFTAR: [] S/D []

PREVIEW CANCEL

RUMAH SAKIT UMUM DAERAH DOMPU
JL. KESEHATAN NO. 1 DOMPU
TELP 0373-21411

LAPORAN INDEX REGISTRASI PASIEN I.G.D

NO	TGL DAFTAR	NOREG RAJAL	NORM	NAMA	CARA MASUK	CARA BAYAR
TANGGAL LAPORAN	Tgl_awal		Tgl_akhir			
[Detail]	TGL_DAFTAR	NOREG_RAJAL	NORM	NAMA	CARA_MASUK	CARA_BAYAR

TGL CETAK: [Date]

Gambar 3.22 Desain Laporan pasien I.G.D

2. Laporan Pasien Rawat Jalan

Pada laporan ini berisi laporan-laporan tentang pasien yang dirawat pada bangsal Rawat Jalan. Isi dari laporan tersebut antara lain register pasien Rawat Jalan, Penyakit rawat jalan, index dokter yang merawat, serta tindakan pasien Rawat Jalan.

LAPORAN PENYAKIT RAWAT JALAN

RUMAH SAKIT UMUM DAERAH DOMPU
JL. KESEHATAN NO. 1 DOMPU
TELP 0373-21411

LAPORAN INDEX PENYAKIT PASIEN RAWAT JALAN

NO	TGL PEMERIKSAAN	NOREG RAJAL	NORM	NAMA	DOKTER PERIKSA	KD DIAGNOSA	DIAGNOSA
PENYAKIT	DIAGNOSA						
[Detail]	TGL_PERIKSA	NOREG_RAJAL	NORM	NAMA	NAMA_DOKTER	ID_DIAGNOSA	DIAGNOSA

[JUMLAH PASIEN PENDERITA] [Detail count]

TGL CETAK: [Date]

Gambar 3.23 Desain Laporan Penyakit Rawat Jalan

3. Laporan pasien Rawat Inap

Sama halnya dengan laporan pasien Rawat Jalan, terdapat laporan register pasien Rawat Inap, Penyakit Raat Inap, dokter, tindakan medis rawat inap, juga terdapat laporan pindah kamar pasien.

LAPORAN DIAGNOSA KOMPLIKASI PASIEN RAWAT INAP

RUMAH SAKIT UMUM DAERAH DOMPU
JL. KESEHATAN NO. 1 DOMPU
TELP 0373-21411

LAPORAN INDEX PENYAKIT PASIEN RAWAT INAP

NO	TGL PEMERIKSAAN	NOREG RAJAL	NORM	NAMA	DOKTER PERIKSA	KD_DIAGNOS
PENYAKIT		DIAGNOSA				
Detail	TGL_PERIKSA	NOREG_RANA	NORM	NAMA	NAMA_DOKTER	KD_DIAGNOS
						JUMLAH PASIEN PENDERITA
TGL CETAK [Date]						

Gambar 3.24 Laporan Diagnosa Rawat Inap

4. Laporan Laboratorium

Laporan ini akan berisi data-data pemeriksaan laboratorium pasien pada Rumah Sakit menurut tanggal pemeriksaan.

5. Laporan Radiologi

Laporan ini berisi laporan dari data-data radiologi pasien menurut tanggal pemeriksaan radiologi.

6. Laporan Operasi

Laporan ini berisi Laporan dari data-data Operasi pasien pada Rumah sakit menurut tanggal Operasi.

BAB IV

IMPLEMENTASI DAN PENGUJIAN HASIL

4.1. Implementasi Sistem

Aplikasi Sistem Informasi Rekam Medis ini dibangun dengan database MySQL 5.0 dan Borland Delphi 7.0 sebagai user interface dan sebagai media penyimpanan, pengolahan, pelaporan dan pendistribusian data pada Rumah Sakit Umum Daerah Dompus.

Hasil implementasi dari Sistem Informasi Rekam Medis ini mencakup pendaftaran pasien baru, pendaftaran bangsal I.G.D, Pendaftaran pasien Rawat Jalan, dan Pendaftaran pasien rawat inap, Pemeriksaan pasien I.G.D, Pemeriksaan pasien Rawat Jalan, Pemeriksaan pasien Rawat Inap, Tindakan pasien I.G.D, Tindakan pasien Rawat Jalan, Tindakan Pasien Rawat Inap, Rujukan pasien I.G.D, Rawat Jalan, Rawat inap, pindah kamar pasien rawat inap, Pemeriksaan Laboratorium, Radiologi, dan Operasi pasien, keadaan keluar pasien dan laporan-laporan yang dibutuhkan. Tampilan Form menu utama Sistem Informasi Rekam Medis seperti berikut :



Gambar 4.1 Menu Utama

Pada saat aplikasi pertama dijalankan, akan muncul form login yang menentukan hak akses atau wewenang terhadap menu aplikasi yang dapat dijalankan oleh user tentunya setelah memasukkan user dan password yang dibuat dan disetujui administrator. Bentuk form login dapat dilihat pada gambar 4.2.

Gambar 4.2 Form Login

Berikut merupakan penjelasan bagian-bagian implementasi sistem yang terbagi menjadi beberapa form aplikasi, yaitu :

4.1.1. Menu Sistem

Pada menu ini terdapat sub-menu Login jika user ingin masuk dan mengakses aplikasi, Logout bagi user untuk keluar dari aplikasi. Serta Penggantian password bagi semua user jika user ingin mengganti password untuk login.



Gambar.4.3 Ganti password user

4.1.2. Menu Setting

Menu ini hanya dapat diakses oleh user yang memiliki hak akses sebagai administrator. Sub-sub menu pada menu setting antara lain :

4.1.2.1 Manajemen User

Sub menu ini berfungsi sebagai tempat manajemen data user yaitu tempat pendaftaran user baru, dan menampilkan nama user yang boleh mengakses aplikasi Sistem Informasi Rekam Medis. Adapun tampilan sub-menu ini seperti gambar 4.4.

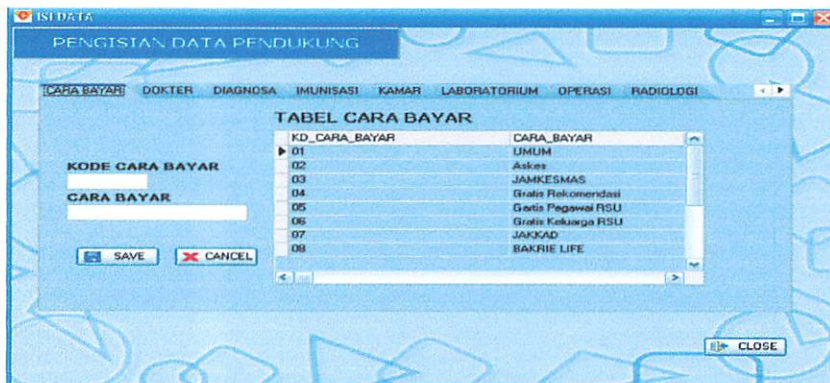


Gambar 4.4 Manajemen User

Seorang admin juga dapat mengganti user atau mengedit dengan menekan tombol edit ataupun dapat men-double klik kolom pada grid data user yang ingin diedit.

4.1.2.2 Isi Data

Sub menu ini digunakan untuk pengisian data-data pendukung pada sistem informasi seperti asuransi atau cara bayar pasien, nama dokter yang ada pada rumah sakit, jenis diagnosa penyakit, sampai pada data kamar jika terdapat penambahan kamar di Rumah sakit.



Gambar 4.5 Isi Data

4.1.2.3 Manajemen Data Pasien

Sub menu ini fungsinya untuk menghapus data pasien jika inputan data sosial pasien salah. Pada form ini terdapat pencarian pasien yang akan dihapus dan kategori pencariannya berdasarkan nama atau no rekam medis pasien. Tampilan form hapus data pasien adalah sebagai berikut:

NORM	NAMA	TL	TTL	PEN
00000002	SODIKIN	BIMA	30/12/1999	SMF
00000003	Rizze	surabaya	30/12/1999	hjk
00000004	RAKIMAN	BLITAR	15/10/1975	S1

Gambar 4.6 Hapus Data Pasien

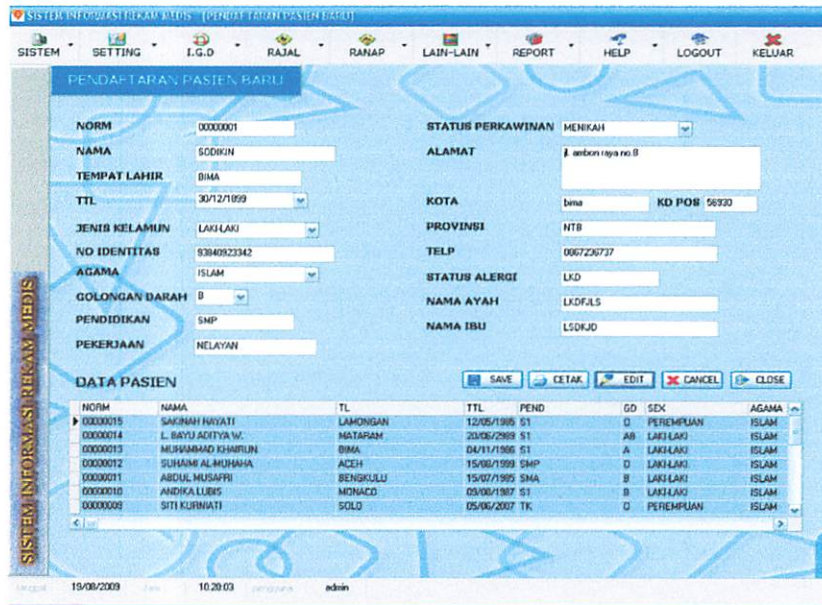
4.1.3 Menu Instalasi Gawat Darurat(I.G.D)

Pada menu Instalasi Gawat Darurat, sub-sub menu yang ada antara lain:

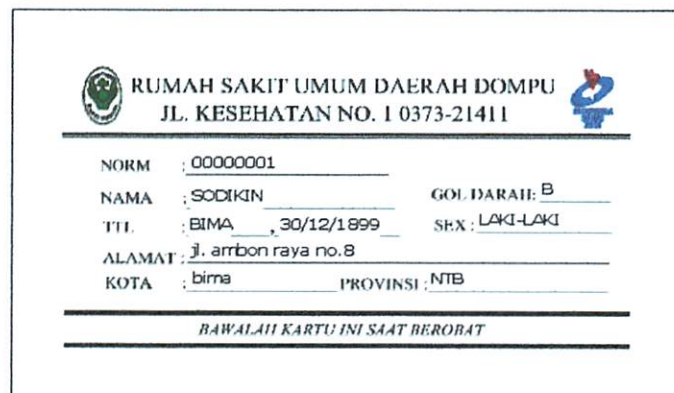
4.1.3.1 Pendaftaran Pasien Baru

Sub-menu ini digunakan untuk memasukkan data pasien yang belum memiliki Kartu berobat atau baru pertama kali dirawat di rumah

sakit sehingga harus didaftarkan dan mendapatkan kartu berobat. Setelah pasien berhasil didaftarkan, No rekam medis yang dimiliki pasien bersifat selamanya dan dapat digunakan untuk mendaftarkan diri pada masing-masing bangsal Rumah sakit saat pasien dirawat kembali.



Gambar 4.7 Pendaftaran Pasien Baru



Gambar 4.8 Pencetakan kartu Berobat pasien

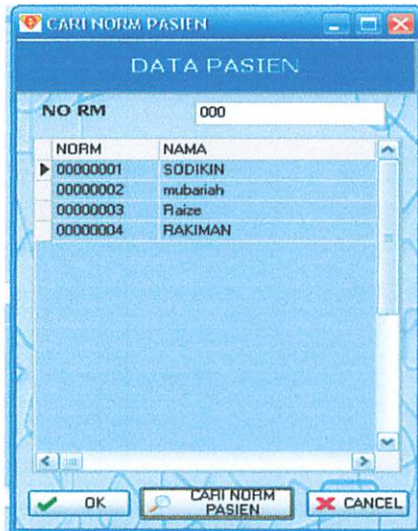
4.1.3.2 Daftar I.G.D

Sub-menu Daftar I.G.D adalah form tempat pendaftaran Pasien yang telah memiliki kartu berobat atau sudah pernah dirawat di rumah sakit yang mendapat atau masuk pada bangsal Instalasi Gawat Darurat. Data yang ditampilkan pada tabel data pasien adalah data pendaftaran pasien menurut tanggal pendaftaran. Akan tetapi user dapat memfilter tabel dengan melihat semua isi data pendaftaran dengan memilih tombol radio pada form.

The screenshot shows a web-based form for patient registration in an Emergency Room (IGD). The form is titled 'PENDAFTARAN PASIEN I.G.D' and is part of a 'SISTEM INFORMASI REKAM MEDIS'. It features several input fields: 'TANGGAL MASUK' (date), 'NOREG IGD' (number '02/09/09-001'), 'NORM' (number), 'CARA MASUK' (method), and 'CARA BAYAR' (payment type). There is also a 'DATA PASIEN' section with fields for 'NORM', 'NAMA', 'SEX', 'GD', 'ALAMAT', 'TELP', and 'NOKTP'. A 'FILTER DATA' section contains two radio buttons: 'SEMUA' (selected) and 'PERTANGGAL'. Below the form is a table with the following columns: 'TGL DAFTAR', 'NOREG IGD', 'NORM', 'NAMA', 'CARA MASUK', and 'KD CARA BAYAR'. The table is currently empty. At the bottom of the form, there are four buttons: 'DAFTAR', 'PASIEN BARU', 'CANCEL', and 'CLOSE'.

Gambar 4.9 Pendaftaran Pasien I.G.D

Pada form pendaftaran pasien I.G.D, terdapat form bantuan untuk pencarian NORM pasien, cara bayar pasien yang menampilkan kode cara pembayaran pasien serta jenis pembayaran. Dan jika terdapat kesalahan pada saat penginputan data oleh petugas, petugas dapat langsung mengedit pendaftaran pasien dengan menklik dua kali kolom pada tabel yang akan di edit.



Gambar 4.10 Pencarian NORM pasien



Gambar 4.11 Pencarian Cara Bayar

4.1.3.3 Pemeriksaan Pasien I.G.D

Sub-menu ini digunakan untuk menyimpan hasil pemeriksaan pasien, dari diagnosa penyakit hingga dokter yang memeriksa dan memberikan hasil diagnosa. No pemeriksaan pada sub-menu ini

bertambah otomatis sesuai dengan noregister I.G.D pasien dikarenakan pemeriksaan pasien dapat dilakukan beberapa kali sesuai dengan penyakit yang diderita pasien.

PEMERIKSAAN PASIEN RAWAT IGD

TANGGAL PERIKSA: 13/07/2009
 NO PEMERIKSAAN: 13/07/09-002-01
 NOREG I.G.D: 13/07/09-002
 DOKTER: 07
 KODE DIAGNOSA: E10.1
 KEADAAN KELUAR: 01

DATA DOKTER
 ID DOKTER: 07
 NAMA DOKTER: dr. H. Kusni Nuarso
 SPESIALIST:

TABEL DIAGNOSA PASIEN

TGL_PERIKSA	NO_PEMERIKSAAN	NOREG_IGD	INDRM	NAMA	ID_DOKTER	NAMA_DOKTER
13/07/2009	13/07/09-002-01	13/07/09-002	00000001	SODIKIN	07	dr. I Ketut Nuarso
04/08/2009	13/07/09-002-02	13/07/09-002	00000001	SODIKIN	03	dr. H. Dasa Indrako, N

Buttons: SAVE, CANCEL, CLOSE

Status bar: 13/08/2009 12:27:58 admin

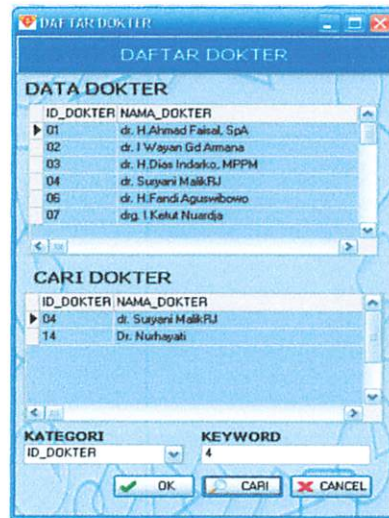
Gambar 4.12 Pemeriksaan Pasien I.G.D

Terdapat juga form bantuan yang digunakan untuk mencari no register I.G.D, kode dokter, dan jenis diagnosa penyakit pasien. Jika tombol OK pada masing-masing form diklik, maka edit pada form Pemeriksaan I.G.D akan terisi dengan kode dari masing-masing form tambahan. Tampilan dari form tambahan tersebut antara lain :



Gambar 4.13 Pencarian Register I.G.D

Pada form ini, pencarian register pasien I.G.D dicari berdasarkan tanggal pendaftaran pasien.



Gambar 4.14 Pencarian Data Dokter

Form ini dilengkapi dengan fasilitas pencarian dokter menurut kategori id dokter ataupun nama dokter.



Gambar 4.15 Pencarian Diagnosa Penyakit

Pada form ini, kode diagnosa dan jenis diagnosa merupakan kode yang telah digunakan di semua instansi kesehatan diseluruh dunia karena jenis diagnosa ini diambil dari buku International Classification of Disease(I.C.D), sehingga perawat saat menginputkan data tidak kesulitan dengan kode yang begitu banyak.

Dan jika terjadi kekeliruan pada saat input data pemeriksaan pasien, perawat dapat mengedit data pemeriksaan dengan mengklik dua kali pada kolom tabel diagnosa pasien yang akan diedit.

4.1.3.4 Tindakan I.G.D

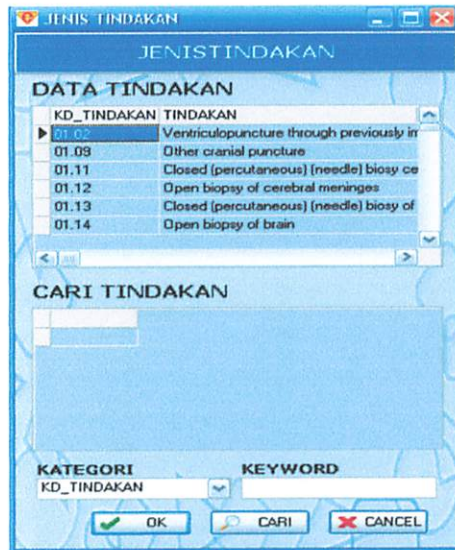
Sub-menu ini digunakan user untuk menginputkan data-data tindakan medis yang dilakukan terhadap pasien yang dirawat pada bangsal I.G.D. Pada form ini terdapat form bantuan yang digunakan untuk mencari jenis tindakan medis yang dilakukan terhadap pasien.

Jika terjadi kesalahan penginputan data tindakan, perawat juga dapat mengubah data dengan cara mengklik dua kali kolom pada tabel tindakan pasien.

No tindakan pun otomatis dengan no register I.G.D sebagai acuannya karena satu no register yang dirawat dapat diberikan tindakan lebih dari satu kali. Sama seperti pada form-form lain, jika terdapat kesalahan pada input data tindakan medis pasien, data yang sudah tersimpan dapat diedit dengan mengklik dua kali pada kolom tabel tindakan pasien yang ingin diubah.

TGL TINDAKAN	NO TINDAKAN	NOREG IGD	NORM	NAMA
11/08/2009	13/07/09-002-01	13/07/09-002	00000001	SODIKIN

Gambar 4.16 Tindakan I.G.D



Gambar 4.17 Pencarian Tindakan Pasien I.G.D

4.1.3.5 Rujukan Pasien

Sub-menu Rujukan I.G.D digunakan jika ada pasien I.G.D yang akan dirujuk ke bangsal Rawat Inap atau ke Rumah Sakit lain. Jika pasien yang dirawat akan dirujuk atau dipindahkan ke bangsal rawat inap, maka ketika tombol rujuk diklik, pendaftaran rawat inap akan langsung muncul dan petugas dapat langsung mendaftarkan pasien pada bangsal rawat inap beserta memilih kamar inap.

Gambar 4.18 Rujukan Pasien I.G.D

Pada form ini juga terdapat form bantuan yang sama seperti form bantuan pada menu pemeriksaan pasien IGD dan menu tindakan IGD yang digunakan untuk mencari no register IGD menurut tanggal pendaftaran.

4.1.4 Menu Rawat Jalan(RAJAL)

Menu ini digunakan oleh user yaitu petugas kesehatan yang bertugas pada bangsal rawat jalan. Jadi setiap pasien yang dirawat pada bangsal ini akan dilakukan pendataan sebagaimana yang dilakukan pada bangsal Instalasi Gawat Darurat. Sub-sub menu pada menu Rawat Jalan ini antara lain :

4.1.4.1 Pendaftaran Pasien Baru

Sub menu ini digunakan untuk pendaftaran pasien yang belum memiliki kartu berobat atau pasien baru pertama kali dirawat pada bangsal rawat jalan.

UIN FOKUSI PEKAM MEDIS - [PENDAFTARAN PASIEN BARU]

SETTING I.G.D RAJAL RANAP LAIN-LAIN REPORT HELP LOGOUT KELUAR

PENDAFTARAN PASIEN BARU

NORM: 0000004
 NAMA: RAKIMAN
 TEMPAT LAHIR: BUTAR
 TTL: 15/10/1975
 JENIS KELAMUN: LAKI-LAKI
 NO IDENTITAS: 072839126293
 AGAMA: BUDHA
 GOLONGAN DARAH: AB
 PENDIDIKAN: S1
 PEKERJAAN: GURU

STATUS PERKAWINAN: MENIKAH
 ALAMAT: JL. SUKUDJFKLLSKJF
 KOTA: MALANG KD POS: 62312
 PROVINSI: LSKDJKA
 TELP: LSKDJFL
 STATUS ALERGI: CSK
 NAMA_AYAH: LFDKSSD
 NAMA_IBU: SFDLSD

DATA PASIEN

SAVE CETAK EDIT CANCEL CLOSE

NORM	NAMA	TL	PEND	GD	SEX	AGAMA
0000001	SODIHN	BIMA	30/12/1999 SMP	B	LAKI-LAKI	ISLAM
0000002	andriah	malang	30/12/1999 SMP	A	PEREMPUAN	ISLAM
0000003	Roice	surabaya	30/12/1999 kjk	B	PEREMPUAN	KRISTEN
0000004	RAKIMAN	BUTAR	15/10/1975 S1	AB	LAKI-LAKI	BUDHA

Gambar 4.19 Pendaftaran pasien baru

Setelah data pasien berhasil disimpan, maka akan muncul secara otomatis report yang menampilkan kartu berobat pasien yang dapat diprint, dan kartu tersebut dapat digunakan jika pasien kembali dirawat pada masing-masing bangsal Rumah Sakit.

4.1.4.2 Pendaftaran Pasien Rawat Jalan

Sub menu ini digunakan pada saat pasien mendaftar pada bangsal Rawat Jalan serta pasien sudah memiliki kartu berobat. Hampir sama dengan register I.G.D, pada pendaftaran Rawat Jalan memiliki alur yang sama, Perbedaan hanya terdapat pada Poliklinik. Jika pada bangsal I.G.D tidak terdapat pilihan poliklinik yang akan dikunjungi

karena I.G.D memiliki bangsal khusus untuk perawatan gawat darurat sendiri, sedangkan pada rawat jalan terdapat beberapa poliklinik yang dapat dikunjungi seperti poli gigi, anak, dan lain-lain. Terdapat filter tampilan tabel register pasien rawat jalan pertanggal atau menampilkan semua *record* yang ada pada database.

The screenshot shows a software interface for patient registration. At the top, there is a menu bar with options: EM, SETTING, I.G.D, RAJAL, RANAP, LAIN-LAIN, REPORT, HELP, and LOGOU. The main window title is 'PENDAF TARAN PASIEN RAWAT JALAN'. Below the title, there are several input fields: 'TANGGAL DAFTAR' (02/08/2009), 'NOREG RAJAL' (02/08/09-001), 'NORM', 'CARA MASUK', 'POLIKLINIK', and 'CARA BAYAR'. To the right, under 'DATA PASIEN:', there are fields for 'NORM', 'NAMA', 'SEX', 'GD', 'ALAMAT', 'TELP', and 'NOKTP'. Below these fields, there is a 'FILTER DATA:' section with radio buttons for 'SEMUA' and 'PERTANGGAL'. At the bottom, there is a table titled 'TABEL DATA PASIEN RAWAT JALAN' with columns: 'TGL DAFTAR', 'NOREG RAJAL', 'NORM', 'NAMA', and 'CARA MASUK'. The table is currently empty. At the bottom right, there are buttons for 'DAFTAR', 'PASIEN BARU', 'CANCEL', and 'CLOSE'.

Gambar 4.20 Registrasi Pasien Rawat Jalan

Pada menu ini Terdapat form bantuan yang sama dengan form bantuan pada menu registrasi IGD yang digunakan untuk mencari NORM pasien, jika petugas pendaftaran kesulitan mencari No Rekam Medis pasien. Form bantuan poliklinik dan cara bayar pasien yang digunakan untuk memilih kode poliklinik yang akan dituju dan cara pembayaran pasien.

4.1.4.3 Pemeriksaan Pasien Rawat Jalan

Sub menu ini digunakan sebagai antarmuka untuk menyimpan data-data pemeriksaan pasien bangsal rawat jalan. Baik itu dokter yang memeriksa, sampai jenis diagnosa pasien.

TGL PERIKSA	NO PEMERIKSAAN	NOREG RAJAL	NORM	NAMA	ID DOKTER	NAMA DOKTER
09/07/2009	09/07/09-001-03	09/07/09-001	00000004	RAKIMAN	06	dr. H.Fandi Aguswibowo

Gambar 4.21 Pemeriksaan Pasien Rawat Jalan

No pemeriksaan pada sub-menu ini bertambah otomatis sesuai dengan noregister Rawat Jalan pasien dikarenakan pemeriksaan pasien dapat dilakukan beberapa kali sesuai dengan penyakit yang diderita pasien.

Terdapat juga form bantuan yang digunakan untuk mencari no register Rawat Jalan, Dokter, dan jenis diagnosa pasien. Form ini digunakan untuk mencari kode atau jenis dari penyakit pasien, noregistrasi dan kode dokter yang memeriksa. Tampilan dari form-form tersebut sama dengan form pada submenu pemeriksaan pasien I.G.D.

4.1.4.4 Tindakan Pasien Rawat Jalan

Sub-menu ini digunakan user untuk menginputkan data-data tindakan medis yang dilakukan terhadap pasien yang dirawat pada bangsal Rawat Jalan. pada form ini terdapat pula tombol pilihan jenis tindakan yang akan menampilkan kode dan jenis tindakan medis yang akan dilakukan terhadap pasien sesuai dengan hasil pemeriksaan penyakit pasien.

No tindakan pun otomatis dengan no register Rawat Jalan sebagai acuannya karena satu no register yang dirawat dapat diberikan tindakan lebih dari satu kali. Sama seperti pada form-form lain, jika terdapat kesalahan pada input data tindakan pasien, data yang sudah tersimpan dapat diedit dengan mendouble klik kolom tabel tindakan pasien yang ingin diubah.

TGL TINDAKAN	NO TINDAKAN	NOREG RAJAL	NORM	NAMA
--------------	-------------	-------------	------	------

Gambar 4.22 Tindakan Medis Pasien Rawat Jalan

4.1.4.5 Rujukan Pasien Rawat Jalan

Sub-menu Rujukan Rawat Jalan digunakan jika ada pasien Rawat Jalan yang akan dirujuk ke bangsal Rawat Inap atau ke Rumah Sakit lain. Jika pasien yang dirawat akan dirujuk atau dipindahkan ke bangsal rawat inap, maka menu pendaftaran pasien rawat inap akan langsung muncul dan petugas dapat langsung mendaftarkan pasien pada bangsal rawat inap beserta memilih kamar inap.

TGL TINDAKAN	NO TINDAKAN	NOREG RAJAL	NORM	NAMA
03/08/2009	13/07/09-002-01	13/07/09-002	00000003	Raise
03/08/2009	13/07/09-002-02	13/07/09-002	00000003	Raise

Gambar 4.23 Rujukan Pasien Rawat Jalan

4.1.5 Menu Pasien Rawat Inap

Menu ini digunakan oleh petugas kesehatan yang bertugas pada bangsal rawat inap, jadi setiap pasien yang dirawat pada bangsal ini akan dilakukan pendataan sebagaimana yang dilakukan pada bangsal-bangsal lain. Yang berbeda adalah pada bangsal rawat inap terdapat proses pindah

kamar, serta pencarian pasien berdasarkan kamar pasien. Sub-sub menu pada menu Rawat Jalan ini antara lain :

4.1.5.1 Pendaftaran Pasien Baru

Sub menu ini digunakan jika pasien yang akan dirawat pada bangsal rawat inap belum memiliki kartu berobat atau baru pertama kali dirawat di rumah sakit.

INFORMASITRUKAM.MEDIS : [PENDAFTARAN PASIEN BARU]

SETTING I.G.D RAJAL RANAP LAIN-LAIN REPORT HELP LOGOUT KELUAR

PENDAFTARAN PASIEN BARU

NORM: 00000004
 STATUS PERKAWINAN: MENIKAH
 NAMA: RAKIMAN
 ALAMAT: JL. SUDJIDIKLLSKIF
 TEMPAT LAHIR: BUTAR
 TTL: 15/10/1975
 KOTA: MALANG KD POS: 62012
 JENIS KELAMIN: LAKI-LAKI
 PROVINSI: LSFDUKA
 NO IDENTITAS: 072839128293
 TELP: LSKDJFL
 AGAMA: BUDHA
 STATUS ALERGI: CSK
 GOLONGAN DARAH: AB
 NAMA_AYAH: LFDKSSD
 PENDIDIKAN: S1
 NAMA_IBU: SFDLSD
 PEKERJAAN: GURU

SAVE CETAK EDIT CANCEL CLOSE

NORM	NAMA	TL	TTL	PEND	GD	SEX	AGAMA
00000001	SODIKUN	BIMA	30/12/1899	SMP	B	LAKI-LAKI	ISLAM
00000002	muhammad	malaram	30/12/1899	SMP	A	PEREMPUAN	ISLAM
00000003	Raino	surabaya	30/12/1899	KJK	B	PEREMPUAN	KRISTEN
00000004	RAKIMAN	BUTAR	15/10/1975	S1	AB	LAKI-LAKI	BUDHA

Gambar 4.24 Pendaftaran Pasien Baru

Sama seperti sub-menu pendaftaran pasien baru pada bangsal lain, setelah pasien berhasil didaftarkan, maka akan muncul report kartu berobat yang dapat di cetak sebagai kartu pasien.

4.1.5.2 Pendaftaran Pasien Rawat Inap

Sub-menu ini digunakan untuk menyimpan data registrasi pasien rawat inap yang sudah memiliki kartu berobat. Pada pendaftaran pasien rawat inap juga terdapat sub menu pemesanan kamar bagi pasien. jadi setelah pasien masuk kamar perawatan, bed pada kamar akan berkurang secara otomatis.

The screenshot shows a web-based interface for inpatient registration. At the top, there is a navigation menu with options like 'SETTING', 'I.G.D', 'RAJAL', 'RANAP', 'LAIN-LAIN', 'REPORT', 'HELP', 'LOGOUT', and 'KELUAR'. The main title is 'PENDAFTARAN PASIEN RAWAT INAP'. The form is divided into several sections:

- TANGGAL DAFTAR:** 02/08/09
- NOREG RANAP:** 30/12/99 001
- NORM:** [input field]
- CARA BAYAR:** [input field]
- CARA PENERIMAAN:** [input field]
- CARA MASUK:** [input field]
- DATA KAMAR:** NO KAMAR [input field], KELAS [input field]. A button 'LIHAT KAMAR KOSONG' is present.
- PENANGGUNG JAWAB PASIEN:** NAMA KERABAT [input field], ALAMAT KERABAT [input field], NO TELEPHONE [input field], NO IDENTITAS [input field].
- DATA PASIEN:** NORM [input field], NAMA [input field], SEX [input field], GD [input field], ALAMAT [input field], TELP [input field], NO_KTP [input field].
- FILTER DATA:** Radio buttons for 'SEPIJA' and 'BELUM PULANG'.
- DATA PASIEN RANAP:** A table with columns: TGL DAFTAR, NOREG RANAP, NORM, NAMA.

At the bottom right, there are buttons: 'DAFTAR', 'PASIEN BARU', 'CANCEL', and 'CLOSE'.

Gambar 4.25 Pendaftaran Pasien Rawat Inap

Pada perawatan pasien rawat inap, terdapat inputan terhadap penanggung jawab pasien yang dirawat sebagai penjamin jika terjadi sesuatu terhadap pasien yang dirawat seperti pasien melarikan diri atau lain sebagainya.

Pada sub-menu ini terdapat tombol pencarian kamar yang jika diklik akan muncul form pencarian kamar kosong pasien.



Gambar 4.26 Pencarian Kamar Kosong

4.1.5.3 Pemeriksaan Pasien Rawat Inap

Pada sub-menu ini terdapat beberapa jenis pemeriksaan pasien antara lain diagnosa utama untuk data diagnosa pasien atau penyakit utama pasien, diagnosa komplikasi jika ada komplikasi penyakit yang diderita pasien, morfologi dan neoplasma jika ada penyakit pasien yang berhubungan dengan pemeriksaan morfologi dan neoplasma, Penyebab luar dan cedera jika pada pemeriksaan terdapat sebab penyakit karena ada cedera dan juga terdapat pemeriksaan imunisasi pasien jika terdapat imunisasi penyakit pasien.

The screenshot shows a web-based medical information system interface. At the top, there is a menu bar with icons and labels: 'SETTING', 'I.G.D', 'RAJAL', 'RANAP', 'LAIN-LAIN', 'REPORT', 'HELP', 'LOGOUT', and 'KELUAR'. Below the menu bar, the main title is 'Pemeriksaan Pasien Rawat Inap'. The form is divided into several sections:

- DATE AND ID:** 'TANGGAL PEMERIKSAAN' (02/08/09) and 'NO PEMERIKSAAN' (input field).
- REGISTRATION:** 'NREG RANAP' (input field).
- DOCTOR INFORMATION:** 'DOKTER' (input field), 'DATA DOKTER' (ID DOKTER, NAMA DOKTER, SPESIALIST) (input fields).
- DIAGNOSIS:** 'KODE DIAGNOSA' (input field), 'DIAGNOSA' (input field).
- DIAGNOSIS TABLE:** 'TABEL DIAGNOSA PASIEN' with columns: 'TGL PERIKSA', 'NO PEMERIKSAAN', 'NREG RANAP', 'NORM', 'NAMA'. The table is currently empty.
- Buttons:** 'SAVE', 'CANCEL', 'KOSONGKAN', and 'CLOSE'.

Gambar 4.27 Pemeriksaan Pasien Rawat Inap

Pada sub-menu ini terdapat form bantuan yang sama dengan form pada menu pemeriksaan pasien IGD dan pemeriksaan pasien rawat jalan yang digunakan untuk mencari kode dan nama dokter yang memeriksa, dan form bantuan untuk mencari kode dan jenis diagnosa penyakit pasien.

4.1.5.4 Tindakan Pasien

Sub-menu ini digunakan untuk menyimpan data-data tindakan medis pasien yang dilakukan berdasarkan penyakit pasien. Tidak jauh beda dengan sub-menu tindakan pada bangsal-bangsal lain.

The screenshot shows a software window titled "TINDAKAN PASIEN RAWAT INAP". The window contains the following fields and a table:

- TANGGAL TINDAKAN:** 20/08/2009
- NO TINDAKAN:** 19/08/09-002-02
- NOREG RANAP:** 19/08/09-002
- KODE TINDAKAN:** (empty)
- RESEP OBAT:** (empty)
- JENIS TINDAKAN:** (empty)
- KD_TINDAKAN:** (empty)
- JENIS TINDAKAN:** (empty)

Below the fields is a table titled "DAFTAR TINDAKAN PASIEN RAWAT INAP":

TGL TINDAKAN	NO TINDAKAN	NOREG RANAP	NORM	NAMA
▶ 20/08/2009	19/08/09-002-01	19/08/09-002	00000012	SUHAIMI AL-MUHAHA

At the bottom of the window are three buttons: "SAVE", "CANCEL", and "CLOSE".

Gambar 4.28 Tindakan Pasien Rawat Inap

4.1.5.5 Pindah Kamar Pasien

Sub-menu ini digunakan untuk menyimpan data-data pasien yang melakukan pindah kamar. Sub menu ini dibutuhkan agar pasien yang pindah kamar dapat tersimpan dan dilihat *historynya*, sehingga dapat diketahui berapa lama pasien berada pada kamar tersebut.



Gambar 4.29 Pindah Kamar Pasien

4.1.5.6 Pencarian Kamar Pasien

Sub-menu pencarian kamar digunakan untuk mencari letak kamar pasien dirawat. Biasanya ini dibutuhkan saat pengunjung mencari ruangan pasien.



Gambar 4.30 Pencarian Kamar Pasien

4.1.5.7 Rujukan Ranap

Sub-menu ini digunakan jika pasien rawat inap harus dirujuk ke rumah sakit lain karena suatu sebab. Pada sub-menu ini hampir sama dengan sub-menu pada bangsal lainnya.

The screenshot shows a software window titled "RUJUKAN PASIEN RAWAT INAP". Inside, there's a header "RUJUKAN RAWAT INAP". Below it are several input fields: "TGL RUJUK" with the value "20/08/2009", "NO RUJUKAN" with "20/08/2009-01", "NO REG RANAP" with "20/08/09-001", and "TUJUAN" which is empty. A section titled "TABEL DAFTAR RUJUKAN :" contains a table with the following headers: "TGL RUJUKAN", "KD RUJUKAN", "NOREG RANAP", "NORM", and "NAMA". The table body is currently empty. At the bottom of the window are three buttons: "SAVE", "CANCEL", and "CLOSE".

Gambar 4.31 Rujukan Pasien Rawat Inap

4.1.5.8 Keadaan Keluar Pasien

Keadaan keluar pasien digunakan untuk menyimpan data keadaan pasien saat keluar dari rumah sakit, apakah sudah sembuh, dan faktor yang mempengaruhi seperti radiasi nuklir, transfusi darah atau infeksi nosokomial.

INFORMASI REKAM MEDIS | KEADAAN KELUAR PASIEN RAWAT INAP

M SETTING I.G.D RAJAL RANAP LAIN-LAIN REPORT HELP

KEADAAN KELUAR PASIEN RAWAT INAP

TANGGAL KELUAR: 20/08/2009

NOREG RANAP:

ADA PENYEBAB INFEKSI: YA

INFEKSI NOSOKOMIAL:

PENYEBAB INFEKSI:

RADIO TERAPI ATAU NUKLIR:

TRANSFUSI DARAH:

KEADAAN KELUAR:

CARA KELUAR:

DATA PASIEN RAJAL:

NOREG RANAP:

NORM:

TGL DAFTAR:

CARA MASUK:

NO KAMAR:

TABEL DAFTAR KEADAAN KELUAR

TANGGAL KELUAR	NOREG RANAP	NORM	NAMA	CARA KELUAR
20/08/2009	19/08/09-002	0000012	SUHAIMI AL-MUHAHA	PULANG PAKSA
20/08/2009	19/08/09-001	0000003	Raike	DIGZINKAN KELUAR

SAVE CANCEL CLOSE

Gambar 4.32 Keadaan Keluar Pasien Rawat Inap

4.1.6 Lain-Lain

Pada menu ini terdapat sub-sub seperti pemeriksaan Laboratorium, pemeriksaan radiologi, operasi, dan pencarian pasien.

4.1.6.1 Laboratorium

Sub-menu ini digunakan sebagai antarmuka untuk menyimpan data pemeriksaan laboratorium pasien. Dari jenis pemeriksaan yang dilakukan hingga hasil dari pemeriksaan tersebut. Pada menu laboratorium terdapat form bantuan yang digunakan untuk mencari No Rekam Medis pasien serta jenis pemeriksaan Laboratorium pasien. Tampilan form bantuan dapat dilihat pada gambar 3.34.

EMR INFORMATION REKAM MEDIS - [LABORATORIUM]

IM SETTING I.G.D RAJAL RANAP LAIN-LAIN REPORT HELP LO

PELAYANAN LABORATORIUM PASIEN

TANGGAL PERIKSA: 02/08/2009

NO PEMERIKSAAN: 02/08/09-001

NORM: 00000002

KD PEMERIKSAAN: 05

HASIL: B

HASIL NORMAL: -

KETERANGAN: -

DATA SOSIAL PASIEN

NORM: 00000002

NAMA: mubawati

SEX: PEREMPUAN

GD: 0

AGAMA: ISLAM

TELP: 081

PEKER: perawat

FILTER DATA: SEMUA PERTANGGAL

TABEL LABORATORIUM

TGL PERIKSA	NO PEMERIKSAAN	NORM	NAMA	KD LABORATORIUM	PEMERIKSAAN
02/08/2009	02/08/09-001	00000002	mubawati	05	GOLONGAN D
02/08/2009	02/08/09-002	00000001	SODIKIN	05	GOLONGAN D

SAVE CANCEL CLOSE

Gambar 4.33 Pelayanan Laboratorium Pasien

JENIS PEMERIKSAAN LAB

KODE DAN JENIS PEMERIKSAAN LABORATORIUM

DATA JENIS PEMERIKSAAN LAB

KD_LAB	PEMERIKSAAN
01	DARAH LENGKAP
02	PPT
03	DDR
04	WIDAL
05	GOLONGAN DARAH
06	HBsAG

PENCARIAN PEMERIKSAAN

KD_LAB	PEMERIKSAAN
01	DARAH LENGKAP
02	PPT
03	DDR
04	WIDAL
05	GOLONGAN DARAH

KATEGORI: KD_LAB KEYWORD: _____

OK CARI CANCEL

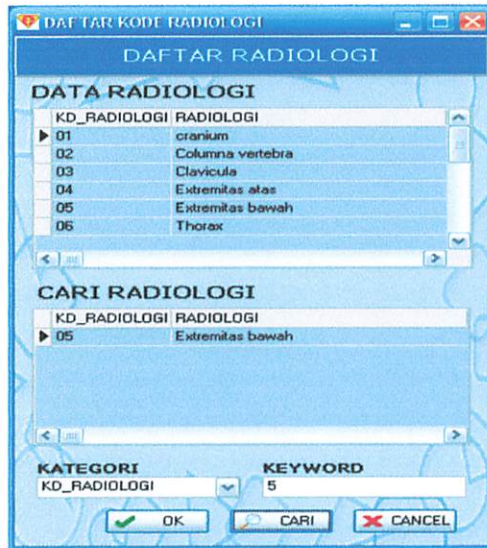
Gambar 4.34 Pencarian pemeriksaan Laboratorium

4.1.6.2 Radiologi Pasien

Sub-menu radiologi pasien digunakan saat pasien melakukan pemeriksaan radiologi(Rontgen). Data-data yang disimpan antara lain jenis radiologi yang dilakukan pasien, serta hasil dari radiologi tersebut. Pada menu ini terdapat form bantuan yang fungsinya untuk mencari No rekam Medis pasien serta Pencarian kode dan jenis Radiologi yang dilakukan pasien. Tampilan form bantuan ini dapat dilihat pada gambar 4.36.

TGL PERIKSA	KD PEMERIKSAAN	NORM	NAMA	KD RADIOLOGI R
-------------	----------------	------	------	----------------

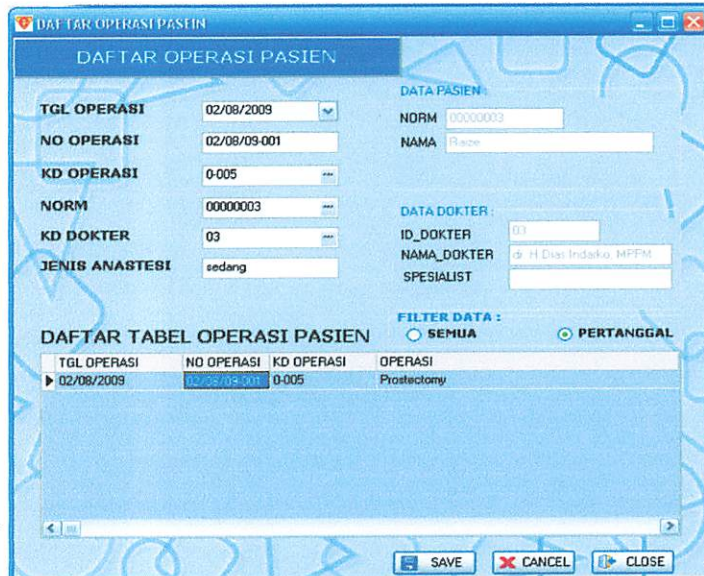
Gambar 4.35 Radiologi Pasien



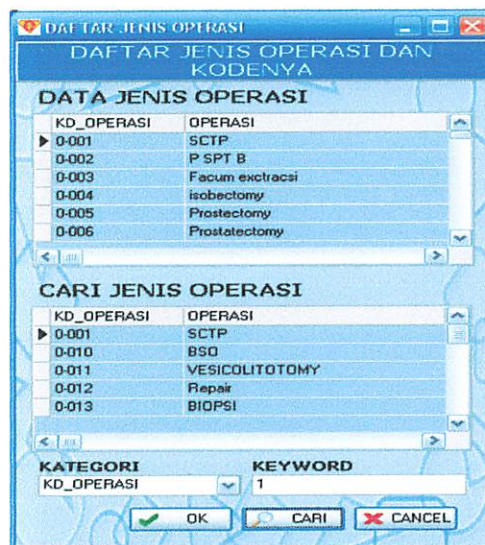
Gambar 4.36 Pencarian Radiologi

4.1.6.3 Operasi Pasien

Sub-menu Operasi Pasien digunakan sebagai antarmuka penyimpanan data-data operasi pasien. Data-data yang disimpan antara lain jenis operasi, dokter yang melakukan operasi, dan jenis anastesi operasi. Pada menu operasi, terdapat form bantuan antara lain form pencarian No rekam medis pasien, kode dan jenis operasi yang digunakan untuk mencari kode dan jenis operasi pasien, serta form untuk mencari kode dan nama dokter yang memimpin operasi.



Gambar 4.37 Operasi Pasien



Gambar 4.38 Pencarian Operasi Pasien

4.1.6.4 Pencarian Pasien

Sub menu ini digunakan untuk pencarian pasien jika ada pasien yang mungkin kehilangan kartu berobat sehingga harus

dicari namanya apakah sudah terdaftar dan dapat dicetak ulang kartu berobatnya.



Gambar 4.39 Pencarian Pasien

4.1.7 Laporan

Menu laporan terdiri dari sub menu laporan yang berisi report-report hasil dari penyimpanan data dari semua aktivitas medis yang dilakukan pada Rumah Sakit dan berbentuk statistik. pada menu ini bentuk formnya adalah pilihan, jadi user dapat memilih report apa yang ingin dilihat ataupun dicetak. Terdapat juga pilihan preview report, berdasarkan tanggal dan berdasarkan hari ini.



Gambar 4.40 Laporan utama

Jenis laporan yang dapat dilihat antara lain register pasien, Penyakit Pasien, index dokter, tindakan medis, laboratorium, radiologi dan operasi pada masing-masing bangsal.

4.1.7.1 Register pasien I.G.D

Laporan ini berisi data pendaftaran pasien I.G.D yang mendaftar pada bangsal I.G.D menurut kategori pemfilteran laporan.

RUMAH SAKIT UMUM DAERAH DOMPU
 JL. KESEHATAN NO. 1 DOMPU
 TELP 0373-21411

LAPORAN INDEX REGISTRASI PASIEN I.G.D

NO	TGL CAPTURE	NORES.PENAL	NCFRM	NPAHS	CWA.MANUL	CWA.BAWA
1	3/10/2009	3/10/2009-001	00000000	RACTANAN		Ganti Pakaian ISU

TOL. QCTAH : 03092209

Gambar 4.41 Laporan Register I.G.D

4.1.7.2 Laporan Penyakit Pasien I.G.D

Laporan ini berisi hasil diagnosa penyakit pasien pada bangsal Instalasi Gawat Darurat. Laporan ini dikelompokkan berdasarkan penyakit yang diderita pasien dengan tujuan agar dapat dibuat statistik penyakit yang terjadi pertanggal sehingga nantinya dapat dilakukan pencegahan-pencegahan melalui penyuluhan ataupun upaya-upaya medis lain.

RUMAH SAKIT UMUM DAERAH DOMPU
JL. KESEHATAN NO. 1 DOMPU
TELP 0373-21411

LAPORAN PENYAKIT PASIEN I.G.D

NO	TGL.PENYAKIT	NOMOR I.G.D	NORH	NAMA	DOKTOR/PERUSAHA	RE_DIAGNOSA	DIAGNOSA
TANGGAL LAPORAN : 01/07/2009 -- 03/09/2009							
1	05/07/2009	0507/09/001	00000002	mutariyah	dr. H.Paniti Aguzelkawa	V01.9	Fractation injured in collision with pedal cycle, unspecified whether traffic or nontraffic
TANGGAL LAPORAN : 01/07/2009 -- 03/09/2009							
2	13/07/2009	1307/09/001	00000003	Rizzo	dr. H.Dian Indarito, M.P.P.M	I01	Flasang kelopak mata lainnya
TANGGAL LAPORAN : 01/07/2009 -- 03/09/2009							
3	13/07/2009	1307/09/002	00000001	SUDIPIN	dr. I Ketut Huanjita	E10.1	Diabetes mellitus dependen insulin dengan ketoasidosis

Mengantar,


User

TOL. CESTAK : 03092009

Gambar 4.42 Laporan Penyakit Pasien I.G.D

4.1.7.3 Laporan Dokter Pemeriksaan

Laporan ini berisi data-data dokter yang memeriksa pasien bangsal I.G.D. jadi laporan ini dikelompokkan berdasarkan nama dokter yang memeriksa.


RUMAH SAKIT UMUM DAERAH DOMPU
 JL. KESEHATAN NO. 1 DOMPU
 TELP 0373-21411

LAPORAN INDEX DOKTER PASIEN I.G.D

NO	TGL. PEMERIKSAAN	NORIS I.G.D	NORIM	NAMA	DOKTER PENJISA	ID_DIAGNOSA	DIAGNOSA
TANGGAL LAPORAN : 01/07/2009 — 03/09/2009							
1	03/07/2009	130709001	0000002	rubanih	dr. H Fandi Aguswasono	V01.9	Postmenstrual irregular in collection with post cycle, unspecified/ekster
TANGGAL LAPORAN : 01/07/2009 — 03/09/2009							
2	13/07/2009	130709001	0000000	Rahza	dr. H Das Indana, MPPM	Y01	Ridang kelopak mata lainnya
TANGGAL LAPORAN : 01/07/2009 — 03/09/2009							
3	13/07/2009	130709002	0000001	SCORIN	dr. I Keldi Nurjaja	EKG.1	Diabetes mellitus disordinasi insulin dengan ketoasidosis

Mengantar,

user

TGL. CETAK : 03/09/2009

Gambar 4.43 Laporan Dokter Pasien I.G.D

4.1.7.4 Laporan Tindakan I.G.D

Laporan ini menampilkan data-data tindakan medis yang dilakukan terhadap pasien pada bangsal I.G.D.


RUMAH SAKIT UMUM DAERAH DOMPU
 JL. KESEHATAN NO. 1 DOMPU
 TELP 0373-21411

LAPORAN TINDAKAN PASIEN RAWAT JALAN

NO	TGL. TINDAKAN	NO TINDAKAN	NORIS I.G.D	NORIM	NAMA	ID TINDAKAN	JENIS TINDAKAN
TANGGAL LAPORAN : 01/07/2009 — 03/09/2009							
1	03/07/2009	030709001	030709001	0000	rubar	01.14	Open biopsy of brain

Mengantar,

user

TGL. CETAK : 03/09/2009

Gambar 4.44 Laporan Tindakan medis Pasien I.G.D

4.1.7.5 Laporan Register Pasien Rawat Jalan

Laporan ini menampilkan register pasien yang dirawat pada bangsal rawat jalan pertanggal rawat. Ini dijadikan statistik seberapa banyak pasien yang dirawat pada rumah sakit khususnya pada bangsal rawat Jalan.

RUMAH SAKIT UMUM DAERAH COMPU
 JL. KESEHATAN NO. 1 COMPU
 TELP 0373-21411

LAPORAN INDEX REGISTRASI PASIEN RAWAT JALAN

NO	TGLDFTWR	NREG RAWAL	NCRH	NAMA	CARA MASUK	CARA BIKER	POLIKLINIK
TANGGAL LAPORAN : 01/07/2009 --- 03/07/2009							
1	09/07/2009	09/07/09-001	00000004	RAGZIMNY	kejur polri	LALM	aruk
2	13/07/2009	13/07/09-001	00000002	rn.banah	datang sendiri	LALM	0601
3	13/07/2009	13/07/09-002	00000003	Raze	kejur polri	LALM	DALAM

Menghitung,


User

TOL OTAK : 03/07/2009

Gambar 4.45 Laporan Register Pasien Rawat Jalan

4.1.7.6 Laporan Penyakit Pasien Rawat Jalan

Laporan ini berisi hasil diagnosa penyakit pasien pada bangsal Rawat Jalan. Laporan ini dikelompokkan berdasarkan penyakit yang diderita pasien dengan tujuan agar dapat dibuat statistik penyakit yang terjadi pertanggal sehingga nantinya dapat dilakukan pencegahan-pencegahan.


RUMAH SAKIT UMUM DAERAH DOMPU
 JL. KESEHATAN NO. 1 DOMPU
 TELP 0373-21411

LAPORAN INDEX PENYAKIT PASIEN RAWAT JALAN

NO	TGL. MEMERIKSA	NOREG RAWAT	NORM	NAMA	DOKTHER PERIKSA	KD. DIAGNOSA	DIAGNOSA
TANGGAL LAPORAN : 01/07/2009 — 03/09/2009 PENYAKIT : Superficial injuries involving multiple regions of upper limb(s)							
1	03/07/2009	030703001	0000004	RAHMATI	dr. H.Fandi Agusubowo	T02.2	Superficial injuries involving multiple regions of upper limb(s)
JUMLAH PASIEN PENDERTA : 1							

Menghitung,


user

TGL. CETAK : 03/09/2009

Gambar 4.46 Laporan Penyakit Pasien Rawat Jalan

4.1.7.7 Laporan Dokter Pasien Rawat Jalan

Laporan ini menampilkan data-data dokter yang memeriksa pasien pada bangsal rawat jalan. Laporan ini dikelompokkan berdasarkan nama dokter yang memeriksa.


RUMAH SAKIT UMUM DAERAH DOMPU
 JL. KESEHATAN NO. 1 DOMPU
 TELP 0373-21411

LAPORAN DOKTER PASIEN RAWAT JALAN

NO	TGL. MEMERIKSA	NO MEMERIKSA	NOREG RAWAT	NORM	NAMA	DOKTHER PERIKSA	DIAGNOSA
TANGGAL LAPORAN : 01/07/2009 — 03/09/2009 DOKTHER YANG MEMERIKSA : dr. H.Fandi Agusubowo							
1	03/07/2009	060703-011-03	060703-001	0000004	RAHMATI	dr. H.Fandi Agusubowo	Superficial injuries involving multiple regions of upper limb(s)

Menghitung,

user

TGL. CETAK : 03/09/2009

Gambar 4.47 Laporan Dokter pasien Rawat Jalan

4.1.7.8 Laporan Tindakan Medis Pasien Rawat Jalan

Laporan ini menampilkan data-data tindakan medis yang dilakukan pertanggal. Laporan ini dikelompokkan berdasarkan jenis tindakan yang dilakukan.

NO	TGL TINDAKAN	NO TINDAKAN	NOMBOR JALAN	NORM	NAMA	ID TINDAKAN	JENIS TINDAKAN
TANGGAL LAPORAN : 01/07/2009 --- 03/09/2009							
TINDAKAN : Percutaneous denervation of facet							
1	03/09/2009	1307/09/002/01	1307/09/002	0000002	Pilze	03.95	Percutaneous denervation of facet
TANGGAL LAPORAN : 01/07/2009 --- 03/09/2009							
TINDAKAN : Closed (percutaneous) (needle)							
2	03/09/2009	1307/09/002/02	1307/09/002	0000003	Pilze	01.13	Closed (percutaneous) (needle) lossy of brain

Mergatiwala,

user

TGL CETAK : 03/09/2009

Gambar 4.48 Laporan Tindakan Pasien Rawat Jalan

4.1.7.9 Laporan Register Rawat Inap

Laporan ini menampilkan data register pasien pada bangsal Rawat Inap pertanggal. Laporan ini juga dilengkapi dengan penghitungan lama perawatan pasien pada bangsal rawat inap.

RUMAH SAKIT UMUM DAERAH DOMPU
JL. KESEHATAN NO. 1 DOMPU
TELP 0373-21411

LAPORAN INDEX REGISTRASI PASIEN RAWAT INAP

NO	TOL DAFTAR	TOL KELUAR	NO REG RAWAT INAP	NIKDM	NAMA	CINA MASUK	CARA BAYAR	NO KAWAL	LAMA RAWAT
1	27/01/2009	01/06/2009	27/01/09-001	00000005	sudarmo	AMBULANCE	LALAM	02	5 HARI

Mergistifid

Lain

TOL CETAK : 04090009

Gambar 4.49 Laporan Register Pasien Rawat Inap

4.1.7.10 Laporan Penyakit Pasien Rawat Inap

Laporan ini menampilkan data-data hasil diagnosa penyakit pasien Rawat Inap yang dikelompokkan berdasarkan penyakit pasien. Laporan ini dibagi lagi menurut kategori-kategori :

1. Diagnosa Utama

Laporan ini menampilkan data-data diagnosa utama pada pemeriksaan pasien rawat inap.

2. Laporan Diagnosa Komplikasi

Laporan ini menampilkan data-data hasil diagnosa komplikasi penyakit pasien pertanggal.

3. **Laporan Morfologi dan Neoplasma**

Laporan ini berisi data-data hasil pemeriksaan pasien bagian morfologi dan neoplasma.

4. **Laporan Penyebab Luar dan Cidera**


Laporan ini berisi data-data hasil pemeriksaan penyebab cidera pasien rawat inap.

5. **Laporan Imunisasi**

Laporan ini menampilkan data-data hasil pemeriksaan imunisasi pasien pada bangsal rawat inap.

4.1.7.11 **Laporan Laboratorium**

Laporan ini menampilkan data-data pemeriksaan Laboratorium beserta jenis pemeriksaan yang dilakukan setiap pasien pada setiap bangsal. Ini dikarenakan laboratorium memiliki tempat sendiri sehingga pemeriksaan dapat langsung dilakukan tanpa harus registrasi pada bangsal.


RUMAH SAKIT UMUM DAERAH DOMPU
 JL. KESEHATAN NO. 1 DOMPU
 TELP 0373-21411

LAPORAN LABORATORIUM PASIEN

NO	TGL. TERDOKSA	NO PEMERIKSAAN	NORM	NAMA	PEMERIKSAAN	HASIL	HASIL NORMAL	KETERANGAN
TANGGAL LAPORAN : 27/07/2009 --- 04082009								
1	02/08/2009	02/08/09-001	00000002	mubanih	GOLONGAN DARAH	B	-	-
2	02/08/2009	02/08/09-002	00000001	SCORON	GOLONGAN DARAH	A	-	2

Mingghu,


user

TGL CETAK : 04082009

Gambar 4.50 Laporan Laboratorium

4.1.7.12 Laporan Operasi pasien

Laporan ini berisikan data-data operasi pertanggal, baik itu jenis operasi, dokter yang memimpin jalannya operasi sampai jenis anastesi operasi.


RUMAH SAKIT UMUM DAERAH DOMPU
 JL. KESEHATAN NO. 1 DOMPU
 TELP 0373-21411

LAPORAN OPERASI PASIEN

NO	TGL. TERDOKSA	NO OPERASI	NORM	NAMA	ID OPERASI	OPERASI	DOKTOR	JENIS ANESTESI
TANGGAL LAPORAN : 27/07/2009 --- 04082009								
1	02/08/2009	02/08/09-001	00000000	Raze	0000	Prostectomy	Dr. H. H. H. H.	sedang

Mingghu,

TGL CETAK : 04082009

Gambar 4.51 Laporan Operasi Pasien

4.1.7.13 Laporan Radiologi pasien

Laporan ini menampilkan data-data pemeriksaan radiologi pasien pertanggal baik itu jenis pemeriksaan radiologi, sampai hasil dari radiologi tersebut.

RUMAH SAKIT UMUM DAERAH DOMPU
JL. KESEHATAN NO. 1 DOMPU
TELP 0373-21411

LAPORAN RADIOLOGI PASIEN

NO	TGL REGISI	NO PEMBUKUAN	NCRH	NAMA	ID RADIOLOGI	RADIOLOGI	HASIL	KETERANGAN
1	04/08/2009	04/08/09-001	00000001	SOEBIN	12	MAMUS	NORMAL	-
2	04/08/2009	04/08/09-002	00000002	Raise	03	Clavicle	RETR. FROAGMENT	RETR.
3	04/08/2009	04/08/09-003	00000003	rubiah	05	Estrontas tulang	-	normal
4	04/08/2009	04/08/09-004	00000004	sulamento	04	Estrontas alve	-	bangkok

Mengelola,

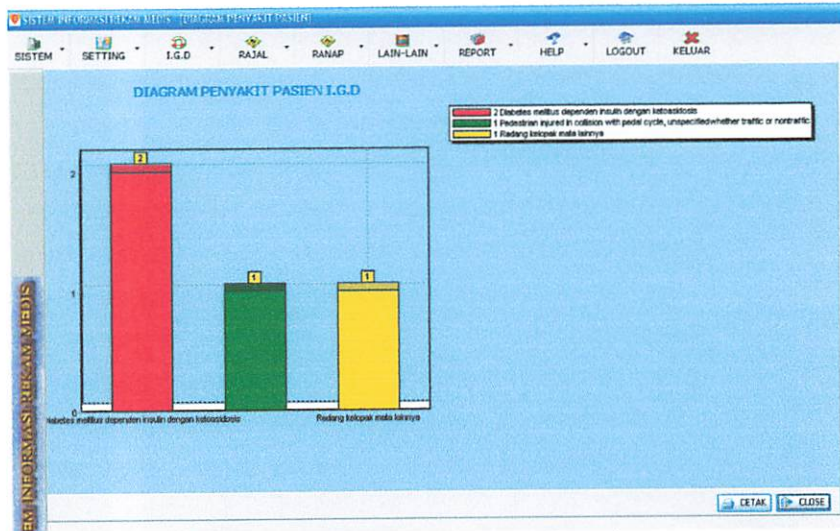
user

TGL CETAK : 04/08/2009

Gambar 4.52 Laporan Radiologi Pasien

4.1.7.14 Statistik Penyakit Pasien

Laporan ini berbentuk diagram yang fungsinya sebagai statistik perhitungan jumlah penderita penyakit pada masing-masing bangsal. Ini digunakan untuk mengetahui penyakit apa yang banyak diderita pasien sehingga pihak-pihak yang terkait dapat melakukan tindakan-tindakan pencegahan untuk mengurangi terjadinya penyakit tersebut.



Gambar 4.53 Diagram statistik Pasien

4.2 Pengujian Sistem

Sebagai cara untuk mengetahui unjuk kerja dari sistem yang dirancang dan dibuat, maka dilakukan pengujian dalam pengolahan data hingga pelaporan. Dalam pengujian ini dilakukan dengan menggunakan dua buah *Personal Computer* dengan menginputkan ± 150 data, dan hasilnya adalah sebagai berikut:

- Input data : Berhasil tanpa error
- Pencarian data : Berhasil tanpa error
- Edit data : Berhasil tanpa error
- Penyimpanan data : Berhasil tanpa error
- Cetak laporan : Berhasil tanpa error

Kemudian dilakukan perbandingan antara sistem lama (*ms. access*) dan sistem baru, hasil perbandingannya antara lain terdapat dalam tabel 4.1.

Tabel 4.1 perbandingan antara sistem lama dan sistem baru

NO.	proses	Sistem Lama (<i>ms. access</i>)	Sistem Baru (SI)
1.	Pendaftaran pasien, hasil pemeriksaan, tindakan medis pasien.	Masih dilakukan secara manual kemudian perawat dan petugas administrasi menginputkan kembali pada sistem lama.	Perawat dan petugas dapat langsung menginputkan data karena terdapat antarmuka dengan sistem.
2.	Hak akses	Semua orang dapat mengakses sistem karena tidak terdapat proses login.	Terdapat proses login dan hak akses terhadap pengguna sehingga pengaksesan data dapat diatur.
3.	Pembuatan laporan dan Statistik	Masih dengan cara manual dengan merekap ulang data	Pengguna dapat langsung dalam bentuk laporan dan dapat dipilih berdasarkan kriteria laporan yang akan dicetak.
4.	Pencetakan laporan	Tidak ada kriteria pencetakan, jadi semua data akan dicetak kemudian di rekap	Terdapat filter data sehingga terdapat kriteria pencetakan

Dari perbandingan kinerja sistem diatas, dapat dilihat keunggulan dari sistem informasi rekam medis yang dibuat. Sehingga dapat membantu memudahkan petugas Rumah sakit dalam melakukan pengelolaan informasi yang ada pada rumah sakit umum daerah dompu.

BAB V

PENUTUP

5.1. Kesimpulan

Dari hasil analisis dan perancangan aplikasi sistem informasi Rekam Medis serta implementasi dan pengujian yang telah dilakukan dapat diambil beberapa kesimpulan sebagai berikut :

1. Sistem informasi rekam medis dapat berjalan lancar walau dibuka pada saat bersamaan pada beberapa komputer.
2. Dengan fasilitas pelaporan dan statistik penyakit, proses pengontrolan dan administrasi lebih cepat dibandingkan dengan proses pelaporan secara manual.
3. Penghapusan data hanya data pasien yang tidak aktif yang dapat dihapus dan penghapusan data hanya dapat dilakukan oleh user yang memiliki group sebagai admin.
4. Dengan sistem pelaporan dan pembuatan statistik yang sudah otomatis, pihak rumah sakit dan jajaran yang terkait dalam pengambilan keputusan dan tindakan pencegahan penyakit menjadi lebih mudah karena tidak memerlukan waktu yang lama dalam pembuatan laporan manual dan pembuatan rekapitulasi penyakit dalam bentuk statistik secara manual.

5.2. Saran

Aplikasi Sistem Informasi Rekam Medis ini hanya dapat berjalan pada sistem operasi Microsoft Windows, dan Masih memerlukan banyak pembenahan, jadi jika program ini nantinya dijadikan acuan oleh pihak – pihak lain kami harapkan untuk lebih dikembangkan lagi menjadi suatu aplikasi sistem informasi yang lebih baik lagi.

DAFTAR PUSTAKA

- [1] Departemen Kesehatan RI, 1991, Petunjuk Teknis Penyelenggaraan Rekam Medis atau Medical Record Rumah Sakit, Jakarta: Dirjen Yanmed Depkes RI.
- [2] Sabarguna, B., 2005, Sistem Informasi Rumah Sakit, Yogyakarta Konsorsium Rumah Sakit Islam Jateng-DIY.
- [3] Hartono, J., 1999, Analisis dan Disain (Sistem Informasi : Pendekatan terstruktur, Teori dan praktek aplikasi bisnis), Yogyakarta: Andi.
- [4] Huffman, E.K. 1994, Health Information Management, Illyonis: Physician Record Company.
- [5] Hidayat, Rahmat. 2005, Belajar Database SQL Server 2000, www.ilmukomputer.com.
- [6] Andi. 2003, *Pemrograman Borland Delphi 7 (jilid I)*; -Ed. I. – Yogyakarta: Andi, Madiun; Madcoms.
- [7] Andi. 2003, *Pemrograman Borland Delphi 7(jilid II)*; -Ed. II. – Yogyakarta: Andi, Madiun; Madcoms.

LAMPIRAN



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Karanglo km 2, Malang

BERITA ACARA UJIAN SKRIPSI FAKULTAS TEKNOLOGI INDUSTRI

Nama : Muhamad Khairun

NIM : 04.12.748

Jurusan : Teknik Elektro S-1

Konsentrasi : Teknik Komputer dan Informatika

Judul Skripsi : **PENGEMBANGAN SISTEM INFORMASI REKAM MEDIS
PADA RUMAH SAKIT UMUM DAERAH DOMPU**

Dipertahankan di hadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :

Hari : Sabtu

Tanggal : 05 September 2009

Dengan Nilai : 83,9 (A) *By*



Ketua Majelis Penguji

Ir. H. Sidik Noertjahjono, MT.
NIP.Y 102 8700 163

Sekretaris Majelis Penguji

Ir. F. Yudi Limpraptono, MT.
NIP.Y 103 9500 274

Penguji I

I Komang Somawirata, ST.MT
NIP.Y. 103 0100 361

Penguji II

Sandy Nataly M., SKom



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Karanglo km 2, Malang

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Komputer dan Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : Muhamad Khairun
NIM : 04.12.748
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer dan Informatika
Hari,Tanggal : Sabtu, 5 September 2009
Judul Skripsi : **PENGEMBANGAN SISTEM INFORMASI REKAM MEDIS
PADA RUMAH SAKIT UMUM DAERAH DOMPU**

No	Materi Perbaikan	Paraf
1.	Proteksi inputan data kosong tidak bisa disimpan	
2.	Perbaiki kesimpulan	
3.	Font dilaporan disamakan 12pt	

Disetujui :
Penguji I

I Komang Somawirata, ST.MT
NIP.Y. 103 0100 361

Mengetahui :

Dosen Pembimbing I

Ir. F. Yudi Limpraptono, MT.
NIP.Y 103 9500 274

Dosen Pembimbing II

Joseph Dedy Irawan, ST, MT.
NIP. 132 315 178



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Karanglo km 2, Malang

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Komputer dan Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : Muhamad Khairun
NIM : 04.12.748
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer dan Informatika
Hari, Tanggal : Sabtu, 5 September 2009
Judul Skripsi : **PENGEMBANGAN SISTEM INFORMASI REKAM MEDIS
PADA RUMAH SAKIT UMUM DAERAH DOMPU**

No	Materi Perbaikan	Paraf
1.	Perbaiki program, inputan data kosong dan nama dokter yang bertanggung jawab atas pasien, kode pos tidak boleh huruf	
2.	Perbaiki DAD level 0	
3.	Perbaiki kesimpulan	
4.	Tambahkan hasil pengujian <i>client-server</i>	

Disetujui :

Penguji II

Sandy Nataly M., SKom

Mengetahui :

Dosen Pembimbing I

Ir. F. Yudi Limpraptono, MT.
NIP. Y 103 9500 274

Dosen Pembimbing II

Joseph Dedy Irawan, ST, MT.
NIP. 132 315 178



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : M. KHARUN
NIM : 04.12.748.
Perbaikan melalui :

- * Font - ukuran semua 12 pt.
- * program Aplikasi di beri efek pengalihan di portabeli jika data besar ke lain di simpan
- * kesimpulan - dari hasil pengujian

Malang, 5 - 09 - 2008



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : M. KHAIRUN
NIM : 0412918
Perbaikan melalui :

- PERBAIKI PROGRAM (PENYIMPANAN DATA RECORD, INPUTAN NAMA DOKTORUS BERTANGGUNG JAWAB ATAS PASIEN) KODEPUS)
- PERbaiki DFD MULAI LEVEL 0
- PERbaiki KESIMPULAN
- TAMBAHKAN HASIL PEMERIKSAAN CLIENT-SERVER

Malang, 5 sept 2009

()



FORMULIR BIMBINGAN SKRIPSI

Nama : MUHAMAD KHAIRUN
Nim : 04.12.748
Masa Bimbingan : 02 Desember 2008 s/d 02 Juni 2009
Judul Skripsi : PENGEMBANGAN SISTEM INFORMASI REKAM MEDIS
PADA RUMAH SAKIT UMUM DAERAH DOMPU

No	Tanggal	Uraian	Paraf Pembimbing
1	12/02/09	Perbaikan Bab I & II	
2	23/07/09	Tambahan struktur argumen di bagian + penyusunan user interface	
3	3/08/09	Bab II diperbaiki	
4	7/08/09	Bab V diperbaiki	
5	24/08/09	Seluruh ok	
6			
7			
8			
9			
10			

Malang,
Dosen Pembimbing I

Ir. F. Yudi Limpraptono, MT
NIP.Y. 1639500274



FORMULIR BIMBINGAN SKRIPSI

Nama : MUHAMAD KHAIRUN
Nim : 04.12.748
Masa Bimbingan : 02 Desember 2008 s/d 02 Juni 2009
Judul Skripsi : PENGEMBANGAN SISTEM INFORMASI REKAM MEDIS
PADA RUMAH SAKIT UMUM DAERAH DOMPU

No	Tanggal	Uraian	Paraf Pembimbing
1	23/7/09	perbaiki Bab I & II	
2	24/7/09	BAB I, II OK	
3	6/8/09	BAB II OK	
4	7/8/09	PENGUDIAN SISTEM (BAB IV) PERBAIKI	
5	9/8/09	BAB IV OK	
6	10/8/09	BAB V OK	
7	24/8/09	SEMINAR HASIL OK	
8	26/8/09	AKU LUMPUR	
9			
10			

Malang,
Dosen Pembimbing II


Joseph Dedy Irawan, ST, MT
NIP. 132315178

Form S-4b

```
unit U_menu_utama;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, Menus, ExtCtrls, StdCtrls, RzLabel, RzPanel, pngimage, RzButton,  
RzEdit, Mask, DB, DBAccess, MyAccess, ComCtrls, ToolWin, ImgList, jpeg,  
JvGIF;
```

```
type
```

```
Tf_menu_utama = class(TForm)  
    Image1: TImage;  
    ds_login: TMyDataSource;  
    ImageList1: TImageList;  
    pp_sistem: TPopupMenu;  
    ToolBar2: TToolBar;  
    ToolBar3: TToolBar;  
    LOGIN2: TMenuItem;  
    LOGOUT1: TMenuItem;  
    N3: TMenuItem;  
    GANTIPASSWORD1: TMenuItem;  
    N4: TMenuItem;  
    EXIT3: TMenuItem;  
    ToolBar1: TToolBar;  
    t_sistem: TToolButton;  
    ToolButton2: TToolButton;  
    t_seting: TToolButton;  
    ToolButton4: TToolButton;  
    t_rajal: TToolButton;  
    pp_seting: TPopupMenu;  
    pp_rajal: TPopupMenu;  
    pp_ranap: TPopupMenu;  
    pp_lab: TPopupMenu;  
    pp_report: TPopupMenu;  
    ToolButton6: TToolButton;  
    t_ranap: TToolButton;  
    ToolButton8: TToolButton;  
    t_lab: TToolButton;  
    ToolButton10: TToolButton;  
    t_report: TToolButton;  
    ToolButton12: TToolButton;  
    t_help: TToolButton;  
    ToolButton14: TToolButton;  
    t_keluar: TToolButton;  
    pp_help: TPopupMenu;
```

```
MANAJEMENUSER1: TMenuItem;  
ISIDATA1: TMenuItem;  
EDITHAPUSPASIEN1: TMenuItem;  
DAFTARPASIENBARU1: TMenuItem;  
REGISTRASIRAJAL1: TMenuItem;  
PemeriksaanRAJAL1: TMenuItem;  
INDAKANRAJAL1: TMenuItem;  
PENDAFTARANPASIENBARU1: TMenuItem;  
REGISTRASIRANAP1: TMenuItem;  
PemeriksaanRANAP1: TMenuItem;  
INDAKANRANAP1: TMenuItem;  
PINDAHKAMARPASIEN1: TMenuItem;  
PENCARIANKAMAR1: TMenuItem;  
KEADAANKELUAR1: TMenuItem;  
LABORATORIUM1: TMenuItem;  
RADIOLOGI1: TMenuItem;  
OPERASI1: TMenuItem;  
ds_user: TMyDataSource;  
PENCARIANPASIEN1: TMenuItem;  
EDITDATAPASIEN1: TMenuItem;  
HAPUSPASIEN1: TMenuItem;  
bt_logout: TToolButton;  
ToolButton3: TToolButton;  
Timer1: TTimer;  
dataTgl: TToolButton;  
dataJam: TToolButton;  
RzPanel1: TRzPanel;  
Image2: TImage;  
ToolButton1: TToolButton;  
ToolButton5: TToolButton;  
t_igd: TToolButton;  
ToolButton9: TToolButton;  
pp_igd: TPopupMenu;  
PENDAFTARANPASIENBARU2: TMenuItem;  
DAFTARIGD1: TMenuItem;  
PemeriksaanIGD1: TMenuItem;  
INDAKANIGD1: TMenuItem;  
RUJUKANIGD1: TMenuItem;  
ToolButton7: TToolButton;  
ToolButton11: TToolButton;  
ToolButton13: TToolButton;  
RUJUKANRAJAL1: TMenuItem;  
ToolButton15: TToolButton;  
ToolButton16: TToolButton;  
ToolButton17: TToolButton;  
ToolButton18: TToolButton;
```

```

ToolButton19: TToolButton;
ToolButton20: TToolButton;
RUJUKANRANAP1: TMenuItem;
LAPORANUTAMA1: TMenuItem;
STATISTIK1: TMenuItem;
PENYAKITIGD1: TMenuItem;
PENYAKITRAJAL1: TMenuItem;
PENYAKITRANAP1: TMenuItem;
procedure t_keluarClick(Sender: TObject);
procedure LOGIN2Click(Sender: TObject);
procedure EXIT3Click(Sender: TObject);
procedure LOGOUT1Click(Sender: TObject);
procedure MANAJEMENUUSER1Click(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure ISIDATA1Click(Sender: TObject);
procedure EDITDATAPASIEN1Click(Sender: TObject);
procedure HAPUSPASIEN1Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure DAFTARPASIENBARU1Click(Sender: TObject);
procedure REGISTRASIRAJAL1Click(Sender: TObject);
procedure PEMERIKSAANRAJAL1Click(Sender: TObject);
procedure INDAKANRAJAL1Click(Sender: TObject);
procedure PENDAFTARANPASIENBARU1Click(Sender: TObject);
procedure REGISTRASIRANAP1Click(Sender: TObject);
procedure PEMERIKSAANRANAP1Click(Sender: TObject);
procedure INDAKANRANAP1Click(Sender: TObject);
procedure PINDAHKAMPASIEN1Click(Sender: TObject);
procedure PENCARIANKAMAR1Click(Sender: TObject);
procedure KEADAANKELUAR1Click(Sender: TObject);
procedure LABORATORIUM1Click(Sender: TObject);
procedure RADIOLOGI1Click(Sender: TObject);
procedure PENCARIANPASIEN1Click(Sender: TObject);
procedure bt_logoutClick(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure OPERASI1Click(Sender: TObject);
procedure GANTIPASSWORD1Click(Sender: TObject);
procedure PENDAFTARANPASIENBARU2Click(Sender: TObject);
procedure DAFTARIGD1Click(Sender: TObject);
procedure PEMERIKSAANIGD1Click(Sender: TObject);
procedure INDAKANIGD1Click(Sender: TObject);
procedure RUJUKANIGD1Click(Sender: TObject);
procedure RUJUKANRAJAL1Click(Sender: TObject);
procedure RUJUKANRANAP1Click(Sender: TObject);
procedure FormCloseQuery(Sender: TObject; var CanClose: Boolean);
procedure LAPORANUTAMA1Click(Sender: TObject);

```

```

procedure PENYAKITIGD1Click(Sender: TObject);
procedure PENYAKITRAJAL1Click(Sender: TObject);
procedure PENYAKITRANAP1Click(Sender: TObject);

```

```

private
{ Private declarations }
procedure show5;
public
{ Public declarations }
end;

```

```

var
f_menu_utama: Tf_menu_utama;
waktu: TDateTime;
implementation

```

```

uses Unit2, U_cara_bayar, U_cari_pasien, U_cari_reg, U_cari_reg_ranap,
U_daftar_rajal, U_daftar_ranap, U_diagnosa, U_edit_pasien, U_imun_dtlpas,
U_isi_data, U_keluar_rajal, U_lab_dtl, U_laboratorium,
U_lihat_data_pasien, U_lihat_dokter, U_managment_user, U_operasi,
U_operasi_detail, U_pasien_hapus, U_pemeriksaan_rajal,
U_pemeriksaan_ranap, U_peny_cidera_dtl, U_pindah_kamar, U_poli,
U_radiologi, U_radiologi_dtl, U_status_kamar, U_tindakan_dtl,
U_tindakan_rajal, U_tindakan_ranap, Unit1, U_kamar_search,
U_lap_index_dkter_rajal, U_lap_pasien_rajalpas, U_lap_pasien_ranap,
U_lap_penyakit_rajal, U_lap_penyakit_ranap, U_login, U_ganti_pass,
U_nom_pasienpas, U_reg_rajal_akhir, U_reg_ranap_akhir, U_splash,
UConnection, UDialogs, U_daftar_igd, U_reg_igd, U_pemeriksaan_igd,
U_tindakan_igd, U_rujuk_igd, U_lap_reg_igd, U_lap_pemeriksaan_igd,
U_lap_tindakan_igd, U_lap_tindakan_rajal, U_lap_diag_komplik,
U_lap_diagnosa_ranap, U_lap_imunisasi, U_lap_morfo_ranap,
U_lap_peny_cidera, U_lap_tindakan_ranap, U_cetak_kartu, U_lap_lab,
U_lap_operasi, U_rujukan_rajal, U_backup, U_rujuk_ranap, U_lap_utama,
PSE_SplashUnit, U_lap_dokter_igd, U_lap_dokter_ranap, U_lap_radio,
U_statistik_penyakit, U_stat_penyakit_rajal, U_stat_penyakit_rana;

```

```
{ $R *.dfm }
```

```

procedure Tf_menu_utama.show5;
begin
f_menu_utama.t_seting.Enabled := false;
f_menu_utama.t_igd.Enabled := false;
f_menu_utama.t_rajal.Enabled := false;
f_menu_utama.t_ranap.Enabled := false;
f_menu_utama.t_lab.Enabled := false;

```

```

f_menu_utama.t_report.Enabled := false;
f_menu_utama.t_help.Enabled := false;
f_menu_utama.bt_logout.Enabled := false;
f_menu_utama.pp_ranap.Items[0].Enabled := true;
f_menu_utama.pp_ranap.Items[1].Enabled := true;
f_menu_utama.pp_ranap.Items[2].Enabled := true;
f_menu_utama.pp_ranap.Items[3].Enabled := true;
f_menu_utama.pp_ranap.Items[4].Enabled := true;
f_menu_utama.pp_ranap.Items[5].Enabled := true;
f_menu_utama.pp_ranap.Items[6].Enabled := true;
f_menu_utama.pp_rajal.Items[0].Enabled := true;
f_menu_utama.pp_rajal.Items[1].Enabled := true;
f_menu_utama.pp_rajal.Items[2].Enabled := true;
end;

procedure Tf_menu_utama.t_keluarClick(Sender: TObject);
begin
    if MessageBox(Handle,'Keluar dari Aplikasi simrm...!', 'SIMRM',MB_YESNO +
MB_ICONQUESTION) = ID_YES then
        application.Terminate;
end;

procedure Tf_menu_utama.LOGIN2Click(Sender: TObject);
var
    i: integer;
    tampil: boolean;
begin
    tampil := False;
    for i := 0 to MDIChildCount - 1 do
        if MDIChildren[i] is Tf_login then
            begin
                tampil := True;
                Break;
            end;
        if not tampil then
            begin
                if F_login = nil then
                    F_login := Tf_login.Create(Self);
                    F_login.Show;
                end
            else
                MDIChildren[i].Show;
            end;
    end;
end;

procedure Tf_menu_utama.EXIT3Click(Sender: TObject);
begin

```

```

        if MessageBox(Handle,'Keluar dari Aplikasi simrm?', 'SIMRM',MB_YESNO +
MB_ICONQUESTION) = ID_YES then
            application.Terminate;
end;

procedure Tf_menu_utama.LOGOUT1Click(Sender: TObject);
begin
    bt_logoutClick(self);
end;

procedure Tf_menu_utama.MANAJEMENUSER1Click(Sender: TObject);
var
    i: integer;
    tampil: boolean;
begin
    tampil := False;
    for i := 0 to MDIChildCount - 1 do
        if MDIChildren[i] is Tf_manajemen_user then
            begin
                tampil := True;
                Break;
            end;
        if not tampil then
            begin
                if f_manajemen_user = nil then
                    F_manajemen_user := Tf_manajemen_user.Create(Self);
                    F_manajemen_user.Show;
                end
            else
                MDIChildren[i].Show;
            end;
    end;

procedure Tf_menu_utama.FormShow(Sender: TObject);
begin
    show5;
end;

procedure Tf_menu_utama.ISIDATA1Click(Sender: TObject);
var
    i: integer;
    tampil: boolean;
begin
    tampil := False;
    for i := 0 to MDIChildCount - 1 do

```

```

if MDIChildren[i] is Tf_isi_data then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if f_isi_data = nil then
F_isi_data := Tf_isi_data.Create(Self);
F_isi_data.Show;
end
else
MDIChildren[i].Show;
end;

procedure Tf_menu_utama.EDITDATAPASIEN1Click(Sender: TObject);
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is Tf_edit_pasien then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if f_edit_pasien = nil then
F_edit_pasien := Tf_edit_pasien.Create(Self);
F_edit_pasien.Show;
end
else
MDIChildren[i].Show;
end;
end;
procedure Tf_menu_utama.HAPUSPASIEN1Click(Sender: TObject);
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is Tf_hapus_pasien then
begin
tampil := True;

```

```

Break;
end;
if not tampil then
begin
if f_hapus_pasien = nil then
F_hapus_pasien := Tf_hapus_pasien.Create(Self);
F_hapus_pasien.Show;
end
else
MDIChildren[i].Show;
end;

procedure Tf_menu_utama.FormCreate(Sender: TObject);
begin
timer1.Enabled:=true;
timer1.Interval:=1;
pp_sistem.Items[1].Enabled := false;
pp_sistem.Items[3].Enabled := false;
pp_sistem.Items[5].Enabled := false;
pp_sistem.Items[6].Enabled := false;

end;

procedure Tf_menu_utama.DAFTARPASIENBARU1Click(Sender: TObject);
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is TF_PENDAFTARAN_PASIEN_BARU then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if F_PENDAFTARAN_PASIEN_BARU = nil then
F_PENDAFTARAN_PASIEN_BARU :=
TF_PENDAFTARAN_PASIEN_BARU.Create(Self);
F_PENDAFTARAN_PASIEN_BARU.Show;
end
else
MDIChildren[i].Show;
end;
end;

```

```

procedure Tf_menu_utama.REGISTRASIRAJAL1Click(Sender: TObject);
begin
    if F_daftar_rajal = nil then
        F_daftar_rajal := TF_daftar_rajal.Create(Self);
        F_daftar_rajal.Show;
    end;

procedure Tf_menu_utama.PEMERIKSAANRAJAL1Click(Sender: TObject);
var
    i: integer;
    tampil: boolean;
begin
    tampil := False;
    for i := 0 to MDIChildCount - 1 do
        if MDIChildren[i] is Tf_pemeriksaan_rajal then
            begin
                tampil := True;
                Break;
            end;
        if not tampil then
            begin
                if f_pemeriksaan_ranap = nil then
                    F_pemeriksaan_rajal := Tf_pemeriksaan_rajal.Create(Self);
                    F_pemeriksaan_rajal.Show;
                end
            else
                MDIChildren[i].Show;
            end;
    end;

procedure Tf_menu_utama.INDAKANRAJAL1Click(Sender: TObject);
var
    i: integer;
    tampil: boolean;
begin
    tampil := False;
    for i := 0 to MDIChildCount - 1 do
        if MDIChildren[i] is Tf_tindakan_rajal then
            begin
                tampil := True;
                Break;
            end;
        if not tampil then
            begin
                if F_tindakan_rajal = nil then
                    F_tindakan_rajal := Tf_tindakan_rajal.Create(Self);
                    F_tindakan_rajal.Show;

```

```

        end
    else
        MDIChildren[i].Show;
    end;

procedure Tf_menu_utama.PENDAFTARANPASIENBARU1Click(Sender:
TObject);
var
    i: integer;
    tampil: boolean;
begin
    tampil := False;
    for i := 0 to MDIChildCount - 1 do
        if MDIChildren[i] is TF_PENDAFTARAN_PASIEN_BARU then
            begin
                tampil := True;
                Break;
            end;
        if not tampil then
            begin
                if F_PENDAFTARAN_PASIEN_BARU = nil then
                    F_PENDAFTARAN_PASIEN_BARU :=
TF_PENDAFTARAN_PASIEN_BARU.Create(Self);
                    F_PENDAFTARAN_PASIEN_BARU.Show;
                end
            else
                MDIChildren[i].Show;
            end;
    end;

procedure Tf_menu_utama.REGISTRASIRANAP1Click(Sender: TObject);
var
    i: integer;
    tampil: boolean;
begin
    tampil := False;
    for i := 0 to MDIChildCount - 1 do
        if MDIChildren[i] is TF_daftar_ranap then
            begin
                tampil := True;
                Break;
            end;
        if not tampil then
            begin
                if F_daftar_ranap = nil then
                    F_daftar_ranap := TF_daftar_ranap.Create(Self);
                    F_daftar_ranap.Show;

```



```
end
else
MDIChildren[i].Show;
end;
```

```
procedure Tf_menu_utama.PEMERIKSAANRANAP1Click(Sender: TObject);
```

```
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is Tf_pemeriksaan_ranap then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if f_pemeriksaan_ranap = nil then
F_pemeriksaan_ranap := Tf_pemeriksaan_ranap.Create(Self);
F_pemeriksaan_ranap.Show;
end
else
MDIChildren[i].Show;
end;
end;
```

```
procedure Tf_menu_utama.INDAKANRANAP1Click(Sender: TObject);
```

```
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is Tf_tindakan_ranap then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if f_tindakan_ranap = nil then
F_tindakan_ranap := Tf_tindakan_ranap.Create(Self);
F_tindakan_ranap.Show;
end
end;
```

```
else
MDIChildren[i].Show;
end;
```

```
procedure Tf_menu_utama.PINDAHKAMARPASIEN1Click(Sender: TObject);
```

```
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is TF_pindah_kamar then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if F_pindah_kamar = nil then
F_pindah_kamar := TF_pindah_kamar.Create(Self);
F_pindah_kamar.Show;
end
else
MDIChildren[i].Show;
end;
end;
```

```
procedure Tf_menu_utama.PENCARIANKAMAR1Click(Sender: TObject);
```

```
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is Tf_kamar_search then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if f_kamar_search = nil then
f_kamar_search := Tf_kamar_search.Create(Self);
f_kamar_search.Show;
end
end
else
```

```

MDIChildren[i].Show;
end;

procedure Tf_menu_utama.KEADAANKELUAR1Click(Sender: TObject);
var
  i: integer;
  tampil: boolean;
begin
  tampil := False;
  for i := 0 to MDIChildCount - 1 do
    if MDIChildren[i] is Tf_keadaan_keluar_ranap then
      begin
        tampil := True;
        Break;
      end;
  end;
  if not tampil then
    begin
      if f_keadaan_keluar_ranap = nil then
        f_keadaan_keluar_ranap := Tf_keadaan_keluar_ranap.Create(Self);
        f_keadaan_keluar_ranap.Show;
      end
    else
      MDIChildren[i].Show;
    end;
end;

```

```

procedure Tf_menu_utama.LABORATORIUM1Click(Sender: TObject);
var
  i: integer;
  tampil: boolean;
begin
  tampil := False;
  for i := 0 to MDIChildCount - 1 do
    if MDIChildren[i] is Tf_laboratorium then
      begin
        tampil := True;
        Break;
      end;
  end;
  if not tampil then
    begin
      if f_laboratorium = nil then
        f_laboratorium := Tf_laboratorium.Create(Self);
        f_laboratorium.Show;
      end
    else
      MDIChildren[i].Show;
    end;
end;

```

```

procedure Tf_menu_utama.RADIOLOGI1Click(Sender: TObject);
var
  i: integer;
  tampil: boolean;
begin
  tampil := False;
  for i := 0 to MDIChildCount - 1 do
    if MDIChildren[i] is Tf_radiologi then
      begin
        tampil := True;
        Break;
      end;
  end;
  if not tampil then
    begin
      if f_radiologi = nil then
        f_radiologi := Tf_radiologi.Create(Self);
        f_radiologi.Show;
      end
    else
      MDIChildren[i].Show;
    end;
end;

```

```

procedure Tf_menu_utama.PENCARIANPASIEN1Click(Sender: TObject);
var
  i: integer;
  tampil: boolean;
begin
  tampil := False;
  for i := 0 to MDIChildCount - 1 do
    if MDIChildren[i] is Tf_cari_pasien then
      begin
        tampil := True;
        Break;
      end;
  end;
  if not tampil then
    begin
      if f_cari_pasien = nil then
        F_cari_pasien := Tf_cari_pasien.Create(Self);
        F_cari_pasien.Show;
      end
    else
      MDIChildren[i].Show;
    end;
end;

```

```

procedure Tf_menu_utama.bt_logoutClick(Sender: TObject);

```

```

var
  i: Integer;
begin
  if MessageBox(Handle,'LogOut dari program?', 'LogOut Sistem', MB_YESNO +
  MB_ICONQUESTION) = ID_YES then
    begin
      for i := 0 to MDIChildCount - 1 do
        MDIChildren[i].Close;
        pp_sistem.Items[0].Enabled := true;
        pp_sistem.Items[1].Enabled := false;
        pp_sistem.Items[3].Enabled := false;
        pp_sistem.Items[5].Enabled := false;
        pp_sistem.Items[6].Enabled := false;
        ToolButton1.Caption := 'user';
        show5;
      end;
    end;

  end;

  procedure Tf_menu_utama.Timer1Timer(Sender: TObject);
  begin
    dataTgl.Caption := DateToStr(Now);
    dataJam.Caption := TimeToStr(Now);
  end;

  procedure Tf_menu_utama.FormClose(Sender: TObject;
  var Action: TCloseAction);
  begin
    Action := caFree;
  end;

  procedure Tf_menu_utama.OPERASI1Click(Sender: TObject);
  var
    i: Integer;
    tampil: boolean;
  begin
    tampil := False;
    for i := 0 to MDIChildCount - 1 do
      if MDIChildren[i] is Tf_operasi then
        begin
          tampil := True;
          Break;
        end;
      if not tampil then
        begin
          if f_operasi = nil then

```

```

          f_operasi := Tf_operasi.Create(Self);
          f_operasi.Show;
        end
      else
        MDIChildren[i].Show;
    end;

  procedure Tf_menu_utama.GANTIPASSWORD1Click(Sender: TObject);
  var
    i: Integer;
    tampil: boolean;
  begin
    tampil := False;
    for i := 0 to MDIChildCount - 1 do
      if MDIChildren[i] is Tf_operasi then
        begin
          tampil := True;
          Break;
        end;
      if not tampil then
        begin
          if f_ganti_pass = nil then
            f_ganti_pass := Tf_ganti_pass.Create(Self);
            f_ganti_pass.Show;
          end
        else
          MDIChildren[i].Show;
        end;
    end;

  procedure Tf_menu_utama.PENDAFTARANPASIENBARU2Click(Sender:
  TObject);
  var
    i: Integer;
    tampil: boolean;
  begin
    tampil := False;
    for i := 0 to MDIChildCount - 1 do
      if MDIChildren[i] is TF_PENDAFTARAN_PASIEN_BARU then
        begin
          tampil := True;
          Break;
        end;
      if not tampil then
        begin
          if F_PENDAFTARAN_PASIEN_BARU = nil then

```

```

F_PENDAFTARAN_PASIEN_BARU :=
TF_PENDAFTARAN_PASIEN_BARU.Create(Self);
F_PENDAFTARAN_PASIEN_BARU.Show;
end
else
MDIChildren[i].Show;
end;

```

```

procedure Tf_menu_utama.DAFTARIGD1Click(Sender: TObject);
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is Tf_daftar_igd then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if f_daftar_igd = nil then
f_daftar_igd := Tf_daftar_igd.Create(Self);
f_daftar_igd.Show;
end
else
MDIChildren[i].Show;
end;

```

```

procedure Tf_menu_utama.PEMERIKSAANIGD1Click(Sender: TObject);
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is Tf_pemeriksaan_igd then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if f_pemeriksaan_igd = nil then
f_pemeriksaan_igd := Tf_pemeriksaan_igd.Create(Self);

```

```

f_pemeriksaan_igd.Show;
end
else
MDIChildren[i].Show;
end;

```

```

procedure Tf_menu_utama.INDAKANIGD1Click(Sender: TObject);
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is Tf_tindakan_igd then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if f_tindakan_igd = nil then
f_tindakan_igd := Tf_tindakan_igd.Create(Self);
f_tindakan_igd.Show;
end
else
MDIChildren[i].Show;
end;

```

```

procedure Tf_menu_utama.RUJUKANIGD1Click(Sender: TObject);
var
i: integer;
tampil: boolean;
begin
tampil := False;
for i := 0 to MDIChildCount - 1 do
if MDIChildren[i] is Tf_rujukan_igd then
begin
tampil := True;
Break;
end;
if not tampil then
begin
if f_rujukan_igd = nil then
f_rujukan_igd := Tf_rujukan_igd.Create(Self);
f_rujukan_igd.Show;
end

```

```

else
  MDIChildren[i].Show;
end;
procedure Tf_menu_utama.RUJUKANRAJAL1Click(Sender: TObject);
var
  i: Integer;
  tampil: boolean;
begin
  tampil := False;
  for i := 0 to MDIChildCount - 1 do
    if MDIChildren[i] is Tf_rujukan_rajal then
      begin
        tampil := True;
        Break;
      end;
    if not tampil then
      begin
        if f_rujukan_rajal = nil then
          f_rujukan_rajal := Tf_rujukan_rajal.Create(Self);
          f_rujukan_rajal.Show;
        end
      end
    else
      MDIChildren[i].Show;
    end;
end;

```

```

procedure Tf_menu_utama.RUJUKANRANAP1Click(Sender: TObject);
var
  i: Integer;
  tampil: boolean;
begin
  tampil := False;
  for i := 0 to MDIChildCount - 1 do
    if MDIChildren[i] is Tf_rujukan_ranap then
      begin
        tampil := True;
        Break;
      end;
    if not tampil then
      begin
        if f_rujukan_ranap = nil then
          f_rujukan_ranap := Tf_rujukan_ranap.Create(Self);
          f_rujukan_ranap.Show;
        end
      end
    else
      MDIChildren[i].Show;
    end;
end;

```

```

end;
procedure Tf_menu_utama.FormCloseQuery(Sender: TObject;
  var CanClose: Boolean);
begin
  CanClose := Confirm('Keluar dari Aplikasi simrm ?');
end;

procedure Tf_menu_utama.LAPORANUTAMA1Click(Sender: TObject);
var
  i: Integer;
  tampil: boolean;
begin
  tampil := False;
  for i := 0 to MDIChildCount - 1 do
    if MDIChildren[i] is Tf_lap_utama then
      begin
        tampil := True;
        Break;
      end;
    if not tampil then
      begin
        if f_lap_utama = nil then
          f_lap_utama := Tf_lap_utama.Create(Self);
          f_lap_utama.Show;
        end
      end
    else
      MDIChildren[i].Show;
    end;
end;

```

```

procedure Tf_menu_utama.PENYAKITIGD1Click(Sender: TObject);
var
  i: Integer;
  tampil: boolean;
begin
  tampil := False;
  for i := 0 to MDIChildCount - 1 do
    if MDIChildren[i] is Tf_statistik_penyakit then
      begin
        tampil := True;
        Break;
      end;
    if not tampil then
      begin
        if f_statistik_penyakit = nil then

```

```

begin
  if f_statistik_penyakit = nil then
    f_statistik_penyakit := Tf_statistik_penyakit.Create(Self);
    f_statistik_penyakit.Isi;
    f_statistik_penyakit.Show;
  end
else
  MDIChildren[i].Show;
end;

procedure Tf_menu_utama.PENYAKITRAJAL1Click(Sender: TObject);
var
  i: integer;
  tampil: boolean;
begin
  tampil := False;
  for i := 0 to MDIChildCount - 1 do
    if MDIChildren[i] is Tf_statistik_penyakit_rajal then
      begin
        tampil := True;
        Break;
      end;
    if not tampil then
      begin
        if f_statistik_penyakit_rajal = nil then
          f_statistik_penyakit_rajal := Tf_statistik_penyakit_rajal.Create(Self);
          f_statistik_penyakit_rajal.Isi;
          f_statistik_penyakit_rajal.Show;
        end
      end
    else
      MDIChildren[i].Show;
    end;
end;

```

```

procedure Tf_menu_utama.PENYAKITRANAP1Click(Sender: TObject);
var
  i: integer;
  tampil: boolean;
begin
  tampil := False;
  for i := 0 to MDIChildCount - 1 do
    if MDIChildren[i] is Tf_statistik_penyakit_ranap then
      begin
        tampil := True;
        Break;
      end;
    end;
end;

```

```

if not tampil then
  begin
    if f_statistik_penyakit_ranap = nil then
      f_statistik_penyakit_ranap := Tf_statistik_penyakit_ranap.Create(Self);
      f_statistik_penyakit_ranap.Isi;
      f_statistik_penyakit_ranap.Show;
    end
  else
    MDIChildren[i].Show;
  end;
end;

```

END.

unit Unit1;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
 Dialogs, Grids, DBGrids, RzDBGrid, RzButton, RzEdit, RzDBEdit, StdCtrls,
 DBCtrls, Mask, ExtCtrls, RzPanel, RzDBCmbo, DB, DBAccess, MyAccess,
 RzLabel, RzCmboBx, jpeg, MemDS;

type

```

TF_PENDAFTARAN_PASIEN_BARU = class(TForm)
  RzPanel1: TRzPanel;
  grid_pasien: TRzDBGrid;
  Label18: TLabel;
  ds_pasien_insert: TMyDataSource;
  ds_pasien_tampil: TMyDataSource;
  ds_pasien_edit: TMyDataSource;
  b_close: TRzBitBtn;
  btn_batal: TRzBitBtn;
  b_edit: TRzBitBtn;
  b_simpan: TRzBitBtn;
  RzBitBtn1: TRzBitBtn;
  Image1: TImage;
  Label1: TLabel;
  e_norm: TRzEdit;
  ed_nama: TRzEdit;
  Label2: TLabel;
  Label3: TLabel;
  ed_tmpt_lahir: TRzEdit;
  Label4: TLabel;
  dte_tgl_lahir: TRzDateTimeEdit;
  Label7: TLabel;
end;

```

```

cb_jenis_kelamin: TRzComboBox;
ed_no_ktp: TRzEdit;
Label19: TLabel;
Label8: TLabel;
cb_agama: TRzComboBox;
cb_gol_darah: TRzComboBox;
Label6: TLabel;
Label5: TLabel;
ed_pendidikan: TRzEdit;
Label10: TLabel;
ed_pekerjaan: TRzEdit;
Label9: TLabel;
cb_status_kawin: TRzComboBox;
Label11: TLabel;
mem_alamat: TRzMemo;
Label13: TLabel;
ed_kota: TRzEdit;
RzLabel1: TRzLabel;
ed_kd_pos: TRzEdit;
Label12: TLabel;
ed_provinsi: TRzEdit;
Label14: TLabel;
ed_telpon: TRzEdit;
Label15: TLabel;
ed_status_alergi: TRzEdit;
Label16: TLabel;
ed_nama_ayah: TRzEdit;
Label17: TLabel;
ed_nama_ibu: TRzEdit;
procedure b_simpanClick(Sender: TObject);
procedure b_closeClick(Sender: TObject);
procedure b_editClick(Sender: TObject);
procedure btn_batalClick(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure grid_pasienDbClick(Sender: TObject);
procedure RzBitBtn1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  F_PENDAFTARAN_PASIEN_BARU: TF_PENDAFTARAN_PASIEN_BARU;

```

implementation

```

uses Unit2, U_edit_pasien, U_pasien_hapus, U_menu_utama, U_cetak_kartu;

```

```

{$R *.dfm}

```

```

procedure TF_PENDAFTARAN_PASIEN_BARU.b_simpanClick(Sender: TObject);
begin
  e_norm.Enabled := true;
  with dm.sp_pasien_insert do
  begin
    params[2].AsString := e_norm.Text;
    params[3].AsString := ed_nama.Text;
    params[4].AsString := ed_tmpt_lahir.Text;
    Params[5].AsDate := dte_tgl_lahir.Date;
    params[6].AsString := ed_pendidikan.Text;
    params[7].AsString := cb_gol_darah.text;
    params[8].AsString := cb_jenis_kelamin.text;
    params[9].AsString := cb_agama.text;
    params[10].AsString := cb_status_kawin.text;
    params[11].AsString := ed_pekerjaan.Text;
    params[12].AsString := mem_alamat.Text;
    params[13].AsString := ed_kota.Text;
    params[14].AsString := ed_provinsi.Text;
    params[15].AsString := ed_kd_pos.Text;
    params[16].AsString := ed_telpon.text;
    params[17].AsString := ed_no_ktp.Text;
    params[18].AsString := ed_status_alergi.Text;
    params[19].AsString := ed_nama_ayah.Text;
    params[20].AsString := ed_nama_ibu.Text;
    ExecProc;
    ShowMessage(params[0].asstring);
    if params[1].AsInteger = 1 then
    begin
      dm.sp_pasien_tampil.Refresh;
      if f_cetak_kartu = nil then
        f_cetak_kartu := Tf_cetak_kartu.Create(self);
      f_cetak_kartu.Visible := false;
      with dm.sp_cetak_kartu do
      begin
        params[0].AsString := e_norm.Text;
        ExecProc;
        f_cetak_kartu.rep_kartu.Preview;
      end;
      btn_batalClick(self);
    end
  end
end

```

```

        else
        e_norm.SetFocus;
        end

end;

procedure TF_PENDAFTARAN_PASIEN_BARU.b_closeClick(Sender: TObject);
begin
    CLOSE;
end;

procedure TF_PENDAFTARAN_PASIEN_BARU.b_editClick(Sender: TObject);
begin
    //f_edit_pasien.ShowModal;
    //f_edit_pasien.ed_norm.Text := dm.T_pasienNORM.Text;
    with dm.sp_pasien_tampil do
    begin
        e_norm.Text := grid_pasien.Fields[0].AsString;
        ed_nama.Text := grid_pasien.Fields[1].AsString;
        ed_tmpt_lahir.Text := grid_pasien.Fields[2].AsString;
        dte_tgl_lahir.Text := grid_pasien.Fields[3].AsString;
        cb_gol_darah.Text := grid_pasien.Fields[5].AsString;
        cb_jenis_kelamin.Text := grid_pasien.Fields[6].AsString;
        cb_agama.Text := grid_pasien.Fields[7].AsString;
        ed_pekerjaan.Text := grid_pasien.Fields[9].AsString;
        ed_pendidikan.Text := grid_pasien.Fields[4].AsString;
        ed_no_ktp.Text := grid_pasien.Fields[15].AsString;
        cb_status_kawin.Text := grid_pasien.Fields[8].AsString;
        mem_alamat.Text := grid_pasien.Fields[10].AsString;
        ed_kota.Text := grid_pasien.Fields[11].AsString;
        ed_kd_pos.Text := grid_pasien.Fields[13].AsString;
        ed_provinsi.Text := grid_pasien.Fields[12].AsString;
        ed_telpon.Text := grid_pasien.Fields[14].AsString;
        ed_status_alergi.Text := grid_pasien.Fields[16].AsString;
        ed_nama_ayah.Text := grid_pasien.Fields[17].AsString;
        ed_nama_ibu.Text := grid_pasien.Fields[18].AsString;
        ExecProc;
    end

end;

procedure TF_PENDAFTARAN_PASIEN_BARU.btn_batalClick(Sender: TObject);
begin
    e_norm.Clear;
    ed_nama.Clear;

```

```

        ed_tmpt_lahir.Clear;
        dte_tgl_lahir.Clear;
        cb_gol_darah.ItemIndex := -1;
        cb_jenis_kelamin.ItemIndex := -1;
        cb_agama.ItemIndex := -1;
        ed_pekerjaan.Clear;
        ed_pendidikan.Clear;
        ed_no_ktp.Clear;
        cb_status_kawin.ItemIndex := -1;
        mem_alamat.Clear;
        ed_kota.Clear;
        ed_provinsi.Clear;
        ed_kd_pos.Clear;
        ed_telpon.Clear;
        ed_status_alergi.Clear;
        ed_nama_ayah.Clear;
        ed_nama_ibu.Clear;
    end;
    procedure TF_PENDAFTARAN_PASIEN_BARU.FormShow(Sender: TObject);
    begin
        dm.sp_pasien_tampil.ExecProc;
        btn_batalClick(self);
        dm.sp_pasien_tampil.first;
        if f_menu_utama.ToolButton5.Caption = '0' then
        begin
            b_edit.Enabled := true;
        end
        else
            b_edit.Enabled := false;
    end;

    procedure TF_PENDAFTARAN_PASIEN_BARU.FormClose(Sender: TObject;
    var Action: TCloseAction);
    begin
        Action := caFree;
        F_PENDAFTARAN_PASIEN_BARU := nil;
    end;

    procedure TF_PENDAFTARAN_PASIEN_BARU.grid_pasienDbClick(Sender:
    TObject);
    begin
        with dm.sp_pasien_tampil do
        begin
            e_norm.Text := grid_pasien.Fields[0].AsString;

```



```

ed_nama.Text := grid_pasien.Fields[1].AsString;
ed_tmpt_lahir.Text := grid_pasien.Fields[2].AsString;
dte_tgl_lahir.Text := grid_pasien.Fields[3].AsString;
cb_gol_darah.Text := grid_pasien.Fields[5].AsString;
cb_jenis_kelamin.Text := grid_pasien.Fields[6].AsString;
cb_agama.Text := grid_pasien.Fields[7].AsString;
ed_pekerjaan.Text := grid_pasien.Fields[9].AsString;
ed_pendidikan.Text := grid_pasien.Fields[4].AsString;
ed_no_ktp.Text := grid_pasien.Fields[15].AsString;
cb_status_kawin.Text := grid_pasien.Fields[8].AsString;
mem_alamat.Text := grid_pasien.Fields[10].AsString;
ed_kota.Text := grid_pasien.Fields[11].AsString;
ed_kd_pos.Text := grid_pasien.Fields[13].AsString;
ed_provinsi.Text := grid_pasien.Fields[12].AsString;
ed_telpon.Text := grid_pasien.Fields[14].AsString;
ed_status_alergi.Text := grid_pasien.Fields[16].AsString;
ed_nama_ayah.Text := grid_pasien.Fields[17].AsString;
ed_nama_ibu.Text := grid_pasien.Fields[18].AsString;
ExecProc;
if f_cetak_kartu = nil then
  f_cetak_kartu := Tf_cetak_kartu.Create(self);
  f_cetak_kartu.Visible := false;
  with dm.sp_cetak_kartu do
    begin
      params[0].AsString := e_norm.Text;
      ExecProc;
      f_cetak_kartu.rep_kartu.Preview;
    end;
end;
btn_batalClick(self);
end;

procedure TF_PENDAFTARAN_PASIEN_BARU.RzBitBtn1Click(Sender: TObject);
begin
  if f_cetak_kartu = nil then
    f_cetak_kartu := Tf_cetak_kartu.Create(self);
    f_cetak_kartu.Visible := false;
    with dm.sp_cetak_kartu do
      begin
        params[0].AsString := e_norm.Text;
        ExecProc;
        f_cetak_kartu.rep_kartu.Preview;
      end
    end;
end;

```

```
end.
```

```
unit U_daftar_igd;
```

```
Interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, RzButton, StdCtrls, DBCtrls, RzDBEdit, jpeg, ExtCtrls, Grids,
DBGrids, RzDBGrid, RzEdit, RzBtnEdt, DB, DBAccess, MyAccess, RzLabel,
Mask, RzPanel, MemDS, RzRadChk, RzRadGrp;
```

```
type
```

```

Tf_daftar_igd = class(TForm)
  RzPanel1: TRzPanel;
  dte_tgl_daftar: TRzDateTimeEdit;
  RzLabel9: TRzLabel;
  RzLabel1: TRzLabel;
  RzLabel2: TRzLabel;
  ed_noreg_igd: TRzMaskEdit;
  ds_pasien_igd_tampil: TMyDataSource;
  ds_sp_igd_insert: TMyDataSource;
  ed_norm: TRzButtonEdit;
  ed_cara_masuk: TRzEdit;
  RzLabel3: TRzLabel;
  ed_cara_bayar: TRzButtonEdit;
  RzLabel7: TRzLabel;
  Label8: TLabel;
  grid_igd: TRzDBGrid;
  GroupBox1: TGroupBox;
  Image2: TImage;
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  Label6: TLabel;
  Label7: TLabel;
  DBEdit1: TDBEdit;
  DBEdit2: TDBEdit;
  DBEdit4: TDBEdit;
  DBEdit5: TDBEdit;
  DBEdit6: TDBEdit;
  DBEdit7: TDBEdit;
  RzDBMemo1: TRzDBMemo;
end;

```

```

bt_daftar_rajal: TRzBitBtn;
bt_pas_baru: TRzBitBtn;
bt_batal: TRzBitBtn;
bt_cancel: TRzBitBtn;
sp_AutoNumber: TMyStoredProc;
Image1: TImage;
RzRadioGroup1: TRzRadioGroup;
rad_tgl: TRzRadioButton;
rad_semua: TRzRadioButton;
ds_sp_igd_tampil2: TMyDataSource;
procedure bt_cancelClick(Sender: TObject);
procedure bt_batalClick(Sender: TObject);
procedure bt_pas_baruClick(Sender: TObject);
procedure bt_daftar_rajalClick(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure ed_normChange(Sender: TObject);
procedure ed_normButtonClick(Sender: TObject);
procedure ed_cara_bayarButtonClick(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure dte_tgl_daftarChange(Sender: TObject);
procedure grid_igdDbiClick(Sender: TObject);
procedure rad_semuaClick(Sender: TObject);
procedure rad_tglClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  f_daftar_igd: Tf_daftar_igd;

implementation

uses Unit2, U_reg_igd, Unit1, U_nom_pasienpas, U_cara_bayar;

{$R *.dfm}

procedure Tf_daftar_igd.bt_cancelClick(Sender: TObject);
begin
  close;
end;

procedure Tf_daftar_igd.bt_batalClick(Sender: TObject);
begin
  ed_cara_bayar.Clear;

```

```

ed_norm.Clear;
ed_cara_masuk.Clear;
dte_tgl_daftarChange(self);
end;

```

```

procedure Tf_daftar_igd.bt_pas_baruClick(Sender: TObject);
begin
  if F_PENDAFTARAN_PASIEN_BARU = nil then
    F_PENDAFTARAN_PASIEN_BARU :=
TF_PENDAFTARAN_PASIEN_BARU.Create(Self);
    F_PENDAFTARAN_PASIEN_BARU.Show;
end;

```

```

procedure Tf_daftar_igd.bt_daftar_rajalClick(Sender: TObject);
begin
  try
    with dm.sp_igd_insert do
      begin
        Params[2].AsString := ed_noreg_igd.text;
        params[3].AsString := ed_norm.Text;
        params[4].AsString := ed_cara_bayar.text;
        params[5].AsDate := dte_tgl_daftar.Date;
        params[6].AsString := ed_cara_masuk.Text;
        ExecProc;
        ShowMessage(params[0].asstring);
        if params[1].AsInteger = 1 then
          begin
            dm.sp_igd_tampil.Refresh;
            bt_batalClick(self);
          end
        else
          ed_norm.SetFocus;

```

```

      end
    except
      MessageDlg('Data Tidak Boleh Kosong!!!',
mtWarning,[mbOK],0);
    end;
end;

```

```

procedure Tf_daftar_igd.FormShow(Sender: TObject);
begin
  dm.T_pasien.Active := false;
  dm.T_pasien.Active := true;
  dm.sp_igd_tampil.ExecProc;
  dm.sp_igd_tampil2.ExecProc;

```

```

dte_tgl_daftar.Date := now;
dte_tgl_daftarChange(self);
ed_normChange(nil);
rad_tgl.Checked := true;
rad_semua.Checked := false;
grid_igd.DataSource := ds_pasien_igd_tampil;
end;

procedure Tf_daftar_igd.ed_normChange(Sender: TObject);
begin
  IF dm.T_pasien.Locate('NORM', ed_norm.Text, [loCaseInsensitive]) then
    dm.ds_pasien.DataSet:=dm.t_pasien
  else
    dm.ds_pasien.DataSet:=nil;
end;

procedure Tf_daftar_igd.ed_normButtonClick(Sender: TObject);
begin
  try
    f_norm_pasien := Tf_norm_pasien.Create(Self);
    if f_norm_pasien.ShowModal = mrOK then
      ed_norm.Text := f_norm_pasien.grid_pasien.Fields[0].AsString;
    except
      end
    end;
end;

procedure Tf_daftar_igd.ed_cara_bayarButtonClick(Sender: TObject);
begin
  try
    f_cara_bayar := Tf_cara_bayar.Create(Self);
    if f_cara_bayar.ShowModal = mrOK then
      ed_cara_bayar.Text :=
f_cara_bayar.grid_cara_bayar_search.Fields[0].AsString;
  Except
  end;
end;

procedure Tf_daftar_igd.FormClose(Sender: TObject);
var Action: TCloseAction);
begin
  Action := caFree;
  f_daftar_igd := nil;
end;

```

```

procedure Tf_daftar_igd.dte_tgl_daftarChange(Sender: TObject);
begin
  with sp_AutoNumber do
    begin
      Params[0].AsString := FormatDateTime('dd/mm/yy-',dte_tgl_daftar.Date);
      Params[1].AsInteger := 3;
      Params[2].AsString := 'igd';
      Params[3].AsString := 'noreg_igd';
      ExecProc;
    end;
  ed_noreg_igd.Text := sp_AutoNumber.Fields[0].AsString;
  with dm.sp_igd_tampil do
    begin
      params[0].AsDate := dte_tgl_daftar.Date;
      ExecProc;
    end
  end;
end;

procedure Tf_daftar_igd.grid_igdDbClick(Sender: TObject);
begin
  if MessageDlg('anda yakin mau mengedit data?',
mtConfirmation,[mbYes,mbNo],0)= mrYes then
  begin
    with dm.sp_igd_tampil do
      begin
        params[0].AsDate := dte_tgl_daftar.Date;
        dte_tgl_daftar.Date := grid_igd.Fields[0].AsDateTime;
        ed_noreg_igd.Text := grid_igd.Fields[1].AsString;
        ed_norm.Text := grid_igd.Fields[2].AsString;
        ed_cara_masuk.Text := grid_igd.Fields[4].AsString;
        ed_cara_bayar.Text := grid_igd.Fields[5].AsString;
        ExecProc;
      end
    end
  end;
end;

procedure Tf_daftar_igd.rad_semuaClick(Sender: TObject);
begin
  if rad_semua.Checked = true then
    begin
      dm.sp_igd_tampil2.ExecProc;
      rad_tgl.Checked := false;
      grid_igd.DataSource := ds_sp_igd_tampil2;
    end
  end;
end;

```

```

end;

procedure Tf_daftar_igd.rad_tglClick(Sender: TObject);
begin
    if rad_tgl.Checked = true then
        begin
            dm.sp_igd_tampil.ExecProc;
            rad_semua.Checked := false;
            grid_igd.DataSource := ds_pasien_igd_tampil;
        end
    end;
end.

unit U_pemeriksaan_igd;

interface

uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, Grids, DBGrids, RzDBGrid, RzButton, DBCtrls, jpeg, ExtCtrls,
    StdCtrls, RzEdit, Mask, RzBtnEdt, RzLabel, RzPanel, RzDBEdit, DB,
    DBAccess, MyAccess, MemDS, RzCmbxBx;

type
    Tf_pemeriksaan_igd = class(TForm)
        RzPanel1: TRzPanel;
        RzLabel1: TRzLabel;
        RzLabel2: TRzLabel;
        RzLabel3: TRzLabel;
        RzLabel4: TRzLabel;
        ed_id_dokter: TRzButtonEdit;
        ed_no_reg_igd: TRzButtonEdit;
        ed_no_periksa: TRzEdit;
        dte_tgl_periksa: TRzDateTimeEdit;
        GroupBox1: TGroupBox;
        Image2: TImage;
        Label1: TLabel;
        Label2: TLabel;
        Label3: TLabel;
        dbed_id_dokter: TDBEdit;
        dbed_nama_dokter: TDBEdit;
        dbed_spesialist: TDBEdit;
        bt_close: TRzBitBtn;
        grid_diagnosa: TRzDBGrid;

```

```

        RzLabel9: TRzLabel;
        ed_kd_diagnosa: TRzButtonEdit;
        RzLabel6: TRzLabel;
        bt_cancel: TRzBitBtn;
        bt_save_diagnosa: TRzBitBtn;
        Image1: TImage;
        ds_dokter: TMyDataSource;
        ds_pemeriksaan_insert: TMyDataSource;
        ds_pemeriksaan_igd_tampil: TMyDataSource;
        sp_AutoNumber: TMyStoredProc;
        cb_keadaan_keluar: TRzComboBox;
        RzLabel5: TRzLabel;
        procedure bt_closeClick(Sender: TObject);
        procedure bt_cancelClick(Sender: TObject);
        procedure FormShow(Sender: TObject);
        procedure bt_save_diagnosaClick(Sender: TObject);
        procedure ed_no_reg_igdButtonClick(Sender: TObject);
        procedure ed_id_dokterButtonClick(Sender: TObject);
        procedure ed_kd_diagnosaButtonClick(Sender: TObject);
        procedure FormClose(Sender: TObject; var Action: TCloseAction);
        procedure ed_id_dokterChange(Sender: TObject);
        procedure ed_no_reg_igdChange(Sender: TObject);
        procedure grid_diagnosaDbiClick(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    f_pemeriksaan_igd: Tf_pemeriksaan_igd;

implementation

uses Unit2, U_daftar_igd, U_reg_igd, U_lihat_dokter, U_diagnosa,
    U_rujuk_igd;

{$R *.dfm}

procedure Tf_pemeriksaan_igd.bt_closeClick(Sender: TObject);
begin
    close;
end;

procedure Tf_pemeriksaan_igd.bt_cancelClick(Sender: TObject);
begin

```

```

ed_no_periksa.Clear;
ed_no_reg_igd.Clear;
ed_id_dokter.Clear;
ed_kd_diagnosa.Clear;
ed_no_periksa.Clear;
cb_keadaan_keluar.ItemIndex := 0;
dte_tgl_periksa.Date := now;
end;

procedure Tf_pemeriksaan_igd.FormShow(Sender: TObject);
begin
dm.t_dokter.Active := false;
dm.t_dokter.Active := true;
dte_tgl_periksa.Date := now;
dm.sp_pemeriksaan_igd_tampil.ExecProc;
ed_id_dokterChange(nil);
bt_cancelClick(self);
end;

procedure Tf_pemeriksaan_igd.bt_save_diagnosaClick(Sender: TObject);
begin
try
with DM.sp_pemeriksaan_igd_insert do
begin
Params[2].AsString := ed_no_periksa.Text;
Params[3].AsString := ed_no_reg_igd.Text;
params[4].AsString := ed_kd_diagnosa.Text;
params[5].AsString := ed_id_dokter.Text;
params[6].AsDateTime := dte_tgl_periksa.Date;
Params[7].AsString := cb_keadaan_keluar.Text;
ExecProc;
ShowMessage(params[0].asString);
if params[1].AsInteger = 1 then
begin
dm.sp_pemeriksaan_igd_tampil.Refresh;
IF cb_keadaan_keluar.Text = 'DIRUJUK' THEN
BEGIN
IF f_rujukan_igd = NIL THEN
f_rujukan_igd := Tf_rujukan_igd.Create(self);
f_rujukan_igd.ed_noreg_igd.Text := ed_no_reg_igd.Text;
END;
bt_cancelClick(self);
end
else
ed_no_periksa.SetFocus;

```

```

end
except
MessageDlg('Data Tidak Boleh Kosong!!!',
mtWarning,[mbOK],0);
end;
end;

procedure Tf_pemeriksaan_igd.ed_no_reg_igdButtonClick(Sender: TObject);
begin
try
f_reg_igd := Tf_reg_igd.Create(Self);
if f_reg_igd.ShowModal = mrOK then
ed_no_reg_igd.Text := f_reg_igd.grid_reg.Fields[0].AsString;
Except
end;
end;

procedure Tf_pemeriksaan_igd.ed_id_dokterButtonClick(Sender: TObject);
begin
try
f_lihat_dokter := Tf_lihat_dokter.Create(Self);
if f_lihat_dokter.ShowModal = mrOK then
ed_id_dokter.Text :=
f_lihat_dokter.grid_dokter_search.Fields[0].AsString;
except
end;
end;

procedure Tf_pemeriksaan_igd.ed_kd_diagnosaButtonClick(Sender: TObject);
begin
try
F_lihat_diagnosa := TF_lihat_diagnosa.Create(Self);
if F_lihat_diagnosa.ShowModal = mrOK then
ed_kd_diagnosa.Text :=
F_lihat_diagnosa.grid_diagnosa_search.Fields[0].AsString;
except
end;
end;

procedure Tf_pemeriksaan_igd.FormClose(Sender: TObject;
var Action: TCloseAction);
begin
Action := caFree;
f_pemeriksaan_igd := nil;
end;

```

```

procedure Tf_pemeriksaan_igd.ed_id_dokterChange(Sender: TObject);
begin
  IF dm.t_dokter.Locate('id_dokter', ed_id_dokter.Text, [loCaseInsensitive]) then
    ds_dokter.DataSet:=dm.t_dokter
  else
    ds_dokter.DataSet:=nil;
end;

```

```

procedure Tf_pemeriksaan_igd.ed_no_reg_igdChange(Sender: TObject);
begin
  with sp_AutoNumber do
  begin
    Params[0].AsString := ed_no_reg_igd.Text+'-';
    Params[1].AsInteger := 2;
    Params[2].AsString := 'pemeriksaan_igd';
    Params[3].AsString := 'no_pemeriksaan';
    ExecProc;
  end;
  ed_no_periksa.Text := sp_AutoNumber.Fields[0].AsString;

```

```

With dm.sp_pemeriksaan_igd_tampil do
begin
  params[0].AsString := ed_no_reg_igd.Text;
  ExecProc;
end;

if ed_no_reg_igd.Text = '' then
begin
  bt_cancelClick(self);
end

```

```
end;
```

```

procedure Tf_pemeriksaan_igd.grid_diagnosaDbClick(Sender: TObject);
begin
  if MessageDlg('anda yakin mau mengedit data?',
  mtConfirmation,[mbYes,mbNo],0)= mrYes then
  begin
    with dm.sp_pemeriksaan_igd_tampil do
    begin
      ed_no_periksa.Text := grid_diagnosa.Fields[0].AsString;
      ed_no_reg_igd.Text := grid_diagnosa.Fields[1].AsString;
      ed_kd_diagnosa.Text := grid_diagnosa.Fields[2].AsString;
      ed_id_dokter.Text := grid_diagnosa.Fields[3].AsString;
      dte_tgl_periksa.date := grid_diagnosa.Fields[4].AsDateTime;

```

```

      ed_no_periksa.Text := grid_diagnosa.Fields[0].AsString;
      ExecProc;
    end
  end;
end;

end.

```

```
unit U_tindakan_igd;
```

```
interface
```

```
uses
```

```

  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, RzButton, Grids, DBGrids, RzDBGrid, RzEdit, RzBtnEdt, RzLabel,
  StdCtrls, DBCtrls, RzDBEdit, Mask, jpeg, ExtCtrls, RzPanel, DB, DBAccess,
  MyAccess, MemDS;

```

```
type
```

```

Tf_tindakan_igd = class(TForm)
  RzPanel1: TRzPanel;
  ds_sp_tindakan_rajal_tampil: TMyDataSource;
  ds_tindakan_rajal_insert: TMyDataSource;
  ds_tindakan_dtl: TMyDataSource;
  RzGroupBox1: TRzGroupBox;
  Image2: TImage;
  Label1: TLabel;
  Label2: TLabel;
  db_ed_kd_tindakan: TDBEdit;
  mem_tindakan: TRzDBMemo;
  Image1: TImage;
  RzLabel3: TRzLabel;
  grid_tindakan_igd: TRzDBGrid;
  bt_save_tindakan: TRzBitBtn;
  bt_batal: TRzBitBtn;
  bt_close: TRzBitBtn;
  sp_AutoNumber: TMyStoredProc;
  RzLabel1: TRzLabel;
  dte_tgl_tindakan: TRzDateTimeEdit;
  RzLabel4: TRzLabel;
  ed_no_tindakan: TRzEdit;
  RzLabel6: TRzLabel;
  ed_noreg_igd: TRzButtonEdit;
  RzLabel2: TRzLabel;
  ed_kd_tindakan: TRzButtonEdit;

```

```

RzLabel5: TRzLabel;
procedure bt_batalClick(Sender: TObject);
procedure bt_save_tindakanClick(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure ed_kd_tindakanChange(Sender: TObject);
procedure bt_closeClick(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure ed_kd_tindakanButtonClick(Sender: TObject);
procedure ed_noreg_igdButtonClick(Sender: TObject);
procedure ed_noreg_igdChange(Sender: TObject);
procedure grid_tindakan_igdDbClick(Sender: TObject);

```

```

private
  { Private declarations }
public
  { Public declarations }
end;

```

```

var
  f_tindakan_igd: Tf_tindakan_igd;

```

```

implementation

```

```

uses Unit2, U_tindakan_dtl, U_reg_igd;

```

```

{$R *.dfm}

```

```

procedure Tf_tindakan_igd.bt_batalClick(Sender: TObject);
begin
  ed_no_tindakan.Clear;
  ed_kd_tindakan.Clear;
  ed_noreg_igd.Clear;
  ed_no_tindakan.Clear;
end;

```

```

procedure Tf_tindakan_igd.bt_save_tindakanClick(Sender: TObject);
begin
  try
    with dm.sp_tindakan_igd_insert do
      begin
        params[2].AsString := ed_no_tindakan.Text;
        params[3].AsString := ed_kd_tindakan.Text;
        params[4].AsString := ed_noreg_igd.Text;
        params[5].AsDateTime := dte_tgl_tindakan.Date;
        ExecProc;
        ShowMessage(params[0].asString);
        if params[1].AsInteger = 1 then

```

```

begin
  dm.sp_tindakan_igd_tampil.Refresh;
  bt_batalClick(self);
end
else
  ed_no_tindakan.SetFocus;
end
except
  MessageDlg('Data Tidak Boleh Kosong!!!',
    mtWarning,[mbOK],0);
end;
end;
end;

```

```

procedure Tf_tindakan_igd.FormShow(Sender: TObject);
begin
  dm.sp_tindakan_igd_tampil.ExecProc;
  dm.sp_tindakan_dtl_tampil.ExecProc;
  dte_tgl_tindakan.Date := date;
  ed_kd_tindakanChange(nil);
end;

```

```

procedure Tf_tindakan_igd.ed_kd_tindakanChange(Sender: TObject);
begin
  if dm.sp_tindakan_dtl_tampil.Locate('kd_tindakan',ed_kd_tindakan.Text,
    [loCaseInsensitive]) then
    ds_tindakan_dtl.DataSet := dm.sp_tindakan_dtl_tampil
  else
    ds_tindakan_dtl.DataSet := nil;
end;

```

```

procedure Tf_tindakan_igd.bt_closeClick(Sender: TObject);
begin
  close;
end;

```

```

procedure Tf_tindakan_igd.FormClose(Sender: TObject;
  var Action: TCloseAction);
begin
  Action := caFree;
  f_tindakan_igd := nil;
end;

```

```

procedure Tf_tindakan_igd.ed_kd_tindakanButtonClick(Sender: TObject);
begin
  try
    f_tindakan_dtl := Tf_tindakan_dtl.Create(Self);

```

```

        if f_tindakan_dtl.ShowModal = mrOK then
            ed_kd_tindakan.Text :=
f_tindakan_dtl.grid_tindakan_search.Fields[0].AsString;
        except
            end;
        end;

procedure Tf_tindakan_igd.ed_noreg_igdButtonClick(Sender: TObject);
begin
    try
        f_reg_igd := Tf_reg_igd.Create(Self);
        if f_reg_igd.ShowModal = mrOK then
            ed_noreg_igd.Text := f_reg_igd.grid_reg.Fields[0].AsString;
        except
            end;
    end;

procedure Tf_tindakan_igd.ed_noreg_igdChange(Sender: TObject);
begin
    with sp_AutoNumber do
        begin
            Params[0].AsString := ed_noreg_igd.Text+'-';
            Params[1].AsInteger := 2;
            Params[2].AsString := 'tindakan_igd';
            Params[3].AsString := 'no_tindakan';
            ExecProc;
        end;
        ed_no_tindakan.Text := sp_AutoNumber.Fields[0].AsString;

    with dm.sp_tindakan_igd_tampil do
        begin
            params[0].AsString := ed_noreg_igd.Text;
            ExecProc;
        end;

    if ed_noreg_igd.Text = '' then
        bt_batalClick(self);
    end;

procedure Tf_tindakan_igd.grid_tindakan_igdDbClick(Sender: TObject);
begin
    if MessageDlg('anda yakin mau mengedit data?',
mtConfirmation,[mbYes,mbNo],0)= mrYes then
        begin
            with dm.sp_tindakan_igd_tampil do
                begin

```

```

                    ed_no_tindakan.Text := grid_tindakan_igd.Fields[1].AsString;
                    ed_noreg_igd.Text := grid_tindakan_igd.Fields[2].AsString;
                    ed_kd_tindakan.Text := grid_tindakan_igd.Fields[5].AsString;
                    dte_tgl_tindakan.Date := grid_tindakan_igd.Fields[0].AsDateTime;
                    ed_no_tindakan.Text := grid_tindakan_igd.Fields[1].AsString;
                end
            end;
        end;

    end.

unit U_rujuk_igd;

interface

uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, RzButton, StdCtrls, RzCmboBx, RzBtnEdt, RzEdit, Grids, DBGrids,
    RzDBGrid, Mask, RzLabel, ExtCtrls, RzPanel, jpeg, DB, DBAccess, MyAccess,
    MemDS;

type
    Tf_rujukan_igd = class(TForm)
        RzLabel11: TRzLabel;
        dte_tgl_rujuk: TRzDateTimeEdit;
        RzLabel12: TRzLabel;
        grid_rujukan_igd: TRzDBGrid;
        ed_no_rujuk: TRzEdit;
        RzLabel5: TRzLabel;
        RzLabel8: TRzLabel;
        ed_noreg_igd: TRzButtonEdit;
        cb_tujuan_rujuk: TRzComboBox;
        RzLabel10: TRzLabel;
        bt_save_rujuk: TRzBitBtn;
        bt_cancel_rujuk: TRzBitBtn;
        bt_close: TRzBitBtn;
        RzPanel1: TRzPanel;
        Image1: TImage;
        ds_rujuk_insert: TMyDataSource;
        ds_rujuk_tampil: TMyDataSource;
        sp_AutoNumber: TMyStoredProc;
        ed_rs: TRzEdit;
        l_rs: TRzLabel;
        procedure bt_closeClick(Sender: TObject);
        procedure bt_cancel_rujukClick(Sender: TObject);
        procedure bt_save_rujukClick(Sender: TObject);
    end;

```



```

procedure FormShow(Sender: TObject);
procedure ed_noreg_igdButtonClick(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure grid_rujukan_igdDbClick(Sender: TObject);
procedure dte_tgl_rujukChange(Sender: TObject);
procedure cb_tujuan_rujukChange(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

```

```

var
  f_rujukan_igd: Tf_rujukan_igd;

```

```

implementation

```

```

uses Unit2, U_reg_igd, U_daftar_ranap;

```

```

{$R *.dfm}

```

```

procedure Tf_rujukan_igd.bt_closeClick(Sender: TObject);
begin
  Close;
end;

```

```

procedure Tf_rujukan_igd.bt_cancel_rujukClick(Sender: TObject);
begin
  ed_noreg_igd.Clear;
  dte_tgl_rujuk.Date := now;
  cb_tujuan_rujuk.ItemIndex := -1;
end;

```

```

procedure Tf_rujukan_igd.bt_save_rujukClick(Sender: TObject);
begin
  try
    with dm.sp_rujukan_igd_insert do
      begin
        params[2].AsString := ed_no_rujuk.Text;
        params[3].asstring := ed_noreg_igd.Text;
        params[4].AsDateTime := dte_tgl_rujuk.Date;
        params[5].AsString := cb_tujuan_rujuk.Text;
        ExecProc;
        ShowMessage(params[0].asstring);
        if params[1].AsInteger = 1 then
          begin

```

```

dm.sp_rujukan_igd_tampil.Refresh;
if cb_tujuan_rujuk.ItemIndex = 0 then
  begin
    if F_daftar_ranap = nil then
      F_daftar_ranap := TF_daftar_ranap.Create(Self);
    F_daftar_ranap.Show;
    F_daftar_ranap.ed_norm_ranap.Text :=
      dm.sp_rujukan_igd_tampilNORM.AsString;
    bt_cancel_rujukClick(self);
    close;
  end
  else
    if cb_tujuan_rujuk.ItemIndex = 1 then
      begin
        ed_no_rujuk.SetFocus;
        bt_cancel_rujukClick(self);
      end
    else
      ed_no_rujuk.SetFocus;
  end
end
except
  MessageDlg('Data Tidak Boleh Kosong!!!',
    mtWarning,[mbOK],0);
end;
end;

procedure Tf_rujukan_igd.FormShow(Sender: TObject);
begin
  dm.sp_rujukan_igd_tampil.ExecProc;
  dte_tgl_rujuk.Date := date;
  dte_tgl_rujukChange(self);
end;

procedure Tf_rujukan_igd.ed_noreg_igdButtonClick(Sender: TObject);
begin
  try
    f_reg_igd := Tf_reg_igd.Create(Self);
    if f_reg_igd.ShowModal = mrOK then
      ed_noreg_igd.Text := f_reg_igd.grid_reg.Fields[0].AsString;
    except
      end;
  end;

end;

procedure Tf_rujukan_igd.FormClose(Sender: TObject;
  var Action: TCloseAction);

```

```

begin
    Action := caFree;
    f_rujukan_igd := nil;
end;

procedure Tf_rujukan_igd.grid_rujukan_igdDbClick(Sender: TObject);
begin
    if MessageDlg('anda yakin mau mengedit data?',
mtConfirmation,[mbYes,mbNo],0)= mrYes then
begin
    with dm.sp_rujukan_igd_tampil do
begin
    ed_no_rujuk.Text := grid_rujukan_igd.Fields[0].AsString;
    ed_noreg_igd.Text := grid_rujukan_igd.Fields[1].AsString;
    cb_tujuan_rujuk.Text := grid_rujukan_igd.Fields[3].AsString;
    dte_tgl_rujuk.Date := grid_rujukan_igd.Fields[2].AsDateTime;
    ed_no_rujuk.Text := grid_rujukan_igd.Fields[0].AsString;
    ExecProc;
end
end;
end;

procedure Tf_rujukan_igd.dte_tgl_rujukChange(Sender: TObject);
begin
    with sp_AutoNumber do
begin
    Params[0].AsString := FormatDateTime('dd/mm/yy-',dte_tgl_rujuk.Date);
    Params[1].AsInteger := 2;
    Params[2].AsString := 'rujukan_igd';
    Params[3].AsString := 'Id_rujukan';
    ExecProc;
end;
    ed_no_rujuk.Text := sp_AutoNumber.Fields[0].AsString;

end;

procedure Tf_rujukan_igd.cb_tujuan_rujukChange(Sender: TObject);
begin
    if cb_tujuan_rujuk.ItemIndex = 1 then
begin
    ed_rs.Visible := true;
    l_rs.Visible := true
end
else
    ed_rs.Visible := false;
    l_rs.Visible := false;
end;
end;
end;

```