



REPUBLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN CIPTAAN

Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC00202114694, 8 Maret 2021

Pencipta

Nama : **Drs. Faisal Syafar., M.Si., M.InfTech., Ph.D.**
Alamat : Kompleks Tabaria Tower E10/23 Kelurahan Mannuruki, Kecamatan Tamalate, Makassar, SULAWESI SELATAN, 90221
Kewarganegaraan : Indonesia

Pemegang Hak Cipta

Nama : **Drs. Faisal Syafar., M.Si., M.InfTech., Ph.D.**
Alamat : Kompleks Tabaria Tower E10/23 Kelurahan Mannuruki, Kecamatan Tamalate, Makassar, SULAWESI SELATAN, 90221
Kewarganegaraan : Indonesia

Jenis Ciptaan : **Program Komputer**
Judul Ciptaan : **Sistem Informasi Pasien Rawat Jalan**
Tanggal dan tempat diumumkan untuk pertama kali : 19 Maret 2016, di Makassar
di wilayah Indonesia atau di luar wilayah Indonesia
Jangka waktu perlindungan : Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor pencatatan : 000241637

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.
Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL



Dr. Freddy Harris, S.H., LL.M., ACCS.
NIP. 196611181994031001

Disclaimer:

Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, menteri berwenang untuk mencabut surat pencatatan permohonan.

Program Komputer

SISTEM INFORMASI PASIEN RAWAT JALAN

Drs. Faisal Syafar, M.Si., M.InfTech., Ph.D.

MAKASSAR

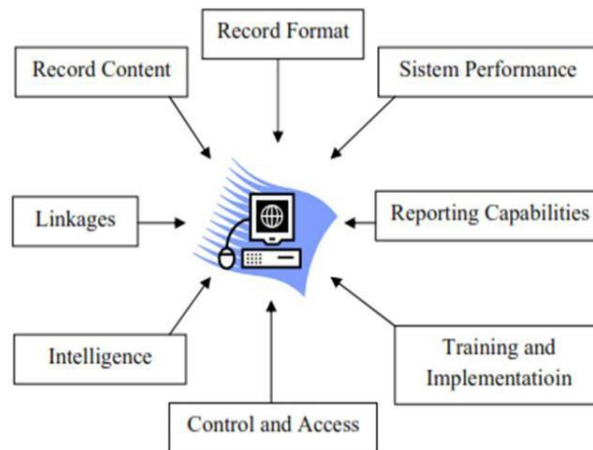
Maret 2016

SISTEM INFORMASI PASIEN RAWAT JALAN

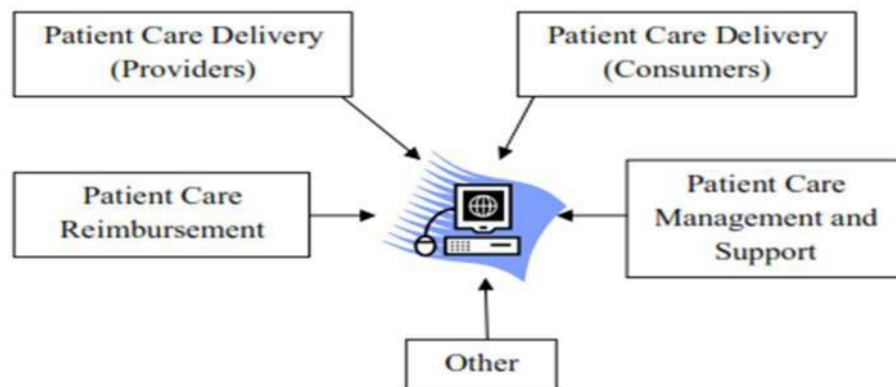
Komponen penting Sistem Informasi Pasien Rawat Jalan (SIPRJ) mengacu pada kebutuhan, sebagai berikut:

- a. *Record Format*, bentuk yang sesuai contoh berbagai pelayanan sesuai kebutuhan.
- b. *Sistem Performance*, seperti pemanggilan kembali, serta mudah dalam perubahan data.
- c. *Reporting Capabilities*, kelengkapan dokumen, mudah untuk dimengerti, dan standar laporan.
- d. *Training and Implementation*, pelatihan yang minimal untuk menggunakan dengan benar.
- e. *Control and Access*, untuk mengakses bagi yang berwenang, tetapi terlindung dari penyalahgunaan.
- f. *Intelligence*, seperti sistem keputusan, sistem tanda bahaya yang sesuai.
- g. *Linkages*, terkait dengan berbagai pelayanan lain, perpustakaan, basis data pasien dan keuangan.

- h. *Record Content*, meliputi standarisasi formulir dan isi, sesuai dengan kode penyakit dan tujuan pelayanan.



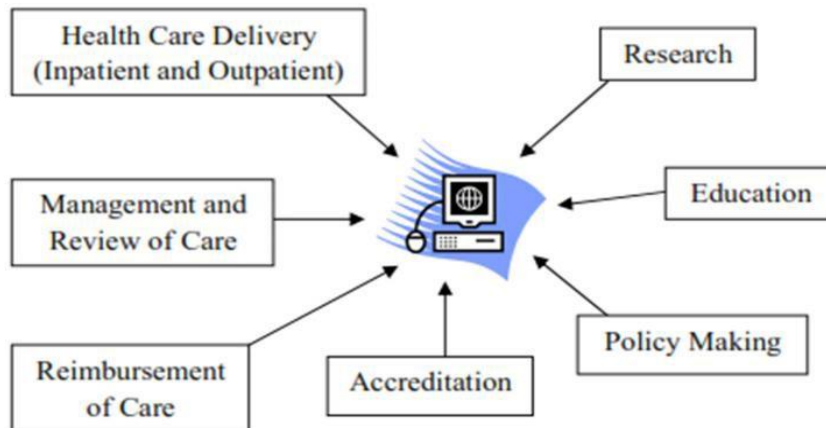
Gambar 1 *User Requirement for Patient Records and Records Sistem*



Gambar 2. *Individual User of Patient Records*

Komponen penting penggunaan rekam medik pada individu adalah sebagai berikut:

- a. *Patient Care Delivery (Consumers)*, untuk pasien keluarga.
- b. *Patient Care Management and Support*, untuk manager mutu, informasi kesehatan, dan manajemen para profesional dan administrasi.
- c. *Other*, seperti akreditasi, kebijakan pemerintah dan penelitian.
- d. *Patient Care Reimbursement*, untuk manager keuangan dan penagihan asuransi.
- e. *Patient Care Delivery (Providers)*, seperti perawat, dokter, dan ahli farmasi.



Gambar 3. *Institutional User of Patient Records*

Rekam medik digunakan oleh berbagai institusi, antara lain yaitu:

- a. *Research*, penelitian penyakit dan kesehatan hukum.
- b. *Education*, untuk pendidikan dokter, perawat dan kesehatan masyarakat.
- c. *Accreditation*, seperti untuk akreditasi, institusi dan para profesional.
- d. *Reimbursement of Care*, untuk kerja sama penagihan antara pusat pelayanan.
- e. *Management and Review of Care*, untuk *Peer Review*, menjaga mutu, dan *Utilization Review* serta manajemen pelayanan.
- f. *Health Care Delivery (Inpatient and Outpatient)*, untuk aliansi pelayanan, jaringan pelayanan, pengembangan administrasi pembebanan.

Di dalam *Microsoft Access* terdapat salah satu komponen dari *basis data* yang disebut dengan tabel. Tabel terdiri atas kolom atau *field* dan baris atau *record*. *Field* merupakan atribut dari tabel, sedangkan *record* merupakan isi dari tabel. Berikut ini adalah sedikit pengenalan tentang *Microsoft Access* :



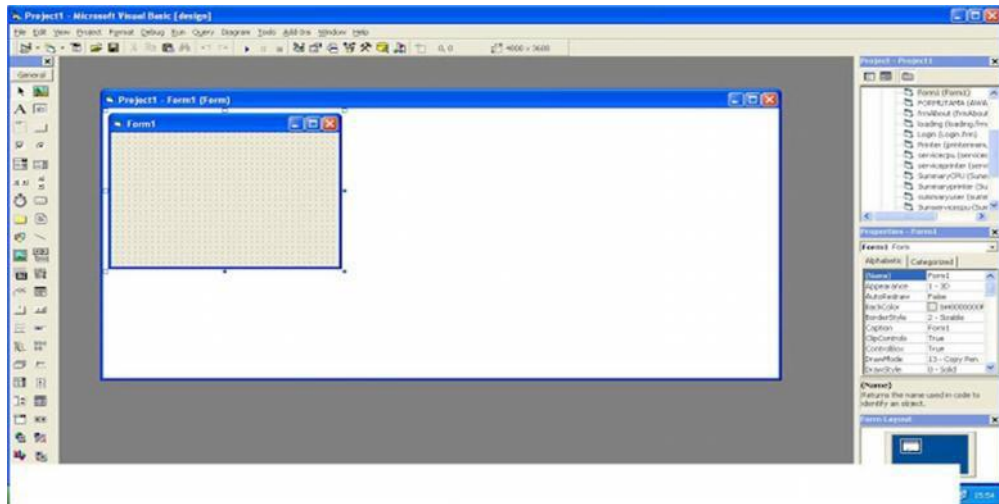
Gambar 4. *Microsoft Access*

1. *Visual Basic 6.0*

Visual Basic 6.0 adalah salah satu aplikasi untuk membuat sistem informasi basis data. *Visual Basic 6.0* ini adalah terusan dari *Visual Basic 1* sampai 5, walau agak tertinggal namun aplikasi *software* ini masih digunakan

untuk pengimplementasiannya di dalam pembuatan sistem basis data. *Visual Basic* ini *support* dengan *operating sistem windows* dan untuk basis datanya bisa menggunakan *microsoft access*, *SQL server* dan *oracle*.

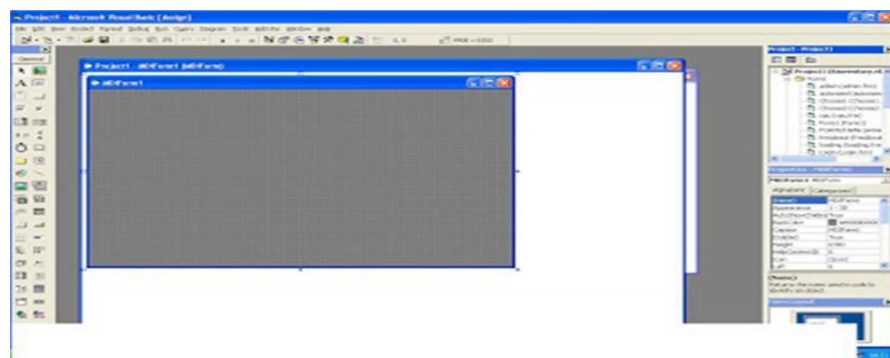
Dalam menu *Visual Basic* terdapat komponen-komponen yang tentu saja diperlukan untuk menentukan *coding* atau *syntax* yang akan digunakan, karena salah titik atau koma dalam *Visual Basic* akan mengakibatkan program sistem *debug* atau *error*. Berikut ini adalah sedikit perkenalan tentang *Visual Basic 6.0*:



Gambar 5. *Visual Basic*

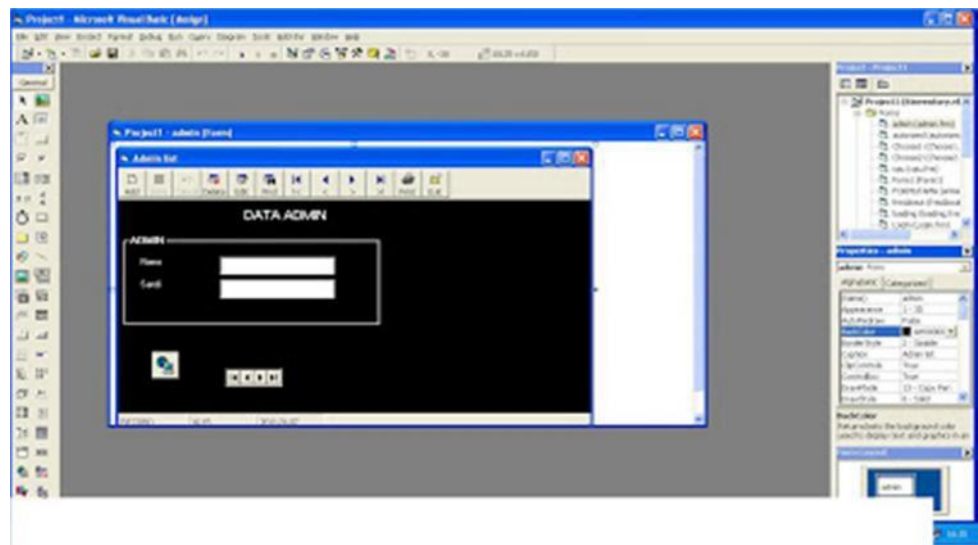
a. Sub Form/MDI Form

MDI form ini adalah form utama dalam *Visual Basic*, karena *MDI form* ini mewakili beberapa formulir yang ada di dalam aplikasi *basis data* yang kita buat. Dalam *MDI form* biasanya hanya terdapat *coding* pemanggilan beberapa formulir yang lain, kita tidak membuat *basis data* di dalamnya karena *MDI form* dirancang/didesain untuk pengatur form-form yang lain.



Gambar 6. Sub Form/MDI Form

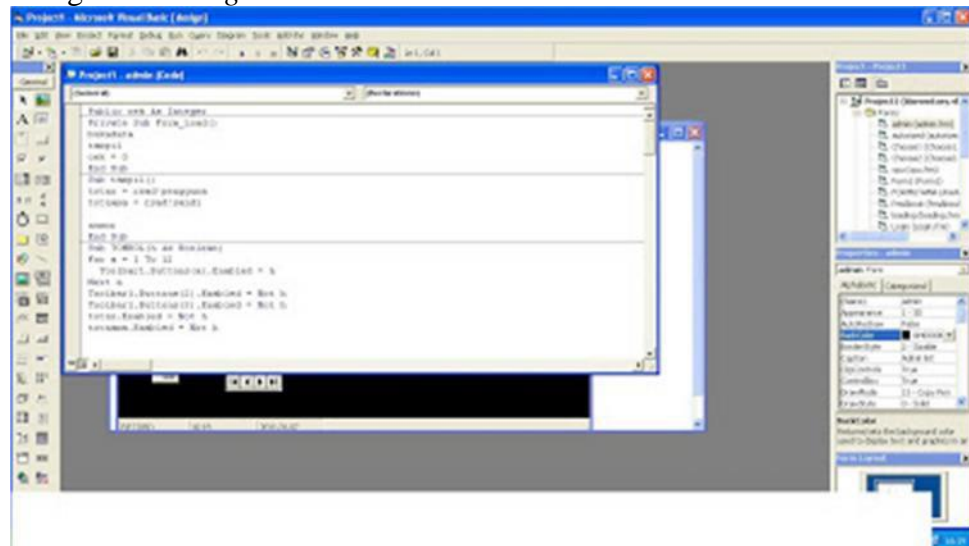
Berikut di bawah ini adalah contoh gambar Form yang telah didesain sesuai kebutuhan:



Gambar 7. Desain Form

Coding atau Syntax

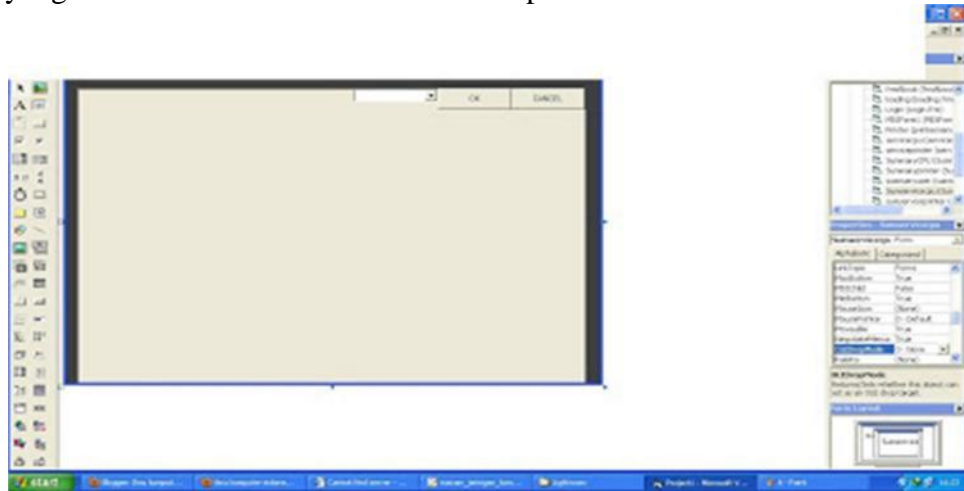
Adalah perintah-perintah dalam bahasa program aplikasi *Visual Basic* yang telah ditentukan, dalam *coding* ini juga yang mempengaruhi sistem dapat berjalan atau tidak. Dalam pembuatan *coding* harus berhati-hati, karena jika salah titik atau koma, akan berakibat program aplikasi yang kita buat mengalami *debug* atau *error*.



Gambar 8. Form Coding

Report

Desain Report juga mempengaruhi hasil *output* yang telah kita eksekusi. Dalam pembuatan *report* harus berhati-hati, karena jika salah maka laporan yang akan dicetak tidak akan sesuai harapan.

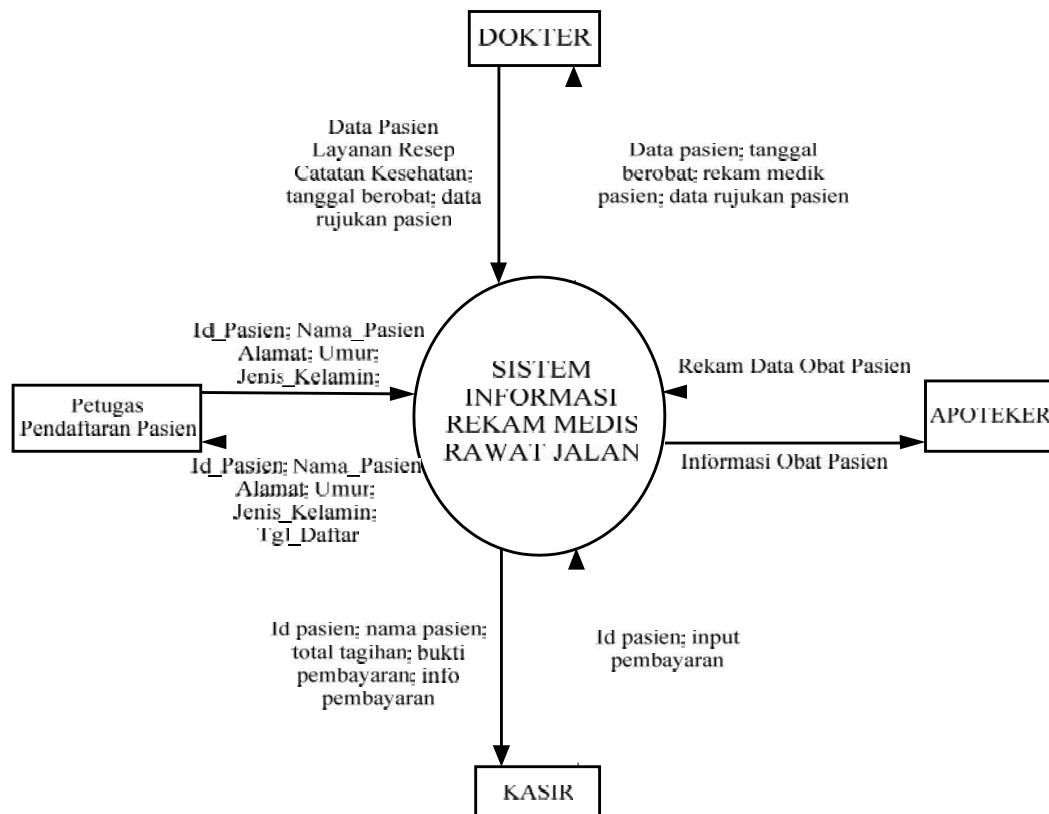


Gambar 9. Report dalam Visual Basic

Persiapan (System Design)

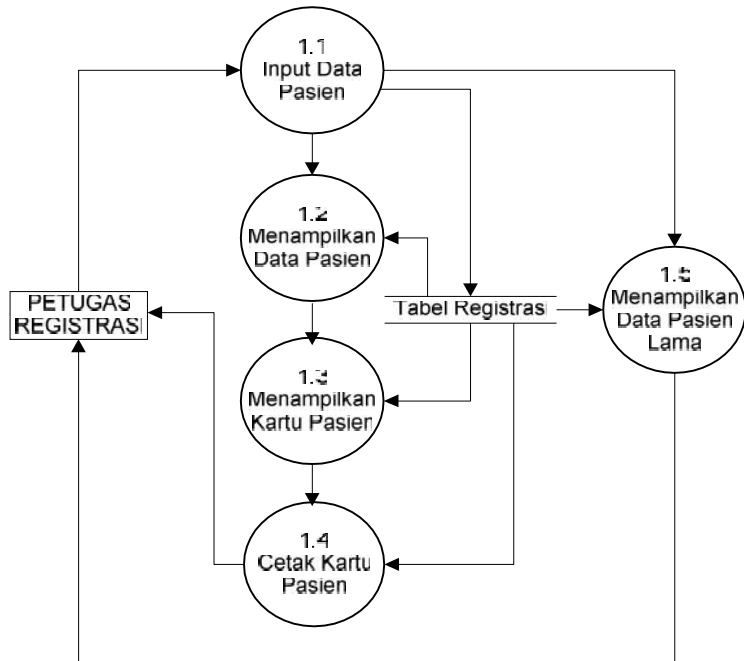
Context Diagram (CD)

Context Diagram (CD) digunakan untuk menampilkan dan menggambarkan hubungan antar entitas yang ada pada Sistem Informasi Rumah Sakit. Sistem Informasi Rumah Sakit ini terdiri atas empat buah entitas, yaitu registrasi pasien, dokter/paramedis, apotek, dan Kasir. Adapun *Context Diagram* (CD)



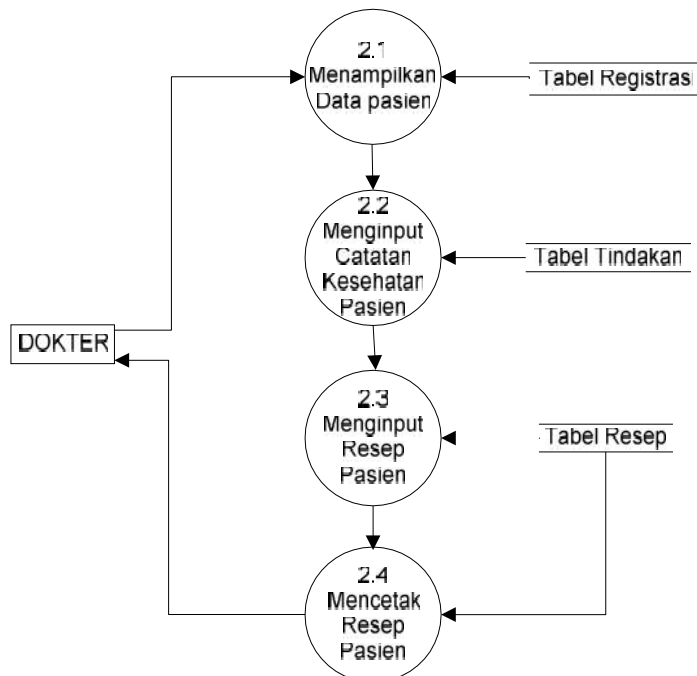
untuk sistem informasi ini adalah sebagai berikut :

Gambar 10 *Context Diagram* (CD) Sistem Informasi Rawat Jalan



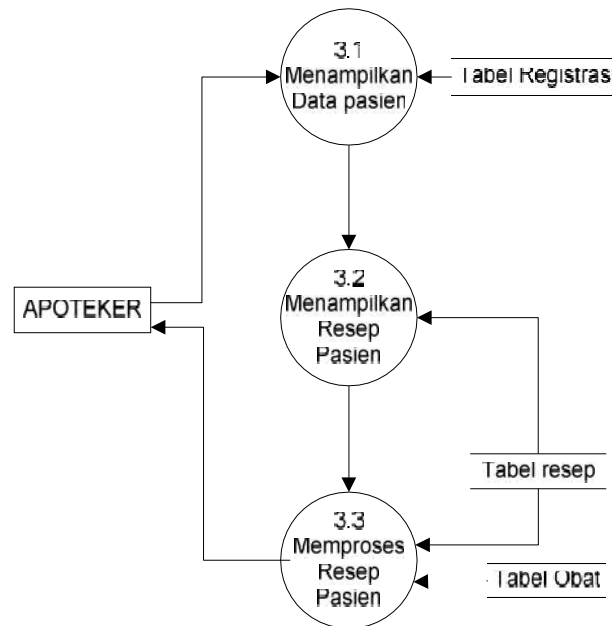
Gambar 12. *Data Flow Diagram (DFD) Registrasi Pasien*

DFD pada entitas Dokter dapat terlihat seperti pada Gambar



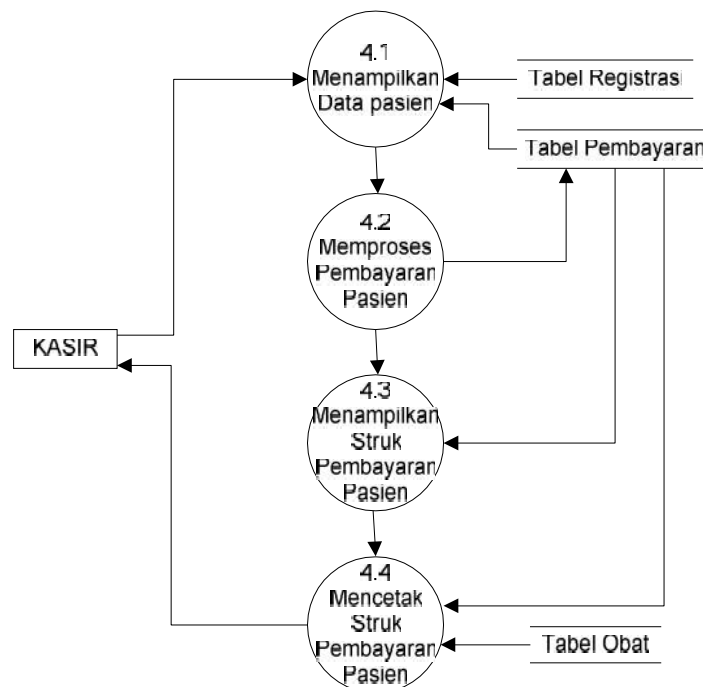
Gambar 13 *Data Flow Diagram (DFD) Dokter*

DFD pada entitas Apoteker dapat terlihat seperti pada Gambar



Gambar 14. Data Flow Diagram (DFD) Apoteker

DFD pada entitas Kasir dapat terlihat seperti pada Gambar



Gambar 15 Data Flow Diagram (DFD) Kasir

Adapun relasi antar basis data adalah sebagai berikut :

Perancangan (*Design*)

Design adalah tahap kedua pada metode perangkat lunak *Modified Waterfall* yang di mana berguna untuk mendefinisikan *Analysis Requirement* pada tahap pertama menjadi bentuk karakteristik yang dapat dimengerti oleh *programmer* sebelum dilakukannya pengkodean. Dalam tahap ini dilakukan perancangan basis data, perancangan antar muka (*user interface*) pemakai, dan perancangan prosedural.

Perancangan *Interface*

Perancangan sistem ini terdapat beberapa *interface* yang dapat digunakan pada *user*, diantaranya:

Login User

Sebelum petugas/*user* masuk ke dalam sistem, *user* harus masuk melalui form untuk *autentifikasi*, yaitu *user* diminta memasukkan *username* dan *password* seperti *form* di bawah ini.

Gambar 17. *Form Login User* Karyawan

Skenario program :

- 1) Jika *user* memasukkan nama pada kolom *User* kemudian menekan tombol *enter* maka dengan otomatis pada kolom *Nama* akan terisi dan kemudian *user* memasukkan *password*, setelah *user* memasukkan *login* dengan benar maka *user* dapat masuk ke dalam sistem/aplikasi.
- 2) Jika *command Edit* atau *Hapus* di klik, maka pencarian data dapat dilakukan dengan mengetik kodenya atau dengan memilih data dalam *grid* kemudian tekan *enter*.

Pendaftaran Pasien

Proses awal transaksi dalam aplikasi ini adalah pendaftaran pasien, baik pasien baru maupun pasien yang telah terdaftar. Untuk itu buatlah *form* seperti gambar di bawah ini.

DATA SOCIAL PASIEN			
Kode	<input type="text"/>	Alamat	<input type="text"/>
Nama	<input type="text"/>		
Tempat	<input type="text"/>	Kota	<input type="text"/>
Tanggal	<input type="text"/>	Kewarganegaraan	<input type="text"/>
Gender	<input type="text"/>	Agama	<input type="text"/>
Status	<input type="text"/>	Pendidikan	<input type="text"/>
Pekerjaan	<input type="text"/>	Telepon	<input type="text"/>
Nama	<input type="text"/>	Hubungan	<input type="text"/>
Alamat	<input type="text"/>	Telepon	<input type="text"/>
		<input type="button" value="Simpan"/>	<input type="button" value="Keluar"/>

Gambar 18. *Form Data Pendaftaran*

- 1) Pasien yang mendaftar ditanya oleh bagian pendaftaran akan menuju ke poli apa atau si pasien sendiri yang menyebutkannya.
- 2) Bagian administrasi menginformasikan dokter yang ada pada saat itu, dan nomor antrian masing-masing dokter.
- 3) Bagian adm menanyakan apakah pasien baru atau telah terdaftar. Jika pasien telah terdaftar maka nomor pasiennya dicari, jika pasien baru maka akan dibuatkan nomor baru.
- 4) Masing-masing dokter praktik memiliki tarif tersendiri.

Data Dokter

Untuk mengolah tabel dokter, buat *form* seperti gambar berikut ini.

Data Dokter	
Kode	Nama Dokter
D1001	
D1002	

Gambar 19 *Form Data dokter*

Skenario program :

- 1) Jika *command input* dijalankan maka kode poli harus dipilih lebih awal. Jika poli GIGI maka program akan mencari kode dokter paling akhir di poli gigi, jika data tidak ditemukan maka akan dibuat kode dokter baru dengan struktur GIG01, jika data ditemukan maka kode dokter terakhir akan ditambah 1. Jika kode dokter terakhir adalah ADM03, maka kode adm baru adalah GIG04. Hal yang sama berlalu untuk kode poli lainnya.
- 2) Jika *command Edit* atau Hapus di klik, maka pencarian data dapat dilakukan dengan mengetik kodenya atau dengan memilih data dalam *grid* kemudian tekan *enter*.

Daftar Poli

Untuk mengolah tabel Poli, buat *form* seperti gambar di bawah ini.

Daftar Poli	
Kode	Nama Poli
P01	
P02	

Gambar 20 *Form Data Poli*

Daftar Nama Obat

Untuk mengolah tabel obat buatlah *form* seperti gambar di bawah ini.

Nama Obat		
Kode	Nama Obat	Harga
O1001		
O1002		

Gambar 21 *Form* Data Obat

Laporan ini terdiri atas tiga model yaitu :

- a) Laporan harian.
- b) Laporan mingguan dan,
- c) Laporan bulanan

Transaksi Pembayaran

Untuk transaksi pembayaran dibuat *form* seperti gambar di bawah ini:

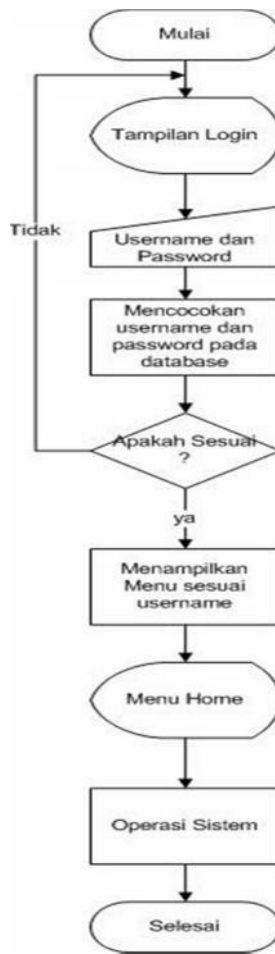
TRANSAKSI PEMBAYARAN				
No.			Poli	
Pasien			Dokter	
Kode	Tindakan		Tarif	
Biaya				
			Grantotal	
			Dibayar	
			Kembali	
<input type="button" value="Simpan"/>		<input type="button" value="Keluar"/>		

Gambar 22. *Form* Laporan Pembayaran

Flowchart (Diagram Alir)

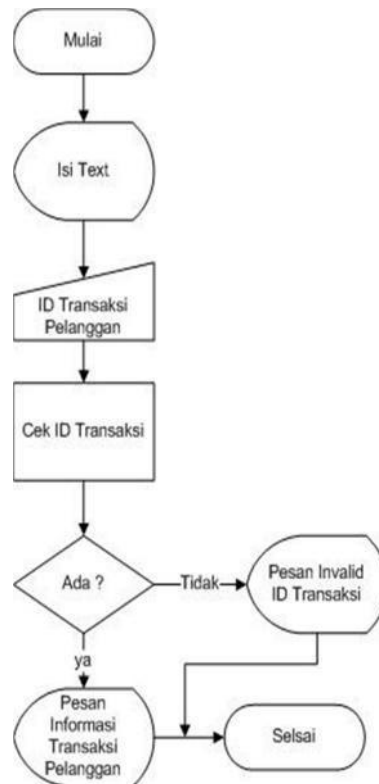
Adapun Diagram Alir untuk sistem informasi rekam medik rawat jalan adalah sebagai berikut :

4.1 Diagram Alir untuk Login



Gambar 23 Diagram Alir *Login*

4.2 Diagram Alir untuk Akses Informasi Pasien



Gambar 24. Diagram Alir Akses Informasi

Implementasi Pengembangan Sistem Informasi

Adapun tahapan-tahapan yang akan dilakukan untuk membangun sistem informasi rekam medik ini adalah sebagai berikut :

1. Membuat sistem informasi yang memiliki kebenaran logika dalam pemecahan masalah maupun penulisan kode program, dengan cara peneliti selalu memperhatikan pedoman pemrograman dari referensi serta mengikuti pesan yang ditampilkan oleh perangkat lunak *Visual Basic*.
2. Dalam pembuatan sistem informasi, penulisan program diusahakan memiliki waktu minimal yang tersedia untuk menuliskan kode program dari awal hingga siap untuk dieksekusi.

3. Diusahakan dalam pemilihan tipe data yang diimplementasikan *programmer* sesuai dengan kebutuhan pengguna sistem informasi. Seandainya harus digunakan variabel berindeks, diusahakan dalam jumlah minimal.
4. Sistem informasi yang dibuat diusahakan dapat dijalankan pada kondisi *platform* yang berbeda-beda, baik itu sistem operasi maupun perangkat keras, dengan cara sistem informasi diusahakan sesuai dengan kemampuan dari sistem operasi dan perangkat keras yang digunakan.
5. Melakukan pengujian terhadap sistem informasi apakah berjalan sesuai dengan yang diinginkan atau tidak.