BAB V DESAIN, IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan diuraikan mengenai proses perancangan sistem informasi manajemen berdasarkan permasalahan dan kebutuhan yang telah dijelaskan pada bab sebelumnya. Selanjutnya, pada bab ini akan membahas proses untuk menciptakan model fisik sebagai representasi awal perancangan sistem dilanjutkan dengan implementasi sistem untuk mengetahui apakah sistem telah berjalan sesuai dengan perencanaan awal dengan melakukan pengujian bertahap yaitu uji verifikasi dan uji validasi.

5.1 Sistem Desain

Tahap sistem desain yang dilakukan adalah tahap setelah dilakukan tahap perencanaan dan analisis. Tujuan dari tahap ini adalah untuk menciptakan model logis dan fisik dari sistem informasi yang telah direncanakan pada tahap sebelumnya. Model ini harus memenuhi daftar kebutuhan dan alur proses bisnis yang telah diuraikan pada DFD. Tahap desain memuat beberapa tahapan yakni desain *database*, desain *user interface*, dan desain algoritma.

5.1.1 Desain Basis Data

Tahap desain basis data atau *database* bertujuan untuk menentukan komponen-komponen yang berkaitan dengan penyimpanan data pada sistem informasi. Dalam tahap ini, konstruksi dari sistem informasi dirancang yang meliputi entitas, atribut dan relasinya. Penggambaran desain *database* dilakukan berdasarkan jenis dari data dokumennya. Desain basis data meliputi desain logis dan desain fisik.

5.1.1.1 Desain Database Logis

Desain *database* logis yang disebut juga dengan *logical model* adalah uraian penjelasan kepada *user* mengenai bagaimana fungsi-fungsi pada sistem informasi bekerja secara logika. Penggambaran dari *logical model* dapat menggunakan *Entity Relation Diagram* (ERD). Pada pembuatan *database* logis berdasarkan ERD, tahap yang perlu untuk dilakukan adalah dengan menentukan daftar entitas, menentukan hubungan relasi antar entitas, dan memastikan bahwa antar entitas telah memenuhi normalisasi (Whitten dan Bentley, 2007). Berikut adalah tahapan dalam pembuatan model ERD.

1. Daftar Entitas

Daftar entitas berisi identifikasi entitas yang terlibat pada jalannya sistem informasi. Entitas tersebut digunakan untuk penyimpanan data berdasarkan atribut dari entitas. Penentuan entitas dilakukan berdasarkan hasil wawancara berupa analisis *sysem requirement checklist* yang telah dilakukan sebelumnya. Pada Tabel 5.1 menjelaskan mengenai daftar identifikasi entitas dan atribut pada sistem informasi yang akan dibuat.

Tabel 5.1 Daftar Entitas dan Atribut

Entitas	Atribut	
Retailer	Kode_Retailer*, ID_Salesman, Site_Number, Nama_Retailer,	
	Site_Name, Alamat, No_Telepon, NPWP, Nama_NPWP,	
	Alamat_NPWP, Tipe_Tax, Credit_Limit, TOP, Status_Retailer, Agen,	
	Password	
Salesman	ID_Salesman*, Nama_Salesman, Area, No_Kendaraan, No_Telepon	
Surat_Pemesanan	No_SP*, Kode_Retailer, ID_Admin, Nama_Pemesan, Tanggal_Order,	
	Waktu_Order	
Produk_SP	No_Pesanan*, No_SP, Kode_Produk, Qty	
Produk	Kode_Produk*, Nama_Produk, Kemasan, Harga	
Admin	ID_Admin*, Nama_Admin, Gender, No_Telepon, Password	
Picking_List	No_PL*, Id_Pegawai_Logistik, ID_Petugas_Gudang, No_SP	
Produk_PL	No_Picking*, No_PL, Kode_Produk, Qty	
Logistik	ID_Pegawai_Logsitik*, Nama, No_SIKA, Gender, No_Telepon,	
	Password	
Petugas_Gudang	ID_Petugas_Gudang*, Nama, Gender, No_Telepon	
Faktur	No_Faktur*, No_PL, ID_Fakturis, ID_Petugas_Pengiriman, Tanggal,	
	Jatuh_Tempo, Nama_Penerima	
Fakturis	ID_Fakturis*, Nama, Gender, No_Telepon, Password	
Petugas_Pengiriman	ID_Petugas_Pengiriman*, Nama, Rayon, Gender, No_Telepon	
Contra_Bond_Receipt	No_Contra_Bond*, ID_Pegawai_Inkaso, No_Faktur, Tanggal,	
	Tanggal_Pembayaran, Nama Penerima	
Inkaso	ID_Pegawai_Inkaso*, Nama, Gender, No_Telepon, Password	
Transaksi_Masuk_Gudang	No_PO*, Tanggal	
Produk_Masuk_Gudang	No_Transaksi*, No_PO, No_Batch, Qty	
Batch_Produk	No_Batch*, Expired_Date, Kode_Produk	
Produk_Gudang	No_Pengecekan*, Kode_Produk, Stock_Opname, Stock_Level	
Transaksi_Keluar_Outlet	No_Nota*, Kode_Retailer, Tanggal	
Produk_Nota	No_Transaksi_Nota*, No_Nota, Kode_Produk, Qty	
Produk_Outlet	No_Control*, Kode_Produk, Kode_Retailer, Stock_Opname,	
	Stock_Level	
Notifikasi	No_Notifikasi, Tanggal, Kode_Retailer, Isi, Tandai_Notifikasi	

2. Relasi

Entitas yang telah diidentifikasi kemudian dihubungkan antar entitas tersebut untuk mengetahui hubungan atau relasinya. Daftar relasi dari entitas pada sistem informasi yang dibuat terdapat di Tabel 5.2.

3. Entity Relationship Diagram (ERD)

Hasil dari tabel relasi antar entitas kemudian dilakukan penggambaran model ERD. ERD menunjukkan objek/entitas data dan relasi/hubungan antar entitas tertentu. ERD dari sistem informasi terdapat pada Gambar 5.1

Tabel 5.2 Relasi Entitas

Entitas	Hubungan	Entitas	Jenis Relasi	Kardinalitas
Salesman	Bertugas pada	Retailer	One to Many	$(1,1) \rightarrow (1,N)$
Retailer	Mengajukan	Surat_Pemesanan	One to Many	$(1,1) \rightarrow (1,N)$
Surat_Pemesanan	Memuat	Produk_SP	One to Many	$(1,1) \rightarrow (1,N)$
Admin	Mengonfirmasi	Surat_Pemesanan	One to Many	$(1,1) \rightarrow (0,N)$
Picking_List	Memproses	Surat_Pemesanan	One to Many	$(1,1) \rightarrow (1,N)$
Picking_List	Memuat	Produk_PL	One to Many	$(1,1) \rightarrow (1,N)$
Petugas_Gudang	Mengerjakan	Picking_List	One to Many	$(1,1) \rightarrow (0,N)$
Logistik	Membuat	Picking_List	One to Many	$(1,1) \rightarrow (0,N)$
Faktur	Memproses	Picking_List	One to One	$(1,1) \rightarrow (1,1)$
Petugas_Pengiriman	Mengantar	Faktur	One to Many	$(1,1) \rightarrow (0,N)$
Fakturis	Menerbitkan	Faktur	One to Many	$(1,1) \rightarrow (0,N)$
Contra_Bond	Menagihkan	Faktur	One to One	$(1,1) \rightarrow (1,1)$
Inkaso	Membuat	Contra_Bond	One to Many	$(1,1) \rightarrow (0,N)$
Produk_SP	Berdasarkan	Produk	One to Many	$(1,1) \rightarrow (1,N)$
Produk_PL	Berdasarkan	Produk	One to Many	$(1,1) \rightarrow (1,N)$
Batch_Produk	Berdasarkan	Produk	One to Many	$(1,1) \rightarrow (1,N)$
Produk	Mengontrol	Produk_Gudang	One to Many	$(1,1) \rightarrow (1,N)$
Produk_Masuk_Gudang	Memasukkan	Batch_Produk	One to One	$(1,1) \rightarrow (1,1)$
Transaksi_Masuk_Gudang	Menambah	Produk_Masuk_ Gudang	One to Many	$(1,1) \rightarrow (1,N)$
Produk_Outlet	Mengontrol	Produk	One to Many	$(1,1) \rightarrow (1,N)$
Produk_Nota	Mengurangi	Produk	One to Many	$(1,1) \rightarrow (1,N)$
Transaksi_Keluar_Outlet	Menambah	Produk_Nota	One to Many	$(1,1) \rightarrow (1,N)$
Retailer	Mendapat	Notifikasi	One to Many	$(1,1) \rightarrow (1,N)$

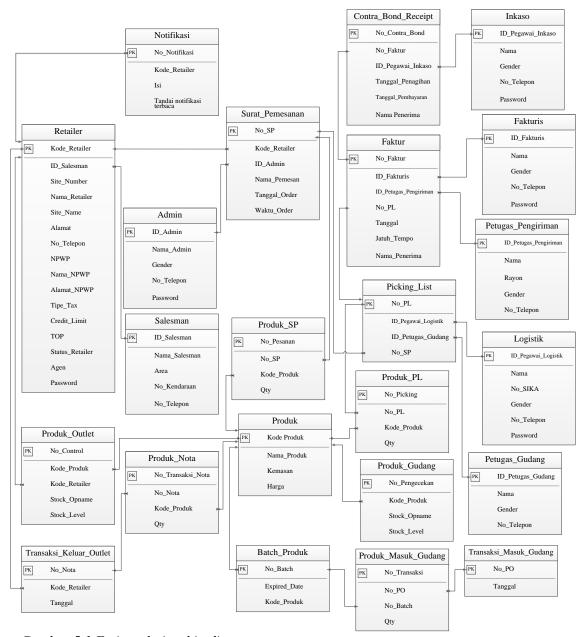
Pada Gambar 5.1 menunjukkan ERD dari sistem informasi antara PT IGM dengan beberapa retailer. ERD tersebut menjelaskan hubungan antara satu entitas dengan entitas lainnya yang ditunjukkan dengan simbol pada garis yang menhubungkan antar entitas. Salah satu contoh dari hubungan antar entitas yaitu entitas Retailer mengajukan Surat Pemesanan dengan hubungan *one to many* dimana maksud dari relasi tersebut yaitu setiap retailer dapat mengajukan sedikitnya satu surat pemesanan atau lebih.

4. Normalisasi

Normalisasi bertujuan untuk menciptakan entitas yang stabil dan tidak adanya redudansi data. Normalisasi merupakan proses mendesain struktur basis data dan teknis analisis data yang mengorganisasikan beberapa atribut data dengan mengelompokkan atribut tersebut sehingga terbentuk entitas yang tidak redundan, stabil dan fleksibel serta menghasilkan suatu tabel yang dianggap normal. Pada perancangan sistem informasi minimal harus memenuhi hingga normalisasi tahap ketiga (3NF), agar tabel dapat dikatakan normal. Pada tahap perancangan desain sistem informasi ini terdapat redudansi data antara entitas surat pemesanan dengan entitas produk. Hal ini dikarenakan entitas surat pemesanan dapat memiliki banyak produk yang berbeda,

sedangkan produk juga dapat dimiliki oleh lebih dari satu entitas surat pemesanan. Relasi antara kedua entitas ini adalah *many to many*. Oleh karena itu, dilakukan normalisasi antara kedua entitas ini agar tidak terjadi redudansi data. Normalisasi yang dilakukan yakni dengan memberikan tabel *dummy* di antara entitas surat pemesanan dan entitas produk. Tabel *dummy* ini diberi nama dengan entitas produk SP. Entitas produk SP berfungsi untuk menampung data produk yang hanya berkaitan dengan surat pemesanan tersebut.

Setelah dilakukan normalisasi maka keseluruhan tabel yang ada pada perancangan sistem informasi ini dapat dikatakan normal dan memenuhi ketentuan normalisasi struktur hingga 3NF sesuai yang dijelaskan pada tinjauan pustaka.



Gambar 5.1 Entity relationship diagram

5.1.1.2 Desain *Database* Fisik

Desain *database* fisik adalah tahapan yang bertujuan untuk menggambarkan desain *database* logis menjadi bentuk aktual dari perancangan sistem informasi. Desain *database* fisik dilakukan dengan menggunakan bahasa pemrograman MySQL. Penggunaan bahasa pemrograman MySQL untuk pembuatan *database* fisik dikarenakan MySQL memberikan lingkungan pemrograman yang lebih fleksibel. *Database* MySQL dapat terhubung dengan *script* PHP menggunakan perintah *query*. Desain *database* fisik bertujuan untuk penyimpanan data ke dalam tabel terpisah dan menempatakan data berdasarkan kesatuan data dari desain logis yang dilakukan sebelumnya. Berikut merupakan contoh desain *database* fisik dari sistem informasi manajemen industri farmasi PT IGM Malang.

1. Entitas Salesman

Pada Tabel 5.3 berikut akan dijelaskan mengenai desain *database* fisik dari entitas/tabel salesman.

Tabel 5.3 Desain *Database* Entitas Salesman

Field	Data Type	Field Size / Format	Key
ID_Salesman	Big Integer	12	Primary key
Nama_Salesman	Varchar	50	
Area	Varchar	30	
No_Kendaraan	Varchar	10	
No_Telepon	Varchar	15	

2. Entitas Surat Pemesanan

Tabel 5.4 menunjukan desain *database* fisik dari entitas/tabel surat pemesanan yang berfungsi untuk mencatat proses pemesanan yang masuk pada sistem informasi.

Tabel 5.4 Desain *Database* Entitas Surat Pemesanan

Field	Data Type	Field Size / Format	Key
No_SP	Big Integer	12	Primary key
Kode_Retailer	Big Integer	12	Foreign Key
ID_Admin	Big Integer	12	Foreign Key
Nama_Pemesan	Varchar	50	
Tanggal_Order	Date		
Waktu_Order	Time		

3. Entitas Produk

Tabel 5.5 menunjukan desain database fisik dari entitas/tabel Produk.

Tabel 5.5 Desain *Database* Entitas Produk

Field	Data Type	Field Size / Format	Key
Kode_Produk	Big Integer	12	Primary key
Nama_Produk	Varchar	50	
Kemasan	Varchar	30	
Harga	Integer	15	

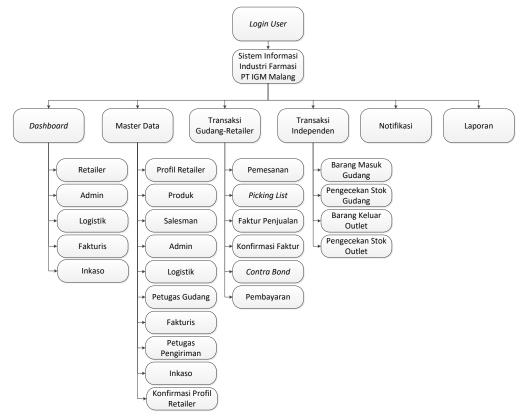
Desain *database* fisik dilakukan pada seluruh entitas yang ada sesuai dengan perancangan. Desain *database* fisik untuk keseluruhan entitas dilakukan dengan cara yang sama seperti tiga contoh sebelumnya. Hasil dari desain *database* secara fisik ini kemudian akan menjadi acuan implementasi pembuatan tabel di MySQL.

5.1.2 Desain User Interface

User Interface menggambarkan bagaimana user berinteraksi dengan sistem informasi. Desain user interface memiliki tujuan untuk memungkinkan user menjalankan setiap tugas pada user requirement. Hal yang harus diperhatikan ketika membuat user interface yaitu memperhatikan kebutuhan dan kenyamanan dari user. Selain itu, user interface mampu diimplementasikan pada pembuatan sistem informasi.

5.1.2.1 Hierarki Menu

Hierarki menu dibuat untuk mempermudah dalam menjelaskan mengenai tingkatan menu yang ditampilkan pada sistem informasi yang dibuat. Penjelasan ini bertujuan untuk memudahkan *user* saat menggunakan sistem informasi. Selain itu, desain hierarki menjadi dasar untuk memudahkan implementasi pembuatan sistem informasi. Gambar 5.2 menampilkan hierarki menu dari sistem informasi yang akan dibuat.



Gambar 5.2 Hierarki Menu

- Master Data adalah halaman menu yang menampilkan data setiap tabel yang ada di sistem informasi yang dikelola oleh admin. Berikut merupakan sub menu dari Master Data.
 - a. Profil retailer, menampilkan data dari *user* yakni retailer. Data yang ditampilkan seperti kode retailer, ID Salesman, *site number*, nama retailer, *site name*, alamat, no telepon, NPWP, nama NPWP, alamat NPWP, tipe *tax*, *credit limit*, TOP, status retailer, agen, *password*. Pada pengolahan data profil retailer diakses oleh dua *user* yaitu retailer yang melakukan pendaftaran retailer untuk kemudian dilakukan konfirmasi profil retailer oleh admin.
 - b. Data Produk, menampilkan data produk yang terdapat di PT IGM Malang. Data yang ditampilkan yakni nama produk, kemasan, dan harga.
 - Data Salesman, menampilkan data salesman yang bekerja pada PT IGM Malang.
 Data yang ditampilkan yakni nama salesman, area, no kendaraan, dan no telepon.
 - d. Data Admin, menampilkan data *user* admin pada PT IGM Malang. Data yang ditampilkan yakni nama admin, *gender*, no telepon dan *password*.
 - e. Data Logistik, menampilkan data *user* pegawai logistik pada PT IGM Malang. Data yang ditampilkan yakni nama, no SIKA (Surat Izin Kerja Apoteker), *gender*, no telepon, dan *password*.
 - f. Data Petugas Gudang, menampilkan data petugas gudang yang bekerja pada PT IGM Malang. Data yang ditampilkan yakni nama, *gender* dan no telepon.
 - g. Data Fakturis, menampilkan data *user* fakturis pada PT IGM Malang. Data yang ditampilkan yakni nama, *gender*, no telepon dan *password*.
 - h. Data Petugas Pengirirman, menampilkan data petugas pengiriman yang bekerja pada PT IGM Malang. Data yang ditampilkan yakni nama, rayon, *gender* dan no telepon.
 - i. Data Inkaso, menampilkan data *user* inkaso pada PT IGM Malang. Data yang ditampilkan yakni nama, *gender*, no telepon dan *password*.
- 2. Transaksi Gudang-Retailer, merupakan menu yang berfungsi untuk melakukan proses transaksi yang digunakan seluruh *user*. Pada menu ini terdapat enam submenu yang dapat digunakan sesuai dengan fungsinya. Enam submenu tersebut yakni pembuatan surat pemesanan oleh dua *user* diantaranya oleh retailer dan disetujui oleh admin, pembuatan *picking list*, pembuatan faktur penjualan, konfirmasi pengiriman atau pembuatan *contra bond* dan konfirmasi pembayaran.

- 3. Transaksi independen, merupakan menu yang berfungsi bagi masing-masing outlet dengan gudang dalam mengatur informasi persediaan. Pada menu ini terdapat empat submenu dengan pembagian dua submenu untuk retailer yaitu barang keluar outlet dan pengecekan stok outlet, serta submenu untuk gudang dikelola oleh *user* logistik yaitu barang masuk gudang dan pengecekan stok gudang.
- 4. Notifikasi, halaman notifikasi menampilkan pemneritahuan yang dilakukan *user* pada sistem informasi untuk kemudian disampaikan kepada *user* lain. Aktivitas yang dimaksud seperti pemberitahuan ketersediaan produk yang tidak sesuai dengan pemesanan untuk retailer, konfirmasi pengiriman dan penagihan pembayaran.
- Laporan, merupakan halaman yang menampilkan ringkasan data dari sistem informasi.
 Laporan dapat diperoleh dari masing-masing data yang telah dimasukkan sebelumnya.

Hierarki menu yang dirancang nantinya diimplementasikan sebagai acuannya. Selain itu terdapat lima *user* dalam sistem informasi ini, yaitu retailer, admin, logistik, fakturis dan inkaso. Setiap *user* memiliki akses sesuai dengan wewenangnya masing-masing. Tabel 5.6 menjelaskan otoritas dari masing-masing *user*.

Tabel 5.6 Otorasi Akses *User*

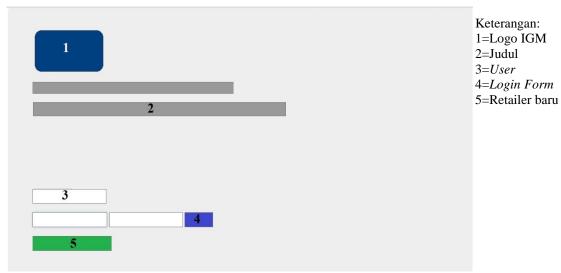
	Menu	User					
	Retailer	Admin	Logistik	Fakturis	Inkaso		
	Retailer	✓					
	Admin		✓				
Dashboard	Logistik			✓			
	Fakturis				✓		
	Inkaso					✓	
	Profil retailer	✓	✓				
	Produk		✓				
	Salesman		✓				
	Admin		✓				
Master Data	Logistik		✓				
	Petugas Gudang		✓				
	Fakturis		✓				
	Petugas Pengiriman		✓				
	Inkaso		✓				
	Pemesanan	✓	✓	✓			
Transaksi	Picking list			✓	✓		
Gudang-	Faktur Penjualan				✓	✓	
Retailer	Contra bond					✓	
	Pembayaran					✓	
	Barang Masuk Gudang			✓			
Transaksi Independen	Pengecekan Stok Gudang			✓			
	Barang Masuk Outlet	✓					
	Pengecekan Stok Outlet	✓					
Notifiles:	Ketersediaan Produk Pesanan	✓		✓			
Notifikasi	Penagihan	✓				✓	
Laporan		✓	✓	✓	✓	✓	

5.1.2.2 Desain *Form*

Tahap desain *Form* bertujuan untuk menentukan tampilan yang sesuai untuk menunjang *interface* dari sistem informasi yang dibuat agar lebih interaktif dan memudahkan pengguna. Perancangan desain *form* dijelaskan sebagai berikut.

1. Form Login

Pada *form login* berfungsi untuk melakukan validasi akses sebelum masuk ke sistem informasi hanya untuk pengguna yang memiliki otorasi terhadap sistem informasi. Tampilan desain *interface* dari *form login* dapat dilihat pada Gambar 5.3.



Gambar 5.3 Form login

2. Dashboard

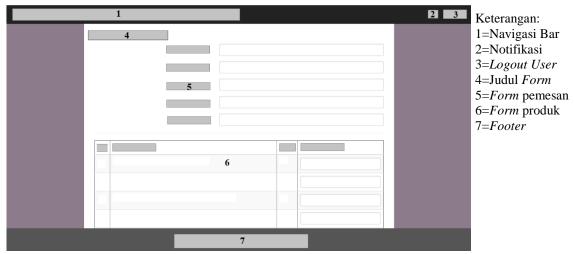
Pada *dashboard* berfungsi sebagai halaman pusat kontrol bagi *user* setelah melakukan proses *login*. Bagi setiap *user* memiliki kebutuhan yang berbeda untuk setiap menunya, namun dengan *template* yang sama seperti pada Gambar 5.4 berikut.



Gambar 5.4 Form login

3. Transaksi Gudang-Retailer

Menu ini berfungsi melakukan transaksi pada sistem informasi yang dibuat. Tampilan setiap *form* yang mudah dan nyaman dalam melakukan akses. Gambar 5.5 menunjukan desain *interface* dari halaman surat pemesanan.



Gambar 5.5 Form picking list

4. Transaksi Independen

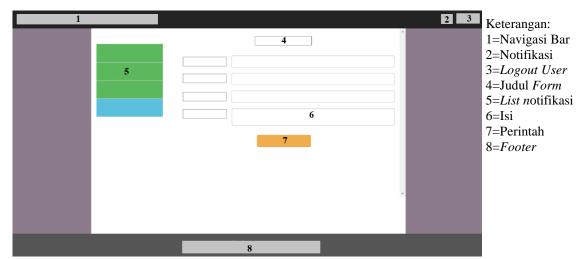
Halaman pada menu ini berfungsi dalam melakukan proses bisnis pada sistem informasi bagi masing-masing outlet (retailer) dan logistik. Tampilan setiap *form* memiliki *template* yang sama agar lebih mudah dan nyaman dalam melakukan akses. Gambar 5.6 menunjukan desain *interface* dari halaman barang keluar outlet.



Gambar 5.6 Form barang keluar outlet

5. Notifikasi

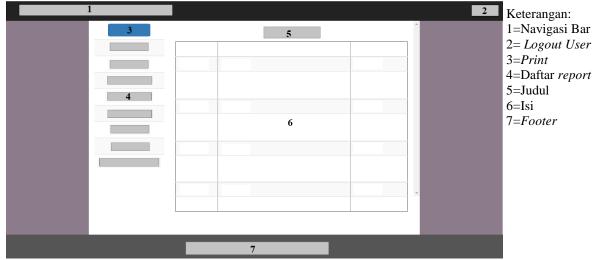
Halaman pada menu ini berfungsi untuk menampilkan pesan notifikasi dari *user* pada indofarma global medika kepada *user* retailer. Gambar 5.7 menunjukan desain *interface* dari halaman notifikasi.



Gambar 5.7 Form notifikasi

6. Desain Report

Desain *report* laporan yang dihasilkan oleh sistem informasi. *Report* bertujuan untuk mengetahui daftar dokumen berserta isi di dalamnya secara ringkas dalam bentuk *print out* ber*form*at pdf. Desain *interface* dari *report* ini dapat dilihat pada Gambar 5.8.



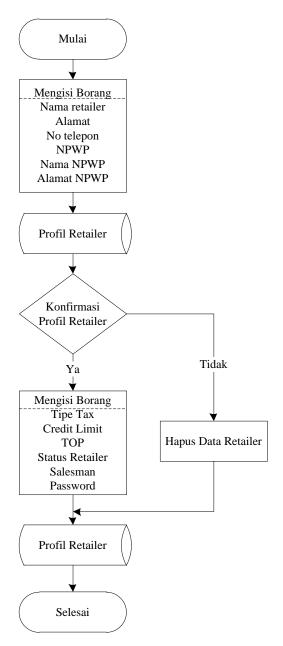
Gambar 5.8 Desain interface report

5.1.3 Desain Algoritma

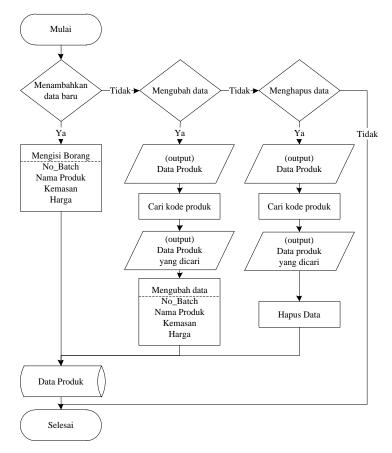
Desain algoritma merupakan tahapan dalam melakukan desain gambaran alur proses. Desain algoritma dibuat berdasarkan pada jalannya sistem informasi dengan hasil akhir *user* mendapatkan *output* yang sesuai dengan kebutuhan. Pada penelitian ini digunakan *flowchart* dalam menggambarkan desain algoritma proses. Desain algoritma proses mengacu pada sistem modelling yang sudah dirancang sebelumnya pada Bab IV. Berikut merupakan penjelasan desain algoritma dengan menggunakan *flowchart* berdasarkan proses berjalannya sistem informasi yang dibuat.

1. Flowchart Input dan Edit Master Data

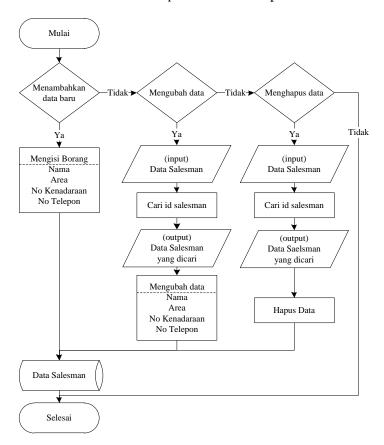
Proses *input* dan *edit* data bertujuan untuk menambah data atau mengubah data yang telah ada di dalam *database* sesuai dengan ketentuan rancangan sistem. Sesuai dengan rancangan *process modelling* yang telah dibuat, terdapat delapan proses dalam pengolahan master data diantaranya pengolahan profil retailer, pengolahan data produk, pengolahan data salesman, pengolahan data admin, pengolahan data logistik, pengolahan data petugas gudang, pengolahan data fakturis, pengolahan data inkaso, dan pengolahan data petugas pengiriman. Gambar 5.9 sampai dengan Gambar 5.17 menampilkan *flowchart* proses *input* dan *edit* master data.



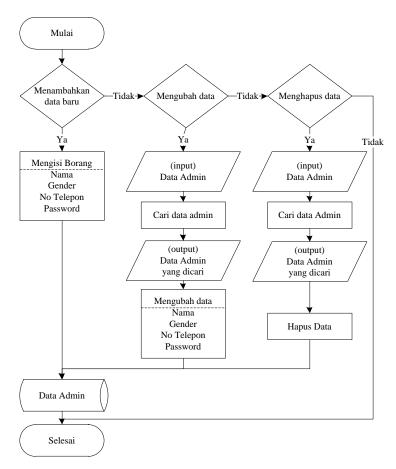
Gambar 5.9 Flowchart input dan edit profil retailer



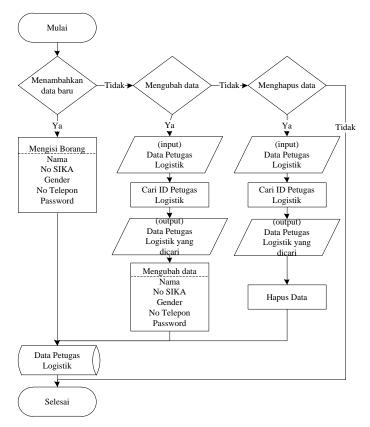
Gambar 5.10 Flowchart input dan edit data produk



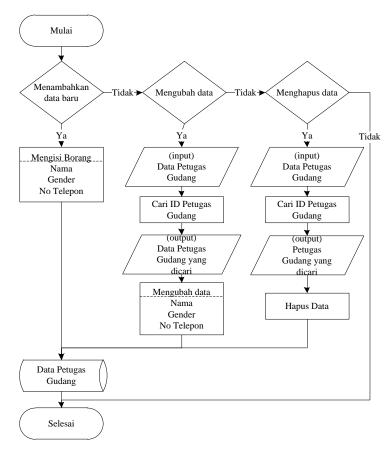
Gambar 5.11 Flowchart input dan edit data salesman



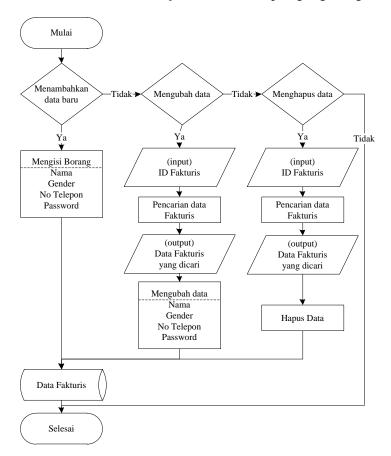
Gambar 5.12 Flowchart input dan edit data admin



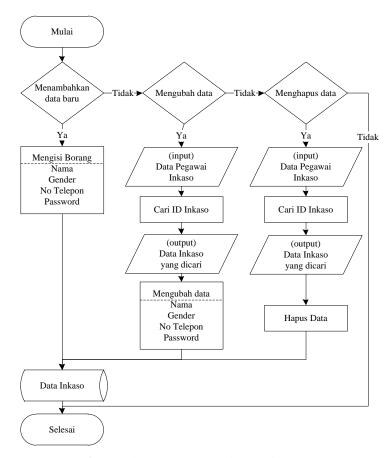
Gambar 5.13 Flowchart input dan edit data petugas logistik



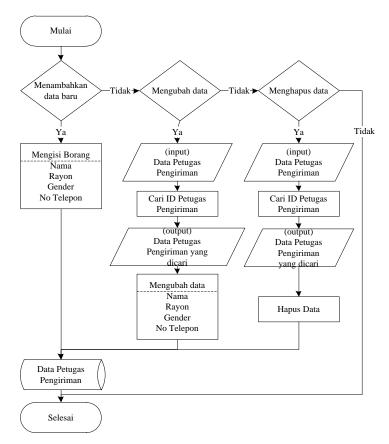
Gambar 5.14 Flowchart input dan edit data petugas gudang



Gambar 5.15 Flowchart input dan edit data fakturis



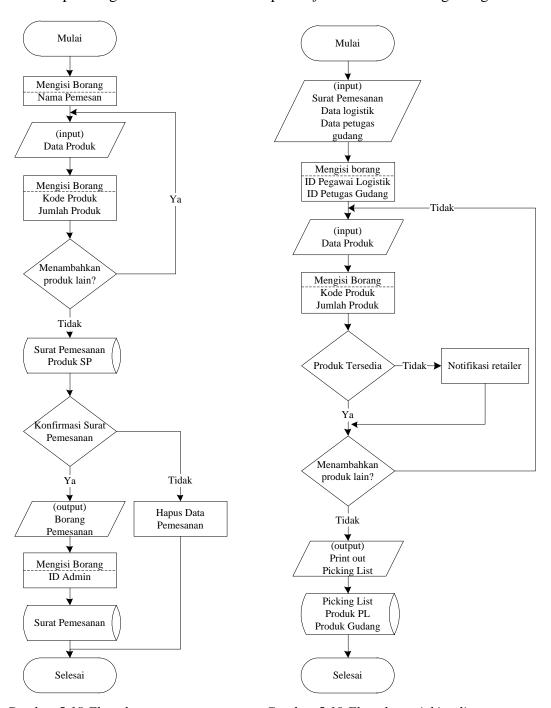
Gambar 5.16 Flowchart input dan edit data inkaso



Gambar 5.17 Flowchart input dan edit data petugas pengiriman

2. Flowchart Transaksi Gudang-Retailer

Proses transaksi gudang-retailer merupakan serangkaian aktivitas dalam proses bisnis antara seluruh *user* gudang dalam memenuhi pesanan retailer berfungsi untuk menambah data ke dalam *database* sesuai dengan ketentuan rancangan sistem. Sesuai dengan rancangan *process modelling* yang telah dibuat, terdapat lima proses dalam sistem transaksi gudang-retailer diantaranya pemesanan, konfirmasi pemesanan, pembuatan *picking list*, pengiriman, konfirmasi pengiriman dan pembayaran. Gambar 5.18 sampai dengan Gambar 5.22 menampilkan *flowchart* transaksi gudang-ratailer.



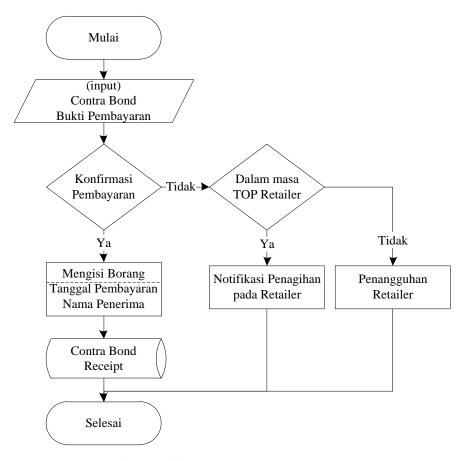
Gambar 5.18 Flowchart surat pemesanan Gambar 5.19 Flowchart picking list



Gambar 5.20 Flowchart informasi pengiriman



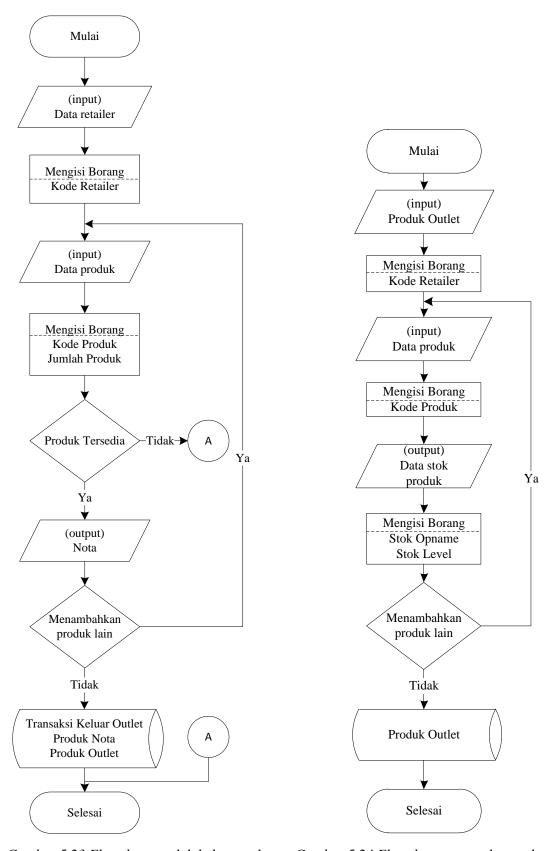
Gambar 5.21 Flowchart konfirmasi pengiriman



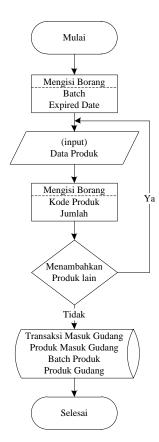
Gambar 5.22 Flowchart konfirmasi pembayaran

3. Flowchart Transaksi Independen Gudang dan Outlet

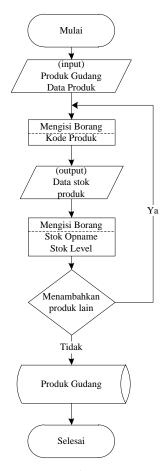
Proses transaksi independen gudang dan outlet merupakan aktivitas transaksi masing-masing *user* retailer dalam mengatur persediaan pada outlet dengan melakukan transaksi keluar dalam penjualan produk kepada *customer*. Selain itu juga dilakukan pengecekan persediaan pada outlet secara berkala dengan memasukkan stok opname. Begitu pula dengan *user* logistik pada gudang dapat mengatur persediaan transaksi masuk gudang. Selain itu *user* logistik juga dapat melakukan pengecekan persediaan pada gudang secara berkala dengan stok opname. Transaksi masuk gudang dan transaksi keluar outlet berfungsi untuk menambah data ke dalam *database* sesuai dengan ketentuan rancangan sistem. Sesuai dengan rancangan *process modelling* yang telah dibuat, terdapat empat proses dalam sistem transaksi gudang dan outlet diantaranya pada bagian logistik persediaan gudang dilakukan pemasukan barang pada gudang dan pengecekkan stok gudang sedangkan pada bagian retailer persediaan gudang dilakukan pengeluaran barang outlet dan pengecekkan stok outlet. Gambar 5.23 sampai dengan Gambar 5.26 menampilkan *flowchart* transaksi independen gudang dan outlet.



Gambar 5.23 Flowchart produk keluar outlet Gambar 5.24 Flowchart pengecekan stok outlet



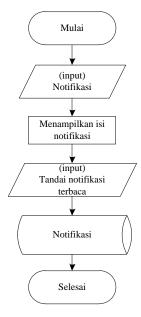
Gambar 5.25 Flowchart transaksi produk masuk gudang



Gambar 5.26 Flowchart pengecekan stok produk gudang

4. Notifikasi

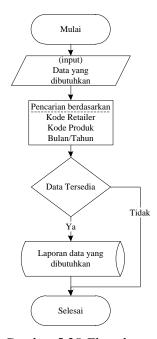
Notifikasi merupakan suatu fitur untuk *user* retailer untuk mengetahui informasi dari *user* pada gudang seperti *user* logsitik dan *user* inkaso berfungsi untuk membaca data dari *database* sesuai dengan ketentuan rancangan sistem. Gambar 5.27 menampilkan *flowchart* proses notifikasi.



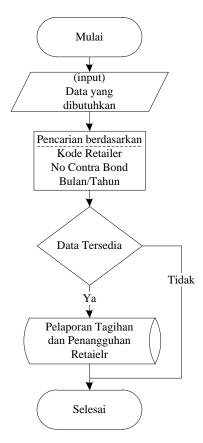
Gambar 5.27 Flowchart notifikasi

5. Flowchart Pelaporan

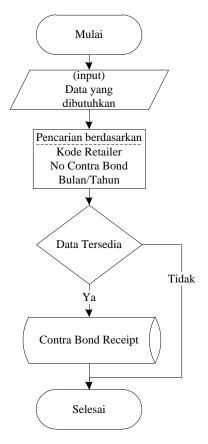
Flowchart pelaporan berfungsi menampilkan proses untuk mendapatkan laporan yang diinginkan dari sistem informasi. *Flowchart* pelaporan dapat dilihat pada Gambar 5.28 sampai dengan Gambar 5.30.



Gambar 5.28 Flowchart pelaporan transaksi bulanan



Gambar 5.29 Flowchart pelaporan tagihan



Gambar 5.30 Flowchart pelaporan pembayaran

5.2 Implementasi

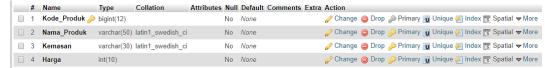
Implementasi merupakan tahapan penerapan rancangan yang telah dibuat sebelumnya ke dalam *software* sehingga sistem dapat digunakan secara langsung oleh *user* untuk dilakukan analisa dan pengujian sistem. Tahap ini terdiri dari dua implementasi yakni implementasi *database*, implementasi antar muka pengguna dan implementasi *report*. Perancangan sistem informasi dilakukan dengan menggunakan bahasa pemrograman PHP dan dibantu dengan *software* XAMPP.

5.2.1 Implementasi Database

Implementasi pembuatan *database* dilakukan dengan pembuatan tabel. Tahap ini dilakukan dengan merancang tabel dengan menggunakan *tools phpmyadmin* berbasis bahasa pemrograman MySQL sesuai dengan tahapan desain yang telah dilakukan sebelumnya. Contoh dari implementasi *database* dapat dilihat sebagai berikut.

1. Tabel Produk

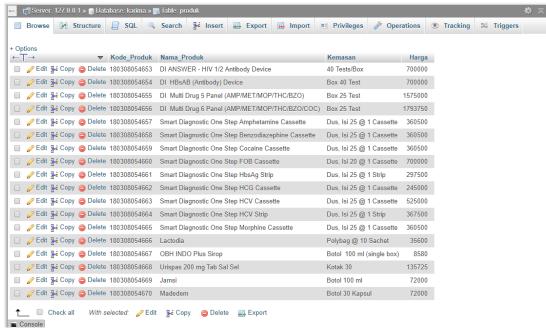
Implementasi tabel produk dapat dilihat pada Gambar 5.31 pada tampilan *phpmyadmin*.



Gambar 5.31 Implementasi tabel produk

2. Data pada Tabel Produk

Implementasi tabel produk dapat menyimpan data dapat dilihat pada Gambar 5.32 pada tampilan *phpmyadmin*.



Gambar 5.32 Implementasi data pada tabel produk

5.2.2 Implementasi Antar Muka Pengguna (*User Interface*)

Form dirancangan bertujuan untuk memudahkan user saat berinteraksi dengan sistem informasi yang dibuat. Implementasi dari form dilakukan berdasarkan tahap awal yang telah dilakukan sebelumnya. Berikut adalah contoh implementasi form dari perancangan sistem informasi industri farmasi Indofarma Global Medika Malang.

1. Form Login

Form login digunakan untuk melakukan validasi akses kedalam sistem informasi. Gambar 5.33 menampilkan hasil implementasi rancangan *user interface form login*.



Gambar 5.33 Implementasi form login

2. Dashboard

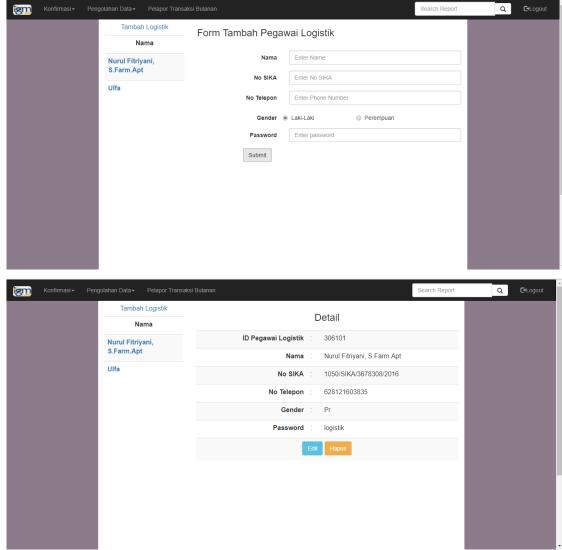
Dashboard merupakan pusat kontrol bagi pengguna digunakan untuk melakukan validasi struktur menu pengguna kedalam sistem informasi. Gambar 5.34 menampilkan hasil implementasi rancangan *user interface dashboard*.



Gambar 5.34 Implementasi dashboard

3. *Form* Master Data

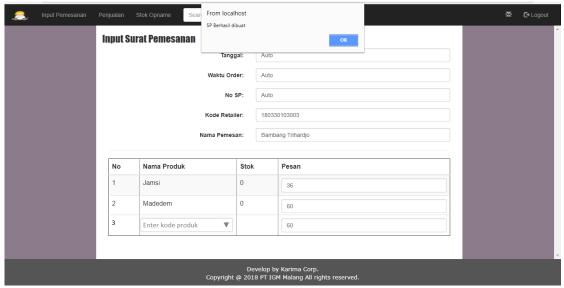
Form master data merupakan form yang digunakan untuk memasukkan data. Selain fungsi input juga dapat dilakukan fungsi lain diantaranya hapus dan memperbarui atau edit data. Contoh implementasi form input/edit pada pengolahan data pegawai logistik dapat dilihat pada Gambar 5.35.



Gambar 5.35 Implementasi form master data

4. Form Transaksi Gudang - Retailer

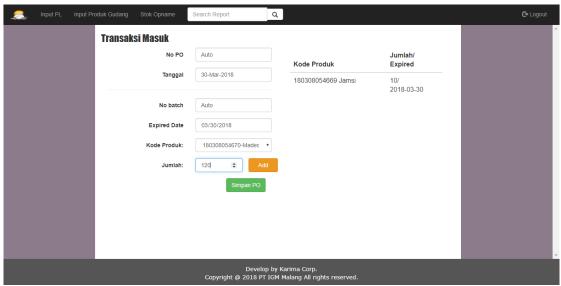
Form transaksi gudang-retailer merupakan form yang digunakan untuk memasukkan data transaksi yang berkaitan dengan proses bisnis antara user pada gudang dengan retailer maupun antar user pada gudang diantaranya admin, logistik, fakturis dan inkaso dalam menjalankan proses bisnis. Implementasi form transaksi pemesanan sebagai contoh transaksi gudang-retailer dapat dilihat pada Gambar 5.36.



Gambar 5.36 Implementasi form transaksi gudang - retailer

5. Form Transaksi Independen

Form transaksi independen merupakan form yang digunakan untuk memasukkan data transaksi yang berkaitan dengan proses bisnis transaksi independen dari masing-masing retailer maupun gudang. Implementasi form transaksi produk masuk gudang sebagai contoh dapat dilihat pada Gambar 5.37.



Gambar 5.37 Implementasi form transaksi independen

6. Tampilan Notifikasi

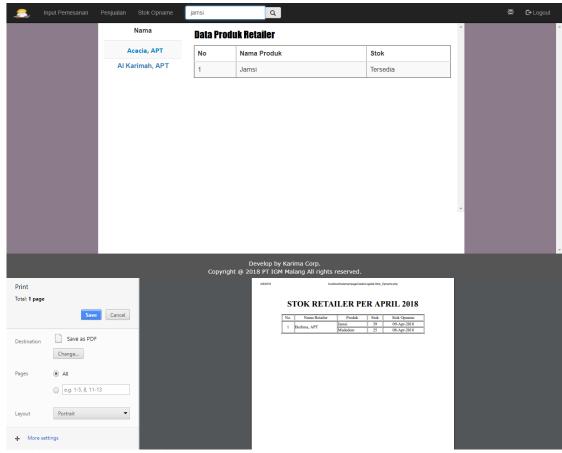
Notifikasi merupakan informasi yang dapat diterima oleh retailer mengenai hal-hal yang diinformasikan dari distributor seperti ketidaktersediaan produk pesanan maupun penagihan. Selain itu notifikasi juga menginformasikan produk apa saja yang dibutuhkan segera dilakukan pemesanan. Implementasi tampilan notifikasi dapat dilihat pada Gambar 5.38.

Input Pemesanan		Search Report	Q	▼ Logout
	No Notifikasi		Detail Notifikasi	^
	2018-03-30			
		No Notifikasi	9	
		Tanggal	2018-03-30	
		Kode Retailer	180330103003	
		Isi Pesan	Surat Pemesanan dengan Nomor: 180330111357 dan Nama Pemesan (Bambang Trihardjo) tidak dapat kami proses dekarenakan kelerbalasan stok pada produk dengan kode:180308054670	
			Tandai sudah dibaca	
			velop by Karima Corp. 3 PT IGM Malang All rights reserved.	

Gambar 5.38 Implementasi form notifikasi

7. Tampilan Report

Report merupakan salah satu fasilitas pada sistem informasi yang digunakan untuk medapatkan ringkasan ataupun detail data yang ingin diketahui informasinya. Report pada sistem informasi ini menampilkan keseluruhan data yang diinginkan. Berikut merupakan hasil implementasi dari perancangan report pada sistem informasi manajemen industri farmasi.



Gambar 5.39 Implementasi report

5.2.3 Implementasi *Pseudocode*

Pseudocode sebagai deskripsi dari algoritma pemrograman yang menggunakan struktur bahasa pemrograman sederhana sehingga dapat memudahkan manusia dalam memahami prinsip algoritma. Pseudocode pada implementasi dibuat berdasarkan algoritma proses yang sebelumnya telah dibuat pada tahap perancangan menggunakan flowchart atau diagram alir. Berikut merupakan implementasi dari pseudocode sistem informasi manajemen industri farmasi.

1. Pseudocode input data

Input data digunakan untuk memasukkan data ke dalam sistem informasi. Berikut merupakan contoh *pseudocode* input data pada sistem informasi manajemen industri farmasi.

Deklarasi

```
BigInt $kode retailer

Varchar $nama_retailer, $alamat, $no_telepon, $npwp, $nama_npwp, $alamat npwp
```

Deskripsi

```
include once db koneksi.php;
write ("Nama: ");
read ($nama retailer);
write ("Alamat: ");
read ($alamat);
write ("No Telepon: ");
read ($no telepon);
write ("NPWP: ");
read ($npwp);
write ("Nama NPWP: ");
read ($nama_npwp);
write ("Alamat NPWP: ");
read ($alamat_npwp);
$query ← INSERT INTO $retailer;
      Kode_Retailer ← $kode_retailer;
      Nama Retailer ← $nama retailer;
      Alamat ← $alamat;
      No Telepon ← $no telepon;
      NPWP ← $npwp;
      Nama NPWP ← $nama npwp;
      Alamat NPWP ← $alamat npwp;
If \$koneksi \leftarrow query (\$query);
      Then echo 'OK';
      Else echo 'false';
End if
```

2. *Pseudocode* menampilkan data

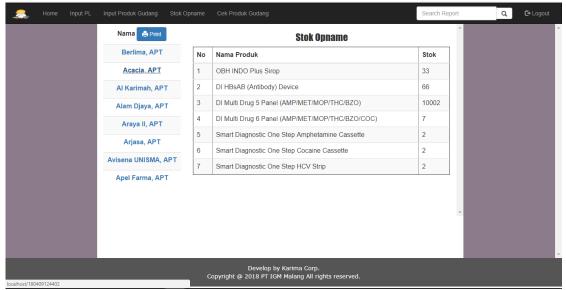
Data yang sudah disimpan dalam *database* sistem informasi dapat ditampilkan kembali dalam bentuk *form* maupun *report*. Berikut merupakan contoh *pseudocode* menampilkan data pada sistem informasi manajemen industri farmasi.

Deklarasi

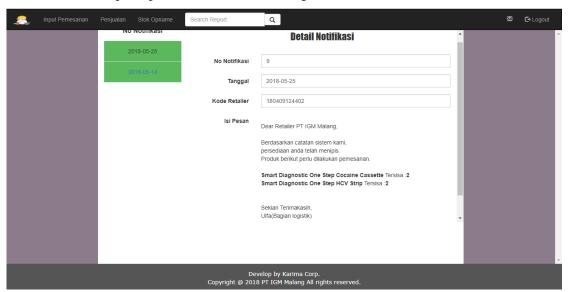
```
BigInt
             $kode retailer, $id salesman
             $nama retailer, $alamat, $no telepon, $npwp, $nama npwp,
  Varchar
             $alamat npwp, $site number, $site name, $agen,
             $tipe tax, $credit limit, $top, $password
Deskripsi
  include once db koneksi.php;
  session start()
        $kode retailer \(\begin{aligned} \$ session['id'];
        $query = SELECT FROM $retailer WHERE Kode Retailer ←
                $kode retailer;
        $result = query ($query) ← $koneksi;
        WHILE $row = $result → fetch assoc(){
             $nama retailer ← $row(Nama Retailer)
             $alamat ← $row(Alamat)
             $no telepon ← $row(No Telepon)
             $npwp ← $row(NPWP)
             $nama npwp ← $row(Nama NPWP)
             $id salesman ← $row(ID Salesman)
             $site number ← $row(Site Number)
             $agen ← $row(Agen)
             $tipe tax ← $row(Tipe Tax)
             $credit limit \(\begin{aligned} \propto \text{row}(\text{Credit Limit}) \end{aligned}\)
             $top \leftarrow $row(TOP)
             $koneksi → close
```

5.2.4 Implementasi Vendor Managed Inventory

Implementasi kebijakan *vendor managed inventory* pada sistem informasi diterapkan dengan implementasi logika perhitungan *reorder point* (ROP) dari produk outlet masingmasing retailer. Produk yang jumlah persediaanya kurang dari sama dengan jumlah ROP yang telah dihitung pada sistem informasi akan diberitahukan kepada retailer untuk dilakukan pemesanan. Logika perhitungan ROP yang diterapkan pada sistem informasi berdasarkan pada nilai penjualan tertinggi produk sebagai nilai safety stock dari produk serta nilai penjualan rata-rata produk sebagai demand rate. Masa tenggang atau lead time yang digunakan pada perhitungan ROP pada sistem informasi ini yaitu selama 1 (satu) hari berdasarkan waktu pengiriman yang ingin diterapkan oleh PT IGM Malang dalam proses bisnisnya. Contoh tampilan implementasi VMI diantaranya yaitu informasi persediaan retailer dapat diketahui oleh logistik PT IGM Malang yang dapat dilihat pada Gambar 5.40 dan pemberitahuan kepada retailer untuk melakukan pemesanan atas produk yang jumlahnya sama dengan atau kurang dari ROP dapat dilihat pada Gambar 5.41.



Gambar 5.40 Tampilan persediaan retailer oleh logistik



Gambar 5.41 Tampilan pemberitahuan bagi retailer untuk melakukan pemesanan

5.3 Pengujian

Tahapan terakhir pada perancangan sistem informasi adalah dengan melakukan pengujian terhadap sistem informasi yang telah diimplementasikan. Tahap pengujian memiliki fungsi untuk mengetahui sistem informasi yang telah dibuat telah sesuai dengan kebutuhan sistem informasi yang diinginkan. Tahap ini terdiri dari tiga pengujian yakni uji verifikasi, uji validasi, dan uji *prototype* jalannya sistem informasi. Penjelasan detail dari masing-masing pengujian dijelaskan sebagai berikut:

5.3.1 Uji Verifikasi

Uji verifikasi merupakan tahapan pengujian yang bertujuan untuk menguji kesesuaian desain dan ketelitian antara logika operasional model sistem informasi dengan logika

diagram alur rancangan sistem. Rancangan yang dimaksud adalah antara logika sistem informasi dengan logika desain. Uji verifikasi dilakukan dengan melihat kesesuaian antara implementasi sistem informasi dengan desain pada *database*, *user interface*, *form*, dan *report*. Berikut merupakan tahap verifikasi yang dilakukan dalam penelitian ini.

1. Verifikasi Database

Kode Produk

Field

Verifikasi *database* dilakukan dengan membandingkan desain *database* dan implementasi *database*. Gambar 5.42 menampilkan uji verifikasi pada salah satu *database* yaitu pada tabel produk.

Field Size / Format

12

Key

Primary key

N	ama_Produk Varchan		Varchar		50		
K	emasan			Varchar	30		
Η	arga			Integer	15		
#	Name		Туре	Collation			
1	Kode_F	Produk 🔑	bigint(12)				
2	Nama_l	Produk	varchar(50)	latin1_swedish_ci			
3	Kemasa	an	varchar(30)	latin1_swedish_ci			
4	Harga		int(10)				
Kod	e_Produk	Nama_Prod	luk		Ken	nasan	Harga
1803	08054653	DI ANSWER	- HIV 1/2 Antib	ody Device	40 T	ests/Box	700000
1803	08054654	DI HBsAB (A	Antibody) Device	9	Box	40 Test	700000
1803	08054655	DI Multi Dru	g 5 Panel (AMP	/MET/MOP/THC/BZO)	Box	25 Test	1575000
1803	08054656	DI Multi Dru	g 6 Panel (AMP	/MET/MOP/THC/BZO/COC)	Box	25 Test	1793750
1803	08054657	Smart Diagn	ostic One Step	Amphetamine Cassette	Dus	, Isi 25 @ 1 Cassette	360500
1803	08054658	Smart Diagn	ostic One Step I	Benzodiazephine Cassette	Dus	, Isi 25 @ 1 Cassette	360500
1803	08054659	Smart Diagn	ostic One Step	Cocaine Cassette	Dus	, Isi 25 @ 1 Cassette	360500
1803	08054660	Smart Diagn	ostic One Step I	OB Cassette	Dus	, Isi 20 @ 1 Cassette	700000
1803	08054661	Smart Diagn	ostic One Step I	HbsAg Strip	Dus	, Isi 25 @ 1 Strip	297500
1803	08054662	Smart Diagn	ostic One Step I	HCG Cassette	Dus	, Isi 25 @ 1 Cassette	245000
1803	08054663	Smart Diagn	ostic One Step I	HCV Cassette	Dus	, Isi 25 @ 1 Cassette	525000
1803	08054664	Smart Diagn	ostic One Step I	HCV Strip	Dus	, Isi 25 @ 1 Strip	367500
1803	08054665	Smart Diagn	ostic One Step I	Morphine Cassette	Dus	, Isi 25 @ 1 Cassette	360500
1803	80308054666 Lactodia			Poly	bag @ 10 Sachet	35600	
1803	08054667	OBH INDO F	Plus Sirop		Boto	ol 100 ml (single box)	8580
1803	08054668	Urispas 200	mg Tab Sal Sel		Kota	ak 30	135725
1803	08054669	Jamsi			Boto	ol 100 ml	72000
1803	08054670	Madedem			Boto	ol 30 Kapsul	72000

Data Type

Big Integer

Gambar 5.42 Verfikasi database produk

Berdasarkan perbandingan antara *database* yang dirancang dengan *database* yang dibuat telah sesuai baik nama dan jumlah kolomnya. Hasil perbandingan antara keduanya dapat dikatakan memenuhi uji verifikasi *database*.

2. Verifikasi Form

Verifikasi *form* dilakukan dengan membandingkan *desain form* dengan desain *database* pada sistem informasi. Pada uji verifikasi ini dibandingkan *form* yang ditampilkan pada gambar 5.43.

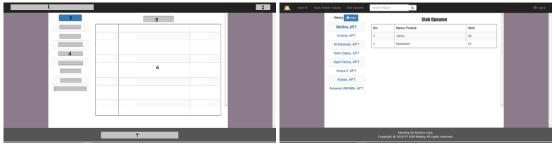


Gambar 5.43 Verfikasi form transaksi

Hasil verifikasi pada Gambar 5.43 dapat diketahui jika implementasi *form* memiliki *header*, menu, dan *form input* sesuai dengan rancangan awal.

3. Verifikasi Report

Verifikasi *report* dilakukan dengan membandingkan hasil implemantasi *report* dengan desain *user interface report*. Gambar 5.44 menampilkan uji verifikasi pada *report*.



Gambar 5.44 Verfikasi report

Hasil yang ditampilkan pada Gambar 5.44 menunjukan jika *report* yang dihasilkan sesuai dengan desain awal yang dilakukan.

4. Verifikasi algoritma

Pengujian ini dilakukan untuk proses pemeriksaan desain dan ketelitian antara logika operasional model (program komputer) dengan model konseptual (diagram alur dan asusmsi) ke dalam bahasa pemrograman secara benar. Berdasarkan pada perancangan *data modelling* yang telah dibuat sebelumnya telah dijelaskan bahwa sistem informasi dirancang untuk dapat memasukan data ke dalam sistem informasi sehingga pada implementasi *pseudocode* dijalaskan bahwa sistem informasi manajemen industri farmasi dapat memasukan data dan menyimpannya pada *database*. Hal ini menujukkan bahwa rancangan data proses telah sesuai dengan implementasi *pseudocode*.

5. Verifikasi penerapan vendor managed inventory (VMI)

Pengujian ini dilakukan dengan proses pemeriksaan apakah hasil yang didapatkan dari logika perhitungan *reorder point* yang diterapkan pada sistem informasi telah sesuai dengan model konseptual. Berdasarkan implementasi penerapan logika perhitungan *reorder point* pada sistem informasi dengan rancangan menunjukkan bahwa sistem informasi telah sesuai dengan implementasi kebijakan *vendor managed inventory*.

Uji verifikasi yang dilakukan secara keseluruhan menunjukan jika sistem informasi yang telah diimplemenasikan telah sesuai dengan desain yang dirancang. Sehingga sistem informasi yang dibuat telah memenuhi tahap uji verifikasi.

5.3.2 Uji Validasi

Validasi merupakan proses menguji apakah model konseptual dari sistem informasi dapat merepresentasikan kebutuhan *user* pada sistem nyata. Tahap yang dilakukan pada pengujian ini dengan pemerikasaan kesesuaian sistem informasi dengan *system requirement checklist* (SRC) berdasarkan analisis rancangan sistem yang telah dibuat. Tabel 5.7 menampilkan kebutuhan sistem informasi yang telah terpenuhi.

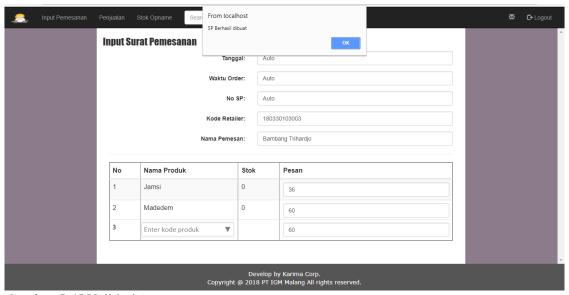
Tabel 5.7 Validasi *System Requirement Checklist* (SRC)

Kategori	Requirement Checklist (SRC) Validasi
Input	Retailer: Sistem informasi dapat memasukkan data-data sesuai dengan yang dibutuhkan seperti profil retailer, transaksi produk stok pada outlet, data pemesan, dan data jumlah pemesanan produk. Admin: Sistem informasi dapat memasukkan data-data sesuai dengan yang dibutuhkan seperti data produk, data salesman, data fakturis, data admin, data petugas gudang, data inkaso, data logistik, dan data petugas pengiriman. Logistik: Sistem informasi dapat memasukkan data-data sesuai dengan yang dibutuhkan seperti transaksi stok produk pada gudang dan data petugas gudang dan logistik. Fakturis: Sistem informasi dapat memasukkan data-data sesuai dengan yang dibutuhkan seperti data fakturis, petugas pengiriman, jatuh tempo penagihan dan data penerima pesanan. Inkaso: Sistem informasi dapat memasukkan data-data sesuai dengan yang dibutuhkan seperti data fakturis, petugas pengiriman, jatuh tempo penagihan dan data penerima pesanan. Inkaso: Sistem informasi dapat memasukkan data-data sesuai dengan yang dibutuhkan seperti TOP, penerima contra bond dan konfirmasi pembayaran.
Output	Contoh uji validasi <i>input</i> dapat dilihat pada Gambar 5.45. Retailer: Sistem informasi dapat menghasilkan informasi seperti informasi profil retailer, jumlah persediaan produk pada outlet, status persediaan produk retailer lain dan surat pemesanan serta laporan yang dapat dilakukan unduh maupun cetak. Admin: Sistem informasi dapat menghasilkan informasi seperti informasi profil retailer, produk, jumlah persediaan produk retailer, salesman, fakturis, admin, petugas gudang, inkaso, logistik, pengiriman dan surat pemesanan serta laporan yang dapat dilakukan unduh maupun cetak. Logistik: Sistem informasi dapat menghasilkan informasi seperti jumlah persediaan produk retailer, petugas gudang, logistik dan <i>picking list</i> serta laporan yang dapat dilakukan unduh maupun cetak. Fakturis: Sistem informasi dapat menghasilkan informasi seperti informasi fakturis, petugas pengiriman dan faktur penjualan serta laporan yang dapat dilakukan unduh maupun cetak. Inkaso: Sistem informasi dapat menghasilkan informasi seperti informasi salesman, inkaso, <i>contra bond receipt</i> , retailer pada masa penagihan serta laporan yang dapat dilakukan unduh maupun cetak. Contoh uji validasi <i>output</i> dapat dilihat pada Gambar 5.46.

Kategori	Validasi
Process	Sistem baru dapat melakukan pencarian data maupun laporan dengan memberikan informasi lengkap mengenai laporan yang telah dilakukan perngarsipan. Selain itu laporan juga dapat dicetak untuk keperluan arsip. Uji validasi ini dapat dilihat pada Gambar 5.47 dan Gambar 5.48. Sistem baru dapat melakukan perhitungan <i>reorder point</i> untuk setiap produk retailer. Produk yang jumlah persediaannya kurang dari sama dengan ROP akan diberitahukan kepada retailer dalam bentuk notifikasi untuk dapat dilakukan pemesanan. Uji validasi dapat dilihat pada Gambar 5.49. Sistem informasi memiliki fitur untuk mengetahui ketersediaan produk pada retailer lain. Seperti halnya pada proses pencarian. Adapun uji validasi dapat dilihat pada Gambar 5.47. Sistem dapat menampilkan transaksi yang telah diselesaikan oleh pengguna sebelumnya untuk selanjutkan dikerjakan oleh pengguna sesuai dengan proses bisnis. Sebagai contoh, surat pemesanan yang telah dilakukan konfirmasi dapat segera diketahui oleh logistik untuk dilakukan proses selanjutnya. Uji validasi dapat dilihat pada Gambar 4.50.
Performance	Sistem informasi dapat bekerja 24 jam perhari. Hal ini dikarenakan sistem menggunakan teknologi berbasis internet yang bisa diakses setiap saat.
Control	Akses ke dalam sistem harus melalui <i>login</i> terlebih dahulu sehingga untuk mengaksesnya diperlukan <i>username</i> dan <i>password</i> . Selain itu masing-masing <i>user</i> punya batasan dalam akses sistem in <i>form</i> asi. Uji validasi <i>control</i> dapat dilihat pada Gambar 5.51.

Hasil uji validasi seperti yang telah dijabarkan pada tabel validasi SRC adalah sebagai berikut.

1. Validasi Input

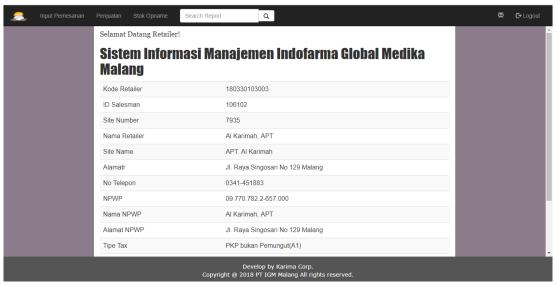


Gambar 5.45 Validasi input

Validasi *input* dilakukan dengan memasukan data pada sistem informasi. Halaman yang dapat dilakukan input diantaranya yaitu halaman master data, transaksi gudang-retailer dan transaksi independen.

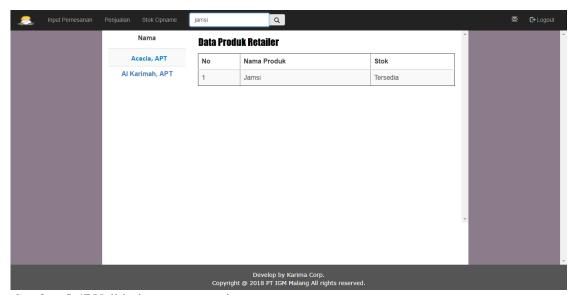
2. Validasi Output

Validasi *output* yakni dengan melakukan suatu proses menampilkan kembali data yang telah dimasukkan sebelumnya. Contohnya pada data profil retailer.

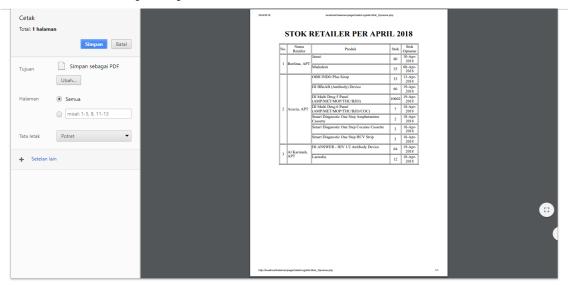


Gambar 5.46 Validasi output

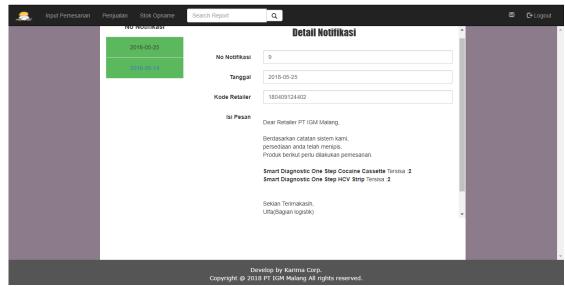
3. Validasi Process



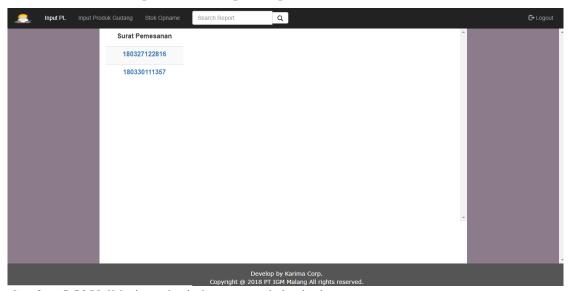
Gambar 5.47 Validasi proses pencarian



Gambar 5.48 Validasi proses cetak laporan



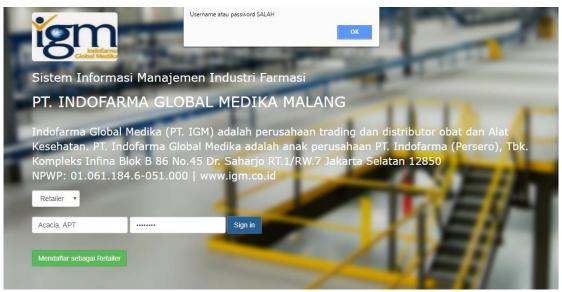
Gambar 5.49 Validasi pemberitahuan produk pesanan berdasarkan ROP



Gambar 5.50 Validasi pemberitahuan transaksi selanjutnya

Validasi proses dilakukan dengan melakukan pengecekkan pada sistem informasi yang telah sesuai dengan kebutuhan pengguna. Kebutuhan yang dapat dilakukan pada validasi proses diantaranya sistem informasi dapat melakukan proses pencarian, dapat melakukan proses cetak, dapat melakukan perhitungan ROP dan menampilkan pemberitahuan produk pemesanan, serta dapat menampilkan transaksi yang akan dilakukan selanjutnya oleh pengguna. Pada proses pencarian, tidak hanya dapat dilakukan untuk melakukan pencarian dokumen pada sistem informasi saja namun juga dapat dilakukan pencarian atas produk yang tersedia pada retailer lain sehingga dapat memudahkan retailer dalam melakukan proses transaksi penjualan serta meningkatkan proses kerjasama antar retialer.

4. Validasi Control



Gambar 5.51 Validasi control

Validasi *control* yang pada Gambar 5.47 menunjukkan perlunya *username* dan *password* untuk mengakses sistem informasi manajemen serta menampilkan halaman selanjutnya sesuai dengan batasan akses dari masing-masing *user*.

5.3.3 Uji Prototype

Uji *Prototype* merupakan pengujian implementasi berdasarkan rencana awal pembuatan sistem informasi. Pengujian ini dilakukan untuk mengetahui apakah sistem informasi dapat berjalan dengan semestinya tanpa adanya kesalahan sistem. Tahap ini dilakukan untuk menjadi solusi dari permasalah dan kelemahan yang ada pada sistem informasi industri farmasi PT Indofarma Global Medika Malang. Uji coba dilakukan dengan melakukan *running* sistem informasi sesuai dengan proses yang ada dan menganalisis apakah PIECES yang telah dibuat sebelumnya dapat terjawab oleh sistem baru yang telah dibuat. Hasil uji ini mendeskripsikan kelebihan sistem baru dibandingkan sistem lama berdasarkan analisis perbandingan PIECES sistem lama dan sistem baru dapat dilihat pada Tabel 5.8.

Tabel 5.8 Uji *Prototype* PIECES

Jenis PIECES Sistem Saat Ini Analisa		PIECES dengan Sistem Informasi
Performance	 Keluaran (throughput) Proses pemesanan yang dilakukan oleh retailer bergantung pada kapan kunjungan salesman dilakukan Pemesanan melalui telepon hanya dapat dilakukan satu transaksi pemesanan dari satu retailer dalam satu waktu 	Keluaran (throughput) 1. Proses pemesanan dilakukan melalui sistem informasi untuk produk outlet yang jumlahnya sama dengan jumlah ROP 2. Pemesanan dari retailer dapat diproses setelah dilakukan konfirmasi oleh admin

Jenis Analisa	PIECES Sistem Saat Ini	PIECES dengan Sistem Informasi		
	Waktu layan (response time) Terdapat selisih waktu yang cukup lama (1-9 hari) dalam memenuhi pesanan yang mana seharusnya pemesanan yang masuk diharapkan dapat dilakukan pengiriman pada hari itu juga Output Terdapat waktu yang cukup lama antara pemesanan hingga pengiriman pesanan sampai pada tempat tujuan Informasi stok kosong kurang memadai karena hanya berupa buku untuk stok kosong seluruh produk dari berbagai distributor Informasi pemesanan dapat diberikan oleh dua pihak berbeda (retailer dan salesman) menyebabkan data tidak valid dan redundan	Waktu layan (response time) Pemesanan dapat segera diproses karena langsung dapat diterima oleh logistik melalui sistem informasi setelah dilakukan konfirmasi tanpa perantara salesman Output 1. Pemesanan dilakukan langsung saat produk telah mencapai ROP melalui sistem informasi sehingga membutuhkan waktu yang lebih sedikit. 2. Informasi produk pada retailer lebih terorganisir dan dapat diketahui dengan mudah oleh logistik sehingga dapat segera dilakukan pemesanan saat retailer menerima pemberitahuan. 3. Sistem informasi diperbarui secara terus menerus sehingga informasi yang terekam valid dan tidak redundan.		
Information	 Input Pemesanan dilakukan melalui salesman atau dengan menelpon admin PT IGM Malang sehingga memungkinkan redundansi data Informasi pemesanan kurang jelas karena hanya berupa lembaran surat pemesanan yang dibawa oleh salesman (masing-masing retailer memiliki format yang berbeda-beda) atau surat pemesanan sementara untuk pemesanan melalui telepon yang membutuhkan konfirmasi langsung berupa tanda tangan pemesan 	Input 1. Pemesanan hanya dilakukan melalui sistem informasi sehingga tidak mengakibatkan redudansi data 2. Informasi pemesanan telah dilakukan standarisasi informasi yang diperlukan		
	1. Informasi pemesanan yang diterima via telepon ditulis kembali secara manual oleh bagian administrasi pada surat pemesanan sementara 2. Bentuk penyajian data stok pada apotek untuk dipesan hanya tercatat pada buku defecta yang ditulis oleh petugas apotek	Store data 1. Informasi pemesanan terekam secara otomatis pada sistem informasi 2. Seluruh kegiatan transaksi produk (masuk dan keluar) baik pada gudang maupun retailer terekam dalam sistem informasi		
Economy	Biaya (cost) 1. Terdapat biaya transportasi salesman dalam pengambilan surat pemesanan retailer 2. Terdapat biaya untuk pemesanan via telepon baik dari pihak retailer maupun salesman karena admin akan melakukan konfirmasi pesanan kepada salesman	Biaya (cost) 1. Pemesanan dilakukan melalui sistem informasi sehingga mengurangi biaya transportasi salesman 2. biaya penggunaan telepon digantikan menjadi biaya internet yang lebih murah		
Control	Keuntungan (profit) Jumlah pemesanan bergantung pada salesman yang melakukan kunjungan pada retailer sehingga apabila salesman jarang berkunjung maka jumlah pemesanan akan menurun Sekuritas Rendah	Keuntungan (profit) Pemesanan dapat dilakukan sewaktuwaktu saat persediaan melewati safety stock sehingga dapat menekan opportunity cost pada outlet		
Comio	Deltarius Renaun			

Jenis Analisa	PIECES Sistem Saat Ini	PIECES dengan Sistem Informasi
	 Data tidak konsisten disimpan dalam file atau <i>database</i>. Hanya data pemesanan pada buku dan sistem informasi gudang distributor Buku stok produk kosong pada apotek dapat diakses oleh salesman dari seluruh distributor yang memasok produk pada retailer tersebut 	 Data tersimpan pada <i>database</i> sistem informasi yang dapat diakses oleh pengguna menggunakan <i>userID</i> dan <i>password</i> sesuai dengan kebutuhan dalam proses bisnis. Masing-masing retailer memiliki hak akses terhadap sistem informasi menggunakan <i>userID</i> dan <i>password</i> sehingga informasi stok hanya dapat dilihat oleh <i>user</i> logistik dan admin IGM
Eficiency	Diperlukan upaya lebih untuk melakukan pencatatan dan pencarian dikarenakan banyaknya data dan sulitnya dilakukan update data Pencatatan surat pemesanan melalui telepon harus dilakukan berkali-kali karena harus dicatat dalam surat pemesanan sementara terlebih dahulu oleh admin.	Data informasi yang terekam pada database lebih terorganisir dan memudahkan proses pencarian informasi Sistem informasi mudah dipahami oleh pengguna Informasi pemesanan yang telah diinput oleh retailer pada sistem informasi akan dilakukan konfirmasi dan akan diproses oleh logistik sehingga tidak memerlukan pencatatan berulang
Services	 Sistem ini tidak fleksibel dengan situasi baru seperti memperbarui informasi ketersediaan produk pesanan Sistem ini tidak fleksibel terhadap perubahan seperti berkurang atau bertambahnya stok produk dalam setiap transaksi Laporan persediaan pada tiap outlet belum terorganisir dengan baik 	 Sistem informasi akan merekam seluruh data baru dalam database Sistem informasi dapat memperbarui informasi secara terus menerus dalam 24 jam sehingga lebih fleksibel Sistem lebih kompatibel terhadap aktifitas yang dilakukan dalam proses bisnis Laporan persediaan terorganisir pada sistem informasi dan dapat diunduh dan dicetak

Berdasarkan penjelasan perbandingan analisis PIECES pada tabel dapat diketahui apabila pembuatann sistem informasi mampu menangani masalah berdasarkan analisis PIECES yaitu *performance, information, economy, control,* dan *service*. Oleh karena itu, sistem informasi yang dibuat telah sesuai dengan perencanaan awal pembuatan sistem informasi industri farmasi PT Indofarma Global Medika Malang.

5.4 Analisis Hasil Rancangan Sistem Informasi Manajemen

Proses perancangan sistem informasi manajemen industri farmasi membantu dalam proses bisnis antara distributor farmasi Indofarma Global Medika Malang dengan retailer di sekitar kota malang diantaranya apotek dan rumah sakit dengan menggunakan bahasa pemrograman PHP dan *database* MySQL. Sistem informasi yang telah selesai dirancang dilakukan pengujian sistem informasi. Hasil pengujian sistem informasi dapat menyelesaikan permasalahan yang terjadi. Permasalahan yang terjadi seperti yang telah dijelaskan pada latar belakang masalah yang selanjutnya dilakukan identifikasi masalah berdasarkan dari studi lapangan. Permasalahan tersebut dapat terjawab dengan pembuatan

sistem informasi menggunakan bahasa pemrograman PHP dan *database* MySQL berbasis internet. Internet menyebabkan sistem informasi dapat diakses dimana pun dan fleksibel terhadap perubahan yang terjadi. *Database* yang diperlukan dapat terekam pada sistem informasi manajemen. Integrasi dari *database* dapat memudahkan *user* untuk melakukan *input*, *edit*, dan pencarian data dan laporan sehingga memudahkan proses bisnis.

Sistem informasi manajemen industri farmasi yang telah dirancang memiliki tujuan untuk mempermudah distributor farmasi PT Indofarma Global Medika Malang dalam mengelola seluruh aktivitas proses bisnis dengan retailernya, dengan adanya sistem ini proses bisnis yang sebelumnya dilakukan secara manual menyebabkan data informasi tidak fleksibel, sulit diperbarui dan mengalami redudansi data. Sistem informasi dapat memudahkan aktivitas dalam proses bisnis sehingga dapat berjalan dengan cepat dan data yang terekam lebih relevan. Perancangan sistem informasi yang dibuat dapat melakukan pendataan data retailer, produk maupun karyawan PT Indofarma Global Medika Malang serta pendataan transaksi proses bisnis dalam melakukan pemesanan oleh retailer. Selain itu sistem infromasi ini juga dapat melakukan pendataan mengenai transaksi independen dari gudang maupun masing-masing retailer sehingga lebih baik dalam manajemen persediaan antara retailer dengan Indofarma Global Medika Malang.

Sistem informasi manajemen industri farmasi Indofarma Global Medika Malang dapat memberikan pengawasan terhadap persediaan produk pada masing-masing retailer sehingga dapat lebih memudahkan dalam proses perencanaan persediaan dan pengawasan terhadap persediaan pada gudang. Selain itu pengguna sebagai retailer juga dapat menerima informasi mengenai ketersediaan produk pesanan maupun penagihan pembayaran dalam proses bisnis. Masing-masing retailer dapat mengetahui kekurangan persediaan produk pada outlet sehingga dapat segera dilakukan pemesanan serta dapat mengetahui status ketersediaan produk pada retailer lain sehingga memudahkan proses transaksi terhadap customer outlet.

Berdasarkan hasil evaluasi dari perancangan sistem informasi manajemen industri Indofarma Global Medika Malang didapatkan bahwa sistem yang telah dibuat dapat mperbaiki kekurangan dari sistem lama yang telah dianalisa pada tabel PIECES. Sistem baru dapat memberikan kemudahan kepada pengguna dalam melakukan proses bisnis dari Indofarma Global Medika Malang dengan retailer dapat melakukan proses manajemen persediaan dengan lebih baik dengan diterapkannya vendor managed inventory pada sistem informasi industri farmasi. Selain itu sistem informasi manajemen industri farmasi menggunakan metode cloud computing berbasis web yang dapat lebih mudah diakses dan dipelajari pada manual book yang diberikan untuk masing-masing user.

Halaman ini sengaja dikosongkan