



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

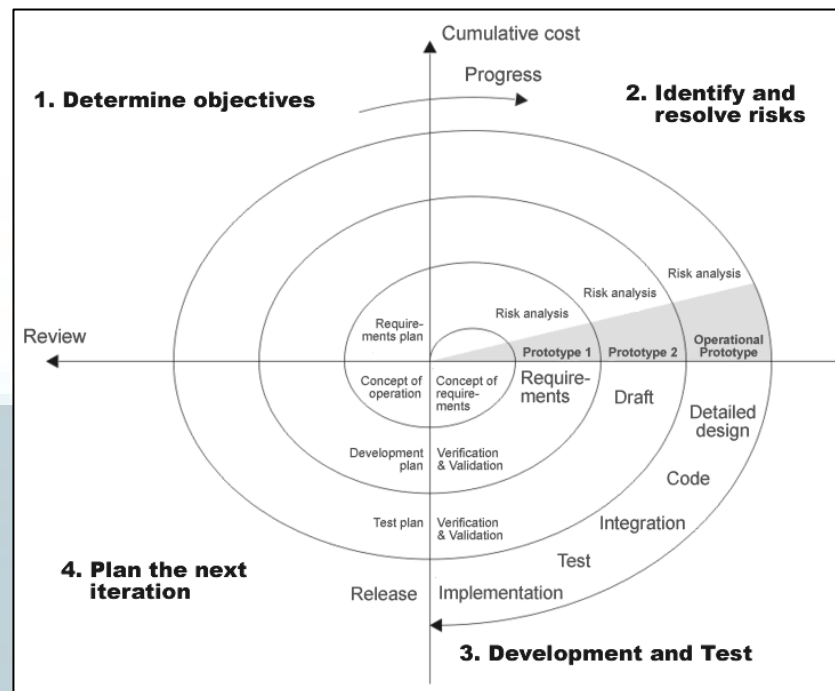
Praktek kerja magang dilakukan pada PT Trimitra Makmur Sentosa mulai dari 2 Juni 2012 hingga 6 September 2012. Posisi yang diduduki pada perusahaan adalah sebagai *Programmer*. Dimana atasan langsung dalam kerja magang ini adalah Bapak Medi selaku manager pada PT Trimitra Makmur Sentosa.

3.2 Tugas yang Dilakukan

Tugas yang dilakukan selama kerja praktek magang pada PT Trimitra Makmur Sentosa adalah membangun sistem Inventori yang akan dipakai oleh pada PT Trimitra Makmur Sentosa sendiri untuk membantu pekerjaan dalam mengelola dan mengendalikan angka persediaan bahan baku. Dimana modul yang dijelaskan dalam laporan ini adalah modul *purchase order* dan penerimaan barang.

Dalam pembangunan sistem piranti lunak ini, model pengembangan yang dipakai adalah model spiral. Pemodelan spiral adalah sebuah model pengembangan piranti lunak yang bersifat iteratif. Umumnya, model ini diimplementasikan pada proyek dengan resiko yang tinggi. Model ini menggabungkan model *waterfall* dan *prototype* (Pressman, 2010).

Model ini digunakan karena progress dari pengembangan sistem piranti lunak ini bisa dipantau oleh kedua belah pihak (user dan developer). Sehingga resiko yang didapat bisa direduksi sebelum menjadi suatu masalah yang besar yang bisa menghambat pengembangan sistem.



Gambar 3.1 Diagram Spiral Model

(Sumber : Boehm, 1986)

Perulangan 4 proses yang ditampilkan dalam gambar akan berlanjut hingga semua kebutuhan terpenuhi (Boehm, 1986). Berikut adalah aktivitas-aktivitas yang dilakukan dalam model spiral :

1. Komunikasi dengan user
2. Perencanaan
3. Analisis Resiko
4. Rekayasa
5. Pembangunan dan Pengembangan
6. Evaluasi dari User

Tahapan – tahapan aktivitas tersebut akan terus dilakukan untuk setiap bagian sistem yang dikembangkan, hingga sistem secara keseluruhan di-integrasi dan di-implementasikan.

3.3 Uraian Pelaksanaan Kerja Magang

Berikut adalah uraian detail mengenai apa saja yang dilakukan selama periode kerja magang dari proses pelaksanaan, perancangan sistem, hingga implementasi.

3.3.1 Proses Pelaksanaan

Adapun proses pelaksanaan kerja magang yang dilakukan tiap harinya dijabarkan menjadi 10 minggu kerja tersebut dilakukan seperti di bawah ini :

Minggu ke 1

- a. Mendapatkan *User Requirement* dengan wawancara kepada pihak manager, administrasi, menganalisis berbagai *form* perusahaan dan mengamati proses dan urutan pekerjaan-pekerjaan yang dilakukan pada bagian Administrasi.
- b. Menentukan komponen pendukung yang dibutuhkan dan instalasi
- c. Mendesain rancangan aplikasi dan *database*
- d. Mempelajari pemakaian komponen yang akan digunakan

Minggu ke 2

- a. Pembuatan database pendukung modul *purchase order*
- b. Pembuatan modul *purchase order* (*insert dan view*)

Minggu ke 3

- a. Pembuatan *master data supplier*
- b. Melakukan *code review* modul *purchase order*
- c. Melakukan review rancangan *database connector*

Minggu ke 4

- a. Pembuatan master data *raw material*.
- b. Pembuatan komponen aplikasi untuk *insert, update, delete* tabel *raw material*.
- c. Me-review modul *purchase order*, apakah sudah sesuai dengan *user requirement*.

Minggu ke 5

- a. Melakukan perbaikan pada modul *purchase order*
- b. Pembuatan modul penerimaan
- c. Pembuatan database pendukung modul *purchase order*.

Minggu ke 6

- a. Pembuatan sejarah pembelian, dan pembagian kategori pada master data *raw material*
- b. Membuat modul penerimaan

Minggu ke 7

- a. Menghubungkan modul penerimaan dengan *purchase order*
- b. Melakukan *code review* pada modul penerimaan

Minggu ke 8

- a. Membuat *report* penerimaan
- b. Membuat proses log-in
- c. Me-review modul penerimaan, apakah sudah sesuai dengan *user requirement*

Minggu ke 9

- a. Membuat *report* barang *reject*
- b. Me-review keseluruhan aplikasi

Minggu ke 10

- a. Memperbaiki hasil review
- b. Implementasi aplikasi

Dalam pelaksanaan praktek kerja magang, proyek sistem *inventory* PT Trimitra Makmur Sentosa, dilakukan proses lengkap dari mendapatkan *user requirement*, perancangan, desain, *implementasi code*, hingga *implementasi real*.

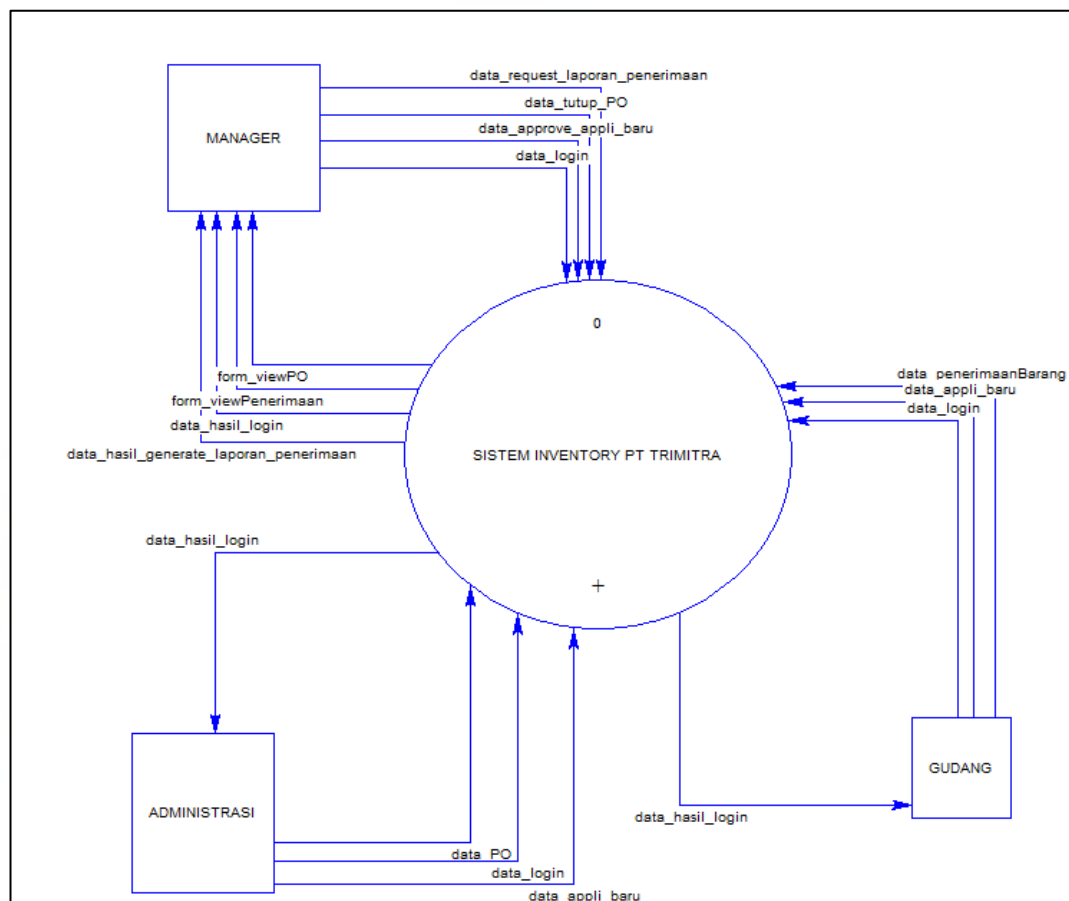
3.3.2 Perancangan Sistem

Berikut ini adalah rancangan sistem *Inventory* pada PT Trimitra Makmur Sentosa bagian modul Purchase Order dan Penerimaan Barang.

A. Data Flow Diagram

Data Flow Diagram adalah model yang digunakan untuk menunjukkan proses bisnis dan pergerakan alur data diantara bisnis proses tersebut (Wixom, 2000). Data Flow Diagram (DFD) yang digambarkan disini menjelaskan tentang proses sistem *inventory* PT Trimitra Makmur Sentosa.

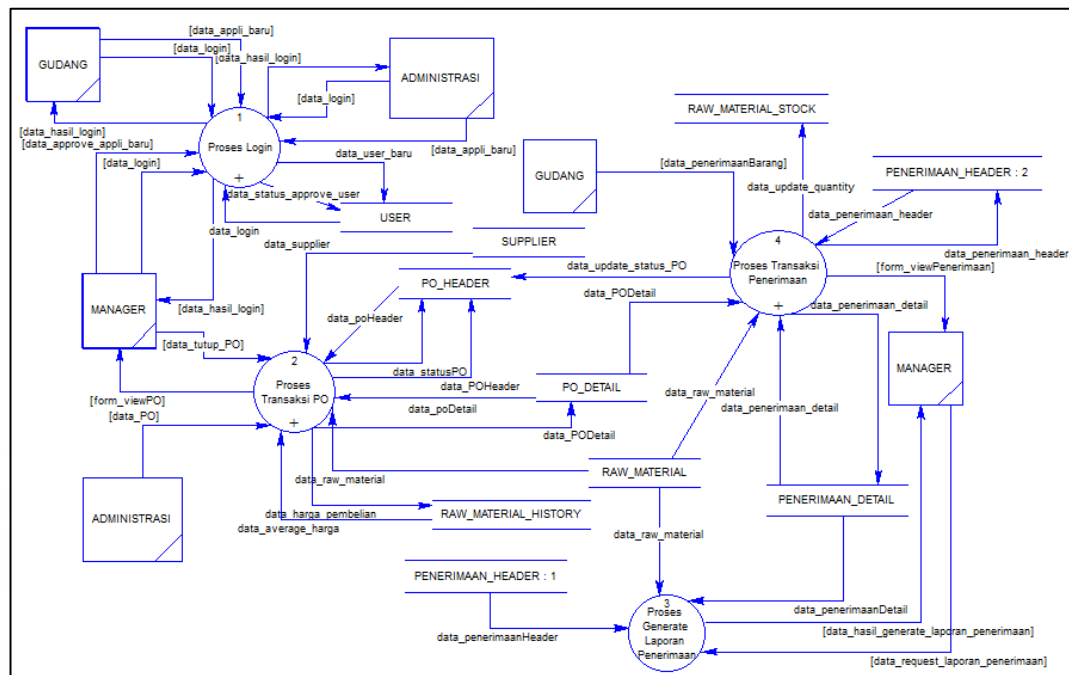
A.1 Konteks Diagram



Gambar 3.2 *Context Diagram* Sistem *Inventory* PT Trimitra

Gambar konteks diagram di atas menjelaskan sistem *inventory* PT Trimitra, yang berhubungan dengan 3 entitas, yaitu Administrasi, Manager, dan Gudang. Sistem secara umum banyak menerima *input* data-data yang nantinya akan diolah dari entitas administrasi. Dari entitas gudang, sistem akan secara umum menerima data penerimaan barang. Sedangkan manager secara umum memberikan *input* berupa *approval* dan menerima output berupa hasil olahan data dan laporan.

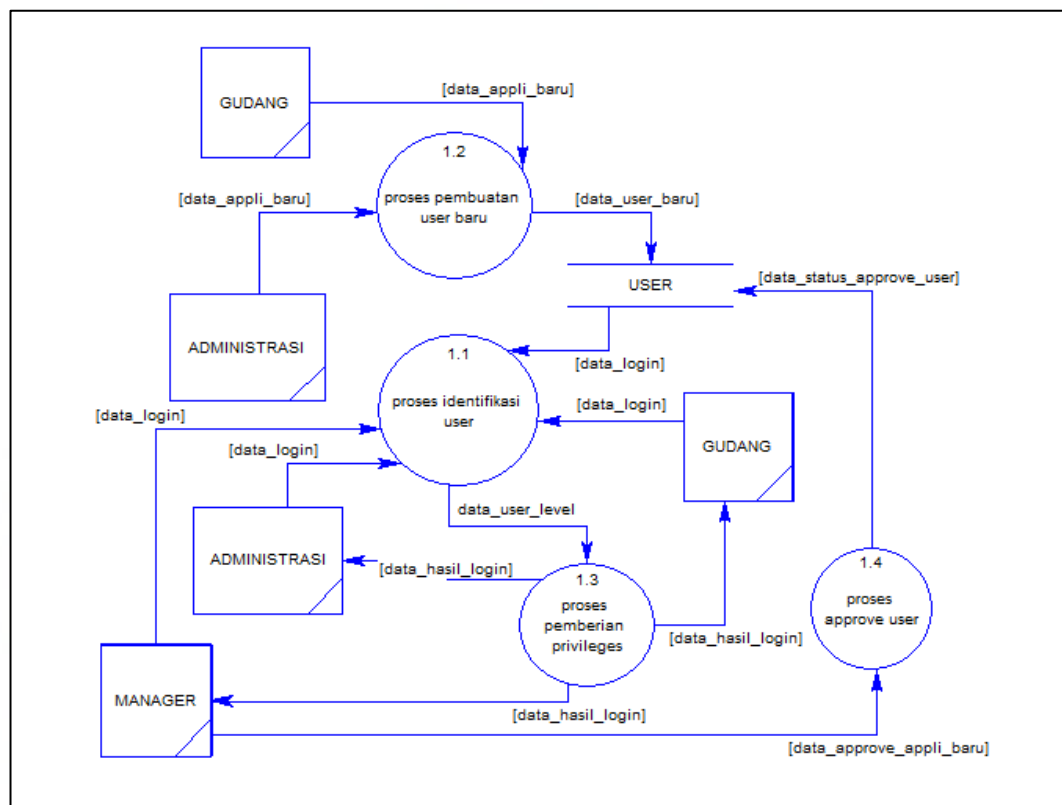
A.2 Level 0



Gambar 3.3 DFD Sistem *Inventory* PT Trimitra : Level 0

Sistem *inventory* ini (pada level 0) mencakup 4 proses yaitu proses login, transaksi *purchase order*, transaksi penerimaan barang, dan *generate* laporan penerimaan. Proses *generate* laporan penerimaan membutuhkan input data dari tabel master RAW_MATERIAL sebagai referensi, 2 tabel penerimaan dan *input request* dari entitas manager, untuk bisa menghasilkan output berupa laporan penerimaan barang, yang bisa dilihat oleh entitas manager. Sedangkan proses login, transaksi *purchase order* dan transaksi penerimaan barang bisa di *decompose* ke level 1, yang akan dijelaskan secara lebih detail.

A.3 Level 1 Subproses Login



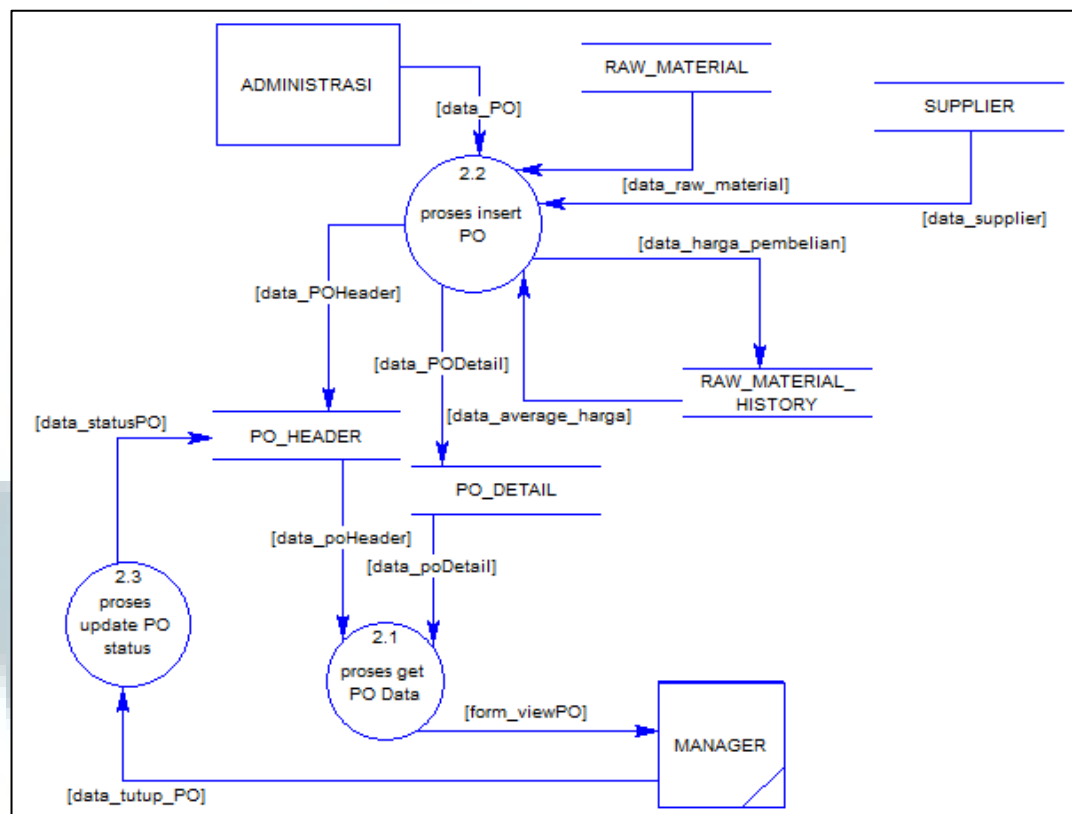
Gambar 3.4 DFD: Subproses *Login* level 1

Subproses *login* dibagi menjadi 4 subproses yaitu proses pembuatan *user* baru, *approve user*, identifikasi *user*, dan pemberian *privilege*.

Keseluruhan subproses ini dimulai dengan entitas GUDANG dan ADMINISTRASI membuat *user* baru (yang dibutuhkan untuk mengakses aplikasi) yang akan di *insert* ke dalam tabel *user*, tetapi masih belum bisa digunakan sebelum entitas MANAGER melakukan proses *approve user* yang meng-*update* status *user* pada tabel *USER*, sehingga *user* yang dibuat sudah valid untuk digunakan.

Untuk bisa *login* ke dalam aplikasi, semua entitas harus mengirimkan data *login* ke dalam aplikasi yang akan diproses dalam proses identifikasi *user*, dilanjutkan dengan proses pemberian *privilege*, yang menentukan *user* ini berhak untuk mengakses menu apa saja yang ada pada aplikasi.

A.4 Level 1 Subproses transaksi *Purchase Order*

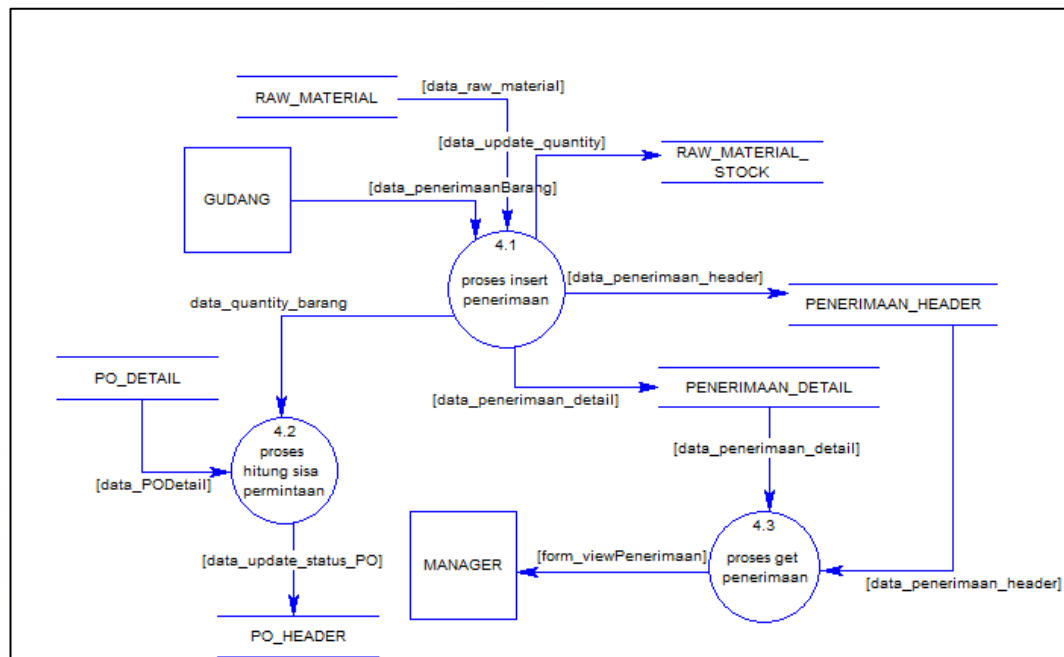


Gambar 3.5 DFD : Subproses transaksi *Purchase Order* level 1

Subproses transaksi *Purchase Order* dibagi menjadi 3 subproses yaitu proses *get PO Data*, *insert PO*, dan *update PO status*. Proses dimulai dari memasukkan data PO ke dalam proses *insert PO* oleh entitas Administrasi, yang juga mengambil referensi data raw material, supplier dan data average harga, masing masing dari tabel master RAW_MATERIAL, SUPPLIER, dan RAW_MATERIAL_HISTORY. Lalu sebagai output dari proses *insert PO* data di-*insert* ke 2 tabel PO, yaitu PO_HEADER dan PO DETAIL, selain itu proses juga meng-*insert* data_harga_pembelian ke tabel RAW_MATERIAL_HISTORY.

Setelah itu hasil *insert PO* yang sudah dimasukkan ke dalam tabel akan di lihat oleh entitas Manager, untuk di-*review* secara manual. Jika dinilai PO harus ditutup, Manager akan meng-*input* data_tutup_PO ke dalam proses update PO status, yang akan mengubah status PO dalam tabel PO_HEADER.

A.5 Level 1 Subproses transaksi Penerimaan Barang



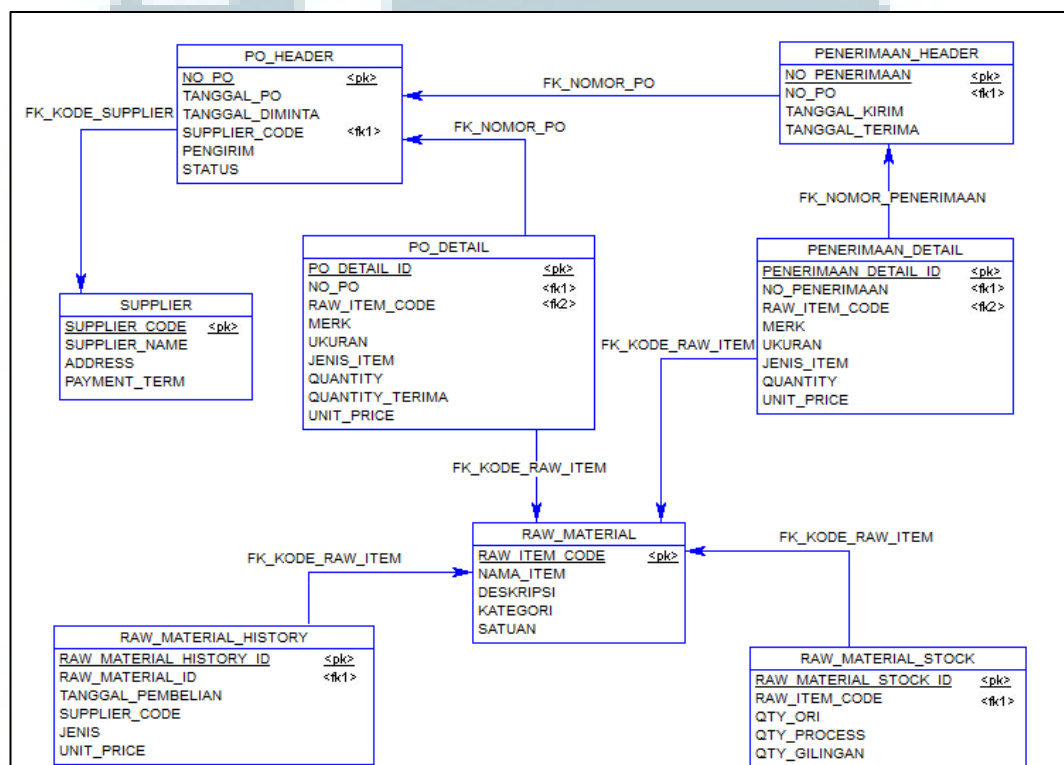
Gambar 3.6 DFD : Subproses transaksi Penerimaan Barang level 1

Subproses transaksi Penerimaan Barang menjadi 3 subproses yaitu proses *get* penerimaan, *insert* penerimaan, dan proses hitung sisa permintaan.

Subproses transaksi penerimaan barang dimulai dengan entitas gudang meng-*input* data penerimaan barang kedalam proses *insert* penerimaan dengan referensi dari tabel master RAW_MATERIAL, yang meng-*insert* ke dalam 2 tabel penerimaan yaitu PENERIMAAN_HEADER dan PENERIMAAN_DETAIL, selain itu proses ini juga meng-*update* jumlah *quantity* stok pada tabel RAW_MATERIAL_STOCK dan juga mengirimkan data_quantity_barang ke proses hitung sisa permintaan. Dengan data tersebut dan *quantity* permintaan yang diambil dari tabel PO_DETAIL, dihitung sisa permintaan untuk kemudian ditentukan status PO yang akan di-*update* ke dalam tabel PO_HEADER.

B. Entity Relationship Diagram

Entity Relationship Diagram (ERD) adalah suatu model yang mengidentifikasi entitas yang ada didalam sistem beserta relasi antar entitas-entitas tersebut (Cornell University, 2004). ERD sering digunakan dalam menggambarkan relasi dari suatu *database* dimana tiap tabel mewakili satu entitas. Berikut ini adalah gambar ERD sistem *inventory* PT Trimitra Makmur Sentosa.



Gambar 3.7 ERD Sistem *Inventory* PT Trimitra

Pada gambar ERD di atas, dapat dilihat hubungan antara tabel-tabel yang ada pada database sistem *inventory* PT Trimitra Makmur Sentosa. Diawali dengan USER yang bisa membuat form purchase order dan penerimaan barang dimana masing-masing tabelnya bernama PO_HEADER dan PENERIMAAN_HEADER.

PO_HEADER memiliki data supplier dari tabel SUPPLIER dan juga list barang-barang yang dipesan pada PO_DETAIL. Sama halnya juga dengan PENERIMAAN_HEADER yang memiliki list barang-barang yang diterima dalam PENERIMAAN_DETAIL. Selain itu, form penerimaan bisa jadi

berdasarkan PO ataupun tidak.

Kedua tabel PO_DETAIL dan PENERIMAAN DETAIL masing-masing mereferensi kepada tabel RAW_MATERIAL, yang memiliki RAW_MATERIAL_HISTORY yang berisi sejarah pembelian barang, dan RAW_MATERIAL_STOCK yang berisi quantity dari bahan baku.

Selain itu, terdapat tabel USER yang berisi informasi mengenai ID pengguna aplikasi, dimana akan dijelaskan secara lebih detail pada bagian struktur tabel.

C. Struktur Tabel

Database yang digunakan dalam sistem piranti lunak ini adalah MySQL 5.5.27. Di bawah ini adalah struktur tabel-tabel yang ada dalam database.

Tabel : PO_HEADER

Fungsi : menyimpan data setiap transaksi *Purchase Order* (transaksi pembelian bahan baku).

Primary Key : NO_PO

Foreign Key : SUPPLIER_CODE

Tabel 3.1 *Purchase Order*

Nama Field	Tipe Data	Panjang	Keterangan
NO_PO	VARCHAR	14	Primary Key untuk menunjukkan setiap form PO Format penulisan : (TM/[Tahun]/[Bulan]-[Nomor Urut PO])
TANGGAL_PO	DATE		Tanggal PO dibuat
TANGGAL_DIMINTA	DATE		Tanggal permintaan barang dibuat
SUPPLIER_CODE	VARCHAR	20	Kode supplier
PENGIRIM	VARCHAR	20	Nama pihak pengirim barang
STATUS	VARCHAR	20	Menandakan status dari PO. “proses” : berarti PO masih

Tabel 3.1 *Purchase Order* (lanjutan)

Nama Field	Tipe Data	Panjang	Keterangan
			dalam tahap menunggu penerimaan barang “diterima” : berarti barang yang di minta, semuanya sudah diterima “ditutup” : berarti barang yang diminta belum semuanya diterima tetapi, PO diputuskan untuk di tutup

Tabel : PO_DETAIL

Fungsi : Menyimpan data setiap barang-barang yang dipesan dalam sebuah *record* PO_HEADER

Primary Key : PO_DETAIL_ID

Foreign Key : NO_PO, RAW_ITEM_CODE

Tabel 3.2 *Purchase Order Detail*

Nama Field	Tipe Data	Panjang	Keterangan
PO_DETAIL_ID	INT	20	Primary key untuk menunjukkan setiap jenis item dalam sebuah form PO
NO_PO	VARCHAR	20	Foreign Key dari tabel PO_HEADER
RAW_ITEM_CODE	VARCHAR	20	Foreign Key dari tabel RAW_METERIAL
MERK	VARCHAR	100	Merk / keterangan tambahan dari barang yang dipesan
UKURAN	VARCHAR	20	Merupakan satuan dari barang yang dipesan
JENIS ITEM	VARCHAR	20	Jenis kualitas dari bahan baku : “Original” : bahan baku yang masih murni

Tabel 3.2 *Purchase Order* Detail (Lanjutan)

Nama Field	Tipe Data	Panjang	Keterangan
			“Process” : bahan baku yang hasil proses daur ulang “Gilingan” : bahan baku hasil gilingan untuk daur ulang
QUANTITY	FLOAT		Banyaknya bahan baku yang dipesan berdasarkan satuan
QUANTITY_TERIMA	FLOAT		Banyaknya bahan baku yang sudah diterima berdasarkan satuan
UNIT_PRICE	FLOAT		Harga beli bahan baku per satuan

Tabel : PENERIMAAN_HEADER

Fungsi : Menyimpan data setiap transaksi Penerimaan Barang.

Primary Key : NO_PENERIMAAN

Foreign Key : NO_PO

Tabel 3.3 Penerimaan Header

Nama Field	Tipe Data	Panjang	Keterangan
NO_PENERIMAAN	VARCHAR	14	Primary key untuk menunjukkan setiap form Penerimaan. Format penulisan : (TM/[Tahun]/[Bulan]-[Nomor Urut Penerimaan])
TANGGAL_KIRIM	DATE		Tanggal barang mulai dikirim dari supplier

Tabel 3.3 Penerimaan Header (Lanjutan)

Nama Field	Tipe Data	Panjang	Keterangan
TANGGAL_TERIMA	DATE		Tanggal barang diterima
NO_PO	VARCHAR	20	Foreign key dari tabel PO_HEADER

Tabel : PENERIMAAN_DETAIL

Fungsi : Menyimpan data setiap barang-barang yang diterima dalam sebuah PENERIMAAN_HEADER

Primary Key : PENERIMAAN_DETAIL_ID

Foreign Key : NO_PENERIMAAN, RAW_ITEM_CODE

Tabel 3.4 Penerimaan Detail

Nama Field	Tipe Data	Panjang	Keterangan
PENERIMAAN_DETAIL_ID	INT	20	Primary key untuk menunjukkan setiap jenis barang dalam sebuah form Penerimaan.
NO_PENERIMAAN	VARCHAR	20	Foreign Key dari tabel PENERIMAAN_HEADER
RAW_ITEM_CODE	VARCHAR	20	Foreign Key dari tabel RAW_METERIAL
MERK	VARCHAR	100	Merk / keterangan tambahan dari barang yang diterima
UKURAN	VARCHAR	20	Merupakan satuan dari barang yang diterima
JENIS ITEM	VARCHAR	20	Jenis kualitas dari bahan baku :
			“Original” : bahan baku yang masih murni “Process” : bahan baku yang hasil proses daur ulang “Gilingan” : bahan baku hasil gilingan untuk daur ulang

Tabel 3.4 Penerimaan Detail (Lanjutan)

Nama Field	Tipe Data	Panjang	Keterangan
QUANTITY	FLOAT		Banyaknya bahan baku yang diterima berdasarkan satuan
UNIT_PRICE	FLOAT		Harga beli bahan baku per satuan

Tabel : RAW_MATERIAL

Fungsi : Merupakan master data yang berkaitan dengan bahan baku.

Primary Key : RAW_ITEM_CODE

Foreign Key : -

Tabel 3.5 Raw Material

Nama Field	Tipe Data	Panjang	Keterangan
RAW_ITEM_CODE	VARCHAR	20	Primary key untuk menunjukkan setiap Jenis Bahan Baku
NAMA_ITEM	VARCHAR	20	Nama bahan material
DESKRIPSI	VARCHAR	100	Deskripsi tambahan mengenai bahan material
KATEGORI	VARCHAR	20	Kategori dari bahan material
SATUAN	VARCHAR	20	Satuan dari bahan material

Tabel : RAW_MATERIAL_STOCK

Fungsi : Menyimpan jumlah stok untuk setiap record RAW_MATERIAL.

Primary Key : RAW_MATERIAL_STOCK_ID

Foreign Key : RAW_ITEM_CODE

Tabel 3.6 Stok Raw Material

Nama Field	Tipe Data	Panjang	Keterangan
RAW_MATERIAL_STOCK_ID	INT	11	Primary key pada tabel RAW_MATERIAL_STOCK
RAW_ITEM_CODE	VARCHAR	20	Foreign Key untuk tabel RAW_MATERIAL

Tabel 3.6 Stok *Raw Material* (Lanjutan)

Nama Field	Tipe Data	Panjang	Keterangan
QTY_ORI	FLOAT		Jumlah stok untuk jenis Original
QTY_PROCESS	FLOAT		Jumlah stok untuk jenis Process
QTY_GILINGAN	FLOAT		Jumlah stok untuk jenis Gilingan

Tabel : RAW_MATERIAL_HISTORY

Fungsi : Menyimpan data *history* pembelian untuk setiap barang tertentu.

Primary Key : RAW_MATERIAL_HISTORY_ID

Foreign Key : RAW_MATERIAL_ID, SUPPLIER_CODE

Tabel 3.7 Sejarah *Raw Material*

Nama Field	Tipe Data	Panjang	Keterangan
RAW_MATERIAL_HISTORY_ID	INT	11	Primary Key untuk menunjukkan ID setiap transaksi pembelian sebuah barang.
RAW_ITEM_CODE	VARCHAR	20	Foreign key pada tabel RAW_MATERIAL
TANGGAL_PEMBELIAN	DATE	20	Tanggal kapan pembelian sebuah barang.
SUPPLIER_CODE	VARCHAR	20	Foreign Key dari tabel SUPPLIER, untuk menunjukkan dari siapa barang tersebut dibeli.
JENIS	VARCHAR	20	Jenis kualitas dari bahan baku : “Original” : bahan baku yang masih murni “Process” : bahan baku yang hasil proses daur ulang “Gilingan” : bahan baku hasil gilingan untuk daur ulang

Tabel 3.7 Sejarah *Raw Material* (Lanjutan)

Nama Field	Tipe Data	Panjang	Keterangan
UNIT_PRICE	FLOAT		Harga beli bahan baku per satuan

Tabel : USER

Fungsi : Menyimpan informasi yang berhubungan dengan ID pengguna.

Primary Key : ID

Foreign Key : -

Tabel 3.8 *User*

Nama Field	Tipe Data	Panjang	Keterangan
ID	VARCHAR	20	Primary Key untuk menunjukkan ID setiap pengguna
PASSWORD	VARCHAR	100	Password dari ID yang dibutuhkan dalam proses login. Di hash dengan MD5
LEVEL	VARCHAR	20	Posisi user, yang berfungsi untuk pemberian <i>privilege</i> .
ACTIVATED	BOOLEAN		Status untuk menunjukkan ID bisa dipakai untuk proses login (aktif) atau tidak.

Tabel : SUPPLIER

Fungsi : Menyimpan informasi mengenai Supplier.

Primary Key : SUPPLIER_CODE

Foreign Key : -

Tabel 3.9 *Supplier*

Nama Field	Tipe Data	Panjang	Keterangan
SUPPLIER_CODE	VARCHAR	20	Primary Key untuk menunjukkan Kode setiap supplier
SUPPLIER_NAME	VARCHAR	50	Nama perusahaan supplier
ADDRESS	VARCHAR	100	Alamat perusahaan supplier
PAYMENT_TERM	VARCHAR	20	Cara pembayaran yang sering digunakan

D. Perancangan Tatap Muka

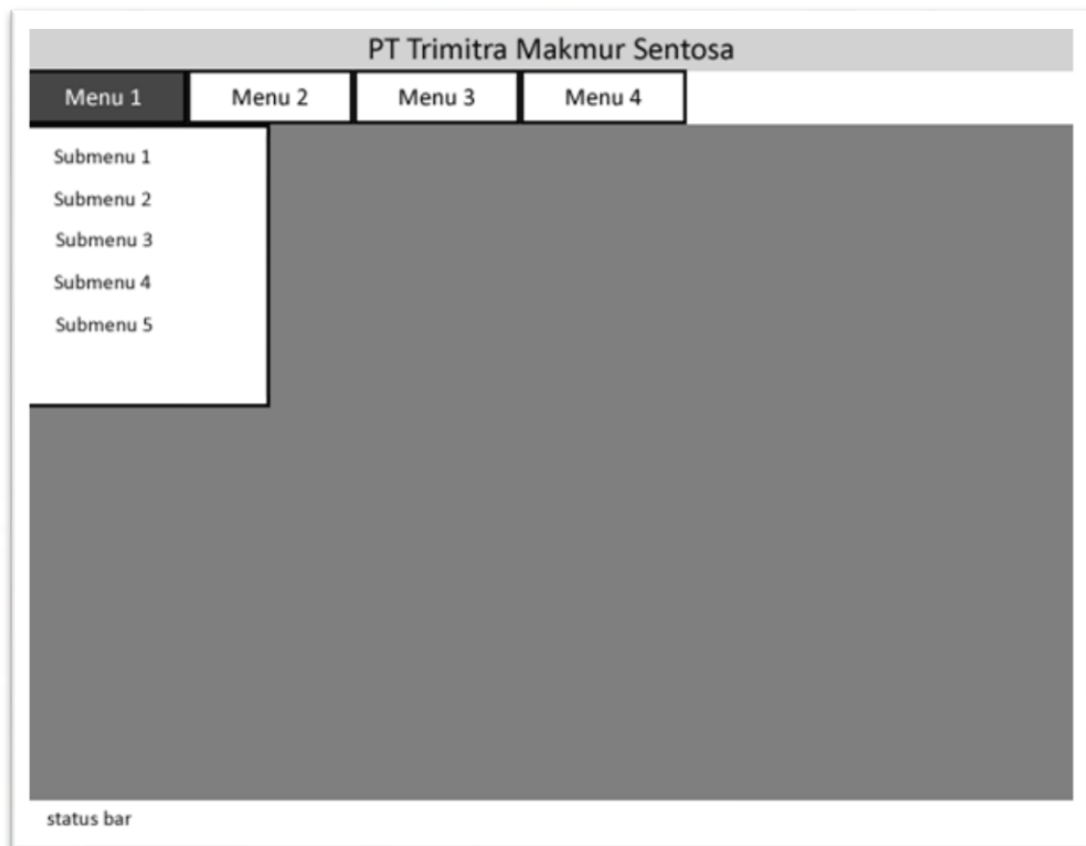
PT Trimitra Makmur Sentosa

ID

Password

Gambar 3.8 Sketsa halaman *login*

Gambar di atas menunjukkan rancangan sketsa halaman login untuk aplikasi. Terdapat 2 buah text box untuk memasukkan ID dan password supaya sistem dapat mengidentifikasi user yang menggunakan aplikasi.



Gambar 3.9 Sketsa halaman menu utama

Setelah halaman login, maka pengguna akan masuk ke halaman menu utama. Halaman menu utama berisi menu bar sebagai navigasi utama untuk ke halaman-halaman yang lain. Halaman menu utama merupakan MDI Parent sehingga memudahkan untuk navigasi atau berganti-ganti halaman dalam aplikasi.

The image shows a wireframe for a 'Form View' page. At the top is a grey header bar with the text 'Form View'. Below this, the page is divided into two main sections: 'Header' and 'Detail'. The 'Header' section contains a search box labeled 'Search' and a large empty table. To the right of the search box are two buttons labeled 'Button 1' and 'Button 2'. The 'Detail' section contains another large empty table.

Gambar 3.10 Sketsa halaman *View*

Gambar di atas merupakan sketsa rancangan dari form yang menampilkan data secara umum. Karena dari rancangan tabel *purchase order* dan penerimaan barang berbentuk header-detail, maka rancangan halaman yang dibuat berisi 2 tabel. Saat salah satu item dari tabel 1 header di select, akan muncul data-data detail pada tabel di bawahnya.

U
M
M
N

The image shows a wireframe for a 'Form Insert' window. The window has a title bar 'Form Insert'. It is divided into two main sections: 'Header' and 'Detail'. The 'Header' section contains two input fields labeled 'Field Header 1' and 'Field Header 2'. The 'Detail' section contains three input fields labeled 'Field Detail 1', 'Field Detail 2', and 'Field Detail 3', followed by a right-pointing arrow button '>>', and a large empty rectangular area. At the bottom of the window are two buttons labeled 'Button 1' and 'Button 2'.

Gambar 3.11 Sketsa halaman *Insert Form*

Gambar di atas merupakan sketsa rancangan dari form yang berguna untuk menginput data. Seperti rancangan pada *form view*, rancangan ini ditujukan untuk menginputkan data ke dalam tabel yang rancangannya bersifat header-detail.

Halaman terbagi atas 2 buah *group box*, yaitu header yang berisi field untuk pengisian data header, dan detail untuk pengisian data detail. Untuk pengisian data detail, menggunakan sistem 'add to table' yang bertujuan supaya user dapat melakukan verifikasi data-data yang telah di-input sebelum akhirnya di save ke dalam tabel.

PT Trimitra Makmur Sentosa					
Jalan Perancis, Komplek Pergudangan 8 Blok SX					
Tangerang					
Nama Laporan					
Tanggal : 01/01/2012					
Kolom 1	Kolom 2	Kolom 3	Kolom 4	Kolom 4	Kolom 5
Data1	Data2				
		Data	Data	Data	Data
		Data	Data	Data	Data
		Data	Data	Data	Data
		Data	Data	Data	Data
Data3	Data4				
		Data	Data	Data	Data
		Data	Data	Data	Data
		Data	Data	Data	Data
		Data	Data	Data	Data
Data5	Data6				
		Data	Data	Data	Data
		Data	Data	Data	Data
		Data	Data	Data	Data
		Data	Data	Data	Data
Tanggal dan waktu cetak			Halaman n		

Gambar 3.12 Sketsa laporan

Gambar di atas adalah sketsa untuk laporan. Pada bagian atas laporan terdapat nama perusahaan, alamat, logo dan disertai judul dan tanggal laporan. Pada bagian bawah akan selalu terdapat waktu laporan dicetak, dan nomor halaman laporan. Pada bagian tengah berisi data-data laporan yang dibutuhkan.

3.3.3 Implementasi

Setelah *user requirement* pada sebelumnya dikumpulkan, dianalisis dan dirancang, dibuatlah sistem *Inventory*. Sistem dibangun dengan menggunakan Microsoft Visual Studio 2008, MySQL 5.5.27 untuk database, dan MySQL ODBC 3.51 untuk koneksi database dengan aplikasi. Sistem di implementasikan pada komputer dengan sistem operasi Windows7.

a. Halaman Login



Log In

TRIMA

ID

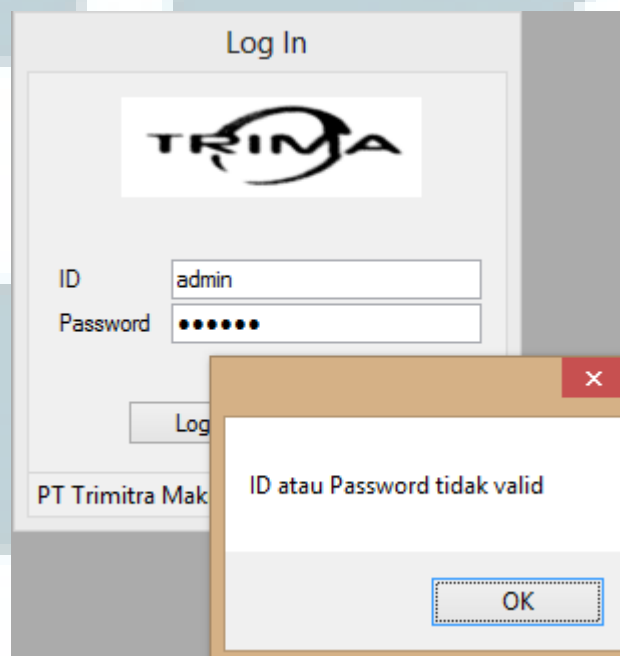
Password

Login Exit

PT Trimitra Makmur Sentosa @2012

Gambar 3.13 Tampilan halaman *Log In*

Aplikasi akan dimulai dengan memberikan halaman *Log In* kepada pengguna, untuk mengidentifikasi pengguna yang menggunakan aplikasi ini, pengguna dapat mengisi ID dan *password* pada form yang tersedia.



Log In

TRIMA

ID admin

Password ●●●●●●●

Log

PT Trimitra Mak

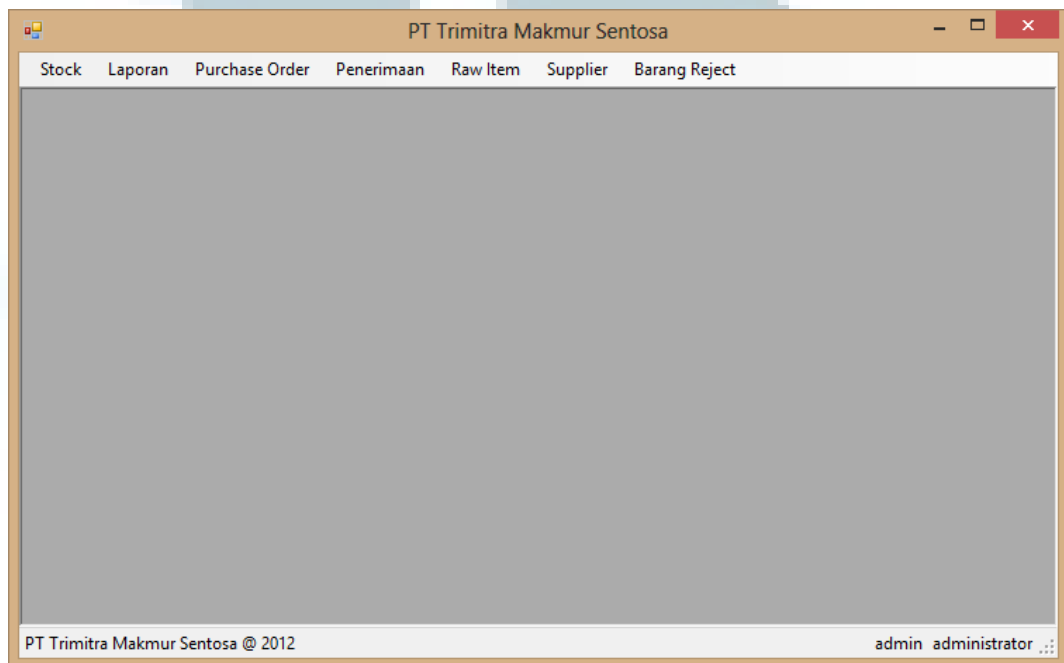
ID atau Password tidak valid

OK

Gambar 3.14 Tampilan notifikasi salah *password*

Gambar di atas merupakan tampilan jika *ID dan password* yang dimasukkan oleh pengguna tidak terdaftar pada *database*. Setelah pengguna teridentifikasi, sistem akan melakukan proses penentuan *privilege*, yang akan dilanjutkan ke tampilan halaman menu utama.

b. Halaman Menu Utama



Gambar 3.15 Tampilan halaman menu utama aplikasi

Gambar di atas merupakan tampilan menu utama, yang berisi keseluruhan navigasi dari sistem aplikasi ini. Menu utama yang dibuat berupa MDI Parent supaya user memiliki space yang luas sehingga bisa lebih mudah melakukan pengaturan dan navigasi form-form *child*-nya. Aplikasi ini juga disertai dengan *User Guide* yang dilampirkan dalam laporan ini.

c. Purchase Order

- Halaman Form pengisian *Purchase Order* baru

The screenshot shows a software window titled "Insert PO". It contains two main sections: "Header" and "Detail".

Header Section:

- No PO: PO_001
- Tanggal PO: Saturday, December 15
- Tanggal Diminta: Saturday, December 15
- Supplier: S001
- Pengirim: PT Sample 1

Detail Section:

- Kode Item: A003
- Nama Item: Raw Item C
- Merk: -
- Ukuran: kg
- Bahan: -
- Jenis Item: Gilingan
- Quantity Diminta: 30
- Unit Price: 3000
- Amount: (empty)

Table:

Kode Item	Nama Item	Merk	Ukuran	Bahan	Jenis	Quantity	Unit Price	Amount
A001	Raw Item A	-	kg	-	Original	10	1000	10000
A002	Raw Item B	-	kg	-	Original	20	2000	40000

Buttons: "Add>>", "Remove", "Save and Next PO", "Exit".

Gambar 3.16 Tampilan halaman *insert Purchase Order*

Gambar di atas merupakan tampilan implementasi form penambahan *Purchase Order*. Setelah bagian *Header* diisi, pengguna bisa mengisi tabel yang berisi barang-barang yang dipesan dengan cara menekan tombol "Add>>" setelah form detail selesai diisi. Pengisian Kode Supplier dan Kode Item, tidak di-input dengan cara diketik langsung. Di sebelah kiri dari *text box* terdapat tombol "..." untuk menampilkan form pemilihan *Supplier* atau *Item*.

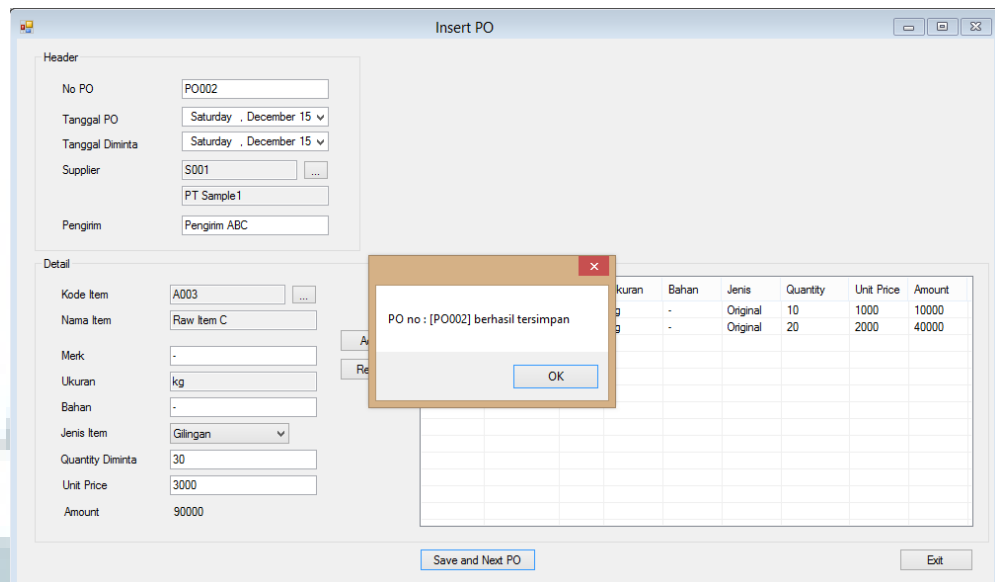
Kode Supplier	Nama Supplier	Alamat	Payment Term
S001	PT Sample1	Jl. Sample1 no. 1, Gading Serpong, Tangerang....	Cash
S002	PT Sample2	Jl. Sample2 no. 2, Gading Serpong, Tangerang....	Kredit

Gambar 3.17 Tampilan halaman pilih *Supplier*

Kode Item	Nama Item	Kategori	Satuan	Qty Ori	Qty Process	Qty Gilingan
A001	Raw Item A	Kategori P	kg	60	0	0
A002	Raw Item B	Kategori P	kg	30	0	0
A003	Raw Item C	Kategori P	kg	0	0	0

Gambar 3.18 Tampilan halaman pilih *Raw Item*

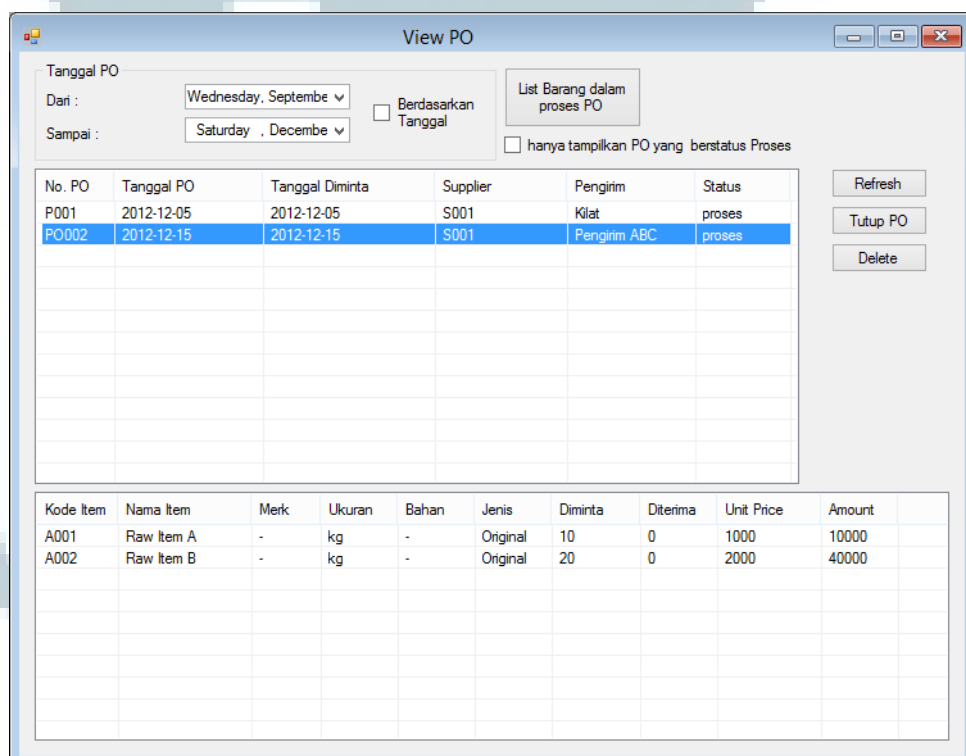
Selanjutnya pengguna bisa menekan tombol “*Save and Next PO*” untuk menyimpan data PO, sekaligus mengosongkan form seperti semula untuk bisa diisi kembali oleh pengguna.



Gambar 3.19 Tampilan notifikasi *Insert PO* berhasil

Setelah tombol "Save and Next PO" ditekan, akan muncul notifikasi pesan hasil dari proses penyimpanan.

- Halaman *Form View Purchase Order*



Gambar 3.20 Tampilan halaman View PO

Gambar di atas merupakan tampilan hasil implementasi halaman view PO. Terdapat fitur filter berdasarkan tanggal, dan status. Selain itu juga ada tombol “Tutup PO” yang bisa di gunakan oleh Manager untuk mengubah status PO.

d. Penerimaan Barang

- Halaman pengisian form Penerimaan Barang

Kode Item	Nama Item	Merk	Ukuran	Bahan	Jenis	Sisa Permintaan	Diterima	Unit Price	Amount

Gambar 3.21 Tampilan halaman *Insert Penerimaan*

Gambar di atas merupakan tampilan dari halaman pengisian form penerimaan yang dimana pengisian form ada 2 model, yaitu berdasarkan *Purchase Order* atau tidak (misalnya bahan baku yang didapat dari proses daur ulang hasil proses produksi).

Jika berdasarkan *Purchase Order*, maka “No. PO” harus diisi terlebih dahulu, dengan menekan tombol “...”.

No. PO	Tanggal PO	Tanggal Diminta	Supplier	Pengirim	Status
P001	2012-12-05	2012-12-05	S001	Kilat	proses
PO002	2012-12-15	2012-12-15	S001	Pengirim ABC	proses

Kode Item	Nama Item	Merk	Ukuran	Bahan	Jenis	Diminta	Diterima	Unit Price	Amount
A001	Raw Item A	-	kg	-	Original	100	65		

Gambar 3.22 Tampilan halaman pilih PO

Gambar di atas merupakan tampilan form pemilihan PO. *Form* ini hanya akan menampilkan *list Purchase Order* yang berstatus “proses” (yang artinya barang yang dipesan belum diterima sepenuhnya). Setelah dipilih, sistem akan membaca list barang pada *database* PO, dan mengisi tabel barang yang diterima pada halaman pengisian Penerimaan.

InsertPenerimaan

No. Form Penerimaan:

Tanggal Kirim: Saturday, December 15, 2012

Tanggal Terima: Saturday, December 15, 2012

Penerimaan

Berdasarkan PO: Ya Tidak

No. Po: PO002

Tanggal PO: 2012-12-15

Tanggal Diminta: 2012-12-15

Supplier: S001

Pengirim: Pengirim ABC

Kode Item	Nama Item	Merk	Ukuran	Bahan	Jenis	Sisa Permintaan	Diterima	Unit Price	Amount
A001	Raw Item A	-	kg	-	Original	10	0	1000	0
A002	Raw Item B	-	kg	-	Original	20	0	2000	0

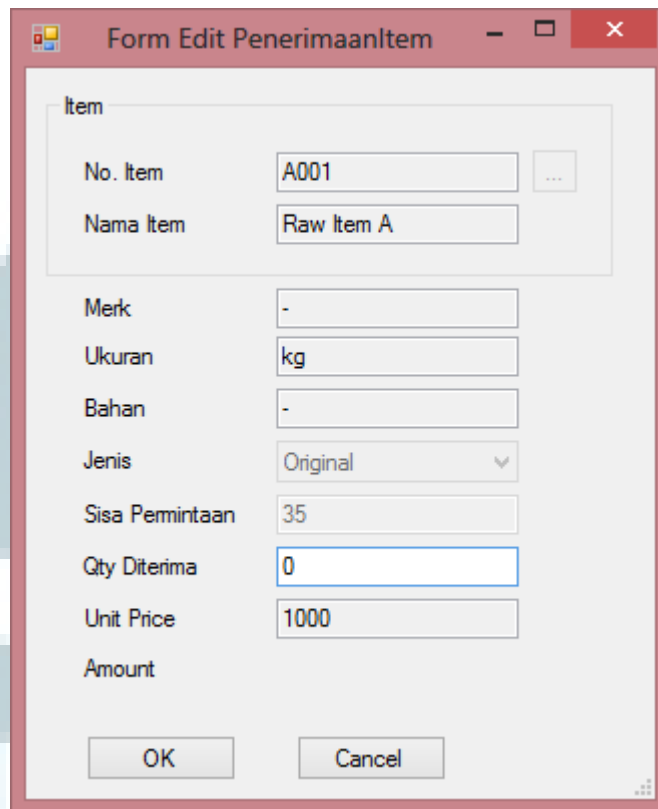
Add Delete

Save and Next Exit

Gambar 3.23 Tampilan halaman *Insert* Penerimaan (Tabel detail terisi)

Setelah itu, pengguna dapat mengubah nilai jumlah barang yang diterima dengan cara *double-click* pada record item. Jika sisa permintaan semua barang sudah habis, maka sistem akan meng-update status PO dari sebelumnya yang berstatus “proses” menjadi “diterima”.

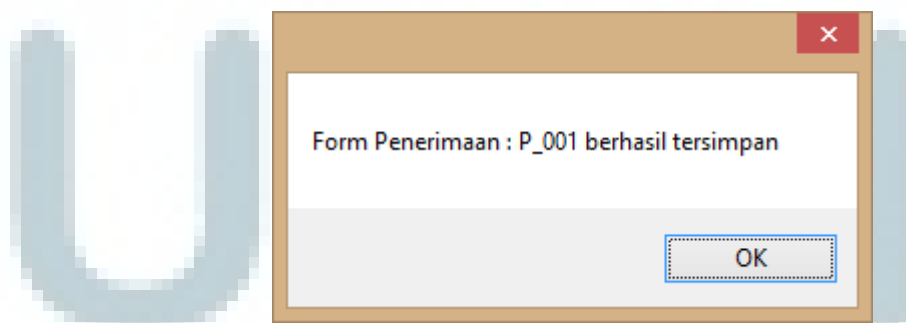
U M N



Item	
No. Item	A001
Nama Item	Raw Item A
Merk	-
Ukuran	kg
Bahan	-
Jenis	Original
Sisa Permintaan	35
Qty Diterima	0
Unit Price	1000
Amount	

Gambar 3.24 Tampilan halaman *edit quantity* penerimaan

Selanjutnya pengguna bisa menekan tombol “Save and Next” untuk menyimpan *form* penerimaan kedalam database, sekaligus mengosongkan *form* untuk siap diisi dengan data yang baru.



Gambar 3.25 Tampilan Notifikasi penyimpanan form penerimaan

- Halaman *Form View* Penerimaan Barang

The screenshot shows a window titled "View Penerimaan" with the following elements:

- Form Fields:**
 - Tanggal Terima: [Dropdown menu]
 - Dari : [Dropdown menu showing "Wednesday, Septembe"]
 - Sampai : [Dropdown menu showing "Saturday, Decembe"]
 - Berdasarkan Tanggal
- Table 1 (Goods Receipt Summary):**

No. Penerimaan	Tanggal Kirim	Tanggal Terima	No PO
P_001	2012-12-15	2012-12-15	PO002
- Buttons:** Refresh, Delete
- Table 2 (Item Details):**

Kode Item	Nama Item	Merk	Ukuran	Bahan	Quantity	Unit Price	Amount
A001	Raw Item A	-	kg	-	5	1000	5000
A002	Raw Item B	-	kg	-	20	2000	40000

Gambar di atas merupakan tampilan halaman penerimaan barang. Jika salah satu item dari tabel *header* di-select, maka tabel detail akan menampilkan data detail dari nomor penerimaan yang di-select. Selain itu pengguna juga bisa mem-filter data PO berdasarkan tanggal dengan menu diatas.

UUMN

e. Laporan

Kode Raw Material	Nama Raw Material	No. Penerimaan	Jenis Item	Quantit
A001	Raw Item A	P_001	Original	5
		P_003	Original	5
		P_002	Original	50
		Total :		60
A002	Raw Item B	P_001	Original	20
		P_003	Original	10
		Total :		30

Gambar 3.26 Tampilan Laporan Penerimaan Stok

Gambar di atas merupakan tampilan laporan penerimaan stok yang di generate dari aplikasi. Data yang ditampilkan adalah semua barang material yang diterima pada hari yang ditentukan, dan ditampilkan secara *per-item*.

3.3.4 Kendala yang Ditemukan

Kendala yang ditemukan dalam proses pembuatan sistem *inventory* PT Trimitra Makmur Sentosa yaitu :

- b. Pengembangan sistem membutuhkan waktu yang cukup lama dikarenakan seringnya perbaikan karena aplikasi dinilai tidak sesuai dengan *user requirement* pada saat *review*.
- c. Terdapat banyak proses penyimpanan ke database yang harus dilakukan secara lengkap. Salah satu contohnya ada pada modul penerimaan, dimana pada saat meng-*input* data penerimaan, selain memasukkan data ke dalam tabel penerimaan, sistem juga akan meng-*update* sisa permintaan dan status pada tabel PO. Jadi sistem harus melakukan *insert* ke-3 tabel yang berbeda,

dimana ketiganya harus dilakukan secara lengkap (atau tidak sama sekali) untuk menghindari tidak validnya data yang di input. Sehingga unsur *atomicity* sangat penting pada transaksi data tersebut.

3.3.5 Solusi atas Kendala dan Kebutuhan yang Ditemukan

Dari kendala yang ditemukan saat pengerjaan, solusi yang didapat adalah

- a. Untuk masalah pengembangan sistem yang memakan waktu, bisa diatasi dengan management pengerjaan yang baik dan terstruktur, serta lebih detail dalam mendapatkan *user requirement*.
- b. Untuk masalah *atomicity* dalam transaksi data, terdapat object '*Transaction*' untuk membantu mekanisme *roll-back* jika terjadi kesalahan dalam proses transaksi data sebelum di *commit*. Contoh perintah (dalam visual studio C# dan menggunakan koneksi MySql ODBC (Huddleston, 2008) :

```
OdbcTransaction trans = connection.BeginTransaction;  
... (melakukan transaksi) ...  
trans.Commit();
```

U
M
M
N