

BAB IV

HASIL DAN PEMBAHASAN

4.1. Perancangan Sistem

Setelah melakukan analisis, perancangan sistem adalah tahap selanjutnya untuk menghasilkan sistem yang lebih baik dengan harapan untuk meningkatkan produktifitas dan efisiensi waktu. Berdasarkan hasil analisis pada sistem yang sedang berjalan, ditemukan kelemahan mengenai pencatatan data maka penulis mengusulkan untuk membuat sebuah sistem yang baru agar permasalahan pencatatan tersebut bisa berkurang dan tidak membuang-buang waktu dalam proses yang akan dilakukan selanjutnya seperti membuat laporan dan lain-lain.

4.1.1. Tujuan perancangan sistem yang diusulkan

Tujuan utama dalam perancangan sistem yang diusulkan ini adalah untuk meningkatkan produktifitas dan efisiensi waktu pada sistem yang akan dibangun berdasarkan masalah-masalah dan hasil evaluasi sistem yang sedang berjalan. Perancangan sistem yang diusulkan ini bermaksud untuk mengetahui dan memperjelas langkah-langkah yang harus ditempuh pada saat merancang dan membangun sebuah sistem.

4.1.2. Gambaran umum sistem yang diusulkan

Pada dasarnya sistem yang akan dibangun tidak memiliki perbedaan yang sangat signifikan hanya merubah sesuatu proses pencatatan yang bersifat *manual* menjadi terkomputerisasi artinya proses bisnisnya dilakukan secara manual

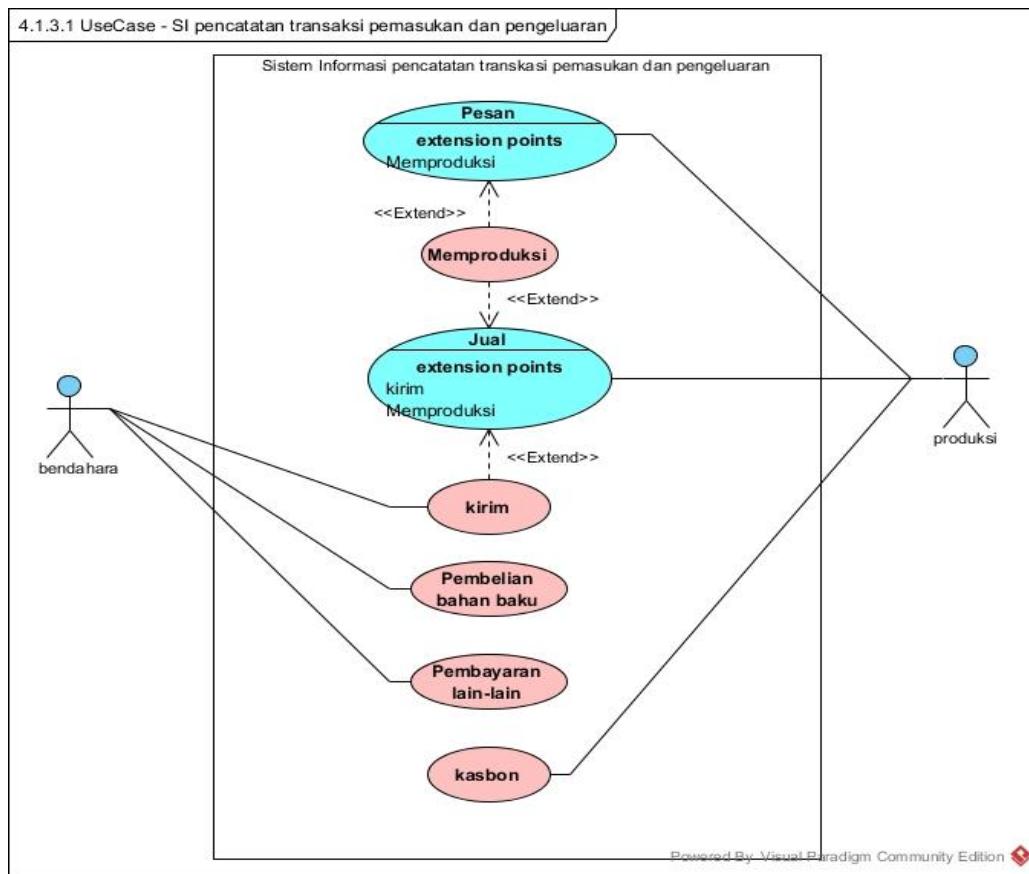
namun proses pencatatannya dilakukan secara komputerisasi seperti membuat kwitansi, membuat laporan dan lain-lain.

4.1.3. Perancangan prosedur yang diusulkan

Prosedur yang diusulkan digambarkan menggunakan *Use case diagram*, *scenario diagram* dan *activity diagram*. Berikut ini adalah perancangan prosedur yang diusulkan:

4.1.3.1. Use case diagram

Berdasarkan prosedur yang diusulkan berikut ini adalah gambaran sistem yang diusulkan berbentuk *Use case diagram*:



Gambar 4.1 Use Case Si Transaksi Pemasukan Dan Pengeluaran.

a. Definisi aktor dan deskripsinya

Berdasarkan *use case diagram* diatas berikut ini adalah definisi aktor dan keterangannya:

Tabel 4.1 Definisi Aktor Sistem Yang Diusulkan

No.	Nama Aktor	Keterangan
1.	Produksi	Bertugas untuk melayani proses penjualan dan pemesanan produk terhadap konsumen.
2.	Bendahara	Bertugas untuk yang bersangkutan dengan keuangan. Seperti berbelanja, penggajian karyawan (produksi), kasbon karyawan, biaya transportasi (kirim).

b. Definisi *use case* dan deskripsinya

Berdasarkan use case diagram berikut ini adalah definisi use case dan keterangannya:

Tabel 4.2 Definisi *Use Case* Sistem Yang Diusulkan

No.	Nama <i>use case</i>	Keterangan
1.	Pesan	Pesan adalah proses pencatatan pesanan penjualan (sebagai catatan untuk pesanan). Pesanan bisa dibatalkan
2.	Produksi	Produksi merupakan proses pencatatan produk yang diproduksi oleh karyawan.
3.	Jual	Jual adalah proses pencatatan penjualan produk.
4.	Kirim	Kirim adalah proses pencatatan pembayaran ongkos kirim untuk transaksi penjualan.
5.	Pembelian bahan baku	Pembelian adalah proses pencatatan transaksi pembelian bahan baku
6.	Pembayaran lain-lain	Pembayaran lain-lain adalah proses pencatatan transaksi pembelian listrik, pembelian pulsa, dan transaksi pembayaran tidak terduga seperti service mobil, kulkas dan lain-lain
7.	Kasbon	Kasbon adalah proses pencatatan transaksi peminjaman dana yang dilakukan oleh karyawan.

4.1.3.2. Skenario *use case*

Berdasarkan use case diagram diatas berikut ini adalah skenario *use case* atau *use case description*:

1. Skenario *use case* sistem pesan yang diusulkan.

Tabel 4.3 Skenario *Use Case* Pesan Yang Diusulkan

<i>Main scenario use case</i> pesan	
Relasi <i>use case</i>	Memproduksi
<i>Primary</i> aktor	Produksi
<i>Support</i> aktor	-
<i>Pre-condition</i>	Konsumen melakukan pemesanan melalui telephone.
<i>Normal Scenario</i>	
<i>Action</i>	
1. Kordinator karyawan bagian produksi menginput pesanan.	2. Kordinator karyawan bagian produksi mencetak pesanan penjualan dan kemudian menyerahkan kepada karyawan bagian produksi
3. Karyawan bagian produksi melakukan pengecekan kulkas.	4. Ketua bagian produksi menginformasikan kepada konsumen produk yang dipesan tersedia. 5. Karyawan bagian produksi akan melakukan <i>packaging</i>
<i>Post-condition</i>	1. Dokumen pesanan penjualan

<i>Alternative scenario use case pesan (lanjutan)</i>	
<i>Action</i>	<i>Response</i>
1. Kordinator karyawan bagian produksi menginputkan pesanan penjualan	2. Kordinator karyawan bagian produksi mencetak dokumen pesanan penjualan dan menyerahkan kepada karyawan bagian produksi.
3. Karyawan bagian produksi melakukan pengecekan dikulkas untuk pesanan penjualan.	4. Informasikan kepada konsumen produk yang tersedia tidak mencukupi.
5. Konsumen konfirmasi pesanan penjualan, menunggu produksi atau ubah pesanan	6. Jika menunggu produksi, karyawan bagian produksi akan memproduksi es krim sesuai dengan pesanan. Jika ubah pesanan informasikan jumlah produk yang tersedia. 7. Karyawan bagian produksi melakukan <i>packaging</i> .
<i>Post-condition</i>	1. Dokumen pesanan penjualan.

2. Skenario *use case* sistem memproduksi yang diusulkan.

Tabel 4.4 Skenario *Use Case* Memproduksi Yang Diusulkan

<i>Main scenario use case</i> memproduksi	
<i>Relasi use case</i>	Pesan
<i>Primary aktor</i>	Produksi
<i>Support aktor</i>	-
<i>Pre-condition</i>	Karyawan bagian produksi menerima pesanan untuk diproduksi
<i>Normal Scenario</i>	
<i>Action</i>	
1. Kordinator karyawan bagian produksi mencetak dokumen pesanan penjualan. 2. Menyerahkan kepada karyawan bagian produksi	3. Karyawan menerima dokumen pesanan penjualan.
4. Karyawan bagian produksi akan memproduksi produk sesuai dengan dokumen tersebut.	5. Kordinator karyawan bagian produksi akan menghitung berapa banyak produk yang diproduksi oleh setiap karyawan.
6. Menginputkan berapa jumlah produk yang diproduksi oleh setiap karyawan.	7. Karyawan bagian produksi menerima gaji berdasarkan hasil dari produksinya.
<i>Post-condition</i>	1. Mencetak struk penggajian karyawan bagian produksi.

3. Skenario *use case* sistem jual yang diusulkan.

Tabel 4.5 Skenario *Use Case* Sistem Jual Yang Diusulkan

<i>Main scenario use case jual</i>	
<i>Relasi use case</i>	Memproduksi, Kirim
<i>Primary aktor</i>	Produksi
<i>Support aktor</i>	-
<i>Pre-condition</i>	Dokumen pesanan penjualan.
<i>Normal Scenario</i>	
<i>Action</i>	
1. Kordinator karyawan bagian produksi menyerahkan dokumen pesanan penjualan kepada ketua bagian produksi.	2. Kordinator karyawan bagian produksi merubah pesanan penjualan menjadi penjualan produk.
3. Kordinator karyawan bagian produksi melakukan konfirmasi kepada konsumen mengenai lokasi pengiriman dan produk yang dipesan akan dikirimkan.	4. Kordinator karyawan bagian produksi bagian mencetak surat jalan untuk supir dan kwitansi penjualan. 5. Karyawan bagian produksi melakukan <i>packaging</i> .
6. Kordinator karyawan bagian produksi menyerahkan surat jalan kepada karyawan bagian supir.	7. Supir akan mengirimkan penjualan produk kepada konsumen sesuai dengan surat jalan yang telah diterima.
8. Konsumen menerima pesanan penjualan dan kwitansi	9. Konsumen melakukan pembayaran secara tunai atau kreidit.
<i>Post-condition</i>	1. Surat jalan 2. Kwitansi penjualan

<i>alternative scenario use case jual (lanjutan)</i>	
<i>Pre-condition</i>	Konsumen datang langsung
<i>Normal Scenario</i>	
<i>Action</i>	<i>Response</i>
1. Konsumen memesan produk	2. Kordinator karyawan bagian produksi menginputkan pesanan. 3. Memeriksa ketersedian produk di <i>software</i> .
4. Informasikan kepada konsumen produk yang dipesan tersedia.	5. Kordinator karyawan bagian produksi mencetak kwitansi penjualan. 6. Karyawan bagian produksi melakukan <i>packaging</i> .
7. Kordinator karyawan bagian menyerahkan kwitansi kepada konsumen	8. Konsumen membayar secara tunai atau kredit.
<i>Post-condition</i>	1. Kwitansi penjualan

<i>alternative scenario use case jual (lanjutan)</i>	
<i>Pre-condition</i>	Konsumen datang langsung
<i>Normal Scenario</i>	
<i>Action</i>	<i>Response</i>
1. Konsumen memesan produk	2. Kordinator karyawan bagian produksi menginputkan pesanan. 3. Memeriksa ketersedian produk di <i>software</i> .
4. Informasikan kepada konsumen produk yang dipesan tidak mencukupi.	5. Konsumen merubah pesanan sesuai dengan produk yang tersedia.
6. Kordinator karyawan bagian produksi mencetak kwitansi penjualan.	7. Karyawan bagian produksi melakukan <i>packaging</i> .
8. Kordinator karyawan bagian produksi menyerahkan kwitansi kepada konsumen.	9. Konsumen membayar secara tunai atau kredit.
<i>Post-condition</i>	1. Kwitansi penjualan

4. Skenario *use case* sistem kirim yang diusulkan.

Tabel 4.6 Skenario Use Case Kirim Yang Diusulkan

<i>Main scenario use case</i> kirim	
<i>Relasi use case</i>	Jual
<i>Primary aktor</i>	Supir
<i>Support aktor</i>	Produksi
<i>Pre-condition</i>	Ketua bagian produksi mencetak surat jalan
<i>Normal Scenario</i>	
<i>Action</i>	<i>Response</i>
1. Kordinator karyawan bagian produksi menyerahkan surat jalan kepada supir.	2. Supir mengirimkan pesanan penjualan berdasarkan surat jalan.
3. Konsumen menanda-tangani surat jalan yang supir berikan kepada konsumen berdasarkan pesanan penjualan yang diterima.	4. Kordinator karyawan bagian produksi mengimputkan biaya pembayaran ongkos transportasi.
<i>Post-condition</i>	1. Surat jalan telah ditanda-tangani oleh konsumen.

5. Skenario *use case* sistem pembelian lain-lain yang diusulkan.

Tabel 4.7 Skenario *Use Case* Pembayaran Lain-Lain Yang Diusulkan

<i>Main scenario use case</i> pembayaran lain-lain	
<i>Primary aktor</i>	Bendahara
<i>Support aktor</i>	-
<i>Pre-condition</i>	Kebutuhan lain-lain seperti service mobil, service kulkas dll
<i>Normal Scenario</i>	
Action	Response
1. Melakukan transaksi pembayaran lain-lain secara tunai. 2. Menerima struk pembayaran.	3. Menginput transaksi pembayaran lain-lain berdasarkan struk pembayaran.
<i>Post-condition</i>	1. Struk pembelian

6. Skenario *use case* sistem kasbon karyawan.

Tabel 4.8 Skenario *Use Case* Sistem Kasbon Yang Diusulkan

<i>Main scenario use case</i> kasbon	
<i>Primary aktor</i>	Produksi
<i>Support aktor</i>	-
<i>Pre-condition</i>	-
<i>Normal Scenario</i>	
Action	Response
1. Supir atau karyawan produksi meminjam uang ke bendahara	2. Bendahara memberikan dana pinjaman kepada karyawannya. 3. Kordinator karyawan bagian produksi menginputkan biaya yang dikeluarkan untuk kasbon karyawan.
<i>Post-condition</i>	-

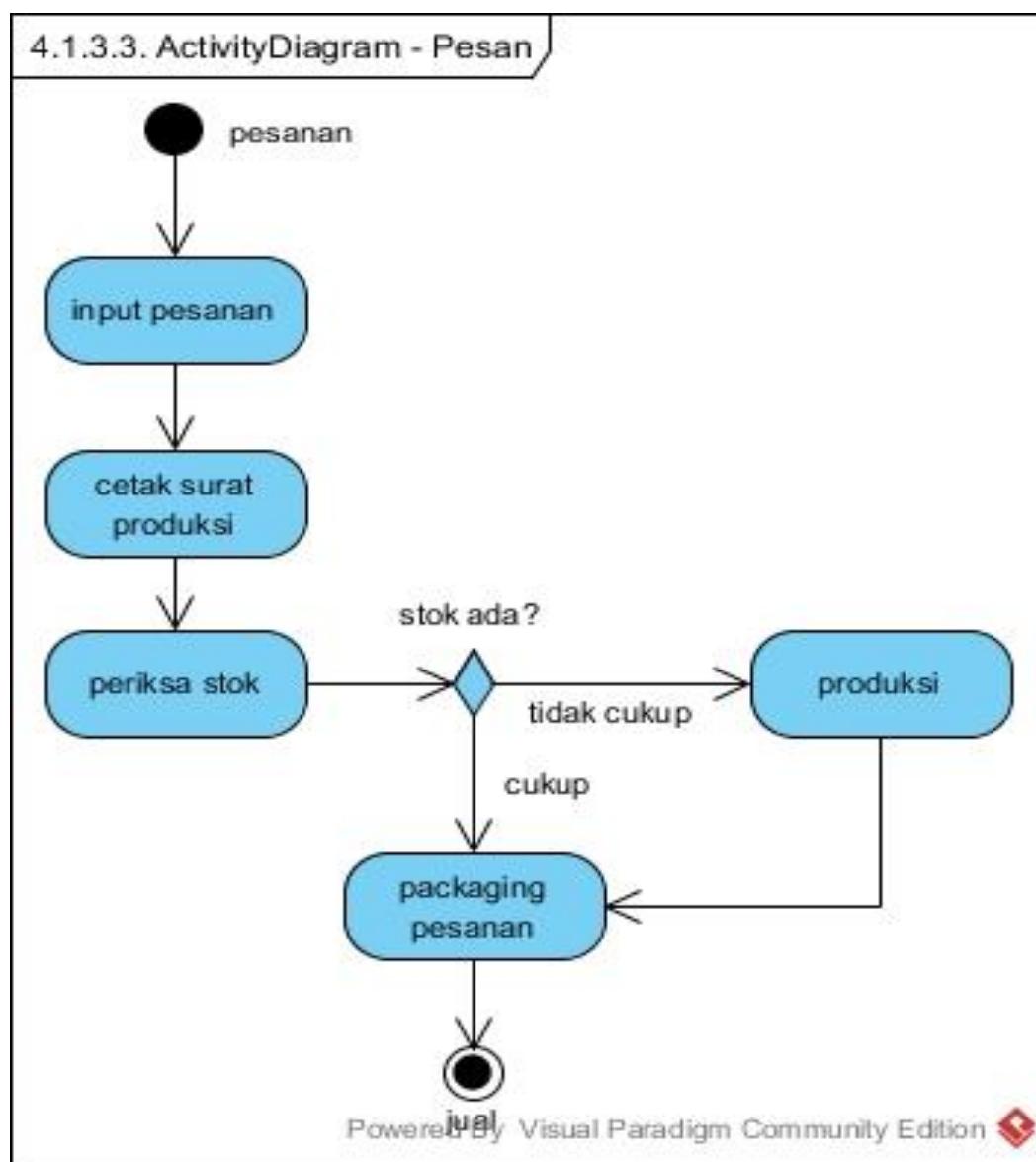
7. Skenario *use case* sistem pembelian bahan baku yang diusulkan.

<i>Main scenario use case</i> pembelian bahan baku	
<i>Primary aktor</i>	Bendahara
<i>Support aktor</i>	-
<i>Pre-condition</i>	-
<i>Normal Scenario</i>	
<i>Action</i>	
1. Bendahara melakukan transaksi pembelian kepada produsen secara tunai	
2. Bendahara menginputkan biaya yang dikeluaran untuk pembelian bahan baku berdasarkan struk pembelian.	
<i>Post-condition</i>	1. Menghasilkan struk pembelian
<i>Alternative scenario use case</i> pembelian bahan baku	
<i>Action</i>	
1. Bendahara melakukan transaksi pembelian kepada produsen secara kredit	
2. Bendahara menginputkan biaya yang dikeluaran untuk pembelian bahan baku berdasarkan struk pembelian.	
3. Bendahara melunasi pembayaran kepada suplayer pada transaksi yang akan datang.	
4. Bendahara mengubah biaya yang masih dikeluaran untuk pembelian bahan baku berdasarkan struk pembelian.	
<i>Post-condition</i>	1. Menghasilkan bon pembelian dan nota pembelian.

4.1.3.3. Activity diagram

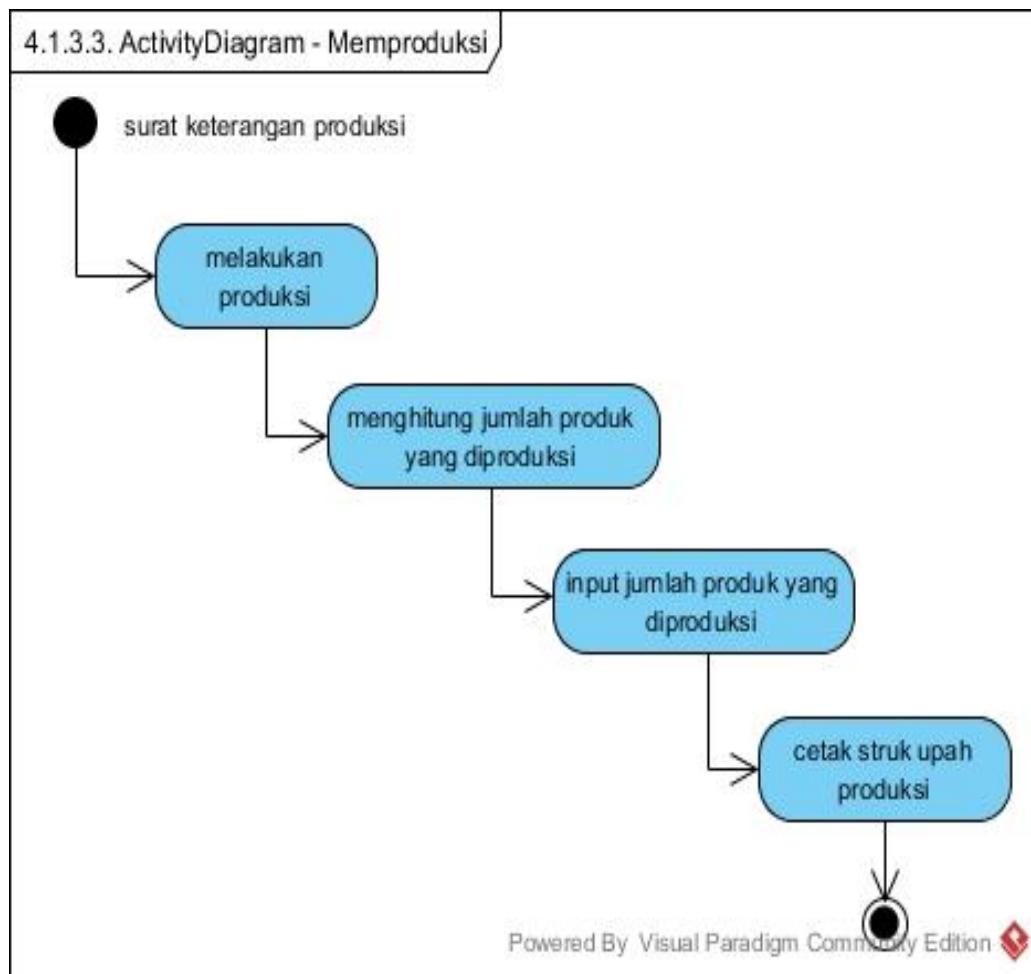
Berdasarkan *use case diagram* berikut ini adalah perancangan alur proses berbentuk *activity diagram*:

1. *Activity diagram* pesan yang diusulkan



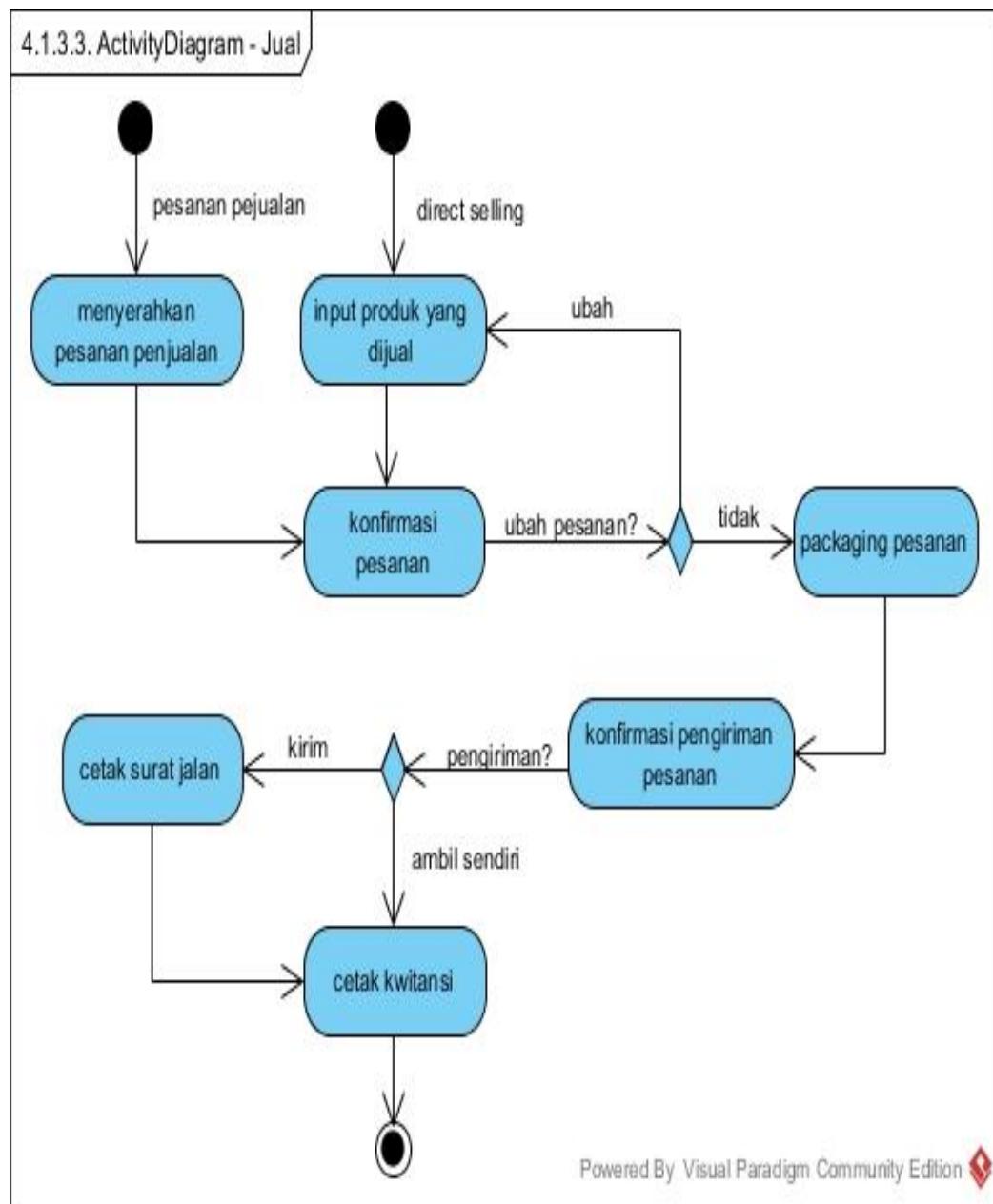
Gambar 4.2 *Activity Diagram* Pesan Yang Diusulkan

2. *Activity diagram* memproduksi yang diusulkan



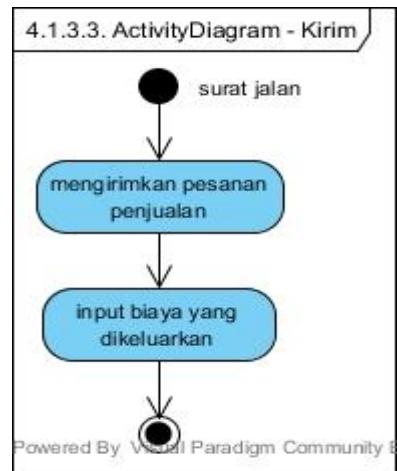
Gambar 4.3 Activity Diagram Memproduksi Yang Diusulkan

3. *Activity diagram jual yang diusulkan*



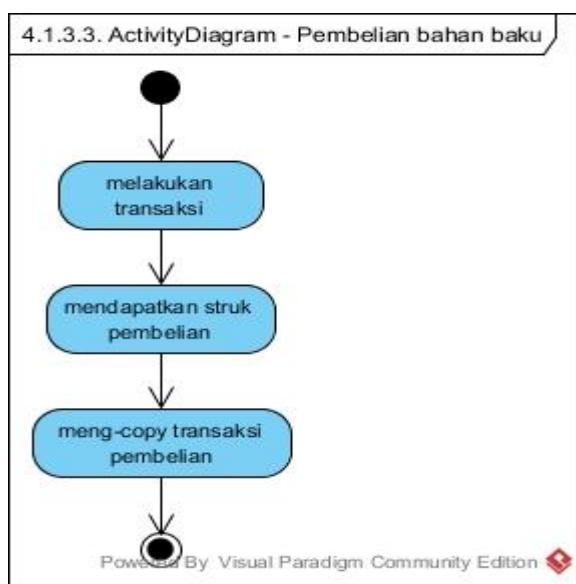
Gambar 4.4 Activity Diagram Jual Yang Diusulkan

4. *Activity diagram* kirim yang diusulkan



Gambar 4.5 Activity Diagram Kirim Yang Diusulkan

5. *Activity diagram* pembelian bahan baku yang diusulkan



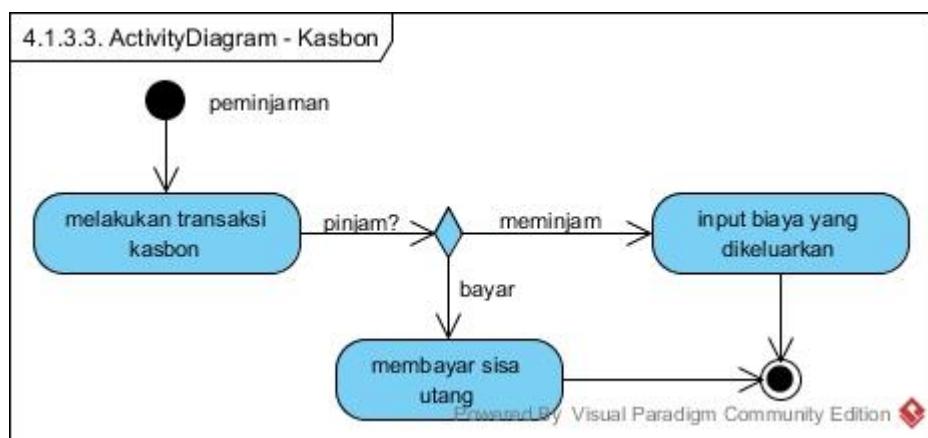
Gambar 4.6 Activity Diagram Pembelian Bahan Baku Yang Diusulkan

6. *Activity diagram* pembayaran lain-lain yang diusulkan



Gambar 4.7 Activity Diagram Pembayaran Lain-Lain Yang Diusulkan

7. *Activity diagram* kasbon yang diusulkan

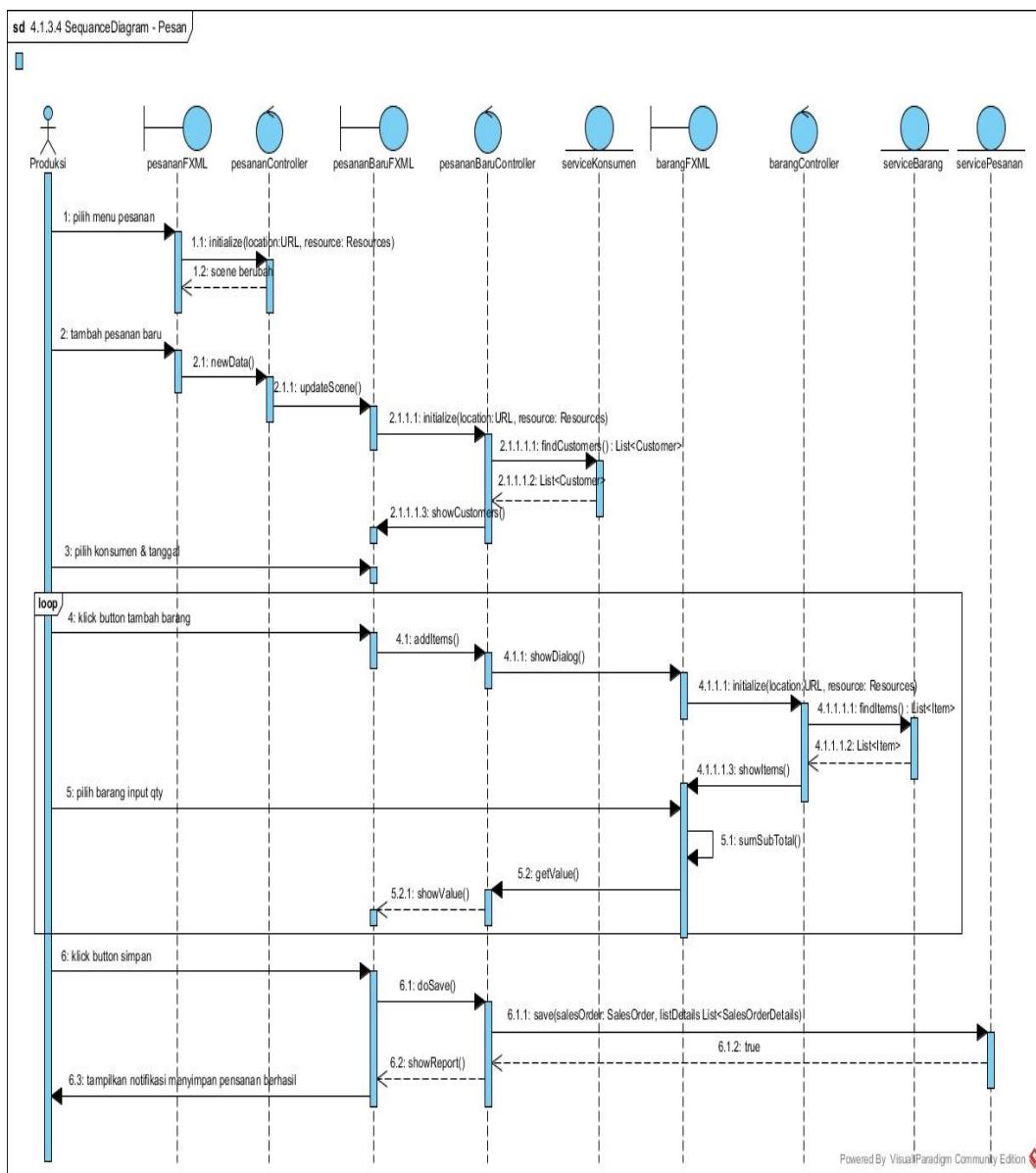


Gambar 4.8 Activity Diagram Kasbon Yang Diusulkan

4.1.3.4. Sequence diagram

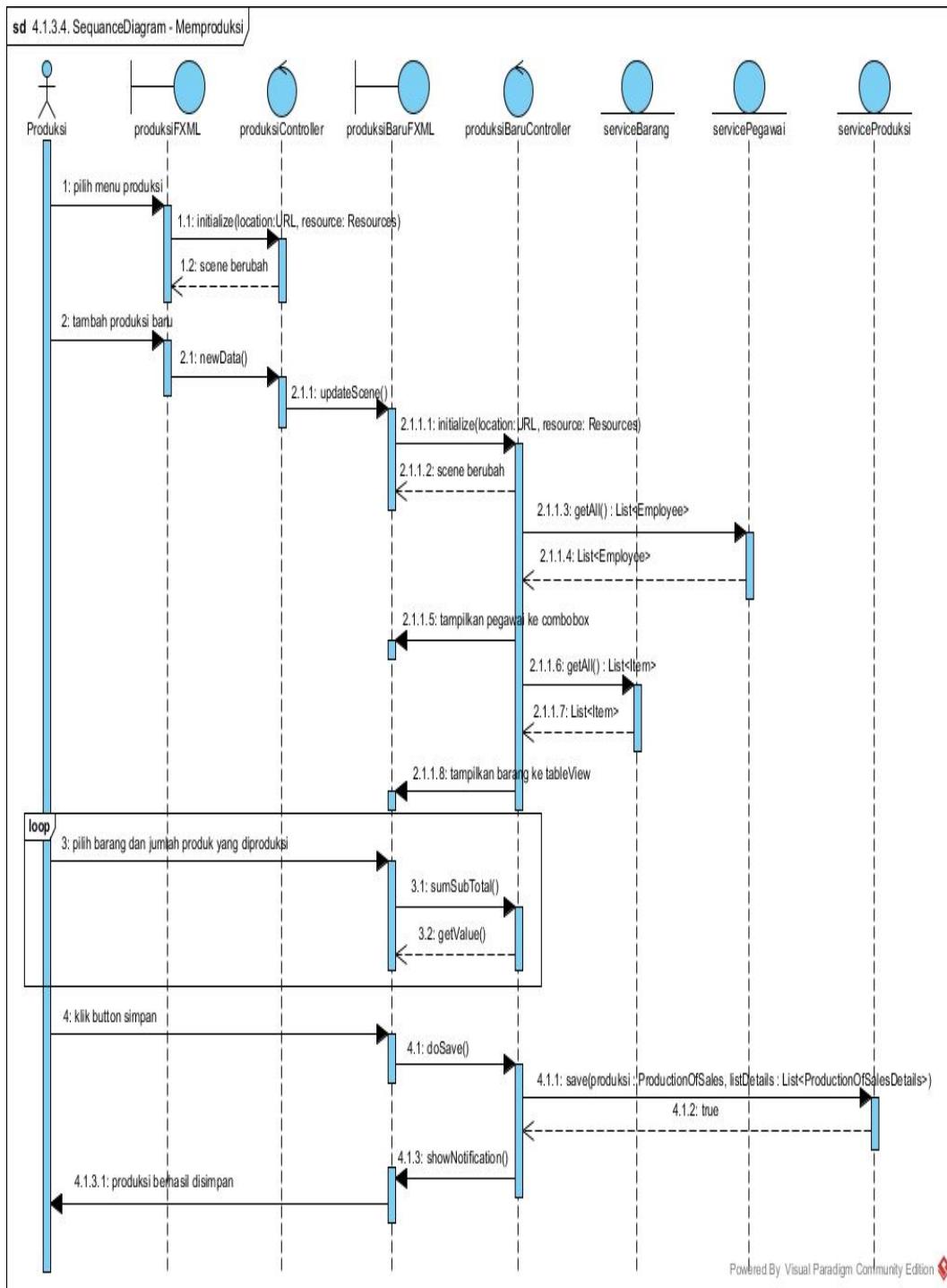
Berikut ini urutan proses untuk sistem yang diusulkan dalam bentuk *sequence diagram* berdasarkan *use case diagram*:

1. *Sequence diagram* pesan untuk sistem yang disulkan.



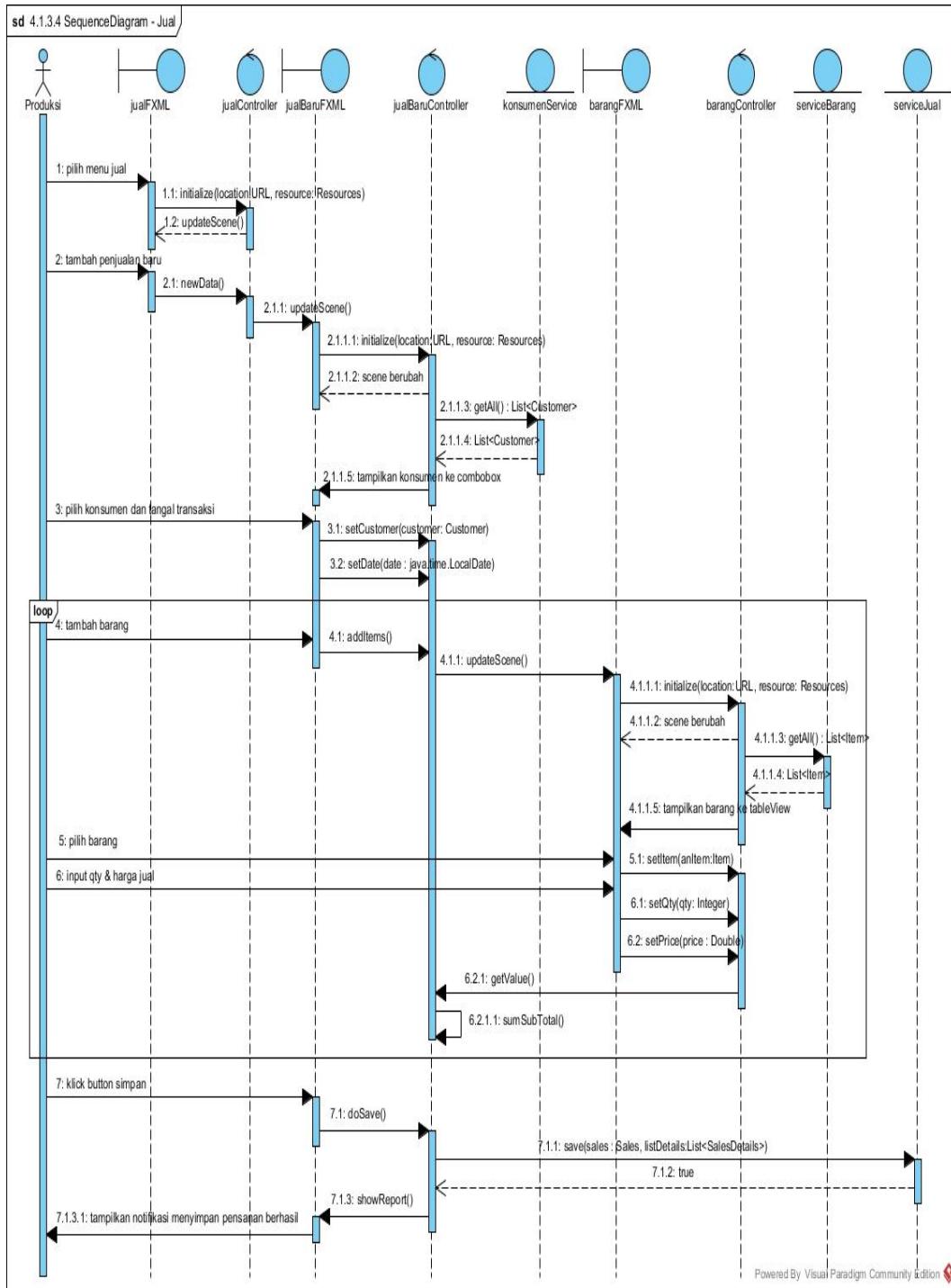
Gambar 4.9 Sequence Diagram Pesan Yang Diusulkan

2. Sequence diagram memproduksi untuk sistem yang diusulkan.



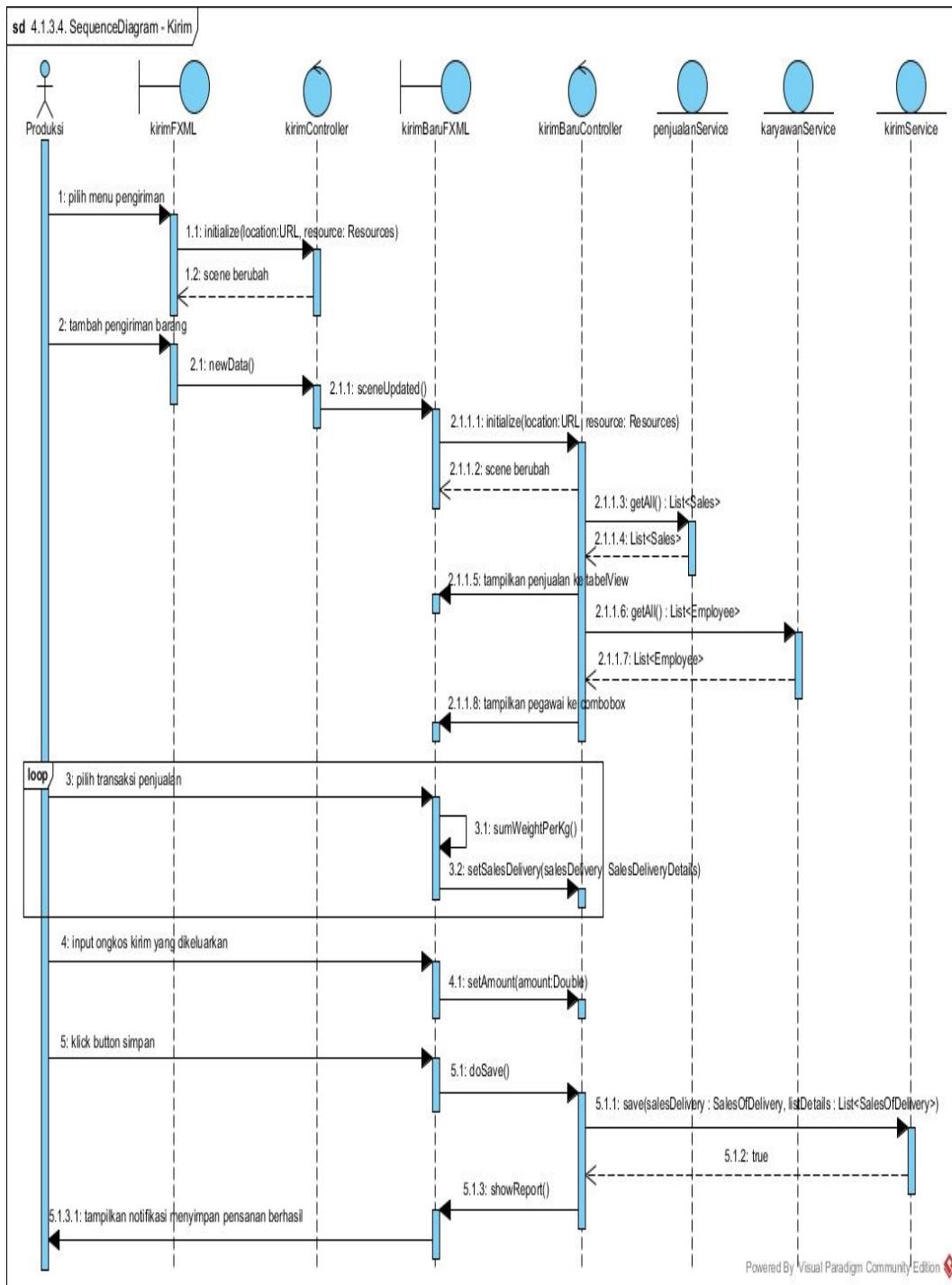
Gambar 4.10 Sequence Diagram Memproduksi Yang Diusulkan

3. Sequence diagram jual untuk sistem yang diusulkan.



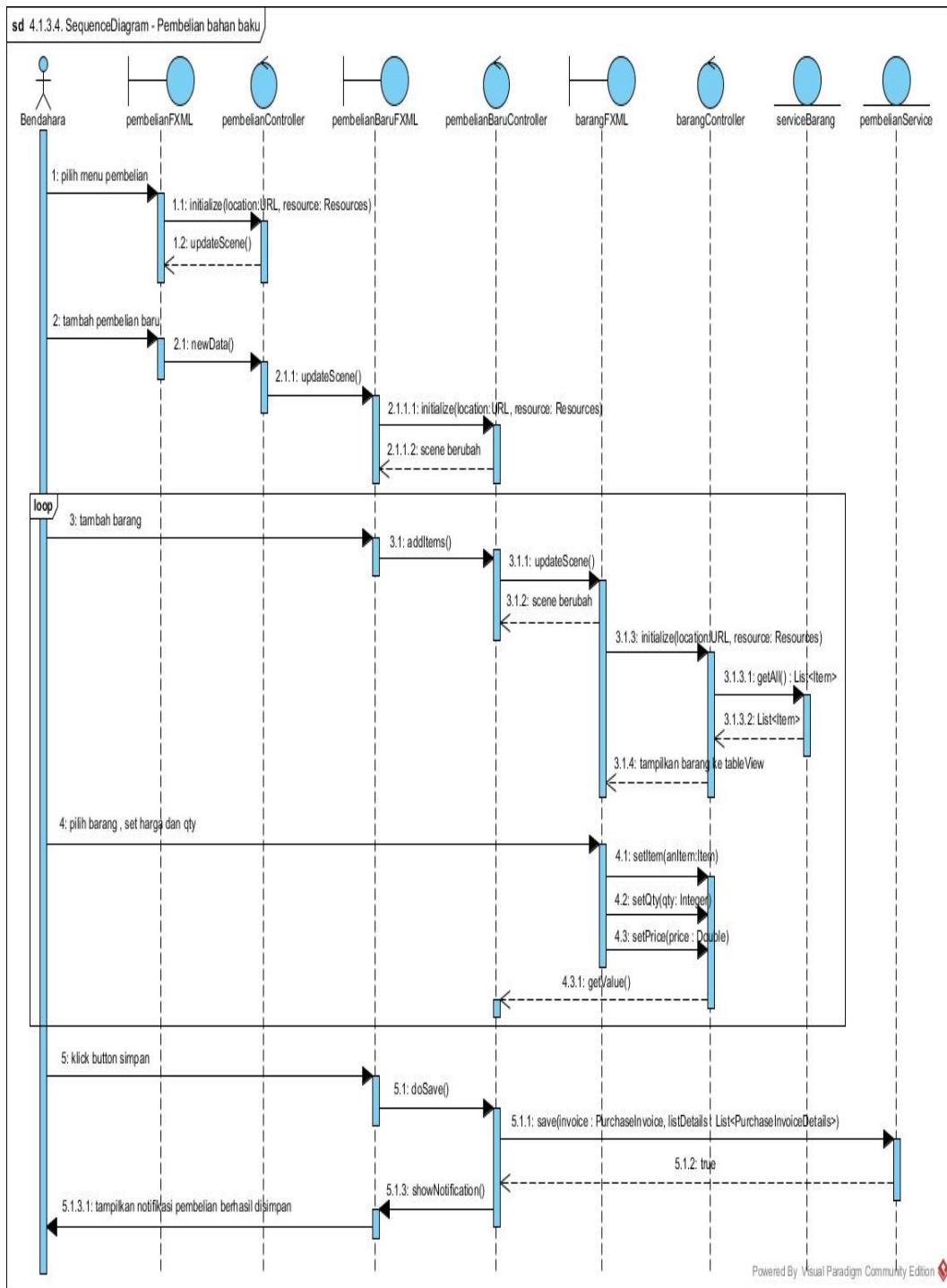
Gambar 4.11 Sequence Diagram Jual Yang Diusulkan

4. Sequence diagram kirim untuk sistem yang diusulkan.



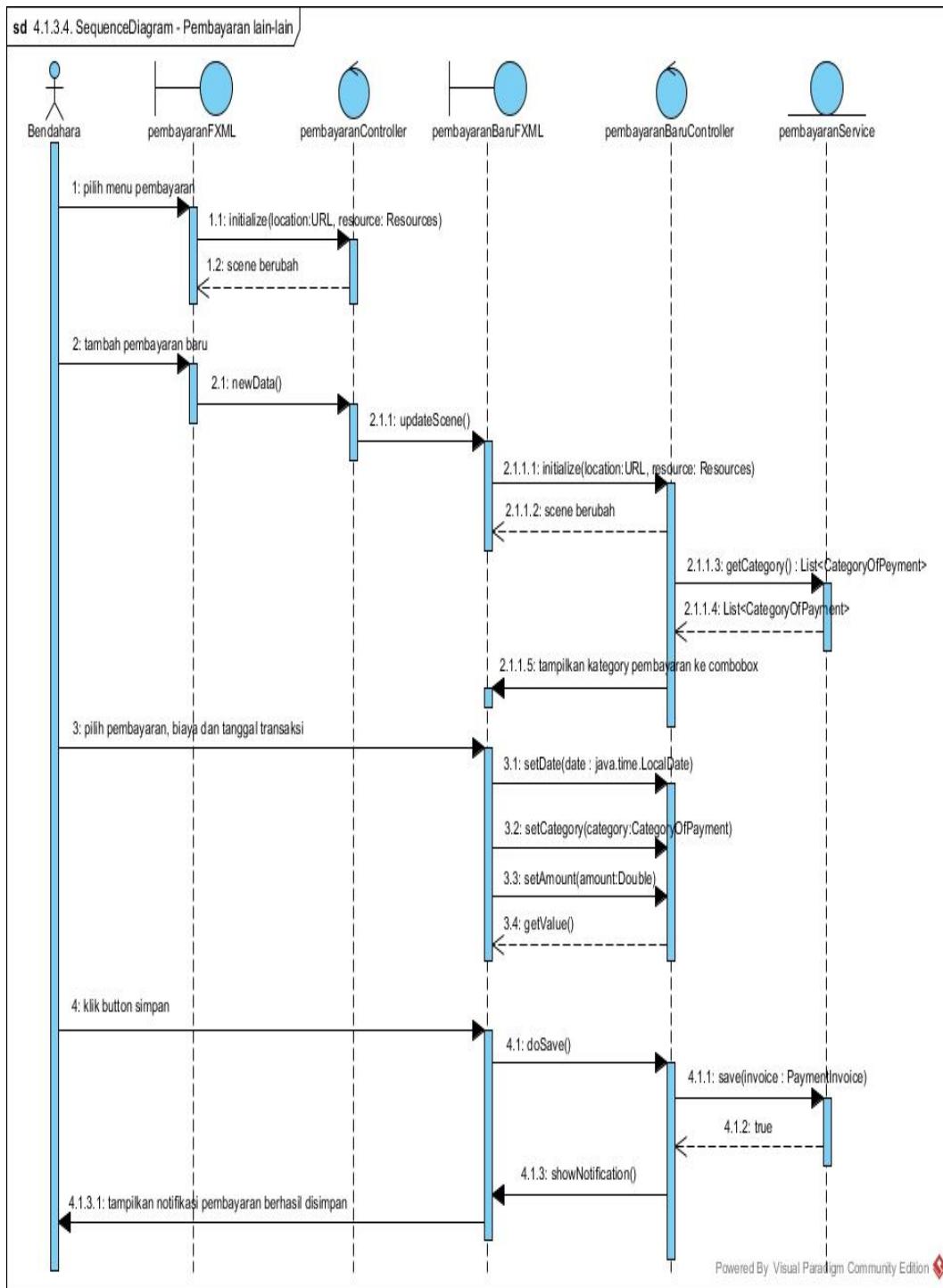
Gambar 4.12 Sequence Diagram Kirim Yang Diusulkan

5. *Sequence diagram* pembelian bahan baku untuk sistem yang diusulkan.



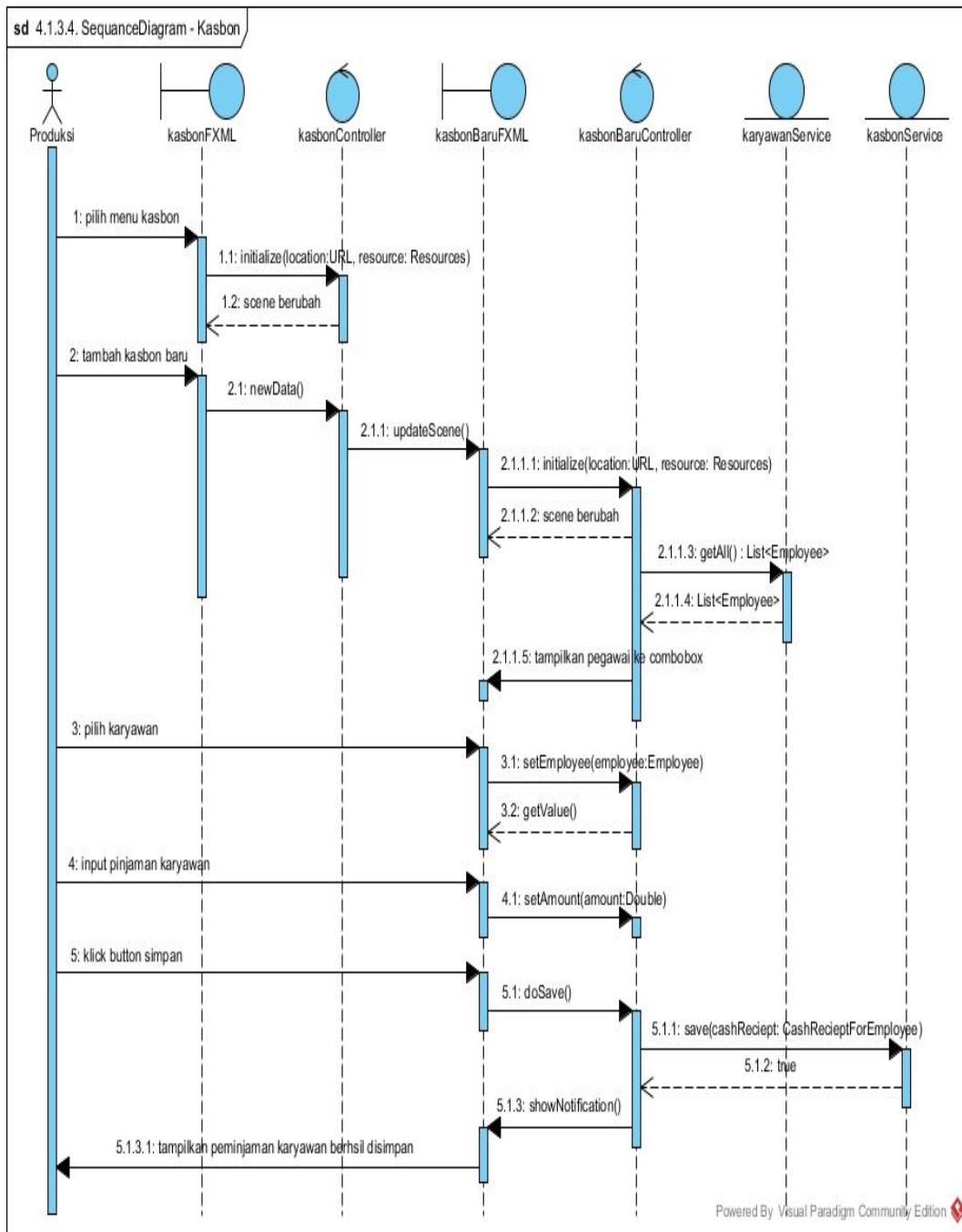
Gambar 4.13 *Sequence Diagram* Pembelian Bahan Baku Yang Diusulkan

6. *Sequence diagram* pembayaran lain-lain untuk sistem yang diusulkan.



Gambar 4.14 Sequence Diagram Pembayaran Lain-Lain Yang Diusulkan

7. Sequence diagram kasbon untuk sistem yang diusulkan.



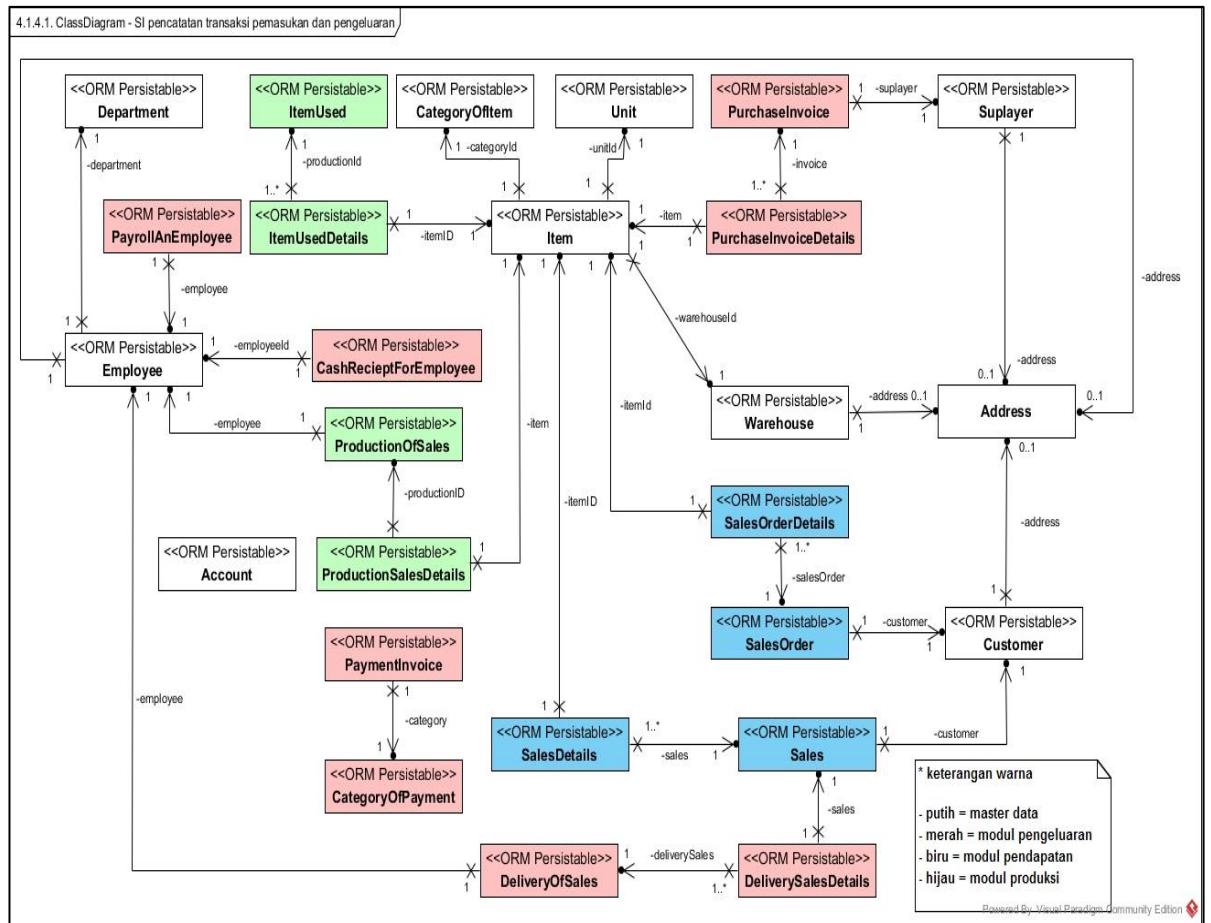
Gambar 4.15 Sequence Diagram Kasbon Yang Diusulkan

4.1.4. Perancangan data

Perancangan data digunakan untuk mengetahui kebutuhan tabel, kolom di basisdata, karena sebuah sistem informasi tidak akan jauh dari penggunaan basis data. Perancangan data terdiri:

4.1.4.1. Class diagram

Karena sistem yang dibangun menggunakan konsep ORM (*object relational mapping*) berikut ini adalah perancangan *entity* yang digambarkan menggunakan *class diagram*:



Gambar 4.16 Class Diagram Yang Diusulkan

Karena *entity* yang terdapat pada gambar *class diagram* terlalu banyak maka penulis tidak menyertakan atributnya karena menyulitkan untuk dibaca, berikut ini adalah atribut, properties, stereotype berserta penjelasannya di setiap *entity* tersebut:

1. *Modul masterdata*, berikut ini adalah penjelasan untuk *modul masterdata*:
 - a. *Entity department*.

Tabel 4.9 Entity Department

Nama	Tipe data	Properties
id	<i>java.lang.String</i>	{ <i>id</i> }
name	<i>java.lang.String</i>	
description	<i>java.lang.String</i>	
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

- b. *Entity warehouse*.

Tabel 4.10 Entity Warehouse

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
name	<i>java.lang.String</i>	
phone	<i>java.lang.String</i>	
address	<i>masterdata.Address</i>	{ <i>embedded</i> }
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

c. *Entity customer*

Tabel 4.11 Entity Customer

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
name	<i>java.lang.String</i>	
phone	<i>java.lang.String</i>	
address	<i>masterdata.Address</i>	{ <i>embedded</i> }
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

d. *Entity employee.*

Tabel 4.12 Entity Employee

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
name	<i>java.lang.String</i>	
address	<i>masterdata.Address</i>	{ <i>embedded</i> }
jobDesc	<i>masterdata.Department</i>	{ <i>OneToOne</i> }
work	<i>java.lang.Boolean</i>	
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

e. *Entity categoryOfItem*

Tabel 4.13 Entity Category Of Item

Nama	Tipe data	Properties
id	<i>java.lang.String</i>	{id}
name	<i>java.lang.String</i>	
description	<i>java.lang.String</i>	
createdBy	<i>java.lang.String</i>	{mappedSuperclass}
createdDate	<i>java.sql.Timestamp</i>	{mappedSuperclass}
lastUpdated	<i>java.sql.Timestamp</i>	{mappedSuperclass}

f. *Entity unit*

Tabel 4.14 Entity Unit Of Item

Nama	Tipe data	Properties
id	<i>java.lang.String</i>	{id}
name	<i>java.lang.String</i>	
description	<i>java.lang.String</i>	
createdBy	<i>java.lang.String</i>	{mappedSuperclass}
createdDate	<i>java.sql.Timestamp</i>	{mappedSuperclass}
lastUpdated	<i>java.sql.Timestamp</i>	{mappedSuperclass}

g. *Entity category of payment*

Tabel 4.15 Entity Category Of Payment

Nama	Tipe data	Properties
id	<i>java.lang.String</i>	{id}
paymentFor	<i>java.lang.String</i>	
description	<i>java.lang.String</i>	
createdBy	<i>java.lang.String</i>	{mappedSuperclass}
createdDate	<i>java.sql.Timestamp</i>	{mappedSuperclass}
lastUpdated	<i>java.sql.Timestamp</i>	{mappedSuperclass}

h. *Entity suplayer*

Tabel 4.16 Entity Suplayer

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{id, generatedValue}
name	<i>java.lang.String</i>	
phone	<i>java.lang.String</i>	
address	<i>masterdata.Address</i>	{embedded}
createdBy	<i>java.lang.String</i>	{mappedSuperclass}
createdDate	<i>java.sql.Timestamp</i>	{mappedSuperclass}
lastUpdated	<i>java.sql.Timestamp</i>	{mappedSuperclass}

i. *Entity account*

Tabel 4.17 Entity Account

Nama	Tipe data	Properties
username	<i>java.lang.String</i>	{id}
fullname	<i>java.lang.String</i>	
password	<i>java.lang.String</i>	
level	<i>java.lang.String</i>	
active	<i>java.lang.String</i>	
lastLogin	<i>java.sql.Timestamp</i>	
createdBy	<i>java.lang.String</i>	{mappedSuperclass}
createdDate	<i>java.sql.Timestamp</i>	{mappedSuperclass}
lastUpdated	<i>java.sql.Timestamp</i>	{mappedSuperclass}

j. *Class address*

Tabel 4.18 Class Address

Nama	Tipe data	Properties
streetAddress	<i>java.lang.String</i>	
city	<i>java.lang.String</i>	
pinCode	<i>java.lang.Integer</i>	
rt	<i>java.lang.Integer</i>	
rw	<i>java.lang.Integer</i>	
district	<i>java.lang.String</i>	

k. *Entity item*

Tabel 4.19 Entity Item

Nama	Tipe data	Properties
itemGenerator	<i>java.lang.Integer</i>	{id, generatedValue}
id	<i>java.lang.String</i>	
name	<i>java.lang.String</i>	
category	<i>masterdata.CategoryOfItem</i>	{OneToOne}
unit	<i>masterdata.Unit</i>	{OneToOne}
qty	<i>java.lang.Integer</i>	
priceBuy	<i>java.lang.Double</i>	
priceSell	<i>java.lang.Double</i>	
sell	<i>java.lang.Boolean</i>	
warehouse	<i>masterdata.Warehouse</i>	{OneToOne}
createdBy	<i>java.lang.String</i>	{mappedSuperclass}
createdDate	<i>java.sql.Timestamp</i>	{mappedSuperclass}
lastUpdated	<i>java.sql.Timestamp</i>	{mappedSuperclass}

2. *Modul production, package production* terdiri dari penggunaan bahan dan produksi untuk produk es krim, berikut ini adalah penjelasan masing-masing *class diagram* untuk *modul production*:

a. *Entity productionOfSales*

Tabel 4.20 Entity Production Of Sales

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{id, generatedValue}
date	<i>java.sql.date</i>	
employee	<i>masterdata.Employee</i>	{OneToOne}
used	<i>java.lang.Boolean</i>	
createdBy	<i>java.lang.String</i>	{mappedSuperclass}
createdDate	<i>java.sql.Timestamp</i>	{mappedSuperclass}
lastUpdated	<i>java.sql.Timestamp</i>	{mappedSuperclass}

b. *Entity productionOfSalesDetails*

Tabel 4.21 Entity Production Of Sales Details

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{id, generatedValue}
item	<i>masterdata.Item</i>	{OneToOne}
qty	<i>java.lang.Integer</i>	
production	<i>production.ProductionOfSales</i>	{ManyToOne}

c. *Entity itemUsed*

Tabel 4.22 Entity Item Used

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{id, generatedValue}
date	<i>java.sql.date</i>	
createdBy	<i>java.lang.String</i>	{mappedSuperclass}
createdDate	<i>java.sql.Timestamp</i>	{mappedSuperclass}
lastUpdated	<i>java.sql.Timestamp</i>	{mappedSuperclass}

d. *Entity itemUsedDetails*

Tabel 4.23 Entity Item Used Details

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{id, generatedValue}
item	<i>masterdata.Item</i>	{OneToOne}
qty	<i>java.lang.Integer</i>	
production	<i>production.ItemUsed</i>	{ManyToOne}

3. *Modul income, package income* terdiri dari pesanan penjualan dan penjualan, berikut ini adalah penjelasan masing-masing *class diagram* untuk *modul income*:

a. *Entity salesOrder*

Tabel 4.24 Entity Sales Order

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
transId	<i>java.lang.String</i>	
date	<i>java.sql.Date</i>	
checklist	<i>java.lang.Boolean</i>	
customer	<i>masterdata.Customer</i>	{ <i>OneToOne</i> }
dp	<i>java.lang.Double</i>	
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

b. *Entity salesOrderDetails*

Tabel 4.25 Entity Sales Order Details

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
item	<i>masterdata.Item</i>	{ <i>OneToOne</i> }
qty	<i>java.lang.Integer</i>	
salesOrder	<i>income.SalesOrder</i>	{ <i>ManyToOne</i> }

c. *Entity sales*

Tabel 4.26 Entity Sales

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
transId	<i>java.lang.String</i>	
customer	<i>masterdata.Customer</i>	{ <i>OneToOne</i> }
date	<i>java.sql.Date</i>	
month	<i>java.lang.String</i>	
year	<i>java.lang.Integer</i>	
shitTo	<i>java.lang.String</i>	
amount	<i>java.lang.Double</i>	
grantTotal	<i>java.lang.Double</i>	
paid	<i>java.lang.Boolean</i>	
sent	<i>java.lang.Boolean</i>	
recieved	<i>java.lang.Boolean</i>	
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

d. *Entity salesDetails*

Tabel 4.27 Entity Sales Details

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
item	<i>masterdata.Item</i>	{ <i>OneToOne</i> }
qty	<i>java.lang.Integer</i>	
price	<i>java.lang.Double</i>	
sales	<i>income.Sales</i>	{ <i>ManyToOne</i> }

4. *Modul expenditure, package expenditure* terdiri dari pengiriman penjualan, pembelian bahan baku, pembayaran lain-lain, penggajian karyawan. Berikut ini adalah penjelasan masing-masing *class diagram* untuk *modul expenditure*:

- a. *Entity deliveryOfSales*

Tabel 4.28 Entity Delivery Of Sales

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
deliveryId	<i>java.lang.String</i>	
employee	<i>masterdata.Employee</i>	{ <i>OneToOne</i> }
date	<i>java.sql.Date</i>	
status	<i>java.lang.Boolean</i>	
grantTotal	<i>java.lang.Double</i>	
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

- b. *Entity deliveryOfSalesDetails*

Tabel 4.29 Entity Delivery Of Sales Details

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
sales	<i>income.Sales</i>	{ <i>OneToOne</i> }
wightPerKg	<i>Java.lang.Integer</i>	
delivery	<i>expenditure.DeliveryOfSales</i>	{ <i>ManyToOne</i> }

c. *Entity purchaseInvoice*

Tabel 4.30 Entity Purchase Invoice

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id</i> , <i>generatedValue</i> }
suplayer	<i>masterdata.Suplayer</i>	{ <i>OneToOne</i> }
transDate	<i>java.sql.Date</i>	
amount	<i>java.lang.Double</i>	
grantTotal	<i>java.lang.Double</i>	
description	<i>java.lang.String</i>	
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

d. *Entity purchaseInvoiceDetails*

Tabel 4.31 Entity Purchase Invoice Details

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id</i> , <i>generatedValue</i> }
time	<i>masterdata.Item</i>	{ <i>OneToOne</i> }
qty	<i>java.lang.Integer</i>	
price	<i>java.lang.Double</i>	
invoice	<i>expenditure.PurchaseInvoice</i>	{ <i>ManyToOne</i> }

e. *Entity paymentInvoice*

Tabel 4.32 Entity Payment Invoice

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
date	<i>java.sql.Date</i>	
category	<i>masterdata.CategoryOfPayment</i>	{ <i>OneToOne</i> }
amount	<i>java.lang.Double</i>	
description	<i>java.lang.String</i>	
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

f. *Entity payrollAnEmployee*

Tabel 4.33 Entity Payroll An Employee

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id, generatedValue</i> }
date	<i>java.sql.Date</i>	
employee	<i>masterdata.Employee</i>	{ <i>OneToOne</i> }
amount	<i>java.lang.Double</i>	
otherAmount	<i>java.lang.Double</i>	
details	<i>List<ProductionOfSales></i>	{ <i>OneToMany</i> }
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

g. Entity *cashReceiptForEmployee*

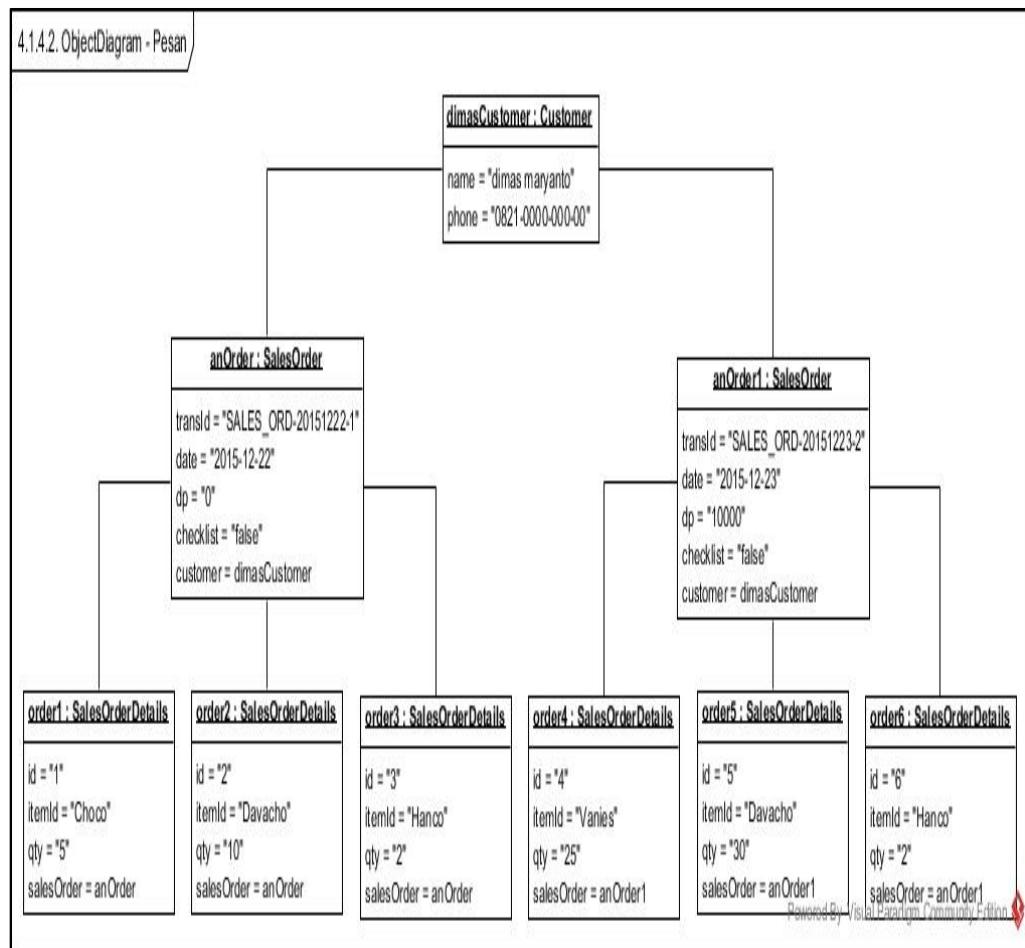
Tabel 4.34 Entity Cash Receipt For Employee

Nama	Tipe data	Properties
id	<i>java.lang.Integer</i>	{ <i>id</i> , <i>generatedValue</i> }
date	<i>java.sql.Date</i>	
employee	<i>masterdata.Employee</i>	{ <i>OneToOne</i> }
amount	<i>java.lang.Double</i>	
payment	<i>java.lang.Double</i>	
paid	<i>java.lang.Boolean</i>	{ <i>OneToMany</i> }
createdBy	<i>java.lang.String</i>	{ <i>mappedSuperclass</i> }
createdDate	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }
lastUpdated	<i>java.sql.Timestamp</i>	{ <i>mappedSuperclass</i> }

4.1.4.2. Object diagram

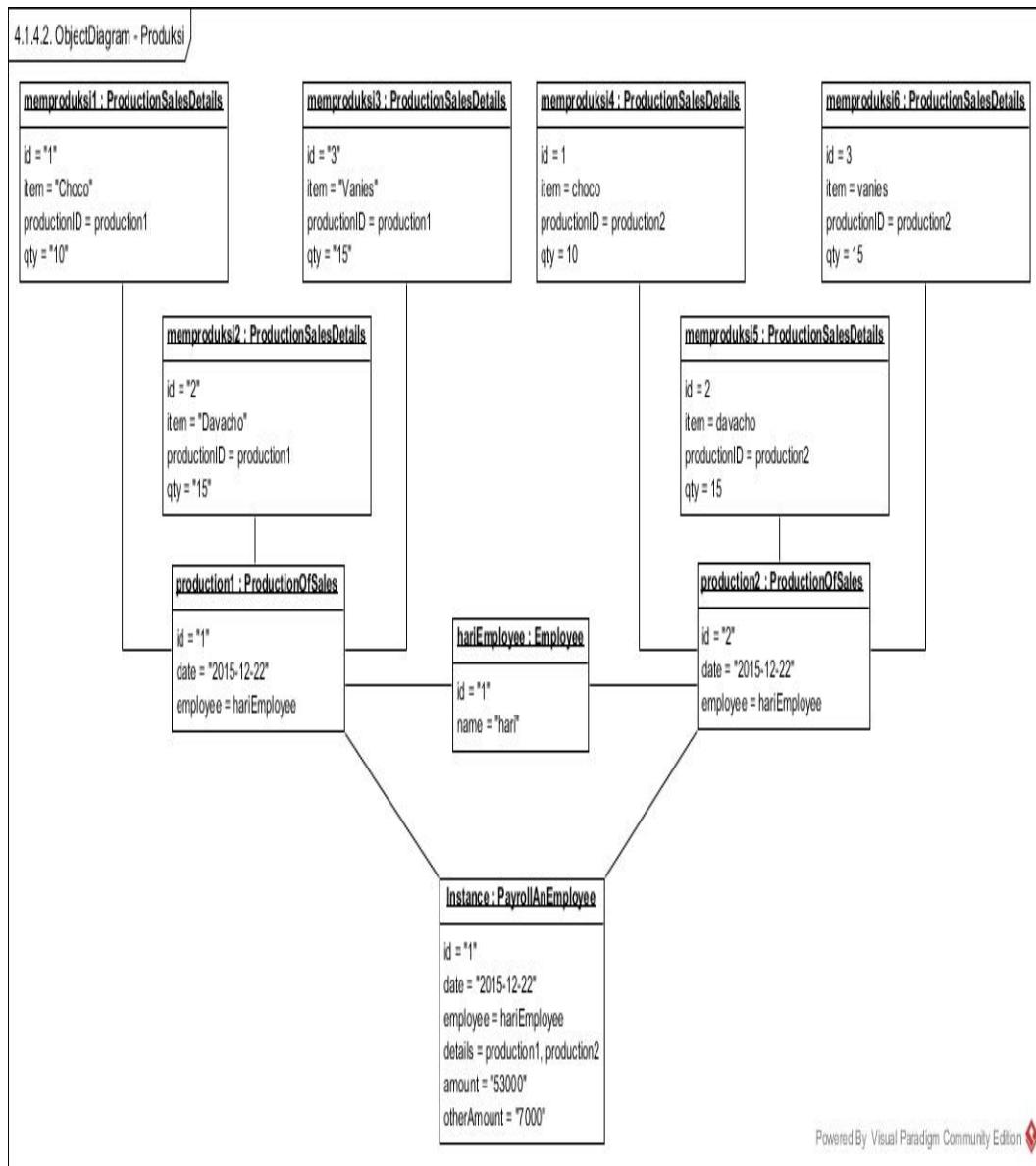
Berdasarkan class diagram berikut ini adalah penggambaran data menggunakan object diagram:

1. Object diagram pesan, berikut ini adalah object diagram yang menggambarkan objek-objek yang terdapat pada modul pesan.



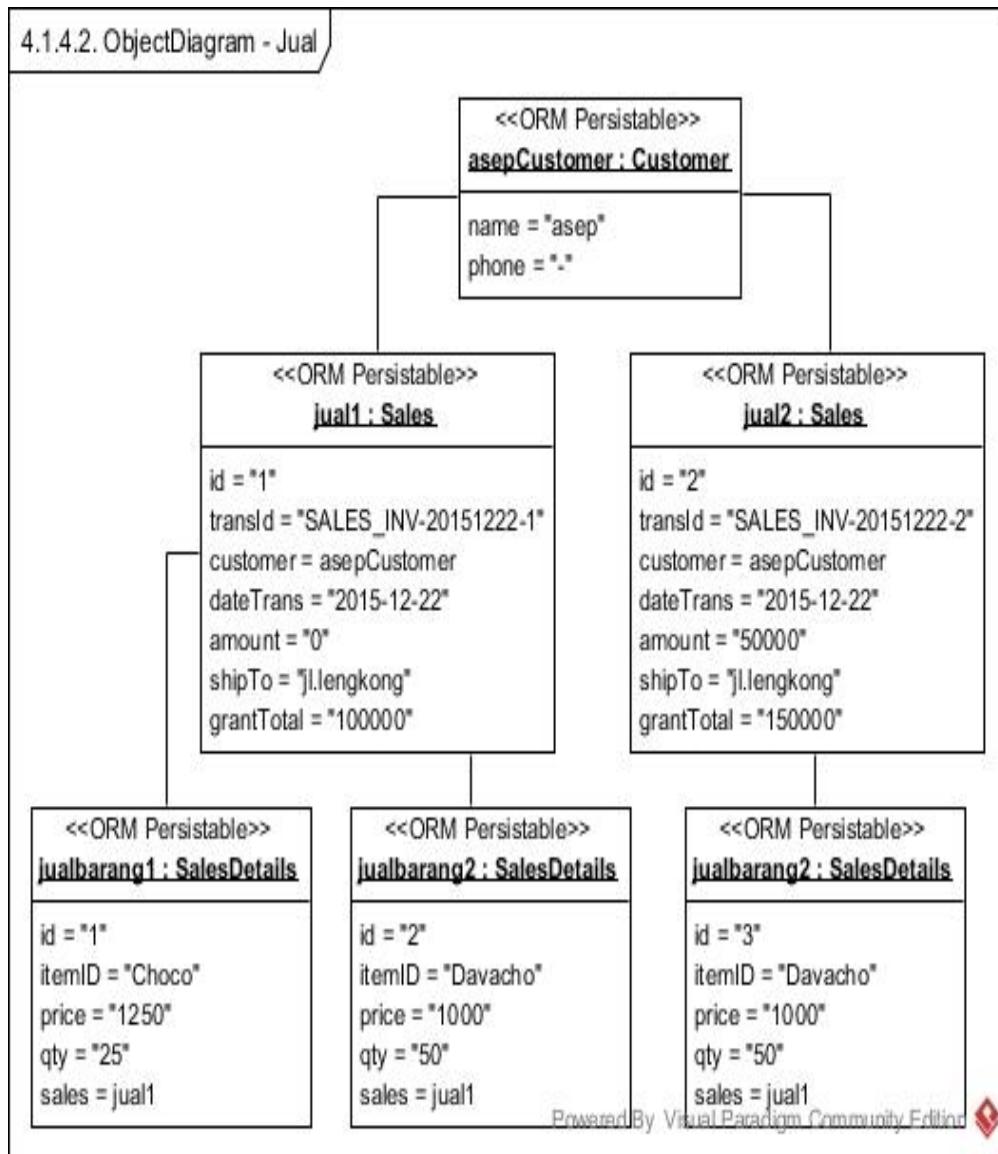
Gambar 4.17 Object Diagram Pesan

2. Object diagram produksi, berikut ini adalah object diagram yang menggabarkan object-object yang berinteraksi untuk menghasilkan proses pengajian karyawan bagian produksi.



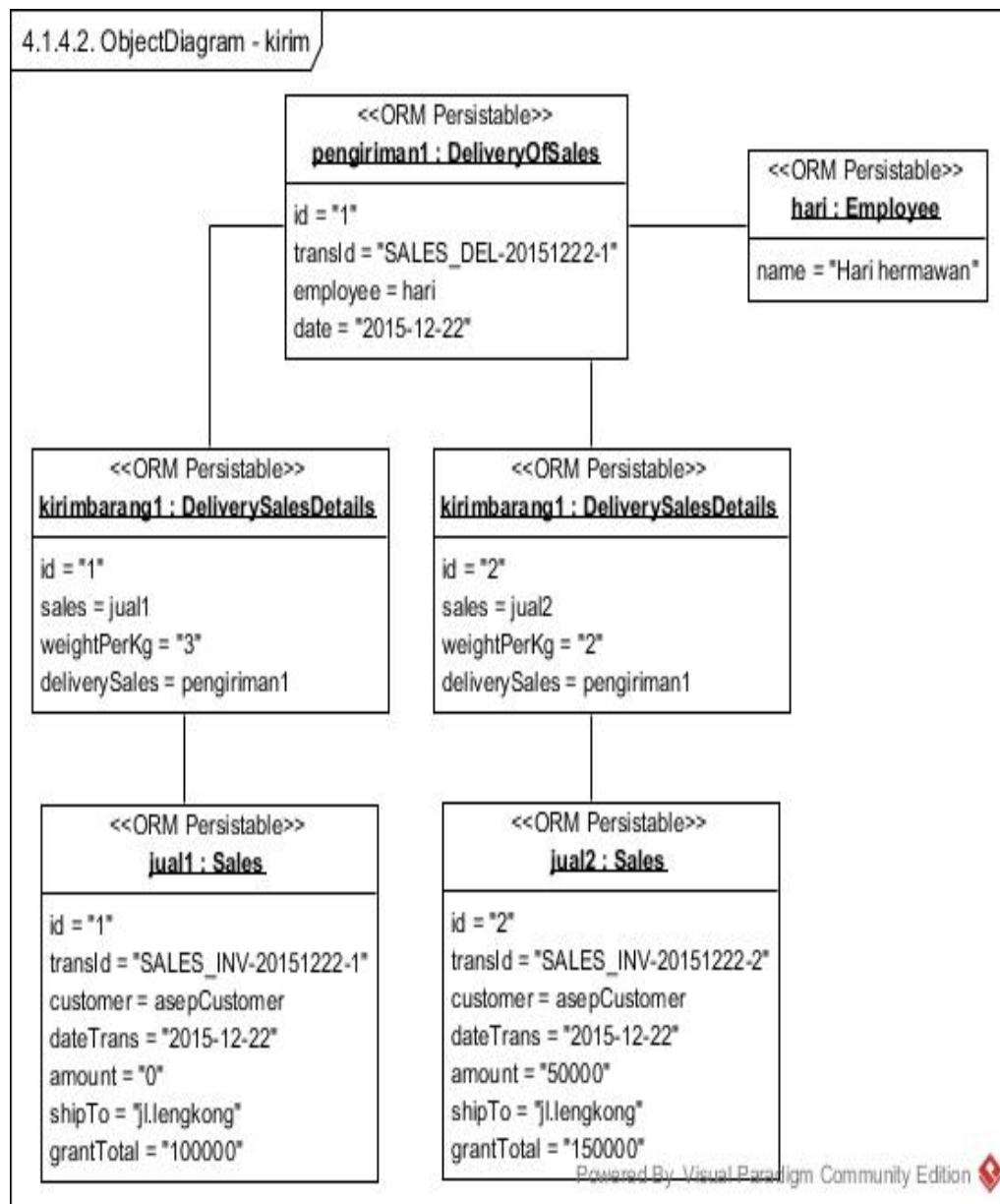
Gambar 4.18 Object Diagram Produksi

3. *Object diagram* penjualan, berikut ini adalah *object diagram* jual untuk menjelaskan objek-objek yang terdapat pada modul penjualan.



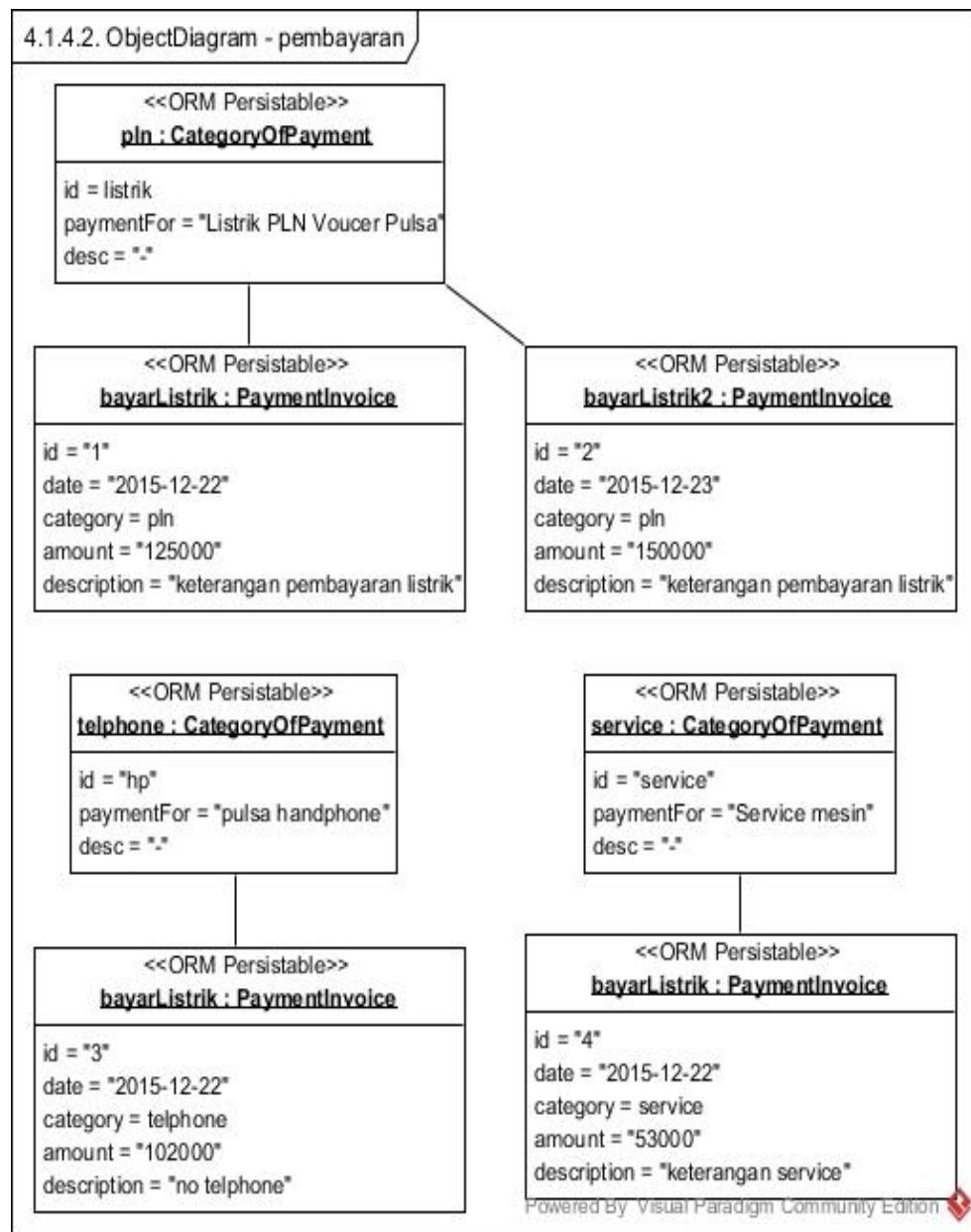
Gambar 4.19 Object Diagram Jual

4. *Object diagram* pengiriman, berikut ini adalah object diagram yang menjelaskan data untuk proses pengiriman penjualan.



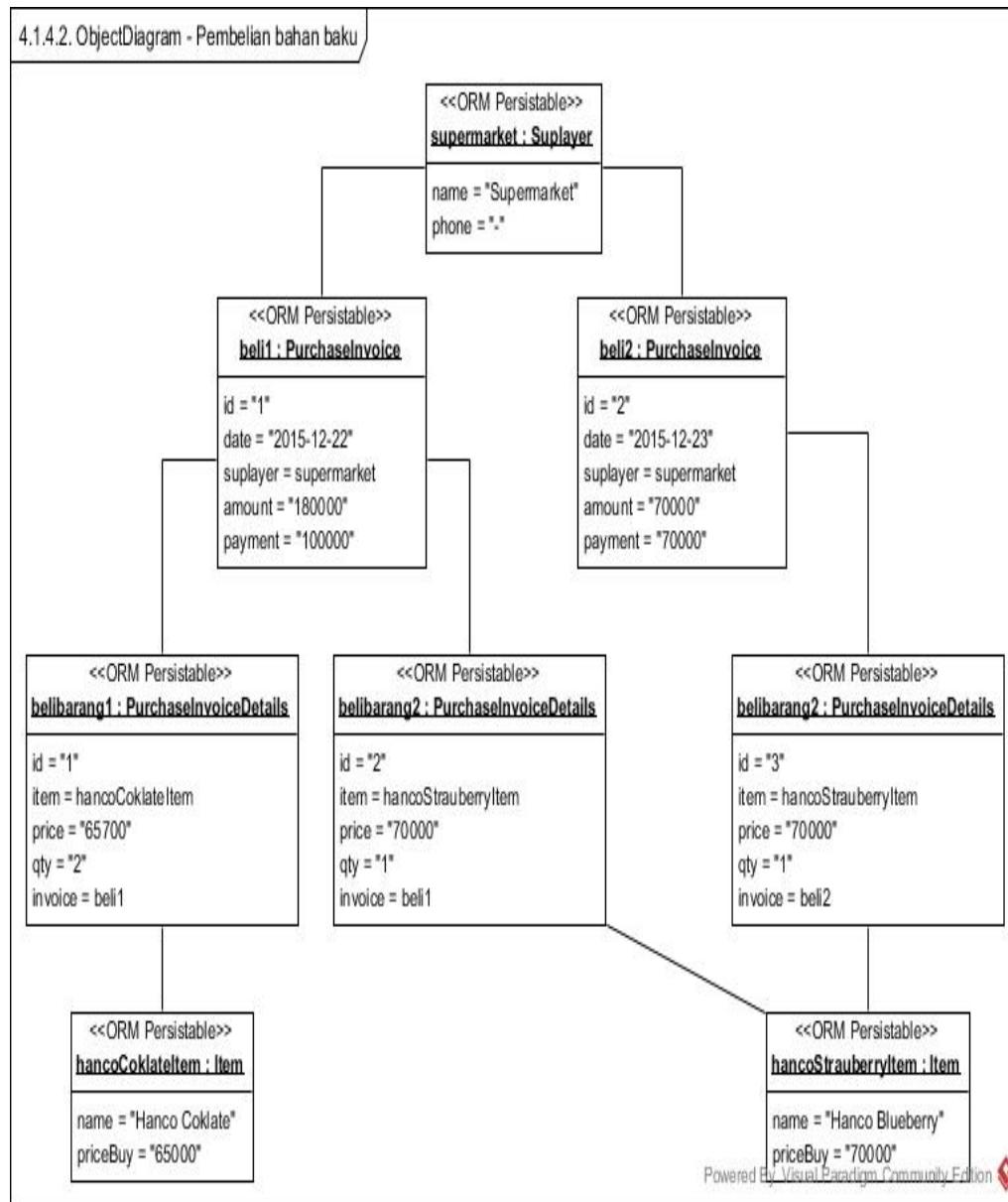
Gambar 4.20 Object Diagram Pengiriman

5. Object diagram pembayaran, berikut ini adalah object diagram yang menjelaskan data untuk proses pembayaran.



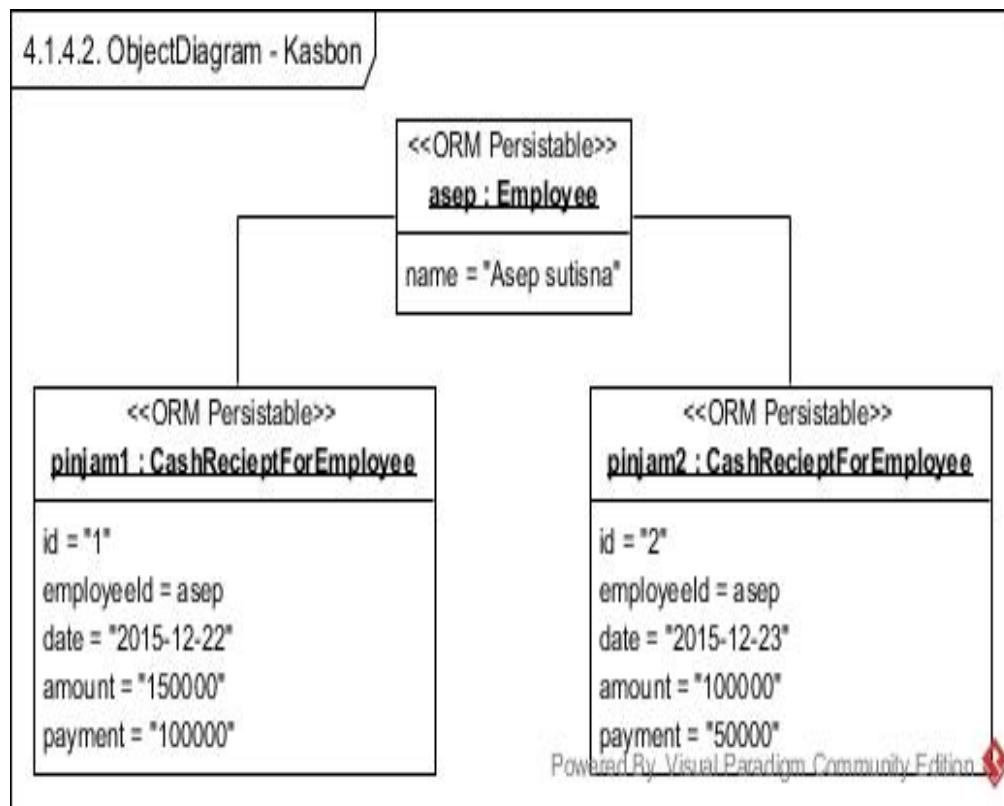
Gambar 4.21 Object Diagram Pembayaran

6. Object diagram pembelian, berikut ini adalah objek diagram yang menjelaskan data untuk proses pembelian bahan baku.



Gambar 4.22 *Object Diagram* Pembelian Bahan Baku

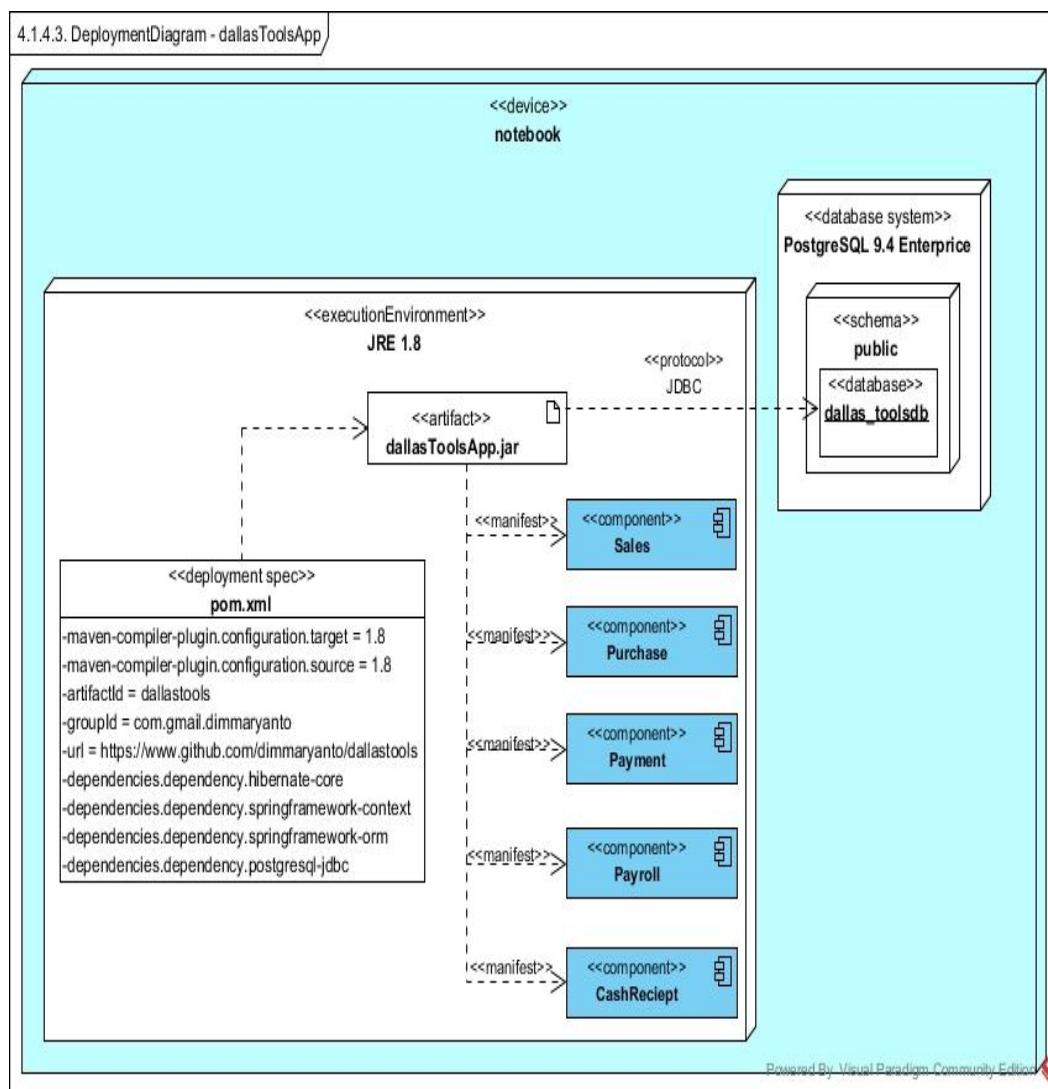
7. Object diagram kasbon, berikut ini adalah object diagram untuk menjelaskan data untuk proses kasbon karyawan



Gambar 4.23 *Object Diagram Kasbon*

4.1.4.3. Deployment diagram

Berikut ini adalah target yang diharapkan dalam perancangan sistem informasi Dallas Ice Cream adalah sebagai berikut:



Gambar 4.24 Deployment Diagram Yang Diusulkan

4.2. Perancangan Antar Muka

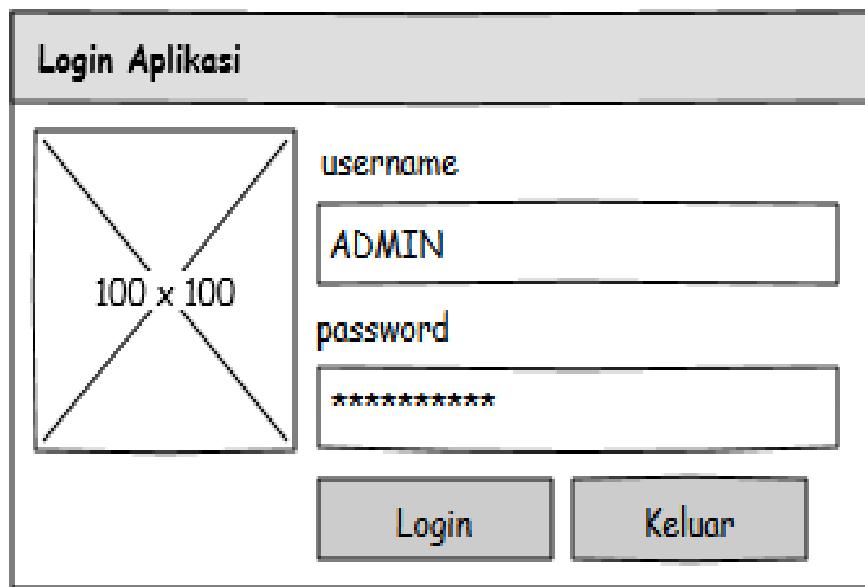
Perancangan antar muka untuk sistem informasi pencatatan transaksi pemasukan dan pengeluaran terdiri dari sebagai berikut:

4.2.1. Perancangan input

Perancangan input digunakan untuk menentukan tampilan program yang berfungsi sebagai alat masukan data. Perancangan input terdiri dari sebagai berikut:

4.2.1.1. Perancangan input data login

Login merupakan data yang digunakan untuk mengakses menu-menu sesuai dengan hak aksesnya. Perancangan input data login terdiri dari username dan password. Berikut ini adalah perancangannya:



The diagram illustrates the design of an application login interface. It features a title bar labeled "Login Aplikasi". Below the title bar is a logo consisting of a square divided by a diagonal cross, with the text "100 x 100" in the center. To the right of the logo are two input fields: one for "username" containing "ADMIN" and another for "password" containing a series of asterisks "*****". At the bottom are two buttons: "Login" and "Keluar".

Gambar 4.25 Perancangan Input Login Aplikasi

Aturan untuk mengakses menu-menu dalam aplikasi, berikut ini adalah kriterianya:

1. Input username, jika username tidak sesuai akan menampilkan “username dan password salah”
2. Input password, jika password tidak sesuai akan menampilkan “username dan password salah”

4.2.1.2. Perancangan input master data

Master data adalah data yang nantinya akan digunakan sebagai referensi atau rujukan. Berikut ini perancangan data master terdiri dari sebagia berikut:

a. Perancangan input data jabatan

Data jabatan merupakan data yang menentukan sebuah karyawan memiliki tugas. Data jabatan terdiri dari kode jabatan, nama jabatan, berserta tugas. Berikut ini adalah perancangan input data jabatan.

data jabatan	
Kode	B-PRO
Nama	Produksi
keterangan	Memproduksi Produk
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 4.26 Perancangan Input Data Jabatan

Perancangan input data jabatan tersebut tidak memiliki kriteria khusus, tetapi ada beberapa hal yang harus diperhatikan. Berikut ini adalah kriteria untuk menginput data jabatan:

1. Kode jabatan,
 - a. Tidak boleh kosong, dan tidak boleh ada data yang sama
 - b. Maksimal 10 karakter termasuk spasi
2. Nama
 - a. Tidak boleh kosong
 - b. Maksimal 255 karakter termasuk spasi

b. Perancangan input data karyawan

Data karyawan merupakan daftar karyawan yang berkerja pada usaha kecil menengah Dallas Ice Cream. Data karyawan terdiri dari nama karyawan, kode jabatan, kota, keluaran, alamat, rt, rw, kode pos. Berikut ini adalah perancangan input data karyawan.

Karyawan	
Nama	Hari Hermawan
pilih jabatan	B-PRO (Produksi)
Kota	Lembang
kelurahan	Lembang
Alamat	Jl.Cenpaka no.188
RT/RW	02 / 03
Kode Pos	40526
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 4.27 Perancangan Input Data Karyawan

Form data karyawan gambar diatas memiliki keterkaitan dengan data jabatan.

Berikut ini adalah kriteria untuk menginputkan data karyawan:

1. Nama karyawan, untuk penggunaan nama tidak ada ketentuan khusus namun berikut ini adalah aturan standarnya:
 - a. Nama tidak boleh kosong
 - b. Maksimal 255 karakter termasuk spasi
2. Pilih karyawan, untuk mendaftarkan karyawan pertama hal yang harus dilakukan adalah menginputkan ada jabatan. Pilih karyawan ini bertipe *combobox* artinya hanya dapat memilih salah satu jabatan.
3. Data lain-lain boleh dikosongkan. Kecuali rt, rw, dan kode pos karena menggunakan *spinner* maka nilai *defaultnya* 0.

c. Perancangan input data kategori barang

Data kategori barang merupakan deskripsi untuk menjelaskan bahwa barang tersebut memiliki jenis-jenis tertentu dengan tujuan membedakan antara barang yang memiliki jenis es krim atau hanya sebagai bahan untuk membuat es krim. Data kategori barang terdiri dari kode, nama, dan keterangan. Berikut ini adalah perancangan input data kategori barang:

Kategori Produk	
Kode	ICE
Nama	ICE CREAM
keterangan	Produk Utama
Simpan Batal	

Gambar 4.28 Perancangan Input Data Kategori Barang

Perancangan input data kategori barang serupa dengan perancangan input data jabatan. Berikut ini kriteria untuk menginput data kategori barang:

1. Kode
 - a. Tidak boleh kosong, dan tidak boleh sama
 - b. Maksimal 50 karakter termasuk spasi
2. Nama
 - a. Tidak boleh kosong
 - b. Maksimal 255 karakter termasuk spasi
3. Keterangan, untuk keterangan tidak memiliki kriteria khusus artinya boleh dikosongkan.

d. Perancangan input data satuan barang

Data satuan barang merupakan data yang merepresentasikan ukuran, berat, dan lain-lain contohnya adalah kg, cm, pcs. Data satuan barang terdiri dari kode, nama, keterangan. Berikut ini adalah perancangan input data kategori barang:

satuan barang	
Kode	pcs
Nama	PIECIES
keterangan	-
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 4.29 Perancangan Input Data Satuan Barang

Perancangan input data satuan barang aturannya sama persis dengan perancangan input data jabatan. Berikut ini kriteria untuk menginput data satuan barang:

1. Kode
 - a. Tidak boleh kosong, dan tidak boleh sama
 - b. Maksimal 50 karakter termasuk spasi
2. Nama
 - a. Tidak boleh kosong
 - b. Maksimal 255 karakter termasuk spasi
3. Keterangan, untuk keterangan tidak memiliki kriteria khusus artinya boleh dikosongkan.

e. Perancangan input data gudang

Data gudang merupakan data yang merepresentasikan lokasi barang disimpan. Data gudang terdiri dari nama, no kontak, kota, kecamatan, alamat, rt, rw, kode pos. Berikut ini adalah perancangan input data gudang:

Gudang	
Nama	Central
No Kontak	0821-XXXX-XXX
Kota	Lembang
Kecamatan	Lembang
Alamat	Jl. Cempaka no.188
RT/RW	02
	03
Kode Pos	40526
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 4.30 Perancangan Input Data Gudang

Perancangan input data gudang tidak memiliki kriteria khusus, berikut ini adalah kriteria untuk menginput data gudang:

1. Nama
 - a. Tidak boleh kosong
 - b. Maksimal 100 karakter termasuk spasi.
2. No kontak, maksimal 15 karakter
3. Data lain-lain, boleh dikosongkan kecuali rt, rw, kode pos karena menggunakan *spinner* maka nilai defautnya 0

f. Perancangan input data barang

Data barang merupakan daftar barang-barang apa saja yang tersedia pada usaha kecil menengah Dallas Ice Cream, yang nantinya akan digunakan untuk transaksi pembelian, penjualan, pemesanan penjualan dan produksi. Data barang terdiri dari nama, kategori, satuan, gudang, harga jual, harga beli, dan status apakah barang tersebut dijual atau hanya digunakan saja. Berikut ini adalah perancangan input data barang:

Barang-Barang	
Nama	Choco
Kategori	ICE (ICE CREAM)
Satuan	PCS (PIECIES)
Gudang	CENTRAL (0821-XXXX-XXX)
Harga Beli	Rp.0,-
Harga Jual	Rp.0,-
<input checked="" type="radio"/> Dijual <input type="radio"/> Tidak Dijual	
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 4.31 Perancangan Input Data Barang

Data barang memiliki keterkaitan dengan data kategori barang, satuan barang, dan gudang. Berikut ini kriteria untuk menginput data barang:

1. Nama
 - a. Tidak boleh kosong, dan tidak boleh ada data yang sama
 - b. Maksimal 100 karakter termasuk spasi

- c. Huruf awalan diawali dengan huruf kapital, karena nantinya akan ditampilkan pada saat dicetak.
- 2. Kategori, satuan, gudang adalah sebuah *combobox*. Artinya harus dipilih salah satu.
- 3. Harga beli dan jual adalah *spinner*. Artinya nilai defautnya adalah 0
- 4. Status
 - a. Dijual, artinya barang yang diinputkan adalah produk yang akan dijual dan diproduksi
 - b. Tidak dijual, artinya barang yang dininputkan adalah barang yang digunakan sehari-hari seperti gas elpiji, gula, cream dan lain-lain.

g. Perancangan input data konsumen

Data konsumen merupakan daftar konsumen bagi usaha kecil menengah Dallas Ice Cream, data konsumen terdiri dari pelanggan baru, depo, mitra dan lain-lain. Perancangan input data konsumen terdiri dari nama, no kontak, kota, kecamatan, alamat, rt, rw, kode pos. Berikut ini adalah perancangan input data konsumen:

Konsumen		
Nama	Dimas M	
No Kontak	0821-XXXX-XXX	
Kota	kab.Bandung	
Kecamatan	Cileunyi	
Alamat	Jl.bukit indah no b8	
RT/RW	06	18
Kode Pos	40526	
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>		

Gambar 4.32 Perancangan Input Data Konsumen

h. Perancangan input data pemasok

Data pemasok merupakan data yang merepresentasikan produsen bagi usaha kecil menengah Dallas Ice Cream. Perancangan input data pemasok terdiri

dari nama, no kontak, kota, kabupaten, alamat, rt, rw, kode pos. Berikut ini adalah perancangan input datap pemasok:

The screenshot shows a window titled "Pemasok". It contains the following fields:

Nama	Supermarket
No Kontak	0821-XXXX-XXX
Kota	Lembang kab.Bandung
Kecamatan	desa kayu ambon
Alamat	Jl.Cempaka no 10
RT/RW	-
Kode Pos	-

At the bottom right are two buttons: "Simpan" and "Batal".

Gambar 4.33 Perancangan Input Data Pemasok

Perancangan input data pemasok memiliki kriteria yang sama dengan konsumen.

i. Perancangan input data kategori pembayaran

Perancangan input data kategori pembayaran terdiri dari kode, nama dan keterangan. Berikut ini adalah perancangan input data kategori pembayaran:

Kategori Pembayaran	
Kode	PLN
Nama	Pembayaran Listrik (Pulsa)
Ket	-
Simpan Batal	

Gambar 4.34 Perancangan Input Data Kategori Pembayaran

Perancangan input data kategori pembayaran memiliki kriteria yang sama seperti perancangan input data kategori barang.

4.2.1.3. Perancangan input data transaksi

Perancangan input data transaksi adalah proses pencatatan transaksi pemasukan dan pengeluaran. Perancangan input data transaksi terdiri dari:

a. Perancangan input data produksi

Data produksi dimaksudkan untuk menginputkan berapa es krim yang dapat produksi oleh karyawan dalam kurun waktu tertentu dan pada akhirnya akan dilakukan proses perhitungan upah untuk karyawan tersebut. Perancangan input data produksi terdiri dari karyawan, tangga, dan daftar barang. Berikut ini adalah perancangannya:

Kode barang	Nama barang	unit	Jumlah
1-ice-choce	Choco	pcs	0
1-ice-davacho	Davacho	pcs	0
1-ice-vanies	Vanies	pcs	4
1-ice-fruitemilk	Fruitmilk	pcs	10

Gambar 4.35 Perancangan Input Data Produksi

Alur untuk menginput data produksi adalah sebagai berikut:

1. Memilih karyawan melalui *combobox*

2. Melakukan klik 2 kali pada cell pada kolom jumlah berdasarkan nama atau kode barang. Setelah itu akan muncul textfield kemudian masukan jumlah barang yang akan diproduksi.
3. Klik *button* simpan

b. Perancangan input data penggunaan bahan

Input data penggunaan bahan digunakan untuk mengurangi bahan atau barang yang digunakan ketika proses produksi. Perancangan input data penggunaan bahan terdiri dari pilih karyawan dan daftar barang. Perancangan input data penggunaan bahan adalah sebagai berikut:

Kode barang	Nama barang	unit	jumlah
1-bahan-gula	Gula pasir	pcs	0
1-bahan-susu	Susu ketal manis	pcs	0
1-kemasan-stik	Stick ice cream	pcs	500
1-bahan-cream	Cream	kg	5

Gambar 4.36 Perancangan Input Data Penggunaan Bahan

Alur untuk menginputkan data penggunaan bahan atau barang sama seperti produksi barang.

c. Perancangan input data pemesanan penjualan

Input data pemesanan penjualan digunakan untuk mencatat pesanan yang nantinya diproses di perancangan input data penjualan dan sebagai surat keterangan untuk melakukan produksi es krim. Perancangan input data pemesanan penjualan terdiri dari tanggal transaksi, konsumen, dan daftar produk. Berikut ini adalah perancangan input data pemesanan penjualan.

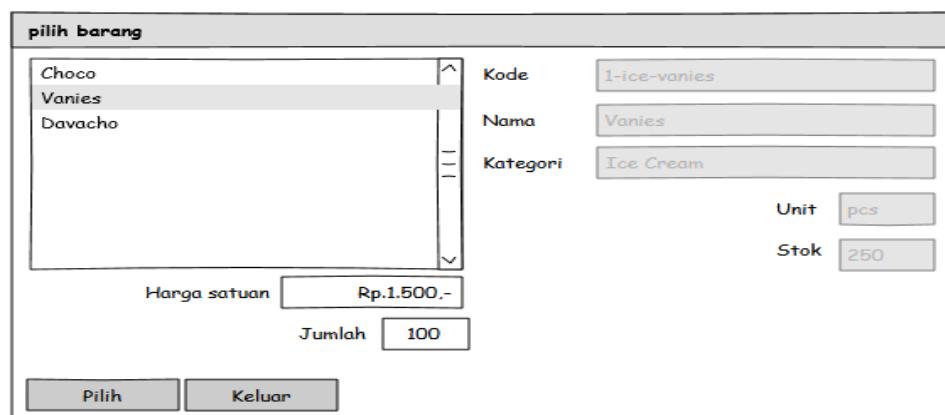
Aksi	Kode barang	Nama Barang	unit	qty	harga	sub total
<input checked="" type="checkbox"/>	1-ice-vanies	Vanes	pcs	100	Rp.1.250,-	Rp.125.000-
<input checked="" type="checkbox"/>	1-ice-chovo	Choco	pcs	100	Rp.1.000,-	Rp.100.000-
<input checked="" type="checkbox"/>	1-material-hanco	Hanco	pcs	1	Rp.60.000,-	Rp.60.000-

Gambar 4.37 Perancangan Input Data Pesanan Penjualan

Alur untuk menginput data pesanan penjualan adalah sebagai berikut:

1. Tanggal, secara otomatis tanggal akan menyesuaikan dengan tanggal yang ada pada sistem operasi.
2. Konsumen, karena ini pesanan penjualan maka harus memilih konsumen. Field konsumen ini berjenis *combobox*.

3. Klik tombol tambah produk kemudian akan menampilkan dialog seperti berikut:

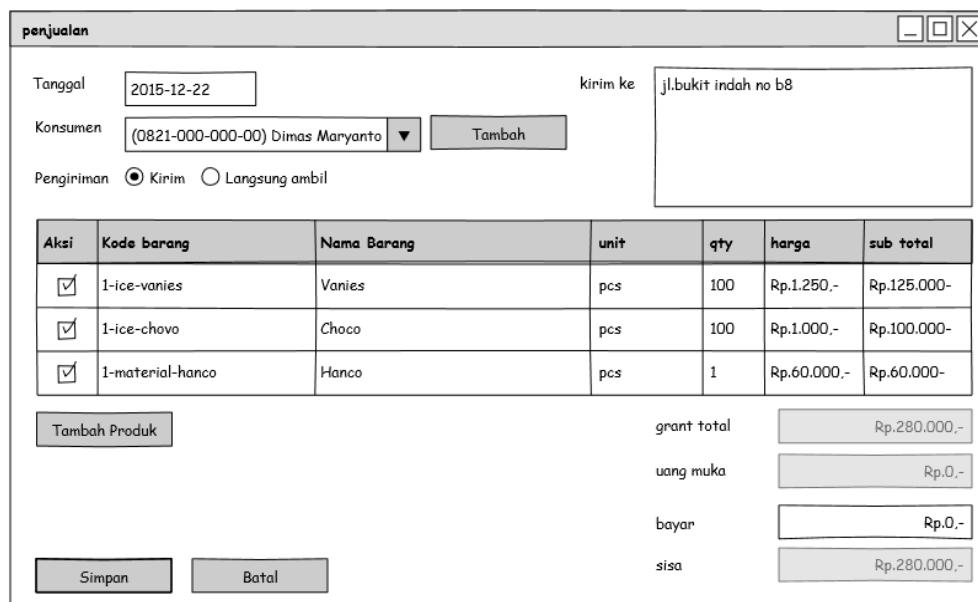


Gambar 4.38 Perancangan Input Data Pilih Produk Pesanan Penjualan

4. Seleksi nama barang pada tableView atau listView, setelah itu data detail produk akan muncul seperti harga, nama, kategori, unit dan lain-lain.
5. Input jumlah yang dipesan.
6. Setelah itu klik *button* pilih untuk menambahkan produk ke tabel atau klik keluar untuk membatalkan menambah produk ke tabel.
7. Kemudian input uang muka jika ada.
8. Lalu klik *button* simpan.

d. Perancangan input data penjualan

Input data penjualan digunakan untuk memcatat transaksi penjualan yang nantinya akan mencetak kwitansi penjualan dan akan diproses untuk membuat laporan keuangan, laporan pendapatan penjualan dan juga omzet penjualan per konsumen. Perancangan input data penjualan terdiri dari tanggal, konsumen, status pengiriman, alamat kirim, daftar produk, uang muka, bayar. Berikut ini adalah perancangan input data penjualan:



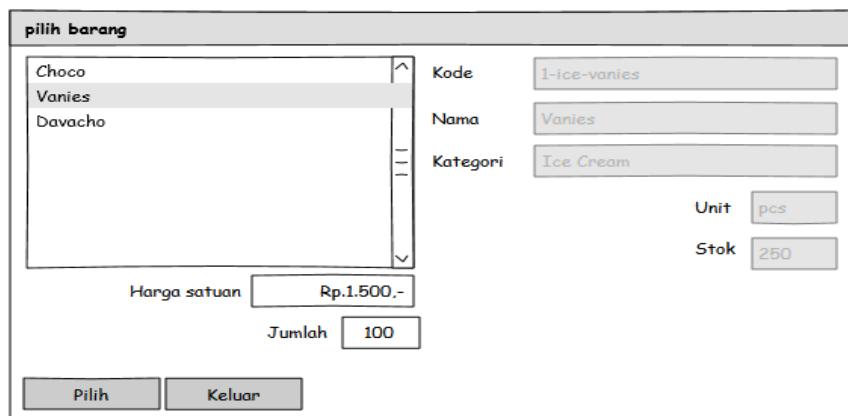
Aksi	Kode barang	Nama Barang	unit	qty	harga	sub total
<input checked="" type="checkbox"/>	1-ice-vanies	Vanes	pcs	100	Rp.1.250,-	Rp.125.000,-
<input checked="" type="checkbox"/>	1-ice-chovo	Choco	pcs	100	Rp.1.000,-	Rp.100.000,-
<input checked="" type="checkbox"/>	1-material-hanco	Hanco	pcs	1	Rp.60.000,-	Rp.60.000,-

Gambar 4.39 Perancangan Input Data Penjualan

Alur untuk menginput data pesanan penjualan adalah sebagai berikut:

1. Tanggal, secara otomatis tanggal akan menyesuaikan dengan tanggal yang ada pada sistem operasi.
2. Konsumen, karena ini pesanan penjualan maka harus memilih konsumen. *Field* konsumen ini berjenis *combobox*.
3. Status pengiriman, terdiri dari sebagai berikut:

- a. Kirim, data transaksi penjualan tersebut akan dimasukan ke form pengiriman penjualan.
 - b. Langsung diambil, data transaksi penjualan tersebut tidak akan melalui proses pengiriman.
4. Klik tombol tambah produk kemudian akan menampilkan dialog seperti berikut:



Gambar 4.40 Perancangan Input Data Pilih Barang Penjualan

5. Seleksi nama barang pada tableView atau listView, setelah itu data detail produk akan muncul seperti harga, nama, kategori, unit dan lain-lain.
6. Input jumlah yang dipesan.
7. Setelah itu klik *button* pilih untuk menambahkan produk ke tabel atau Oklik keluar untuk membatalkan menambah produk ke tabel.
8. Kemudian input bayar jika ada konsumen melakukan pembayaran.
9. Lalu klik *button* simpan untuk menyimpan transaksi penjualan.

e. Perancangan input data pengiriman penjualan

Input data pengiriman penjualan digunakan untuk membuat surat jalan kepada supir bedasarkan transaksi penjualan yang telah dilakukan. Perancangan input data pengiriman penjualan terdiri dari tanggal, keterangan, daftar penjualan yang akan dikirimkan, biaya. Berikut ini adalah perancangannya:

Aksi	kode transaksi	no kontak	nama konsumen	alamat
<input checked="" type="checkbox"/>	SALES_INV-20151222-1	(022) XXX-XXXX-XXXX	asep	jl.komp bukit indah no 02
<input checked="" type="checkbox"/>	SALES_INV-20151222-2	(022) XXX-XXXX-XXXX	ujeh	jl.komp bukit indah no 113
<input type="checkbox"/>	SALES_INV-20151222-3	(022) XXX-XXXX-XXXX	firman	jl.komp bukit indah no 522

Tanggal: 2015-12-22
Keterangan: -
Biaya: Rp.0.-
Simpan Batal

Gambar 4.41 Perancangan Input Data Pengiriman Penjualan

Alur penggunaan atau menginput data pengiriman penjualan adalah sebagai berikut:

1. Tanggal, secara otomatis tanggal akan terisi dengan tanggal di sistem operasi.
2. Keterangan, boleh tidak diisi.
3. Daftar penjualan, dalam tabel terdapat kolom aksi berisi checkbox. Ceklis checkbox tersebut sesuai dengan kode transaksi yang akan dikirimkan.
4. Input biaya yang dikeluarkan.
5. Klik button simpan.

f. Perancangan input data pembelian

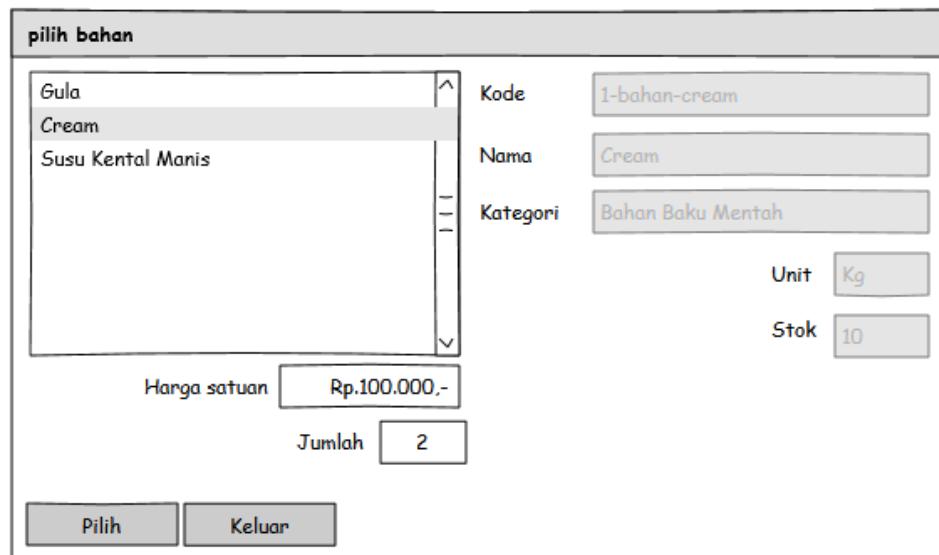
Input data pembelian digunakan untuk memcatat transaksi pembelian, menambahkan stok bahan digudang dan nantinya akan diproses pada laporan keuangan untuk biaya pengeluaran. Perancangan input data pembelian terdiri dari pemasok, tanggal, keterangan, daftar barang, pembayaran. Berikut ini adalah perancangannya:

Aksi	Kode barang	Nama Barang	unit	qty	harga	sub total
<input checked="" type="checkbox"/>	1-bahan-gula	Gula	kg	10	Rp.75.000,-	Rp.750.000,-
<input checked="" type="checkbox"/>	1-bahan-cream	Fullcream	kg	1	Rp.100.000,-	Rp.100.000,-

Gambar 4.42 Perancangan Input Data Pembelian

Alur penggunaan input data pembelian berdasarkan gambar diatas adalah sebagai berikut:

1. Pemasok, pilih pemasok berdasarkan data yang tersedia pada *combobox*.
2. Tanggal, tanggal secara otomatis akan diisi sesuai dengan tanggal di sistem operasi
3. Klik *button* tambah bahan, kemudian akan menampilkan dialog sebagai berikut:



Gambar 4.43 Perancangan Input Data Pilih Bahan Baku

4. Pilih barang yang ada pada *tableView* atau *listView*, maka data barang akan tambil ke *textfield* seperti harga, unit, kategori dan lain-lain.
5. Input jumlah barang yang dibeli,
6. Input harga barang sesuai dengan struk pembelian.
7. Klik *button* pilih, maka dialog pilih bahan akan di-*close* dan data yang dipilih akan masuk ke dalam tabel pembelian bahan.
8. Input pembayaran jika ada pembayaran.
9. Klik *button* simpan, untuk menyimpan transaksi pembelian.

g. Perancangan input data pembayaran

Input data pembayaran dilakukan untuk memcatat biaya pengeluaran seperti service mesin kulkas, service mobil, dan lain-lain. Perancangan input data pembayaran terdiri dari tanggal, kategori pembayaran, biaya dan keterangan. Berikut ini adalah perancangannya:

pembayaran lain-lain	
Tanggal	2015-12-22
Kategori pembayaran	(PLN) Pembayaran Listrik
Biaya	Rp.500.000,-
Keterangan	-
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 4.44 Perancangan Input Data Pembayaran

Aturan menggunakan atau menginput data transaksi pembayaran lain-lain adalah sebebagai berikut:

1. Tanggal, tanggal akan secara otomatis diisi dengan tanggal yang sama dengan sistem operasi pada saat mengakses form pembayaran lain-lain.
2. Kategori pembayaran, harus dipilih tidak boleh kosong.
3. Biaya, tidak boleh lebih kecil sama dengan Rp.0,-
4. Keterangan, boleh kosong.

h. Perancangan input data penggajian

Input data penggajian digunakan untuk menghitung upah yang harus dikeluarkan oleh perusahaan kepada karyawan yang telah melakukan produksi. Perancangan input data penggajian terdiri dari karyawan, tanggal, daftar produksi produks, daftar barang yang telah diproduksi. Berikut ini adalah perancangannya:

penggajian				
karyawan	(pro) Dimas Maryanto	pilih	Kode Produksi	Tanggal
Tanggal	2015-12-22	<input checked="" type="checkbox"/>	PROD-20151222-1	2015-12-22
		<input type="checkbox"/>	PROD-20151221-2	2015-12-21
		<input checked="" type="checkbox"/>	PROD-20151218-3	2015-12-18
kode	Nama produk	qty	harga	sub total
1-ice-choco	Choco	250	Rp.150,-	Rp.37.500,-
1-ice-vanies	Vanies	150	Rp.100,-	Rp.15.000,-
1-ice-davacho	Davacho	50	Rp.50	Rp.2500,-
			Gaji pokok	Rp.55.000,-
			Bonus	Rp.5.000,-
			Total	Rp.60.000,-
Simpan		Batal		

Gambar 4.45 Perancangan Input Data Penggajian Karyawan

Alur menggunakan atau menginput data penggajian karyawan bagian produksi ini adalah sebagai berikut:

1. Karyawan, secara otomatis akan diisi dengan nama karyawan yang telah dipilih pada saat mengakses menu penggajian.
2. Tanggal, secara otomatis akan diisi dengan tanggal yang sesuai dengan sistem operasi pada saat mengakses form tersebut.

3. Daftar produksi, pada tabel daftar produksi terdapat kolom pilih yang berisikan komponen checkbox, cheklis checkbox tersebut untuk melakukan perhitungan berapa jumlah barang yang diproduksi.
4. Daftar barang, input harga per produk dengan cara *double click* pada cell harga sesuai dengan kode atau nama produk, setelah itu akan tampil textfield, kemudian inputkan nominalnya.
5. Setelah itu sistem akan menghitung gaji pokok.
6. Input bonus jika ada
7. Kemudian hitung total.
8. Klik button simpan, untuk menyimpan transaksi penggajian karyawan bagaian produksi.

i. Perancangan input data kasbon karyawan

Input data kasbon digunakan untuk mencatat peminjaman dana untuk karyawan. Perancangan input data kasbon karyawan terdiri dari karyawan, tanggal, dana pinjaman, biaya yang telah dibayar. Berikut ini adalah perancangannya:

kasbon	
Karyawan	(pro) Dimas Maryanto
Tanggal	2015-12-22
Pinjam	Rp.500.000,-
Dibayar	Rp.500.000,-
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 4.46 Perancangan Input Data Kasbon Karyawan

Alur menggunakan atau menginput data penggajian karyawan bagian produksi ini adalah sebagai berikut:

1. Karyawan, secara otomatis akan diisi dengan nama karyawan yang telah dipilih pada saat mengakses menu penggajian.
2. Tanggal, secara otomatis akan diisi dengan tanggal yang sesuai dengan sistem operasi pada saat mengakses form tersebut.
3. Input biaya pinjaman kepada karyawan.
4. Input biaya yang telah dibayar oleh karyawan
5. Klik *button simpan*, untuk menyimpan transaksi tersebut.

4.2.2. Perancangan output

Perancangan output merupakan hasil dari proses yang telah dilakukan oleh program aplikasi. Berikut ini perancangan output:

1. Perancangan output pesanan penjualan.

Dallas Ice Cream jl. cendana no.12 Lembang					SALES ORD-20151222-1
SALES ORDER					Toko / Tuan : Asep
					alamat: -
waktu cetak : 2015-12-22 / 21:31 PM					
No.	Nama Barang	unit	qty	keterangan	
1	Vanies	pcs	10	-	
2	Davacho	pcs	25	-	
3	Fruitmilk	pcs	20	-	
4	Galon	pcs	1	-	
5	Bunnut	pcs	20	-	
ketua bagian produksi (nama jelas)					

Gambar 4.47 Perancangan Output Surat Pesanan Penjualan

2. Perancangan output kwitansi penjualan

Dallas Ice Cream jl. cendana no 12 Lembang					SALES INV-20151222-1
SALES INVOICE					Toko / Tuan: Asep
					Alamat: jl.bukit indah no 15 rt.06 rw.18 kabupaten bandung
No.	Nama barang	QTY	Harga	Sub Total	
1	Davacho	10	Rp.1.250,-	Rp.12.500,-	
2	Vanies	20	Rp.1.000,-	Rp.20.000,-	
3	Fruitmilk	10	Rp.650	Rp.6.500,-	
4	Galon	2	Rp.60.000,-	Rp.120.000,-	
Konfirmasi: Phone: - BBM: - WA: - HP: -		Hormat kami (ketua produksi) (nama jelas)		Konsumen (nama jelas)	Total Rp. Uang muka Rp. Sisa Rp.

Gambar 4.48 Perancangan Output Kwitansi Penjualan

3. Perancangan output surat jalan.

Dallas Ice Cream		SALES_DEL-20151222-1
jl. cendana no 12 Lembang		Supir: Ujang
SALES DELIVERY		
No.	Sales Invoice	Keterangan
1	SALES_INV-20151222-3	
2	SALES_INV-20151222-32	
3	SALES_INV-20151222-12	
4	SALES_INV-20151222-5	
Konfirmasi:		
Phone: -		
BBM: -		
WA: -		
HP: -		

Gambar 4.49 Perancangan Output Surat Jalan Pengiriman Penjualan

4. Perancangan output surat keterangan produksi.

Dallas Ice Cream		waktu cetak : 2015-12-22 / 21:31 PM
jl. cendana no.12 Lembang		
PRODUCTIONS		
No.	Nama Barang	unit
1	Vanies	pcs
2	Davacho	pcs
3	Fruitmilk	pcs
4	Galon	pcs
5	Burnnut	pcs
ketua bagian produksi		
(nama jelas)		

Gambar 4.50 Perancangan Output Surat Keterangan Produksi

5. Perancangan output laporan keuangan.

Dallas Ice Cream			
jl. cendana no.12 Lembang			
waktu cetak : 2015-12-22 / 21:31 PM			
FINANCIAL STATEMENT			
Periode: 2015-NOV-30 sd 2015-DES-31			
Pendapatan		PENDAPATAN BERSIH	
Total Penjualan	Rp.11.250.000,-	Penjualan Bersih	Rp.11.000.000,-
Piutang	Rp.250.000,-	Biaya Operasional	(Rp.2.000.000,-)
Penjualan Bersih	Rp.11.000.000,-	Beban lain-lain	(Rp.3.500.000,-) +
Beban operational		Total	Rp.5.500.000,-
transportasi	Rp.1.250.000,-		
Produksi	Rp.550.000,-		
pembayaran lain-lain	Rp.250.000,-		
total biaya operasional	Rp.2.000.000,-		
Beban lain-lain			
pembelian	Rp.1.000.000,-		
kasbon	Rp.2.500.000,-		
total beban lain-lain	Rp.3.500.000,-		

Gambar 4.51 Perancangan Output Laporan Keuangan

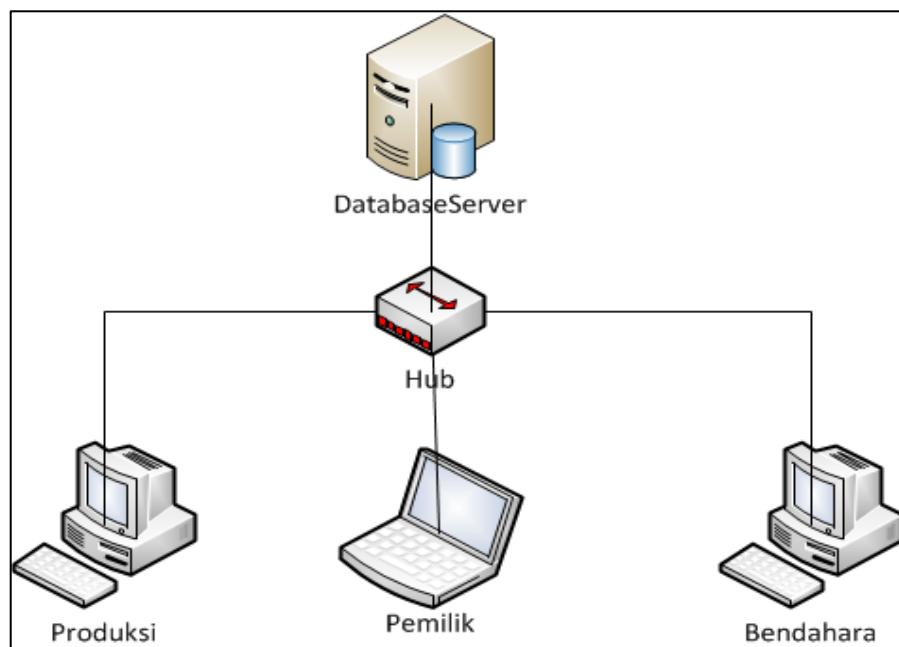
6. Perancangan output laporan produksi per karyawan

Dallas Ice Cream			
jl. cendana no.12 Lembang			
waktu cetak : 2015-12-22 / 21:31 PM			
LAPORAN PRODUKSI PEGAWAI			
periode: 2016-01-01 s/d 2016-01-31			
nama karyawan : Ecep			
Nama Barang	Satuan	Jumlah	
Cup Kecil 70ml	pcs	1020	
Cup Besar 100ml	pcs	506	
Galon 8 Liter	galon	5	
	Jumlah	1531	

Gambar 4.52 Perancangan Output Laporan Produksi per Karyawan

4.3. Perancangan Arsitektur Jaringan

Arsitektur jaringan yang digunakan adalah menggunakan topologi jaringan *star* jadi topologi jaringan ini menggunakan sebuah konsentrator atau yang lebih dikenal dengan *switch* atau *hub*, berikut ini adalah arsitektur jaringan yang digunakan pada penelitian ini:



Gambar 4.53 Arsitekur Jaringan

Pada gambar diatas terdapat 4 komputer yang terhubung ke jaringan yaitu komputer DatabaseServer adalah komputer yang digunakan untuk menyimpan data atau berfungsi sebagai *storage*, komputer Produksi adalah komputer yang digunakan oleh karyawan bagian produksi, komputer pemilik adalah komputer yang digunakan oleh pemilik perusahaan dan komputer Bendahara adalah komputer yang digunakan untuk bendahara.

4.4. Pengujian

Pengujian perangkat lunak atau *software testing* merupakan hal yang paling utama setelah melakukan proses koding, pengujian akan dilakukan dengan menggunakan *black-box testing* karena keterbatasan waktu dan untuk menghemat biaya. Pengujian dilakukan oleh penulis dan pengguna akhir yaitu pemilik perusahaan bpk.Hari Hermawan.

4.4.1. Rencana pengujian

Berikut ini adalah rencana pengujian *software* yang akan dilakukan:

Tabel 4.35 Rencana Pengujian

<i>Test id</i>	Kelas uji	Bulir uji
1.	Login aplikasi	a. <i>Sign-in</i> aplikasi dengan akses “administrator” b. <i>Sign-in</i> aplikasi dengan akses “produksi” c. <i>Sign-in</i> aplikasi dengan akses “bendahara”
2.	Data master	a. Input data jabatan b. Input data karyawan c. Input data kategori barang d. Input data satuan barang e. Input data barang f. Input data konsumen g. Input data pemasok h. Input data kategori pembayaran

Test id	Kelas uji	Bulir uji
3.	Data transaksi	<ul style="list-style-type: none"> a. Input data produksi barang b. Input data penggunaan barang c. Input data pesanan penjualan d. Jual pesanan penjualan e. Input data penjualan f. Input data biaya transportasi g. Input data pembelian h. Input data pembayaran lain-lain i. Input data penggajian. j. Input data kasbon.

4.4.2. Kasus dan hasil pengujian

Berikut ini adalah hasil pengujian menggunakan *black-box testing* berdasarkan rencana pengujian:

1. Kasus dan hasil pengujian *login* aplikasi, diasumsikan bahwa telah terdapat data didalam *database* pada tabel “*mst_account*” ada tiga data sebagai berikut:

Tabel 4.36 Contoh Data Pemakai

username	password	level
admin	admin	administrator
hari	hari	produksi
asep	asep	bendahara

- a. *Sign-in* aplikasi dengan akses “administrator”

Tabel 4.37 Kasus Dan Hasil Login Aplikasi Sebagai Admin

Kasus dan hasil pengujian proses login sebagai admin (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik button “login”	Menampilkan <i>form login</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Input username “admin” dan password “admin”	<i>form login</i> di-close dan semua menu diaktifkan	Sesuai dengan harapan	[√] diterima [] ditolak
Klik button “logout”	semua menu dinon aktifkan dan tidak dapat diakses	Sesuai dengan harapan	[√] diterima [] ditolak

Kasus dan hasil pengujian proses login sebagai admin (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik button “login”	Menampilkan <i>form login</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Input username “admin” dan password “adm”	Menampilkan notifikasi “maaf username dan password salah”.	Sesuai dengan harapan	[√] diterima [] ditolak

b. *Sign-in* aplikasi dengan akses “produksi”

Tabel 4.38 Kasus Dan Hasil Pengujian Proses Login Sebagai Produksi

Kasus dan hasil pengujian proses login sebagai produksi (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik button “login”	Menampilkan <i>form login</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Input username “hari” dan password “hari”	<i>form login</i> di-close dan mengaktifkan menu data master, pemesaman penjualan, penjualan, produksi, penggunaan bahan, penggajian dan kasbon karyawan	Sesuai dengan harapan	[√] diterima [] ditolak
Klik button “logout”	semua menu di-non aktifkan dan tidak dapat diakses	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses login sebagai produksi (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik button “login”	Menampilkan <i>form login</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Input username “hari” dan password “hr”	Menampilkan notifikasi “maaf username dan password salah”.	Sesuai dengan harapan	[√] diterima [] ditolak

c. *Sign-in* aplikasi dengan akses “bendahara”

Tabel 4.39 Kasus Dan Hasil Pengujian Proses Login Sebagai Bendahara

Kasus dan hasil pengujian proses login sebagai produksi (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik button “login”	Menampilkan form <i>login</i>	Sesuai dengan harapan	[✓] diterima [] ditolak
Input username “asep” dan password “asep”	<i>form login</i> di-close dan mengaktifkan menu data master, pembelian dan pembayaran	Sesuai dengan harapan	[✓] diterima [] ditolak
Klik button “logout”	semua menu di-non aktifkan dan tidak dapat diakses	Sesuai dengan harapan	[✓] diterima [] ditolak
Kasus dan hasil pengujian proses login sebagai produksi (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik button “login”	Menampilkan form <i>login</i>	Sesuai dengan harapan	[✓] diterima [] ditolak
Input username “asep” dan password “aep”	Menampilkan notifikasi “maaf username dan password salah”.	Sesuai dengan harapan	[✓] diterima [] ditolak

2. Data master

- a. Kasus dan hasil pengujian input data jabatan, diasumsikan data yang telah tersedia pada tabel “*mst_department*” di basis data adalah sebagai berikut:

Tabel 4.40 Contoh Data Jabatan

kode	nama	keterangan
prod	Produksi	Karyawan bagian produksi
own	Owner	Pemilik perusahaan
bend	Bendahara	-

Berikut adalah kasus dan hasil pengujian input data jabatan:

Tabel 4.41 Hasil Pengujian Proses Input Data Jabatan

Kasus dan hasil pengujian proses input data jabatan (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data jabatan dengan kode “spr”, nama “Supir” dan keterangan “-“	Data berhasil disimpan dan menampilkan notifikasi “data jabatan berhasil disimpan”	Sesuai dengan harapan	[✓] diterima [] ditolak

Kasus dan hasil pengujian proses input data jabatan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data jabatan dengan kode “own”, nama “Pemilik” dan keterangan “-“	Tampilkan pesan kesalahan “Tidak dapat menyimpan data jabatan karena kode jabatan telah tersedia”	Sesuai dengan harapan	[√] diterima [] ditolak

- b. Input data karyawan, diasumsikan berikut ini adalah data yang telah tersedia pada tabel “*mst_employee*” di basis data:

Tabel 4.42 Contoh Data Karyawan

kode	nama	jabatan
1	Hari hermawan	own
2	Asep	prod

Berikut ini adalah kasus dan hasil pengujian input data karyawan:

Tabel 4.43 Hasil Pengujian Input Data Karyawan

Kasus dan hasil pengujian proses input data karyawan (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data karyawan dengan nama “usep”, dan pilih jabatan “prod”	Data berhasil disimpan dan menampilkan notifikasi “data karyawan berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak

Kasus dan hasil pengujian proses input data karyawan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data jabatan dengan data tidak lengkap atau tidak diisi	<i>Button “Simpan” tidak bisa diklik dan tidak bisa menyimpan data karyawan.</i>	Sesuai dengan harapan	[✓] diterima [] ditolak

- c. Input data kategori barang, diasumsikan berikut ini adalah data yang telah tersedia pada tabel “*mst_category_of_item*” di basis data:

Tabel 4.44 Contoh Data Kategori Barang

kode	nama	keterangan
ice	Ice Cream	-
b_baku_m	Bahan Baku Mentah	Bahan baku mentah seperti gula, garam, cream dan lain-lain
b_baku_j	Bahan Baku Jadi	Bahan baku siap pakai contohnya hanco dan lain-lain.
asset	Harta tetap	Contohnya kendaraan mobil, motor, dan lain-lain.

Berikut ini adalah kasus dan hasil pengujian proses input data kategori barang:

Tabel 4.45 Hasil Pengujian Proses Input Data Kategori Barang

Kasus dan hasil pengujian			
proses input data kategori barang (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data kategori barang dengan kode “cover”, nama “Kemasan” dan keterangan “-“	Data berhasil disimpan dan menampilkan notifikasi “data kategori barang berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian			
proses input data kategori barang (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data kategori barang dengan data tidak lengkap atau tidak diisi	Button “Simpan” tidak bisa diklik dan tidak bisa menyimpan data kategori barang.	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian			
proses input data kategori barang (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data kategori barang dengan kode “ice”, nama “Ice Cream” dan keterangan “-“	Tampilkan pesan kesalahan “tidak dapat menyimpan data kategori barang karena kode barang telah tersedia”	Sesuai dengan harapan	[√] diterima [] ditolak

- d. Input data satuan barang, diasumsikan data berikut ini adalah data yang telah tersedia pada tabel “*mst_unit*” pada basis data:

Tabel 4.46 Contoh Data Satuan Barang

kode	nama	keterangan
pcs	PCs	-
kg	Kg	Kilogram

Berikut ini adalah kasus dan hasil pengujian proses input data satuan barang:

Tabel 4.47 Hasil Pengujian Proses Input Satuan Barang

Kasus dan hasil pengujian			
proses input data satuan barang (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data satuan barang dengan kode “gr”, nama “Gram” dan keterangan “-“	Data berhasil disimpan dan menampilkan notifikasi “data satuan barang berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian			
proses input data satuan barang (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data satuan barang dengan data tidak lengkap atau tidak diisi	Button “Simpan” tidak bisa diklik dan tidak bisa menyimpan data satuan barang.	Sesuai dengan harapan	[√] diterima [] ditolak

Kasus dan hasil pengujian proses input data satuan barang (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data kategori barang dengan kode “pcs”, nama “PCS” dan keterangan “-“	Tampilkan pesan kesalahan “tidak dapat menyimpan data satuan barang karena kode telah tersedia”	Sesuai dengan harapan	[√] diterima [] ditolak

- e. Input data gudang, diasumsikan berikut ini adalah data yang terdapat pada tabel “*mst_warehouse*” di dalam basisdata:

Tabel 4.48 Contoh Data Gudang

kode	nama	Kontak personal
1	CENTRAL	-
2	Kantor Cabang	-

Berikut ini adalah kasus dan hasil pengujian input data gudang:

Tabel 4.49 Hasil Pengujian Input Data Gudang

Kasus dan hasil pengujian proses input data gudang (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data gudang dengan nama “Kantor cabang sumedang”, dan kontak personal “-”	Data berhasil disimpan dan menampilkan notifikasi “data gudang berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak

f. Input data barang, diasumsikan berikut ini adalah data yang telah tersedia pada tabel “mst_items” di dalam basis data:

Tabel 4.50 Contoh Data Barang

kode	nama	kategori	unit	gudang
1-ice-choco	Choco	ice	pcs	1
1-ice-davacho	Davacho	ice	pcs	1

Berikut ini adalah kasus dan hasil pengujian input data barang:

Tabel 4.51 Hasil Pengujian Input Data Barang

Kasus dan hasil pengujian proses input data barang (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data barang dengan nama “loly”, pilih kategori “ice”, pilih unit “pcs” dan pilih gudang “CENTRAL”	Data berhasil disimpan dan menampilkan notifikasi “data barang berhasil disimpan”	Sesuai dengan harapan	[✓] diterima [] ditolak
Kasus dan hasil pengujian proses input data barang (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data barang dengan data tidak lengkap atau tidak diisi	Button “Simpan” tidak bisa diklik dan tidak bisa menyimpan data barang.	Sesuai dengan harapan	[✓] diterima [] ditolak

Kasus dan hasil pengujian proses input data barang (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data barang dengan nama “Choco”, pilih kategori “pcs”, pilih satuan barang “pcs” dan pilih gudang “CENTRAL”	Tampilkan pesan kesalahan “tidak dapat menyimpan data barang karena kode barang telah tersedia di gudang “CENTRAL”	Sesuai dengan harapan	[√] diterima [] ditolak

g. Input data konsumen, diasumsikan berikut ini adalah data yang telah tersedia pada tabel “*mst_customers*” didalam basisdata:

Tabel 4.52 Contoh Data Konsumen

kode	nama	Kontak personal
1	Hendri subang	-
2	Jaja garut	-

Berikut ini adalah kasus dan hasil pengujian proses input data konsumen:

Tabel 4.53 Hasil Pengujian Proses Input Data Konsumen

Kasus dan hasil pengujian proses input data konsumen (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data konsumen dengan nama “usman bandung”, dan kontak personal “-”	Data berhasil disimpan dan menampilkan notifikasi “data konsumen berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses input data konsumen (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data konsumen yang tidak lengkap	Button “Simpan” tidak bisa di klik dan tidak dapat menyimpan data konsumen	Sesuai dengan harapan	[√] diterima [] ditolak

- h. Input data pemasok, diasumsikan berikut ini adalah data yang telah tersedia pada tabel “*mst_suplayers*” didalam basisdata:

Tabel 4.54 Contoh Data Suplayer

kode	nama	Kontak personal
1	supermarket	-
2	minimarket	-

Berikut ini adalah kasus dan hasil pengujian proses input data suplayer:

Tabel 4.55 Hasil Pengujian Proses Input Data Suplayer

Kasus dan hasil pengujian proses input data suplayer (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data konsumen dengan nama “borma”, dan kontak personal “-”	Data berhasil disimpan dan menampilkan notifikasi “data suplayer berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses input data suplayer (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data konsumen yang tidak lengkap	Button “Simpan” tidak bisa diklik dan tidak dapat menyimpan data konsumen	Sesuai dengan harapan	[√] diterima [] ditolak

- i. Input data kategori pembayaran, diasumsikan berikut ini adalah data yang telah tersedia pada tabel “*mst_category_of_payment*” didalam basisdata:

Tabel 4.56 Contoh Data Kategori Pembayaran

kode	Nama	Keterangan
pln	PLN voucher	Pembayaran listrik dengan voucher
srv	Service	Kategori pembayaran untuk service

Berikut ini adalah kasus dan hasil pengujian proses input data kategori pembayaran:

Tabel 4.57 Hasil Pengujian Proses Input Data Kategori Pembayaran

Kasus dan hasil pengujian proses input data kategori pembayaran (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data jabatan dengan kode “telp”, nama “Telephone” dan keterangan “-“	Data berhasil disimpan dan menampilkan notifikasi “data kategori pembayaran berhasil disimpan”	Sesuai dengan harapan	[✓] diterima [] ditolak

3. Data transaksi

- a. Input data produksi, berikut ini adalah kasus dan hasil pengujian input data produksi:

Tabel 4.58 Hasil Pengujian Input Data Produksi

Kasus dan hasil pengujian proses input data produksi (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data tanggal, pilih karyawan dan jumlah produk yang diproduksi kemudian klik button “Simpan”	Data berhasil disimpan dan menampilkan notifikasi “data produksi berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses input data produksi (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data tanggal, pilih karyawan kemudian klik button “Simpan”	Menampilkan notifikasi “Tidak ada barang yang diproduksi”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses input data produksi (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data produksi tanpa memiliki karyawan	Button “Simpan” tidak dapat diklik dan tidak dapat menyimpan data produksi	Sesuai dengan harapan	[√] diterima [] ditolak

b. Input data penggunaan barang, berikut ini adalah kasus dan hasil pengujian input data penggunaan barang:

Tabel 4.59 Hasil Pengujian Input Data Penggunaan Bahan

Kasus dan hasil pengujian proses input			
data penggunaan bahan (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data tanggal jumlah barang yang digunakan kemudian klik button “Simpan”	Data berhasil disimpan dan menampilkan notifikasi “data penggunaan bahan berhasil disimpan”	Sesuai dengan harapan	[✓] diterima [] ditolak
Kasus dan hasil pengujian proses input			
data penggunaan bahan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Input data tanggal kemudian klik button “Simpan”	Menampilkan notifikasi “Tidak ada barang yang digunakan”	Sesuai dengan harapan	[✓] diterima [] ditolak

c. Input data pesanan penjualan, berikut ini adalah kasus dan hasil pengujian proses input data pemesanan penjualan:

Tabel 4.60 Hasil Pengujian Proses Input Data Pemesanan Penjualan

Kasus dan hasil pengujian proses input			
data pesanan penjualan (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih konsumen	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[✓] diterima [] ditolak
Klik button “Tambah Barang”	Menampilkan daftar barang	Sesuai dengan harapan	[✓] diterima [] ditolak
Pilih barang dan jumlahnya	Menambahkan data barang yang dipilih ke <i>table-view</i> dan menhitung subtotal	Sesuai dengan harapan	[✓] diterima [] ditolak
Klik button “Simpan”	Data berhasil disimpan dan menampilkan notifikasi “Data pesanan penjualan berhasil disimpan”	Sesuai dengan harapan	[✓] diterima [] ditolak

Kasus dan hasil pengujian proses input			
data pesanan penjualan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih konsumen	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Klik <i>button “Simpan”</i>	Menampilkan notifikasi “Tidak ada barang yang dipesan”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses input			
data pesanan penjualan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih konsumen	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Klik <i>button “Tambah Barang”</i>	Menampilkan <i>form</i> daftar barang	Sesuai dengan harapan	[√] diterima [] ditolak
Pilih barang dan jumlahnya 0	Button “Tambah” tidak aktif dan tidak bisa menambahkan barang ke <i>table-view</i>	Sesuai dengan harapan	[√] diterima [] ditolak

- d. Jual pesanan penjualan, berikut ini adalah kasus dan hasil pengujian proses jual pesanan penjualan:

Tabel 4.61 Hasil Pengujian Proses Jual Pesanan Penjualan

Kasus dan hasil pengujian proses jual pesanan penjualan (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih pesanan penjualan dan kemudian klik button “Jual”	Menampilkan <i>form</i> penjualan, tampilkan barang yang dipesan berserta jumlahnya, hitung subtotal	Sesuai dengan harapan	[√] diterima [] ditolak
Klik button “Simpan”	Data berhasil disimpan dan menampilkan notifikasi “Data penjualan berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses jual pesanan penjualan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih pesanan penjualan dan kemudian klik button “Jual”	Menampilkan <i>form</i> penjualan, tampilkan barang yang dipesan berserta jumlahnya, hitung subtotal	Barang yang dipilih ternyata stoknya tidak mencukupi	[√] diterima [] ditolak
Klik button “Simpan”	Menampilkan notifikasi “produk tidak mencukupi”	Sesuai dengan harapan	[√] diterima [] ditolak

- e. Input data penjualan, berikut ini adalah kasus dan hasil pengujian input data penjualan:

Tabel 4.62 Hasil Pengujian Proses Input Data Penjualan

Kasus dan hasil pengujian proses input data penjualan (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih konsumen	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Klik button “Tambah Barang”	Menampilkan daftar barang	Sesuai dengan harapan	[√] diterima [] ditolak
Pilih barang dan jumlahnya	Menambahkan data barang yang dipilih ke <i>table-view</i> dan menhitung subtotal	Sesuai dengan harapan	[√] diterima [] ditolak
Klik button “Simpan”	Data berhasil disimpan dan menampilkan notifikasi “Data penjualan berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak

Kasus dan hasil pengujian proses input			
data penjualan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih konsumen	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Klik button “Tambah Barang”	Menampilkan daftar barang	Sesuai dengan harapan	[√] diterima [] ditolak
Pilih barang dan jumlahnya sama dengan 0 atau melebihi stok barang yang tersedia	Tidak dapat menambahkan data barang yang dipilih ke <i>table-view</i>	<i>Button “Tambah”</i> tidak aktif	[√] diterima [] ditolak
Kasus dan hasil pengujian proses input			
data penjualan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih konsumen	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Klik button “Simpan”	Menampilkan notifikasi “Tidak ada barang yang dijual”	Sesuai dengan harapan	[√] diterima [] ditolak

f. Input data pengiriman penjualan, berikut ini adalah kasus dan hasil pengujian proses input data pengiriman penjualan:

Tabel 4.63 Hasil Pengujian Proses Input Data Pengiriman Penjualan

Kasus dan hasil pengujian proses input data pengiriman penjualan (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih karyawan	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Ceklik kode transaksi penjualan	Transaksi penjualan tersebut akan dikirimkan	Sesuai dengan harapan	[√] diterima [] ditolak
Klik <i>button “Simpan”</i>	Data berhasil disimpan dan menampilkan notifikasi “Data pengiriman berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses input data pengiriman penjualan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih karyawan	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Klik <i>button “Simpan”</i>	Menampilkan notifikasi “tidak ada penjualan yang dikirimkan”	Sesuai dengan harapan	[√] diterima [] ditolak

g. Input data pembelian

Tabel 4.64 Hasil Pengujian Proses Input Data Pembelian

Kasus dan hasil pengujian proses input data pembelian (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih suplayer	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Klik button “Tambah Barang”	Menampilkan daftar barang	Sesuai dengan harapan	[√] diterima [] ditolak
Pilih barang, jumlahnya dan harganya	Menambahkan data barang yang dipilih ke <i>table-view</i> dan menhitung subtotal	Sesuai dengan harapan	[√] diterima [] ditolak
Klik button “Simpan”	Data berhasil disimpan dan menampilkan notifikasi “Data pembelian berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses input data pembelian (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih suplayer	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Klik button “Simpan”	Mampulkan notifikasi “Tidak ada barang yang dibeli”	Sesuai dengan harapan	[√] diterima [] ditolak

h. Input data penggajian, berikut ini adalah kasus dan hasil pengujian proses input data penggajian karyawan bagian produksi:

Tabel 4.65 Hasil Pengujian Proses Penggajian Karyawan

Kasus dan hasil pengujian proses			
input data penggajian karyawan (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih karyawan	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Ceklis kode transaksi produksi yang telah dilakukan	Menghitung jumlah per produk yang telah diproduksi	Sesuai dengan harapan	[√] diterima [] ditolak
Input upah per produk	menhitung subtotal	Sesuai dengan harapan	[√] diterima [] ditolak
Klik <i>button “Simpan”</i>	Data berhasil disimpan dan menampilkan notifikasi “Data penggajian karyawan berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak

Kasus dan hasil pengujian proses			
input data penggajian karyawan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih karyawan	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Klik <i>button “Simpan”</i>	Menampilkan notifikasi “Tidak ada produk yang diproduksi”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses			
input data penggajian karyawan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih karyawan	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Ceklis kode transaksi produksi yang telah dilakukan	Menghitung jumlah per produk yang telah diproduksi	Sesuai dengan harapan	[√] diterima [] ditolak
Klik <i>button “Simpan”</i>	Menampilkan notifikasi “Tidak ada ubah yang dikeluarkan”	Sesuai dengan harapan	[√] diterima [] ditolak

- i. Input data kasbon, berikut ini adalah kasus dan hasil pengujian proses input data kasbon:

Tabel 4.66 Hasil Pengujian Proses Input Data Kasbon Karyawan

Kasus dan hasil pengujian proses			
input data kasbon karyawan (<i>Data valid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih karyawan	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Input biaya kasbon	Menghitung jumlah per produk yang telah diproduksi	Sesuai dengan harapan	[√] diterima [] ditolak
Klik <i>button “Simpan”</i>	Data berhasil disimpan dan menampilkan notifikasi “Data kasbon karyawan berhasil disimpan”	Sesuai dengan harapan	[√] diterima [] ditolak
Kasus dan hasil pengujian proses			
input data kasbon karyawan (<i>Data invalid</i>)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih karyawan	<i>Button “Simpan” aktif</i>	Sesuai dengan harapan	[√] diterima [] ditolak
Klik <i>button “Simpan”</i>	Tampilkan notifikasi “Jumlah kasbon Rp.0”	<i>Button “Simpan” tidak aktif</i>	[√] diterima [] ditolak

4.4.3. Kesimpulan hasil pengujian

Berdasarkan pengujian yang dilakukan secara minimal tersebut, maka dapat ditarik kesimpulan bahwa perangkat lunak yang dibuat cukup baik dan secara fungsional outputnya rata-rata memiliki hasil sesuai dengan apa yang diharapkan.

4.5. Implementasi

Implementasi sistem merupakan tahap selanjutnya setelah perancangan sebagai perbandingan apakah aplikasi yang telah dibangun sesuai dengan apa yang telah dirancang. Berikut ini adalah penjabaran mengenai implementasi sistem:

4.5.1. Implementasi perangkat lunak

Berdasarkan *deployment diagram*, perangkat lunak atau software yang akan digunakan menggunakan teknologi java, maka untuk menjalankan program berbasis *desktop* membutuhkan *java runtime environment* (JRE). Berikut ini adalah proses install jre:

1. *Java runtime environment* dapat diunduh pada alamat website berikut ini
<http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>
2. kemudian pilih sesuai dengan sistem operasi yang digunakan. Karena penulis menggunakan sistem operasi windows 7 64bit maka pilih seperti gambar berikut ini:

Java SE Runtime Environment 8u65		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.		
Product / File Description	File Size	Download
Linux x86	48.98 MB	jre-8u65-linux-i586.rpm
Linux x86	70.46 MB	jre-8u65-linux-i586.tar.gz
Linux x64	46.87 MB	jre-8u65-linux-x64.rpm
Linux x64	68.38 MB	jre-8u65-linux-x64.tar.gz
Mac OS X x64	64.93 MB	jre-8u65-macosx-x64.dmg
Mac OS X x64	55.93 MB	jre-8u65-macosx-x64.tar.gz
Solaris SPARC 64-bit	52.06 MB	jre-8u65-solaris-sparcv9.tar.gz
Solaris x64	49.83 MB	jre-8u65-solaris-x64.tar.gz
Windows x86 Online	0.56 MB	jre-8u65-windows-i586.exe
Windows x86 Offline	47.81 MB	jre-8u65-windows-i586.exe
Windows x86	59.28 MB	jre-8u65-windows-i586.tar.gz
Windows x64	54.29 MB	jre-8u65-windows-x64.exe
Windows x64	62.61 MB	jre-8u65-windows-x64.tar.gz

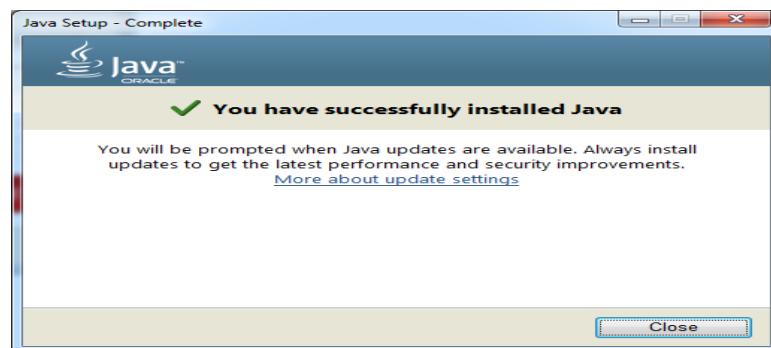
Gambar 4.54 Menguduh Java Runtime Environment

3. lakukan installasi untuk software jre-8uxx-windows-x64.exe dengan cara double klik maka akan muncul dialog sebagai berikut:



Gambar 4.55 Install Java Runtime Environment Versi 1.8

4. setelah itu klick install, jika telah selesai maka akan tampil dialog seperti berikut:



Gambar 4.56 Notifikasi Sukses Menginstal Java Runtime Environment

4.5.2. Implementasi perangkat keras

Perangkat keras yang akan digunakan untuk aplikasi ini adalah sebuah personal komputer atau notebook dengan berbagai macam sistem operasi, aplikasi ini diarancang agar dapat berjalan diberbagai macam *platform* berbasis *desktop*. Berikut ini adalah spesifikasi minimum dan rekomendasi:

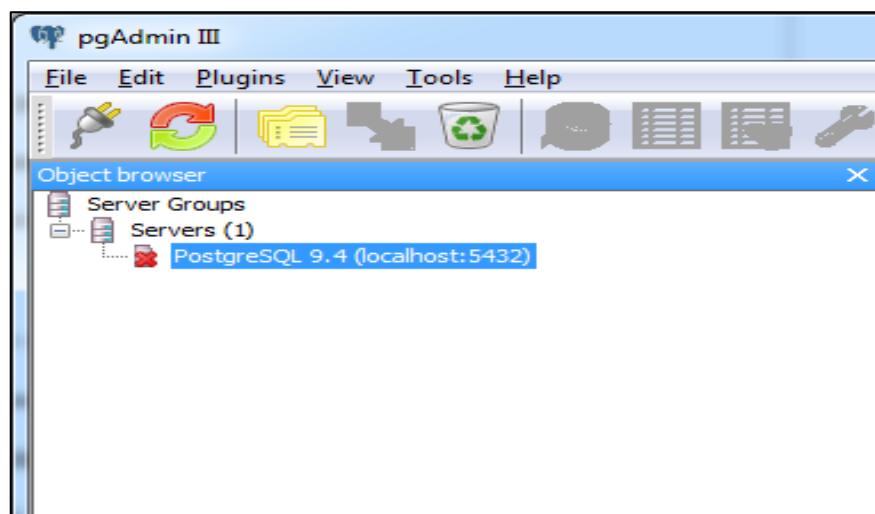
1. Spesifikasi minimum
 - a. CPU : intel celeron (2 core) 1.3 Ghz
 - b. RAM : 512 Mb (tidak termasuk sistem operasi)
 - c. VGA : Intel (*onboard*)
 - d. Hardisk : 128 Gb
 - e. Perangkat input : Mouse, Keyboard
 - f. Perangkat output : Printer
2. Spesifikasi rekomendasi
 - a. CPU : intel core i3 (2 core) 2.2Ghz atau lebih tinggi
 - b. RAM : 2 Gb atau lebih tinggi
 - c. VGA : Intel (*Onboard*)
 - d. Hardisk : 500 Gb atau lebih tinggi
 - e. Perangkat input : Mouse, Keyboard
 - f. Perangkat output : Printer

Aplikasi ini berjalan secara *stand alone* (berdiri sendiri) dan tidak membutuhkan koneksi internet untuk mengaksesnya.

4.5.3. Implementasi basis data

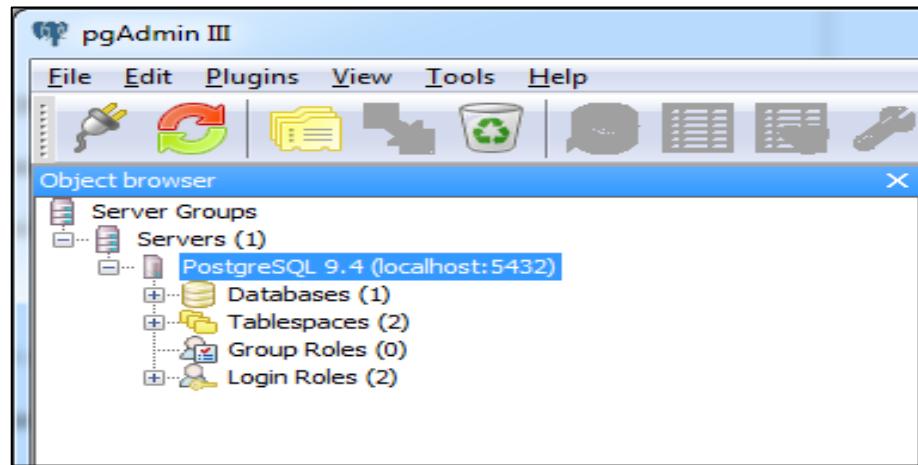
Sebuah sistem informasi tidak akan lepas dari database, berikut ini adalah langkah yang harus dipersiapkan sebelum menggunakan aplikasi:

1. Instalasi database PostgreSQL versi 9.x
2. Membuat *schema database*, berikut ini adalah langkah-langkah untuk membuat *schema di database* PostgreSQL.
 - a. Jalankan pgAdminIII.



Gambar 4.57 Pgadmin3

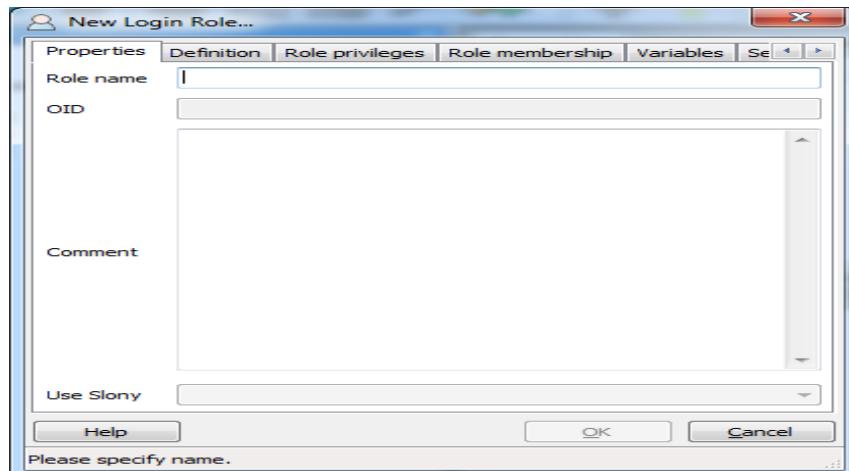
- b. Login sebagai postgres, dengan password yang sesuai proses konfigurasi instalasi PostgreSQL, jika telah login maka hasilnya akan seperti berikut:



Gambar 4.58 Pgadmin3 Setelah Login

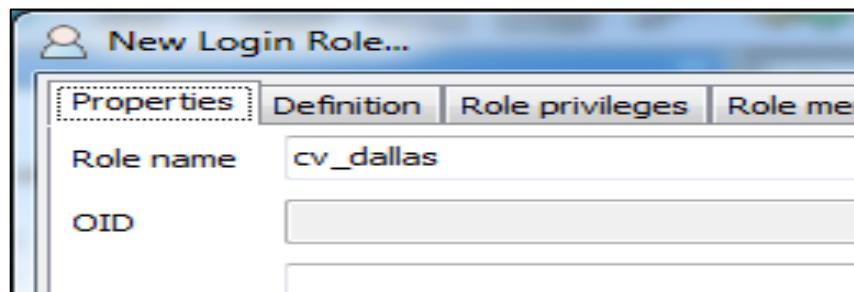
c. Kemudian pilih *Login Role*, setelah itu klik kanan pilih *New Login*

Role akan tampil dialog sebagai berikut:

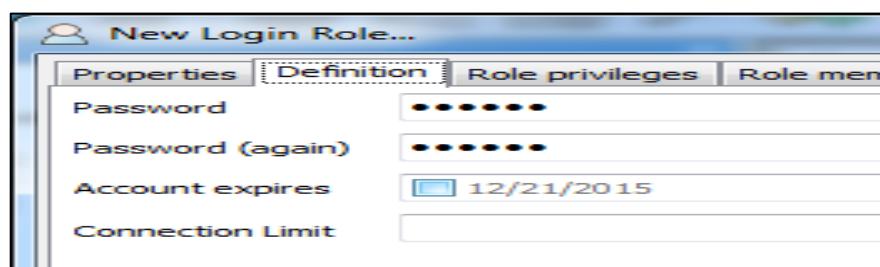


Gambar 4.59 Pgadmin3 New Role

d. Setelah itu pada *tab properties* inputkan *rolename*-nya adalah ‘cv_dallas’ kemudian pindah ke *tab definition password* dan *password (again)* adalah ‘dallas’ seperti berikut:

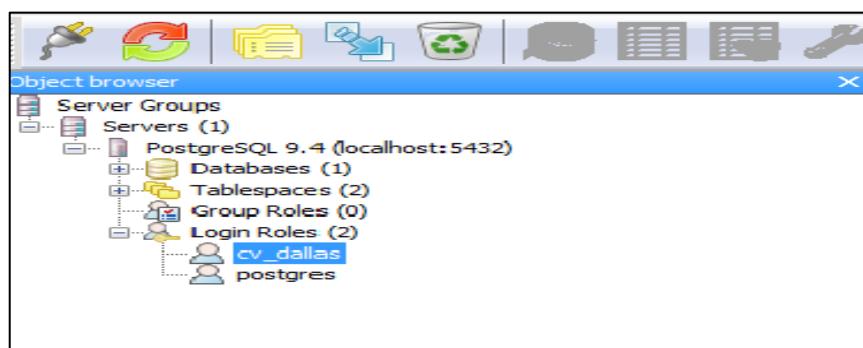


Gambar 4.60 Input New Role Name



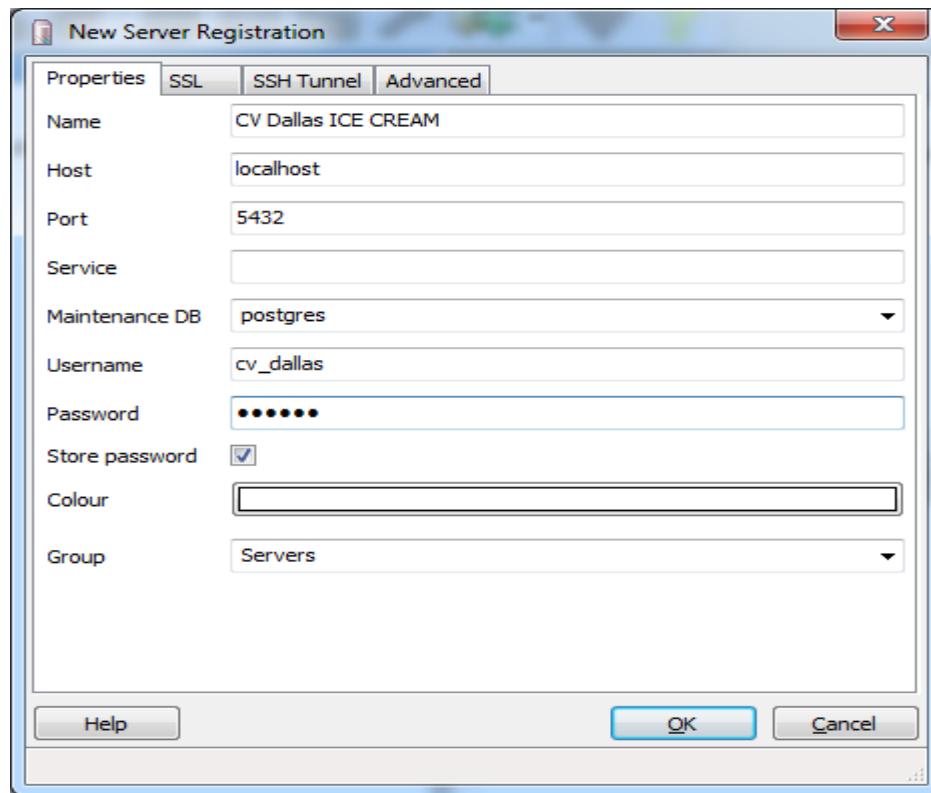
Gambar 4.61 Input Password New Role

- e. Setelah itu klik OK, dan berikut adalah hasilnya:



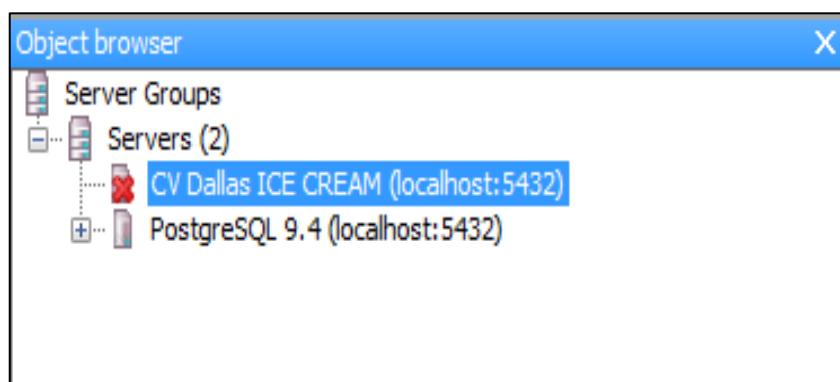
Gambar 4.62 Hasil Membuat Schema

3. Membuat *database*, berikut ini adalah langkah-langkah untuk membuat database di PostgreSQL.
- a. Login sebagai user cv_dallas, klik menu file kemudian add server maka akan tampil seperti berikut:



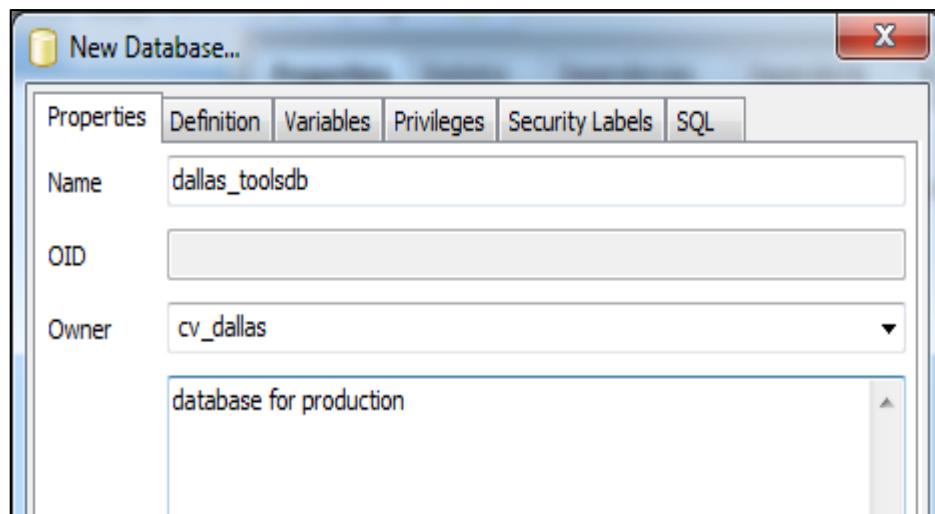
Gambar 4.63 Login Sebagai Schema Cv_Dallas

- b. Kemudian isikan nama, hostname, username dan password seperti pada gambar diatas.
- c. Setelah itu klik OK, maka hasilnya seperti berikut:



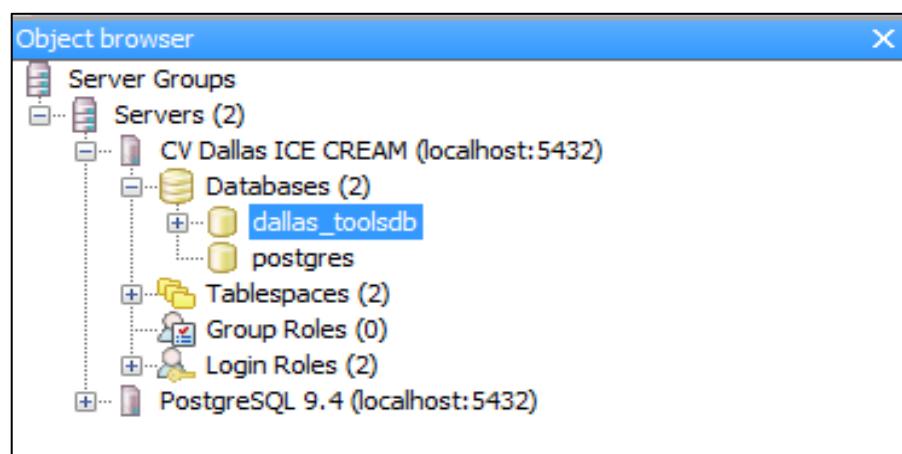
Gambar 4.64 User Cv_Dallas Terdaftar Pada Object Browser

- d. Kemudian klik dua (2) kali setelah itu seleksi *menu item databases*, kemudian klik kanan di *menu item database*, kemudian pilih *new database* maka akan tampil dialog seperti berikut:



Gambar 4.65 Membuat Database

- e. Kemudian isikan name-nya adalah “dallas_toolsdb” dan pilih owner-nya adalah “cv_dallas” kemudian klik OK, maka hasilnya seperti berikut:



Gambar 4.66 Hasil Membuat Database

4. Membuat *table*, proses pembuatan tabel tidak perlu dilakukan karena sudah otomatis dibuatkan ketika aplikasi pertama kali dijalankan berikut ini adalah *script* untuk membuat *table* pada *database*:

a. Tabel *account*

```
-- Table: mst_accounts

-- DROP TABLE mst_accounts;

CREATE TABLE mst_accounts(
    unique_name character varying(25) NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    is_active boolean NOT NULL,
    name character varying(255) NOT NULL,
    last_login timestamp without time zone,
    security_accessed character varying(15) NOT NULL,
    identified_passwd character varying(150) NOT NULL,
    CONSTRAINT mst_accounts_pkey PRIMARY KEY (unique_name)
) WITH (OIDS=FALSE);
ALTER TABLE mst_accounts OWNER TO cv_dallas;
```

b. Tabel *category of item*

```
-- Table: mst_category_of_items

-- DROP TABLE mst_category_of_items;

CREATE TABLE mst_category_of_items(
    category_id character varying(50) NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    category_description text,
    category_name character varying(100) NOT NULL,
    CONSTRAINT mst_category_of_items_pkey PRIMARY KEY
(category_id) WITH (OIDS=FALSE);
ALTER TABLE mst_category_of_items OWNER TO cv_dallas;
```

c. Tabel *customer*

```
-- Table: mst_customers

-- DROP TABLE mst_customers;

CREATE TABLE mst_customers(
    id integer NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    city character varying(100) NOT NULL,
    district character varying(100) NOT NULL,
    pin_code integer,
    rt integer NOT NULL,
    rw integer NOT NULL,
    street_address text NOT NULL,
    customer_name character varying(100) NOT NULL,
    phone_number character varying(50) NOT NULL,
    CONSTRAINT mst_customers_pkey PRIMARY KEY (id)
) WITH (OIDS=FALSE);
ALTER TABLE mst_customers OWNER TO cv_dallas;
```

d. Tabel *department*

```
-- Table: mst_departments

-- DROP TABLE mst_departments;

CREATE TABLE mst_departments(
    job_id character varying(10) NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    job_description text,
    job_name character varying(100) NOT NULL,
    CONSTRAINT mst_departments_pkey PRIMARY KEY (job_id)
) WITH (OIDS=FALSE);
ALTER TABLE mst_departments OWNER TO cv_dallas;
```

e. Tabel *employee*

```
-- Table: mst_employees

-- DROP TABLE mst_employees;

CREATE TABLE mst_employees(
    id integer NOT NULL, created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    city character varying(100) NOT NULL,
    district character varying(100) NOT NULL,
    pin_code integer, rt integer NOT NULL, rw integer NOT NULL,
    street_address text NOT NULL,
    employee_name character varying(100) NOT NULL,
    is_working boolean NOT NULL,
    job_id character varying(10),
    CONSTRAINT mst_employees_pkey PRIMARY KEY (id),
    CONSTRAINT fk923jseu47oircc7rtu7o95dfs FOREIGN KEY (job_id)
        REFERENCES mst_departments (job_id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION)
WITH (OIDS=FALSE);
ALTER TABLE mst_employees OWNER TO cv_dallas;
```

f. Tabel *category of payment*

```
-- Table: mst_payment_category

-- DROP TABLE mst_payment_category;

CREATE TABLE mst_payment_category(
    payment_id character varying(25) NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    description text,
    payment_for character varying(255) NOT NULL,
    CONSTRAINT mst_payment_category_pkey PRIMARY KEY
    (payment_id))WITH (OIDS=FALSE);
ALTER TABLE mst_payment_category OWNER TO cv_dallas;
```

g. Tabel *item*

```
-- Table: mst_items

-- DROP TABLE mst_items;

CREATE TABLE mst_items(
    item_generator integer NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    item_id character varying(25) NOT NULL,
    item_name character varying(100) NOT NULL,
    price_buy double precision NOT NULL,
    price_sell double precision NOT NULL,
    qty integer NOT NULL, is_sell boolean NOT NULL,
    category_id character varying(50) NOT NULL,
    unit_id character varying(25) NOT NULL,
    warehouse_id integer NOT NULL,
    CONSTRAINT mst_items_pkey PRIMARY KEY (item_generator),
    CONSTRAINT fkg99mkujia6t872rd6jins5ff5 FOREIGN KEY
    (category_id)
        REFERENCES mst_category_of_items (category_id) MATCH
    SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT fkqbxg06i2cv2ym7gkk25ma642n FOREIGN KEY
    (warehouse_id)
        REFERENCES mst_warehouse (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT fktlcu4xcufy7mct3sw5ppa7cjj FOREIGN KEY (unit_id)
        REFERENCES mst_unit_of_items (unit_id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT uk_76w5vp8jelvakr8wsgk6891sv UNIQUE (item_id) WITH
    (OIDS=FALSE);
ALTER TABLE mst_items
    OWNER TO cv_dallas;

ALTER TABLE mst_payment_category OWNER TO cv_dallas;
```

h. Tabel suplayer

```
-- Table: mst_suplayers

-- DROP TABLE mst_suplayers;

CREATE TABLE mst_suplayers(
    suplayer_id integer NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    city character varying(100) NOT NULL,
    district character varying(100) NOT NULL,
    pin_code integer,
    rt integer NOT NULL,
    rw integer NOT NULL,
    street_address text NOT NULL,
    suplayer_name character varying(255) NOT NULL,
    contact_person character varying(25),
    CONSTRAINT mst_suplayers_pkey PRIMARY KEY (suplayer_id) WITH
(OIDS=FALSE);
ALTER TABLE mst_suplayers OWNER TO cv_dallas;
```

i. Tabel unit of item

```
-- Table: mst_unit_of_items

-- DROP TABLE mst_unit_of_items;

CREATE TABLE mst_unit_of_items(
    unit_id character varying(25) NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    unit_description text,
    unit_name character varying(100) NOT NULL,
    CONSTRAINT mst_unit_of_items_pkey PRIMARY KEY (unit_id) WITH
(OIDS=FALSE);
ALTER TABLE mst_unit_of_items OWNER TO cv_dallas;
```

j. Tabel *warehouse*

```
-- Table: mst_warehouse

-- DROP TABLE mst_warehouse;

CREATE TABLE mst_warehouse(
    id integer NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    city character varying(100) NOT NULL,
    district character varying(100) NOT NULL,
    pin_code integer, rt integer NOT NULL,
    rw integer NOT NULL, street_address text NOT NULL,
    name character varying(100) NOT NULL,
    phone_number character varying(25) NOT NULL,
    CONSTRAINT mst_warehouse_pkey PRIMARY KEY (id),
    CONSTRAINT uk_kyeuola5shmfwcdggxoslykt2 UNIQUE
(phone_number),
    CONSTRAINT uk_oyj2g712uwcm9lb92tjaghox5 UNIQUE (name))WITH
(OIDS=FALSE); ALTER TABLE mst_warehouse OWNER TO cv_dallas;
```

k. Tabel *employee cash receipt*

```
-- Table: trans_employee_cash_reciept

-- DROP TABLE trans_employee_cash_reciept;

CREATE TABLE trans_employee_cash_reciept(
    id integer NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    amount double precision NOT NULL,
    date date NOT NULL, is_paid boolean NOT NULL,
    payment double precision NOT NULL, employee_id integer,
    CONSTRAINT trans_employee_cash_reciept_pkey PRIMARY KEY
(id),CONSTRAINT fk7r7mwtgpamoqm9k478vtefpxr FOREIGN KEY
(employee_id)
    REFERENCES mst_employees (id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)WITH( OIDS=FALSE); ALTER TABLE trans_employee_cash_reciept
OWNER TO cv_dallas;
```

1. Table *payroll*

```
-- Table: trans_employee_payroll

-- DROP TABLE trans_employee_payroll;

CREATE TABLE trans_employee_payroll(
    id integer NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    amount double precision NOT NULL,
    date date NOT NULL, other_amount double precision NOT NULL,
    employee_id integer,
    CONSTRAINT trans_employee_payroll_pkey PRIMARY KEY (id),
    CONSTRAINT fklyiupseewuauyliefy2ve5tma FOREIGN KEY
(employee_id)
        REFERENCES mst_employees (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT uq_payroll_employee UNIQUE (date, employee_id)
)WITH (OIDS=FALSE);
ALTER TABLE trans_employee_payroll OWNER TO cv_dallas;
```

m. Tabel *payroll details*

```
-- Table: trans_employee_payroll_details

-- DROP TABLE trans_employee_payroll_details;

CREATE TABLE trans_employee_payroll_details(
    trans_id integer NOT NULL,
    production_id integer NOT NULL,
    CONSTRAINT fke6xsmp4phnjcein10chq0kb3x FOREIGN KEY
(production_id)
        REFERENCES trans_production_of_sales (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT fkn97n0p2vmemiqgfs22agmjdmq FOREIGN KEY (trans_id)
        REFERENCES trans_employee_payroll (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT uk_70oxo1mq6p61adfctvmsagmf3 UNIQUE
(production_id)
)WITH ( OIDS=FALSE);
ALTER TABLE trans_employee_payroll_details OWNER TO cv_dallas;
```

n. Tabel *item used*

```
-- Table: trans_item_used

-- DROP TABLE trans_item_used;

CREATE TABLE trans_item_used(
    id integer NOT NULL, created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    date date NOT NULL,
    CONSTRAINT trans_item_used_pkey PRIMARY KEY (id)
) WITH (OIDS=FALSE);
ALTER TABLE trans_item_used OWNER TO cv_dallas;
```

o. Tabel *item used details*

```
-- Table: trans_item_used_details

-- DROP TABLE trans_item_used_details;

CREATE TABLE trans_item_used_details(
    id integer NOT NULL, qty integer NOT NULL, item_id integer,
    item_used integer,
    CONSTRAINT trans_item_used_details_pkey PRIMARY KEY (id),
    CONSTRAINT fkfqxsx038yiuccnrc4h1mhlqb11 FOREIGN KEY (item_id)
        REFERENCES mst_items (item_generator) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT fkjaagprics6p4b0nedybsaqqp9 FOREIGN KEY
(item_used)
        REFERENCES trans_item_used (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
) WITH (OIDS=FALSE);
ALTER TABLE trans_item_used_details OWNER TO cv_dallas;
```

p. Tabel *payment invoice*

```
-- Table: trans_payment_invoice

-- DROP TABLE trans_payment_invoice;

CREATE TABLE trans_payment_invoice
(
    id integer NOT NULL,
    created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    ammount double precision NOT NULL,
    description text,
    trans_date date NOT NULL,
    category_id character varying(25),
    CONSTRAINT trans_payment_invoice_pkey PRIMARY KEY (id),
    CONSTRAINT fkm46rhssnxrxjjm287gwomc9ov FOREIGN KEY
    (category_id)
        REFERENCES mst_payment_category (payment_id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
)WITH (OIDS=FALSE);
ALTER TABLE trans_payment_invoice OWNER TO cv_dallas;
```

q. Table *production of sales*

```
-- Table: trans_production_of_sales

-- DROP TABLE trans_production_of_sales;

CREATE TABLE trans_production_of_sales(
    id integer NOT NULL, created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    production_date date NOT NULL,
    used boolean NOT NULL, employee_id integer,
    CONSTRAINT trans_production_of_sales_pkey PRIMARY KEY (id),
    CONSTRAINT fkka07o9b6j7va8i71uwdmho2u4 FOREIGN KEY
    (employee_id)
        REFERENCES mst_employees (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
)WITH (OIDS=FALSE);
ALTER TABLE trans_production_of_sales OWNER TO cv_dallas;
```

r. Tabel *production of sales details*

```
-- Table: trans_production_of_sales_details

-- DROP TABLE trans_production_of_sales_details;

CREATE TABLE trans_production_of_sales_details(
    id integer NOT NULL, item_count integer NOT NULL,
    item_id integer, production_id integer,
    CONSTRAINT trans_production_of_sales_details_pkey PRIMARY KEY
(id), CONSTRAINT fk6pfkbwuemmre8env0uhuqtx72 FOREIGN KEY
(production_id)
    REFERENCES trans_production_of_sales (id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT fko94kw33c05cymehpb3k7ghreo FOREIGN KEY (item_id)
        REFERENCES mst_items (item_generator) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
) WITH (OIDS=FALSE);
ALTER TABLE trans_production_of_sales_details OWNER TO
cv_dallas;
```

s. Tabel *purchase invoice*

```
-- Table: trans_purchase_invoice

-- DROP TABLE trans_purchase_invoice;

CREATE TABLE trans_purchase_invoice(
    id integer NOT NULL, created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    amount double precision NOT NULL, description text,
    grant_total double precision NOT NULL,
    trans_date date NOT NULL, suplayer_id integer,
    CONSTRAINT trans_purchase_invoice_pkey PRIMARY KEY (id),
    CONSTRAINT fkptossp746273bd5i61ae3qyp4 FOREIGN KEY
(suplayer_id)
    REFERENCES mst_suplayers (suplayer_id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)WITH (OIDS=FALSE);
ALTER TABLE trans_purchase_invoice OWNER TO cv_dallas;
```

t. Tabel *purchase invoice details*

```
-- Table: trans_purchase_invoice_details

-- DROP TABLE trans_purchase_invoice_details;

CREATE TABLE trans_purchase_invoice_details(
    id integer NOT NULL, price_buy double precision NOT NULL,
    qty integer NOT NULL, purchase_id integer, item_id integer,
    CONSTRAINT trans_purchase_invoice_details_pkey PRIMARY KEY
(id),
    CONSTRAINT fkfnsh8d36dblx1xx4x707h16yy7 FOREIGN KEY
(purchase_id)
        REFERENCES trans_purchase_invoice (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT fkoypq68mehqtfte2ognchnokfs FOREIGN KEY (item_id)
        REFERENCES mst_items (item_generator) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
)WITH (OIDS=FALSE);
ALTER TABLE trans_purchase_invoice_details OWNER TO cv_dallas;
```

u. Tabel *sales*

```
-- Table: trans_sales

-- DROP TABLE trans_sales;

CREATE TABLE trans_sales(
    id integer NOT NULL, created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    ammount double precision NOT NULL,
    transaction_date date NOT NULL,
    grant_total double precision NOT NULL,
    transaction_month character varying(255) NOT NULL,
    is_paid boolean NOT NULL, is_recieved boolean NOT NULL,
    is_sent boolean NOT NULL, ship_to text NOT NULL,
    trans_id character varying(255) NOT NULL,
    transaction_year integer NOT NULL, customer_id integer,
    CONSTRAINT trans_sales_pkey PRIMARY KEY (id),
    CONSTRAINT fk4daxwlkvaluxqxf8j86fkdeug FOREIGN KEY
(customer_id) REFERENCES mst_customers (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT uk_sw37fqqqvmyocs4rmwe7r0xbg UNIQUE (trans_id)
)WITH (OIDS=FALSE); ALTER TABLE trans_sales OWNER TO cv_dallas;
```

v. Tabel sales details

```
-- Table: trans_sales_details

-- DROP TABLE trans_sales_details;

CREATE TABLE trans_sales_details(
    id integer NOT NULL, sell_price double precision NOT NULL,
    sell_qty integer NOT NULL, item_id integer,
    sales_id integer,
    CONSTRAINT trans_sales_details_pkey PRIMARY KEY (id),
    CONSTRAINT fk179aqyoivubx7fjh0w4f1n0n8 FOREIGN KEY (sales_id)
        REFERENCES trans_sales (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT fksasli6uuhr900sk2juqo5o71r FOREIGN KEY (item_id)
        REFERENCES mst_items (item_generator) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
)WITH (OIDS=FALSE); ALTER TABLE trans_sales_details OWNER TO
cv_dallas;
```

w. Tabel *sales order*

```
-- Table: trans_sales_order

-- DROP TABLE trans_sales_order;

CREATE TABLE trans_sales_order(
    id integer NOT NULL, created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    todo_checklist boolean NOT NULL,
    down_payment double precision NOT NULL,
    order_date date NOT NULL, trans_id character varying(255) NOT
NULL, customer_id integer, CONSTRAINT trans_sales_order_pkey
PRIMARY KEY (id), CONSTRAINT fk10ua81jfp54uql6gpmt6rvvue
FOREIGN KEY (customer_id)
    REFERENCES mst_customers (id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT uk_gfqn3nmk57e6htaalfenf55y1 UNIQUE (trans_id)
)WITH ( OIDS=FALSE);
ALTER TABLE trans_sales_order OWNER TO cv_dallas;
```

x. Tabel *sales order details*

```
-- Table: trans_sales_order_details

-- DROP TABLE trans_sales_order_details;

CREATE TABLE trans_sales_order_details(
    id integer NOT NULL, qty integer NOT NULL, item_id integer,
    order_id integer,
    CONSTRAINT trans_sales_order_details_pkey PRIMARY KEY (id),
    CONSTRAINT fk9cbps8wpln3n8bsaupxsa4crk FOREIGN KEY (order_id)
        REFERENCES trans_sales_order (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT fknriirq9xq8pu1h5su80sjgtach FOREIGN KEY (item_id)
        REFERENCES mst_items (item_generator) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
)WITH (OIDS=FALSE);
ALTER TABLE trans_sales_order_details OWNER TO cv_dallas;
```

y. Tabel *sales delivery*

```
-- Table: trans_sales_delivery

-- DROP TABLE trans_sales_delivery;

CREATE TABLE trans_sales_delivery(
    id integer NOT NULL, created_by character varying(255),
    created_date timestamp without time zone,
    last_updated_date timestamp without time zone,
    date_sent date NOT NULL, delivery_id character varying(255)
NOT NULL, grant_total double precision NOT NULL,
    in_process boolean NOT NULL, employee_id integer,
    CONSTRAINT trans_sales_delivery_pkey PRIMARY KEY (id),
    CONSTRAINT fk4e8is794tf8olrrv3gnsfr6d6 FOREIGN KEY
(employee_id)
        REFERENCES mst_employees (id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT uk_aid7buccts dokikgky79sxbc8 UNIQUE (delivery_id)
)WITH (OIDS=FALSE);
ALTER TABLE trans_sales_delivery OWNER TO cv_dallas;
```

z. Tabel *sales delivery details*

```
-- Table: trans_sales_delivery_details

-- DROP TABLE trans_sales_delivery_details;

CREATE TABLE trans_sales_delivery_details
(
    id integer NOT NULL, wight_per_kg integer NOT NULL,
    delivery_id integer, sales_id integer,
    CONSTRAINT trans_sales_delivery_details_pkey PRIMARY KEY
(id), CONSTRAINT fkaxaa9ns2pgon3h1lwgcrbx4i9 FOREIGN KEY
(delivery_id)
    REFERENCES trans_sales_delivery (id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT fkuiijfcn4bonn7g5aqeup6nvq FOREIGN KEY (sales_id)
    REFERENCES trans_sales (id) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION
)WITH (OIDS=FALSE);
ALTER TABLE trans_sales_delivery_details OWNER TO cv_dallas;
```

4.5.4. Implementasi antar muka

Berdasarkan perancangan input dan output berikut ini adalah implementasi antar muka yang telah selesai dikonstruksi, sesuai dengan sequence diagram implementasi antar muka untuk teknologi javafx terdiri dari *controller*, *view*, *model* dan *service*.

Tabel 4.67 Design Pattern Dengan Javafx

Pattern	keterangan
Controller	<i>Pattern controller</i> adalah <i>java class</i> yang berisi <i>trigger</i> yang berkaitan dengan <i>view (fxml)</i> seperti <i>ActionEvent</i> , <i>Listener</i> dan lain-lain.
View	<i>Pattern view</i> adalah sekumpulan <i>graphic user interface</i> yang dibuat terpisah dari event-nya.
Model	<i>Pattern model</i> adalah <i>java class</i> yang berisikan atau merepresentasikan atribut-atribut pada tabel di dalam basis data atau merepresentasikan data.
Service	<i>Pattern service</i> adalah <i>java class</i> yang merepresentasikan proses bisnis.

Berikut ini adalah implementasi antar muka yang dibagi menjadi tiga yaitu implementasi antar muka data utama, implementasi antar muka data transaksi dan implementasi antar muka laporan-laporan:

4.5.4.1. Implementasi antar muka data utama

Data utama atau data master merupakan data yang nantinya akan digunakan untuk transaksi:

a. Data jabatan

Berikut ini adalah implementasi antar muka input data utama untuk data jabatan yang berbentuk source code:

Tabel 4.68 Implementasi Antar Muka Input Data Jabatan

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\masterdata	DepartmentDataAction.java
View	src\main\resources\stage\scene\masterdata	department_data.fxml
Model	src\main\java\dallastools\models\masterdata	Department.java
Service	src\main\java\dallastools\services	ServiceOfDepartment.java

dan berikut ini adalah implemetasi antar muka untuk input data jabatan ketika dijalankan:



Gambar 4.67 Implementasi Antar Muka Input Data Jabatan

b. Data karyawan

Berikut ini adalah implementasi antar muka input data utama untuk data karyawan yang berbentuk source code:

Tabel 4.69 Implementasi Antar Muka Input Data Jabatan

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\masterdata	EmployeeDataAction.java
View	src\main\resources\stage\scene\masterdata	employee_data.fxml
Model	src\main\java\dallastools\models\masterdata	Employee.java
Service	src\main\java\dallastools\services	ServiceOfEmployee.java

dan berikut adalah implementasi antar muka untuk input data karyawan setelah dijalankan:

The screenshot shows a JavaFX application window titled "Data karyawan". The window has a light gray background and contains several input fields and dropdown menus. On the left side, there are labels: "Nama", "Pilih jabatan", "Kota", "Kelurahan", "Alamat lengkap", "RT", "RW", and "Kode Pos". To the right of these labels are corresponding input fields or dropdown menus. At the bottom of the window, there are two buttons: a blue "Simpan" button with a save icon and a gray "Batal" button with a cancel icon.

Gambar 4.68 Implementasi Antar Muka Input Data Karyawan

c. Data kategori barang

Berikut ini adalah implementasi antar muka input data utama untuk data kategori barang yang berbentuk source code:

Tabel 4.70 Implementasi Antar Muka Input Data Kategori Barang

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\masterdata	CategoryOfItemDataAction.java
View	src\main\resources\stage\screens\masterdata	Category_of_item_data.fxml
Model	src\main\java\dallastools\models\masterdata	CategoryOfItem.java
Service	src\main\java\dallastools\services	ServiceOfItemCategory.java

dan berikut ini adalah implementasi antar muka untuk input data kategori barang ketika dijalankan:

Gambar 4.69 Implementasi Antar Muka Input Data Kategori Barang

d. Data satuan barang

Berikut ini adalah implementasi antar muka input data utama untuk kategori barang yang berbentuk source code:

Tabel 4.71 Implementasi Antar Muka Input Data Satuan Barang

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\masterdata	UnitDataAction.java
View	src\main\resources\stage\scene\masterdata	unit_of_item_data.fxml
Model	src\main\java\dallastools\models\masterdata	Unit.java
Service	src\main\java\dallastools\services	ServiceOfUnit.java

dan berikut ini adalah implementasi antar muka input data untuk satuan barang ketika dijalankan:

Gambar 4.70 Implementasi Antar Muka Input Data Satuan Barang

e. Data gudang

Berikut ini adalah implementasi antar muka input data utama untuk gudang yang berbentuk source code:

Tabel 4.72 Implementasi Antar Muka Input Data Gudang

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\masterdata	WarehouseDataAction.java
View	src\main\resources\stage\scene\masterdata	warehouse_data.fxml
Model	src\main\java\dallastools\models\masterdata	Warehouse.java
Service	src\main\java\dallastools\services	ServiceOfWarehouse.java

Dan berikut adalah implementasi antar muka input data untuk gudang ketika dijalankan:

Gambar 4.71 Implementasi Antar Muka Untuk Input Data Gudang

f. Data barang

Berikut ini adalah implementasi antar muka input data utama untuk barang-barang yang berbentuk source code:

Tabel 4.73 Implementasi Antar Muka Input Data Barang

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\masterdata	ItemDataAction.java
View	src\main\resources\stage\scene\masterdata	item_data.fxml
Model	src\main\java\dallastools\models\masterdata	Item.java
Service	src\main\java\dallastools\services	ServiceOfItem.java

dan berikut adalah implementasi antar muka input data untuk barang ketika dijalankan:

The screenshot shows a JavaFX application window titled "Data barang". The window has a light gray background and a blue header bar. Inside, there are several input fields and a radio button group. The fields include "Nama" (with placeholder "Kode Barang"), "Kategori", "Satuan", and "Lokasi barang", each with dropdown menus. Below these are two numeric input fields for "Harga jual" and "Harga beli", both set to 0. At the bottom left is a radio button group for "Dijual" with options "Tidak untuk d..." and "Dijual", where "Tidak untuk d..." is selected. At the bottom right are two buttons: "Simpan" with a disk icon and "Batal" with a cancel icon.

Gambar 4.72 Implementasi Antar Muka Input Data Barang

g. Data konsumen

Berikut ini adalah implementasi antar muka input data utama untuk konsumen yang berbentuk source code:

Tabel 4.74 Implementasi Antar Muka Input Data Konsumen

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\mas terdata	CustomerDataAction.java
View	src\main\resources\stage\scene\maste rdata	customer_data.fxml
Model	src\main\java\dallastools\models\mas terdata	Customer.java
Service	src\main\java\dallastools\services	ServiceOfCustomer.java

dan berikut ini adalah implementasi antar muka input data untuk konsumen ketika dijalankan:

The screenshot shows a Java Swing application window titled "Data konsumen". The window has a light gray background and contains several input fields and buttons. At the top left is a diamond icon. The fields are labeled as follows: "Nama" (Name), "Kontak Personal" (Personal Contact), "Kota" (City), "Kelurahan" (Sub-District), "Alamat lengkap" (Full Address), "Kode Pos" (Postcode), "RW", and "RT". Each label is followed by a text input field. Below these fields are three numeric spinners with up and down arrows. At the bottom of the window are two buttons: a blue "Simpan" (Save) button on the left and a gray "Batal" (Cancel) button on the right.

Gambar 4.73 Implementasi Antar Muka Input Data Konsumen

h. Data pemasok

Berikut ini adalah implementasi antar muka input data utama untuk pemasok yang berbentuk source code:

Tabel 4.75 Implementasi Antar Muka Input Data Pemasok

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\mas terdata	SuplayerDataAction.java
View	src\main\resources\stage\scene\maste rdata	suplayer_data.fxml
Model	src\main\java\dallastools\models\mas terdata	Suplayer.java
Service	src\main\java\dallastools\services	ServiceOfSuplayer.java

dan berikut ini adalah implementasi antar muka input data untuk suplayer ketika dijalankan:

The screenshot shows a Java Swing application window titled "Data pemasok". The interface includes the following fields:

- Nama:** A text input field labeled "Nama pemasok".
- Kontak Personal:** A text input field labeled "Kontak personal".
- Kota:** A text input field labeled "Kota".
- Kelurahan:** A text input field labeled "Kelurahan".
- RT / RW:** Two dropdown menus for selecting RT and RW numbers.
- Kode Pos:** Two dropdown menus for selecting Kode Pos.
- Alamat lengkap:** A large text area for entering full address.

At the bottom of the window are two buttons: "Simpan" (Save) and "Batal" (Cancel).

Gambar 4.74 Implemetasi Antar Muka Input Data Pemasok

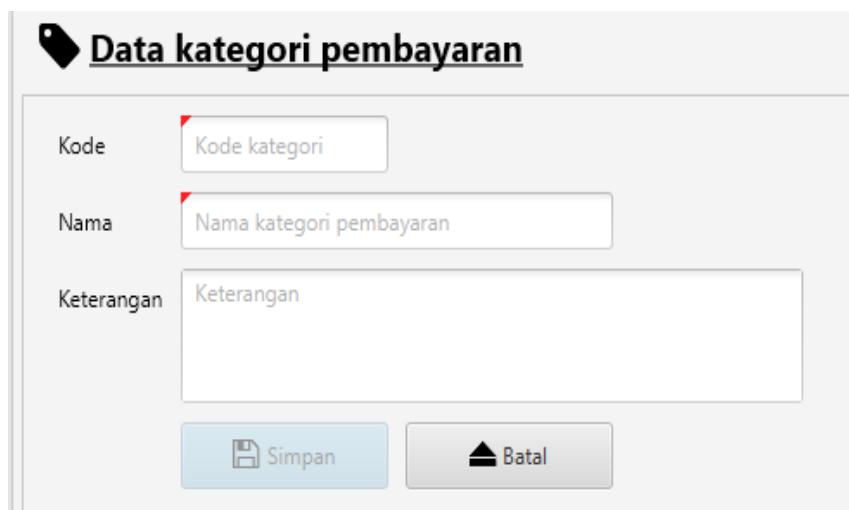
i. Data kategori pembayaran

Berikut ini adalah implementasi antar muka input data utama untuk kategori pembayaran yang berbentuk source code:

Tabel 4.76 Implementasi Antar Muka Input Data Kategori Pembayaran

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\masterdata	SuplayerDataAction.java
View	src\main\resources\stage\scene\masterdata	suplayer_data.fxml
Model	src\main\java\dallastools\models\masterdata	Suplayer.java
Service	src\main\java\dallastools\services	ServiceOfSuplayer.java

dan berikut ini adalah implementasi antar muka input data untuk kategori pembayaran ketika dijalankan:



Gambar 4.75 Implementasi Antar Muka Input Data Kategori Pembayaran

4.5.4.2. Implementasi antar muka data transaksi

Implementasi antar muka data transaksi adalah data yang berkaitan dengan transaksi pencatatan pemasukan dan pengeluaran. Data yang digunakan berasal dari data master data seperti barang, *customer*, *suplayer* dan lain-lain. Berikut ini adalah implementasi antar muka data transaksi:

- a. Data produksi barang

Berikut ini adalah implementasi antar muka input data untuk produksi yang berbentuk *source code*:

Tabel 4.77 Implementasi Antar Muka Produksi Barang

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\productions	ProductionSalesDataAction.java
View	src\main\resources\stage\scene\transactions\productions	production_of_sales_data.fxml
Model	src\main\java\dallastools\models\productions	ProductionOfSales.java
Service	src\main\java\dallastools\services	ServiceOfProduction.java

dan berikut ini adalah implementasi antar muka untuk input data produksi ketika dijalankan:

The screenshot shows a user interface for entering production data. At the top, there are two input fields: 'Nama' with a dropdown menu showing 'Pilih Karyawan' and 'Tanggal' with a date input field showing '22/12/2015'. Below these is a table titled 'Spesifikasi Barang' with the following columns: Kode, Nama, Nama gudang, Kategori barang, Satuan barang, and Jumlah. A message 'No content in table' is centered in the table area. At the bottom of the form are two buttons: 'Batal' (Cancel) and 'Simpan' (Save).

Gambar 4.76 Implementasi Antar Muka Input Data Produksi

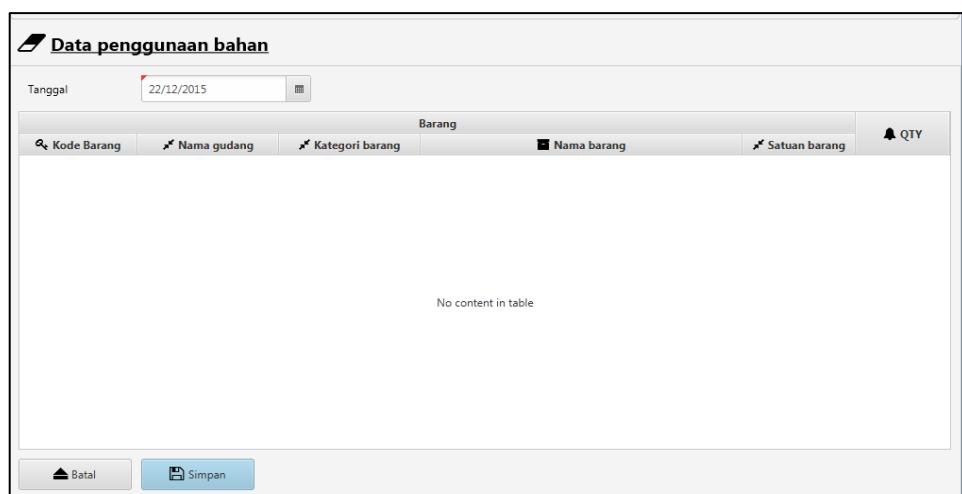
b. Data penggunaan barang

Berikut ini adalah implementasi antar muka input data untuk penggunaan bahan yang berbentuk *source code*:

Tabel 4.78 Implementasi Antar Muka Input Data Penggunaan Bahan

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\productions	ItemUsedDataAction.java
View	src\main\resources\stage\scene\transactions\productions	item_used_data.fxml
Model	src\main\java\dallastools\models\productions	ItemUsed.java
Service	src\main\java\dallastools\services	ServiceOfItemUsed.java

dan berikut ini adalah implementasi antar muka input data untuk penggunaan bahan ketika dijalankan



Gambar 4.77 Implementasi Antar Muka Input Data Penggunaan Barang

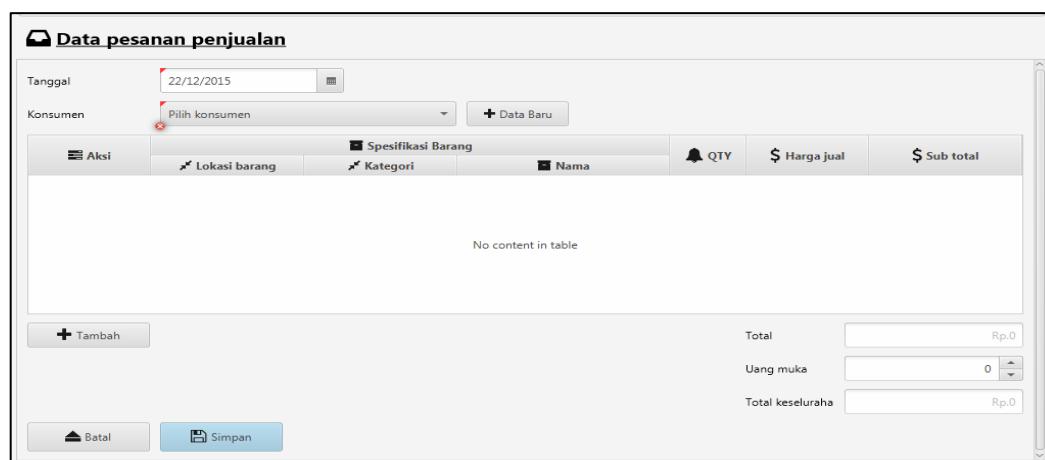
c. Data pemesanan penjualan

Berikut ini adalah implementasi antar muka input data untuk pesanan penjualan yang berbentuk *source code*:

Tabel 4.79 Implementasi Antar Muka Input Data Pesanan Penjualan

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\income	SalesOrderDataAction.java
View	src\main\resources\stage\scene\transactions\incomes	sales_order_data.fxml
Model	src\main\java\dallastools\models\income	SalesOrder.java
Service	src\main\java\dallastools\services	ServiceOfSalesOrder.java

dan berikut adalah implementasi antar muka input data penjualan ketika dijalankan:



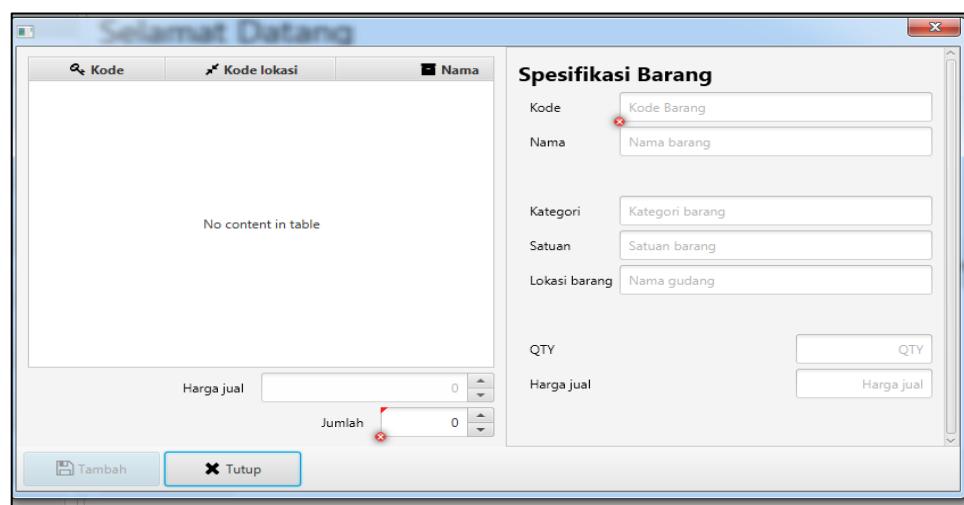
Gambar 4.78 Implementasi Antar Muka Input Data Pemesanan Penjualan

ketika melakukan klik *button tambah barang* maka akan tampil dialog dari *source code* berikut:

Tabel 4.80 Implementasi Antar Muka Pilih Barang Pesanan Penjualan

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\income	AddItemToSalesOrderAction.java
View	src\main\resources\stage\scene\transactions\incomes	add_an_item_to_sales_order.fxml
Model	src\main\java\dallastools\models\income	SalesOrderDetails.java
Service	src\main\java\dallastools\services	ServiceOfItem.java

dan berikut ini adalah tampilan dialog atau implementasi antar muka pilih barang ketika dijalankan:



Gambar 4.79 Implementasi Antar Muka Pilih Barang Pesanan Penjualan

d. Data penjualan

Berikut ini adalah implementasi antar muka input data untuk penjualan yang berbentuk *source code*:

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\income	SalesDataAction.java
View	src\main\resources\stage\scene\transactions\incomes	sales_data.fxml
Model	src\main\java\dallastools\models\income	Sales.java
Service	src\main\java\dallastools\services	ServiceOfSales.java

dan berikut ini adalah implementasi antar muka input data untuk penjualan ketika dijalankan:

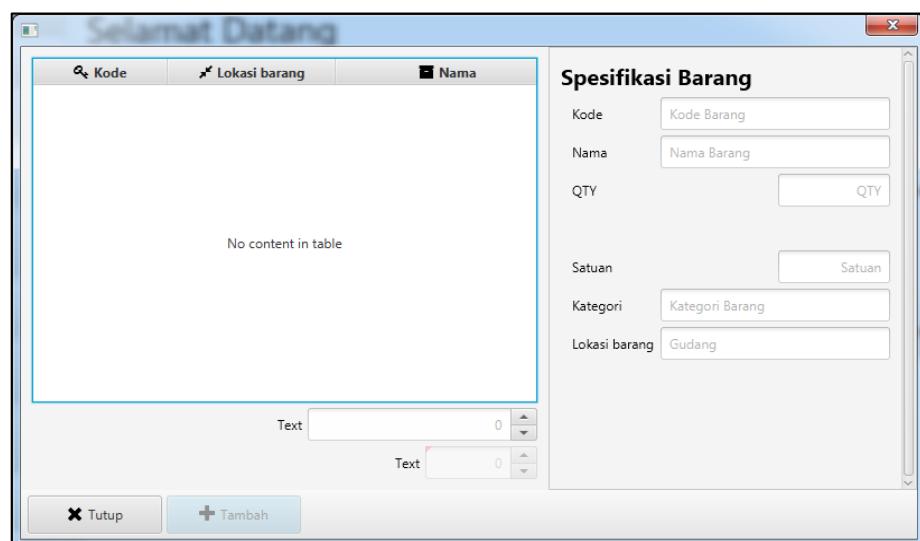
Gambar 4.80 Implementasi Antar Muka Input Data Penjualan

ketika melakukan klik *button tambah barang* maka akan tampil dialog dari *source code* berikut:

Tabel 4.81 Implementasi Antar Muka Pilih Data Barang Penjualan

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\income	AddItemToSalesInvoiceAction.java
View	src\main\resources\stage\scene\transactions\incomes	add_an_item_to_sales_invoice.fxml
Model	src\main\java\dallastools\models\income	SalesDetails.java
Service	src\main\java\dallastools\services	ServiceOfItem.java

dan berikut ini adalah implementasi antar muka pilih barang untuk penjualan ketika dijalankan:



Gambar 4.81 Implementasi Antar Muka Pilih Data Barang Penjualan

e. Data biaya transportasi

Berikut ini adalah implementasi antar muka input data untuk pesanan penjualan yang berbentuk *source code*:

Tabel 4.82 Implementasi Antar Muka Input Data Biaya Transaportasi

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\expeditur	SalesDeliveryDataAction.java
View	src\main\resources\stage\scene\transactions\expenditur	delivery_of_sales_data.fxml
Model	src\main\java\dallastools\models\expenditur	DeliveryOfSales.java
Service	src\main\java\dallastools\services	ServiceOfSalesDelivery.java

dan berikut ini adalah implementasi antar muka input data biaya transportasi ketika dijalankan:



Gambar 4.82 Implementasi Antar Muka Input Data Biaya Transportasi

f. Data pembelian bahan baku

Berikut ini adalah implementasi antar muka input data untuk pembelian bahan baku yang berbentuk *source code*:

Tabel 4.83 Implementasi Antar Muka Input Data Pembelian Bahan

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\expeditur	PurchaseInvoiceDataAction.java
View	src\main\resources\stage\sce ne\transactions\expeditur	purchase_invoice_data.fxml
Model	src\main\java\dallastools\mo dels\expeditur	PurchaseInvoice.java
Service	src\main\java\dallastools\ser vices	ServiceOfPurchaseInvoice.java

berikut ini adalah implementasi antar muka input data pembelian ketika dijalankan:

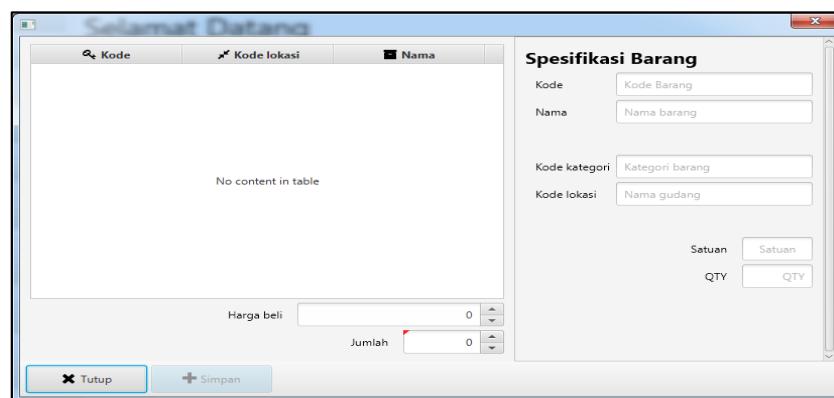
Gambar 4.83 Implementasi Atar Muka Input Data Pembelian

sama halnya seperti penjualan ketika melakukan klik *button* tambah barang akan mempilkan dialog berikut ini adalah *source code* untuk menampilkan dialog tersebut:

Tabel 4.84 Implementasi Antar Muka Pilih Barang Pembelian

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallasto ols\actions\expeditur	AddItemToPurchaseInvoiceAction.java
View	src\main\resources\sta ge\scene\transactions\ expeditur	add_an_item_to_purchase_invoice.fxml
Model	src\main\java\dallasto ols\models\expeditur	PurchaseInvoiceDetails.java
Service	src\main\java\dallasto ols\services	ServiceOfItem.java

dan berikut adalah implementasi antar muka pilih barang untuk pembelian ketika dijalankan:



Gambar 4.84 Implementasi Antar Muka Pilih Bahan

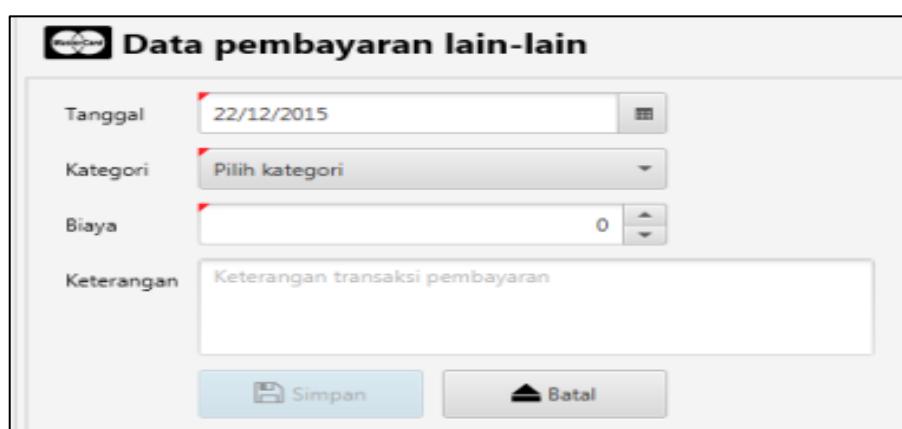
g. Data pembayaran lain-lain

Berikut ini adalah implementasi antar muka input data untuk pembayaran lain-lain yang berbentuk *source code*:

Tabel 4.85 Implementasi Antar Muka Input Pembayaran Lain-Lain

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\expeditur	PaymentInvoiceDataAction.java
View	src\main\resources\stage\scene\transactions\expeditur	payment_invoice_data.fxml
Model	src\main\java\dallastools\models\expeditur	PaymentInvoice.java
Service	src\main\java\dallastools\services	ServiceOfPaymentInvoice.java

dan berikut ini adalah implementasi antar muka input data pembayaran lain-lain ketika dijalankan:



Gambar 4.85 Implementasi Antar Muka Input Data Pembayaran

h. Data penggajian

Berikut ini adalah implementasi antar muka input data untuk penggajian karyawan yang berbentuk *source code*:

Tabel 4.86 Implementasi Antar Muka Input Data Penggajian

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\expeditur	EmployeePayrollDataAction.java
View	src\main\resources\stage\scene\transactions\expeditur	employee_payroll_data.fxml
Model	src\main\java\dallastools\models\expeditur	PayrollAnEmployee.java
Service	src\main\java\dallastools\services	ServiceOfPayrollEmployee.java

dan berikut adalah implementasi antar muka input data penggajian karyawan bagian produksi ketika dijalankan:

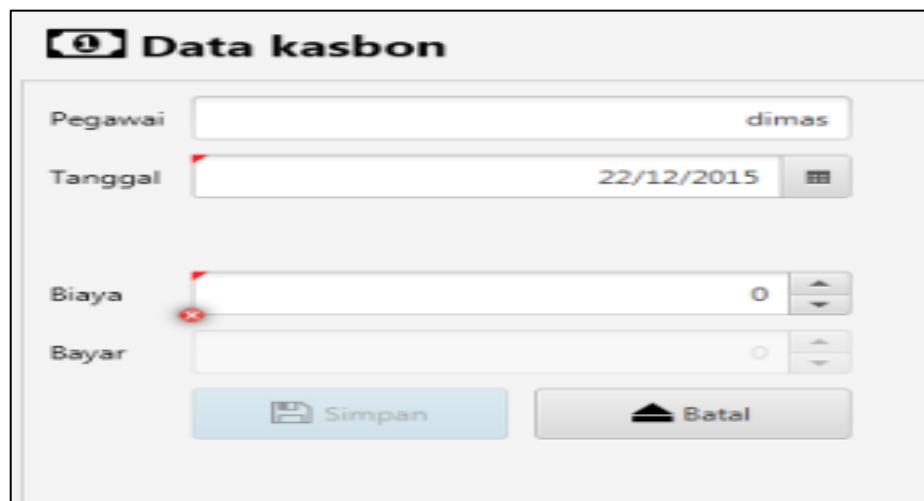
Gambar 4.86 Implementasi Antar Muka Input Data Penggajian

i. Data kasbon karyawan

Berikut ini adalah implementasi antar muka input data untuk kasbon karyawan yang berbentuk *source code*:

Pattern	Lokasi file	Nama file
Controller	src\main\java\dallastools\actions\expeditur	EmployeeCashReceiptAction.java
View	src\main\resources\stage\scene\transactions\expeditur	employee_cash_receipt_data.fxml
Model	src\main\java\dallastools\models\expeditur	CashReceiptForEmployee.java
Service	src\main\java\dallastools\services	ServiceOfCashReceipt.java

Dan berikut adalah implementasi antar muka input data kasbon karyawan ketika dijalankan:



Gambar 4.87 Implemetasi Antar Muka Input Data Kasbon

4.5.4.3. Implementasi antar muka laporan-laporan

Berdasarkan perancangan antar muka untuk laporan-laporan berikut ini adalah implementasi antar muka laporan yang terdiri dari laporan surat keterangan produksi, kwitansi penjualan, dan surat jalan pengiriman.

1. Laporan keterangan produksi.

Dallas Ice Cream		SALES_ORD-2-20151226-3	
		Toko / Tuan : dimas maryanto	
		alamat : jl.bukit indah no b8	kab.bandung kec.cileunyi
		rt.6 rw.8	
Tanggal Pesan : 2015-12-26		Waktu Cetak : Des 27, 2015 1:08:15 PM	
No.	Nama Barang	QTY	Keterangan
1	Hanco Strawberry	1	
2	Stick	2	

Gambar 4.88 Implementasi Antar Muka Pesanan Penjualan

DALLAS ICE CREAM		SALES_INV-2-20151227-6	
		Toko / Tuan : dimas maryanto	Ship to : jl.bukit indah no b8
		kab.bandung kec.cileunyi rt.6 rw.8	
waktu cetak : Des 27, 2015 1:11:29 PM			
No.	Nama Barang	QTY	Harga Satuan
1	Hanco Strawberry	10	Rp62.500
2	Cup Besar 100ml	10	Rp1.000
3	Galon	2	Rp75.000
Jumlah		Rp 625.000	
Uang Muka		Rp0	
Sisa		Rp 785.000	

Gambar 4.89 Implementasi Antar Muka Kwitansi Penjualan

3. Laporan surat jalan untuk pengiriman barang untuk penjualan.

Dallas Ice Cream jl cendana no.12 Lembang SALES DELIVERY		SALES_DEL-20151227-5 Supir: hari hermawati	
Waktu Cetak : 27 Des 2015, 13:12 PM			
Kode Transaksi	Berat/ Kg	Alamat	Paraf
SALES_INV-2-20151227-6	0	j.bukit indah no 68	

Gambar 4.90 Implementasi Antar Muka Surat Pengiriman Penjualan

4. Laporan surat keterangan produksi.

Dallas Ice Cream			
Surat Ket Prouksi		waktu cetak: Jan 2, 2016 7:32:57 AM	
No.	Nama Barang	Satuan	Jumlah
1	Davacho	pcs	5
2	Vanies	pcs	15

Gambar 4.91 Implementasi Antar Muka Surat Keterangan Produksi

5. Laporan keuangan.

Dallas Ice Cream			
jl. cempaka no.12 Lembang			
waktu cetak: Jan 2, 2016 7:35:00 AM			
Laporan Keuangan			
periode: 2015-Jan-01 s/d 2016-Jan-31			
Pendapatan		Pendapatan Bersih	
Total Penjualan	Rp17.500	Total Pendapatan	Rp17.500
Piutang Penjualan	Rp0	Total Beban Operasional	Rp-0
Total Pendapatan	Rp17.500	Total Beban lain-lain	Rp-0
		Pendapatan bersih	Rp17.500
Beban Operasional			
Transpotasi	Rp0		
Produksi	Rp0		
Pembelian	Rp0		
Pembayaran lain-lain	Rp0		
Total Beban Operasional	Rp0		
Beban lain-lain			
kasbon	Rp0		
Total Beban lain-lain	Rp0		

Gambar 4.92 Implementasi Antar Muka Laporan Keuangan

4.5.5. Implementasi instalasi program

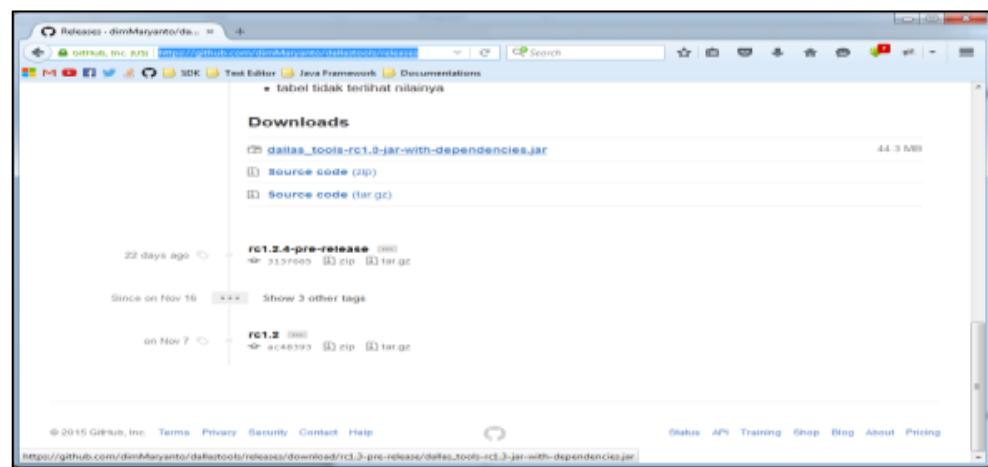
Aplikasi ini dirancang untuk multiplatform sistem operasi maka aplikasi ini tidak perlu melakukan installasi cukup lakukan langkah berikut ini:

1. Mengunduh *binary* dallasToolsApp di alamat berikut ini
<https://github.com/dimMaryanto/dallastools/releases>



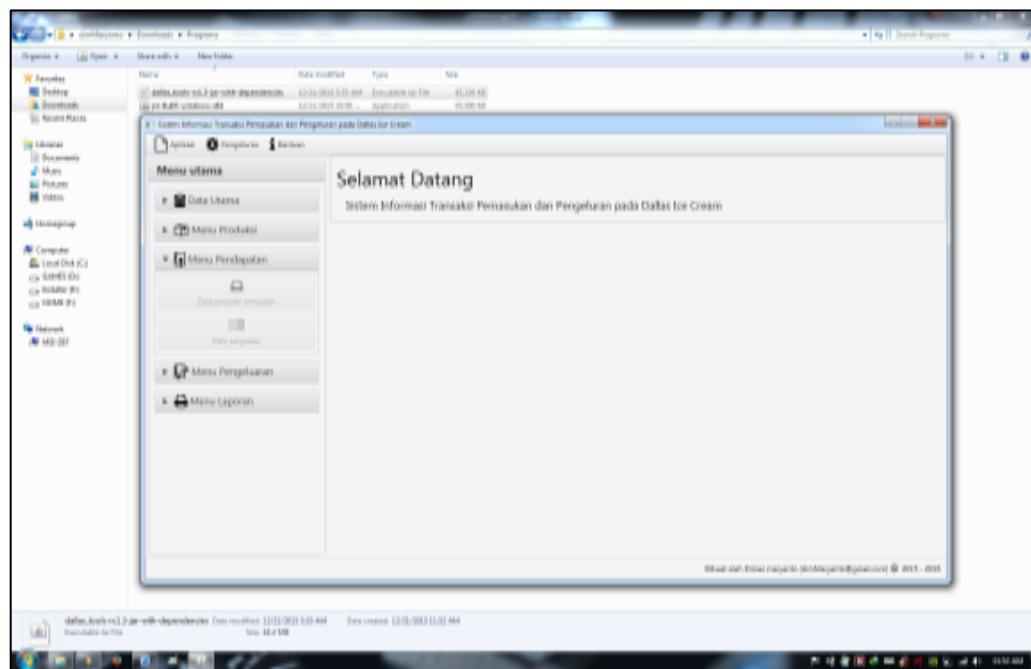
Gambar 4.93 Halaman Download Aplikasi

2. Pilih versi yang terbaru karena akan ada update setiap bulannya



Gambar 4.94 Mengunduh Aplikasi

3. Setelah selesai diunduh kemudian jalankan dallastools-rc1.3-with-dependency.jar maka akan tampil form sebagai berikut:



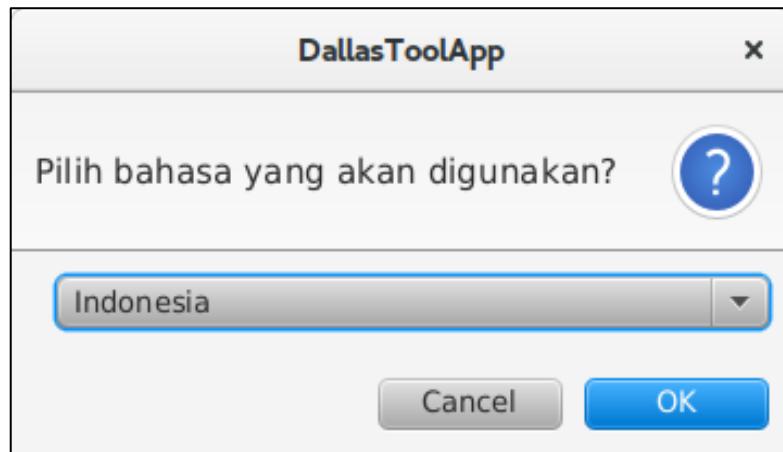
Gambar 4.95 Menjalankan Aplikasi dallas_tools-rcx.x

4.5.6. Penggunaan program

Berikut ini adalah cara penggunaan program untuk aplikasi DallasToolsApp yang dibagi menjadi beberapa bagian yaitu sebagai berikut:

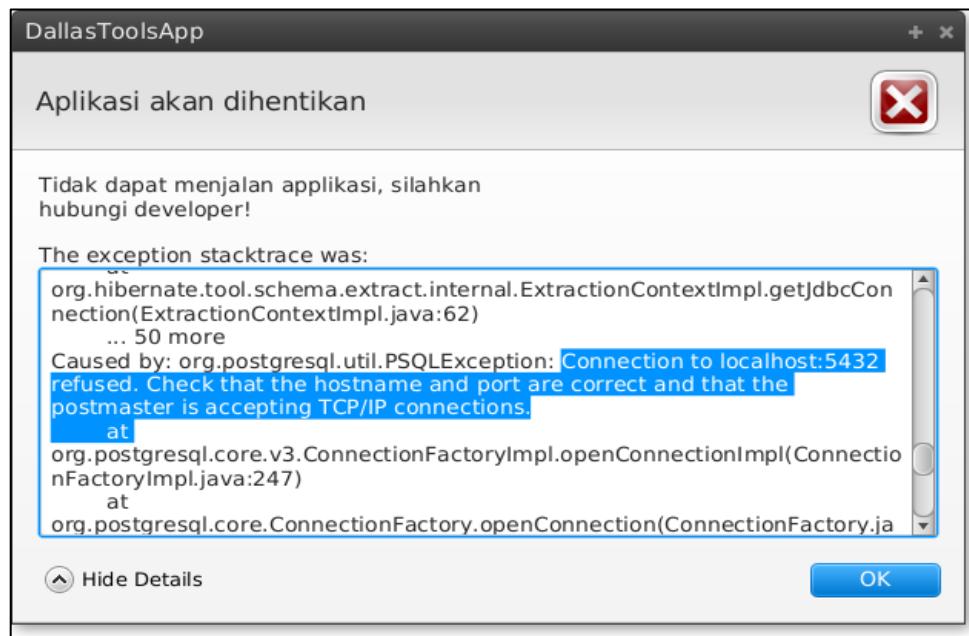
4.5.6.1. Penggunaan program menjalankan aplikasi

Pada saat pertama kali menjalankan aplikasi, akan muncul *dialog* untuk memilih bahasa, bahasa yang tersedia ada bahasa Indonesia dan bahasa ingris seperti gambar berikut ini:



Gambar 4.96 Penggunaan Program Pilih Bahasa

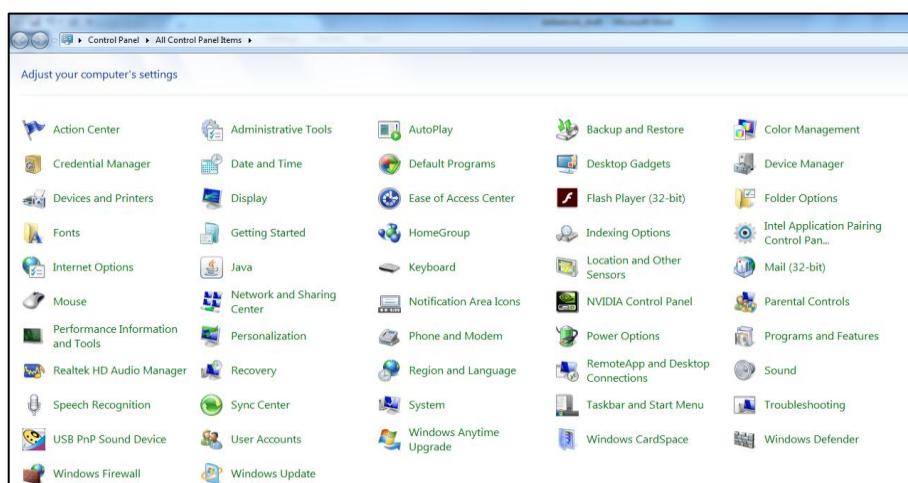
decara *default* bahasa yang diseleksi adalah bahasa Indonesia, setelah itu klik ok untuk melanjutkan menjalankan aplikasi. Setelah itu jika aplikasi ini baru pertama kali dijalankan pada komputer tersebut akan membutuhkan waktu beberapa menit untuk menampilkan halama utama karena aplikasi membutuhkan proses untuk mengsingkronisasi data dengan basis data PostgreSQL. Jika menemukan pesan sebagai berikut:



Gambar 4.97 Penggunaan Program Tidak Dapat Menjalankan Aplikasi

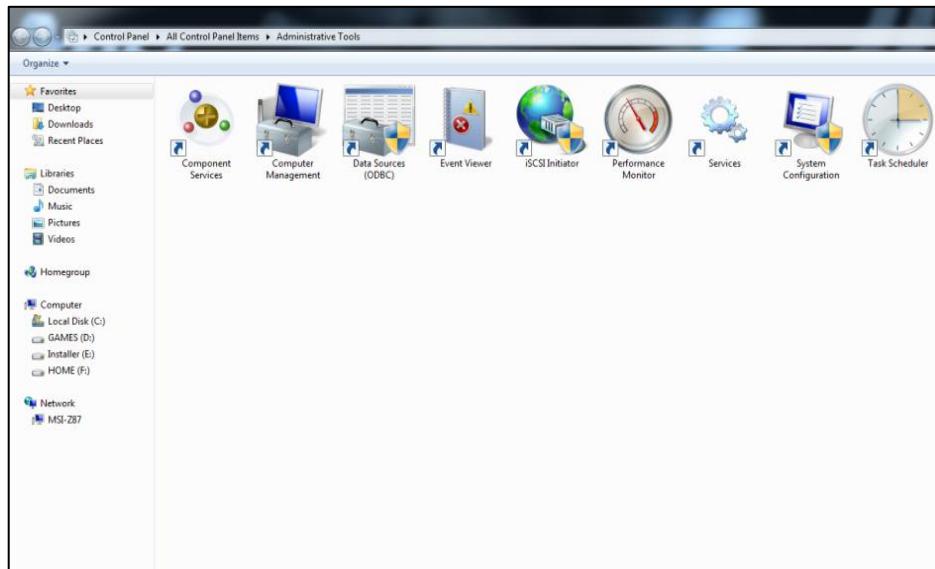
solusinya adalah periksa apakah *service postgresql* telah berjalan. Berikut ini adalah langkah untuk menjalankan service di sistem operasi windows:

1. Klik *Start menu* → pilih *Control Panel* maka akan mucul window seperti berikut:



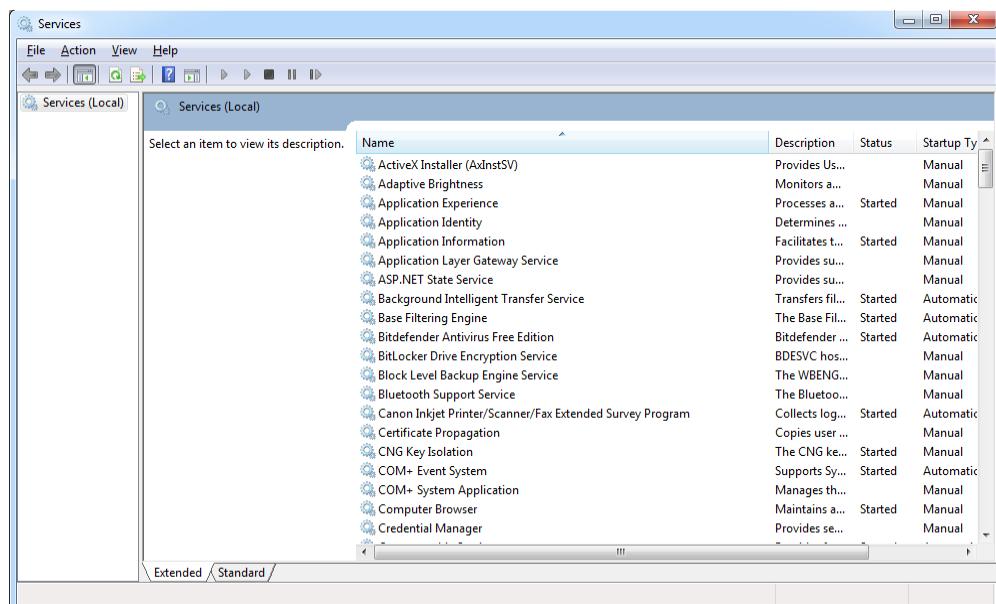
Gambar 4.98 Menu Control Panel

2. Klik *Administrative Tools* maka akan muncul window seperti berikut:



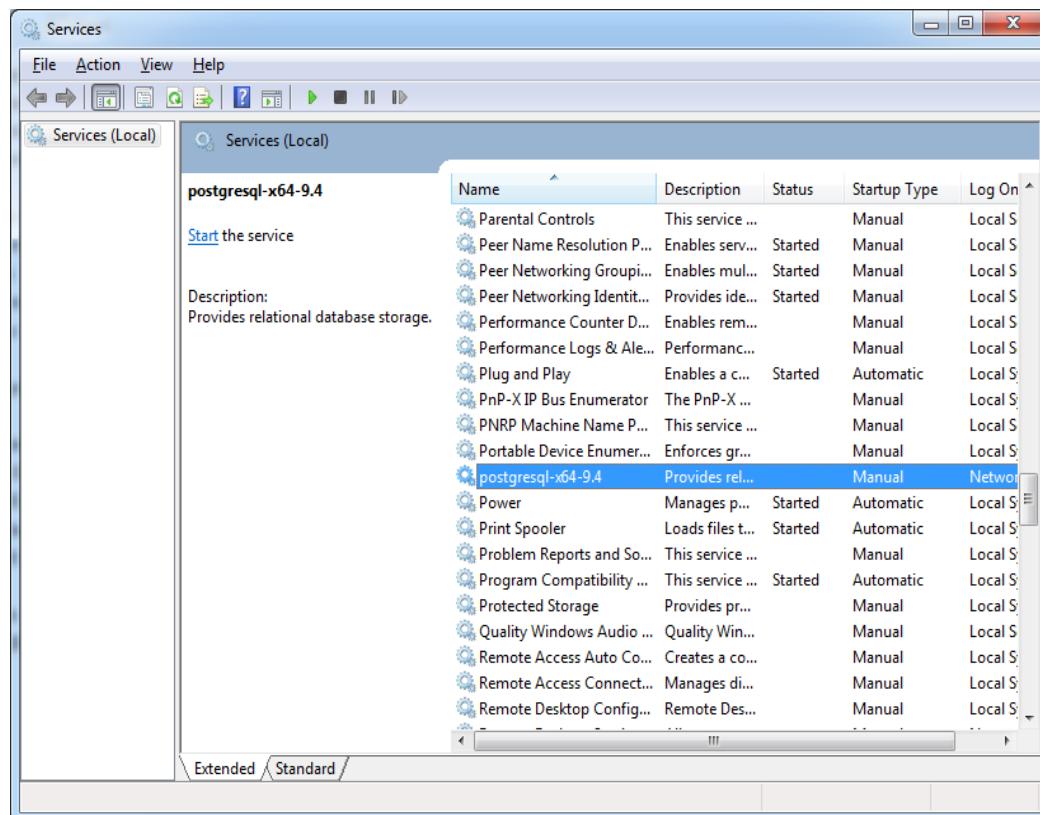
Gambar 4.99 Menu *Administrative Tools*

3. Kemudian double klik icon service, maka akan muncul dialog sebagai berikut:



Gambar 4.100 Dialog Service Pada Window

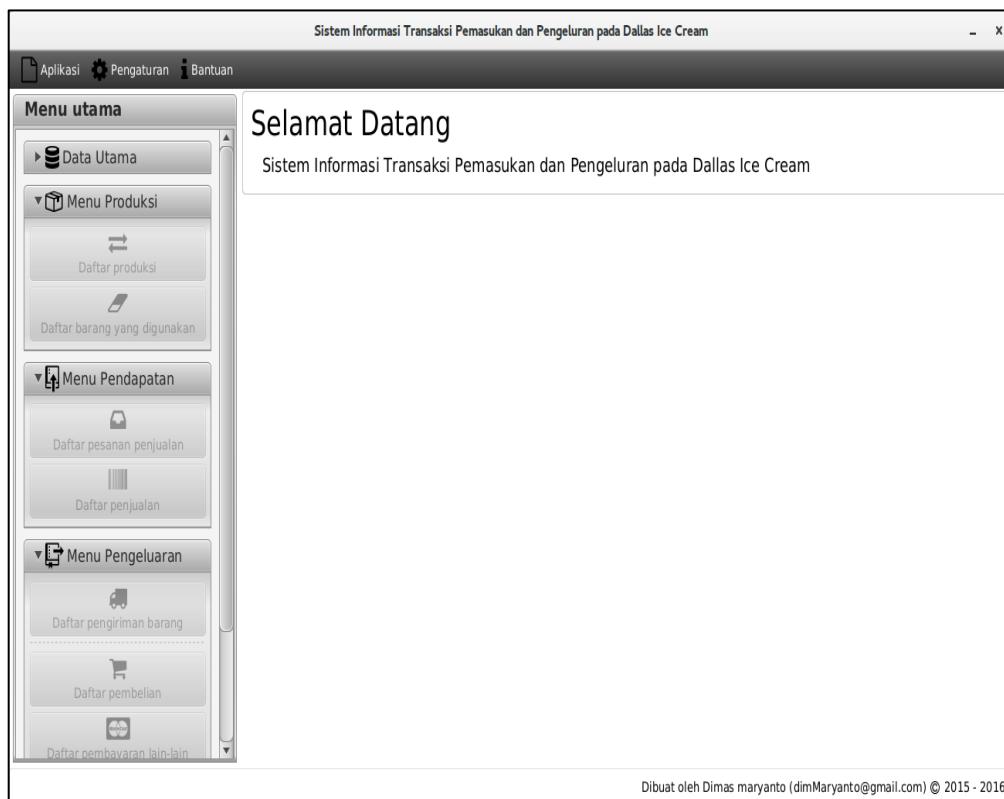
4. Kemudian cari berdasarkan name postgresql-x64-9.4 seperti gambar berikut:



Gambar 4.101 Pilih postgresql-x64-9.4 Pada Dialog Service

5. Kemudian klik kanan pada *service* dengan nama postgresql-x64-9.4 kemudian pilih start. Setelah itu jalankan kembali aplikasi DallasToolsApp.

Setelah menjalankan *service postgresql* dan menjalankan ulang aplikasi maka akan muncul menu utama aplikasi sebagai berikut:



Gambar 4.102 Menu Utama Dallastoolsapp Terkunci

diketahui bahwa menu-menu di menu utama tersebut dalam keadaan terkunci, untuk mengakses menu tersebut memerlukan *authentication* yang sesuai dengan hak aksesnya. Berikut ini adalah perbandingan hak aksesnya:

Tabel 4.87 Perbandingan *Authentication User Akes*

Menu	Admin	Bendahara	Produksi	Owner
Data user	✓	✗	✗	✗
Data master	✓	✓	✓	✓
Data produksi es krim	✓	✗	✓	✗
Data penggunaan bahan	✓	✗	✓	✗
Data pemesanan penjualan	✓	✗	✓	✗
Data penjualan	✓	✗	✓	✗
Data pengiriman barang	✓	✗	✓	✗
Data pembelian bahan	✓	✓	✗	✗
Data pembayaran lain-lain	✓	✓	✗	✗
Data penggajian karyawan	✓	✗	✓	✗
Data kasbon karyawan	✓	✗	✓	✗
Laporan-laporan	✓	✗	✗	✓

untuk mengakses menu-menu tersebut membutuhkan login berikut ini adalah cara login:

1. Klik menu aplikasi → pilih *login*, maka akan muncul dialog sebagai berikut:



Gambar 4.103 Dialog Login Aplikasi

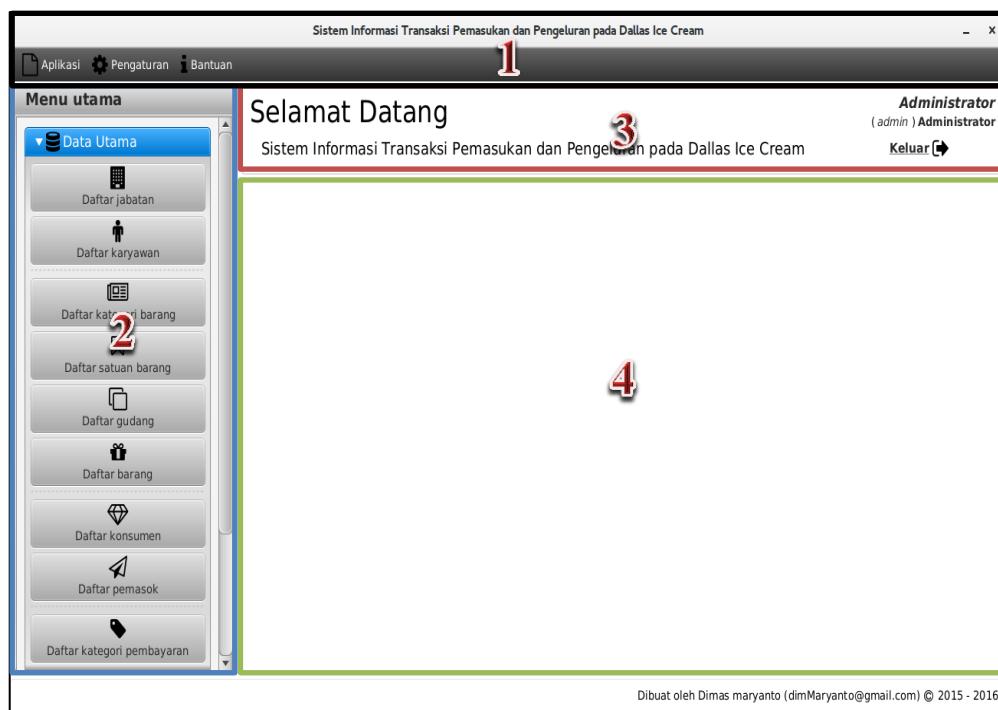
2. Input *username* dan *password* yang telah terdaftar. Jika berhasil maka hasilnya sebagai berikut:



Gambar 4.104 Menu Utama Sudah Login

4.5.6.2. Penggunaan program data master

Untuk menggunakan data master, buka *titledpanel* untuk “Data utama” seperti gambar berikut ini:



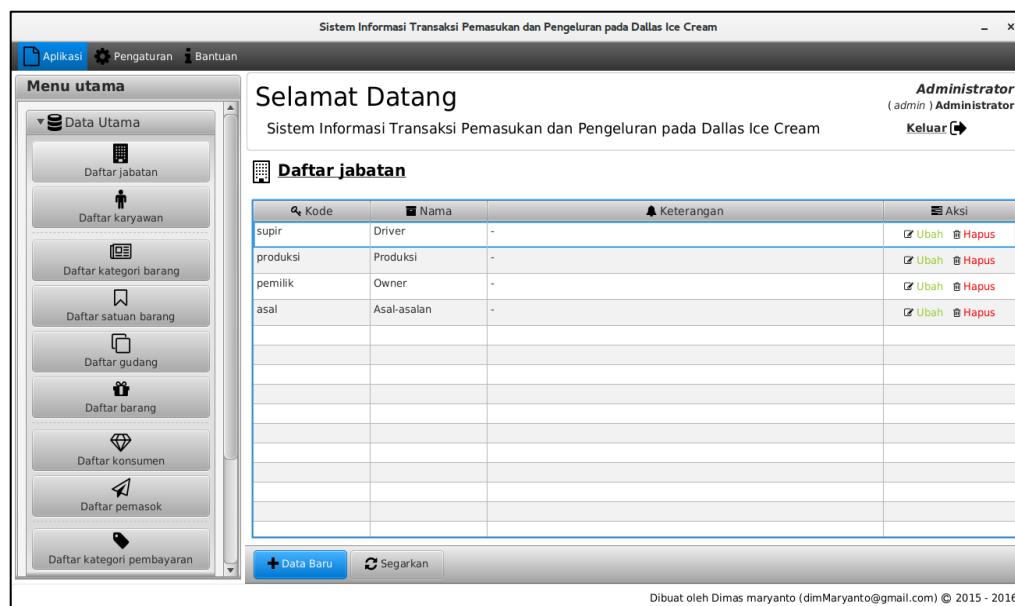
Gambar 4.105 Titledpanel Data Utama

Berikut ini adalah *layout* aplikasi berdasarkan gambar diatas:

1. *Menu bar*, berisi Aplikasi, Pengaturan dan Bantuan.
2. *Left side bar*, berisi menu utama seperti data master, produksi, pemasukan, pengeluaran dan laporan.
3. *Header content*, berisi informasi authentication user yang sedang aktif dan ucapan selamat datang.
4. *Center content*, berisi konten-konten yang dipanggil dari *left side bar*.
5. *Footer*. Berisikan *copyright* aplikasi.

a. Data master jabatan

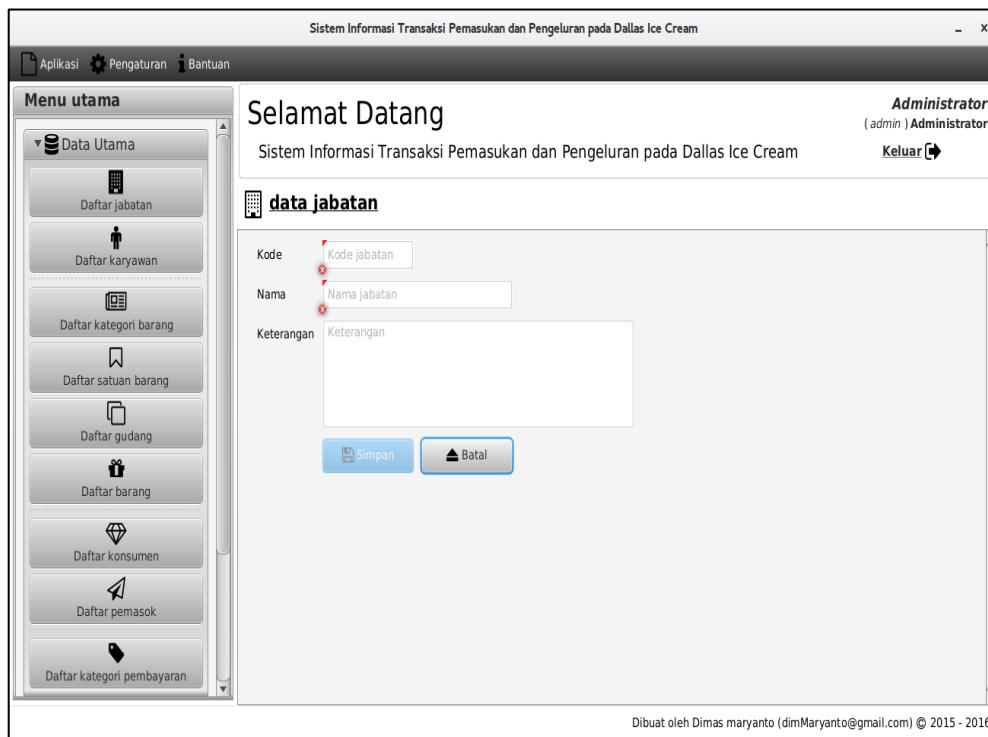
Untuk menggunakan data master jabatan, pilih “Daftar jabatan” pada *menu item* “Data Utama” kemudian pada *center content* akan menampilkan seperti pada gambar berikut:



Gambar 4.106 Daftar Data Jabatan

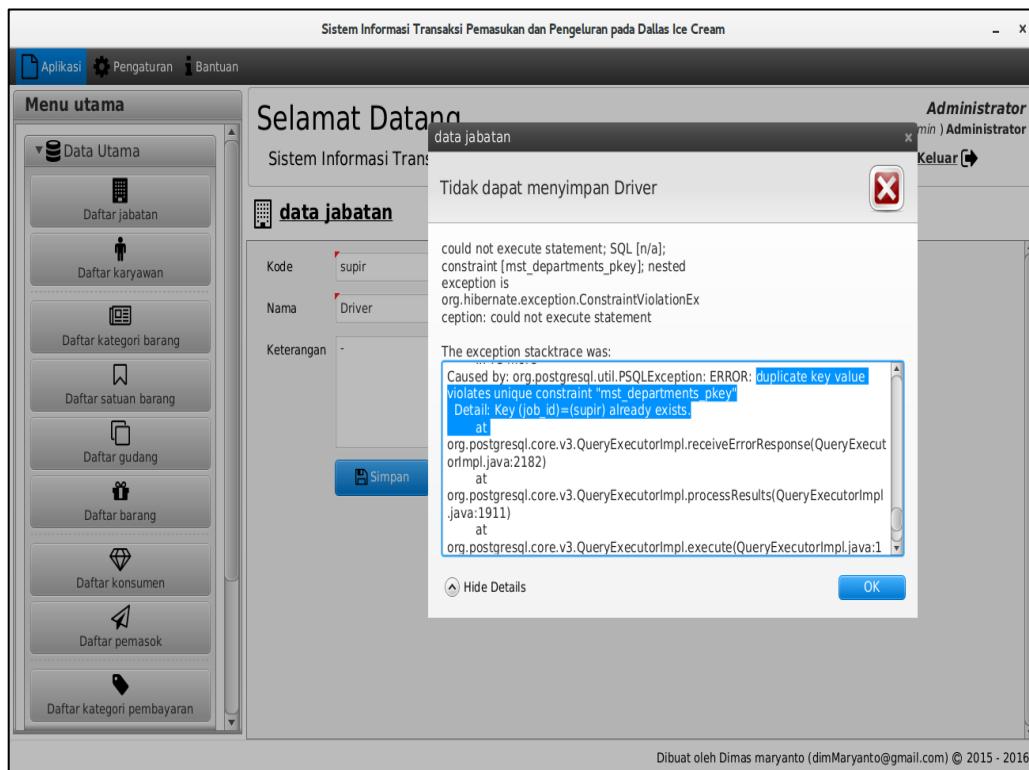
dalam data master jabatan terdapat 3 proses yaitu tambah, ubah, dan hapus data jabatan, berikut ini adalah cara tambah, ubah dan hapus data jabatan:

1. Tambah data jabatan, berikut adalah langkah-langkah untuk menambah data jabatan:
 - a. klik *button* “Data Baru” setelah itu akan muncul form input data jabatan seperti pada gambar berikut:



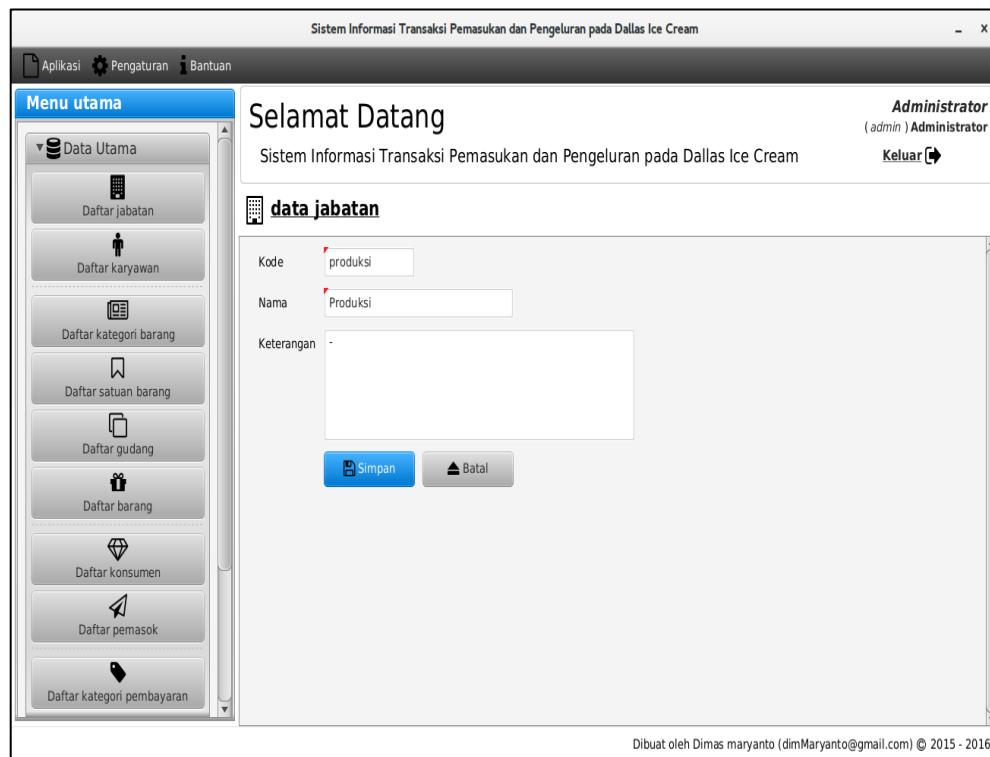
Gambar 4.107 Form Input Data Jabatan

- b. Tampilan awal *button* “Simpan” akan tidak aktif ketika kode dan nama masih kosong, berikut ini adalah aturan yang harus dipenuhi untuk mengaktifkan *button* “Simpan”.
 - i. Kode jabatan hanya dapat diisi dengan jumlah minimal 1 dan maksimal 10 karakter.
 - ii. Nama jabatan hanya dapat diisi dengan jumlah minimal 1 dan maksimal 100 karakter.
- c. Setelah kedua hal tersebut telah sesuai maka *button* simpan akan aktif dan kemudian klik *button* simpan. Jika berhasil akan ada notifikasi “Data jabatan berhasil disimpan”, jika gagal maka akan muncul pesan kesalahan seperti pada gambar berikut:



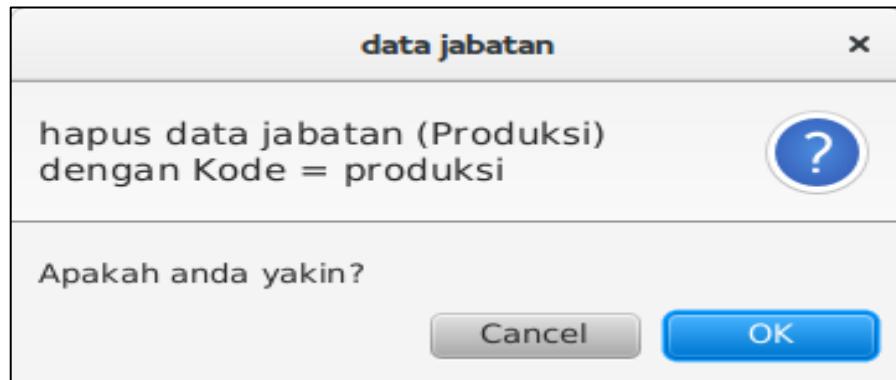
Gambar 4.108 Pesan Kesalahan Menyimpan Data Jabatan

- d. Dari gambar tersebut terlihat bahwa data dengan kode supir telah tersedia. Kemudian klik button “OK”.
 - e. Untuk kembali ke daftar data jabatan. Klik batal atau klik daftar jabatan pada *left side context*. Maka anda akan diarahkan kembali ke menu daftar jabatan.
2. Ubah data jabatan, berikut ini adalah langkah-langkah untuk mengubah informasi data jabatan.
- a. klik *hyperlink* “Ubah” pada kolom aksi dan baris yang ingin diubah. Contohnya melakukan klik pada baris ke 2 yaitu dengan kode produksi. Maka akan tampil form sebagai berikut:



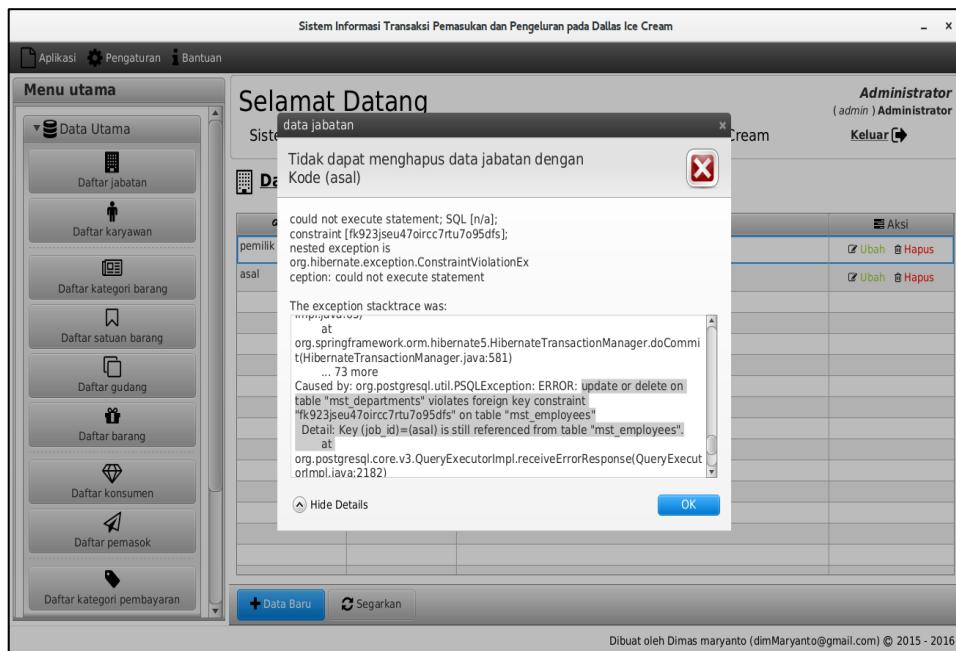
Gambar 4.109 Form Ubah Data Jabatan

- b. Pada *field* kode tidak dapat diubah, artinya hanya dapat merubah pada *field* nama dan keterangan saja.
 - c. Setelah data diubah maka klik *button* “Simpan” untuk menyimpan perubahan data dan kemudian jika berhasil dirubah akan muncul notifikasi “Data jabatan berhasil diubah”, dan form akan diarahkan ke halaman daftar data jabatan.
3. Hapus data jabatan.
- a. Untuk menghapus data jabatan klik *hyperlink* “Hapus” pada kolom aksi dan baris yang ingin dihapus. Contohnya melakukan klik pada baris ke 2 yaitu dengan kode produksi. Maka akan tampil konfirmasi *dialog* sebagai berikut:



Gambar 4.110 Konfirmasi Hapus Data Jabatan

- b. Setelah itu klik “Ok”, untuk menghapus. Klik “Cancel”, untuk membatalkan hapus data produksi.
- c. Jika muncul pesan kesalahan sebagai berikut:



Gambar 4.111 Pesan Kesalahan Hapus Data Jabatan

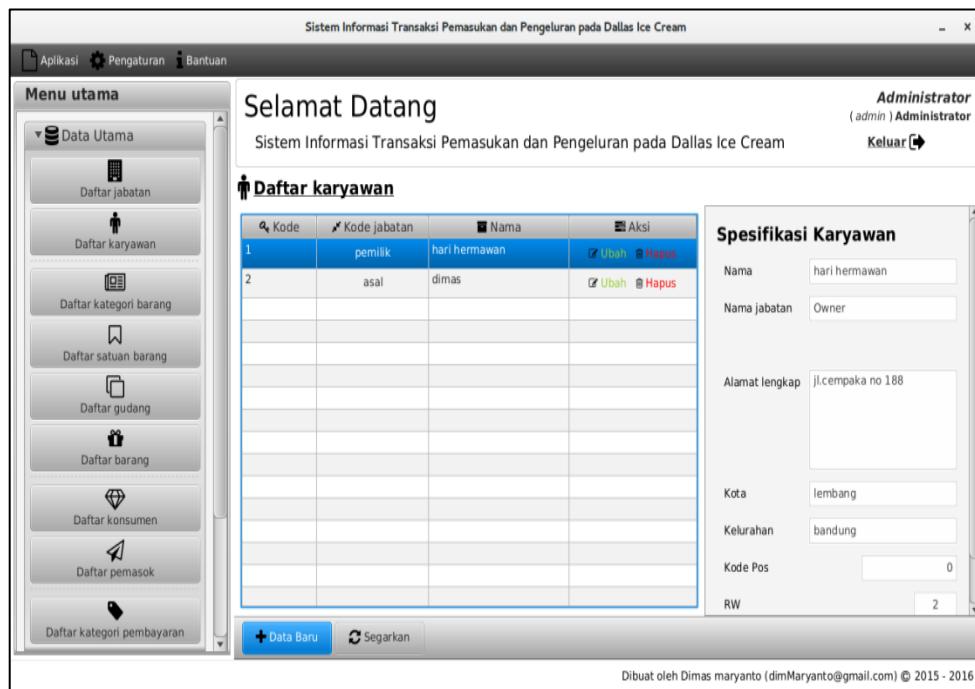
- d. Maka data jabatan akan dibatalkan untuk dihapus karena data tersebut telah digunakan oleh data karyawan.

b. Data master karyawan

Untuk menggunakan data master karyawan, pilih “Daftar karyawan” pada *menu item* “Data master”, kemudian pada *center content* akan menampilkan *form* sebagai berikut:

Gambar 4.112 Daftar Data Karyawan

dalam *menu* daftar karyawan terdiri dari 2 bagian *table-view* (tabel daftar karyawan) dan *employee-details* (spesifikasi karyawan). Jadi ketika pada *table-view* di klik pada baris tertentu maka employee details akan terisi secara otomatis berdasarkan kode karyawan, contohnya pada *table-view* diklik pada baris ke 2 yaitu dengan kode “1” maka pada *employee-details* akan tampil seperti berikut:



Gambar 4.113 Table View Employees Row Selected

Dalam data master karyawan terdiri dari 3 proses utama yaitu tambah, ubah, hapus data karyawan. Berikut ini adalah proses tambah, ubah, dan hapus data karyawan:

1. Tambah data karyawan, berikut ini adalah langkah-langkah untuk menambahkan data karyawan baru.
 - a. klik *button* “Data Baru” kemudian akan muncul *form* sebagai berikut:

The screenshot shows the 'data karyawan' (Employee Data) form. The left sidebar has a 'Menu utama' section with various icons and labels: Data Utama, Daftar jabatan, Daftar karyawan, Daftar kategori barang, Daftar satuan barang, Daftar gudang, Daftar barang, Daftar konsumen, Daftar pemasok, and Daftar kategori pembayaran. The main area displays a 'Selamat Datang' message and the system title 'Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream'. On the right, there's an 'Administrator (admin) Administrator' label and a 'Keluar' button. The 'data karyawan' form contains fields for 'Nama' (Name), 'Pilih jabatan' (Select Position), 'Kota' (City), 'Kelurahan' (Sub-District), 'Alamat lengkap' (Full Address), 'RT' (RT), 'RW' (RW), and 'Kode Pos' (Zip Code). At the bottom are 'Simpan' (Save) and 'Batal' (Cancel) buttons. A footer note at the bottom right states 'Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016'.

Gambar 4.114 Form Data Karyawan Baru

- b. Tampilan awal, *button* “Simpan” akan tidak aktif karena ada beberapa data harus dipenuhi seperti berikut:
 - i. Nama karyawan tidak boleh kosong.
 - ii. Nama karyawan harus diisi dengan minimal 1 dan maksimal 100 karakter termasuk spasi.
 - iii. Jabatan harus dipilih.
 - iv. Data lain-lain boleh tidak diisi, tetapi disarankan untuk diisi.
- c. Setelah data tersebut terpenuhi maka *button* “Simpan” akan aktif dan klik *button* “Simpan” untuk mengimpor data karyawan baru. Jika sukses maka akan menampilkan notifikasi “Data karyawan

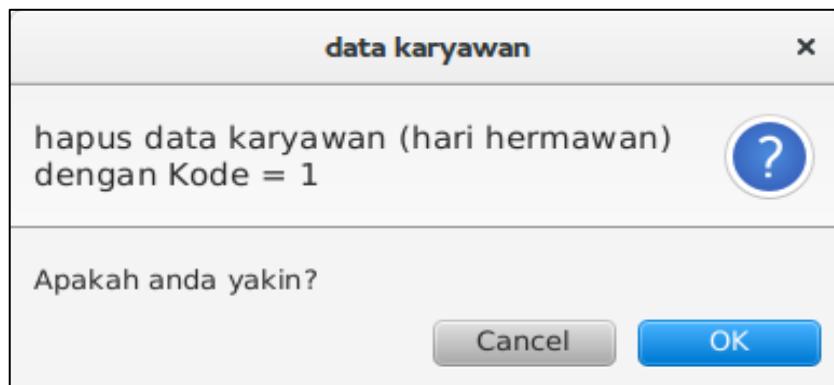
berhasil disimpan” atau klik *button* “Batal” untuk kembali ke menu Daftar karyawan.

2. Ubah data karyawan, berikut adalah langkah-langkah untuk mengubah informasi data karyawan.
 - a. Untuk mengubah data karyawan klik *Hyperlink* “Ubah” pada kolom aksi dan pada baris tertentu, contohnya akan mengubah data karyawan yang memiliki kode 1, maka akan tampil form data karyawan sebagai berikut:

The screenshot shows the 'Edit Data Karyawan' form in the Dallas Ice Cream system. The form includes fields for Name, Position, City, Sub-district, Full Address, RT, RW, and Pos Code. The 'Nama' field contains 'hari', 'Pilih jabatan' dropdown shows '(supir) Driver', 'Kota' is 'lembang', 'Kelurahan' is empty, 'Alamat lengkap' is 'jl.cempaka no 8', 'RT' is '0', 'RW' is '0', and 'Kode Pos' is '0'. At the bottom are 'Simpan' and 'Batal' buttons. The top right shows the user 'Dimas Maryanto (dimas) Administrator' and a 'Keluar' button.

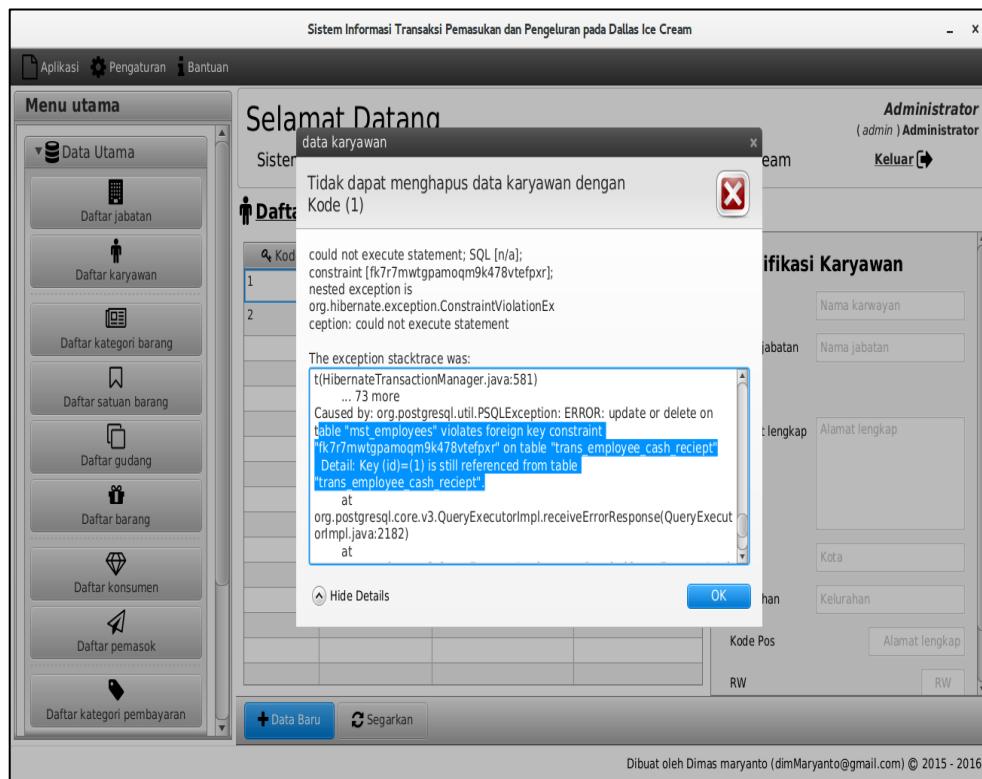
Gambar 4.115 Form Edit Data Karyawan

- b. Kemudian setelah itu klik button “Ok” untuk menyimpan hasil perubahan dan anda akan diarahkan kembali ke menu daftar data karyawan.
3. Hapus data karyawan, berikut ini adalah langkah-langkah untuk menghapus data karyawan:
- a. Untuk menghapus data karyawan klik *Hyperlink* “Hapus” pada kolom aksi dan pada baris tertentu, Contohnya akan menghapus karyawan yang memiliki kode 1, maka akan muncul konfirmasi dialog sebagai berikut:



Gambar 4.116 Konfirmasi Hapus Data Karyawan

- b. Setelah itu klik *button* “Ok” untuk menghapus atau klik *button* “Cancel” untuk membatalkan menghapus data.
- c. Ketika button “Ok” diklik tetapi menemukan pesan kesalahan berarti data karyawan tersebut telah digunakan diantaranya untuk pengajian, kasbon karyawan, produksi, dan pengiriman penjualan, seperti berikut:

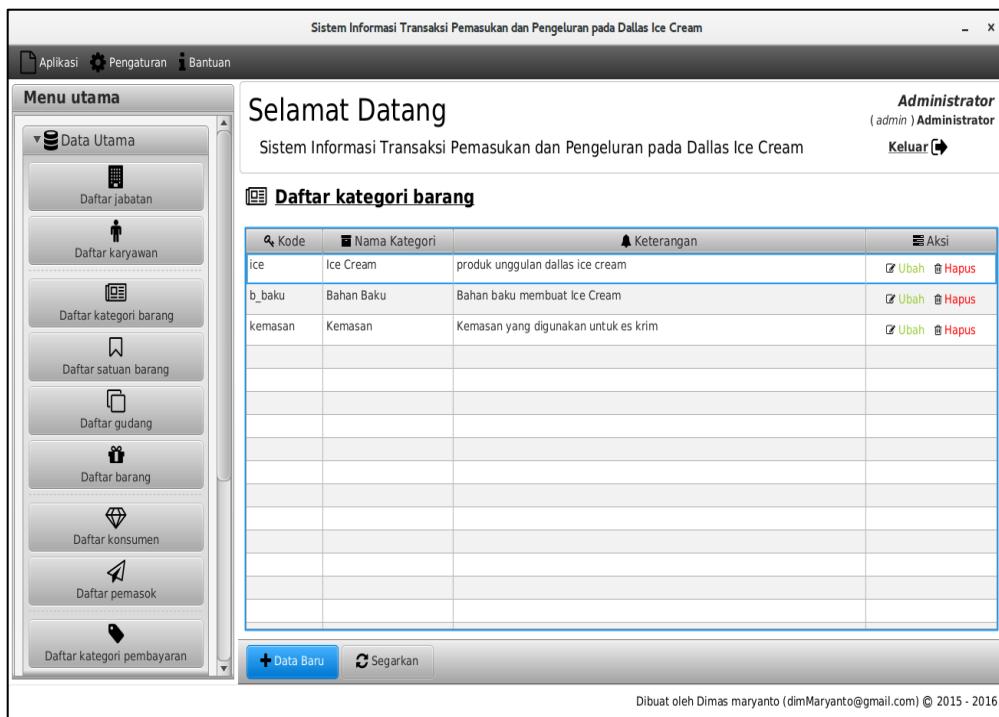


Gambar 4.117 Pesan Kesalahan Hapus Data Karyawan

4. Refresh data karyawan, button “Segarkan” digunakan untuk memuat ulang data atau melakukan sinkronisasi dengan basis data.

c. Data master kategori barang

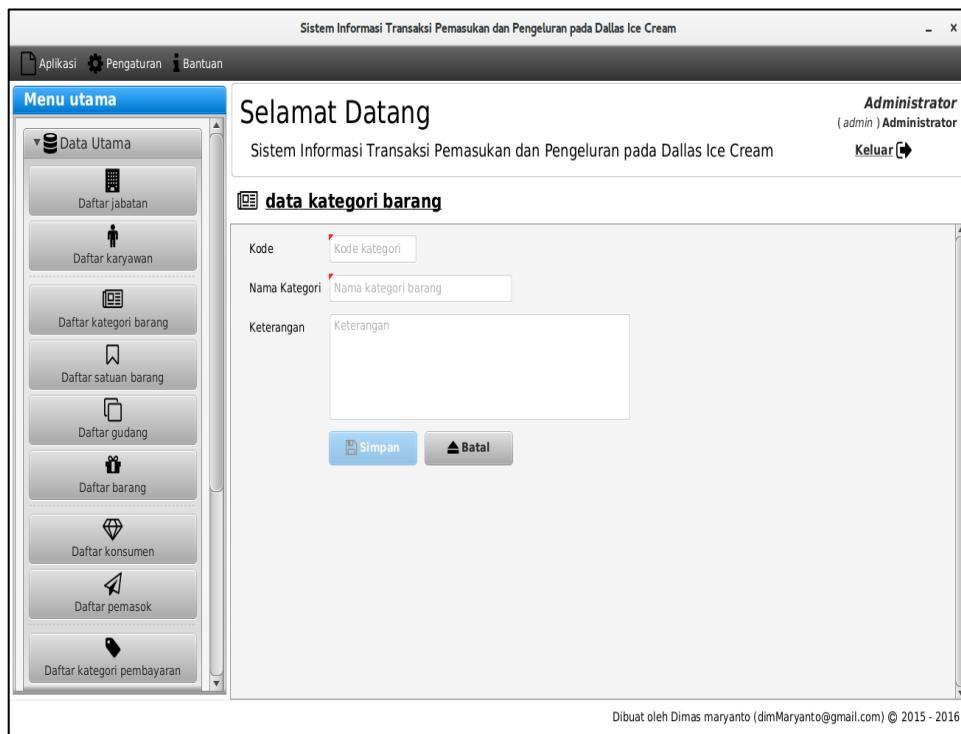
Untuk menggunakan data master kategori barang pilih *menu item* “Daftar kategori barang” pada menu “Data Utama” maka akan menampilkan *form* pada *center content* sebagai berikut:



Gambar 4.118 Daftar Data Kategori Barang

data master kategori barang terdiri dari 3 proses utama yaitu tambah, ubah, hapus seperti berikut ini:

1. Tambah data kategori barang, berikut ini adalah langkah-langkah untuk menambahkan data kategori barang:
 - a. klik *button* “Data baru” setelah itu akan tampil *form* input data kategori barang seperti berikut:



Gambar 4.119 Form Kategori Barang Baru

- b. Tampilan awal *button* “Simpan” tidak aktif, karena ada beberapa ketentuan yang harus dipenuhi yaitu sebagai berikut:
 - i. Kode tidak boleh kosong, minimal harus diisi 1 karakter, batas maksimal 100 karakter dan harus bersifat *unique*.
 - ii. Nama kategori tidak boleh kosong, minimal harus diisi dengan 1 karakter dan batas maksimal 100 karakter
 - iii. Keterangan boleh tidak diisi.
- c. Setelah semua ketentuan tersebut terpenuhi maka *button* “Simpan” akan aktif. Kemudian klik *button* “Simpan”
2. Ubah data kategori barang, berikut ini adalah langkah-langkah untuk mengubah informasi data kategori barang:

a. klik *hyperlink* “Ubah” pada kolom aksi dan pada baris tertentu.

Contohnya ingin mengubah kategori barang dengan kode “ice” caranya adalah klik *hyperlink* “Ubah” pada baris pertama, kemudian akan muncul *form* kategori barang sebagia berikut:

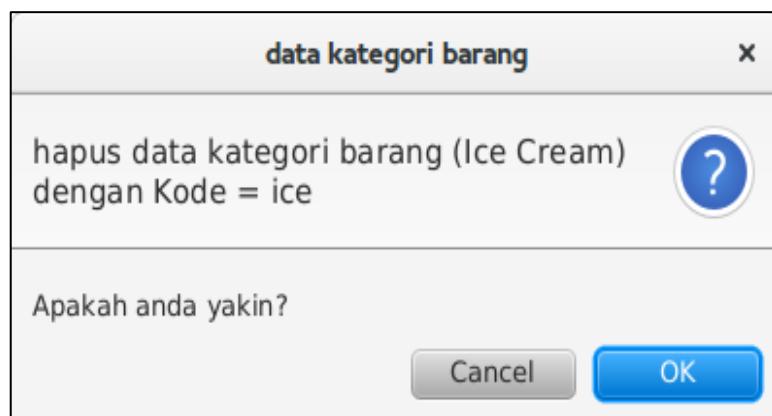
The screenshot shows a Windows application window titled "Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream". The menu bar includes "Aplikasi", "Pengaturan", and "Bantuan". The top right corner shows the administrator status: "Administrator (admin) Administrator" and a "Keluar" (Logout) button. On the left, a sidebar titled "Menu utama" lists various data management modules: Data Utama (Daftar jabatan, Daftar karyawan, Daftar kategori barang, Daftar satuan barang, Daftar gudang, Daftar barang, Daftar konsumen, Daftar pemasok, Daftar kategori pembayaran). The main content area displays a message "Selamat Datang" and the title "data kategori barang". A single category item is listed with the following details: Kode: ice, Nama Kategori: Ice Cream, Keterangan: produk unggulan dallas ice cream. At the bottom are "Simpan" and "Batal" buttons. The footer of the application window indicates it was created by Dimas Maryanto (dimMaryanto@gmail.com) © 2015 - 2016.

Gambar 4.120 Form Edit Kategori Barang

- b. Aturan menggunakan *form edit* kategori barang adalah sebagai berikut:
- Kode tidak dapat diubah.
 - Nama kategori harus diisi minimal 1 dan maksimal 100 karakter.
 - Keterangan boleh tidak diisi.
- c. Setelah data diubah maka klik *button* “Simpan” untuk menyimpan perubahan data yang terjadi. Jika berhasil maka akan menampilkan

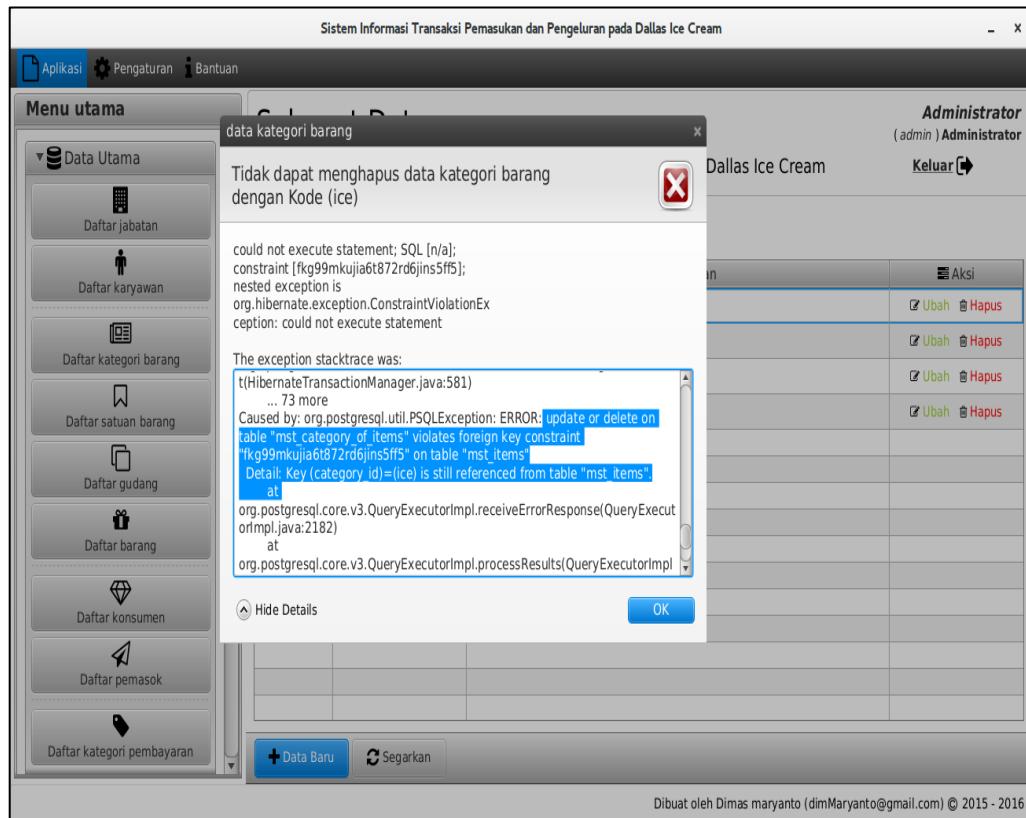
notifikasi “Data jabatan berhasil diubah” dan anda akan diarahkan kembali ke menu “Daftar kategori barang”.

3. Hapus data kategori barang, berikut ini adalah langkah-langkah untuk menghapus data kategori barang:
 - a. klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu. Contohnya ingin menghapus kategori barang dengan kode “ice”, caranya adalah klik *hyperlink* “Hapus” pada kolom aksi dan pada baris pertama kemudian akan tampil dialog konfirmasi sebagai berikut:



Gambar 4.121 Konfirmasi Hapus Data Kategori Barang

- b. Setelah itu klik button “Ok” untuk menghapus data kategori barang dengan kode “ice”.
- c. Jika berhasil dihapus maka akan tampil notifikasi “Data kategori barang berhasil dihapus”, jika gagal maka akan tampil pesan kesalahan yang mungkin disebabkan oleh data kategori barang telah digunakan seperti berikut:

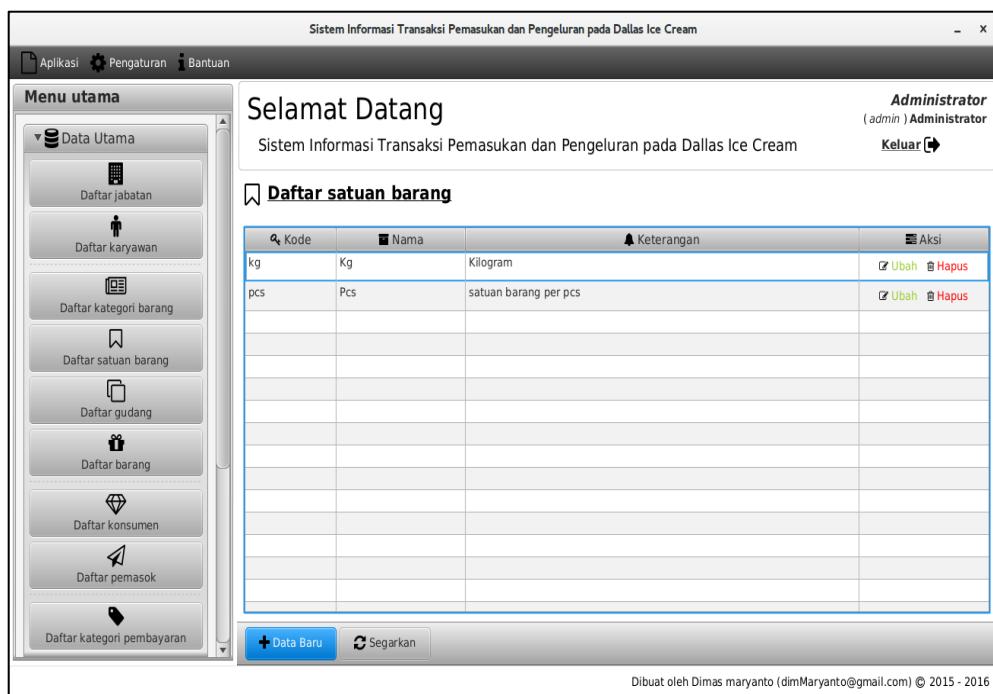


Gambar 4.122 Pesan Kesalahan Untuk Hapus Data Kategori Barang

4. *Refresh* data kategori barang, button “Segarkan” digunakan untuk memuat ulang atau singkronisasi dengan data yang terdapat di basis data.

d. Data master satuan barang

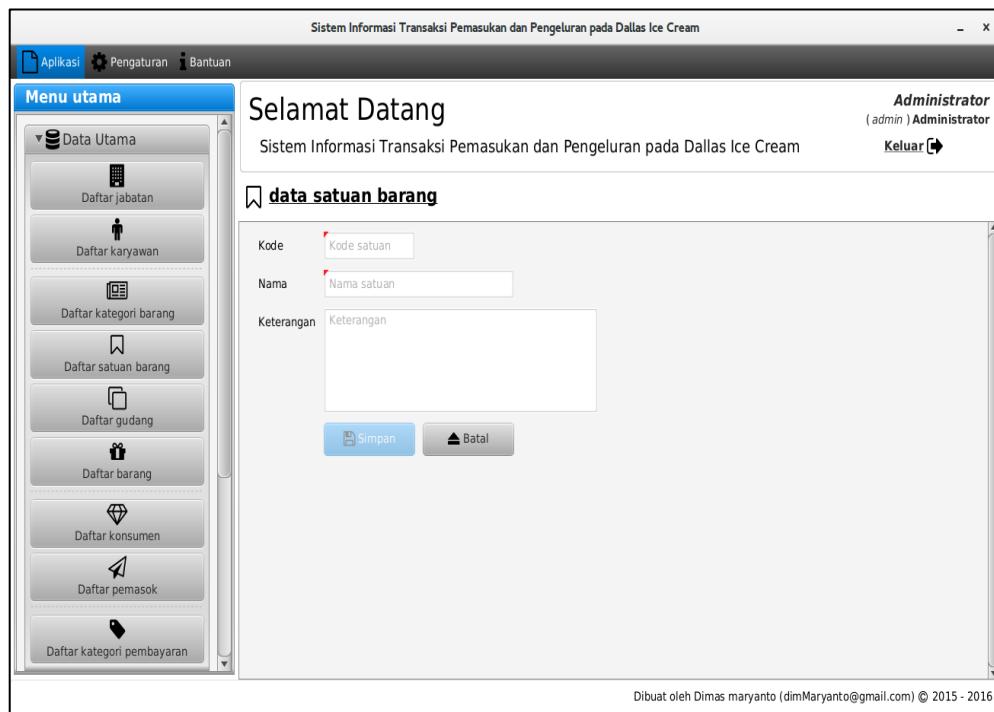
Untuk menggunakan data master satuan barang, pilih “Daftar satuan barang” pada menu “Data utama” setelah itu *center content* akan menampilkan *form* sebagai berikut:



Gambar 4.123 Daftar Data Satuan Barang

dalam data master satuan barang terdapat 3 proses utama yaitu tambah, ubah dan hapus data, berikut ini adalah cara tambah, ubah dan hapus data satuan barang:

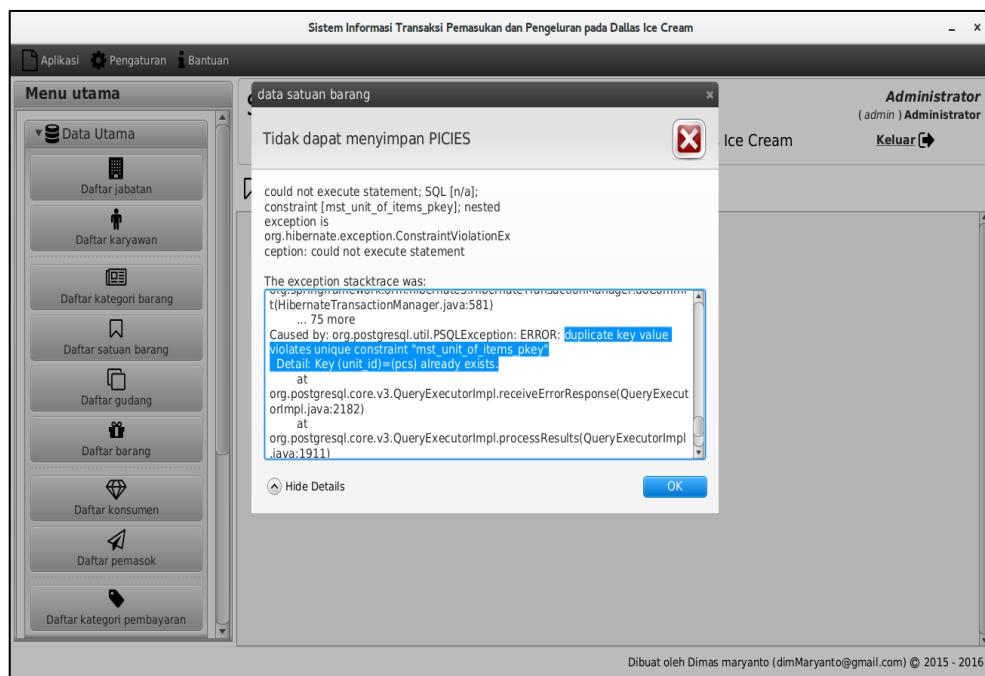
1. Tambah data satuan barang, berikut ini adalah langkah-langkah untuk menambahkan data satuan barang:
 - a. Klik *button* “Data Baru” seperti pada gambar diatas, kemudian akan tampil *form input* data satuan barang seperti berikut:



Gambar 4.124 Form Input Data Satuan Barang

- b. Tampilan awal *button* “Simpan” tidak aktif, untuk mengaktikannya berikut adalah aturannya:
 - i. Kode satuan barang harus diisi minimal 1 dan maksimal 10 karakter.
 - ii. Kode satuan barang harus bersifat *unique* atau tidak boleh ada data yang sama
 - iii. Nama satuan barang harus diisi minimal 1 dan maksimal 100 karakter.
 - iv. Keterangan satuan barang boleh tidak diisi.
- c. Setelah aturan tersebut terpenuhi maka secara otomatis *button* “Simpan” akan aktif.

- d. kemudian klik button “Simpan” untuk menyimpan data satuan barang baru.
- e. Jika data berhasil disimpan maka akan menampilkan notifikasi “Data satuan barang berhasil disimpan”, tetapi jika tidak akan menampilkan pesan kesalahan diantarnya kode satuan barang telah digunakan seperti berikut:



Gambar 4.125 Pesan Kesalahan Input Data Satuan Barang

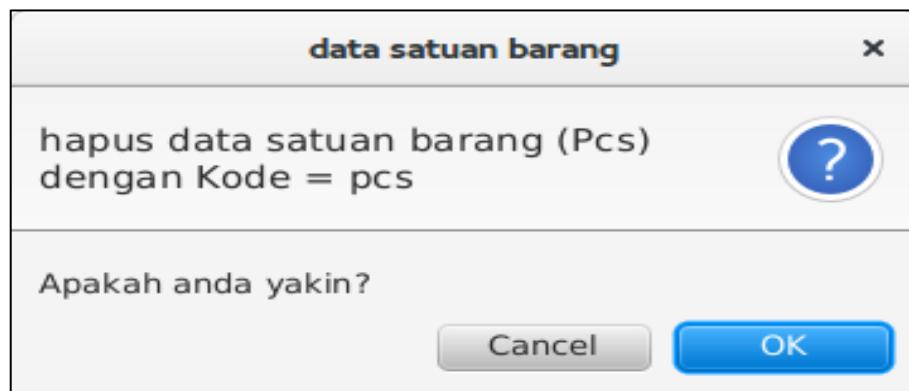
2. Ubah data satuan barang, berikut ini adalah langkah-langkah untuk mengubah data satuan barang:
 - a. Klik *hyperlink* “Ubah” pada kolom aksi dan baris tertentu. Contohnya ingin mengubah data satuan barang dengan kode “kg”. maka klik *hyperlink* “Ubah” pada baris pertama, maka akan muncul *form edit* data satuan barang seperti berikut:



Gambar 4.126 Form Edit Data Satuan Barang

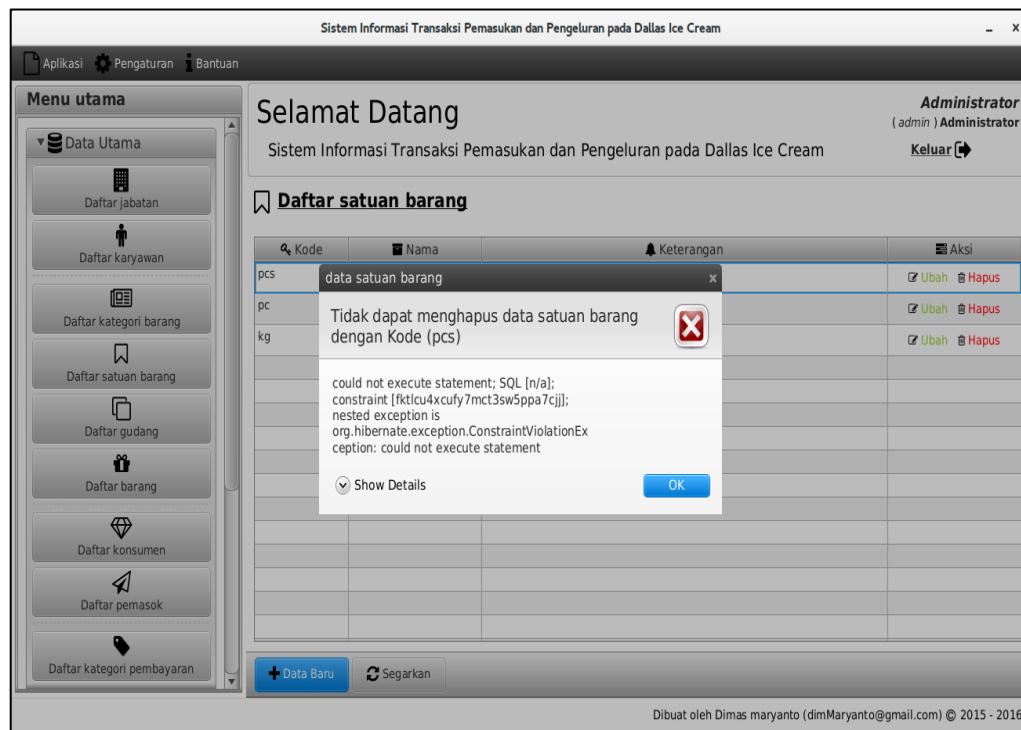
- b. Aturan penggunaan *form edit* data satuan barang adalah sebagai berikut:
 - i. Kode tidak dapat diubah.
 - ii. Nama harus diisi minimal 1 dan maksimal 100 karakter.
 - iii. Keterangan boleh tidak diisi.
- c. Setelah data diubah kemudian klik button “Simpan” untuk menyimpan perubahan data satuan barang, jika berhasil maka anda akan diarahkan kembali ke menu “Daftar satuan barang”
- 3. Hapus data satuan barang, berikut ini adalah langkah-langkah untuk menghapus data satuan barang:
 - a. Klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu. Contohnya ingin menghapus satuan barang dengan kode “kg”, maka klik

hyperlink “Hapus” pada baris pertama kemudian akan tampil konfirmasi dialog sebagai berikut:



Gambar 4.127 Konfirmasi Hapus Data Satuan Barang

- b. Klik *button* “Ok” untuk menghapus data satuan barang dengan kode “kg” atau klik *button* “Cancel” untuk membatalkan untuk menghapus data satuan barang dengan kode “kg”
- c. Ketika *button* “Ok” diklik dan menampilkan notifikasi “Data satuan barang berhasil dihapus” berarti data satuan barang dengan kode “kg” berhasil dihapus tetapi jika menampilkan pesan kesalahan “tidak dapat menghapus data satuan barang” berarti data satuan barang dengan kode “kg” telah digunakan untuk barang tertentu contohnya seperti berikut:



Gambar 4.128 Pesan Kesalahan Hapus Data Satuan Barang

4. Refresh data satuan barang, button “Segarkan” digunakan untuk memuat ulang atau melakukan sinkronisasi data aplikasi dengan data basis data.

e. Data master gudang

Untuk menggunakan data master gudang, pilih “Daftar gudang” pada menu “Data utama” setelah itu pada *center content* akan menampilkan *form* sebagai berikut:

Kode	Telpo	Nama	Aksi
1	987492575	CENTRAL	<input checked="" type="checkbox"/> Ubah <input type="checkbox"/> Hapus

Spesifikasi Gudang

Nama	<input type="text" value="Nama gudang"/>
Telpo	<input type="text" value="Kontak personal"/>
Alamat lengkap	<input type="text" value="Alamat lengkap"/>
Kota	<input type="text" value="Kota"/>
Kelurahan	<input type="text" value="Kelurahan"/>
Kode Pos	<input type="text" value="Kode Pos"/>
RW	<input type="text" value="RW"/>
RT	<input type="text" value="RT"/>

[+ Data Baru](#) [Segarkan](#)

Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

Gambar 4.129 Daftar Data Gudang

dalam data master gudang terdapat 3 proses utama yaitu tambah, ubah dan hapus data gudang, berikut ini adalah cara tambah, ubah dan hapus data gudang:

1. Tambah data gudang, berikut ini adalah langkah-langkah untuk menambahkan data gudang:
 - a. Klik button “Data baru” seperti pada gambar diatas, kemudian akan menampilkan *form input* data gudang seperti berikut:

The screenshot shows a Windows application window titled "Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream". The window has a top navigation bar with "Applikasi", "Pengaturan", and "Bantuan". On the right, there is a user session area showing "Administrator (admin) Administrator" and a "Keluar" button. The main content area is titled "Selamat Datang" and "data gudang". It contains several input fields: "Nama" (with placeholder "Nama gudang"), "Telpon" (with placeholder "Kontak personal"), "Alamat lengkap" (with placeholder "Alamat lengkap"), "Kota" (with placeholder "Kota"), "Kelurahan" (with placeholder "Kelurahan"), "RT" (with value "0"), "RW" (with value "0"), and "Kode Pos" (with value "0"). At the bottom are two buttons: "Simpan" (highlighted in blue) and "Batal". A copyright notice at the bottom right reads "Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016".

Gambar 4.130 Form Input Data Gudang

- b. Tampilan awal *button* “Simpan” tidak aktif, untuk mengaktifkannya berikut adalah aturannya:
 - i. Nama gudang harus diisi minimal 1 dan maksimal 100 karakter termasuk spasi
 - ii. Telpon harus diisi dengan angka dan hanya dapat menggunakan karakter pemisah “-“
 - iii. Data lain-lain boleh dikosongkan.
 - c. Setelah aturan tersebut terpenuhi maka secara otomatis *button* “Simpan” akan aktif.
 - d. Kemudian klik *button* “Simpan” untuk mengimpor data gudang baru.
2. Ubah data gudang, berikut ini adalah langkah-langkah untuk mengubah data gudang:
 - a. Klik *hyperlink* “Ubah” pada kolom aksi dari baris tertentu untuk data gudang. Contohnya ingin mengubah data gudang dengan kode “1”,

maka klik *hyperlink* “Ubah” pada baris pertama kemudian akan muncul *form edit* data gudang seperti berikut:

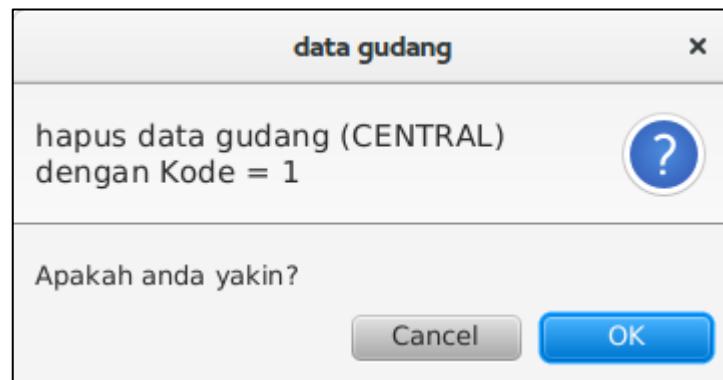
The screenshot shows a software interface titled "Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream". On the left is a sidebar menu titled "Menu utama" with various options like "Data Utama", "Daftar jabatan", "Daftar karyawan", etc. The main area is titled "Selamat Datang" and "data gudang". It displays a form with the following data:

Nama	CENTRAL
Telpoin	987492575
Alamat lengkap	jl.cenpaka no 2 Lembang bandung
Kota	lembang
Kelurahan	bandung
RT	0
RW	0
Kode Pos	0

At the bottom are "Simpan" and "Batal" buttons.

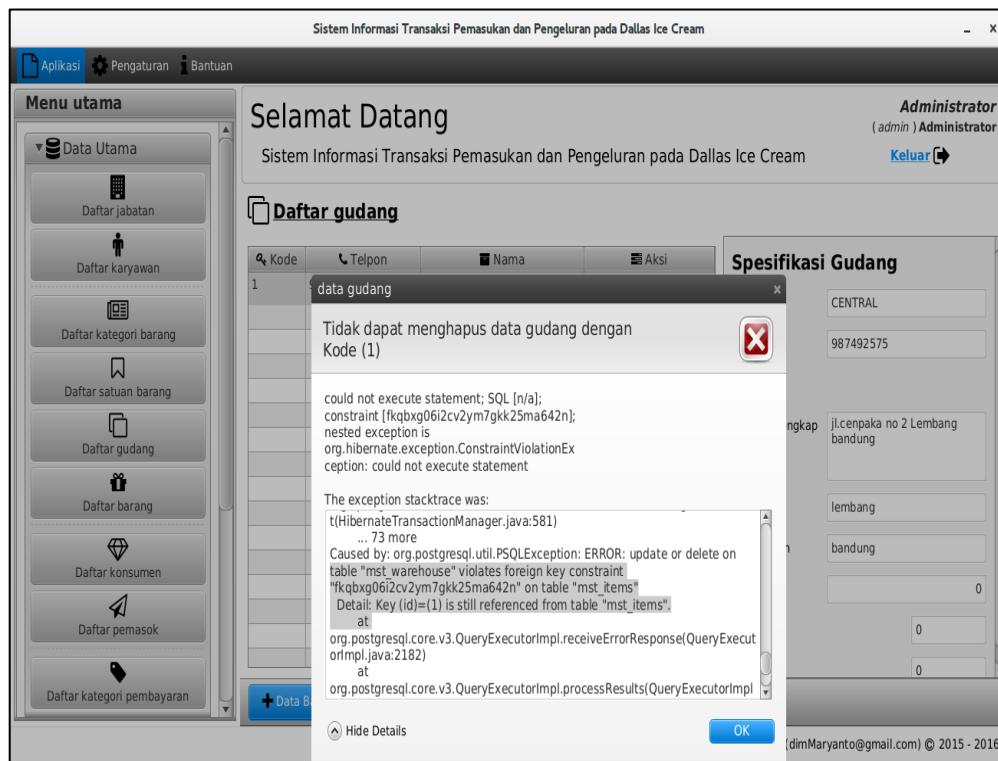
Gambar 4.131 Form Edit Data Gudang

- b. Tidak ada aturan yang berbeda dengan input data gudang.
 - c. Setelah data diubah kemudian klik button “Simpan” untuk menyimpan perubahan data gudang, jika berhasil maka anda akan diarahkan kembali ke halaman “Daftar gudang”.
3. Hapus data gudang, berikut ini adalah langkah-langkah untuk menghapus data gudang:
- a. Klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu pada tabel daftar data gudang, contohnya ingin menghapus data gudang dengan nama “CENTRAL” maka klik *hyperlink* “Hapus” pada baris yang memiliki nama “CENTRAL” kemudian akan tampil dialog konfirmasi seperti berikut:



Gambar 4.132 Konfirmasi Hapus Data Gudang

- b. Klik *button* “OK” untuk menghapus data gudang dengan nama “CENTRAL” atau klik *button* ”Cancel” untuk membatalkan hapus data gudang dengan nama “CENTRAL”.
- c. Ketika *button* “OK” diklik dan menampilkan notifikasi “Data gudang telah berhasil dihapus” berarti data gudang dengan nama “CENTRAL” telah dihapus tetapi jika menampilkan pesan kesalahan “tidak dapat menghapus data gudang” berarti data gudang telah digunakan untuk barang tertentu contohnya seperti berikut:



Gambar 4.133 Pesan Kesalahan Hapus Data Gudang

4. *Refresh* data gudang, button “Segarkan” digunakan untuk memuat ulang atau melakukan singkronisasi data aplikasi dengan data pada tabel gudang di dalam basis data.

f. Data master barang

Untuk menggunakan data master barang, pilih “Daftar barang” pada menu “Data utama” setelah itu pada *center content* akan menampilkan *form* seperti berikut:

Kode lokasi	Kode lokasi	Nama	QTY/Satuan	Aksi
CENTRAL	Ice Cream	Cup Kecil 70ml	0/pcs	[Ubah]
CENTRAL	Ice Cream	Fruitmilk	0/pcs	[Ubah]
CENTRAL	Ice Cream	Choco	0/pcs	[Ubah]
CENTRAL	Ice Cream	Vanies	0/kg	[Ubah]
CENTRAL	Bahan Baku	Hanco Vanila	0/pcs	[Ubah]
CENTRAL	Bahan Baku	Hanco Coklat	0/pcs	[Ubah]
CENTRAL	Kemasan	Tutup Cup	0/pcs	[Ubah]
CENTRAL	Kemasan	Corn	0/pcs	[Ubah]
CENTRAL	Kemasan	Ball	0/pcs	[Ubah]
CENTRAL	Kemasan	Stick	0/pcs	[Ubah]
CENTRAL	Ice Cream	Berrnut	0/kg	[Ubah]
CENTRAL	Ice Cream	Davacho	0/kg	[Ubah]

Gambar 4.134 Daftar Data Barang

dalam data master barang terdiri dari 2 kategori utama yaitu produk dan bahan baku. Produk adalah barang yang dijual berdasarkan hasil produksi yang dilakukan, sedangkan bahan baku merupakan barang yang dibeli yang digunakan untuk memproduksi dan memungkinkan bisa dijual kembali.

Data master barang terdapat 3 proses utama yaitu tambah, ubah dan hapus data barang, berikut ini adalah cara untuk tambah, ubah dan hapus data barang:

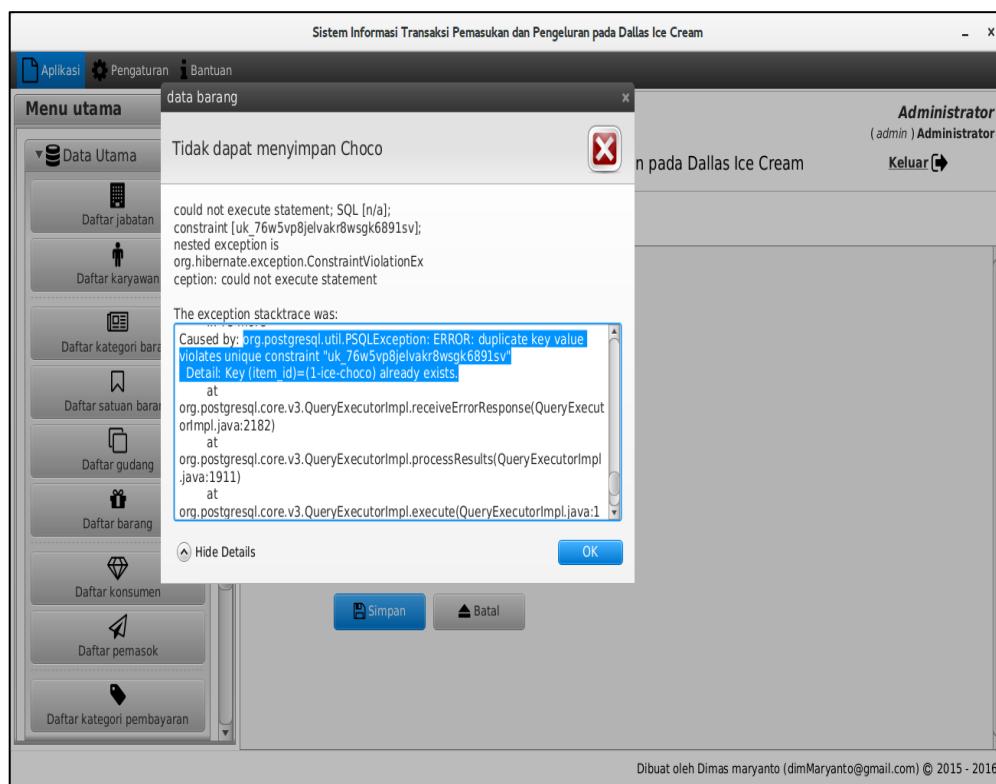
1. Tambah data barang, berikut ini adalah langkah-langkah untuk menambahkan data barang:
 - a. Klik button “Data baru” seperti pada gambar diatas, kemudian akan tampil *form* input data seperti berikut:

The screenshot shows a software interface titled "Selamat Datang" for managing inventory at "Dallas Ice Cream". On the left, there's a sidebar menu under "Menu utama" with various options like "Data Utama", "Daftar jabatan", "Daftar karyawan", etc. The main area is titled "data barang" and contains a form for adding a new item. The form fields are: Nama (Nama barang), Kategori (Kategori barang), Satuan (Satuan barang), Lokasi barang (Lokasi barang), Harga jual (Harga jual), Harga beli (Harga beli), and Dijual (Dijual). The "Simpan" button is highlighted in blue, indicating it is active. The bottom right corner of the window displays the copyright information: "Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016".

Gambar 4.135 Form Input Data Barang

- b. Tampilan awal button “Simpan” tidak aktif, untuk mengaktifkannya terikut adalah aturannya:
 - i. Nama barang tidak boleh kosong.
 - ii. Nama barang harus diisi minimal 1 dan maksimal 25 karakter.
 - iii. Kategori barang harus dipilih.
 - iv. Satuan barang harus dipilih.

- v. Lokasi atau gudang harus dipilih.
- c. Setelah aturan tersebut terpenuhi maka secara otomatis *button* “Simpan” akan aktif.
- d. Kemudian klik *button* “Simpan” untuk menyimpan data barang baru.
- e. Jika berhasil maka akan menampilkan notifikasi “Data barang berhasil ditambahkan”, tetapi jika tidak maka akan menampilkan pesan kesalahan diantaranya kode barang telah digunakan seperti pesan berikut ini:



Gambar 4.136 Pesan Kesalahan Input Data Barang

2. Ubah data barang, berikut ini adalah langkah-langkah untuk mengubah data barang:

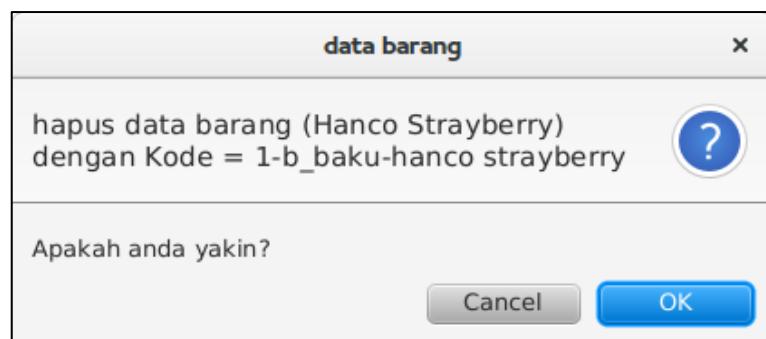
- a. Klik *hyperlink* “Ubah” pada kolom aksi dan baris tertentu. Contohnya ingin mengubah data barang dengan nama “Cup Kecil 70ml” maka klik *hyperlink* “Ubah” pada baris pertama kemudian akan muncul *form edit* barang seperti berikut:

The screenshot shows a Windows application window titled "Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream". The main title bar says "Selamat Datang" and "Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream". On the right, there's an "Administrator (admin) Administrator" label and a "Keluar" button. The left sidebar is labeled "Menu utama" and contains a list of items: Data Utama, Daftar jabatan, Daftar karyawan, Daftar kategori barang, Daftar satuan barang, Daftar gudang, Daftar barang, Daftar konsumen, Daftar pemasok, and Daftar kategori pembayaran. The main panel is titled "data barang" and displays a form for editing a product. The form fields are: "Nama" (Name) with value "Cup Kecil 70ml", "Kategori" (Category) with value "ice", "Satuan" (Unit) with value "pcs", "Lokasi barang" (Location) with value "(1) CENTRAL", "Harga jual" (Selling Price) with value "700", "Harga beli" (Purchase Price) with value "0", and "Dijual" (For Sale) with the radio button selected for "Dijual". At the bottom are "Simpan" (Save) and "Batal" (Cancel) buttons. The footer of the window says "Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016".

Gambar 4.137 Form Edit Data Barang

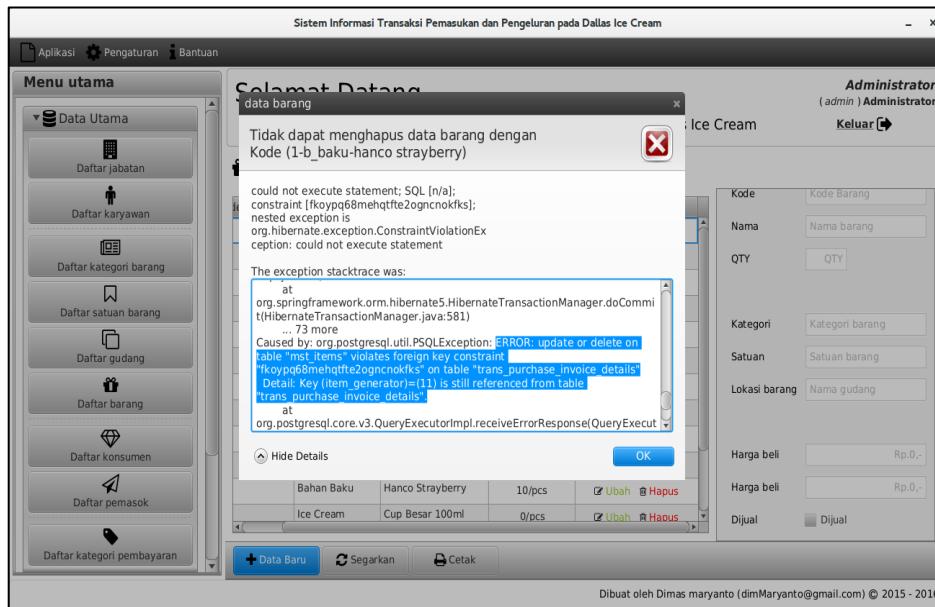
- b. Aturan penggunaan *form edit* data barang sama seperti input data barang.
- c. Setelah diubah kemudian klik *button* “Simpan” untuk menyimpan perubahan data barang, jika berhasil maka anda akan diarahkan kembali ke menu “Daftar barang”
3. Hapus data barang, berikut ini adalah langkah-langkah untuk menghapus data barang:
- a. Klik *hyperlink* “Hapus” pada *table-view* barang dikolom aksi dan baris tertentu. Contohnya ingin menghapus data barang dengan nama

“Hanco Strawberry” maka klik *hyperlink* “Hapus” berdasarkan kolom nama yang memiliki nama “Hanco Strawberry” kemudian akan tampil konfirmasi dialog sebagai berikut:



Gambar 4.138 Konfirmasi Hapus Data Barang

- b. Klik *button* “OK” untuk menghapus data barang atau klik *button* “Cancel” untuk membatalkan hapus data barang.
- c. Ketika *button* “OK” diklik dan menampilkan notifikasi “Data barang telah berhasil dihapus” berarti data barang dengan nama “Hanco Strawberry” telah berhasil dihapus tetapi jika menampilkan pesan kesalahan “tidak dapat menghapus data barang” berarti data master barang dengan nama tersebut telah digunakan untuk transaksi diantaranya penjualan, pesanan penjualan, pembelian, produksi dan lain-lain contohnya seperti berikut:



Gambar 4.139 Pesan Kesalahan Hapus Data Barang

4. Refresh data barang, button “Segarkan” digunakan untuk memuat ulang atau melakukan singkronisasi data aplikasi dengan data yang ada pada tabel barang di basisdata.
5. Cetak data barang, button “Cetak” digunakan untuk mencetak daftar data master barang contohnya seperti berikut ketika button “Cetak” diklik:

Kode	Nama Barang	Kategori	Satuan	Stok	Harga Beli	Harga Jual	Gud
1-ice-choco	Choco	Ice Cream	pcs	0	Rp0	Rp800	CENTRAL
1-ice-vanies	Vanies	Ice Cream	kg	0	Rp0	Rp1.400	CENTRAL
1-b_baku-hanco	Hanco Vanilla	Bahan Baku	pcs	0	Rp65.000	Rp60.000	CENTRAL
1-b_baku-hanco	Hanco Coklat	Bahan Baku	pcs	0	Rp62.000	Rp60.500	CENTRAL
1-kemasan-tutup	Tutup Cup	Kemasan	pcs	0	Rp50	Rp40	CENTRAL
1-kemasan-corn	Corn	Kemasan	pcs	0	Rp350	Rp350	CENTRAL
1-kemasan-ball	Ball	Kemasan	pcs	0	Rp350	Rp350	CENTRAL
1-kemasan-stick	Stick	Kemasan	pcs	0	Rp150	Rp150	CENTRAL

Gambar 4.140 Cetak Daftar Data Barang

g. Data master konsumen

Untuk menggunakan data master konsumen, pilih “Daftar konsumen” pada menu “Data utama” setelah itu pada *center content* akan menampilkan *form* sebagai berikut:

Kode	Kota	Kontak Personal	Nama	Aksi
1	-	-	dimas	<input checked="" type="checkbox"/> Ubah <input type="checkbox"/> Hapus
2	-	082117355133	dimas maryanto	<input checked="" type="checkbox"/> Ubah <input type="checkbox"/> Hapus

Spesifikasi Konsumen

 Nama:
 Kontak Personal:
 Alamat lengkap:
 Kota:
 Kelurahan:
 Kode Pos:
 RW:
 RT:

Gambar 4.141 Daftar Data Konsumen

dalam data master konsumen terdapat 3 proses utama yaitu tambah, ubah dan hapus data konsumen. Berikut ini adalah cara tambah, ubah dan hapus data konsumen:

1. Tambah data konsumen, berikut ini adalah langkah-langkah untuk menambahkan data konsumen:
 - a. Klik *button* “Data baru” kemudian akan tampil *form input* data konsumen seperti berikut:

Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

Administrator
(admin) Administrator

Keluar

data konsumen

Nama: Nama konsumen

Kontak Personal: Kontak personal

Kota: Kota

Kelurahan: Kelurahan

Alamat lengkap: Alamat lengkap

Kode Pos: 0

RW: 0

RT: 0

Simpan **Batal**

Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

Gambar 4.142 Form Input Data Konsumen Baru

- b. Tampilan awal *button* “Simpan” tidak aktif, untuk mengaktifkannya berikut ini adalah aturannya:
 - i. Nama konsumen tidak boleh kosong
 - ii. Nama konsumen harus diisi minimal 1 dan maksimal 100 karakter.
 - iii. Kontak personal tidak boleh kosong dan hanya dapat diisi dengan angka 0-9 dan karakter “-“.
 - iv. Data lain-lain boleh dikosongkan.
- c. Setelah aturan tersebut terpenuhi maka secara otomatis *button* “Simpan” akan aktif.
- d. Kemudian klik *button* “Simpan” untuk menambahkan konsumen baru Dan akan menampilkan notifikasi “Data konsumen berhasil ditambahkan”.

2. Ubah informasi konsumen, berikut ini adalah langkah-langkah untuk mengubah data informasi kosumen.
- Klik *hyperlink* “Ubah” pada *table-view* konsumen di kolom aksi dan baris tertentu. Contohnya ingin mengubah data informasi konsumen dengan nama “dimas maryanto” maka klik *hyperlink* “Ubah” berdasarkan dengan nama tersebut kemudian akan tampil *form edit* data konsumen seperti berikut:

The screenshot shows a Windows-style application window titled "Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream". On the left is a vertical menu titled "Menu utama" with several options under "Data Utama": Daftar jabatan, Daftar karyawan, Daftar kategori barang, Daftar satuan barang, Daftar gudang, Daftar barang, Daftar konsumen, Daftar pemasok, and Daftar kategori pembayaran. The main content area has a header "Selamat Datang" and "Administrator (admin) Administrator Keluar". Below it, a section titled "data konsumen" contains a form with the following data:

Nama	dimas maryanto
Kontak Personal	082117355133
Kota	-
Kelurahan	-
Alamat lengkap	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8
Kode Pos	40426
RW	18
RT	6
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

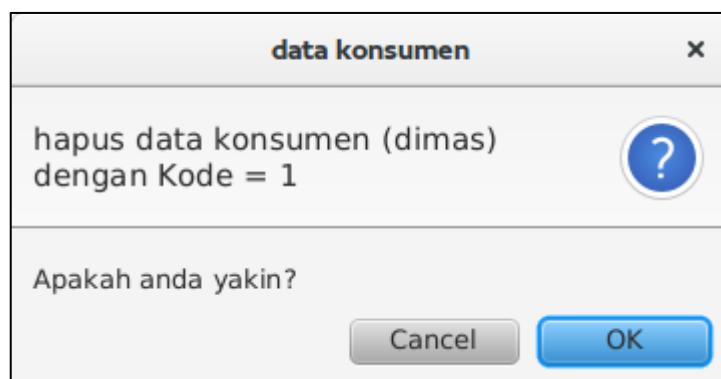
At the bottom right of the form, there is a small note: "Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016".

Gambar 4.143 Form Edit Data Konsumen

- Aturan penggunaan *form edit* data kosumen sama seperti *form input* data konsumen.
- Setelah data diubah kemudian klik *button* “Simpan” untuk menyimpan perubahan data informasi konsumen, jika berhasil maka akan menampilkan notifikasi “Data konsumen berhasil diubah” dan kemudian anda akan diarahkan kembali ke menu “Daftar konsumen”.

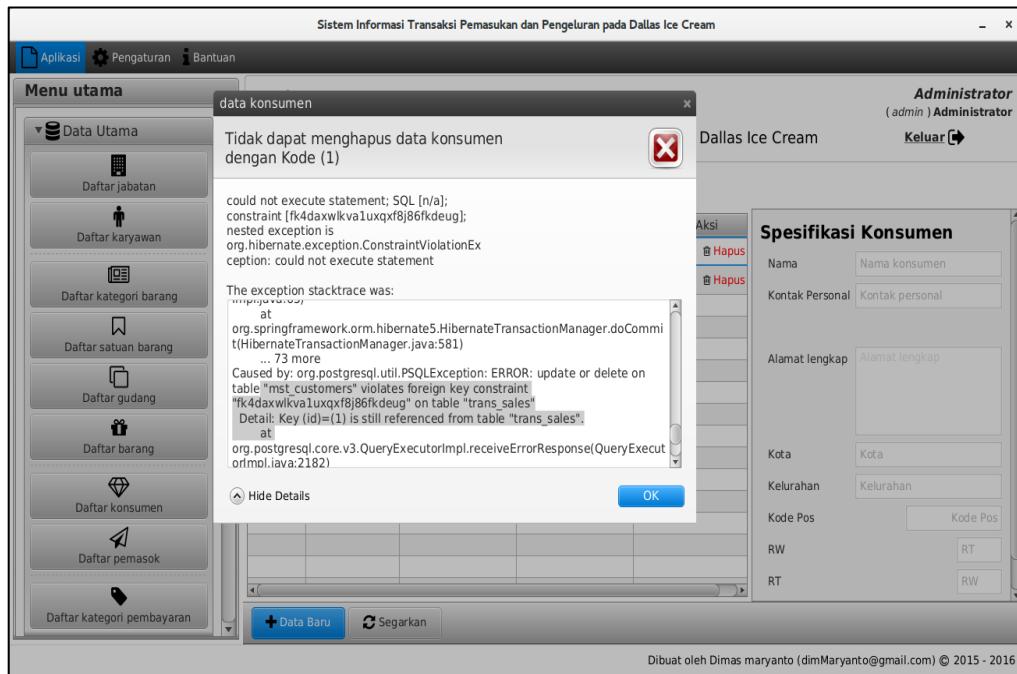
3. Hapus data konsumen, berikut ini adalah langkah-langkah untuk menghapus data konsumen:

- a. Klik *hyperlink* “Hapus” pada *table-view* di kolom aksi dan baris tertentu. Contohnya ingin menghapus data konsumen dengan nama “dimas maryanto” maka klik *hyperlink* “Hapus” berdasarkan nama tersebut kemudian akan tampil konfirmasi dialog seperti berikut:



Gambar 4.144 Konfirmasi Hapus Data Konsumen

- b. Klik *button* “OK” untuk menghapus data konsumen dengan nama dimas maryanto atau klik *button* “Cancel” untuk membatalkan untuk menghapus data konsumen dengan nama tersebut.
- c. Ketika *button* “OK” diklik dan menampilkan notifikasi “Data konsumen berhasil dihapus” berarti data konsumen dengan nama dimas maryanto telah dihapus tetapi jika menampilkan pesan kesalahan “Tidak dapat menghapus data kosumen” berarti data kosumen telah digunakan untuk transaksi pemesanan penjualan dan penjualan produk contohnya seperti berikut:



Gambar 4.145 Pesan Kesalahan Hapus Data Konsumen

4. Refresh data konsumen, button “Segarkan” digunakan untuk memuat ulang atau melakukan singkronisasi data aplikasi dengan data konsumen yang ada tabel kosumen di basis data.

h. Data master pemasok

Untuk menggunakan data master pemasok, pilih “Daftar pemasok” pada menu “Data utama” setelah itu pada *center content* akan menampilkan *form* sebagai berikut:

The screenshot shows the main interface of the "Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream". The top navigation bar includes links for "Aplikasi", "Pengaturan", and "Bantuan". The title "Selamat Datang" and subtitle "Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream" are displayed prominently. On the left, a sidebar menu lists various modules: Data Utama, Daftar jabatan, Daftar karyawan, Daftar kategori barang, Daftar satuan barang, Daftar gudang, Daftar barang, Daftar konsumen, Daftar pemasok, and Daftar kategori pembayaran. The central content area is titled "Daftar pemasok" and displays a table with columns: Kode, Kontak Personal, and Aksi. One row is visible, showing "supermarket" in the Name column with "Ubah" and "Hapus" buttons. Below the table are "Data Baru" and "Segarkan" buttons. To the right, a panel titled "Spesifikasi Pemasok" contains input fields for Name, Contact Person, Address, City, Subdistrict, Postcode, RW, and RT. The top right corner shows the administrator status "(admin) Administrator" and a "Keluar" button.

Gambar 4.146 Daftar Data Pemasok

dalam data master pemasok terdapat 3 proses utama yaitu tambah, ubah dan hapus data konsumen. Berikut ini adalah cara tambah, ubah dan hapus data konsumen:

1. Tambah data pemasok, berikut ini adalah langkah-langkah untuk menambahkan data pemasok:
 - a. Klik *button* “Data baru” kemudian akan tampil *form input* data pemasok seperti berikut:

The screenshot shows a Windows application window titled "Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream". The main title bar also displays the same text. In the top right corner, there is an "Administrator" status with "(admin)" and a "Keluar" button. The left side features a vertical "Menu utama" (Main Menu) with several items under "Data Utama": Daftar jabatan, Daftar karyawan, Daftar kategori barang, Daftar satuan barang, Daftar gudang, Daftar barang, Daftar konsumen, Daftar pemasok, and Daftar kategori pembayaran. The main content area is titled "Selamat Datang" and "data pemasok". It contains a form with the following fields: Nama (Nama pemasok), Kontak Personal (Contact personal), Kota (Kota), Kelurahan (Kelurahan), RT / RW (0, 0), Kode Pos (0), and Alamat lengkap (Alamat lengkap). At the bottom of the form are two buttons: "Simpan" (Save) and "Batal" (Cancel). The footer of the window includes the text "Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016".

Gambar 4.147 Form Input Data Pemasok

- b. Tampilan awal *button* “Simpan” tidak aktif, untuk mengaktifkannya berikut ini adalah aturannya:
 - i. Nama pemasok tidak boleh kosong
 - ii. Nama pemasok harus diisi minimal 1 dan maksimal 100 karakter.
 - iii. Kontak personal tidak boleh kosong dan hanya dapat diisi dengan angka 0-9 dan karakter “-“.
 - iv. Data lain-lain boleh dikosongkan.
- c. Setelah aturan tersebut terpenuhi maka secara otomatis *button* “Simpan” akan aktif.
- d. Kemudian klik *button* “Simpan” untuk menambahkan pemasok baru. Dan akan menampilkan notifikasi “Data pemasok berhasil ditambahkan”.

2. Ubah informasi pemasok, berikut ini adalah langkah-langkah untuk mengubah data informasi pemasok.
- Klik *hyperlink* “Ubah” pada *table-view* pemasok di kolom aksi dan baris tertentu. Contohnya ingin mengubah data informasi pemasok dengan nama “supermarket” maka klik *hyperlink* “Ubah” berdasarkan dengan nama tersebut kemudian akan tampil *form edit* data pemasok seperti berikut:

The screenshot shows a Windows application window titled "Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream". The menu bar includes "Applikasi", "Pengaturan", and "Bantuan". The top right corner shows the administrator status "(admin) Administrator" and a "Keluar" (Logout) button. On the left, a sidebar titled "Menu utama" lists various data management modules: Data Utama (Daftar jabatan, Daftar karyawan, Daftar kategori barang, Daftar satuan barang, Daftar gudang, Daftar barang, Daftar konsumen, Daftar pemasok, Daftar kategori pembayaran). The main content area displays a "Selamat Datang" message and the title "data pemasok". The form contains the following fields:

Nama	supermarket
Kontak Personal	01248094802140
Kota	-
Kelurahan	-
RT / RW	0
Kode Pos	0
Alamat lengkap	[Large text input field]

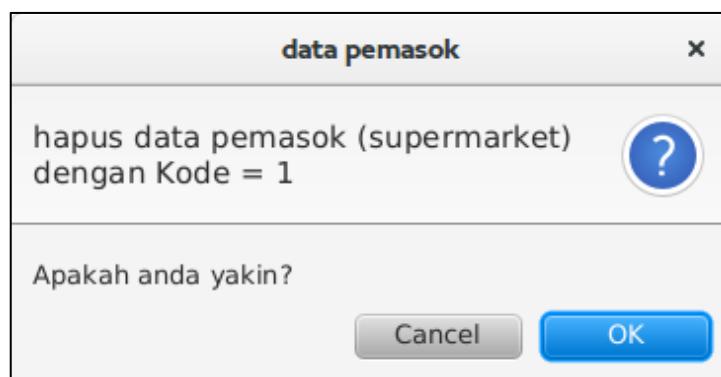
At the bottom of the form are two buttons: "Simpan" (Save) and "Batal" (Cancel).

Gambar 4.148 Form Edit Data Pemasok

- Aturan penggunaan *form edit* data kosumen sama seperti *form input* data konsumen.
- Setelah data diubah kemudian klik *button* “Simpan” untuk menyimpan perubahan data informasi pemasok, jika berhasil maka akan menampilkan notifikasi “Data pemasok berhasil diubah” dan kemudian anda akan diarahkan kembali ke menu “Daftar pemasok”.

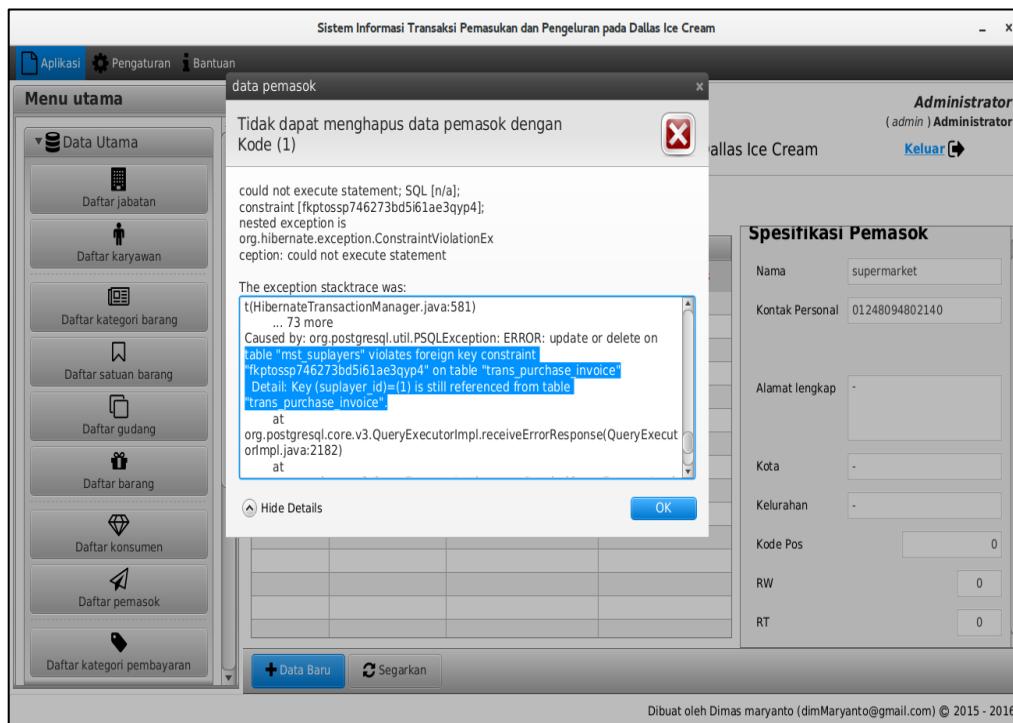
3. Hapus data pemasok, berikut ini adalah langkah-langkah untuk menghapus data pemasok:

- a. Klik *hyperlink* “Hapus” pada *table-view* di kolom aksi dan baris tertentu. Contohnya ingin menghapus data pemasok dengan nama supermarket maka klik *hyperlink* “Hapus” berdasarkan nama tersebut kemudian akan tampil konfirmasi dialog seperti berikut:



Gambar 4.149 Konfirmasi Hapus Data Pemasok

- b. Klik *button* “OK” untuk menghapus data pemasok dengan nama “supermarket” atau klik *button* “Cancel” untuk membatalkan untuk menghapus data pemasok dengan nama tersebut.
- c. Ketika *button* “OK” diklik dan menampilkan notifikasi “Data pemasok berhasil dihapus” berarti data pemasok dengan nama “supermarket” telah dihapus tetapi jika menampilkan pesan kesalahan “Tidak dapat menghapus data pemasok” berarti data pemasok telah digunakan untuk transaksi pemesanan pembelian bahan baku contohnya seperti berikut:

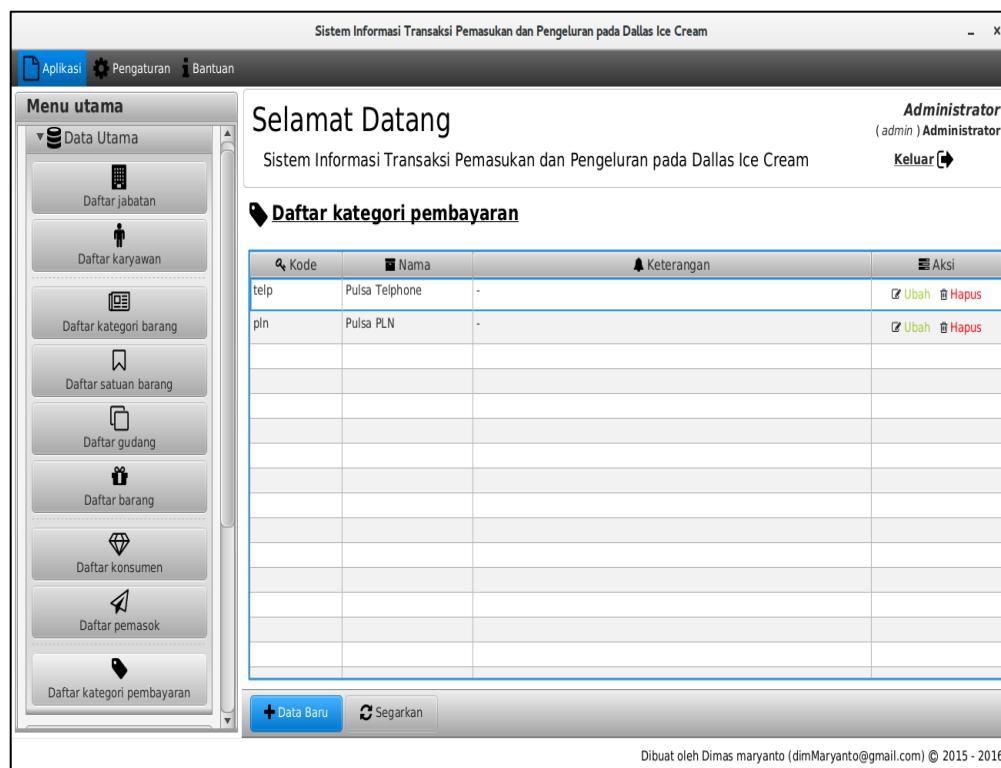


Gambar 4.150 Pesan Kesalahan Hapus Data Pemasok

4. Refresh data pemasok, button “Segarkan” digunakan untuk memuat ulang atau melakukan singkronisasi data aplikasi dengan data pemasok yang ada tabel pemasok di basis data.

i. Data master kategori pembayaran

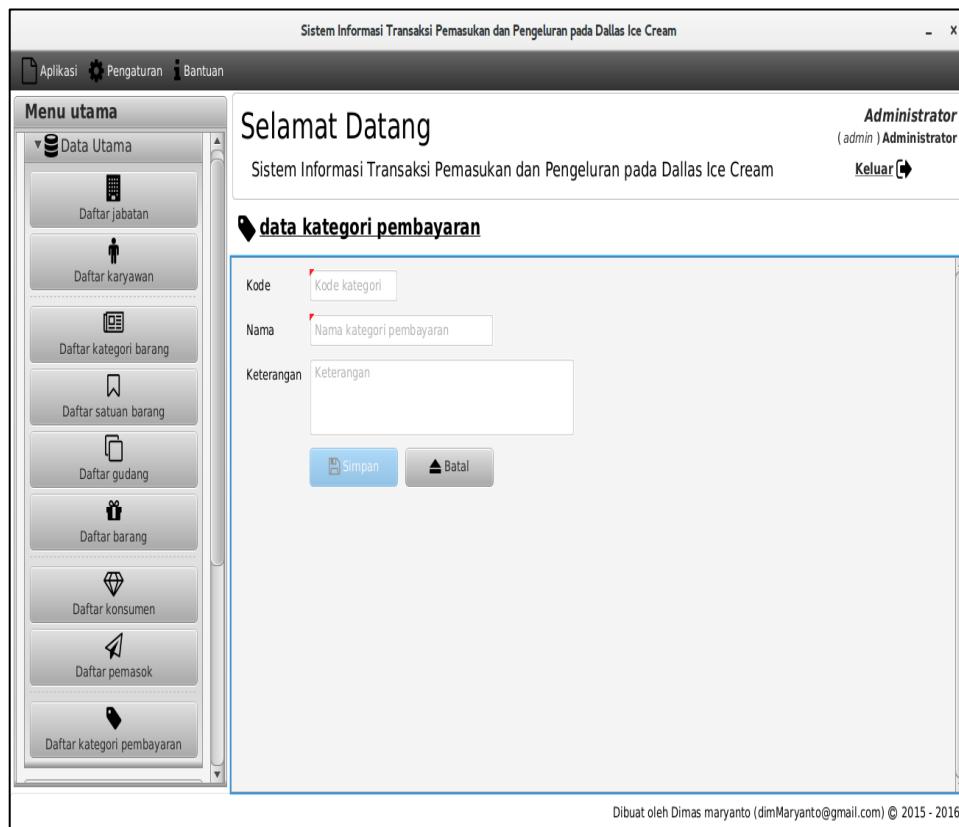
Untuk menggunakan data master kategori pembayaran pilih *menu item* “Daftar kategori pembayaran” pada menu “Data Utama” maka akan menampilkan *form* pada *center content* sebagai berikut:



Gambar 4.151 Daftar Data Kategori Pembayaran

data master kategori barang terdiri dari 3 proses utama yaitu tambah, ubah, hapus seperti berikut ini:

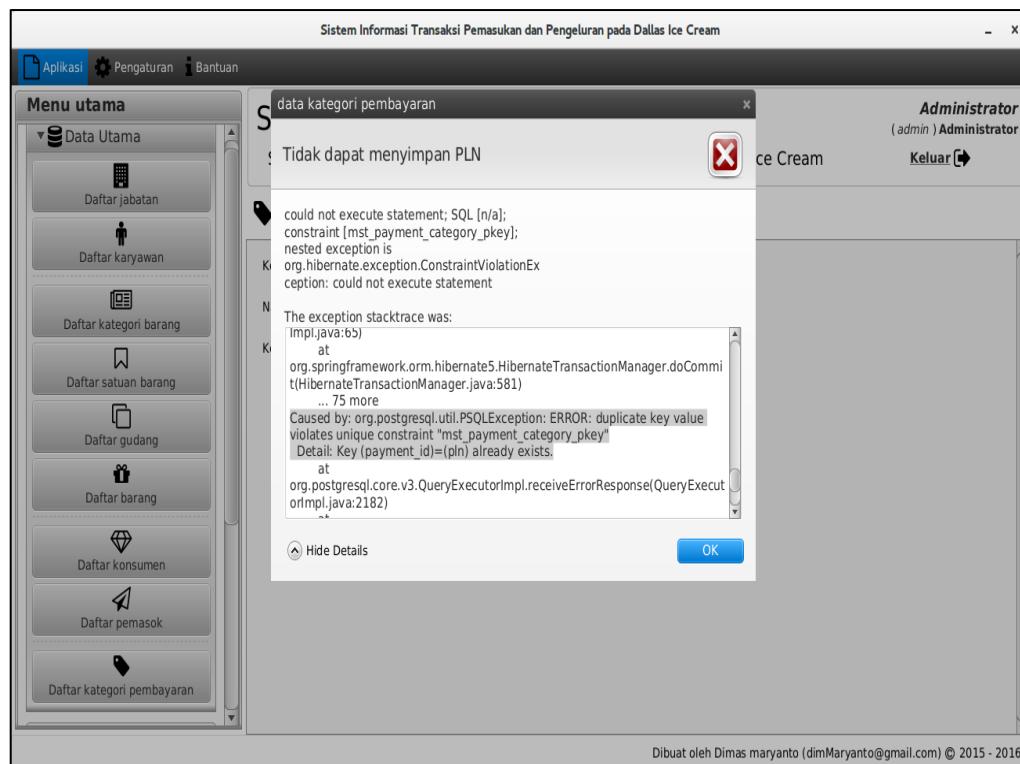
1. Tambah data kategori pembayaran, berikut ini adalah langkah-langkah untuk menambahkan data kategori pembayaran:
 - a. klik button “Data baru” setelah itu akan tampil *form* input data kategori pembayaran seperti berikut:



Gambar 4.152 Form Input Data Kategori Pembayaran Baru

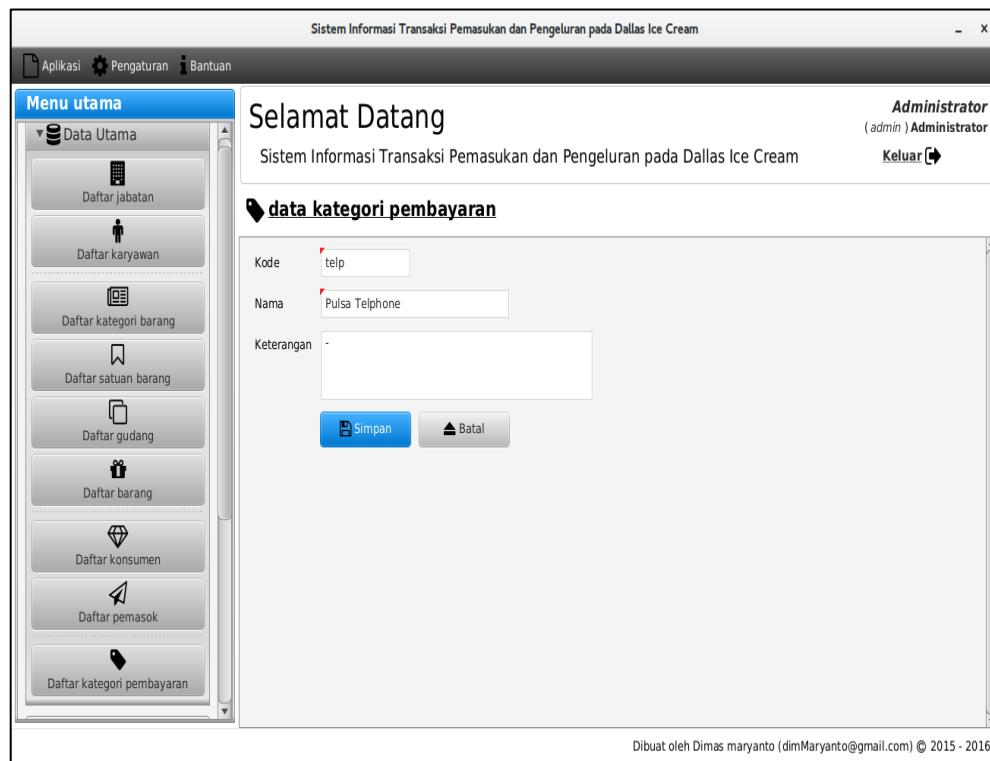
- b. Tampilan awal *button* “Simpan” tidak aktif, karena ada beberapa ketentuan yang harus dipenuhi yaitu sebagai berikut:
 - i. Kode tidak boleh kosong, minimal harus diisi 1 karakter, batas maksimal 100 karakter dan harus bersifat *unique*.
 - ii. Nama kategori tidak boleh kosong, minimal harus diisi dengan 1 karakter dan batas maksimal 100 karakter
 - iii. Keterangan boleh tidak diisi.
- c. Setelah semua ketentuan tersebut terpenuhi maka *button* “Simpan” akan aktif. Kemudian klik *button* “Simpan”

- d. Jika data berhasil disimpan maka akan menampilkan notifikasi “Data kategori pembayaran berhasil disimpan” tetapi jika tidak maka akan menampilkan pesan kesalahan seperti berikut:



Gambar 4.153 Pesan Kesalahan Input Data Kategori Pembayaran

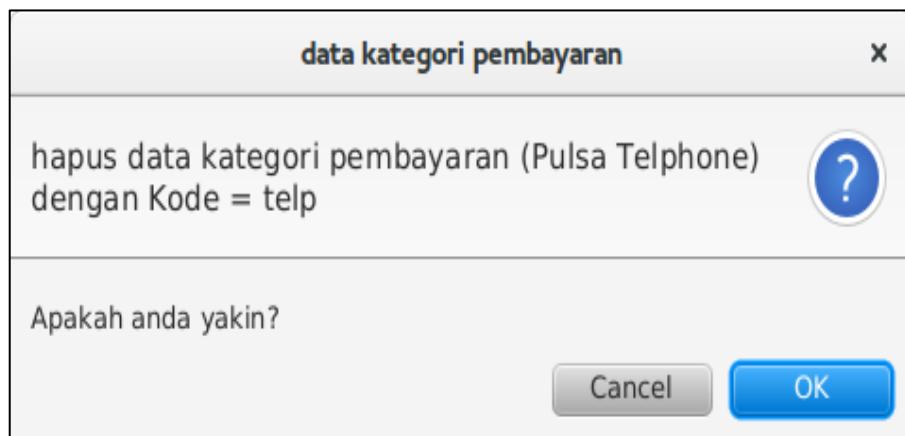
2. Ubah data kategori pembayaran, berikut ini adalah langkah-langkah untuk mengubah informasi data kategori pembayaran:
 - a. klik *hyperlink* “Ubah” pada kolom aksi dan pada baris tertentu. Contohnya ingin mengubah kategori pembayaran dengan kode “telp” caranya adalah klik *hyperlink* “Ubah” berdasarkan kode tersebut, kemudian akan muncul *form* kategori pembayaran sebagai berikut:



Gambar 4.154 Form Edit Data Kategori Pembayaran

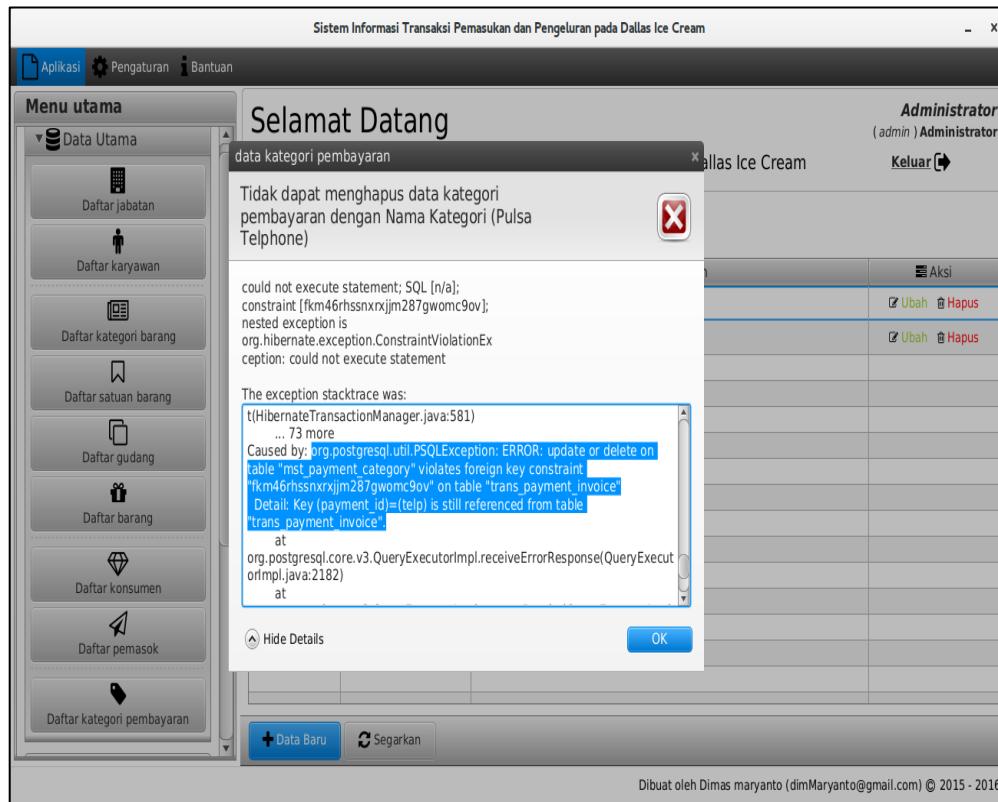
- b. Aturan menggunakan *form edit* kategori barang adalah sebagai berikut:
 - i. Kode tidak dapat diubah.
 - ii. Nama kategori harus diisi minimal 1 dan maksimal 100 karakter.
 - iii. Keterangan boleh tidak diisi.
- c. Setelah data diubah maka klik *button* “Simpan” untuk menyimpan perubahan data yang terjadi. Jika berhasil maka akan menampilkan notifikasi “Data kategori pembayaran berhasil diubah” dan anda akan diarahkan kembali ke *menu* “Daftar kategori pembayaran”.
3. Hapus data kategori pembayaran
 - a. klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu. Contohnya ingin menghapus kategori barang dengan kode “telp”, caranya adalah

klik *hyperlink* “Hapus” berdasarkan kode tersebut kemudian akan tampil dialog konfirmasi sebagai berikut:



Gambar 4.155 Konfirmasi Hapus Data Kategori Pembayaran

- b. Setelah itu klik *button* “Ok” untuk menghapus data kategori pembayaran dengan kode “telp”.
- c. Jika berhasil dihapus maka akan tampil notifikasi “Data kategori pembayaran berhasil dihapus”, jika gagal maka akan tampil pesan kesalahan yang mungkin disebabkan oleh data kategori pembayaran telah digunakan seperti berikut:

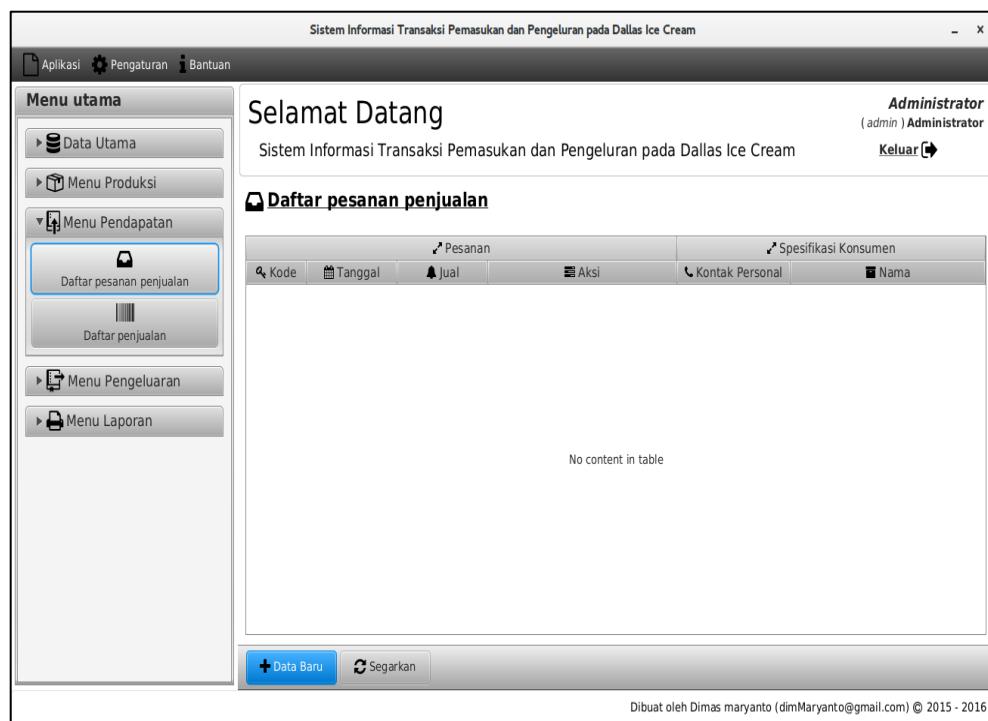


Gambar 4.156 Pesan Kesalahan Untuk Hapus Data Kategori Pembayaran

4. *Refresh* data kategori pembayaran, button “Segarkan” digunakan untuk memuat ulang atau singkronisasi dengan data yang terdapat di basis data.

4.5.6.3. Penggunaan program pemesanan penjualan

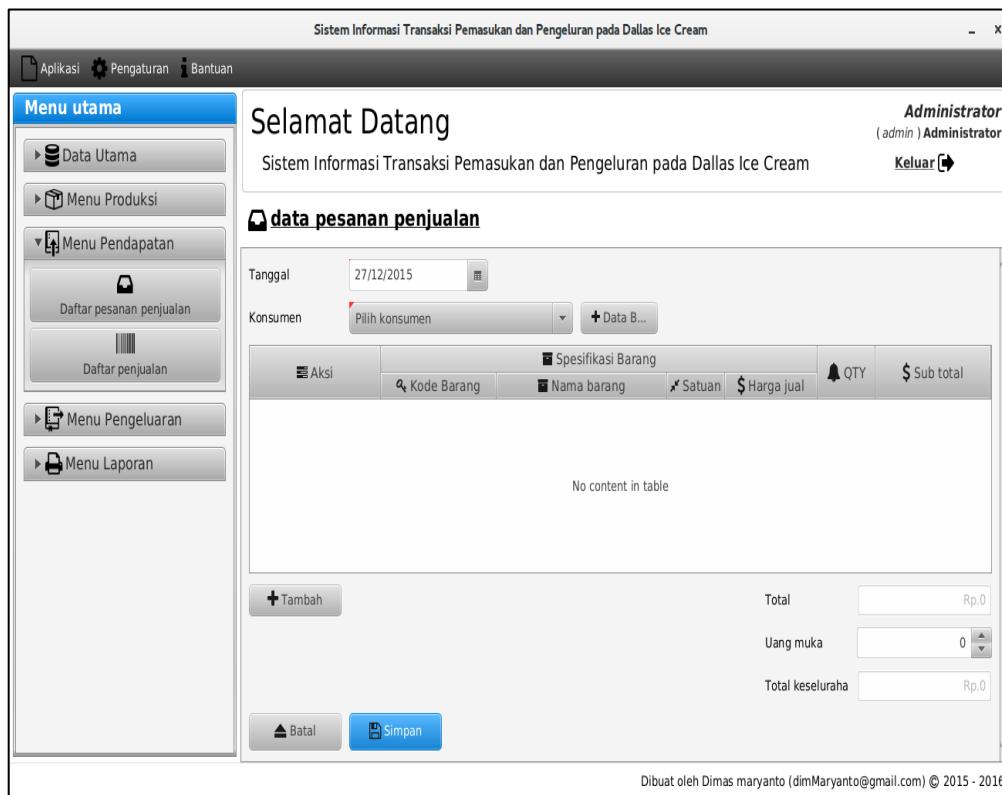
Untuk menggunakan data transaksi pemesanan penjualan, pilih “Daftar pesanan penjualan” pada menu “Menu pendapatan” setelah itu pada *center content* akan menampilkan *form* sebagai berikut:



Gambar 4.157 Daftar Data Pemesanan Penjualan

Dalam data transaksi pemesanan penjualan terdapat 3 proses utama yaitu tambah, ubah, hapus, dan jual. Berikut ini adalah cara tambah, ubah, hapus, jual pesanan penjualan:

1. Tambah data pesanan penjualan, berikut ini adalah langkah-langkah untuk mencatat pesanan penjualan:
 - a. Klik button “Data Baru” kemudian akan tampil *form input* pesanan penjualan seperti berikut:



Gambar 4.158 Form Input Data Pesanan Penjualan

- b. Tampilan awal button “Simpan” tidak dapat diklik, untuk mengaktifkanya berikut ini adalah aturannya:
- i. Pilih konsumen, jika data konsumen masih kosong maka tambahkan konsumen caranya adalah klik button “Data baru” yang ada disamping combobox konsumen kemudian akan muncul *form input data* konsumen seperti berikut:

data konsumen

Nama: [Text Input]

Kontak Personal: [Text Input] Kontak personal

Alamat lengkap: [Text Input]

RT / RW: [Number Input] 0 [Up Arrow] [Number Input] 0 [Up Arrow]

Simpan **Batal**

Gambar 4.159 Form Input Data Konsumen Dari Pesanan Penjualan

- ii. Pilih tanggal pesanan.
- c. Klik button “Tambah” yang letaknya di bawah tabel barang, untuk menambahkan barang yang dipesan dan berserta jumlahnya. Berikut ini adalah *form input data* barang pesanan penjualan:

Kode	Nama barang
1-kemasan-stick	Stick
1-kemasan-ball	Ball
1-kemasan-corn	Corn
1-kemasan-tutup cup	Tutup Cup
1-b_baku-hanco str...	Hanco Strawberry
1-b_baku-hanco co...	Hanco Coklat
1-b_baku-hanco va...	Hanco Vanila
1-ice-cup kecil 70ml	Cup Kecil 70ml
1-ice-galon	Galon
1-ice-cup besar 10...	Cup Besar 100ml

Harga jual: [Number Input] 0 [Up Arrow]
Jumlah: [Number Input] 0 [Up Arrow]

Spesifikasi Barang

Kode: [Text Input] Kode Barang
Nama: [Text Input] Nama barang
Kategori: [Text Input] Kategori barang
Lokasi barang: [Text Input] Nama gudang
Satuan: [Text Input] Satuan barang
QTY: [Text Input] QTY

Tambah **Tutup**

Gambar 4.160 Form Pilih Barang Pesanan Penjualan

- d. Pada tabel barang pilih salah satu berdasarkan kode atau nama barang tersebut maka pada bagian spesifikasi barang akan menampilkan data dari barang tersebut, contohnya seperti berikut:

Kode	Nama barang
1-kemasan-stick	Stick
1-kemasan-ball	Ball
1-kemasan-corn	Corn
1-kemasan-tutup cup	Tutup Cup
1-b_baku-hanco str...	Hanco Strayberry
1-b_baku-hanco co...	Hanco Coklat
1-b_baku-hanco va...	Hanco Vanila
1-ice-cup kecil 70ml	Cup Kecil 70ml
1-ice-galon	Galon
1-ice-cup besar 10...	Cup Besar 100ml

Harga jual Jumlah

Spesifikasi Barang

Kode	<input type="text" value="1-b_baku-hanco strayberry"/>
Nama	<input type="text" value="Hanco Strayberry"/>
Kategori	<input type="text" value="Bahan Baku"/>
Lokasi barang	<input type="text" value="CENTRAL"/>
Satuan	<input type="text" value="pcs"/>
QTY	<input type="text" value="10"/>

+ Tambah ✖ Tutup

Gambar 4.161 Form Pilih Barang Yang Dipilih Untuk Pesanan Penjualan

- e. Kemudian klik button “Tambah” untuk menambahkan barang yang dipilih ke tabel pemesanan, berikut ini adalah sample data yang telah diinputkan:

Tanggal Konsumen + Data B...

Aksi	Kode Barang	Nama barang	Satuan	Harga jual	QTY	Sub total
Ubah	1-b_baku-hanco str...	Hanco Strayberry	pcs	62500.0	15	937500.0
Ubah	1-kemasan-tutup cup	Tutup Cup	pcs	40.0	10	400.0
Ubah	1-ice-galon	Galon	pcs	75000.0	1	75000.0

+ Tambah Total
+ Batal Uang muka
Simpan Total keseluruhna

Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

Gambar 4.162 Form Input Data Contoh Pesanan Penjualan

f. Jika ingin merubah jumlah barang yang dipesan caranya adalah klik *hyperlink* “Ubah” pada kolom aksi berdasarkan barangnya maka akan menampilkan *form* pilih barang seperti berikut:

Kode	Nama barang
1-kemasan-stick	Stick
1-kemasan-ball	Ball
1-kemasan-corn	Corn
1-kemasan-tutup cup	Tutup Cup
1-b_baku-hanco str...	Hanco Strawberry
1-b_baku-hanco co...	Hanco Coklat
1-b_baku-hanco va...	Hanco Vanila
1-ice-cup kecil 70ml	Cup Kecil 70ml
1-ice-galon	Galon
1-ice-cup besar 10...	Cup Besar 100ml

Harga jual 62500
Jumlah 10

Spesifikasi Barang

Kode 1-b_baku-hanco strawberry
Nama Hanco Strawberry
Kategori Bahan Baku
Lokasi barang CENTRAL
Satuan pcs
QTY 10

Aksi **Spesifikasi Barang**

Aksi **Spesifikasi Barang**

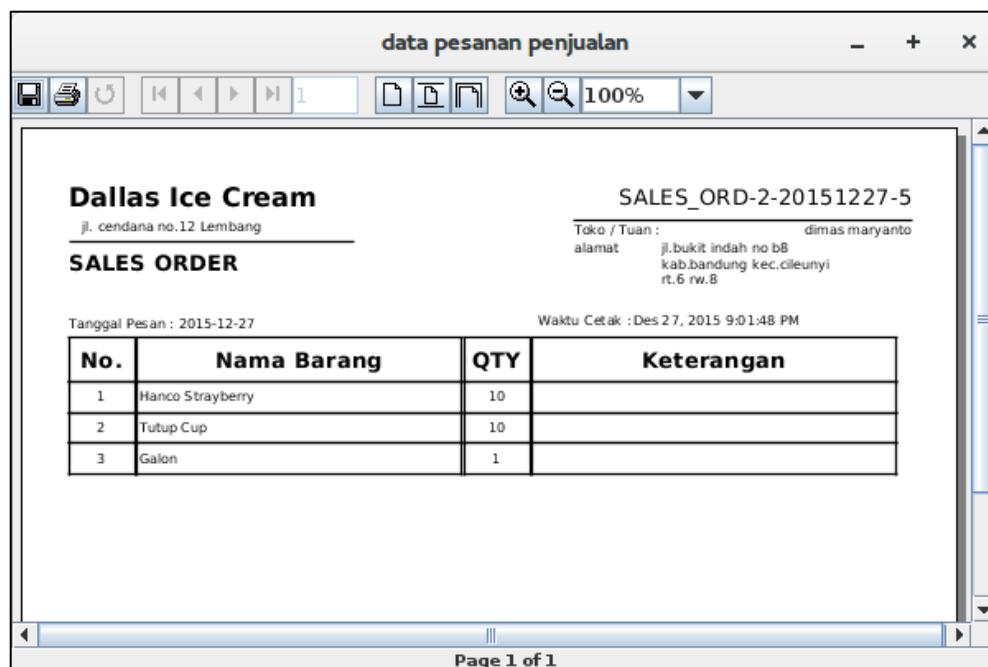
Gambar 4.163 Form Edit Qty Barang Pesanan Penjualan

g. Kemudian ubah qtnya setelah itu klik *button* “Tambah” untuk menyimpan perubahan barang, berikut ini adalah hasilnya setelah ditambahkan:

Aksi	Spesifikasi Barang				QTY	Sub total
	Kode Barang	Nama barang	Satuan	Harga jual		
Ubah Hapus	1-b_baku-hanco str...	Hanco Strawberry	pcs	62500.0	10	625000.0
Ubah Hapus	1-kemasan-tutup cup	Tutup Cup	pcs	40.0	10	400.0
Ubah Hapus	1-ice-galon	Galon	pcs	75000.0	1	75000.0

Gambar 4.164 Form Input Pesanan Penjualan Setelah Di-Edit

- h. Input uang muka jika terjadi pembayaran contohnya seperti berikut:
- i. Kemudian klik button “Simpan” jika berhasil disimpan akan menampilkan notifikasi “Data pesanan penjualan berhasil disimpan” dan menampilkan surat pesanan penjualan seperti berikut:



Gambar 4.165 Surat Keterangan Pesanan Penjualan

- j. Tetapi jika tidak maka akan menampilkan pesan kesalahan “Tidak dapat menyimpan pesanan penjualan”
2. Ubah data pesanan penjualan, berikut ini adalah langkah-langkah untuk mengubah data pesanan penjualan:
- a. Berikut ini adalah data yang telah dinputkan:

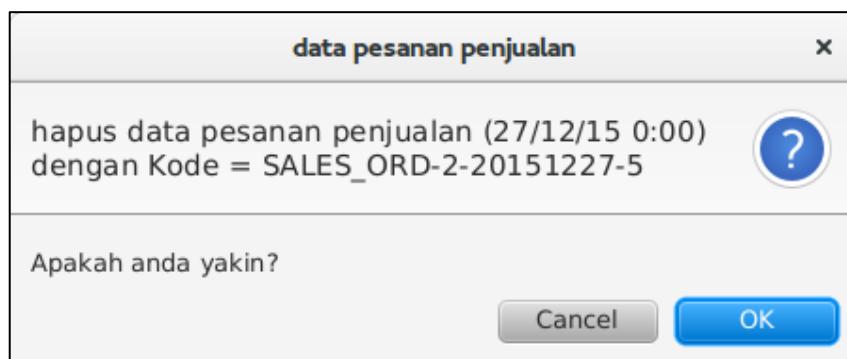
Gambar 4.166 Daftar Data Pesanan Penjualan

- b. Klik *hyperlink* “Ubah” pada kolom aksi dan baris tertentu maka akan menampilkan *form edit* pesanan penjualan seperti berikut:

Gambar 4.167 Form Edit Pesanan Penjualan

- c. Aturan penggunaan *form edit* sama seperti input data pemesanan penjualan.

- d. Kemudian klik *button* “Simpan” jika berhasil disimpan maka akan menampilkan notifikasi “Data pesanan penjualan berhasil diubah” dan menampilkan laporan seperti pada gambar surat keterangan pesanan penjualan diatas.
3. Hapus data pesanan penjualan, berikut ini adalah langkah-langkah untuk menghapus data pesanjualan:
- a. Klik *button* “Hapus” pada kolom aksi dan baris tertentu, kemudian akan menampilkan konfirmasi dialog sebagai berikut:



Gambar 4.168 Konfirmasi Dialog Hapus Data Pesanan Penjualan

- b. Klik *button* “OK” untuk menghapus data pesanan penjualan atau *button* “Cancel” untuk membatalkan data pesanan penjualan tersebut.
4. Jual data pesanan penjualan, berikut ini adalah langkah-langkah untuk menjual data pesanan:
- a. Klik *button* “Jual” pada kolom “Jual” dan pada baris tertentu maka akan menampilkan *form input data penjualan*.
- b. Kemudian klik *button* “Simpan” jika berhasil maka akan menampilkan notifikasi “Data penjualan produk berhasil disimpan”.

5. Lihat data pesanan penjualan, cara melihat data adalah klik *hyperlink* “Lihat” pada kolom aksi dan pada baris tertentu berdasarkan kode pesanan penjualan maka akan menampilkan *form* lihat data pemesanan penjualan seperti berikut:

The screenshot shows the application's main window with a sidebar menu and a central data entry form.

Menu utama:

- Data Utama
- Menu Produksi
- Menu Pendapatan
 - Daftar pesanan penjualan
 - Daftar penjualan
- Menu Pengeluaran
- Menu Laporan

Selamat Datang Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

Administrator (admin) Administrator Keluar

data pesanan penjualan

Tanggal: 27/12/2015

Konsumen: (082117355133) dimas maryanto

Kode Barang	Nama barang	Spesifikasi Barang		QTY	\$ Sub total
		Satuan	Harga jual		
1-b_baku-hanco str...	Hanco Strawberry	pcs	62500.0	10	625000.0
1-kemasan-tutup cup	Tutup Cup	pcs	40.0	10	400.0
1-ice-galon	Galon	pcs	75000.0	1	75000.0
		Total	700400.0		
		Uang muka	50000		
		Total keseluruh	650400.0		

Batal

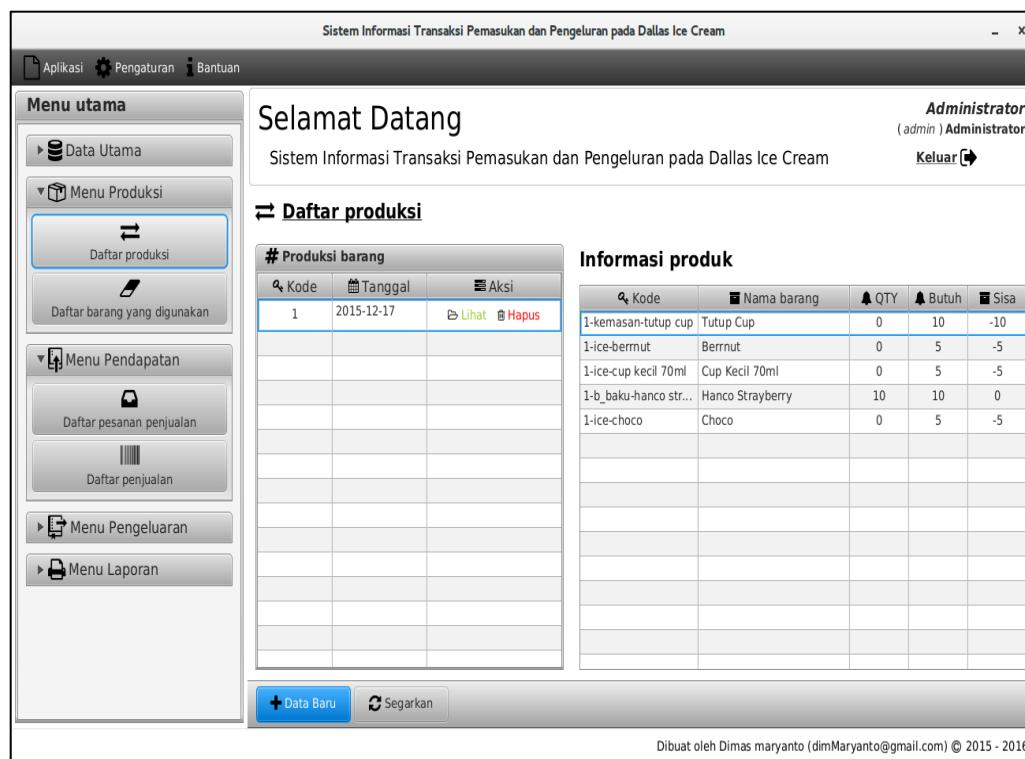
Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

Gambar 4.169 Form Lihat Data Pesanan Penjualan

6. Refresh data pesanan penjualan, *button* “Segarkan” digunakan untuk memuat ulang atau melakukan singkronisasi data pesanan penjualan antara aplikasi dengan basisdata.

4.5.6.4. Penggunaan program produksi

Untuk menggunakan data transaksi produksi, pilih “Daftar produksi” pada menu “Menu produksi” setelah itu pada *center content* akan menampilkan *form* sebagai berikut:



Gambar 4.170 Daftar Data Produksi Es Krim

Dalam data produksi terdapat 4 proses utama yaitu tambah, lihat, hapus dan cetak data produk yang harus segera diproduksi:

1. Tambah data produksi, berikut ini adalah langkah-langkah untuk menambahkan data produksi es krim:
 - a. Klik button “Data baru” kemudian akan menampilkan *form input* data produksi seperti berikut:

Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

The screenshot shows the 'Selamat Datang' (Welcome) page of the system. On the right, it displays 'Administrator (admin) Administrator' and a 'Keluar' (Logout) button. On the left, there's a vertical navigation menu with options like 'Data Utama', 'Menu Produksi' (selected), 'Daftar produksi', 'Daftar barang yang digunakan', 'Menu Pendapatan', 'Daftar pesanan penjualan', 'Daftar penjualan', 'Menu Pengeluaran', and 'Menu Laporan'. The main content area is titled 'data produksi'. It includes fields for 'Nama' (Owner: hari hermawan) and 'Tanggal' (27/12/2015). A table titled 'Spesifikasi Barang' lists various ice cream products with their details. At the bottom are 'Batal' (Cancel) and 'Simpan' (Save) buttons.

Kode	Nama barang	Lokasi barang	Kategori	Satuan	Jumlah
1-ice-cup besar...	Cup Besar 100ml	CENTRAL	Ice Cream	pcs	0
1-ice-galon	Galon	CENTRAL	Ice Cream	pcs	0
1-ice-cup kecil ...	Cup Kecil 70ml	CENTRAL	Ice Cream	pcs	0
1-ice-choco	Choco	CENTRAL	Ice Cream	pcs	0
1-ice-davacho	Davacho	CENTRAL	Ice Cream	kg	0
1-ice-bernut	Berrnut	CENTRAL	Ice Cream	kg	0
1-ice-vanies	Vanies	CENTRAL	Ice Cream	kg	0

Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

Gambar 4.171 Form Input Data Produksi

- b. Pilih karyawan yang melakukan produksi.
- c. Input jumlah barang yang diproduksi pada kolom "Jumlah" dengan cara klik 2 kali atau *double click* setelah itu akan muncul *textfield* baru *input* berapa jumlahnya tetapi minimal nilainya adalah "1" dan jika tidak diproduksi biarkan nilainya "0", contohnya seperti berikut:

Nama	(Owner) hari hermawan				
Tanggal	27/12/2015				
Spesifikasi Barang					
Kode	Nama barang	Lokasi barang	Kategori	Satuan	Jumlah
1-ice-cup besar...	Cup Besar 100ml	CENTRAL	Ice Cream	pcs	15
1-ice-galon	Galon	CENTRAL	Ice Cream	pcs	0
1-ice-cup kecil ...	Cup Kecil 70ml	CENTRAL	Ice Cream	pcs	20
1-ice-choco	Choco	CENTRAL	Ice Cream	pcs	20
1-ice-davacho	Davacho	CENTRAL	Ice Cream	kg	20
1-ice-bermut	Berrnut	CENTRAL	Ice Cream	kg	20
1-ice-vanies	Vanies	CENTRAL	Ice Cream	kg	20

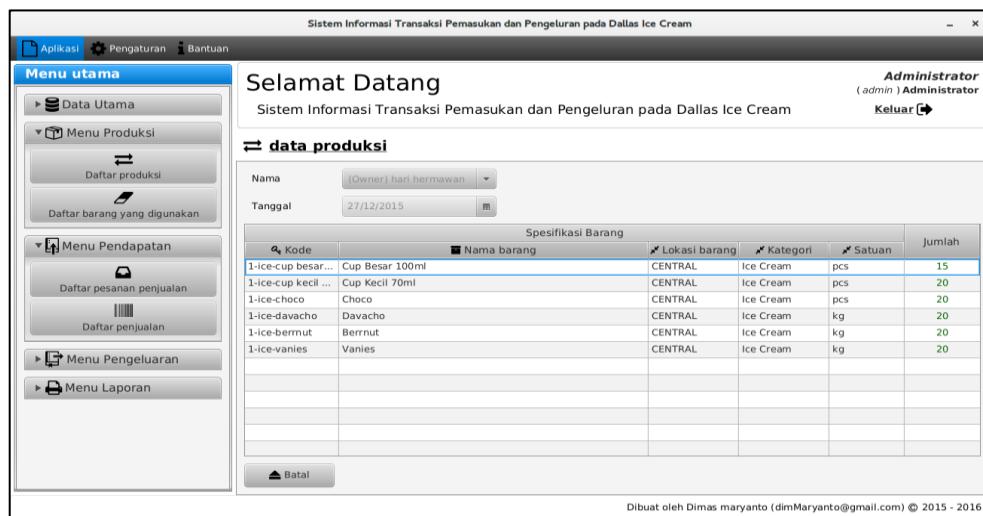
Gambar 4.172 Form Input Data Produksi Diisi

- d. Klik button “Simpan”
 - e. Jika data berhasil disimpan maka akan menampilkan notifikasi “Data produksi berhasil disimpan”
2. Lihat data produksi, berikut ini adalah langkah-langkah untuk melihat *detail* produksi yang telah dilakukan:
- a. Pastikan data telah tersedia seperti berikut ini:

The screenshot shows the SITPP application interface. On the left, there is a sidebar titled "Menu utama" (Main Menu) with several options: "Data Utama", "Menu Produksi" (which is expanded and has "Daftar produksi" selected), "Menu Pendapatan", "Menu Pengeluaran", and "Menu Laporan". On the right, the main content area displays a welcome message "Selamat Datang" and the title "Daftar produksi". Below this is a table titled "# Produksi barang" with columns "Kode", "Tanggal", and "Aksi". A single row is visible with the value "1" in the Kode column and "2015-12-17" in the Tanggal column. There are also "Lihat" and "Hapus" buttons under the Aksi column.

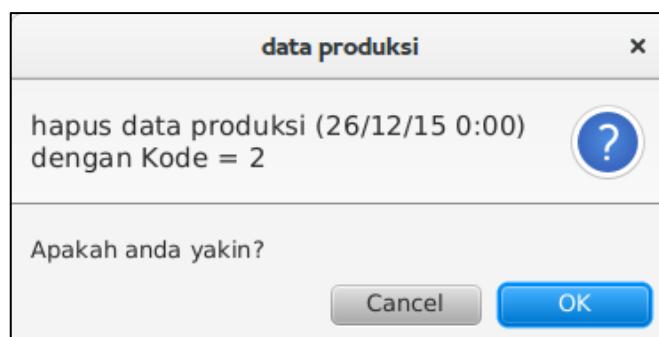
Gambar 4.173 Data Produksi Setelah Input Data

- b. Klik *hyperlink* “Lihat” pada kolom aksi kemudian akan menampilkan data *form* seperti berikut:



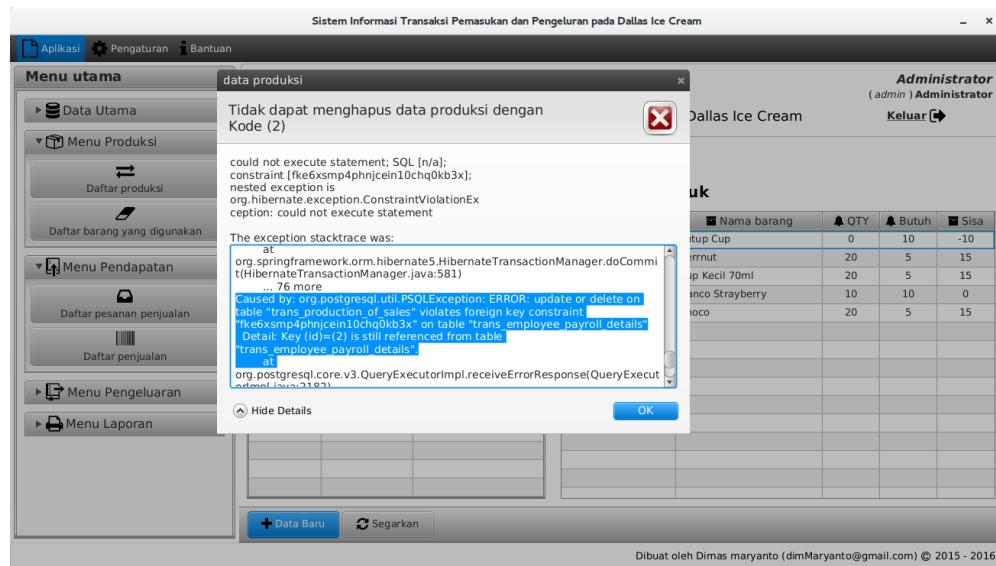
Gambar 4.174 Lihat Data Produksi

3. Hapus data produksi, berikut ini adalah langkah-langkah untuk menghapus data produksi:
- Klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu maka akan menampilkan konfirmasi dialog sebagai berikut:



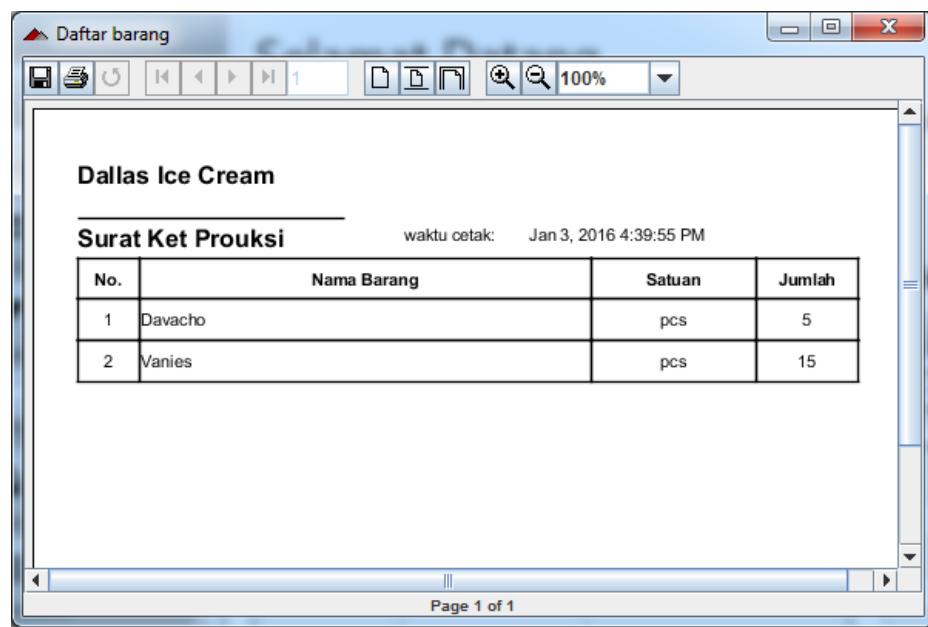
Gambar 4.175 Konfirmasi Dialog Hapus Data Produksi

- b. Jika data sukses dihapus maka akan menampilkan notifikasi “Data produksi telah dihapus” tetapi jika tidak maka akan menampilkan pesan kesalahan seperti berikut:



Gambar 4.176 Pesan Kesalahan Hapus Data Produksi

4. Cetak data produk yang harus diproduksi, button “Cetak” digunakan untuk mencetak data produk yang harus diproduksi berdasarkan pesanan yang masuk berikut ini adalah tampilannya:

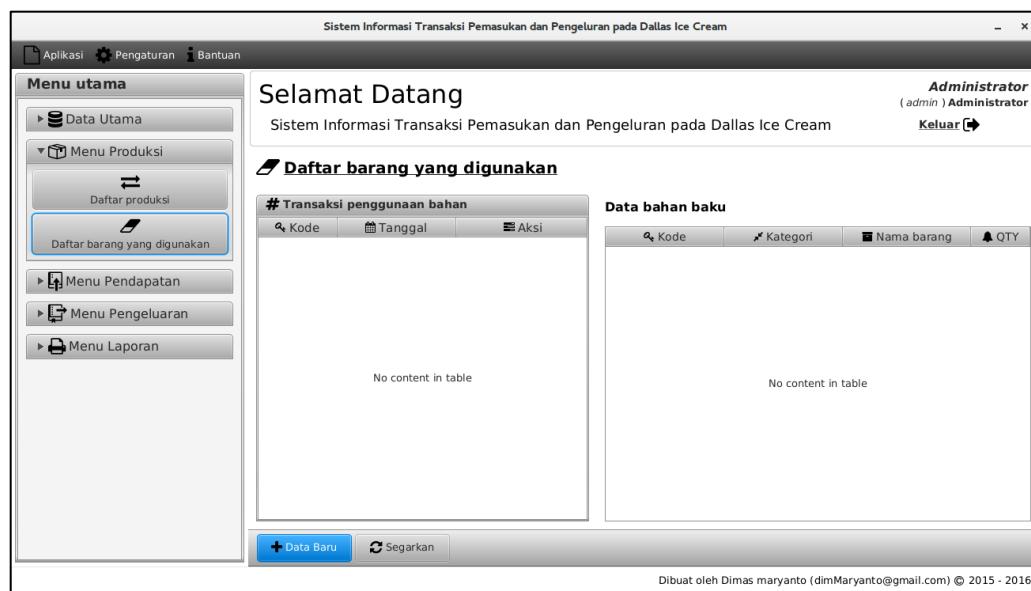


Gambar 4.177 Cetak Data Barang Yang Harus Diproduksi

5. Refresh data produksi, button “Segarkan” digunakan untuk memuat ulang atau melakukan singkronisasi data produksi dan data barang yang harus diproduksi antar aplikasi dan basisdata.

4.5.6.5. Penggunaan program penggunaan bahan

Untuk menggunakan data transaksi produksi, pilih “Daftar penggunaan bahan” pada menu “Menu produksi” setelah itu pada center content akan menampilkan *form* sebagai berikut:



Gambar 4.178 Daftar Data Penggunaan Bahan

Dalam data transaksi penggunaan bahan terdapat 3 proses utama yaitu tambah, lihat, dan hapus data penggunaan bahan:

1. Tambah data penggunaan bahan, berikut ini adalah langkah-langkah untuk menambahkan data penggunaan bahan:
 - a. Klik button “Data baru” kemudian akan menampilkan *form input data* penggunaan bahan seperti berikut:

Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

Administrator
(admin) Administrator
Keluar

data penggunaan barang

Tanggal: 27/12/2015

Spesifikasi Barang					
Kode Barang	Lokasi barang	Kategori	Nama barang	Satuan	QTY
1-b_baku-gula...	CENTRAL	Bahan Baku	Gula putih	pcs	0
1-b_baku-fullcr...	CENTRAL	Bahan Baku	Fullcream	kg	0

Batal Simpan

Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

Gambar 4.179 Form Input Data Pengguanan Bahan

- b. Input jumlah bahan yang digunakan, caranya adalah *double click* pada kolom “Jumlah” berdasarkan baris tertentu kemudian akan tampil *textfield* barulah input jumlahnya contohnya seperti berikut:

Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

Administrator
(admin) Administrator
Keluar

data penggunaan barang

Tanggal: 27/12/2015

Spesifikasi Barang					
Kode Barang	Lokasi barang	Kategori	Nama barang	Satuan	QTY
1-b_baku-gula...	CENTRAL	Bahan Baku	Gula putih	pcs	15
1-b_baku-fullcr...	CENTRAL	Bahan Baku	Fullcream	kg	2

Batal Simpan

Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

Gambar 4.180 Form Input Data Pengguanan Setelah Diisi

- c. Klik button “Simpan” untuk menyimpan data penggunaan bahan
 - d. Jika data berhasil disimpan maka akan menampilkan notifikasi “Data penggunaan bahan berhasil disimpan” tetapi jika tidak akan menampilkan pesan kesalahan seperti berikut “Tidak dapat menyimpan data penggunaan bahan“
2. Lihat data penggunaan bahan, berikut ini adalah cara untuk melihat data penggunaan bahan. Caranya adalah klik *hyperlink* “Lihat” pada kolom aksi dan pada baris tertentu kemudian setelah itu akan menampilkan *form* seperti berikut:

The screenshot shows a web-based application interface for managing inventory and sales at Dallas Ice Cream. The main menu on the left includes 'Data Utama', 'Menu Produksi' (with 'Daftar produksi' and 'Daftar barang yang digunakan'), 'Menu Pendapatan' (with 'Daftar pesanan penjualan' and 'Daftar penjualan'), 'Menu Pengeluaran', and 'Menu Laporan'. The current page is titled 'data penggunaan barang' under the heading 'Selamat Datang'. It displays a table of usage data with the following columns: Kode Barang, Lokasi barang, Kategori, Nama barang, Satuan, and QTY. Two rows are visible:

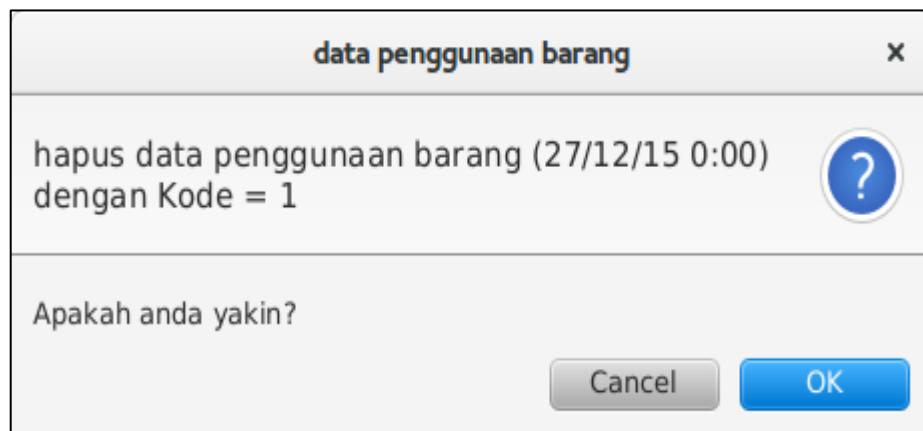
Spesifikasi Barang					
Kode Barang	Lokasi barang	Kategori	Nama barang	Satuan	QTY
1-b_baku-gula ...	CENTRAL	Bahan Baku	Gula putih	pcs	15
1-b_baku-fullcr...	CENTRAL	Bahan Baku	Fulicream	kg	2

At the bottom of the form, there is a 'Batal' (Cancel) button.

Gambar 4.181 Form Lihat Data Penggunaan Bahan

3. Hapus data penggunaan bahan, berikut ini adalah langkah-langkah untuk menghapus data penggunaan bahan:

- a. Klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu kemudian akan menampilkan konfirmasi dialog seperti berikut:



Gambar 4.182 Konfirmasi Dialog Hapus Data Penggunaan Bahan

- b. Kemudian klik *button* “Ok” untuk menghapus data penggunaan bahan atau klik *button* “Cancel” untuk membatalkan hapus data penggunaan bahan.
 - c. Jika data penggunaan bahan berhasil dihapus maka akan menampilkan notifikasi “Data penggunaan bahan berhasil dihapus” tetapi jika tidak akan menampilkan pesan kesalahan seperti berikut “Tidak dapat menghapus data penggunaan bahan”
4. Refresh data penggunaan bahan, *button* “Segarkan” digunakan untuk memuat ulang atau melakukan singkronisasi data penggunaan bahan antara aplikasi dan basisdata

4.5.6.6. Penggunaan program penjualan

Untuk menggunakan data transaksi penjualan, pilih “Daftar penjualan” pada menu “Menu pendapatan” setelah itu pada *center content* akan menampilkan *form* sebagai berikut:

Kode	Tanggal	Aksi	Spesifikasi penjualan			Informasi penjualan			Spesi...
			telah dikirim	telah diteri...	telah lunas	telah kadal...	Tidak		
2	2015-12-22	<input type="checkbox"/> Lihat <input type="checkbox"/> melunasi <input checked="" type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Tidak	dimas
1	2015-12-17	<input type="checkbox"/> Lihat <input type="checkbox"/> melunasi <input checked="" type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Sudah	<input type="checkbox"/> Tidak	dimas
6	2015-12-27	<input type="checkbox"/> Lihat <input type="checkbox"/> melunasi <input checked="" type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	dimas mar
4	2015-12-26	<input type="checkbox"/> Lihat <input type="checkbox"/> melunasi <input checked="" type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	dimas mar
5	2015-08-03	<input type="checkbox"/> Lihat <input type="checkbox"/> melunasi <input checked="" type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Tidak	dimas mar				
3	2015-12-26	<input type="checkbox"/> Lihat <input type="checkbox"/> melunasi <input checked="" type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	dimas mar

Gambar 4.183 Daftar Data Penjualan

dalam data transaksi penjualan terdapat 3 proses utama yaitu tambah, pelunasan, hapus data transaksi penjualan:

1. Tambah data penjualan, berikut ini adalah langkah-langkah untuk menambahkan data penjualan:
 - a. Klik button “Data baru” setelah itu akan menampilkan *form input data* penjualan seperti berikut:

Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

Aplikasi Pengaturan Bantuan

Administrator
(admin) Administrator

Selamat Datang

Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

Keluar

data penjualan

Tanggal: 27/12/2015

Kirim kepada: Alamat lengkap

Konsumen: Pilih konsumen

Proses pengiriman: Pengiriman (selected) Langsung

Aksi	Spesifikasi Barang Kode	QTY	Harga jual	Sub total
No content in table				

+ Tambah

Total: Rp.0

Uang muka: 0

Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

Gambar 4.184 Form Input Data Penjualan

- b. Tampilan awal *button* “Simpan” tidak dapat diklik, untuk mengaktifkannya berikut ini adalah aturannya:
- i. Pilih tanggal, secara *default* tanggal yang diisi adalah tanggal yang sesuai dengan sistem.
 - ii. Pilih konsumen, jika belum tersedia maka input data konsumen dengan cara klik *button* “Data baru” yang letaknya bersebelahan dengan *combobox* konsumen setelah itu akan tampil dialog seperti berikut:

data konsumen

Nama:

Kontak Personal: Kontak personal

Alamat lengkap: Alamat lengkap

RT / RW: 0 0

Simpan Batal

Gambar 4.185 Form Input Data Konsumen Untuk Penjualan

- c. Klik button “Tambah” yang lokasinya berada dibawah tabel penjualan barang kemudian akan menampilkan dialog seperti berikut:

Daftar barang

Kode Barang	Nama barang
1-kemasan-stick	Stick
1-kemasan-ball	Ball
1-kemasan-corn	Corn
1-kemasan-tutup cup	Tutup Cup
1-b_baku-hanco str...	Hanco Strayberry
1-b_baku-hanco co...	Hanco Coklat
1-b_baku-hanco va...	Hanco Vanila
1-ice-cup besar 10...	Cup Besar 100ml
1-ice-choco	Choco
1-ice-cup kecil 70ml	Cup Kecil 70ml

Spesifikasi Barang

Kode: Kode Barang
 Nama: Nama Barang
 Kategori: Kategori Barang
 Lokasi barang: Gudang
 QTY: QTY
 Satuan: Satuan

Harga jual: 0
 Jumlah: 0

Tutup Tambah

Gambar 4.186 Form Pilih Barang Untuk Penjualan

- d. Pilih barang pada *table-view* daftar barang kemudian pada spesifikasi barang akan menampilkan informasi tentang barang tersebut, berikut ini adalah contohnya:

The screenshot shows a software window titled "Daftar barang". On the left, there is a table view with columns "Kode Barang" and "Nama barang". The table contains several rows of data, with one row highlighted in blue: "1-b_baku-hanco str... Hanco Strawberry". On the right side of the window, there is a panel titled "Spesifikasi Barang" (Item Specifications) which displays details for the selected item. The details are as follows:

Kode	1-b_baku-hanco strawberry
Nama	Hanco Strawberry
Kategori	Bahan Baku
Lokasi barang	CENTRAL
QTY	10
Satuan	pcs

Below the specification panel, there are two input fields: "Harga jual" (Price) set to 62500 and "Jumlah" (Quantity) set to 0. A red arrow points to the "Jumlah" field.

At the bottom of the window, there are two buttons: "Tutup" (Close) and "Tambah" (Add).

Gambar 4.187 Form Pilih Barang Untuk Penjualan Yang Diseleksi

- e. Input jumlah dan harga jual jika ingin dirubah. Ketentuan input jumlah tidak boleh melebihi dari skok yang ada contohnya seperti berikut pada layout spesifikasi barang di “QTY” akan berkurang sesuai dengan jumlah yang diambil:

Daftar barang

Kode Barang	Nama barang
1-kemasan-stick	Stick
1-kemasan-ball	Ball
1-kemasan-corn	Corn
1-kemasan-tutup cup	Tutup Cup
1-b_baku-hanco str...	Hanco Strawberry
1-b_baku-hanco co...	Hanco Coklat
1-b_baku-hanco va...	Hanco Vanilla
1-ice-cup besar 10...	Cup Besar 100ml
1-ice-choco	Choco
1-ice-cup kecil 70ml	Cup Kecil 70ml

Spesifikasi Barang

Kode	1-b_baku-hanco strawberry
Nama	Hanco Strawberry
Kategori	Bahan Baku
Lokasi barang	CENTRAL
QTY	5
Satuan	pcs

Harga jual: 62500
Jumlah: 5

X Tutup **+ Tambah**

Gambar 4.188 Form Pilih Barang Qty Berkurang

- f. Kemudian klik button “Tambah” untuk menambahkan ke tabel penjualan *invoice*. Berikut ini adalah sample datanya:

Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

Administrator (admin) Administrator Keluar

Menu utama

- Data Utama
- Menu Produksi
- Menu Pendapatan
 - Daftar pesanan penjualan
 - Daftar penjualan
- Menu Pengeluaran
- Menu Laporan

Selamat Datang
Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

data penjualan

Tanggal: 27/12/2015 Kirim kepada: jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8

Konsumen: (082117355133) dimas mary... + Data B...

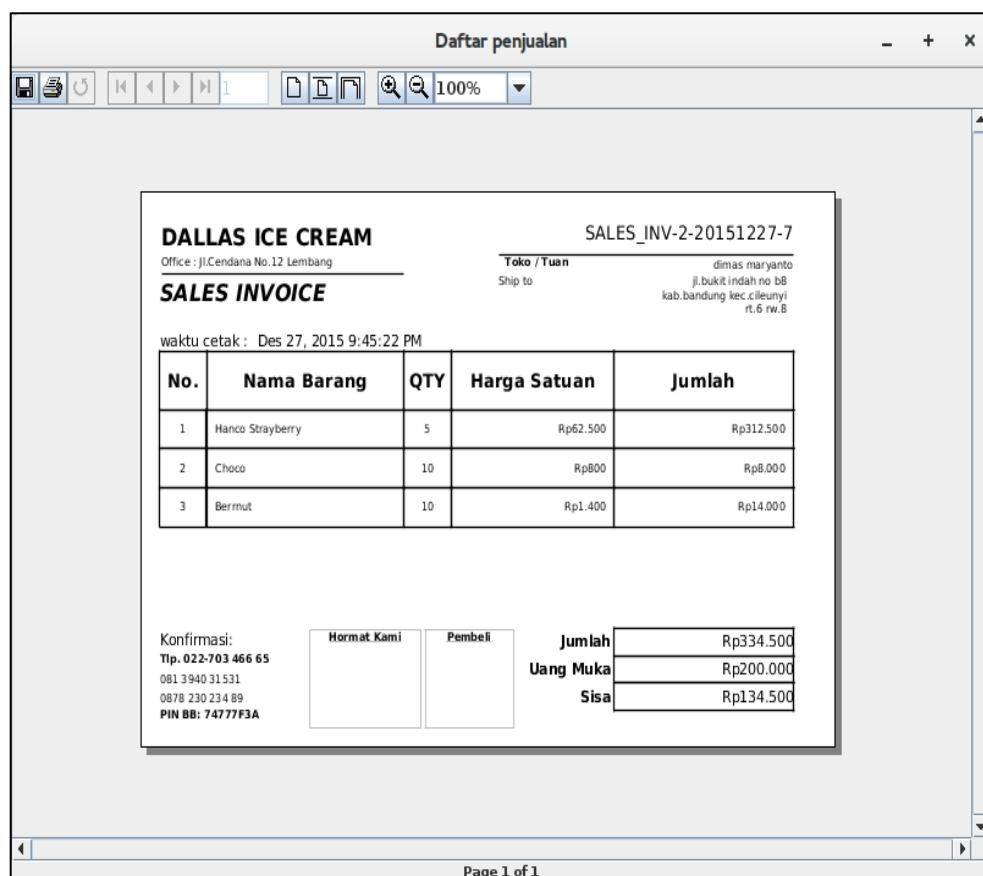
Proses pengiriman: Pengiriman Langsung

Aksi	Kode	Nama barang	QTY	Harga jual	Sub total
Hapus	1-b_baku-hanco str...	Hanco Strawberry	5	62500.0	312500.0
Hapus	1-ice-choco	Choco	10	800.0	8000.0
Hapus	1-ice-bermut	Berrnut	10	1400.0	14000.0
+ Tambah					
Total: 334500.0					
Uang muka: 0					

Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

Gambar 4.189 Form Input Data Penjualan Sample Data

- g. Input pembayaran jika terjadi pembayaran
- h. Klik button “Simpan” untuk menyimpan data penjualan.
- i. Jika data berhasil disimpan maka akan menampilkan notifikasi “Data penjualan berhasil disimpan” dan menampilkan nota penjualan seperti berikut:



Gambar 4.190 Cetak Nota Penjualan

- j. Tetapi jika tidak maka akan menampilkan pesan kesalahan “Tidak dapat menyimpan data penjualan”

2. Pelunasan data penjualan, berikut ini adalah langkah-langkah untuk melunasi pembayaran data penjualan:

a. Berikut ini adalah data yang telah tersedia untuk diubah datanya:

Daftar penjualan									
Kode	Tanggal	Aksi	Spesifikasi penjualan			Informasi penjualan			Spes...
			telah dikirim	telah diter...	telah lunas	telah kadal...			
2	2015-12-22	Lihat melunasi Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Tidak	dimas		
1	2015-12-17	Lihat melunasi Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Tidak	dimas		
6	2015-12-27	Lihat melunasi Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Belum	<input checked="" type="checkbox"/> Belum	<input checked="" type="checkbox"/> Belum	dimas mar		
4	2015-12-26	Lihat melunasi Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Belum	dimas mar		
5	2015-08-03	Lihat melunasi Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Tidak	dimas mar		
3	2015-12-26	Lihat melunasi Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Belum	dimas mar		

Gambar 4.191 Contoh Data Penjualan Yang Telah Tersedia

b. Klik *hyerlink* “Melunasi” pada kolom aksi dan pada baris tertentu, jika transaksi telah lunas maka akan menampilkan notifikasi “Transaksi penjualan tersebut telah lunas” tetapi jika belum akan akan menampilkan *form* pelunasan seperti berikut:

Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

Aplikasi Pengaturan Bantuan

Administrator
(admin) Administrator

Selamat Datang

Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

Keluar

data penjualan

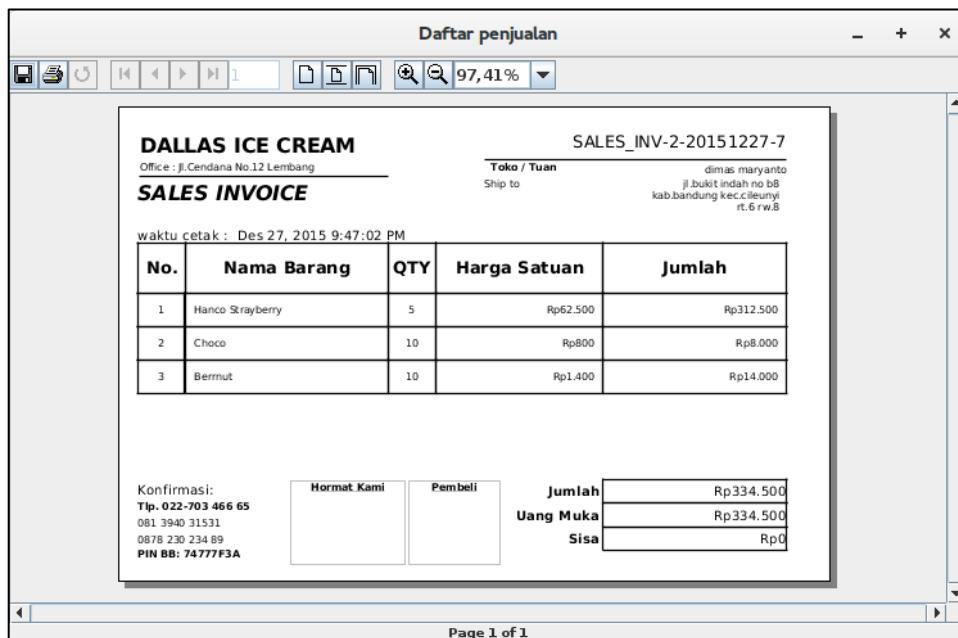
Spesifikasi Barang		QTY	\$ Harga jual	\$ Sub total
Kode	Nama barang			
1-b_baku-hanco str...	Hanco Strawberry	5	62500.0	312500.0
1-ice-choco	Choco	10	800.0	8000.0
1-ice-bernut	Bernut	10	1400.0	14000.0
		Total	334500.0	
		Uang muka	334500	
		bayar	0	
		Kembalian	0.0	

Batal Simpan

Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

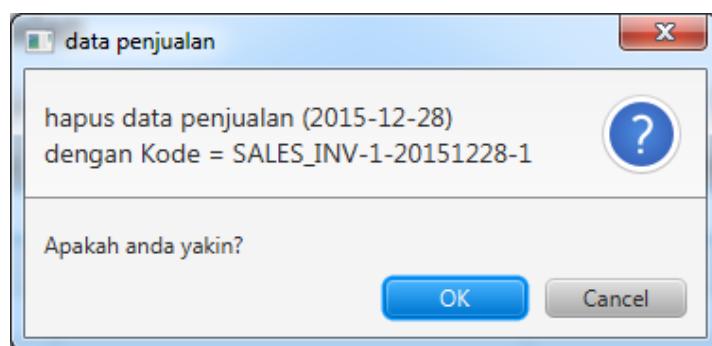
Gambar 4.192 Form Pelunasan Penjualan Produk

- c. Input jumlah nominal yang dibayarkan
- d. Klik button “Simpan” untuk menyimpan perubahan
- e. Jika data gagal diubah maka akan menampilkan pesan kesalahan “Tidak dapat merubah data penjualan” tetapi jika berhasil diubah maka akan menampilkan notifikasi “Data penjualan berhasil diubah” dan menampilkan nota penjualan ulang” seperti berikut:



Gambar 4.193 Cetak Nota Penjualan Setelah Pelunasan

3. Hapus data penjualan, berikut ini adalah cara untuk menhapus data penjualan:
 - a. Berdasarkan data yang telah tersedia klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu maka akan menampilkan konfirmasi dialog seperti berikut:



Gambar 4.194 Konfirmasi Dialog Hapus Data Penjualan

- b. Urutkan berdasarkan transaksi penjualan yang telah diterima, caranya adalah dengan melakukan klik pada kolom “Telah diterima” hasilnya akan seperti berikut:

Spesifikasi penjualan		Informasi penjualan				Spesifikasi Kons.	
Tanggal	Aksi	<input type="checkbox"/> telah dikirim	<input type="checkbox"/> telah diterima	<input type="checkbox"/> telah lunas	<input type="checkbox"/> telah kadaluwarsa	<input type="checkbox"/> Tidak	<input type="checkbox"/> Nama
2015-12-27	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Tidak	dimas maryanto
2015-12-27	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Sudah	<input type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	dimas maryanto
2015-12-22	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Sudah	<input type="checkbox"/> Tidak	dimas
2015-12-17	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Tidak	dimas			
2015-12-26	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	dimas maryanto
2015-08-03	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Sudah	<input type="checkbox"/> Tidak	dimas maryanto
2015-12-26	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	dimas maryanto

Gambar 4.196 Data Penjualan Urutkan Berdasarkan Telah Diterima

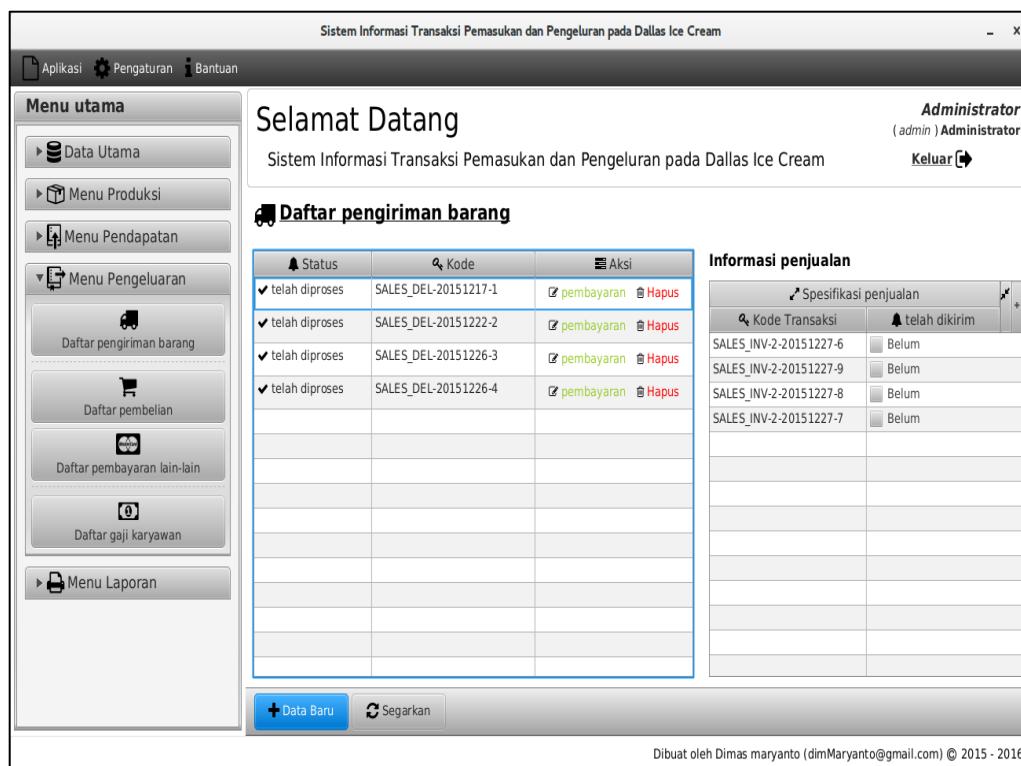
- c. Urutkan berdasarkan transaksi penjualan yang telah lunas, caranya adalah dengan melakukan klik pada kolom “Telah Lunas” hasilnya akan seperti berikut:

Spesifikasi penjualan		Informasi penjualan				Spesifikasi Kons.	
Tanggal	Aksi	<input type="checkbox"/> telah dikirim	<input type="checkbox"/> telah diterima	<input type="checkbox"/> telah lunas	<input type="checkbox"/> telah kadaluwarsa	<input type="checkbox"/> Tidak	<input type="checkbox"/> Nama
2015-12-27	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Tidak	dimas maryanto
2015-12-22	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Tidak	dimas			
2015-12-17	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Tidak	dimas			
2015-08-03	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Tidak	dimas maryanto			
2015-12-27	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	dimas maryanto
2015-12-26	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	dimas maryanto
2015-12-26	<input type="button"/> Lihat <input type="checkbox"/> melunasi <input type="checkbox"/> Hapus	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input checked="" type="checkbox"/> Sudah	<input type="checkbox"/> Belum	<input type="checkbox"/> Belum	dimas maryanto

Gambar 4.197 Data Penjualan Urutkan Berdasarkan Telah Lunas

4.5.6.7. Penggunaan program pengiriman penjualan

Untuk menggunakan data transaksi pengiriman penjualan, pilih “Daftar pengiriman” pada menu “Menu pengeluaran” setelah itu pada *center content* akan menampilkan *form* sebagai berikut:



Gambar 4.200 Daftar Data Pengiriman Penjualan

Dalam data transaksi pengiriman terdapat 3 proses utama yaitu tambah, pembayaran, dan hapus data pengiriman penjualan:

1. Tambah data pengiriman penjualan, berikut ini adalah langkah-langkah untuk menambahkan data pengiriman penjualan:
 - a. Klik button “Data baru” kemudian akan menampilkan *form* sebagai berikut:

Sistem Informasi Transaksi Pemasukan dan Pengeluaran pada Dallas Ice Cream

Administrator
(admin) Administrator
Keluar

data pengiriman barang

Aksi	Kode Transaksi	Berat/Kg	Kirim kepada	Lihat
Tidak akan dikirim	SALES_INV-2-20151227-6	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>
Tidak akan dikirim	SALES_INV-2-20151227-9	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>
Tidak akan dikirim	SALES_INV-2-20151227-8	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>
Tidak akan dikirim	SALES_INV-2-20151227-7	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>

Bayar

Status

Dibuat oleh Dimas maryanto (dimMaryanto@gmail.com) © 2015 - 2016

Gambar 4.201 Form Input Data Pengriman Penjualan

- b. Tampilan awal button “Simpan” tidak aktif, untuk mengaktifkannya berikut ini adalah aturannya:
- Pilih Tanggal pengiriman, tanggal pengiriman akan terisi secara otomatis menyesuaikan dengan tanggal yang ada disistem
 - Pilih karyawan.
- c. Jika ingin melihat transaksi penjualan klik button ”Lihat” pada kolom lihat berdasarkan baris tertentu maka akan menampilkan form seperti berikut:

rincian penjualan SALES_INV-2-20151227-6

2015-12-27	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8			
dimas maryanto				
# Nama barang	✓ Satuan	● QTY	\$ Harga jual	\$ Sub total
Hanco Strayberry	pcs	10	62500.0	625000.0
Cup Besar 100ml	pcs	10	1000.0	10000.0
Galon	pcs	2	75000.0	150000.0

Total 785000.0

Uang muka 0.0

Kembalian 785000.0

x Tutup

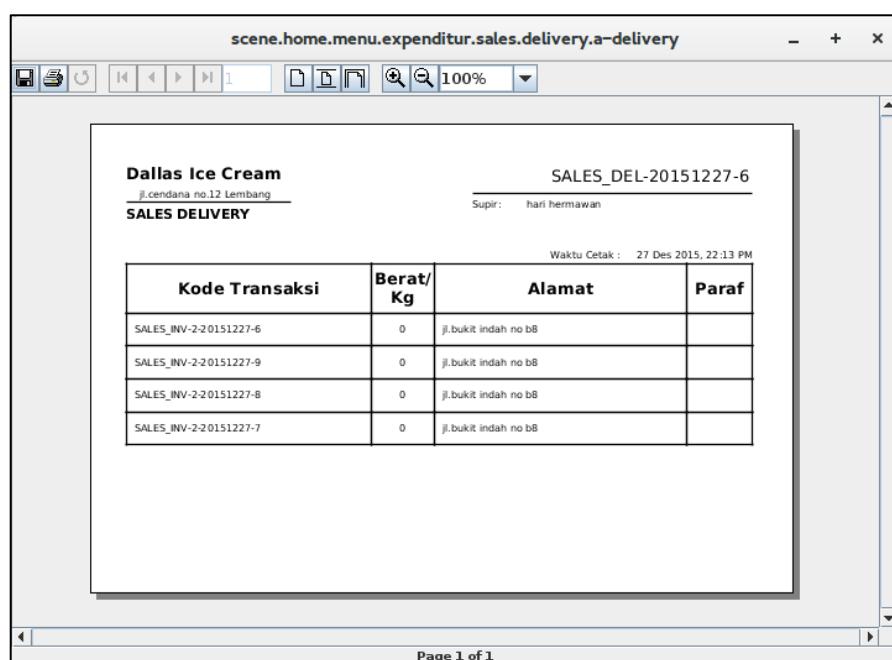
Gambar 4.202 Form Lihat Transaksi Penjualan Untuk Pengiriman Barang

- d. Kemudian pilih transaksi yang akan dikirim, caranya pada kolom aksi terdapat *checkbox* ceklis transaksi yang akan dikirimkan contohnya seperti berikut:

Aksi	Kode Transaksi	Berat/Kg	Kirim kepada	Lihat
<input checked="" type="checkbox"/> akan dikirim	SALES_INV-2-20151227-6	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	Lihat
<input checked="" type="checkbox"/> akan dikirim	SALES_INV-2-20151227-9	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	Lihat
<input checked="" type="checkbox"/> akan dikirim	SALES_INV-2-20151227-8	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	Lihat
<input checked="" type="checkbox"/> akan dikirim	SALES_INV-2-20151227-7	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	Lihat

Gambar 4.203 Form Input Data Pengiriman Dengan Sample Data

- e. Setelah itu klik *button* “Simpan” untuk menyimpan data pengiriman penjualan
- f. Jika data berhasil disimpan maka akan menampilkan notifikasi “Data pengiriman penjualan berhasil disimpan” dan menampilkan surat jalan untuk mengiriman penjualan seperti berikut:



Gambar 4.204 Cetak Surat Jalan Pengiriman Penjualan

- g. Tetapi jika tidak maka akan menampilkan pesan kesalahan “Tidak dapat menyimpan data pengiriman penjualan”

2. Pembayaran data pengiriman penjualan, berikut ini adalah langkah-langkah untuk melakukan pembayaran pengiriman penjualan:
- Berikut ini adalah data yang tersedia untuk dilakukan pembayaran:

Status	Kode	Aksi
✓ telah diproses	SALES_DEL-20151217-1	pembayaran Hapus
✓ telah diproses	SALES_DEL-20151222-2	pembayaran Hapus
✓ telah diproses	SALES_DEL-20151226-3	pembayaran Hapus
✓ telah diproses	SALES_DEL-20151226-4	pembayaran Hapus
✗ sedang diproses	SALES_DEL-20151227-6	pembayaran Hapus

Gambar 4.205 Daftar Data Pengiriman Penjualan Setelah Input Data

- Klik *hyperlink* “Pembayaran” pada kolom aksi dan baris tertentu yang memiliki status sedang diproses, maka akan menampilkan *form* sebagai berikut:

Aksi	Kode Transaksi	BeratKg	Kirim kepada	Lihat
belum diterima	SALES_INV-2-20151227-6	0	Jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	Lihat
belum diterima	SALES_INV-2-20151227-9	0	Jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	Lihat
belum diterima	SALES_INV-2-20151227-8	0	Jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	Lihat
belum diterima	SALES_INV-2-20151227-7	0	Jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	Lihat

Gambar 4.206 Form Pembayaran Pengiriman Penjualan

- c. Ceklis transaksi penjualan yang telah diterima, contohnya seperti berikut:

data pengiriman barang

Aksi	Kode Transaksi	Berat/Kg	Kirim kepada	Lihat
<input checked="" type="checkbox"/> telah diterima	SALES_INV-2-20151227-6	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>
<input checked="" type="checkbox"/> telah diterima	SALES_INV-2-20151227-9	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>
<input checked="" type="checkbox"/> telah diterima	SALES_INV-2-20151227-8	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>
<input checked="" type="checkbox"/> telah diterima	SALES_INV-2-20151227-7	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>

Gambar 4.207 Form Pembayaran Pengiriman Ceklis Penjualan

- d. Input biaya yang telah dikeluarkan untuk membeli bahan bakar kedaraan dan lain-lain, contohnya seperti berikut:

data pengiriman barang

Aksi	Kode Transaksi	Berat/Kg	Kirim kepada	Lihat
<input checked="" type="checkbox"/> telah diterima	SALES_INV-2-20151227-6	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>
<input checked="" type="checkbox"/> telah diterima	SALES_INV-2-20151227-9	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>
<input checked="" type="checkbox"/> telah diterima	SALES_INV-2-20151227-8	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>
<input checked="" type="checkbox"/> telah diterima	SALES_INV-2-20151227-7	0	jl.bukit indah no b8 kab.bandung kec.cileunyi rt.6 rw.8	<input type="button" value="Lihat"/>

Bayar

Status

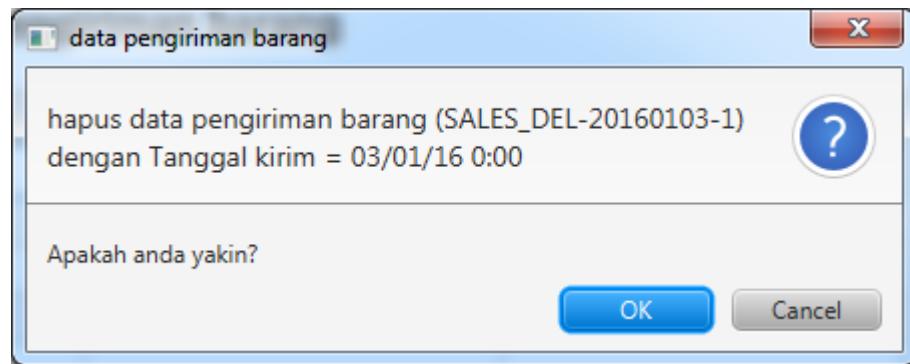
Gambar 4.208 Form Pembayaran Pengiriman Penjualan Input Biaya

- e. Klik button “Simpan” untuk menyimpan perubahan data berikut adalah hasil dari pembayaran yang status awalnya “Sedang diproses” menjadi “Telah diproses”

Status	Kode	Aksi
✓ telah diproses	SALES_DEL-20151217-1	pembayaran Hapus
✓ telah diproses	SALES_DEL-20151222-2	pembayaran Hapus
✓ telah diproses	SALES_DEL-20151226-3	pembayaran Hapus
✓ telah diproses	SALES_DEL-20151226-4	pembayaran Hapus
✓ telah diproses	SALES_DEL-20151227-6	pembayaran Hapus

Gambar 4.209 Daftar Data Pengirian Penjualan Setelah Data Diproses

3. Hapus data pengiriman penjualan, berikut ini adalah langkah-langkah untuk menghapus data pengiriman penjualan:
- Untuk menghapus data pengiriman penjualan, data statusnya harus sebelum “Telah diproses” jika melakukan klik data pengiriman dengan status tersebut maka akan menampilkan notifikasi “Data pengiriman tidak dapat dihapus karena telah diproses”
 - klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu, maka akan menampilkan konfirmasi dialog seperti berikut:



Gambar 4.210 Konfirmasi Dialog Hapus Data Pengiriman Penjualan

- c. Kemudian klik *button* “OK” untuk menghapus data pengiriman penjualan atau klik *button* “Cancel” untuk membatalkan hapus data pengiriman penjualan tersebut.

4. Refresh data pengiriman penjualan, *button* “Segarkan” digunakan untuk memuat ulang atau melakukan sinkronisasi data pengiriman penjualan antara aplikasi dengan basisdata.

4.5.6.8. Penggunaan program pembelian bahan

Untuk menggunakan data transaksi pembelian, pilih “Daftar pembelian” pada menu “Menu pengeluaran” setelah itu pada *center content* akan menampilkan *form* sebagai berikut:

Spesifikasi Pembelian		Status	Aksi	Spesifikasi Pemasok
Kode	Tanggal	Lunas	Jatuh tempo	Nama
1	2015-12-17	<input checked="" type="checkbox"/> Lunas	<input checked="" type="checkbox"/> telah lunas	<input type="button" value="Lihat"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/> supermarket
2	2015-12-26	<input type="checkbox"/> Belum	<input checked="" type="checkbox"/> segera untuk dil...	<input type="button" value="Lihat"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/> supermarket

Gambar 4.211 Daftar Data Pembelian Bahan

Dalam data transaksi pembelian terdiri dari 3 proses utama yaitu tambah, ubah, hapus data pembelian bahan, berikut ini adalah cara untuk melakukan tambah, ubah, dan hapus data pembelian bahan:

1. Tambah data pembelian, berikut ini adalah langkah-langkah untuk menambahkan data pembelian:
 - a. Klik button “Data baru” maka akan menampilkan *form input data* pembelian seperti berikut:



Gambar 4.212 Form Input Data Pembelian

- b. Tampilan awal *button* “Simpan” tidak aktif, untuk mengaktifkan *button* tersebut berikut ini adalah aturannya:
- Pilih pemasok, jika belum ada maka klik *button* “Tambah” yang lokasinya bersebelahan dengan *combobox* pemasok maka akan menampilkan *form input* pemasok seperti berikut:

Gambar 4.213 Form Input Data Pemasok Untuk Pembelian

- ii. Pilih tanggal transaksi, secara *default* tanggal transaksi akan diisi dengan tanggal yang sesuai dengan sistem operasi.
- iii. Input keterangan pembelian jika ada.
- c. Klik *button* “Tambah” yang lokasinya berada dibawah *table-view* fungsinya adalah untuk menambahkan barang, kemudian akan menampilkan *form* pilih barang seperti berikut:

Kode	Nama barang
1-kemasan-stick	Stick
1-kemasan-ball	Ball
1-kemasan-corn	Corn
1-kemasan-tutup cup	Tutup Cup
1-b_baku-hanco str...	Hanco Strawberry
1-b_baku-hanco co...	Hanco Coklat
1-b_baku-hanco va...	Hanco Vanila

Spesifikasi Barang

Kode	Kode Barang
Nama	Nama barang
Nama Kategori	Kategori barang
Lokasi barang	Nama gudang
Satuan	Satuan
QTY	QTY

Harga beli: 0
Jumlah: 0

X Tutup **+ Simpan**

Gambar 4.214 Form Pilih Barang Untuk Pembelian

- d. Pilih barangnya maka akan menampilkan spesifikasi barang yang dipilih.
- e. Input jumlah dan harga satuannya kemudian klik “Tambah” untuk menambahkan barang tersebut agar dihitung sub-totalnya. Berikut ini adalah *sample data* yang telah diinputkan:

data pembelian

Aksi	Spesifikasi Barang				QTY	\$ Sub total
	Kode	Nama	Satuan	\$ Harga		
Ubah Hapus	1-b_baku-hanco va...	Hanco Vanila	pcs	65000.0	5	325000.0
Ubah Hapus	1-b_baku-hanco co...	Hanco Coklat	pcs	62000.0	5	310000.0

Total 635000.0
 0

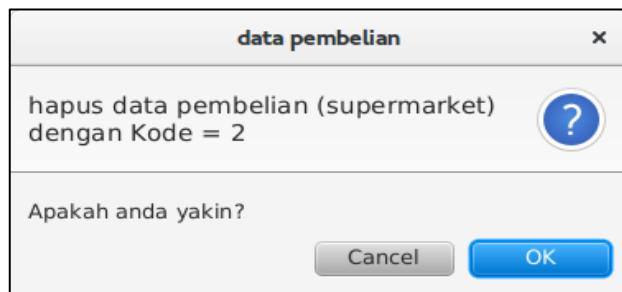
Gambar 4.215 Form Input Data Pembelian Sample Data

- f. Setelah itu input biaya yang telah dibayarkan
 - g. Kemudian klik button “Simpan” untuk menyimpan transaksi pembelian.
 - h. Jika berhasil akan menampilkan notifikasi “Data pembelian berhasil disimpan” tetapi jika tidak maka akan menampilkan pesan kesalahan “Tidak dapat menyimpan data pembelian”
2. Ubah data pembelian, ubah data pembelian ini digunakan untuk mengubah informasi transaksi pembelian contohnya salah melakukan input data, pelunasan utang dan lain-lain. Berikut ini adalah langkah-langkah untuk mengubah data pembelian bahan:
- a. Klik *hyperlink* “Ubah” pada kolom aksi dan pada baris tertentu kemudian akan menampilkan *form edit data* pembelian seperti berikut:



Gambar 4.216 Form Edit Data Pembelian

- b. Aturan penggunaannya sama seperti *form input data pembelian*
 - c. Setelah data diubah klik button “Simpan” untuk menyimpan perubahan data pembelian.
 - d. Jika data berhasil disimpan maka akan menampilkan notifikasi “Data pembelian berhasil disimpan” tetapi jika tidak maka akan menampilkan pesan kesalahan “Tidak dapat menyimpan data pembelian”
3. Hapus data pembelian, berikut ini adalah langkah-langkah untuk menghapus data pembelian:
- a. Klik *hyperlink* “Hapus” pada kolom aksi dan pada baris tertentu, kemudian akan menampilkan konfirmasi dialog seperti berikut:



Gambar 4.217 Konfirmasi Dialog Hapus Data Pembelian

- b. Klik button “OK” untuk menghapus data pembelian tersebut atau klik “Cancel” untuk membatalkan pembelian tersebut.
 - c. Jika data berhasil dihapus maka akan menampilkan notifikasi “Data pembelian berhasil dihapus” tetapi jika tidak maka akan menampilkan pesan kesalahan “Tidak dapat menghapus data pembelian”
-
-
-
-
-
-
4. Lihat data transaksi pembelian, berikut ini adalah cara untuk melihat *detail* transaksi pembelian. Klik *hyperlink* “Lihat” pada kolom aksi dan pada baris tertentu maka akan menampilkan *form* lihat transaksi pembelian seperti berikut:

data pembelian					
Pemasok	(1) supermarket	Tanggal	01/09/2015		
Keterangan					
Spesifikasi Barang					
Kode	Nama	Satuan	Harga	QTY	Sub total
1-b_baku-hanco co...	Hanco Coklat	pcs	62000.0	5	310000.0
Total: 31.0000.0 Bayar: 0					

Gambar 4.218 Form Lihat Transaksi Pembelian

5. Refresh data pembelian, *button* “Segarkan” digunakan untuk memuat ulang atau melakukan singkronisasi data pembelian antara aplikasi dengan basis data.
6. Sorting data, berdasarkan status pembelian apakah udah lunas atau belum dan sudah jatuh tempo atau belum. Contohnya seperti berikut:
 - a. Urutkan berdasarkan lunas, caranya adalah dengan melakukan klik pada kolom “Lunas” seperti pada gambar berikut ini:

Kode	Tanggal	Lunas	Jatuh tempo	Aksi	Nama
1	2015-12-17	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Lunas	<input type="button" value="Lihat"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/>	supermarket
3	2015-12-27	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Lunas	<input type="button" value="Lihat"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/>	supermarket
2	2015-12-26	<input type="checkbox"/>	<input type="checkbox"/> Belum	<input type="button" value="Lihat"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/>	supermarket

Gambar 4.219 Data Pembelian Diurutkan Berdasarkan Lunas

- b. Urutkan berdasarkan pembayaran yang telah jatuh tempo, caranya adalah dengan melakukan klik pada kolom “Jatuh Tempo” seperti pada gambar berikut ini:

Kode	Tanggal	Lunas	Jatuh tempo	Aksi	Nama
4	2015-09-01	<input type="checkbox"/>	<input checked="" type="checkbox"/> Belum	<input type="button" value="Lihat"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/>	supermarket
1	2015-12-17	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Lunas	<input type="button" value="Lihat"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/>	supermarket
2	2015-12-26	<input checked="" type="checkbox"/>	<input type="checkbox"/> Belum	<input type="button" value="Lihat"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/>	supermarket
3	2015-12-27	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> Lunas	<input type="button" value="Lihat"/> <input type="button" value="Ubah"/> <input type="button" value="Hapus"/>	supermarket

Gambar 4.220 Data Pembelian Diurutkan Berdasarkan Jatuh Tempo

4.5.6.9. Penggunaan program pembayaran lain-lain

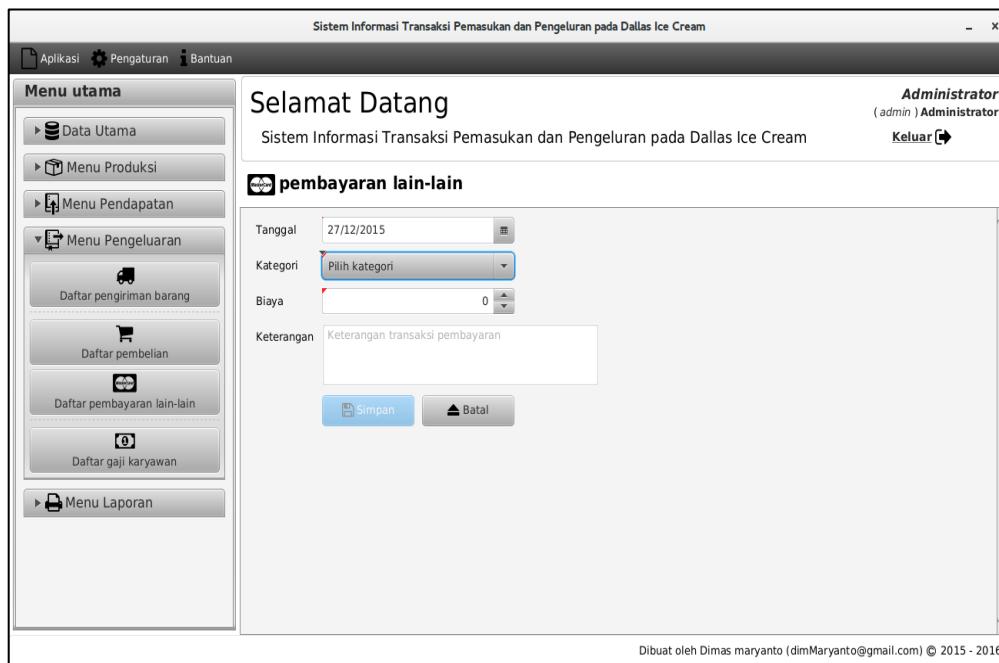
Untuk menggunakan data transaksi pembayaran lain-lain, pilih “Daftar pembayaran” pada menu “Menu pengeluaran” setelah itu pada center content akan menampilkan form berikut ini:



Gambar 4.221 Daftar Data Pembayaran Lain-Lain

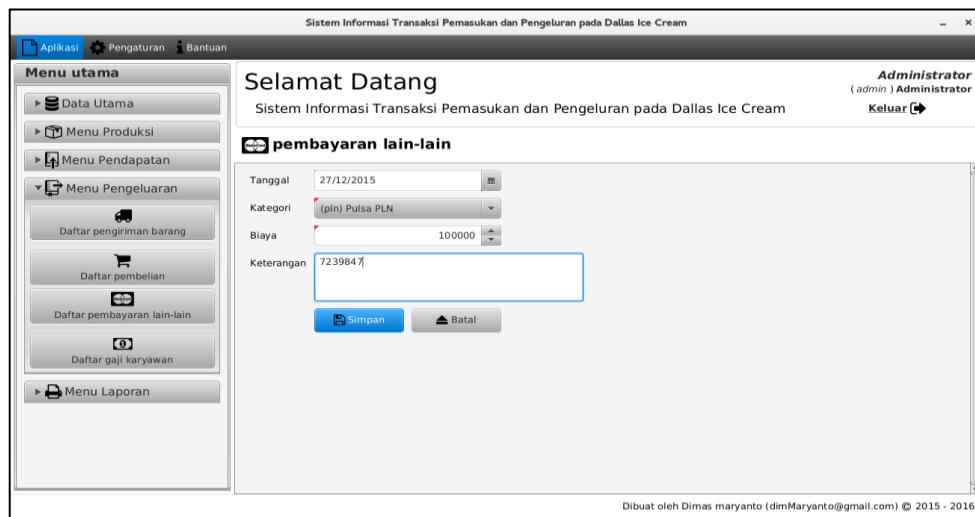
Dalam data transaksi pembayaran lain-lain terdapat 3 proses utama yaitu tambah dan hapus data pembayaran, berikut ini adalah cara tambah, ubah dan hapus data pembayaran lain-lain:

1. Tambah data pembayaran lain-lain, berikut ini adalah langkah-langkah untuk menambah data pembayaran lain-lain:
 - a. Klik button “Data baru” maka akan menampilkan form input data pembayaran lain-lain sebagai berikut:



Gambar 4.222 Form Input Data Pembayaran Lain-Lain

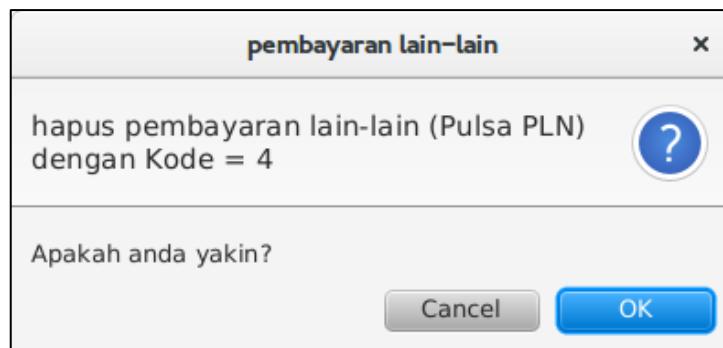
- b. Kemudian pilih kategori pembayaran
 - c. Input biaya yang dikeluarkan
 - d. Klik *button* “Simpan”, jika data berhasil disimpan maka akan menampilkan notifikasi “Data pembayaran lain-lain berhasil disimpan” tetapi jika tidak maka akan menampilkan pesan kesalahan “Tidak dapat menyimpan data pembayaran lain-lain”
2. Ubah data pembayaran lain-lain, berikut ini adalah langkah-langkah untuk mengubah data pembayaran lain-lain:
 - a. Klik *hyperlink* “Ubah” pada kolom aksi dan baris tertentu maka akan memampulkan *form edit* pembayaran lain-lain seperti berikut:



Gambar 4.223 Form Edit Data Pembayaran Lain-Lain

- b. Setelah data diubah maka klik button “Simpan” untuk menyimpan perubahan data pembayaran lain-lain

3. Hapus data pembayaran lain-lain, berikut ini adalah langkah-langkah untuk menghapus data pembayaran lain-lain:
 - a. Klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu maka akan menampilkan konfirmasi dialog sebagai berikut:

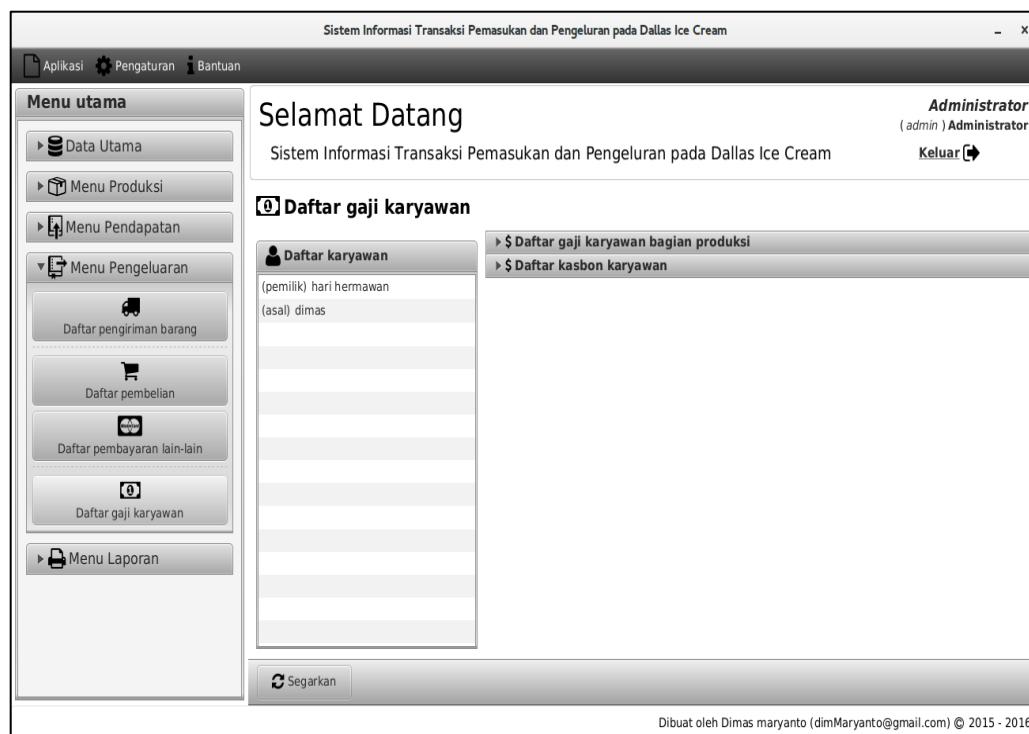


Gambar 4.224 Konfirmasi Dialog Hapus Data Pembayaran Lain-Lain

- -
 -
 - b. Klik *button* “OK” untuk menghapus data pembayaran lain-lain atau klik “Cancel” untuk membatalkan hapus data pembayaran lain-lain.
4. Refresh data pembayaran lain-lain, *button* “Segarkan” digunakan untuk memuat ulang atau melakukan singkronisasi data pembayaran lain-lain antara aplikasi dengan basisdata.

4.5.6.10. Penggunaan program penggajian karyawan

Untuk menggunakan data transaksi penggajian karyawan, pilih “Daftar penggajian” pada menu “Menu pengeluaran” setelah itu pada *center content* akan menampilkan *form* seperti berikut:

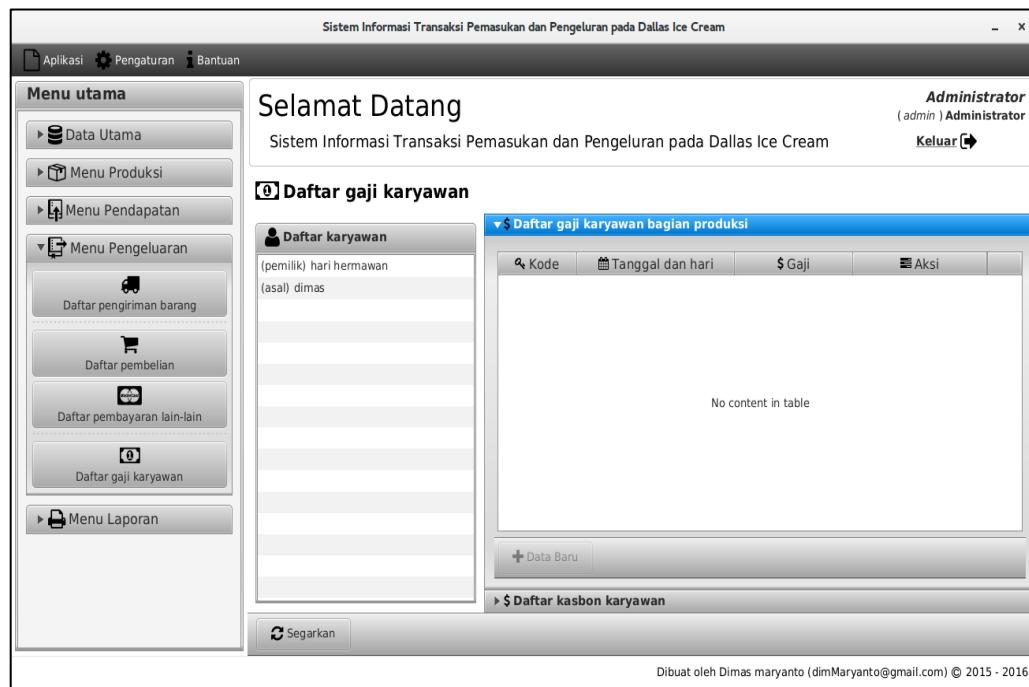


Gambar 4.225 Daftar Data Karyawan Untuk Penggajian

Jika diperhatika terdapat 2 menu yaitu daftar gaji karyawan bagian produksi dan daftar kasbon karyawan berikut ini adalah penjelasannya:

a. Penggajian karyawan bagian produksi

Untuk menggunakan *menu* penggajian bagian produksi klik judul “Daftar karyawan bagian produksi” kemudian akan menampilkan *form* seperti berikut:



Gambar 4.226 Daftar Gaji Karyawan Bagian Produksi

Untuk berinteraksi dengan menu ini anda terlebih dahulu harus memilih karyawan pada “Daftar karyawan” jika tidak button “Simpan” tidak aktif contohnya seperti berikut:



Gambar 4.227 Daftar Gaji Karyawan Bagian Produksi Pilih Karyawan

Setelah karyawan dipilih maka akan menampilkan data penggajian yang telah dilakukan oleh karyawan tersebut dan *button* “Data baru” aktif. Dalam menu ini terdapat 2 proses utama yaitu tambah dan hapus data penggajian karyawan bagian karyawan produksi yaitu seperti berikut:

1. Tambah data penggajian karyawan bagian produksi, berikut ini adalah langkah-langkah untuk menambahkan data penggajian bagian produksi:
 - a. Klik *button* “Data baru” maka akan menampilkan *form input data* penggajian karyawan bagian produksi sebagai berikut:

Pilih	Kode	Tanggal
<input checked="" type="checkbox"/> belum dihitung	3	2015-12-27

Kode Barang	Nama barang	Satuan	Qty	Harga	Sub total
No content in table					

Gaji Pokok: Rp. 0
Bonus: 0
Total: Rp. 0

Gambar 4.228 Form Input Data Penggajian Karyawan Bagian Produksi

- b. Tampilan awal *button* “Simpan” tidak aktif, untuk mengaktifkannya berikut ini adalah aturannya:
 - i. Karyawan, akan secara otomatis diisi sesuai dengan daftar karyawan yang telah dipilih.

- ii. Tanggal, akan secara otomatis diisi dengan menyesuaikan tanggal yang ada pada sistem operasi
- iii. Pada tabel daftar produksi ceklik data produksi yang ingin dihitung maka secara otomatis pada tabel barang akan menampilkan jumlah barang yang diproduksi. Contohnya seperti berikut:

④ data penggajian untuk karyawan bagian produksi

Karyawan	hari hermawan	Pilih	Kode	Tanggal																																										
Tanggal	27/12/2015	<input checked="" type="checkbox"/> telah dihitung	3	2015-12-27																																										
<table border="1"> <thead> <tr> <th>Kode Barang</th> <th>Nama barang</th> <th>Satuan</th> <th>Qty</th> <th>Harga</th> <th>Sub total</th> </tr> </thead> <tbody> <tr> <td>1-ice-vanies</td> <td>Vanies</td> <td>Ice Cream</td> <td>20</td> <td>0.0</td> <td>0.0</td> </tr> <tr> <td>1-ice-cup besar 100ml</td> <td>Cup Besar 100ml</td> <td>Ice Cream</td> <td>15</td> <td>0.0</td> <td>0.0</td> </tr> <tr> <td>1-ice-cup kecil 70ml</td> <td>Cup Kecil 70ml</td> <td>Ice Cream</td> <td>20</td> <td>0.0</td> <td>0.0</td> </tr> <tr> <td>1-ice-choco</td> <td>Choco</td> <td>Ice Cream</td> <td>20</td> <td>0.0</td> <td>0.0</td> </tr> <tr> <td>1-ice-davacho</td> <td>Davacho</td> <td>Ice Cream</td> <td>20</td> <td>0.0</td> <td>0.0</td> </tr> <tr> <td>1-ice-bernut</td> <td>Bernut</td> <td>Ice Cream</td> <td>20</td> <td>0.0</td> <td>0.0</td> </tr> </tbody> </table>					Kode Barang	Nama barang	Satuan	Qty	Harga	Sub total	1-ice-vanies	Vanies	Ice Cream	20	0.0	0.0	1-ice-cup besar 100ml	Cup Besar 100ml	Ice Cream	15	0.0	0.0	1-ice-cup kecil 70ml	Cup Kecil 70ml	Ice Cream	20	0.0	0.0	1-ice-choco	Choco	Ice Cream	20	0.0	0.0	1-ice-davacho	Davacho	Ice Cream	20	0.0	0.0	1-ice-bernut	Bernut	Ice Cream	20	0.0	0.0
Kode Barang	Nama barang	Satuan	Qty	Harga	Sub total																																									
1-ice-vanies	Vanies	Ice Cream	20	0.0	0.0																																									
1-ice-cup besar 100ml	Cup Besar 100ml	Ice Cream	15	0.0	0.0																																									
1-ice-cup kecil 70ml	Cup Kecil 70ml	Ice Cream	20	0.0	0.0																																									
1-ice-choco	Choco	Ice Cream	20	0.0	0.0																																									
1-ice-davacho	Davacho	Ice Cream	20	0.0	0.0																																									
1-ice-bernut	Bernut	Ice Cream	20	0.0	0.0																																									
<input type="text"/> Gaji Pokok Rp.0 <input type="text"/> Bonus 0 <input type="text"/> Total Rp.0																																														

Gambar 4.229 Form Input Data Penggajian Karyawan Pilih Produksi

- iv. Input harga satuan barang, dengan cara *double click* pada kolom harga dan *cell* tertentu maka akan menampilkan *textfield* kemudian input harganya contohnya seperti berikut:

Kode Barang	Nama barang	Satuan	Qty	Harga	Sub total
1-ice-vanies	Vanies	Ice Cream	20	150.0	3000.0
1-ice-cup besar 100ml	Cup Besar 100ml	Ice Cream	15	200.0	3000.0
1-ice-cup kecil 70ml	Cup Kecil 70ml	Ice Cream	20	100.0	2000.0
1-ice-choco	Choco	Ice Cream	20	100.0	2000.0
1-ice-davacho	Davacho	Ice Cream	20	100.0	2000.0
1-ice-bernut	Bernut	Ice Cream	20	100.0	2000.0

Gambar 4.230 Form Input Data Penggajian Karyawan Input Harga

v. Input bonus jika ada biaya tambahan contohnya seperti berikut:

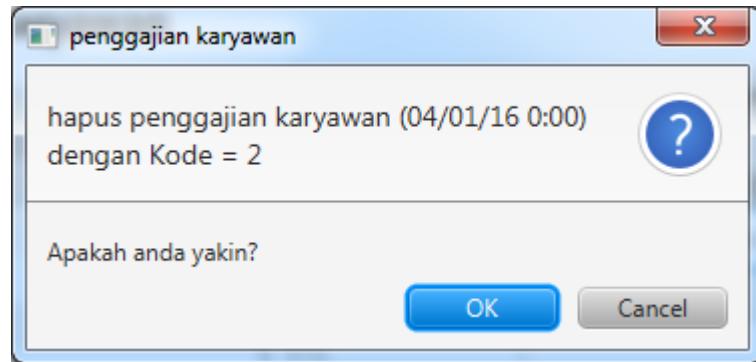
data penggajian untuk karyawan bagian produksi					
Kode Barang	Nama barang	Satuan	Qty	Harga	Sub total
1-ice-vanies	Vanies	Ice Cream	20	150.0	3000.0
1-ice-cup besar 100ml	Cup Besar 100ml	Ice Cream	15	200.0	3000.0
1-ice-cup kecil 70ml	Cup Kecil 70ml	Ice Cream	20	100.0	2000.0
1-ice-choco	Choco	Ice Cream	20	100.0	2000.0
1-ice-davacho	Davacho	Ice Cream	20	100.0	2000.0
1-ice-bermut	Berrnut	Ice Cream	20	100.0	2000.0

Gaji Pokok: 14000.0
Bonus: 1600
Total: 30000.0

Batal **Simpan**

Gambar 4.231 Form Input Data Penggajian Karyawan Input Bonus

- c. Setelah itu klik button “Simpan” untuk menyimpan data transaksi penggajian karyawan bagian produksi.
 - d. Jika data berhasil disimpan akan menampilkan notifikasi “Data penggajian karyawan bagian produksi berhasil disimpan” tetapi jika tidak maka akan menampilkan pesan kesalahan “Tidak dapat menyimpan data penggajian karyawan bagian produksi”
2. Hapus data penggajian karyawan bagian produksi, berikut ini adalah langkah-langkah untuk menghapus data penggajian bagian produksi:
- a. Klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu maka akan menampilkan konfirmasi dialog seperti berikut:

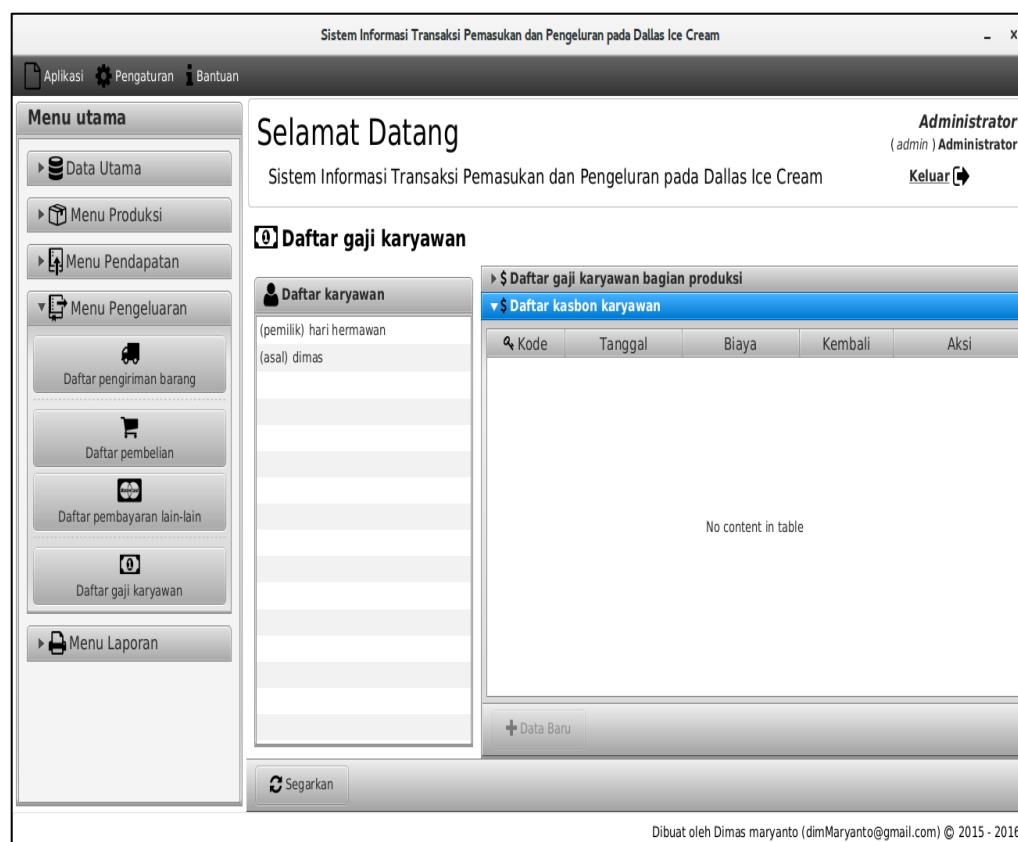


Gambar 4.232 Konfirmasi Dialog Hapus Data Penggajian Karyawan

- b. Klik *button* “Ok” untuk menghapus data transaksi penggajian tersebut atau klik *button* “Cancel” untuk membatalkan hapus data transaksi penggajian tersebut.
- c. Jika data berhasil dihapus maka akan menampilkan notifikasi “Data penggajian karyawan berhasil dihapus” tetapi jika tidak maka akan menampilkan pesan kesalahan “Data penggajian karyawan bagian produksi tidak dapat dihapus”.

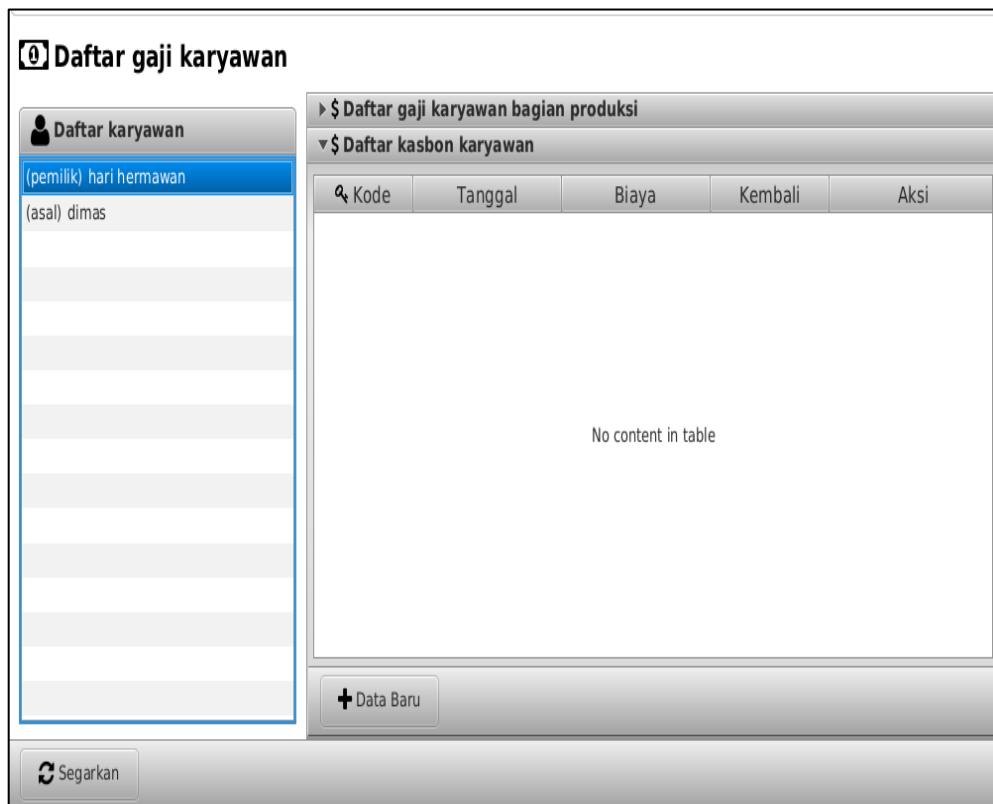
b. Kasbon karyawan

Untuk menggunakan data kasbon karyawan, klik menu “Daftar kasbon karyawan” pada form “Daftar penggajian karyawan” hasilnya adalah seperti berikut:



Gambar 4.233 Daftar Kasbon Karyawan

Untuk berinteraksi dengan menu “Daftar kasbon karyawan” terlebih dahulu harus memilih karyawan pada “Daftar karyawan” contohnya adalah seperti berikut:



Gambar 4.234 Daftar Data Kasbon Karyawan Pilih Karyawan

Dalam data kasbon karyawan terdapat 3 proses utama yaitu tambah, pelunasan dan hapus data kasbon karyawan, seperti berikut ini:

1. Tambah data kasbon karyawan, berikut ini adalah langkah-langkah untuk menambahkan data kasbon karyawan:
 - a. Klik button “Data baru” maka akan tampil *form input data* kasbon karyawan seperti berikut:

The screenshot shows a Java Swing application window titled "data kasbon". The interface includes:

- A "Pegawai" field containing "hari hermawan".
- A "Tanggal" field showing "27/12/2015" with a calendar icon.
- A "Biaya" field with a red asterisk (*) and a value of "0" with up/down arrows.
- A "Bayar" field with a value of "0" and up/down arrows.
- Two buttons at the bottom: a blue "Simpan" button with a disk icon and a grey "Batal" button with a triangle icon.

Gambar 4.235 Form Input Data Kasbon Karyawan

- b. Tampilan awal *button* “Simpan” tidak aktif, untuk mengaktifkannya dengan cara input nominal yang dinjam oleh karyawan
 - c. Kemudian klik *button* “Simpan” untuk menyimpan data kasbon karyawan jika data berhasil disimpan maka akan memampilkan notifikasi “Data kasbon karyawan berhasil disimpan” tetapi jika tidak akan menampilkan pesan kesalahan “Tidak dapat menyimpan data kasbon karyawan”
2. Pelunasan data kasbon karyawan, berikut ini adalah langkah-langkah untuk melakukan pelunasan kasbon karyawan:
- a. Klik *hyperlink* “Ubah” pada kolom aksi dan baris tertentu setelah itu maka akan muncul *form edit data* kasbon karyawan seperti berikut:

The screenshot shows a software interface titled "data kasbon". It contains several input fields and buttons. At the top left is a logo consisting of a square with a circle inside. The title "data kasbon" is centered above the input fields. The first field is labeled "Pegawai" with the value "hari hermawan". The second field is labeled "Tanggal" with the value "27/12/2015" and a calendar icon. The third field is labeled "Biaya" with the value "120000" and an up/down arrow icon. The fourth field is labeled "Bayar" with the value "120000" and an up/down arrow icon. Below these fields are two buttons: a blue "Simpan" button with a disk icon and a grey "Batal" button with a triangle icon.

Gambar 4.236 Form Edit Data Kasbon Karyawan

- b. Input biaya yang dibayarkan
- c. Klik *button* “Simpan” untuk menyimpan perubahan, jika data berhasil disimpan maka akan menampilkan notifikasi “Data kasbon karyawan berhasil diubah” tetapi jika tidak maka akan menampilkan pesan kesalahan “Tidak dapat merubah data kasbon tersebut”
3. Hapus data kasbon karyawan, berikut ini adalah langkah-langkah untuk menghapus data kasbon karyawan:
 - a. Klik *hyperlink* “Hapus” pada kolom aksi dan baris tertentu maka akan menampilkan konfirmasi dialog seperti berikut:
 - b. Klik *button* “Ok” untuk menghapus data kasbon karyawan atau klik *button* “Cancel” untuk membatalkan hapus data kasbon karyawan
 - c. Setelah *button* “Ok” diklik jika berhasil disimpan akan menampilkan notifikasi “Data kasbon karyawan telah dihapus” tetapi jika tidak akan menampilkan pesan kesalahan “Tidak dapat menghapus data kasbon karyawan tersebut”.