

## BAB IV

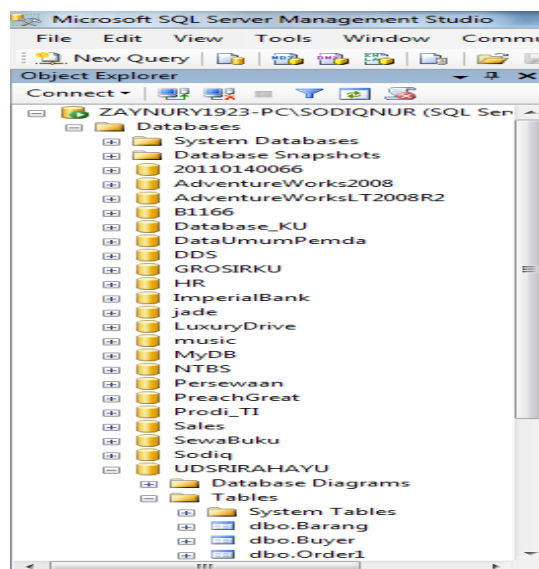
### HASIL DAN PEMBAHASAN

#### 4.1 Pembangunan Sistem

Pembangunan sistem aplikasi *input* data order dimulai dengan membuat data *buyer*, data barang dan data order. Penulis juga membuat basis data yang memuat data buyer, data barang, serta data order. Setelah membuat basis data yang dibutuhkan kemudian penulis melakukan implementasi *user interface* yang telah dibuat sebelumnya, setelah itu membuat sistem yang dapat mengexport data ke Microsoft Excel.

#### 4.2 Pembuatan Basis Data

Dalam pembuatan basis data yang digunakan untuk menyimpan seluruh data pada sistem, penulis menggunakan *database* Microsoft SQL Server 2008 dan Visual Studio 2010 sebagai *software* pendukungnya. Hasil dan pembahasan basis data yang dibuat dapat dilihat pada gambar 4.1.



Gambar 4.1 Basis Data Aplikasi

Berikut ini merupakan penjelasan dari masing-masing tabel yang ada pada sistem yang dibuat:

a. Barang

Pada tabel barang berisi tentang detail data barang yaitu kode Barang, nama barang, finishing, panjang, lebar, tinggi dan harga barang.

b. Buyer

Pada tabel *buyer* terdapat informasi tentang *buyer* yaitu kode *buyer*, nama *buyer*, nama perusahaan dan negara asal *buyer*.

c. Order1

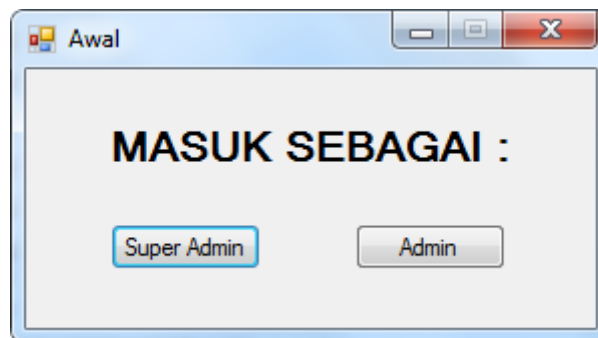
Pada tabel order1 berisi semua informasi tentang order yaitu kode order, kode *buyer*, nama *buyer* yang melakukan order, kode barang, nama barang, finishing, panjang, lebar, tinggi, harga per pcs, jumlah barang yang dipesan, harga total dan kontainer tanggal keberangkatan.

### 4.3 Implementasi Aplikasi

Implementasi aplikasi ini akan ditunjukkan dengan menampilkan *User Interface* yang dibuat dimulai dari halaman input data buyer, input data barang, input data order sampai lihat order *details*.

#### 4.3.1 Form Awal

*Form Awal* merupakan halaman pertama yang muncul dari sistem aplikasi yang dijalankan. Pada halaman ini *user* dapat memilih untuk *login* sebagai *Super Admin* atau *Admin*. Tampilan *form Awal* dapat dilihat pada gambar 4.2.

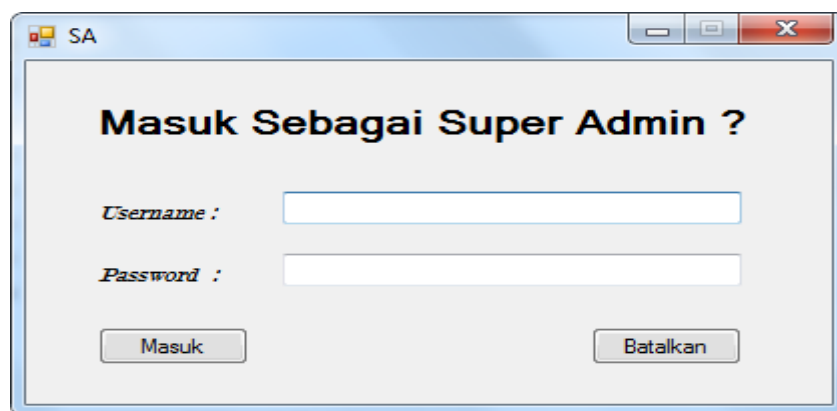


**Gambar 4.2** *Form Awal*

Pada *Form Awal* terdapat dua fungsi yaitu fungsi *button Super Admin* dan *button Admin*. *Button Super Admin* berfungsi untuk masuk ke halaman *login* untuk *Super Admin*. Dan *button Admin* digunakan untuk masuk ke halaman *login Admin*.

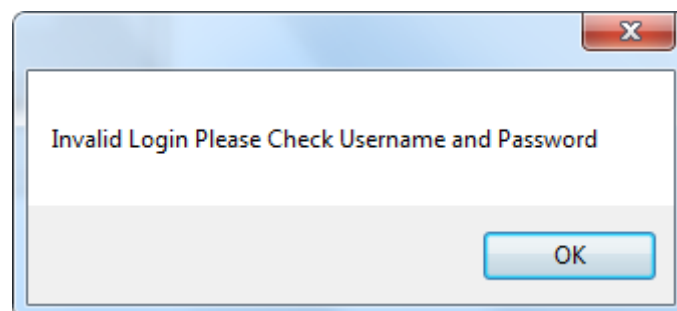
#### 4.3.2 Form Login Super Admin

*Form login Super Admin* merupakan halaman ketika pengguna memilih fungsi *button Super Admin* pada halaman awal. Halaman ini digunakan untuk *Super Admin* untuk masuk ke halaman berikutnya. Diperlukan *username* dan *password* yang benar untuk dapat masuk ke halaman berikutnya. Tampilan *Form Login Super Admin* dapat dilihat pada gambar 4.3.



**Gambar 4.3** *Form Login Super Admin*

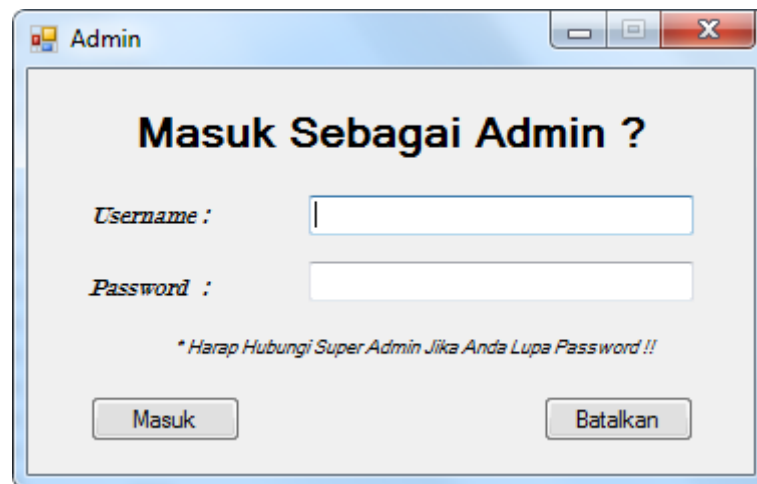
Pada *Form Login Super Admin* terdapat dua fungsi yaitu fungsi *button* masuk dan *button* batalkan. *Button* masuk di gunakan untuk masuk ke halaman berikutnya setelah pengguna memasukan *username* dan *password*. Dan *button* batalkan digunakan untuk membatalkan dan kembali ke halaman awal. Untuk dapat masuk ke halaman berikutnya pengguna harus memasukan *username* dan *password* yang benar. Jika *username* dan *password* yang dimasukan salah maka akan muncul peringatan untuk memasukan *username* dan *password* yang benar yang dapat dilihat pada gambar 4.4.



**Gambar 4.4** Tampilan Peringatan *Login Super Admin*

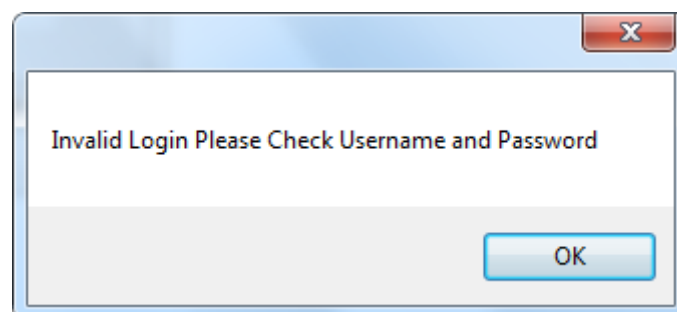
### 4.3.3 Form Login Admin

*Form login Admin* merupakan halaman ketika pengguna memilih fungsi *button Admin* pada halaman awal. Halaman ini digunakan untuk *Admin* untuk masuk ke halaman berikutnya. Diperlukan *username* dan *password* yang benar untuk dapat masuk ke halaman berikutnya. Tampilan *Form Login Admin* dapat dilihat pada gambar 4.5.



**Gambar 4.5** *Form Login Admin*

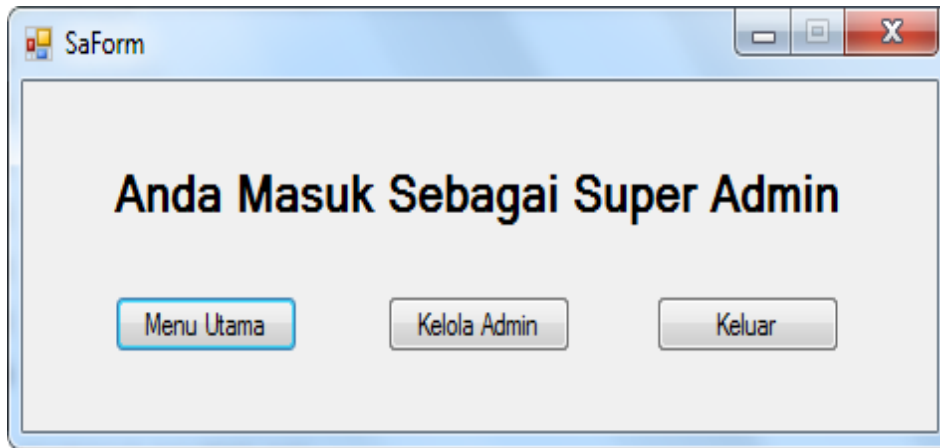
Pada *Form Login Admin* terdapat dua fungsi yaitu fungsi *button* masuk dan *button* batalan. *Button* masuk di gunakan untuk masuk ke halaman berikutnya setelah pengguna memasukan *username* dan *password*. Dan *button* batalan digunakan untuk membatalkan dan kembali ke halaman awal. Untuk dapat masuk ke halaman berikutnya pengguna harus memasukan *username* dan *password* yang benar. Jika *username* dan *password* yang dimasukan salah maka akan muncul peringatan untuk memasukan *username* dan *password* yang benar yang dapat dilihat pada gambar 4.6.



**Gambar 4.6** Tampilan Peringatan *Login Admin*

#### 4.3.4 Form Utama Super Admin

*Form Utama Super Admin* merupakan halaman ketika *Super Admin* berhasil *login*. Pada halaman ini digunakan pengguna untuk memilih masuk ke menu utama atau untuk mengelola *admin*. Tampilan *Form Utama Super Admin* dapat dilihat pada gambar 4.7.

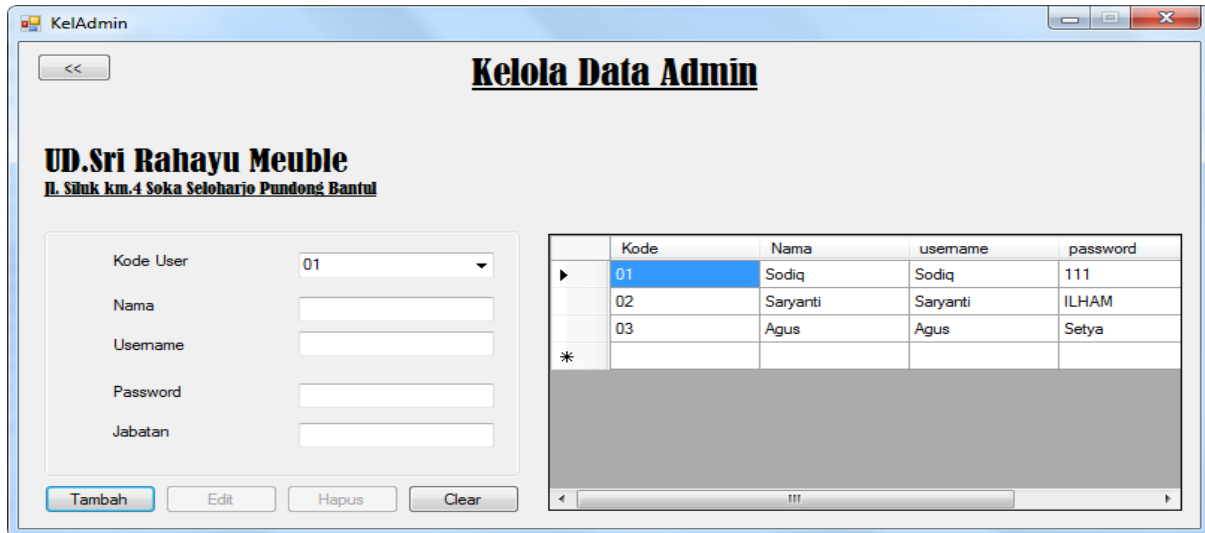


**Gambar 4.7** *Form Utama Super Admin*

Pada *Form Utama Super Admin* terdapat fungsi *button* menu utama, *button* kelola *admin* dan *button* keluar. *Button* menu utama digunakan untuk masuk ke tampilan utama dari aplikasi, sedangkan *button* kelola *admin* digunakan untuk masuk ke halaman kelola *admin*.

### 4.3.5 Form Kelola Admin

Form Kelola Admin merupakan halaman ketika *Super Admin* memilih *button* kelola admin pada form utama *super admin*. Halaman ini digunakan *super admin* untuk mengelola *admin* yang dapat mengakses aplikasi, baik itu menambah, menghapus dan mengubah data *admin*. Tampilan form kelola admin dapat dilihat pada gambar 4.8.



The screenshot shows a web application window titled 'KelAdmin'. The main heading is 'Kelola Data Admin'. Below the heading, the user's name and address are displayed: 'UD.Sri Rahayu Meuble' and 'Jl. Siliuk km.4 Soka Seloharjo Pundong Bantul'. The form contains several input fields: 'Kode User' (a dropdown menu with '01' selected), 'Nama', 'Username', 'Password', and 'Jabatan'. At the bottom of the form are four buttons: 'Tambah', 'Edit', 'Hapus', and 'Clear'. To the right of the form is a table with the following data:

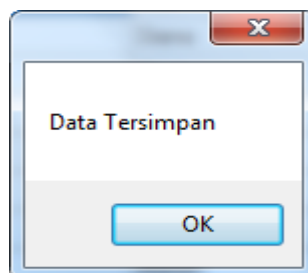
Kode	Nama	username	password
01	Sodiq	Sodiq	111
02	Saryanti	Saryanti	ILHAM
03	Agus	Agus	Setya
*			

Gambar 4.8 Form Kelola Admin

Pada *form* input data buyer terdapat 5 fitur untuk mengolah data yaitu tambah data, hapus data, *edit* data, *clear*.

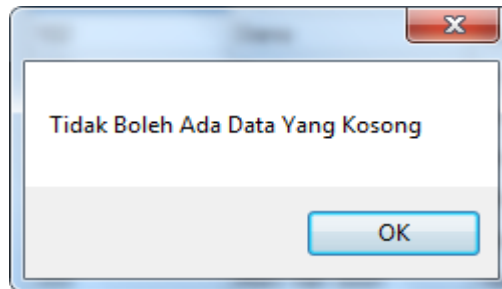
Fitur *clear* digunakan untuk membersihkan *textbox* jika data yang dimasukan salah atau tidak sesuai.

Fitur tambah data digunakan untuk memasukan data admin ke *database*. *User* dapat memasukan data dengan mengisi data pada *combobox* maupun *textbox* yang tersedia. Setelah semua data terisi user dapat memasukan data ke *database* dengan memilih tombol tambah pada *form* kelola admin. Kemudian akan muncul *messagebox* jika data berhasil disimpan, dapat kita lihat pada gambar 4.9.



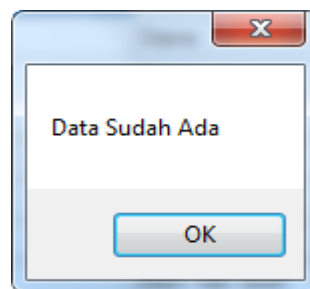
Gambar 4.9 Messagebox Data Tersimpan

User harus mengisi semua data secara lengkap pada *combobox* maupun *textbox*. Jika terdapat *textbox* atau *combobox* yang kosong maka akan muncul *messagebox* seperti pada gambar 4.10.



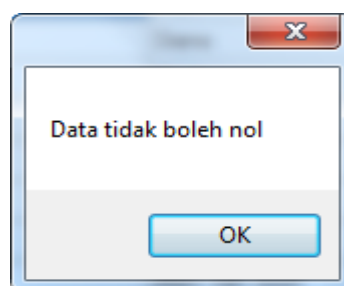
**Gambar 4.10** *Messagebox* data tidak boleh kosong

User tidak boleh memasukan kode *user* yang sama pada *combobox*. Jika data kode buyer yang dimasukan sama maka akan muncul *messagebox* seperti pada gambar 4.11.



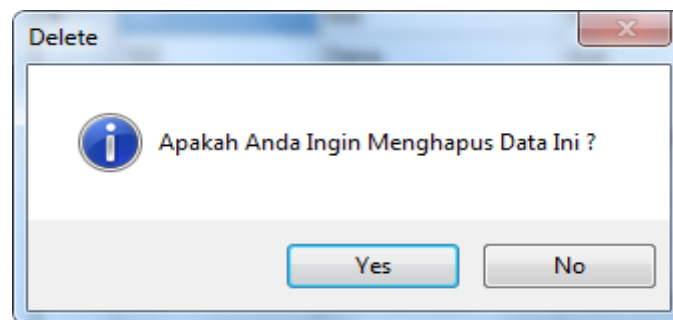
**Gambar 4.11** *Messagebox* data sudah ada

User tidak boleh memasukan data 0 saja pada *combobox* maupun *textbox*. Jika data yang dimasukan 0 saja maka akan muncul *messagebox* seperti pada gambar 4.12.



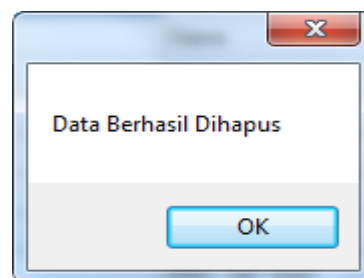
**Gambar 4.12** *Messagebox* data tidak boleh nol

User juga dapat menghapus data yang diinginkan dengan menggunakan fitur hapus data pada *form* kelola *admin*. User dapat memilih data yang akan dihapus dengan memilih kode *user* pada *combobox* atau dapat memilih dengan cara *double click* data yang ingin dihapus pada *datagridview* yang ada pada *form* kelola *admin*. Setelah memilih data yang akan dihapus *user* dapat menghapus data dengan menekan tombol hapus. Setelah tombol hapus ditekan maka akan muncul *messagebox* untuk memastikan apakah data yang dipilih benar – benar akan dihapus seperti pada gambar 4.13.



**Gambar 4.13** *Messagebox Alert Hapus Data*

Apabila *user* ingin membatalkan data yang akan dihapus, *user* dapat menekan tombol *No*. Jika data yang dipilih benar – benar akan dihapus, *user* dapat menekan tombol *Yes*. Setelah itu akan muncul *messagebox* bahwa data sudah terhapus seperti pada gambar 4.8.

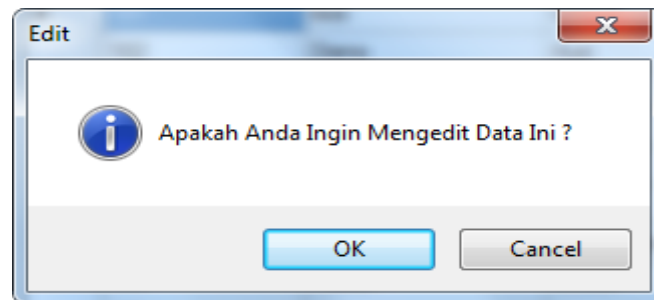


**Gambar 4.14** *Messagebox Data Berhasil Dihapus*

User juga dapat merubah data yang diinginkan dengan menggunakan fitur *edit* data pada *form* kelola *admin*. User dapat memilih data yang akan dirubah dengan memilih kode *user* pada *combobox* atau dapat memilih dengan cara *double click* data yang ingin dirubah pada *datagridview* yang ada pada *form* kelola *admin*. Setelah memilih data yang akan dirubah *user* dapat merubah data dengan menekan tombol *edit*. Setelah tombol *edit* ditekan maka akan

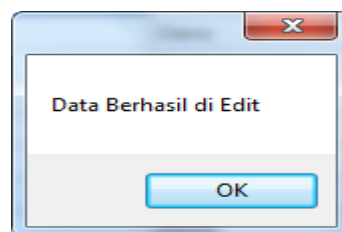


muncul *messagebox* untuk memastikan apakah data yang dipilih benar – benar akan dirubah seperti pada gambar 4.15.



**Gambar 4.15** *Messagebox Alert Edit Data*

Apabila *user* ingin membatalkan data yang akan dirubah, *user* dapat menekan tombol *No*. Jika data yang dipilih benar – benar akan dirubah, *user* dapat menekan tombol *Yes*. Setelah itu akan muncul *messagebox* bahwa data berhasil dirubah seperti pada gambar 4.16.



**Gambar 4.16** *Messagebox Data Berhasil di Edit*

#### **4.3.6 Form Input Data Buyer**

*Form* input data buyer digunakan untuk mengelola semua data *buyer*, mulai dari kode *buyer*, nama *buyer*, nama perusahaan dan negara *buyer*. Tampilan *form input* data *buyer* dapat dilihat pada gambar 4.17.

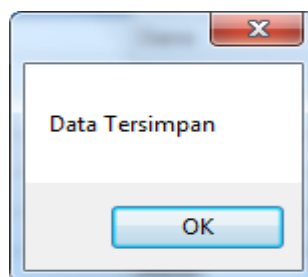
Kode Buyer	Nama Buyer	Nama Perusahaan	Negara
101	Bob	Unique	Australia
102	Diana	Ausi	Australia
103	Dinda	Inc	Denmark
105	Inad	dani	Kroasia
111	Kita	Adalah	Satu
222	Mitha	Kayyu	Indonesia
333	Marc Van Soon	Teak Wood House	Nedherland
444	Marten	Scotlet	Gemary
555	Andhy	Batik Indah	England
666	Nisa	Collped	Itali
777	Kamu	Cinta	Aku
888	Saya	Dan	Kamu
999	Sarah	PT. Maju Terus	Indonesia

**Gambar 4.17** Form Input Data Buyer

Pada form input data buyer terdapat 5 fitur untuk mengolah data yaitu tambah data, hapus data, edit data, clear dan export to excel.

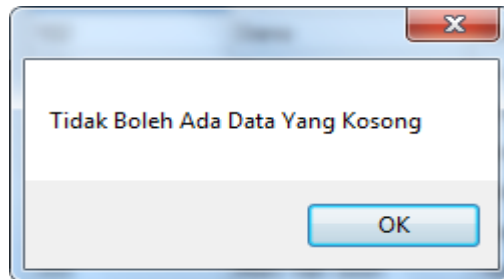
Fitur clear digunakan untuk membersihkan textbox jika data yang dimasukan salah atau tidak sesuai.

Fitur tambah data digunakan untuk memasukan data buyer ke database. User dapat memasukan data dengan mengisi data pada combobox maupun textbox yang tersedia. Setelah semua data terisi user dapat memasukan data ke database dengan memilih tombol tambah pada form input data buyer. Kemudian akan muncul messagebox jika data berhasil disimpan, dapat kita lihat pada gambar 4.18.



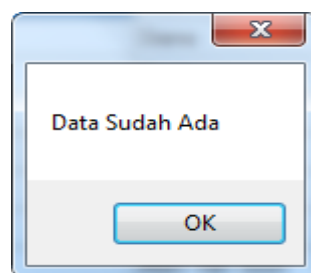
**Gambar 4.18** Messagebox Data Tersimpan

User harus mengisi semua data secara lengkap pada *combobox* maupun *textbox*. Jika terdapat *textbox* atau *combobox* yang kosong maka akan muncul *messagebox* seperti pada gambar 4.19.



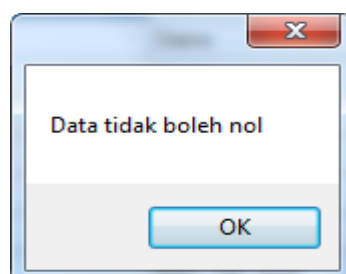
**Gambar 4.19** *Messagebox* data tidak boleh kosong

User tidak boleh memasukan kode *buyer* yang sama pada *combobox*. Jika data kode *buyer* yang dimasukan sama maka akan muncul *messagebox* seperti pada gambar 4.20.



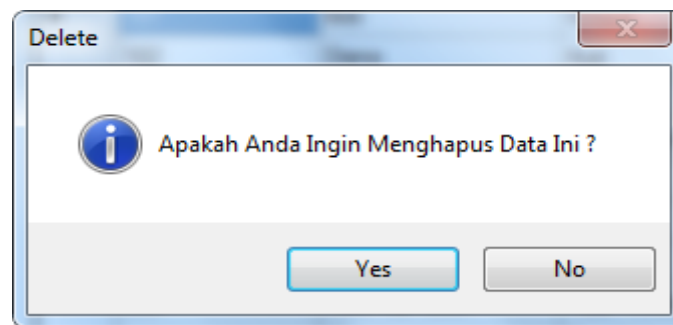
**Gambar 4.20** *Messagebox* data sudah ada

User tidak boleh memasukan data 0 saja pada *combobox* maupun *textbox*. Jika data yang dimasukan 0 saja maka akan muncul *messagebox* seperti pada gambar 4.21.



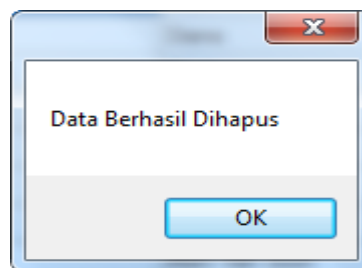
**Gambar 4.21** *Messagebox* data tidak boleh nol

User juga dapat menghapus data yang diinginkan dengan menggunakan fitur hapus data pada *form input* data *buyer*. User dapat memilih data yang akan dihapus dengan memilih kode *buyer* pada *combobox* atau dapat memilih dengan cara *double click* data yang ingin dihapus pada *datagridview* yang ada pada *form input* data *buyer*. Setelah memilih data yang akan dihapus user dapat menghapus data dengan menekan tombol hapus. Setelah tombol hapus ditekan maka akan muncul *messagebox* untuk memastikan apakah data yang dipilih benar – benar akan dihapus seperti pada gambar 4.22.



**Gambar 4.22** *Messagebox* Alert Hapus Data

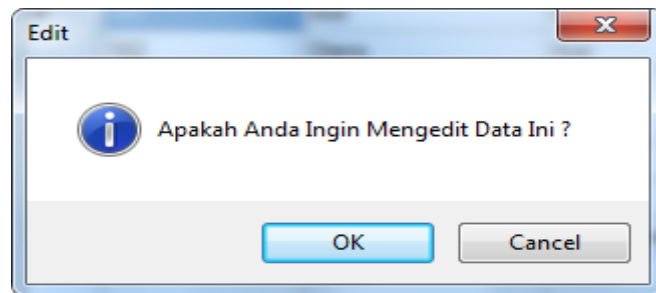
Apabila user ingin membatalkan data yang akan dihapus, user dapat menekan tombol *No*. Jika data yang dipilih benar – benar akan dihapus, user dapat menekan tombol *Yes*. Setelah itu akan muncul *messagebox* bahwa data sudah terhapus seperti pada gambar 4.23.



**Gambar 4.23** *Messagebox* Data Berhasil Dihapus

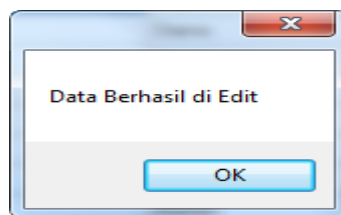
User juga dapat merubah data yang diinginkan dengan menggunakan fitur *edit* data pada *form input* data *buyer*. User dapat memilih data yang akan dirubah dengan memilih kode *buyer* pada *combobox* atau dapat memilih dengan cara *double click* data yang ingin dirubah pada *datagridview* yang ada pada *form input* data *buyer*. Setelah memilih data yang akan dirubah user dapat merubah data dengan menekan tombol *edit*. Setelah tombol *edit*

ditekan maka akan muncul *messagebox* untuk memastikan apakah data yang dipilih benar – benar akan dirubah seperti pada gambar 4.24.



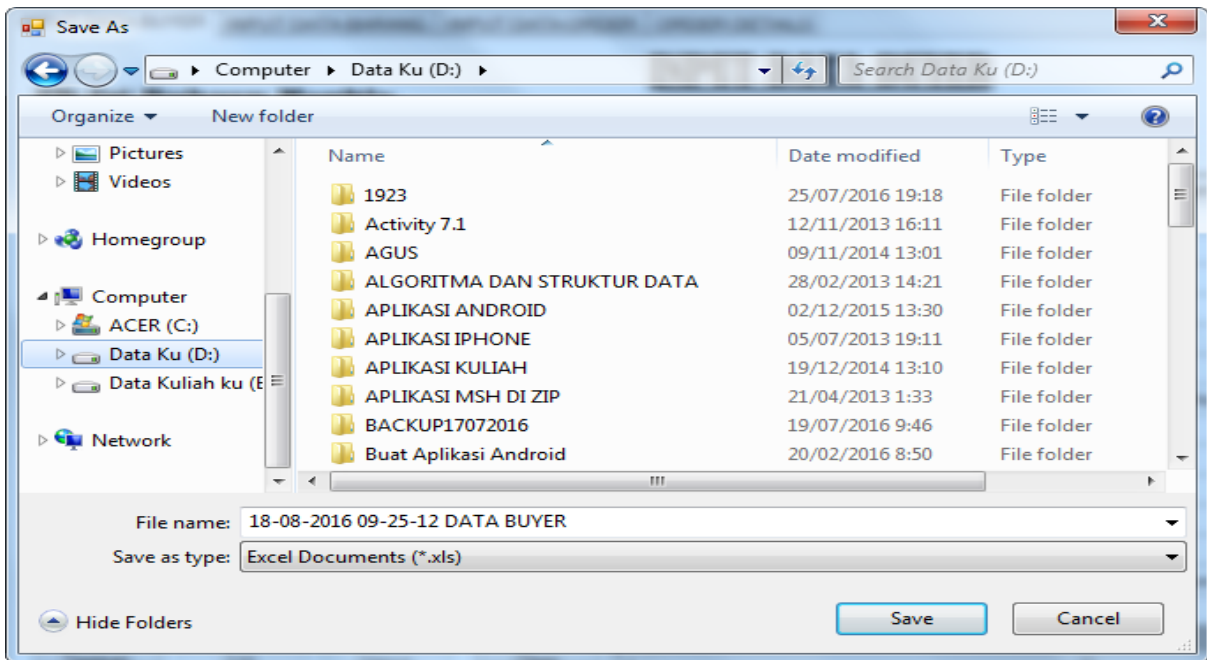
**Gambar 4.24** *Messagebox Alert Edit Data*

Apabila *user* ingin membatalkan data yang akan dirubah, *user* dapat menekan tombol *No*. Jika data yang dipilih benar – benar akan dirubah, *user* dapat menekan tombol *Yes*. Setelah itu akan muncul *messagebox* bahwa data berhasil dirubah seperti pada gambar 4.25.



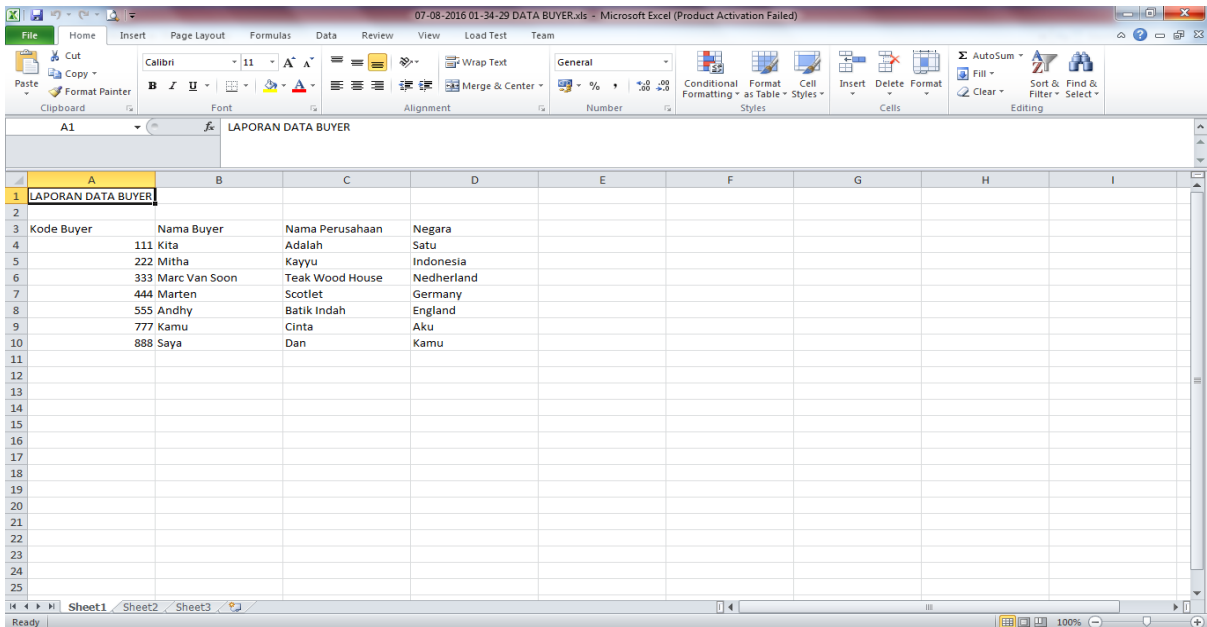
**Gambar 4.25** *Messagebox Data Berhasil di Edit*

Selain itu pada *form* input data *buyer* juga terdapat fitur *export to excel*. Fitur ini dapat digunakan untuk mengexport data dari *datagridview* menjadi file *xls* data *Microsoft Excel*. *User* dapat menekan tombol *export to excel* setelah itu pilih folder dimana akan menyimpan file seperti pada gambar 4.26.



**Gambar 4.26** Save File Ms. Excel

User dapat merubah data file *xls* dengan membuka file yang sudah disimpan sebelumnya melalui aplikasi *Microsoft Excel*. Seperti pada gambar 4.27.



**Gambar 4.27** File Microsoft Excel

### 4.3.7 Form Input Data Barang

Form input data barang digunakan untuk mengelola semua data barang, mulai dari kode barang, nama barang, jenis finishing barang, ukuran barang ( panjang, lebar dan tinggi ) dan harga barang. Tampilan form input data barang dapat dilihat pada gambar 4.28.

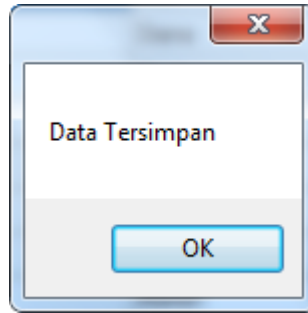
Kode Barang	Nama Barang	Finishing	Panjang	Lebar
M001	Cabinet 2 Drawer	Natural	200	110
M002	Cabinet 3 Drawer	Natural	225	110
M003	Bedside 2 Drawer	Old Teak	210	80
M004	Bedside 5 Drawer	New Teak	300	200
M005	Table Top 5 cm	Old Teak	300	110
M006	Bar Chair	Old Teak	200	120
M007	Bar Chair	New Teak	100	40
M008	Kaa	ds	43	55

Gambar 4.28 Form Input Data Barang

Pada form input data barang terdapat 5 fitur untuk mengolah data yaitu tambah data, hapus data, edit data, clear dan export to excel.

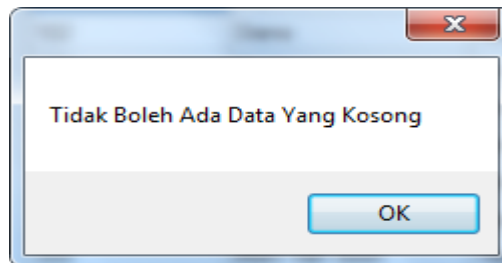
Fitur clear digunakan untuk membersihkan textbox jika data yang dimasukan salah atau tidak sesuai.

Fitur tambah data digunakan untuk memasukan data barang ke database. User dapat memasukan data dengan mengisi data pada combobox maupun textbox yang tersedia. Setelah semua data terisi user dapat memasukan data ke database dengan memilih tombol tambah pada form input data barang. Kemudian akan muncul messagebox jika data berhasil disimpan, dapat kita lihat pada gambar 4.29.



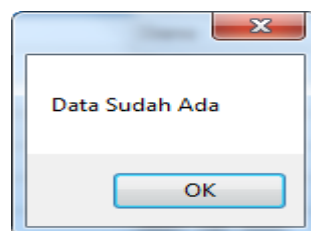
**Gambar 4.29** *Messagebox* Data Tersimpan

*User* harus mengisi semua data secara lengkap pada *combobox* maupun *textbox*. Jika terdapat *textbox* atau *combobox* yang kosong maka akan muncul *messagebox* seperti pada gambar 4.30.



**Gambar 4.30** *Messagebox* data tidak boleh kosong

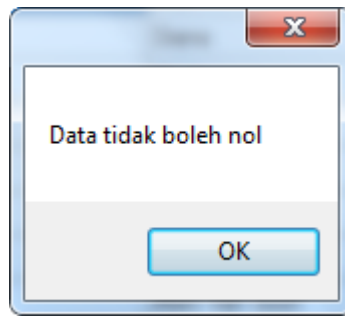
*User* tidak boleh memasukan kode barang yang sama pada *combobox*. Jika data kode barang yang dimasukan sama maka akan muncul *messagebox* seperti pada gambar 4.31.



**Gambar 4.31** *Messagebox* data sudah ada

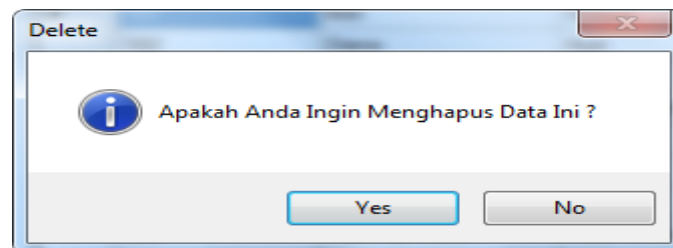
*User* tidak boleh memasukan data 0 saja pada *combobox* maupun *textbox*. Jika data yang dimasukan 0 saja maka akan muncul *messagebox* seperti pada gambar 4.32.





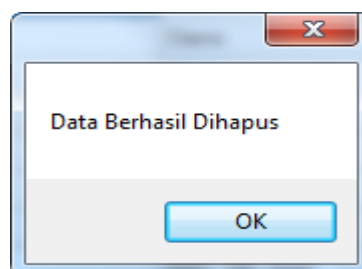
**Gambar 4.32** *Messagebox* data tidak boleh nol

User juga dapat menghapus data yang diinginkan dengan menggunakan fitur hapus data pada *form* input data barang. User dapat memilih data yang akan dihapus dengan memilih kode barang pada *combobox* atau dapat memilih dengan cara *double click* data yang ingin dihapus pada *datagridview* yang ada pada *form* input data barang. Setelah memilih data yang akan dihapus user dapat menghapus data dengan menekan tombol hapus. Setelah tombol hapus ditekan maka akan muncul *messagebox* untuk memastikan apakah data yang dipilih benar – benar akan dihapus seperti pada gambar 4.33.



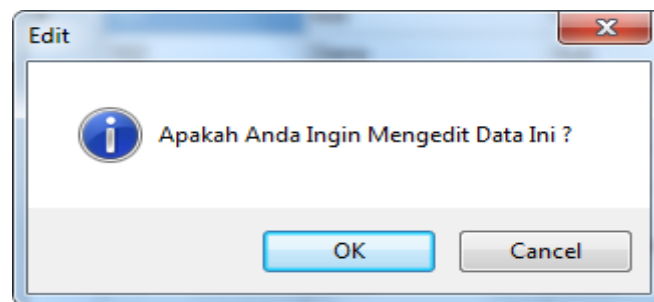
**Gambar 4.33** *Messagebox Alert* Hapus Data

Apabila user ingin membatalkan data yang akan dihapus, user dapat menekan tombol *No*. Jika data yang dipilih benar – benar akan dihapus, user dapat menekan tombol *Yes*. Setelah itu akan muncul *messagebox* bahwa data sudah terhapus seperti pada gambar 4.34.



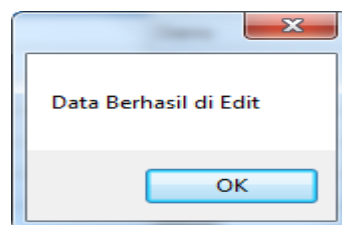
**Gambar 4.34** *Messagebox* Data Berhasil Dihapus

User juga dapat merubah data yang diinginkan dengan menggunakan fitur *edit* data pada *form input* data barang. User dapat memilih data yang akan dirubah dengan memilih kode barang pada *combobox* atau dapat memilih dengan cara *double click* data yang ingin dirubah pada *datagridview* yang ada pada *form input* data barang. Setelah memilih data yang akan dirubah user dapat merubah data dengan menekan tombol *edit*. Setelah tombol *edit* ditekan maka akan muncul *messagebox* untuk memastikan apakah data yang dipilih benar – benar akan dirubah seperti pada gambar 4.35.



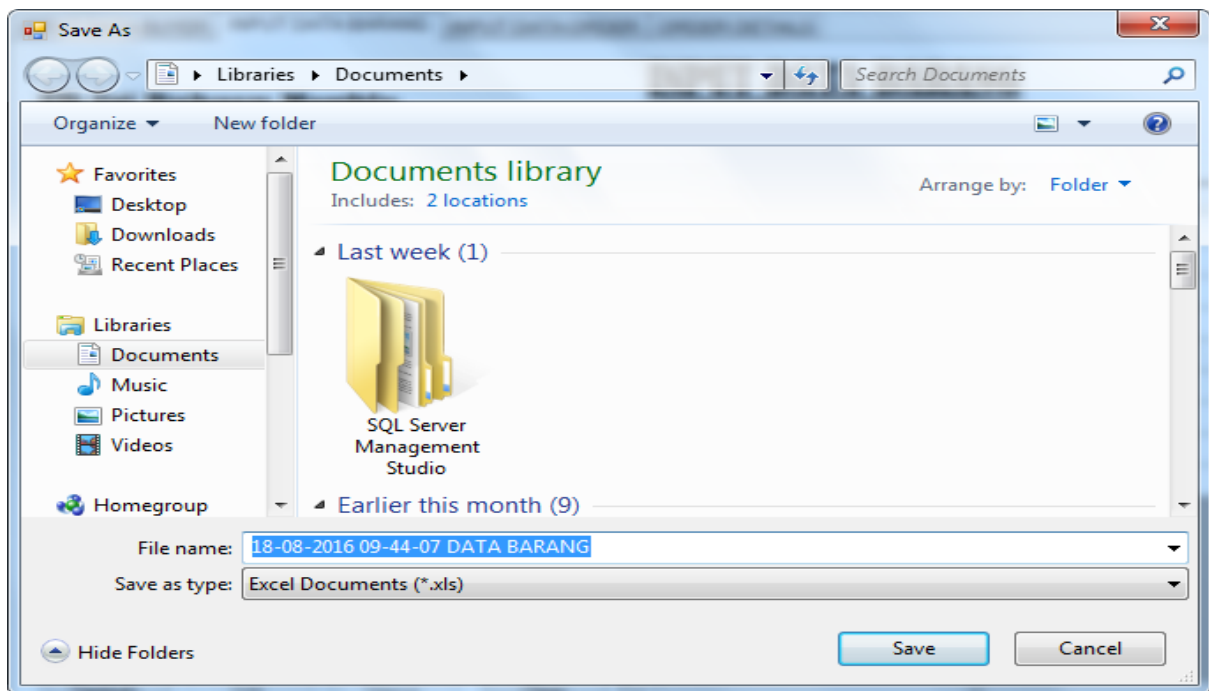
**Gambar 4.35** *MessageBox Alert Edit Data*

Apabila *user* ingin membatalkan data yang akan dirubah, *user* dapat menekan tombol *No*. Jika data yang dipilih benar – benar akan dirubah, *user* dapat menekan tombol *Yes*. Setelah itu akan muncul *messagebox* bahwa data berhasil dirubah seperti pada gambar 4.36.



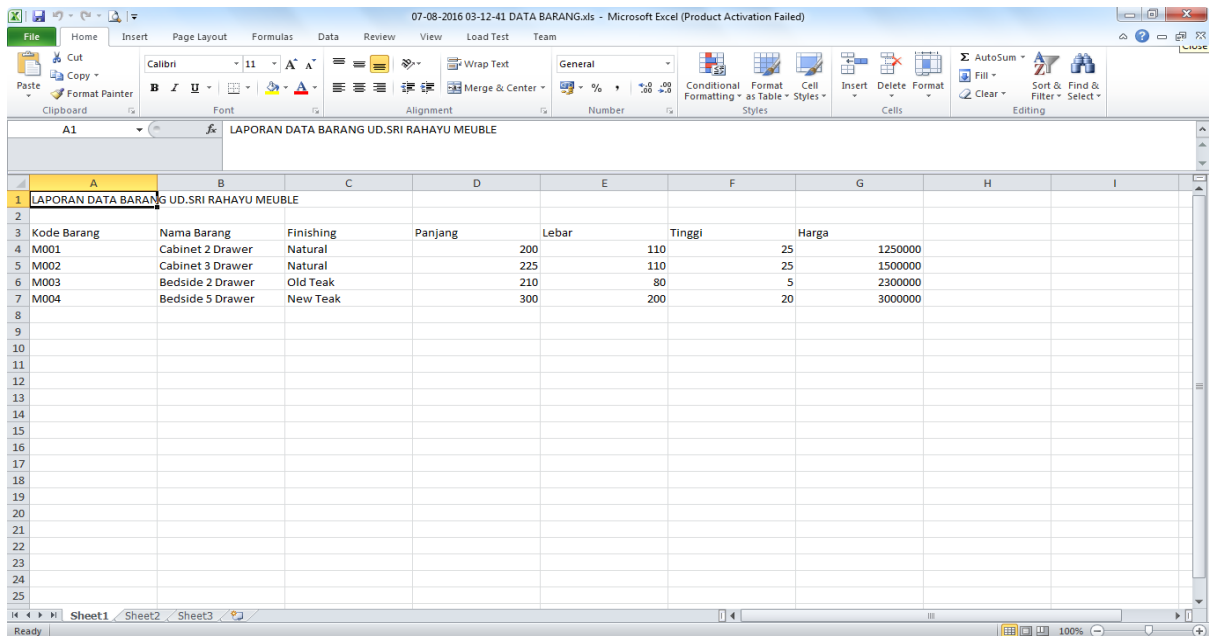
**Gambar 4.36** *MessageBox Data Berhasil di Edit*

Selain itu pada *form input* data barang juga terdapat fitur *export to excel*. Fitur ini dapat digunakan untuk mengexport data dari *datagridview* menjadi file *xls* data *Microsoft Excel*. User dapat menekan tombol *export to excel* setelah itu pilih folder dimana akan menyimpan *file* seperti pada gambar 4.37.



**Gambar 4.37** *Save File Ms. Excel*

User dapat merubah data file *xls* dengan membuka file yang sudah disimpan sebelumnya melalui aplikasi *Microsoft Excel*. Seperti pada gambar 4.38.



**Gambar 4.38** *File Microsoft Excel*

### 4.3.8 Form Input Data Order

Form input data order digunakan untuk mengelola semua data order, mulai dari kode order, kode *buyer*, nama *buyer* pemesan, kode barang yang dipesan, nama barang yang dipesan, jenis *finishing* barang, ukuran barang ( panjang, lebar dan tinggi ), harga barang per pcs, harga total barang dan kontainer tanggal keberangkatan. Tampilan *form* input data order dapat dilihat pada gambar 4.39.

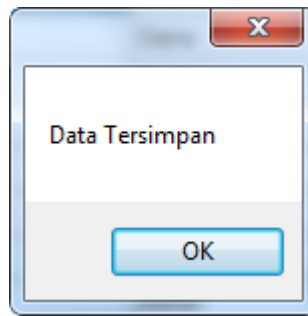
Kode Order	Kode Buyer	Nama Buyer	Kode Barang	Nama Barang
A001	222	Mitha	M003	Bedside 2 Drawer
A002	444	Marten	M004	Bedside 5 Drawer
A003	888	Saya	M002	Cabinet 3 Drawer
A004	333	Marc Van Soon	M003	Bedside 2 Drawer
A005	222	Mitha	M002	Cabinet 3 Drawer
A006	222	Mitha	M003	Bedside 2 Drawer
A007	222	Mitha	M003	Bedside 2 Drawer
A008	222	Mitha	M002	Cabinet 3 Drawer
A009	222	Mitha	M004	Bedside 5 Drawer
A010	333	Marc Van Soon	M002	Cabinet 3 Drawer
A011	444	Marten	M004	Bedside 5 Drawer
A012	444	Marten	M003	Bedside 2 Drawer
A013	888	Saya	M004	Bedside 5 Drawer
A014	777	Kamu	M003	Bedside 2 Drawer

Gambar 4.39 Form Input Data Order

Pada *form* input data order terdapat 5 fitur untuk mengolah data yaitu tambah data, hapus data, *edit* data, *clear* dan *export to excel*.

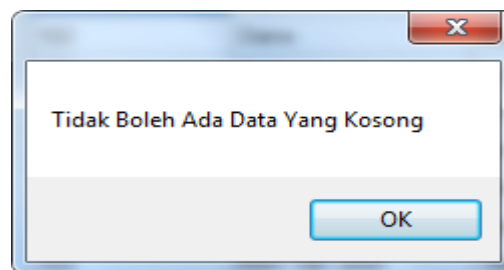
Fitur *clear* digunakan untuk membersihkan *textbox* jika data yang dimasukan salah atau tidak sesuai.

Fitur tambah data digunakan untuk memasukan data order ke *database*. *User* dapat memasukan data dengan mengisi data pada *combobox* maupun *textbox* yang tersedia. Setelah semua data terisi *user* dapat memasukan data ke *database* dengan memilih tombol tambah pada *form input* data order. Kemudian akan muncul *messagebox* jika data berhasil disimpan, dapat kita lihat pada gambar 4.40.



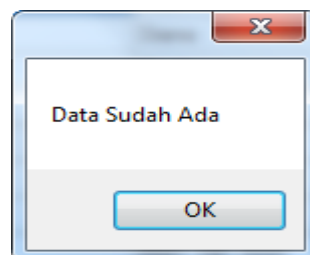
**Gambar 4.40** *Messagebox* Data Tersimpan

User harus mengisi semua data secara lengkap pada *combobox* maupun *textbox*. Jika terdapat *textbox* atau *combobox* yang kosong maka akan muncul *messagebox* seperti pada gambar 4.41.



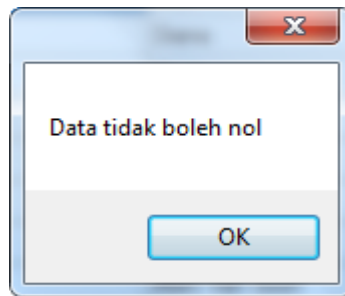
**Gambar 4.41** *Messagebox* data tidak boleh kosong

User tidak boleh memasukan kode order yang sama pada *combobox*. Jika data kode order yang dimasukan sama maka akan muncul *messagebox* seperti pada gambar 4.42.



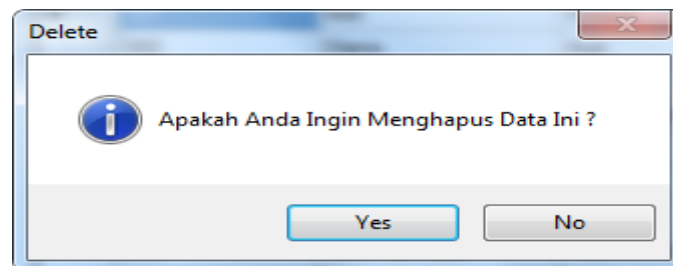
**Gambar 4.42** *Messagebox* data sudah ada

User tidak boleh memasukan data 0 saja pada *combobox* maupun *textbox*. Jika data yang dimasukan 0 saja maka akan muncul *messagebox* seperti pada gambar 4.43.



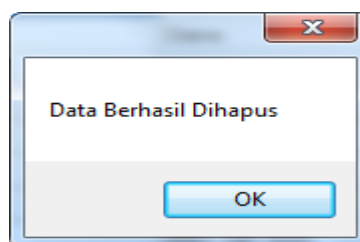
**Gambar 4.43** *Messagebox* data tidak boleh nol

*User* juga dapat menghapus data yang diinginkan dengan menggunakan fitur hapus data pada *form input* data order. *User* dapat memilih data yang akan dihapus dengan memilih kode order pada *combobox* atau dapat memilih dengan cara *double click* data yang ingin dihapus pada *datagridview* yang ada pada *form input* data order. Setelah memilih data yang akan dihapus *user* dapat menghapus data dengan menekan tombol hapus. Setelah tombol hapus ditekan maka akan muncul *messagebox* untuk memastikan apakah data yang dipilih benar – benar akan dihapus seperti pada gambar 4.44.



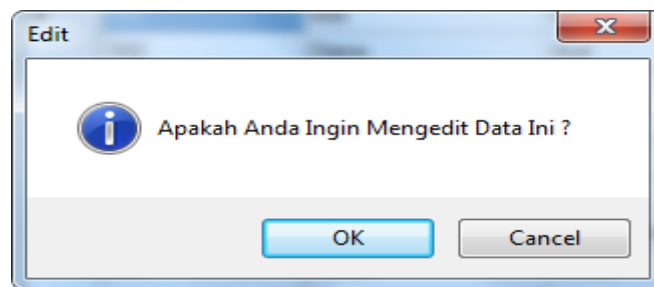
**Gambar 4.44** *Messagebox Alert* Hapus Data

Apabila *user* ingin membatalkan data yang akan dihapus, *user* dapat menekan tombol *No*. Jika data yang dipilih benar – benar akan dihapus, *user* dapat menekan tombol *Yes*. Setelah itu akan muncul *messagebox* bahwa data sudah terhapus seperti pada gambar 4.45.



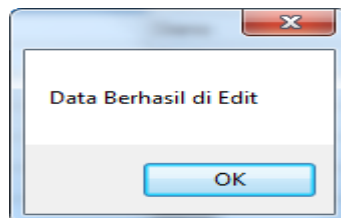
**Gambar 4.45** *Messagebox* Data Berhasil Dihapus

User juga dapat merubah data yang diinginkan dengan menggunakan fitur *edit* data pada *form input* data order. User dapat memilih data yang akan dirubah dengan memilih kode order pada *combobox* atau dapat memilih dengan cara *double click* data yang ingin dirubah pada *datagridview* yang ada pada *form input* data order. Setelah memilih data yang akan dirubah user dapat merubah data dengan menekan tombol *edit*. Setelah tombol *edit* ditekan maka akan muncul *messagebox* untuk memastikan apakah data yang dipilih benar – benar akan dirubah seperti pada gambar 4.46.



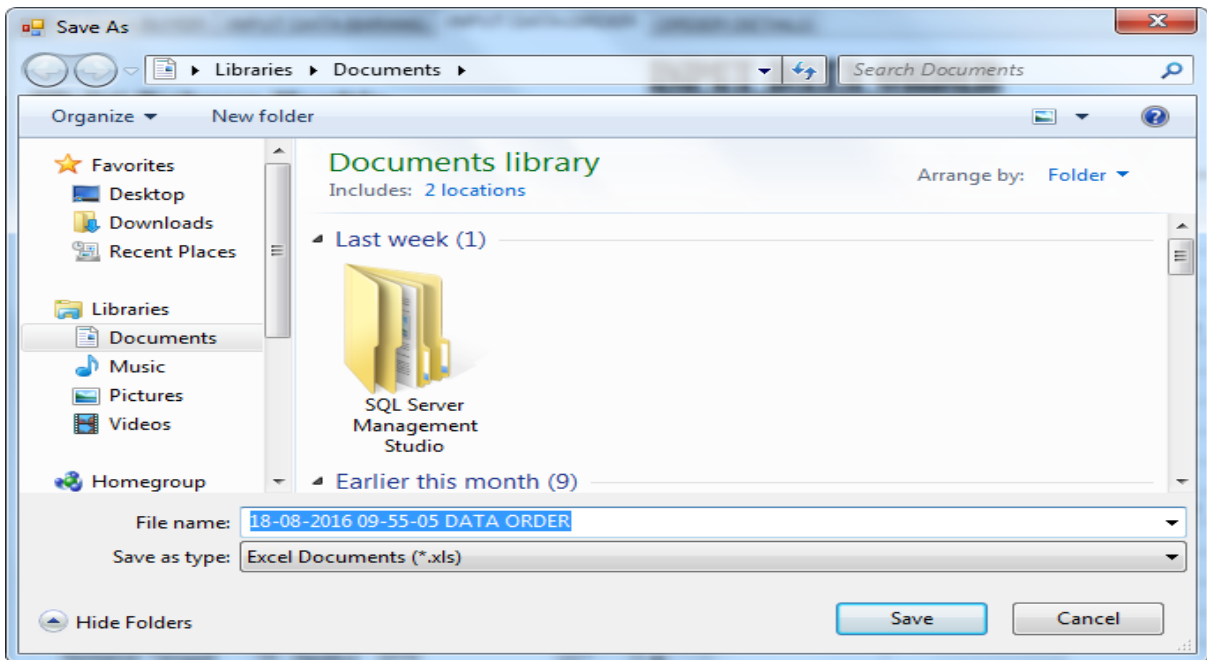
**Gambar 4.46** *Messagebox Alert Edit Data*

Apabila user ingin membatalkan data yang akan dirubah, user dapat menekan tombol *No*. Jika data yang dipilih benar – benar akan dirubah, user dapat menekan tombol *Yes*. Setelah itu akan muncul *messagebox* bahwa data berhasil dirubah seperti pada gambar 4.47.



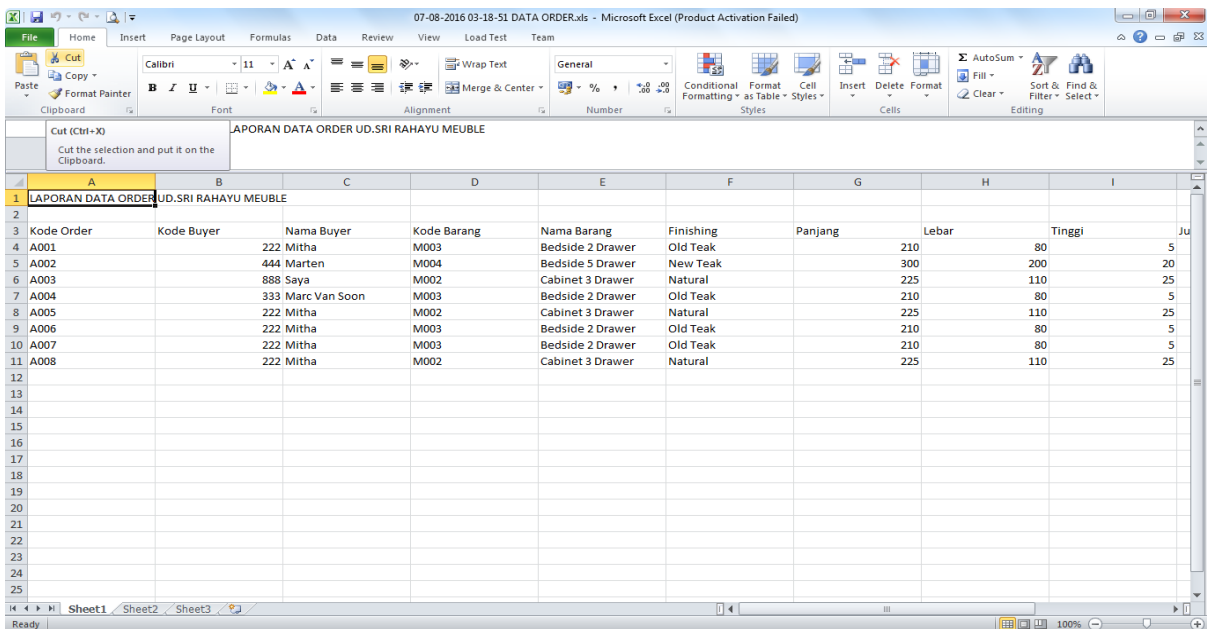
**Gambar 4.47** *Messagebox Data Berhasil di Edit*

Selain itu pada *form input* data barang juga terdapat fitur *export to excel*. Fitur ini dapat digunakan untuk mengexport data dari *datagridview* menjadi file *xls* data *Microsoft Excel*. User dapat menekan tombol *export to excel* setelah itu pilih folder dimana akan menyimpan *file* seperti pada gambar 4.48.



**Gambar 4.48** Save File Ms. Excel

User dapat merubah data file *xls* dengan membuka *file* yang sudah disimpan sebelumnya melalui aplikasi *Microsoft Excel*. Seperti pada gambar 4.49.

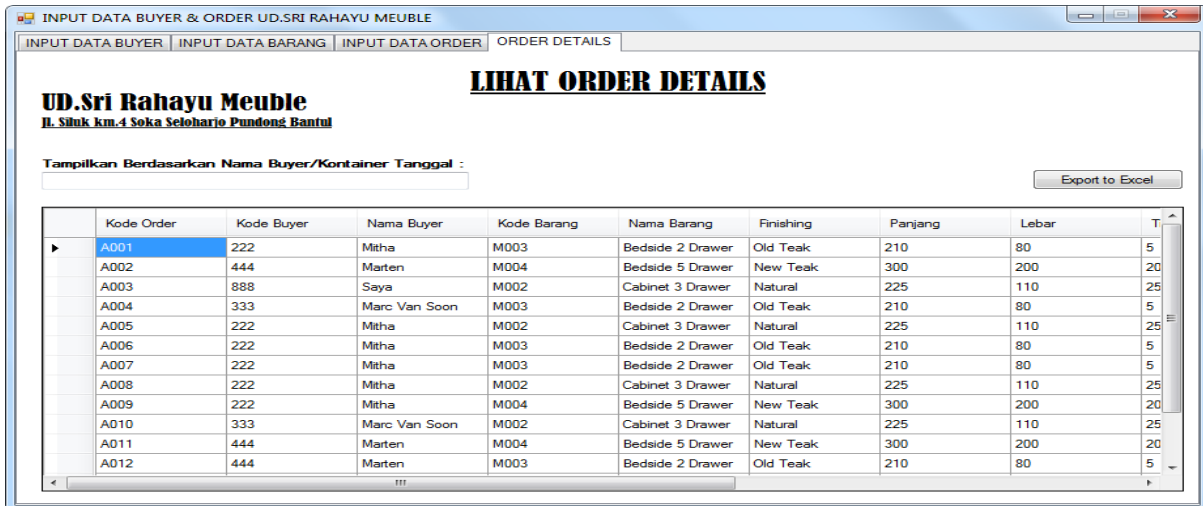


**Gambar 4.49** File Microsoft Excel



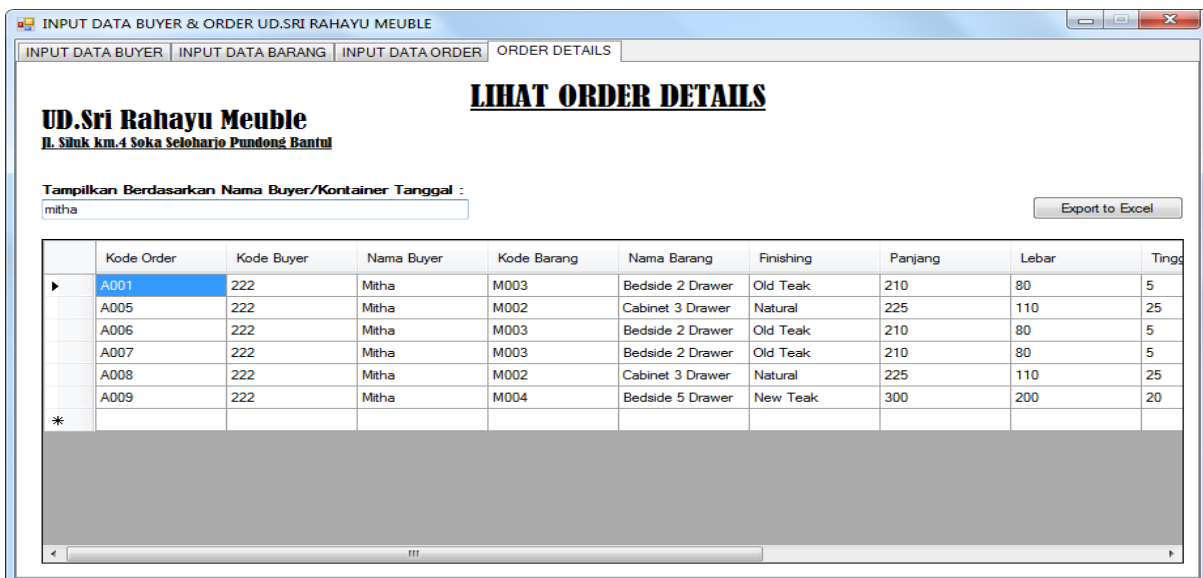
### 4.3.9 Form Lihat Order Details

Form lihat order details digunakan untuk mengelola data order berdasarkan kata kunci yang diinginkan yaitu menurut nama *buyer* dan kontainer tanggal menggunakan *textbox*. Tampilan *form* lihat data order details dapat dilihat pada gambar 4.50.



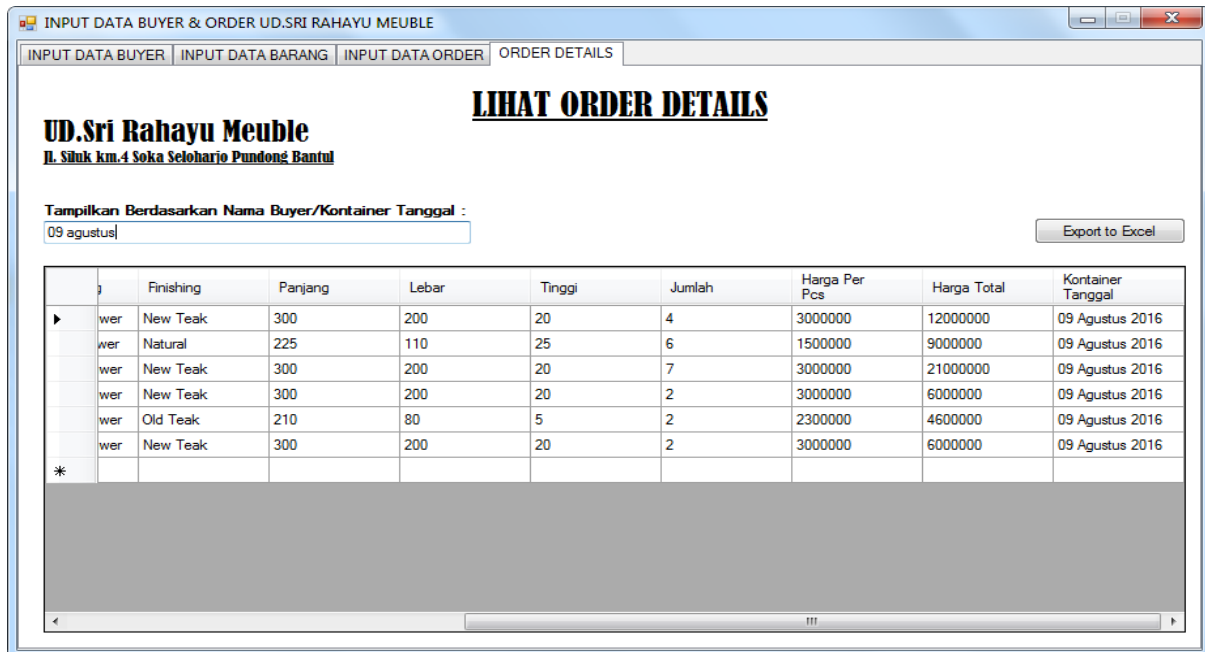
Gambar 4.50 Form Lihat Order Details

Pada *textbox filtering data form* lihat order details dimasukan kata kunci menurut nama buyer Mitha. Kemudian pada *datagridview* data yang ditampilkan merupakan data yang mempunyai nama *buyer* Mitha. Tampilan *filtering data* menurut nama *buyer* dapat dilihat pada gambar 4.51.



Gambar 4.51 Filtering data menurut nama

Pada *textbox filtering data form* lihat order *details* dimasukan kata kunci menurut kontainer tanggal 09 Agustus 2016. Kemudian pada *datagridview* data yang ditampilkan merupakan data yang mempunyai kontainer tanggal 09 Agustus 2016. Tampilan *filtering data* menurut kontainer tanggal dapat dilihat pada gambar 4.52.



**Gambar 4.52** *Filtering data* menurut kontainer tanggal

## 4.4 Pengujian Sistem

Dalam pengujian sistem aplikasi yang dibuat terdapat tiga jenis pengujian yaitu pengujian *user interface*, pengujian dasar sistem, dan pengujian validasi.

### 4.4.1 Pengujian *User Interface*

Pengujian *user interface* bertujuan untuk mengetahui fungsionalitas dari elemen-elemen interface yang terdapat di dalam halaman sistem. Elemen yang diujikan adalah elemen *button* dan *textbox filtering* pada aplikasi. Hasil pengujian dapat dilihat pada Tabel 4.1.

**Tabel 4.1** Hasil Pengujian *User Interface*

No	Kasus yang Diuji	Hasil yang Diharapkan	Hasil yang Didapatkan	Status
1.	Tombol Hapus	Sistem dapat menghapus item yang dipilih	Setelah tombol hapus diklik, sistem menghapus data yang dipilih	Berhasil
3.	Tombol <i>Edit</i>	Sistem dapat mengedit data yang diinginkan	Setelah tombol <i>edit</i> diklik, sistem mengedit data yang dipilih	Berhasil
4.	Tombol Tambah	Sistem dapat menambahkan data pada tabel <i>buyer</i> barang dan order1	Setelah tombol tambah diklik, data yang ditambahkan dapat bertambah pada table	Berhasil
5.	Tombol <i>Clear</i>	Sistem dapat menghapus <i>text</i> pada <i>textbox</i>	Setelah tombol <i>clear</i> diklik, <i>text</i> pada <i>textbox</i> dapat dikosongkan	Berhasil
6.	<i>Textbox Filtering</i>	Sistem dapat memproses fungsi <i>textbox filtering</i> data sesuai data yang diinginkan	Sistem menjalankan fungsi <i>textbox filtering</i> sesuai data yang diinginkan	Berhasil
7.	Tombol <i>Export to Excel</i>	Sistem dapat mengexport data dari <i>datagridview</i> ke Ms. Excel	Setelah tombol <i>export to excel</i> diklik, data dari <i>datagridview</i> di <i>export</i> ke Ms. Excel	Berhasil

Dari hasil pengujian *user interface*, tombol hapus, *edit*, tambah, *clear* dan *export to excel* serta *textbox filtering* data dapat berfungsi sebagaimana yang dirancang.

#### 4.4.2 Pengujian Dasar Sistem

Pengujian fungsi dasar sistem bertujuan untuk mengetahui fungsi-fungsi dasar yang ada di dalam aplikasi. Hasil pengujian dapat dilihat pada Tabel 4.2.

**Tabel 4.2** Hasil Pengujian Dasar Sistem

No.	Kasus Yang Diuji	Hasil Yang Diharapkan	Hasil Yang Didapatkan	Status
1.	Pengujian fungsi hitung harga total pada <i>textbox</i> harga total <i>form input</i> data order	Sistem dapat menghitung secara otomatis harga total suatu barang setelah memasukan data jumlah barang	Sistem otomatis menghitung harga total suatu barang setelah memasukan jumlah barang	Berhasil
2.	Pengujian fungsi tambah data	Sistem dapat menyimpan data yang ditambah ke dalam <i>database</i>	Data yang ditambah sudah tersimpan ke dalam <i>database</i>	Berhasil
3.	Pengujian fungsi ubah data	Sistem dapat mengubah data di dalam <i>database</i>	Di dalam <i>database</i> , data sudah berubah	Berhasil
4.	Pengujian fungsi hapus data	Sistem dapat menghapus data di dalam <i>database</i>	Di dalam <i>database</i> , data sudah tidak ada	Berhasil
5.	Pengujian fungsi <i>export to excel</i>	Sistem dapat mengexport data dari <i>database</i> ke Ms. Excel	Sistem mengexport data ke Ms. Excel sesuai kebutuhan	Berhasil
6.	Pengujian fungsi <i>filtering</i> data	Sistem dapat memfilter data yang ditampilkan pada <i>datagridview</i> lihat <i>order details</i>	Sistem menampilkan data pada <i>datagridview</i> sesuai keinginan	Berhasil

Dari hasil pengujian sistem dari fungsi hitung harga total pada *textbox* harga total *form input* data order, pengujian fungsi tambah data, pengujian fungsi ubah data, pengujian fungsi hapus data, pengujian fungsi *export to excel* dan pengujian fungsi *filtering* data, sistem dapat berfungsi sebagaimana yang dirancang.

#### 4.4.3 Pengujian Validasi

Pengujian validasi bertujuan untuk mengetahui apakah validasi-validasi yang di dalam sistem sudah berjalan dengan baik. Hasil pengujian dapat dilihat pada Tabel 4.3.

**Tabel 4.3** Hasil Pengujian Validasi

No	Kasus Yang Diuji	Hasil Yang Diharapkan	Hasil Yang Didapatkan	Status
1.	Validasi jika data <i>textbox</i> belum terisi atau kosong	Sistem dapat menampilkan pesan peringatan bahwa tidak boleh ada data yang kosong	Muncul pesan peringatan tidak boleh ada data yang kosong	Berhasil
2.	Validasi jika data yang dimasukan 0 atau nol	Sistem dapat menampilkan pesan peringatan bahwa data yang dimasukan tidak boleh 0 atau nol	Muncul pesan peringatan yang memberitahukan bahwa tidak boleh ada data 0 atau nol	Berhasil
3.	Validasi jika data dimasukan sudah ada	Sistem dapat menampilkan pesan peringatan bahwa data yang dimasukan sudah ada pada <i>database</i>	Muncul pesan peringatan yang memberitahukan bahwa data yang dimasukan sudah ada	Berhasil
4.	Validasi <i>textbox</i> jumlah tidak boleh diisi selain angka	Sistem hanya dapat menerima <i>input</i> data angka	<i>Textbox</i> tidak merespon input selain angka	Berhasil

Dari hasil pengujian validasi mulai dari validasi jika data *textbox* belum terisi atau kosong, validasi jika data yang dimasukan 0 atau nol, validasi jika data dimasukan sudah ada dan validasi *textbox* jumlah tidak boleh diisi selain angka dapat berfungsi sebagaimana yang dirancang.

#### 4.5 Hasil Kuesioner

Semua data yang diperoleh melalui kuesioner, selanjutnya akan diolah menjadi sebuah informasi. Responden yang dilibatkan dalam pengambilan data sebanyak 3 orang yaitu karyawan kantor UD. Sri Rahayu Meuble.

Dari hasil jawaban kuesioner ini nantinya akan disimpulkan hasil dari uji kelayakan aplikasi yang dibuat. Hasil jawaban kuesioner dapat dilihat pada tabel 4.4.

**Tabel 4.4** Hasil Jawaban Kuesioner

No.	Pertanyaan	Jawaban	
		Ya	Tidak
1.	Apakah aplikasi Pengolahan Data Order UD. Sri Rahayu Meuble yang dibuat mudah untuk digunakan ?	3	-
2.	Apakah aplikasi Pengolahan Data Order UD. Sri Rahayu Meuble yang dibuat sudah mampu melakukan proses pengolahan data (simpan, edit, hapus, lihat data) secara tepat ?	3	-
3.	Apakah aplikasi Pengolahan Data Order UD. Sri Rahayu Meuble yang dibuat mampu mempercepat proses pengolahan data order ?	2	1
4.	Apakah aplikasi Pengolahan Data Order UD. Sri Rahayu Meuble yang dibuat dapat memudahkan proses pengolahan data order ?	3	-
5.	Apakah tampilan aplikasi Pengolahan Data Order UD. Sri Rahayu Meuble yang dibuat sudah menarik ?	2	1

Dari hasil kuesioner pada tabel 4.4 diperoleh hasil yaitu pada pertanyaan nomor 1, 2 dan 4, semua responden memberikan jawaban Ya. Dan pada pertanyaan nomor 3 dan 5, responden menjawab 2 Ya dan 1 Tidak. Sehingga dari jawaban tersebut dapat disimpulkan bahwa aplikasi yang dibuat mudah untuk digunakan, aplikasi yang dibuat sudah mampu melakukan proses pengolahan data secara tepat, aplikasi yang dibuat mampu mempercepat dan memudahkan proses pengolahan data order dan aplikasi yang dibuat memiliki tampilan yang sudah cukup menarik.

## 4.6 Pembahasan

Sesuai dengan tujuan penelitian penulis yaitu membuat sebuah aplikasi yang dapat membantu proses pengolahan data order untuk memudahkan dan mempercepat proses pengolahan data dan memudahkan dalam pembuatan laporan data order dan menyimpan data pada *database server*.

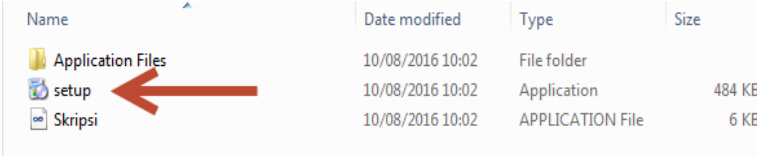
Aplikasi yang dibuat dapat mengolah data *buyer*, data barang dan data order, mulai dari *input* data, hapus data dan *edit* data. Aplikasi yang dibuat juga dapat menampilkan laporan berupa tabel di *Microsoft Excel* seperti laporan data order pada gambar 4.34.

Aplikasi yang dibuat juga mampu menyimpan data order mulai dari data *buyer*, data barang dan data order ke *database server*.

## 4.7 Distribusi Aplikasi

Sebelum di gunakan, aplikasi di instal terlebih dahulu dengan langkah-langkah sebagai berikut:

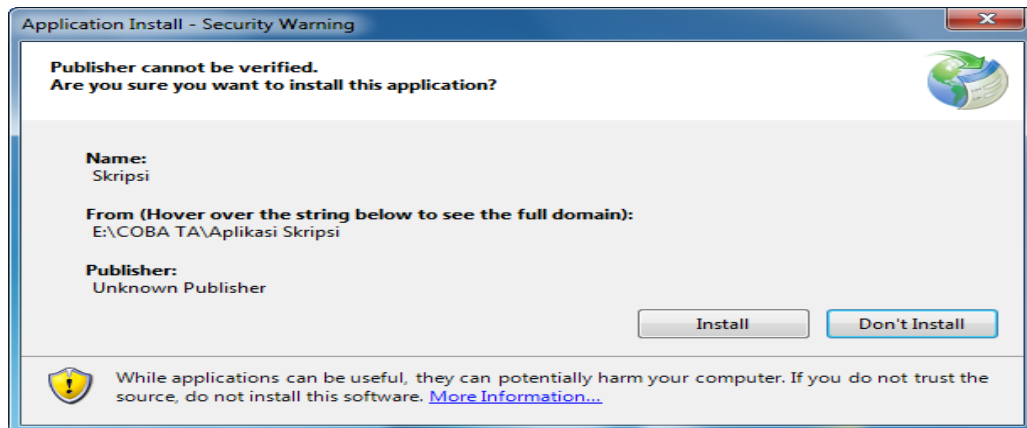
- 1) Pastikan konfigurasi *database* sama dengan nama servernya sebelum aplikasi *dipublish*.
- 2) Setelah aplikasi dipublish tekan 2x *file setup .exe* seperti pada Gambar 4.53.



Name	Date modified	Type	Size
Application Files	10/08/2016 10:02	File folder	
setup	10/08/2016 10:02	Application	484 KB
Skripsi	10/08/2016 10:02	APPLICATION File	6 KB

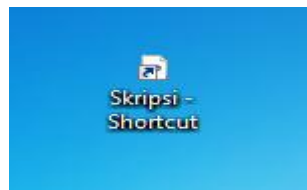
**Gambar 4.53** File .exe

- 3) Setelah itu, akan muncul seperti Gambar 4.54 kemudian pengguna menekan *Install*. Apabila pengguna akan membatalkan instalasi dapat dengan menekan tombol *Don't Install*.



**Gambar 4. 54** Kotak Dialog Instalasi

- 4) Setelah menekan *Install* pada Gambar 4.54, maka aplikasi tersebut secara otomatis langsung terinstal.
- 5) Pengguna dapat *right click* aplikasi *skripsi .exe* kemudian *Create Shortcut*, maka ikon aplikasi akan muncul pada *desktop* komputer seperti Gambar 4.55.



**Gambar 4.55** Ikon Aplikasi

#### **4.8 Keterbatasan Perangkat Lunak**

Dari hasil pembahasan aplikasi memiliki beberapa fitur mulai dari *input* data, hapus dan ubah data. Aplikasi yang dibuat juga dapat menampilkan laporan berupa data yang di *export* ke Microsoft Excel. Akan tetapi aplikasi yang dibuat memiliki keterbatasan atau kekurangan yaitu terbatasnya fitur yang ada pada aplikasi, desain *interface* yang masih biasa dan laporan yang dihasilkan dalam format *xls* dirasa masih harus diperbaiki melalui Microsoft Excel.