

BAB III

METODE PENELITIAN

3.1 Alat dan Bahan

3.1.1 Alat

Dalam penelitian ini, alat yang di gunakan adalah sebagai berikut:

1. Perangkat Keras (*Hardware*)

- a) *Personal Computer (PC)/Laptop 32/64 bit architecture processor, 32/64 bit architecture processor.*
- b) *Printer*

2. Perangkat Lunak (*Software*)

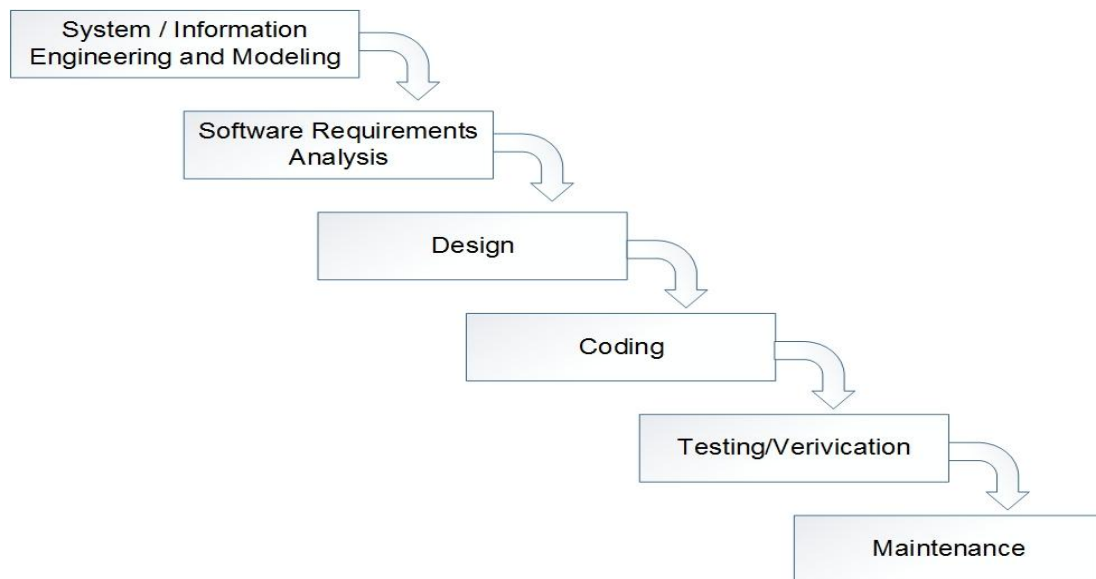
- a) Sistem Operasi Windows 7
- b) Microsoft Visual Studio 2010
- c) Microsoft SQL Server
- d) Microsoft Excel 2010

3.1.2 Bahan

- a. Data pada studi literature yang telah dibuat pada penelitian sebelumnya yang mempunyai keterkaitan dan sesuai dengan aplikasi pada sistem aplikasi yang dibuat.
- b. Data barang, data buyer dan data order UD. Sri Rahayu Meuble.
- c. Data UD. Sri Rahayu Meuble yang mendukung dibuatnya aplikasi.

3.2 Alur Penelitian

Tahapan penelitian yang digunakan menggunakan metode *waterfall*. Berikut ini merupakan tahapan *SDLC Waterfall* yang digunakan akan disajikan dengan gambar 3.1:



Gambar 3.1 SDLC Waterfall

a. *System or Information Engineering and Modeling.*

Tahap pertama diawali dengan mencari kebutuhan dari keseluruhan sistem yang akan diaplikasikan ke dalam bentuk *software*. Hal ini perlu dilakukan karena *software* harus dapat berinteraksi dengan elemen-elemen yang lain seperti *hardware* dan *database*.

b. *Software Requirements Analysis*

Proses *Software Requirements Analysis* difokuskan pada *software*. Tujuannya adalah mengetahui sifat dari program yang akan dibuat, seperti fungsi yang dibutuhkan dan *user interface*.

c. *Design*

Proses *design* dilakukan dengan tujuan untuk mengubah kebutuhan-kebutuhan diatas menjadi “*blueprint*” sebelum memulai *coding*.

d. *Coding*

Proses *coding* dilakukan agar desain aplikasi yang dibuat dapat dimengerti oleh mesin. Bahasa pemrograman yang digunakan adalah bahasa pemrograman *java*.

e. *Testing or Verification*

Setelah proses *coding* selesai perlu diadakan proses *testing or verification*. Proses tersebut memiliki manfaat yaitu untuk menguji aplikasi yang dibuat apakah masih terdapat *error* dan hasilnya benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

f. *Maintenance*

Pemeliharaan sebuah *software* diperlukan, termasuk di dalamnya adalah pengembangan, karena *software* yang dibuat tidak selamanya dapat memenuhi tugas-tugas yang diperlukan.

3.3 Analisis Kebutuhan Perangkat Lunak

3.3.1 Kebutuhan Proses

Kebutuhan proses dalam aplikasi untuk UD.Sri Rahayu Meuble, yaitu :

1. Proses *Input* data *buyer / customer*.
2. Proses *Input* data barang.
3. Proses *Input* data order.
4. Proses membuat *file* ke dalam *Ms.Excel*

3.3.2 Kebutuhan *Input*

Input atau masukan dari aplikasi *desktop* ini, yaitu:

- 1) Data untuk *Buyer*, Barang dan Order
- 2) Data untuk pengeluaran laporan

3.3.3 Kebutuhan *Output*

Laporan yang berisi rincian data yang sudah di olah.

3.4 Rancangan Sistem

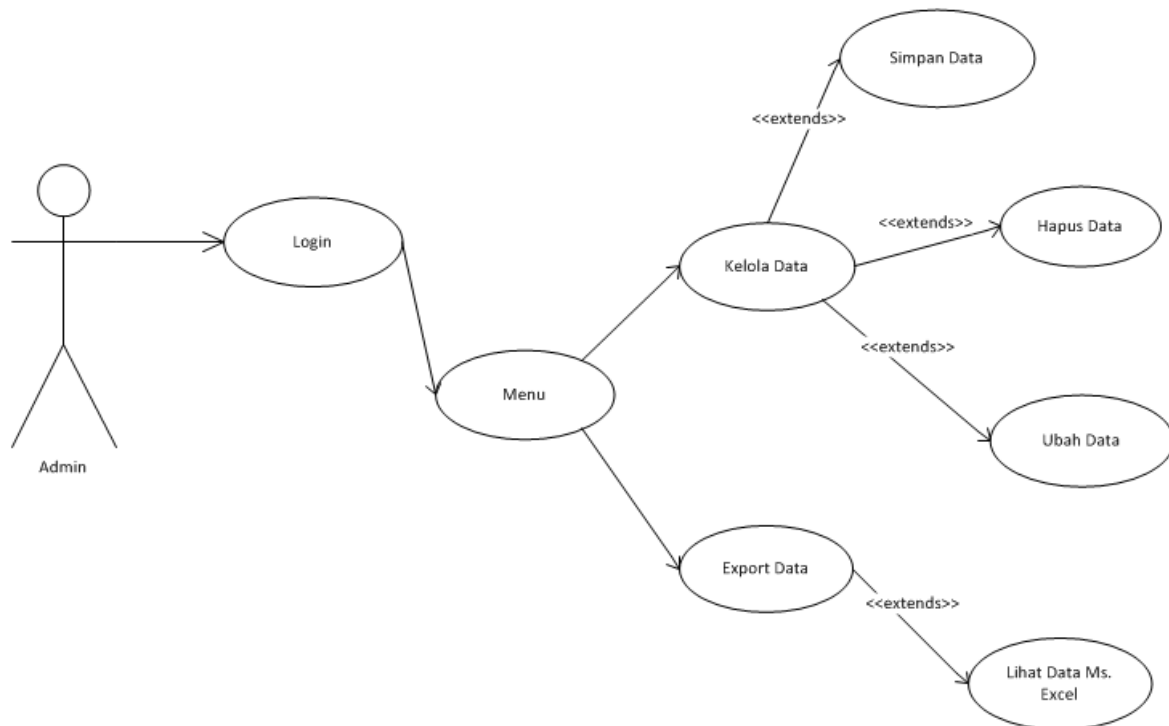
Metode perancangan yang di lakukan untuk mengembangkan perangkat lunak ini adalah dengan menggunakan diagram *Unified Modeling Language* (UML). *Unified Modeling Language* (UML) adalah bahasa spesifikasi standar yang di gunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan sebuah perangkat lunak. *Unified Modeling Language* (UML) merupakan metodologi untuk mengembangkan sistem berorientasi objek dan juga merupakan *tool* untuk mendukung pengembangan sistem.

Pada penelitian ini, diagram-diagram yang di gunakan untuk memodelkan aplikasi berorientasi objek dalam *Unified Modeling Language* (UML) meliputi *use case* diagram, *sequence* diagram, dan *activity* diagram.

3.4.1 Use Case Diagram

Use Case Diagram digunakan untuk memodelkan proses bisnis berdasarkan pandangan pengguna sistem. *Use case* diagram lebih menekankan pada “siapa” melakukan “apa” dalam lingkungan sistem perangkat lunak yang akan di bangun.

Use case diagram untuk aplikasi ini dapat dilihat pada Gambar 3.1.



Gambar 3.2 *Use Case Diagram*

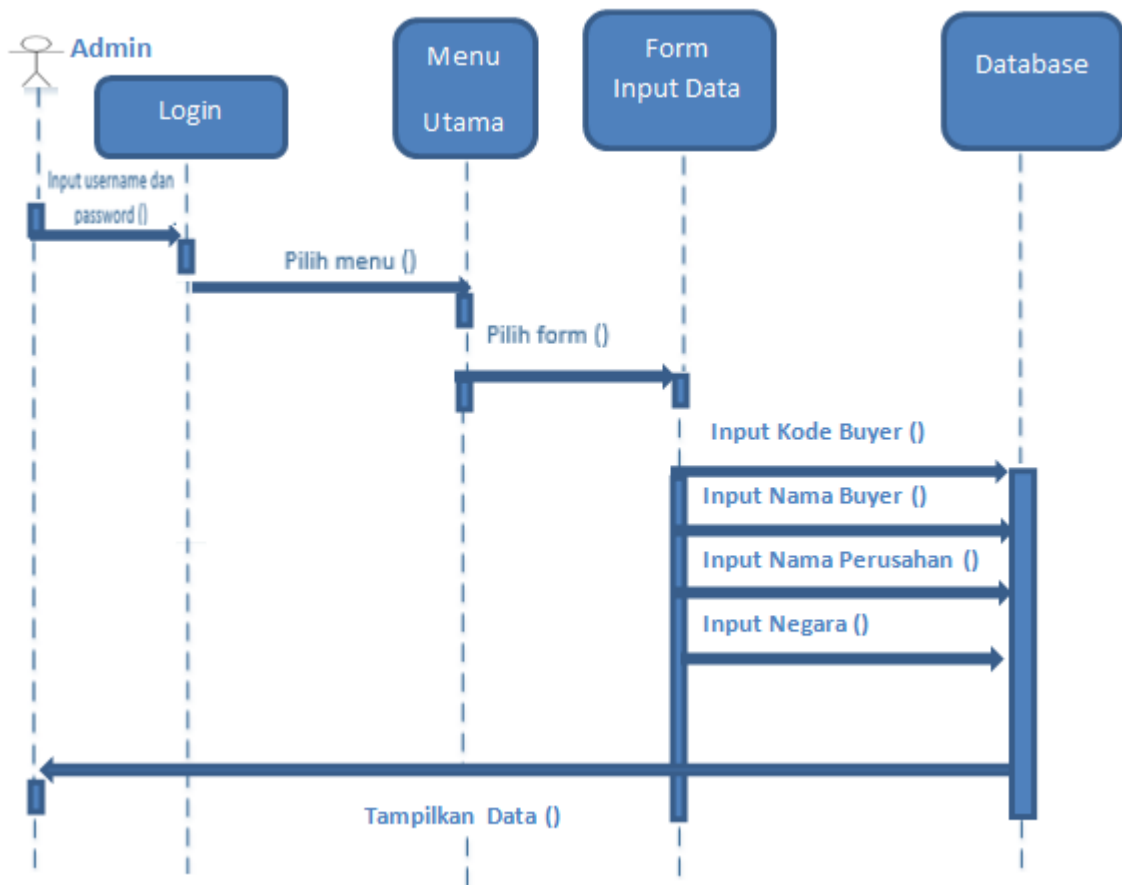
Penjelasan tentang *Use Case Diagram* pada gambar 3.2

1. Aktor pada *use case* diagram yaitu *Admin*.
2. Pada sistem yang dibuat *Admin* memiliki wewenang untuk mengelola seluruh kegiatan mulai dari *input*, *hapus*, *ubah data* dan *lihat data barang, buyer* dan *order*.

3.4.2 Sequence Diagram

Sequence Diagram di gunakan untuk memodelkan pengiriman pesan antar objek. Diagram ini menunjukkan serangkaian pesan yang di pertukarkan oleh objek-objek yang melakukan tugas atau aksi tertentu.

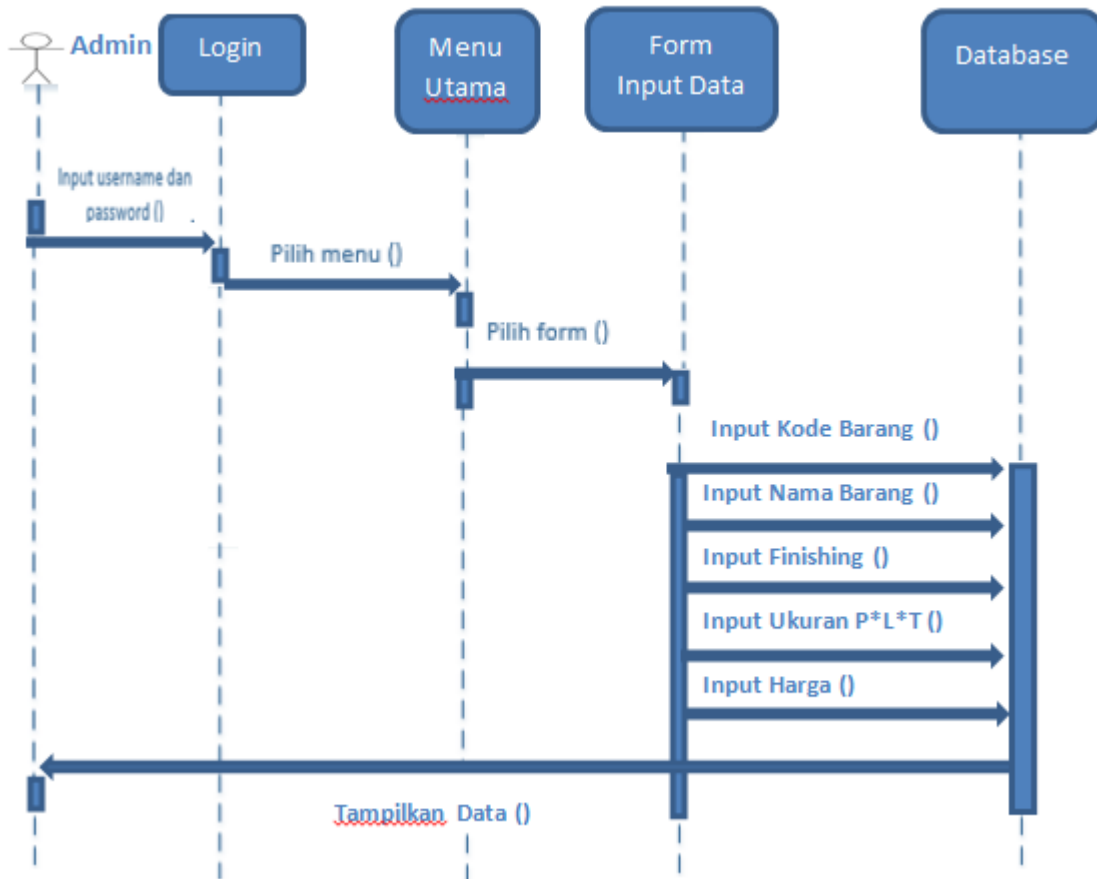
Sequence diagram yang di gunakan untuk menyimpan Data *Buyer* pada pengolahan data order dapat di lihat pada Gambar 3. 3



Gambar 3.3 *Sequence* Diagram Simpan Data Buyer

Skenario ini menggambarkan *input* untuk salah satu data order yaitu Data *Buyer*. Seorang admin memilih menu sesuai *form* data yang di perlukan yaitu *form* data. Admin dapat melakukan *input* data sesuai kolom yang tersedia dan menampilkannya pada tabel yang terdapat dalam *form* tersebut.

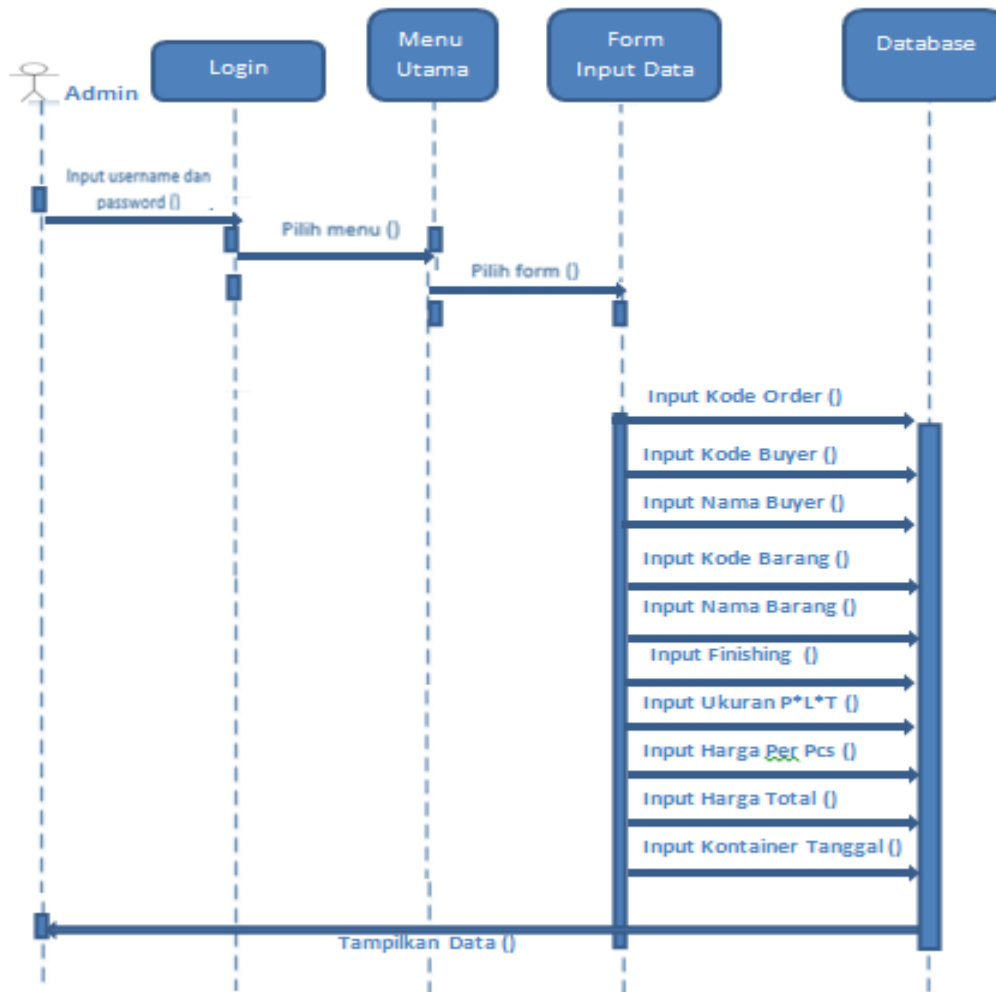
Sequence diagram yang di gunakan untuk menyimpan Data Barang pada pengolahan data order dapat di lihat pada Gambar 3.4



Gambar 3.4 Sequence Diagram Simpan Data Barang

Skenario ini menggambarkan *input* untuk salah satu data order yaitu Data Barang. Seorang admin memilih menu sesuai *form* data yang di perlukan yaitu *form* data. Admin dapat melakukan *input* data sesuai kolom yang tersedia dan menampilkannya pada tabel yang terdapat dalam *form* tersebut.

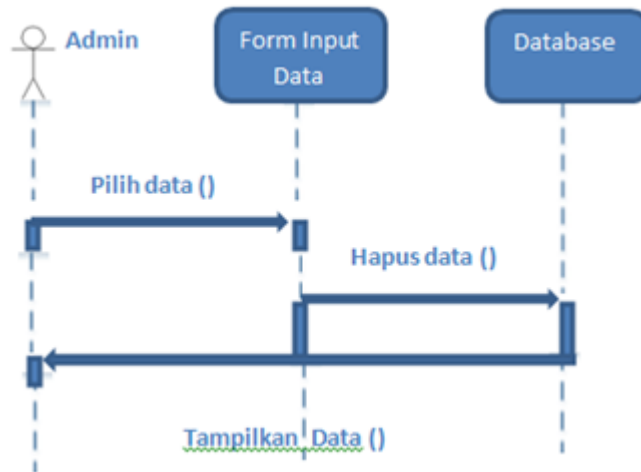
Sequence diagram yang di gunakan untuk menyimpan Data Order pada pengolahan data order dapat dilihat pada Gambar 3.5



Gambar 3.5 *Sequence Diagram* Simpan Data Order

Skenario ini menggambarkan *input* untuk Data Order. Seorang admin memilih menu sesuai *form* data yang di perlukan yaitu *form* data. Admin dapat melakukan *input* data sesuai kolom yang tersedia dan menampilkannya pada tabel yang terdapat dalam *form* tersebut.

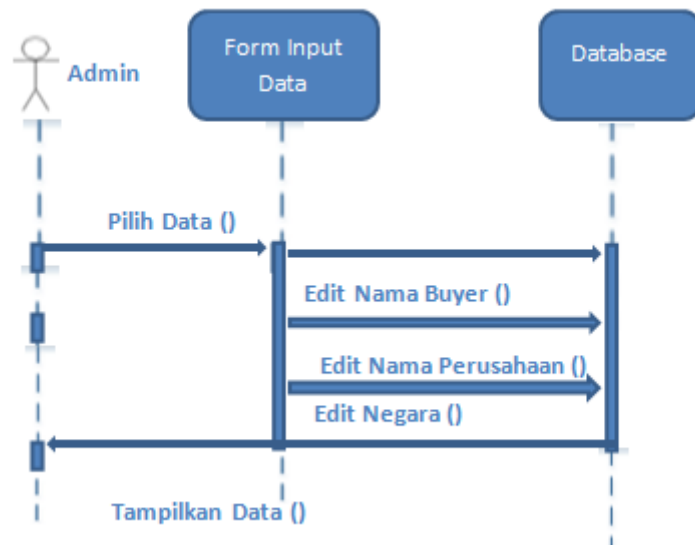
Sequence diagram yang di gunakan untuk menghapus satu baris dapat di lihat pada Gambar 3.6



Gambar 3.6 Sequence Diagram Hapus Data

Skenario ini menggambarkan menghapus data. Admin memilih satu baris data yang akan di hapus. Kemudian admin dapat menghapus satu baris data sesuai yang di perlukan dan menampilkan data yang ada pada tabel di *form* tersebut.

Sequence diagram yang di gunakan untuk mengedit data dapat di lihat pada Gambar 3.7



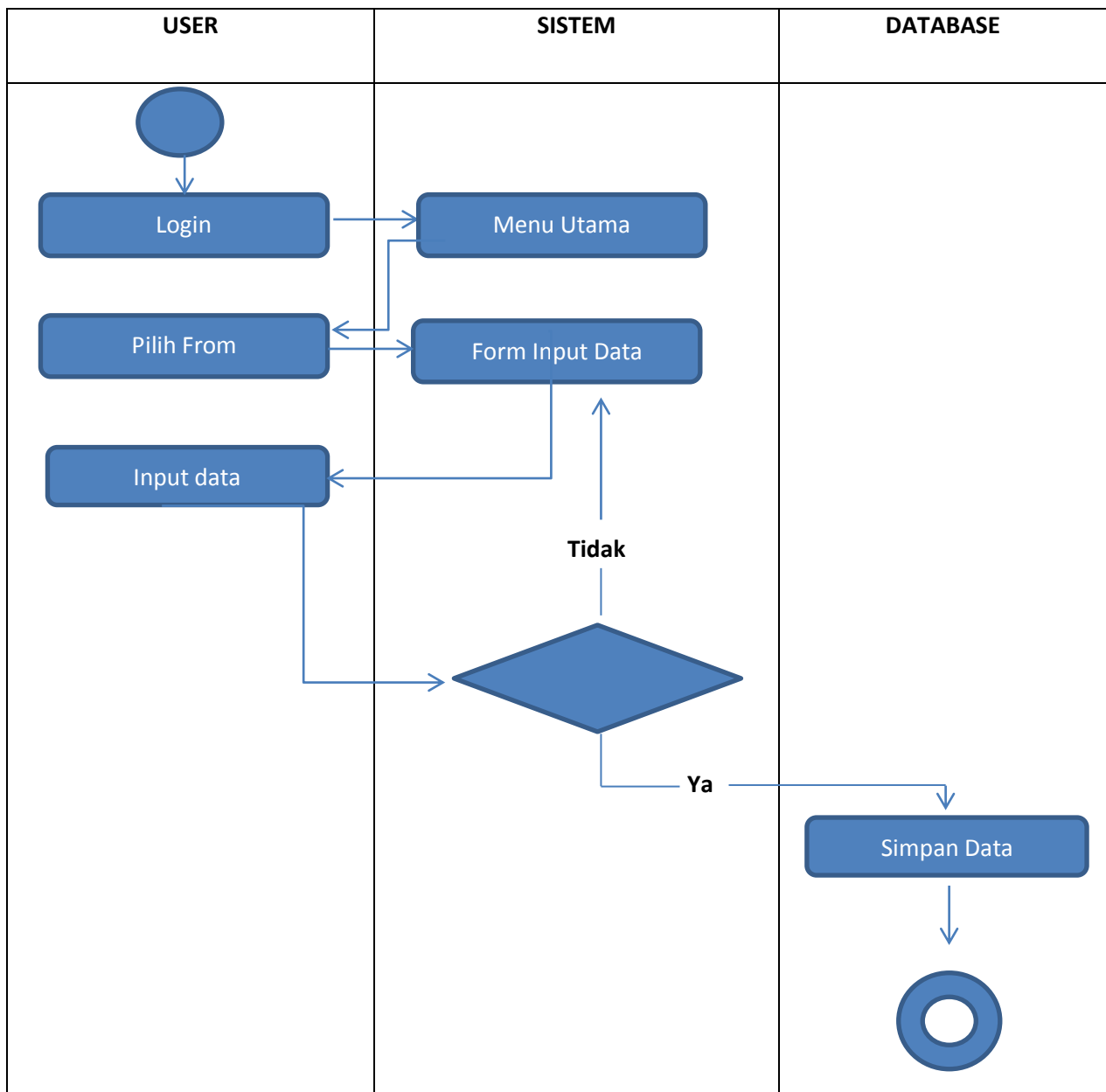
Gambar 3.7 Sequence Diagram Edit Data

Skenario ini menggambarkan mengedit kolom pada data. Admin memilih salah satu data yang akan di ubah. Kemudian admin dapat megubah data sesuai yang di perlukan. Data yang telah di ubah dapat di lihat pada tabel yang terdapat dalam *form* tersebut.

3.5.3 Activity Diagram

Activity Diagram di gunakan untuk memodelkan perilaku *use case* pada objek di dalam sistem. *Activity* diagram berfungsi untuk menggambarkan aliran kerja dari suatu proses dan urutan aktivitas dalam sebuah proses.

Activity diagram untuk melakukan *input* data yang masuk dapat di lihat pada Gambar 3.8



Gambar 3.8 Activity Diagram Data Masuk

Pada Gambar 3.8 User memilih *form input* data kemudian sistem menampilkan *form input* data, kemudian *user* memasukan data jika data yang dimasukan sesuai sistem akan memasukan data ke *database*, jika data yang dimasukan tidak sesuai maka akan kembali ke *form input* data.

3.5 Rancangan Antar Muka

Perancangan *interface* sistem diperlukan untuk memudahkan *user* dalam melakukan proses interaksi terhadap sistem. *Interface* menyediakan tampilan halaman sebuah sistem yang digunakan untuk proses *input* hingga menghasilkan *output* yang sesuai dengan kebutuhan. *Interface* untuk sistem yang akan dibuat adalah sebagai berikut:

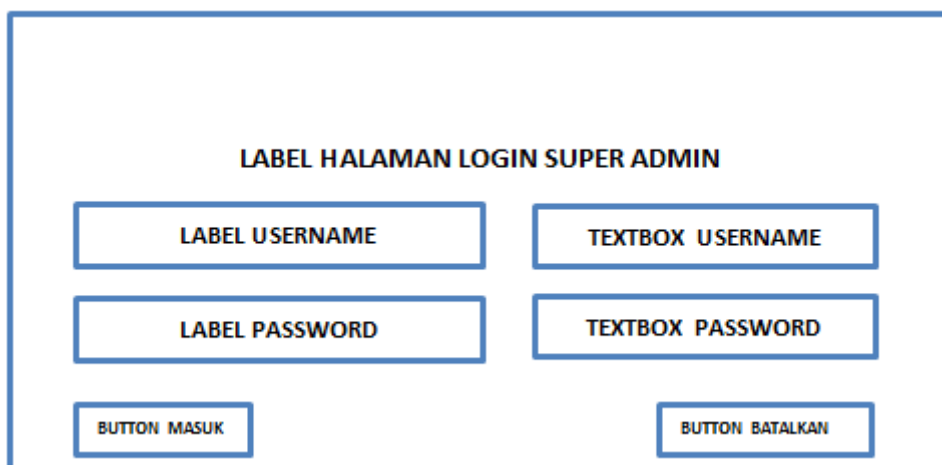
3.5.1 Desain Form Awal



Gambar 3.9 Desain Form Halaman Awal

Gambar 3.9 Merupakan desain halaman awal. Pada halaman ini pengguna dapat memilih untuk *login* sebagai *Super Admin*, atau *Admin*.

3.5.2 Desain Form Login Super Admin



Gambar 3.10 Desain Halaman Login Super Admin

Gambar 3.10 Merupakan desain halaman *login* untuk *Super Admin*. Pengguna harus memasukkan *username* dan *password* yang benar untuk dapat masuk ke halaman selanjutnya.

3.5.3 Desain *Form Login Admin*

The diagram shows a rectangular frame containing the layout for the Admin Login form. At the top center is the label 'LABEL HALAMAN LOGIN ADMIN'. Below this, there are two columns of elements. The left column contains 'LABEL USERNAME' and 'LABEL PASSWORD' stacked vertically, with 'BUTTON MASUK' at the bottom. The right column contains 'TEXTBOX USERNAME' and 'TEXTBOX PASSWORD' stacked vertically, with 'BUTTON BATALKAN' at the bottom.

Gambar 3.11 Desain *Form Login Admin*

Gambar 3.11 Merupakan desain halaman *login* untuk *Admin*. Pengguna harus memasukkan *username* dan *password* yang benar untuk dapat masuk ke halaman selanjutnya.

3.5.4 Desain *Form Super Admin*

The diagram shows a rectangular frame containing the layout for the Super Admin main page. At the top center is the label 'LABEL HALAMAN SUPER ADMIN'. Below this, there are three buttons arranged horizontally: 'BUTTON MENU UTAMA', 'BUTTON KELOLA ADMIN', and 'KELUAR'.

Gambar 3.12 Desain Halaman *Super Admin*

Gambar 3.12 Merupakan desain halaman utama untuk *Super Admin*. Pada halaman ini pengguna dapat memilih untuk masuk ke menu utama atau untuk mengelola admin dan keluar atau kembali ke halaman awal.

3.5.5 Desain Form Kelola Admin Super Admin

Gambar 3.13 Desain Halaman Kelola Admin

Gambar 3.13 Merupakan desain halaman *Super Admin* untuk mengelola *admin*. Pada halaman ini pengguna dapat menambah menghapus dan mengubah data *admin*.

Tabel 3.1 Fitur Form Kelola Admin

| NO | Fitur | Fungsi |
|----|---------------------------|--|
| 1 | <i>ComboBox Kode User</i> | Untuk <i>input</i> dan memilih kode <i>User</i> |
| 2 | <i>TextBox Nama</i> | Untuk <i>input</i> data nama <i>admin</i> |
| 3 | <i>TextBox Username</i> | Untuk <i>input</i> data <i>username admin</i> |
| 4 | <i>TextBox Password</i> | Untuk <i>input</i> data <i>password admin</i> |
| 5 | <i>TextBox Jabatan</i> | Untuk <i>input</i> data jabatan <i>admin</i> |
| 6 | <i>Button Tambah</i> | Untuk menambahkan data <i>admin</i> ke <i>database</i> |
| 7 | <i>Button Edit</i> | Untuk mengedit data <i>admin</i> pada <i>database</i> |
| 8 | <i>Button Hapus</i> | Untuk menghapus data <i>admin</i> pada <i>database</i> |
| 9 | <i>Button Clear</i> | Untuk menghapus <i>text</i> pada semua <i>textbox</i> |
| 10 | <i>Datagrid view</i> | Untuk menampilkan data <i>admin</i> dari <i>database</i> |

3.5.6 Desain Form Tab Page Input Data Buyer

Gambar 3.14 Desain Form Tab Page Input Data Buyer

Gambar 3.14 merupakan desain untuk *form tab page input data buyer*. *Form tab page input data buyer*, merupakan form utama ketika aplikasi dijalankan. *Form ini* digunakan untuk mengolah data *buyer*. Pengguna dapat memasukkan data pada setiap *combobox*, *textbox* yang telah ada selain itu pengguna juga dapat menghapus data, memindah data ke *Microsoft Excel*.

Pada *form tab page input data buyer* memiliki beberapa fitur Fungsi dari fitur-fitur tersebut di jelaskan pada Tabel 3.2.

Tabel 3.2 Fitur Form Tab Page Input Data Buyer

| NO | Fitur | Fungsi |
|----|--------------------------------|--|
| 1 | <i>ComboBox Kode Buyer</i> | Untuk <i>input</i> dan memilih kode <i>buyer</i> |
| 2 | <i>TextBox Nama Buyer</i> | Untuk <i>input</i> data nama <i>buyer</i> |
| 3 | <i>TextBox Nama Perusahaan</i> | Untuk <i>input</i> data nama perusahaan |
| 4 | <i>TextBox Negara</i> | Untuk <i>input</i> data negara <i>buyer</i> |
| 5 | <i>Button Tambah</i> | Untuk menambahkan data <i>buyer</i> ke <i>database</i> |
| 6 | <i>Button Edit</i> | Untuk mengedit data <i>buyer</i> pada <i>database</i> |
| 7 | <i>Button Hapus</i> | Untuk menghapus data <i>buyer</i> pada <i>database</i> |
| 8 | <i>Button Clear</i> | Untuk menghapus <i>text</i> pada semua <i>textbox</i> |

| | | |
|----|-------------------------------|--|
| 9 | <i>Button Export to Excel</i> | Untuk mengexport data dari <i>datagridview</i> menjadi Ms. Excel |
| 10 | <i>Datagrid view</i> | Untuk menampilkan data <i>buyer</i> dari <i>database</i> |

3.5.7 Desain Form Tab Page Input Data Barang

Gambar 3.15 Desain Form Tab Page Input Data Barang

Gambar 3.15 merupakan desain untuk tab page *input* data barang pada aplikasi. *Form* ini digunakan untuk mengolah data barang. Pengguna dapat memasukan data pada setiap *combobox*, *textbox* yang telah ada selain itu pengguna juga dapat menghapus data, memindah data ke *Microsoft Excel*.

Pada *form* Data Umum memiliki beberapa fitur Fungsi dari fitur-fitur tersebut di jelaskan pada Tabel 3.3.

Tabel 3.3 Fitur Form Tab Page Input Data Barang

| NO | Fitur | Fungsi |
|----|------------------------------|---|
| 1 | <i>ComboBox</i> Kode Barang | Untuk <i>input</i> dan memilih kode barang |
| 2 | <i>TextBox</i> Nama Barang | Untuk <i>input</i> data nama barang |
| 3 | <i>TextBox</i> Finishing | Untuk <i>input</i> data finishing barang |
| 4 | <i>TextBox</i> P (Panjang) | Untuk <i>input</i> data ukuran panjang barang |

| | | |
|----|-------------------------------|--|
| 5 | <i>TextBox</i> L (Lebar) | Untuk <i>input</i> data ukuran lebar barang |
| 6 | <i>TextBox</i> T (Tinggi) | Untuk <i>input</i> data ukuran tinggi barang |
| 7 | <i>TextBox</i> Harga | Untuk <i>input</i> data harga barang |
| 8 | <i>Button</i> Tambah | Untuk menambahkan data barang ke <i>database</i> |
| 9 | <i>Button</i> Edit | Untuk mengedit data barang pada <i>database</i> |
| 10 | <i>Button</i> Hapus | Untuk menghapus data barang pada <i>database</i> |
| 11 | <i>Button</i> Clear | Untuk menghapus <i>text</i> pada semua <i>textbox</i> |
| 12 | <i>Button</i> Export to Excel | Untuk mengexport data dari <i>datagridview</i> menjadi Ms. Excel |
| 13 | <i>Datagrid view</i> | Untuk menampilkan data barang dari <i>database</i> |

3.5.8 Desain Form Tab Page Input Data Order

Gambar 3.16 Desain Form Tab Page Input Data Order

Gambar 3.16 merupakan desain untuk mengolah data order. Pengguna dapat memasukkan data pada setiap *combobox*, *textbox* dan *datetimepicker* yang telah ada selain itu pengguna juga dapat menghapus data, memindah data ke *Microsoft Excel*.

Pada *form* Tab Page Input Data Order memiliki beberapa fitur Fungsi dari fitur-fitur tersebut di jelaskan pada Tabel 3.4.

Tabel 3.4 Fitur *Form Tab Page Input Data Order*

| NO | Fitur | Fungsi |
|-----------|---|--|
| 1 | <i>ComboBox</i> Kode Order | Untuk <i>input</i> dan memilih kode order |
| 2 | <i>ComboBox</i> Kode Buyer | Untuk <i>input</i> dan memilih kode <i>buyer</i> |
| 3 | <i>TextBox</i> Nama Buyer | Untuk <i>input</i> data nama <i>buyer</i> |
| 4 | <i>ComboBox</i> Kode Barang | Untuk <i>input</i> dan memilih kode barang |
| 5 | <i>TextBox</i> Nama Barang | Untuk <i>input</i> data nama barang |
| 6 | <i>TextBox</i> <i>Finishing</i> | Untuk <i>input</i> data <i>finishing</i> barang |
| 7 | <i>TextBox</i> P (Panjang) | Untuk <i>input</i> data ukuran panjang barang |
| 8 | <i>TextBox</i> L (Lebar) | Untuk <i>input</i> data ukuran lebar barang |
| 9 | <i>TextBox</i> T (Tinggi) | Untuk <i>input</i> data ukuran tinggi barang |
| 10 | <i>TextBox</i> Harga | Untuk <i>input</i> data harga barang |
| 11 | <i>TextBox</i> Harga Total | Untuk <i>input</i> data harga total barang yang di order |
| 12 | <i>DateTimePicker</i> Kontainer tanggal | Untuk <i>input</i> data kontainer tanggal order |
| 13 | <i>Button</i> Tambah | Untuk menambahkan data order ke <i>database</i> |
| 14 | <i>Button</i> Edit | Untuk mengedit data order pada <i>database</i> |
| 15 | <i>Button</i> Hapus | Untuk menghapus data order pada <i>database</i> |
| 16 | <i>Button</i> Clear | Untuk menghapus <i>text</i> pada semua <i>textbox</i> |
| 17 | <i>Button</i> Export to Excel | Untuk mengexport data dari <i>datagridview</i> menjadi Ms. Excel |
| 18 | <i>Datagrid view</i> | Untuk menampilkan data order dari <i>database</i> |

3.5.9 Desain *Form Tab Page* Lihat Order Details

Gambar 3.17 Desain *Form Tab Page* Lihat Order Details

Gambar 3.17 merupakan tampilan dari *form* tab page lihat order details. *Form* ini digunakan untuk melihat order berdasarkan nama *buyer* atau kontainer tanggal yang akan di tampilkan pada *datagridview* pengguna juga dapat memindah data tersebut ke *Microsoft Excel*.

Pada *form* Tab Page Lihat Order Details memiliki beberapa fitur fungsi dari fitur-fitur tersebut di jelaskan pada Tabel 3.5.

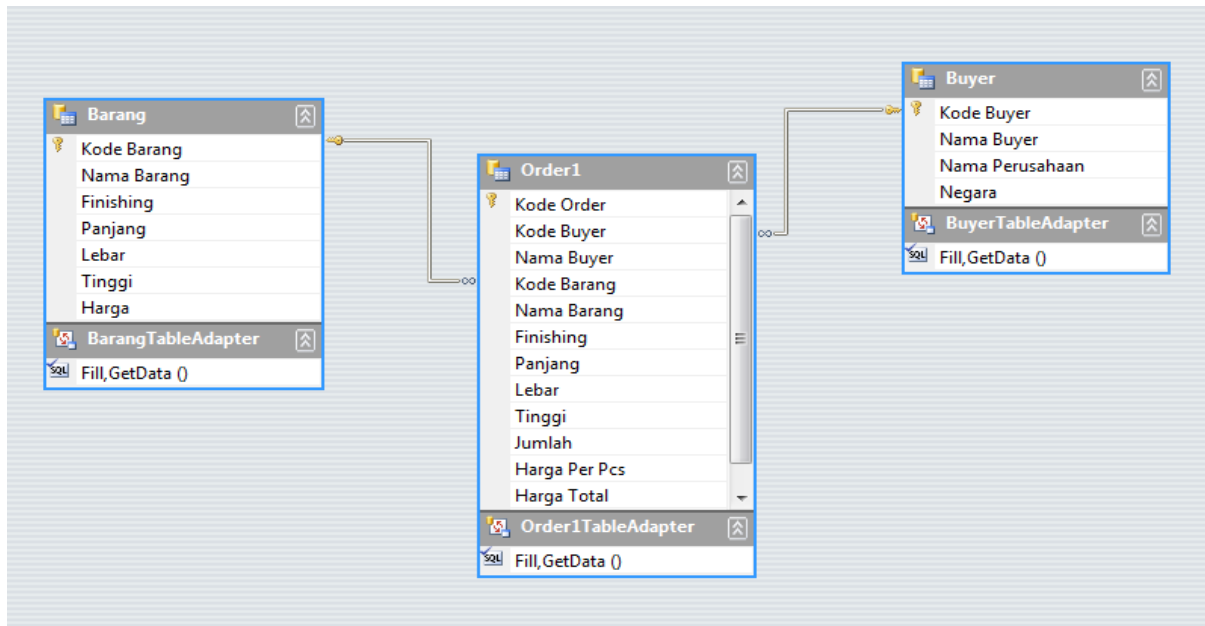
Tabel 3.5 Fitur *Form Tab Page* Lihat Order Details

| NO | Fitur | Fungsi |
|----|---|--|
| 1 | <i>TextBox</i> Filtering Nama Buyer/Kontainer Tanggal | Untuk memasukan <i>text filtering</i> data berdasarkan nama buyer atau kontainer tanggal |
| 2 | <i>DataGrid View</i> Database | Untuk menampilkan data berdasarkan <i>filtering</i> data |

3.6 Rancangan Database

3.6.1 Class Diagram

Database yang di gunakan pada aplikasi yaitu database UD.SRI RAHAYU. Adapun tabel yang digunakan yaitu dbo.Barang, dbo.Buyer, dbo.Order1. Gambar 3.18 berikut ini class diagram dari database yang terdapat dalam sistem di UD. Sri Rahayu Meuble



Gambar 3.18 Class Diagram

Tabel 3.7 Kamus Data Tabel Barang

| Nama Kolom | Tipe Data | Panjang Kolom | Keterangan |
|-------------|-----------|---------------|--|
| Kode Barang | Varchar | 50 | Kolom ini merekam tentang data kode suatu barang. Kolom ini sebagai <i>primary key</i> pada tabel ini. |
| Nama Barang | Varchar | 50 | Kolom ini merekam tentang data nama suatu barang. |
| Finishing | Varchar | 50 | Kolom ini merekam tentang jenis <i>finishing</i> suatu barang. |
| Panjang | Integer | - | Kolom ini merekam tentang ukuran panjang suatu barang. |
| Lebar | Integer | - | Kolom ini merekam tentang ukuran lebar suatu barang. |

| | | | |
|--------|----------------|---|---|
| Tinggi | <i>Integer</i> | - | Kolom ini merekam tentang ukuran tinggi suatu barang. |
| Harga | <i>Money</i> | - | Kolom ini merekam tentang harga suatu barang. |

Tabel 3.8 Kamus Data Tabel *Buyer*

| Nama Kolom | Tipe Data | Panjang Kolom | Keterangan |
|-------------------|------------------|----------------------|---|
| Kode Buyer | <i>Varchar</i> | 50 | Kolom ini merekam tentang data kode <i>buyer</i> . Kolom ini sebagai <i>primary key</i> pada tabel ini |
| Nama Buyer | <i>Varchar</i> | 50 | Kolom ini merekam nama <i>buyer</i> . |
| Nama Perusahaan | <i>Varchar</i> | 50 | Kolom ini merekam nama perusahaan dari suatu <i>buyer</i> . |
| Negara | <i>Varchar</i> | 50 | Kolom ini merekam tentang nama negara suatu <i>buyer</i> . |

Tabel 3.9 Kamus Data Tabel Order1

| Nama Kolom | Tipe Data | Panjang Kolom | Keterangan |
|-------------------|------------------|----------------------|--|
| Kode Oder | <i>Varchar</i> | 50 | Kolom ini merekam kode order dari suatu order. Kolom ini sebagai <i>primary key</i> pada tabel ini |
| Kode Buyer | <i>Varchar</i> | 50 | Kolom ini merekam tentang kode <i>buyer</i> dari data <i>buyer</i> pemesan. |
| Nama Buyer | <i>Varchar</i> | 50 | Kolom ini merekam tentang nama <i>buyer</i> dari data <i>buyer</i> pemesan. |
| Kode Barang | <i>Varchar</i> | 50 | Kolom ini merekam tentang kode barang yang di order. |
| Nama Barang | <i>Varchar</i> | 50 | Kolom ini merekam tentang nama barang yang di order. |
| Finishing | <i>Varchar</i> | 50 | Kolom ini merekam tentang jenis <i>finishing</i> dari suatu barang yang di order |
| Panjang | <i>Integer</i> | - | Kolom ini merekam tentang ukuran panjang dari suatu barang yang di order. |
| Lebar | <i>Integer</i> | - | Kolom ini merekam tentang ukuran lebar dari suatu barang yang di order. |

| | | | |
|-------------------|----------------|----|--|
| Tinggi | <i>Integer</i> | - | Kolom ini merekam tentang ukuran tinggi dari suatu barang yang di order. |
| Harga Per Pcs | <i>Money</i> | - | Kolom ini merekam tentang harga dari suatu barang yang di order. |
| Harga Total | <i>Money</i> | - | Kolom ini merekam tentang harga total dari suatu barang yang di order. |
| Kontainer Tanggal | <i>Varchar</i> | 50 | Kolom ini merekam tentang jadwal tanggal keberangkatan kontainer dari suatu order. |

3.7 Pengembangan Sistem

Dalam sistem aplikasi yang dibuat *user* atau pengguna dalam hal ini kita sebut *admin*. Disini *admin* adalah pegawai kantor di UD. Sri Rahayu Meuble. *Admin* berperan sebagai orang yang mengelola data baik itu data order, data *buyer* dan data barang.

Berikut ini merupakan prosedur yang dapat dilakukan untuk mengembangkan aplikasi yang dibuat:

1. Menyiapkan perangkat keras dan perangkat lunak pendukung yang dibutuhkan dan menyiapkan kebutuhan sistem dalam aplikasi.
2. Merancang *User Interface* agar pengguna dapat menjalankan aplikasi dengan mudah.
3. Membuat basis data sesuai kebutuhan aplikasi.
4. Membuat sistem yang akan digunakan melalui proses *coding*.
5. Membuat sistem yang dapat memasukan data order, data *buyer* dan data barang ke database dan menampilkan dalam format xls.

3.9 Metode Pengujian

Metode pengujian adalah cara atau teknik untuk menguji perangkat lunak, mempunyai mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak secara lengkap dan mempunyai kemungkinan tinggi untuk menemukan kesalahan.

Metode yang digunakan dalam pengujian sistem aplikasi adalah *black box testing*. *Black box testing* atau tes fungsional adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak yang dikembangkan.

Pengujian dan pengembangan aplikasi dilakukan bersama-sama oleh pengembang dan pengguna yang terlibat sebagai pemberi input seperti data buyer, data order dan data barang. Dan melihat hasil yang didapat dari data yang dimasukkan dalam hal ini merupakan menampilkan data dalam format xls.

Beberapa hal yang diujikan terhadap aplikasi dengan tujuan mengetahui apakah aplikasi ini memenuhi tuntutan dari tujuan awal pembuatan aplikasi diantara lain:

1. Sistem dapat menambah, menghapus dan merubah data *buyer*, data barang dan data order.
2. Sistem dapat mencetak menampilkan data dalam format xls.
3. *User Interface* yang dibuat pada sistem tidak menyulitkan pengguna.