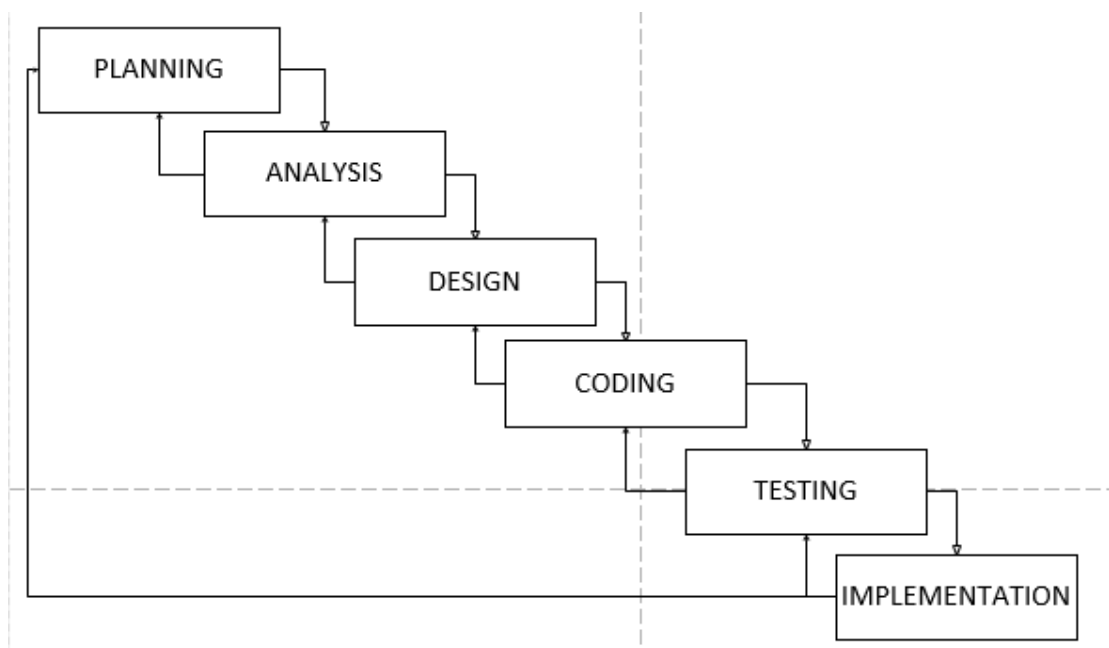


## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Langkah Penelitian

Penelitian menggunakan metode *Software Development Life Cycle (SDLC) model waterfall* karena melihat fungsi dari *model waterfall* yaitu ketika sedang membangun sistem informasi *website* E-commerce UKM tersebut pengerjaan dari setiap fase dalam metode *waterfall* harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase selanjutnya. Fokus dalam melakukan setiap fase-fase dapat dilakukan secara maksimal, karena adanya pengerjaan yang bersifat paralel. Jika dalam perjalanan pengembangan sistem terjadi kesalahan, kerusakan, ataupun *error* akan kembali dilakukan dari tahap yang menyebabkan kesalahan itu terjadi. *Model waterfall* akan digambarkan seperti pada **Gambar 3.1**.



**Gambar 3. 1** Model Waterfall

##### 3.1.1 *Planning* (Perencanaan)

Tahap perencanaan menyangkut studi tentang kebutuhan pengguna dan kelayakan sistem baik secara teknik maupun secara teknologi. Pada tahap ini

dilakukan perencanaan tentang sistem yang akan dibangun. Dalam hal ini adalah *website* E-Commerce UKM CV. Warisan Multi Tenun.

### **3.1.2 Analysis (Analisis)**

Tahap analisis merupakan proses pendalaman mengenai segala permasalahan dan resiko pada pengguna, proses pencarian kebutuhan diintensifkan dan difokuskan pada *software*. Pada tahap analisis sendiri peneliti melakukan proses observasi dan wawancara.

### **3.1.3 Design (Desain)**

Proses *design* digunakan untuk mengubah kebutuhan-kebutuhan di atas menjadi representasi ke dalam bentuk blueprint *software* sebelum coding dimulai. Tahap design menyangkut perancangan sistem dimana kita akan memberikan solusi dari masalah yang muncul pada tahap analisis. Perancangan menggunakan *UML, Use Case Diagram Activity Diagram, ER Diagram, Class Diagram* dan perancangan antarmuka.

### **3.1.4 Coding**

Tahap *coding* adalah tahap mengubah bentuk desain menjadi bentuk yang dimengerti oleh mesin, yaitu membuat sistem yang telah diimplementasikan pada tahap desain. Sehingga pada tahap ini kita bisa memulai pengerjaan pada *website* E-Commerce UKM.

### **3.1.5 Testing (Pengujian)**

Sesuatu yang dibuat haruslah diuji. Demikian juga *software*, semua fungsi-fungsi *software* harus diuji secara teliti agar *software* bebas dari error, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan pada tahap sebelumnya. Tahap pengujian juga merupakan tahap yang menentukan apakah desain yang sudah dibuat sesuai kebutuhan pengguna atau belum. Tujuan dari pengujian ini adalah untuk meminimalisir cacat desain *website* sehingga sistem yang dikembangkan benar-benar dapat berjalan dengan sebaik mungkin. Pengujian ini akan dilakukan dengan metode *black box*.

### **3.1.6 Implementasi**

Tahap implementasi adalah perencanaan sistem diimplementasikan ke situasi nyata dengan pemilihan perangkat keras dan penyusunan desain (*coding*). Untuk implementasi sistem yaitu dengan memasukkan desain yang sudah dibuat ke dalam perangkat yang disediakan.

## **3.2 Analisis Sistem Dan Perencanaan**

### **3.2.1 Observasi**

Metode observasi ini dilakukan untuk mengamati secara langsung objek penelitian agar peneliti dapat mengumpulkan data dan menyimpulkan data secara langsung di lapangan. Seperti mengamati proses transaksi yang terdapat pada UKM CV. Warisan Multi Tenun.

Pada tahap observasi terdapat metode transaksi yang dilakukan oleh UKM CV. Warisan Multi Tenun kurang efisien, dikarenakan pada metode transaksi sendiri pada CV. Warisan Multi Tenun masih mengharuskan *customer* datang ke UKM CV. Warisan Multi Tenun atau dengan cara mendatangi pengrajin. Hal tersebut membuat *customer* pada daerah luar Klaten maupun luar daerah Jawa akan terbebani dengan metode ini.

### **3.2.2 Hasil Wawancara**

Melakukan tanya jawab dengan pihak yang terkait secara langsung guna mendapatkan data-data yang dibutuhkan. Metode ini dinamakan wawancara. Metode wawancara lebih efisien karena wawancara dilakukan oleh peneliti dan narasumber dari CV. Warisan Multi Tenun untuk mendapatkan data-data yang dibutuhkan untuk sistem informasi untuk CV. Warisan Multi Tenun.

Setelah melakukan wawancara dengan pihak CV. Warisan Multi Tenun ternyata hasil wawancara tersebut mendapatkan permasalahan dalam pelaporan. Pada metode pelaporan terdapat kekurangan dari CV. Warisan Multi Tenun yaitu pelaporan tentang penjualan produk yang ada. Pelaporan tersebut dibuat dengan cara pembukuan manual maka metode tersebut kurang efisien dalam pengumpulan

laporan penjualan. Sehingga hal ini menyulitkan bagi pengurus(admin) yang bertugas untuk mengolah data pada UKM CV. Warisan Multi Tenun.

### **3.2.3 Analisis Sistem Berjalan**

Berdasarkan observasi dan wawancara, sistem yang saat ini terdapat pada UKM CV Warisan Multi Tenun belum memenuhi kebutuhan sistem. Yaitu terdapat pelaporan penjualan dengan cara manual yaitu pembukuan. Pada transaksi *customer* masih dengan cara manual yaitu *customer* diharuskan datang ke tempat UKM ataupun pada pengrajin. Dari sistem yang ada pada UKM CV. Warisan Multi Tenun membutuhkan sistem yang lebih baik dalam pengelolaannya.

### **3.2.4 Analisis Sistem Kebutuhan**

Berdasarkan analisis kebutuhan sistem, bahwa metode – metode yang ada dalam CV. Warisan Multi Tenun belum memenuhi kebutuhan sistem yang ada dalam pelaksanaannya. Dengan analisis permasalahan yang ada, peneliti memberikan solusi untuk mengatasi metode-metode yang kurang efisien menjadi lebih baik yang nantinya terdapat pada *website E-commerce* UKM CV. Warisan Multi Tenun. Untuk solusi tersebut nantinya terdapat pada bagian Admin dan User.

Pada bagian admin terdapat fitur-fitur seperti:

1. Pelaporan Hasil Penjualan
2. Pelaporan Pengiriman Produk
3. Pelaporan Transaksi
4. Pelaporan Konfirmasi Pembayaran

Fitur pendukung pada Admin:

1. Login
2. Insert, Update, Delete Barang (Produk)
3. Insert dan Update Stok
4. Delete pada Data *Customer (User)*
5. Update status pada data Pembelian

6. Insert, Update, Delete pada data Shipping
7. Insert, Update, Delete pada data Kategori
8. Insert, Update, Delete pada data Katalog
9. Mengelola data Admin
10. Insert, Update, Delete pada data bank

Serta pada bagian User (*customer*) akan terdapat fitur utama:

1. Transaksi secara langsung dengan *website*.
2. Proses Pembayaran mengharuskan *customer* mengunggah bukti transfer.
3. Konfirmasi jika barang sudah diterima.

Fitur pendukung pada User:

1. Register
2. Login
3. Produk Katalog
4. Checkout

Peralatan pendukung pada sistem informasi terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*). Perangkat yang digunakan dalam penelitian adalah sebagai berikut:

Adapun perangkat keras yang dibutuhkan dalam membangun *website* E-Commerce UKM CV. Warisan Multi Tenun:

- a. Processor Intel Core i3-2348M CPU @ 2.30GHz
- b. Memory 8 GB RAM
- c. Hardisk 455 GB
- d. Mouse
- e. Keyboard

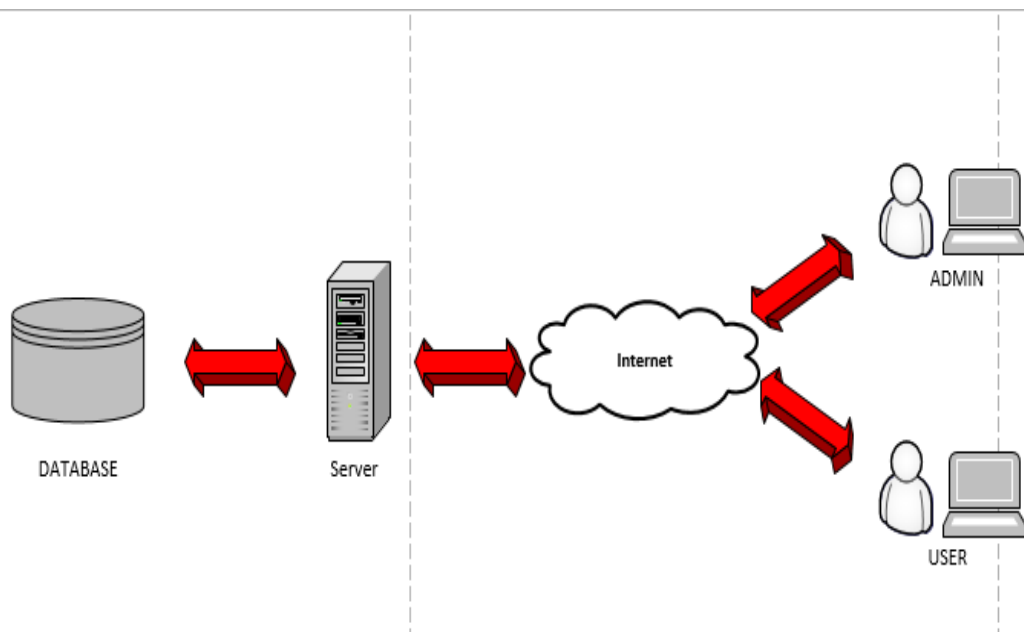
Adapun perangkat lunak yang dibutuhkan dalam membangun *website* E-Commerce UKM CV. Warisan Multi Tenun:

- a. Netbeans 8.1
- b. XAMPP (Apache, phpMyAdmin)
- c. Windows 10
- d. Browser: Internet Explorer, Google Chrome

### 3.3 Perancangan Sistem

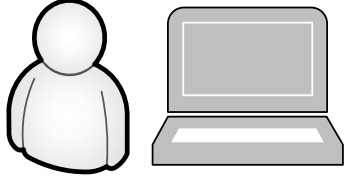
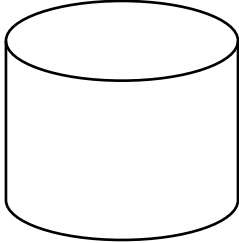
Pada tahap ini dilakukan penentuan perancangan sistem yang akan digunakan. Berikut ini adalah perancangan sistem yang diusulkan:


*Database server* yang digunakan pada aplikasi *MySQL* dan menggunakan XAMPP sebagai *web server*. Komunikasi antara pengguna *user*. Saat *user* mengakses aplikasi maka *web server* memuat *interface* dan mengambil data yang diperlukan pada *database server*. Melalui *interface* yang dimuat di *web server* pengguna bisa menyimpan data ke dalam *database server*.



**Gambar 3. 2** Arsitektur Sistem yang Diusulkan

**Tabel 3.1** Keterangan Komponen Arsitektur

 <p>User</p>	<p><i>Customer</i></p> <p>Admin</p>
 <p>Database</p>	<p>Berisi data-data sebagai berikut:</p> <ol style="list-style-type: none"><li>1. Produk</li><li>2. Kategori</li><li>3. Katalog</li><li>4. Stok</li><li>5. Shipping</li><li>6. Keranjang</li><li>7. Selesai</li><li>8. <i>Customer</i></li><li>9. Superuser</li><li>10. Pembelian</li><li>11. Tujuan</li><li>12. Konfirmasi</li><li>13. Bank</li></ol>

 <p data-bbox="572 658 684 692">Server</p>	<p data-bbox="868 309 1230 450">Server digunakan sebagai tempat penyimpanan data-data.</p>
---	--

### 3.4 Perancangan Proses

Dalam pembuatan aplikasi *website* UKM CV. Warisan Multi Tenun dilakukan perancangan *database* menggunakan *Diagram ER*. Metode perancangan lain yang digunakan dalam pembuatan aplikasi ini adalah menggunakan *Unified Markup Language* (UML) yang digunakan dalam pengembangan aplikasi yaitu model *Use Case Diagram*, *Activity Diagram*, dan *Class Diagram*.



### 3.4.1 Use Case Diagram



**Gambar 3. 3** Use Case Diagram Aplikasi

Berikut penjelasan Gambar 3.3:

1. Terdapat 2 aktor pada *use case diagram* aplikasi yaitu *customer* dan *admin*.
2. Admin mengelola semua yang berkaitan dengan data-data yang akan di tampilkan pada website serta mengelola penambahan atau pengurangan jumlah member.
3. Admin mengelola status pembelian dari *customer* dengan mengubah status transaksi
4. Admin mengelola pengiriman barang setelah *customer* melakukan konfirmasi pembayara melalui *website*
5. Admin mengelola harga pengiriman barang setiap kota.

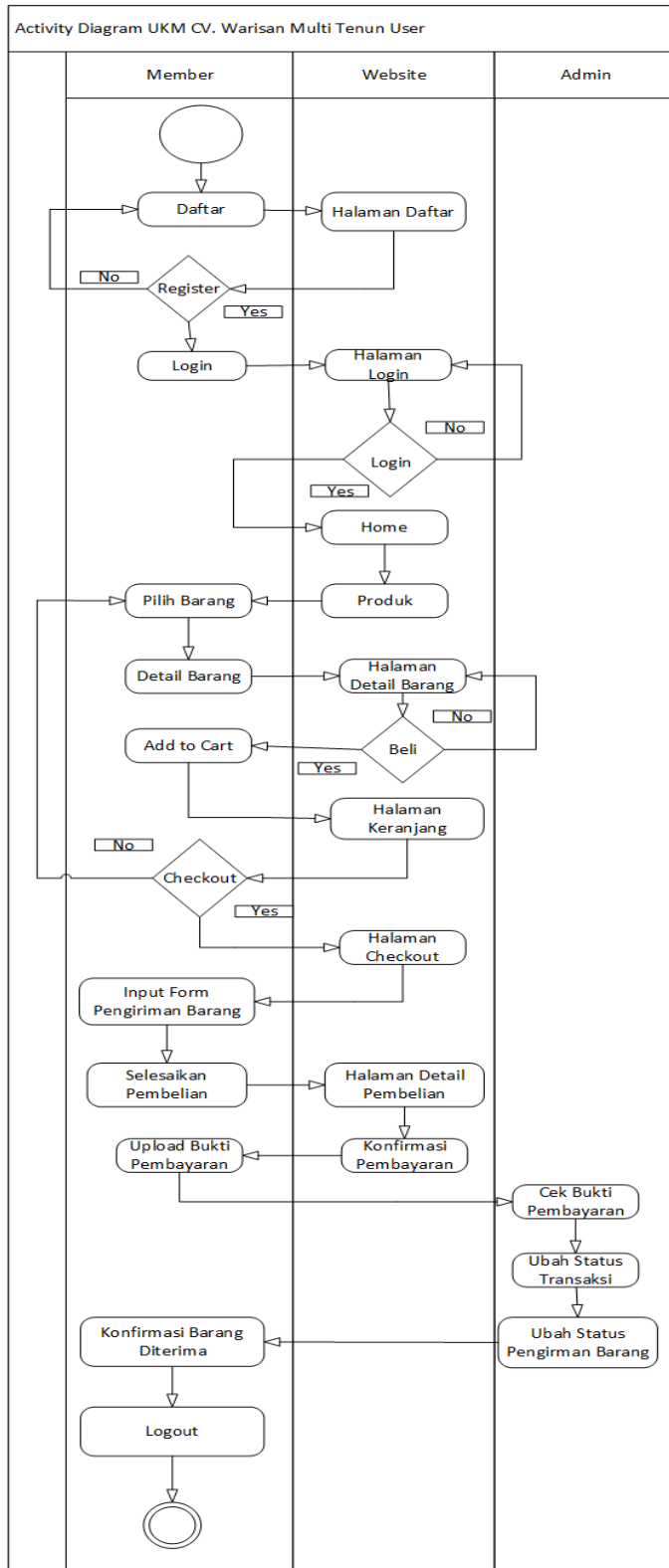
6. *Customer* dapat melakukan konfirmasi pembayaran dengan mengupload data bukti transfer pada halaman terakhir pada proses transaksi
7. *Customer* diharuskan mengonfirmasi untuk kedua kalinya yaitu mengonfirmasi jika barang sudah diterima.

### **3.4.2 Activity Diagram**

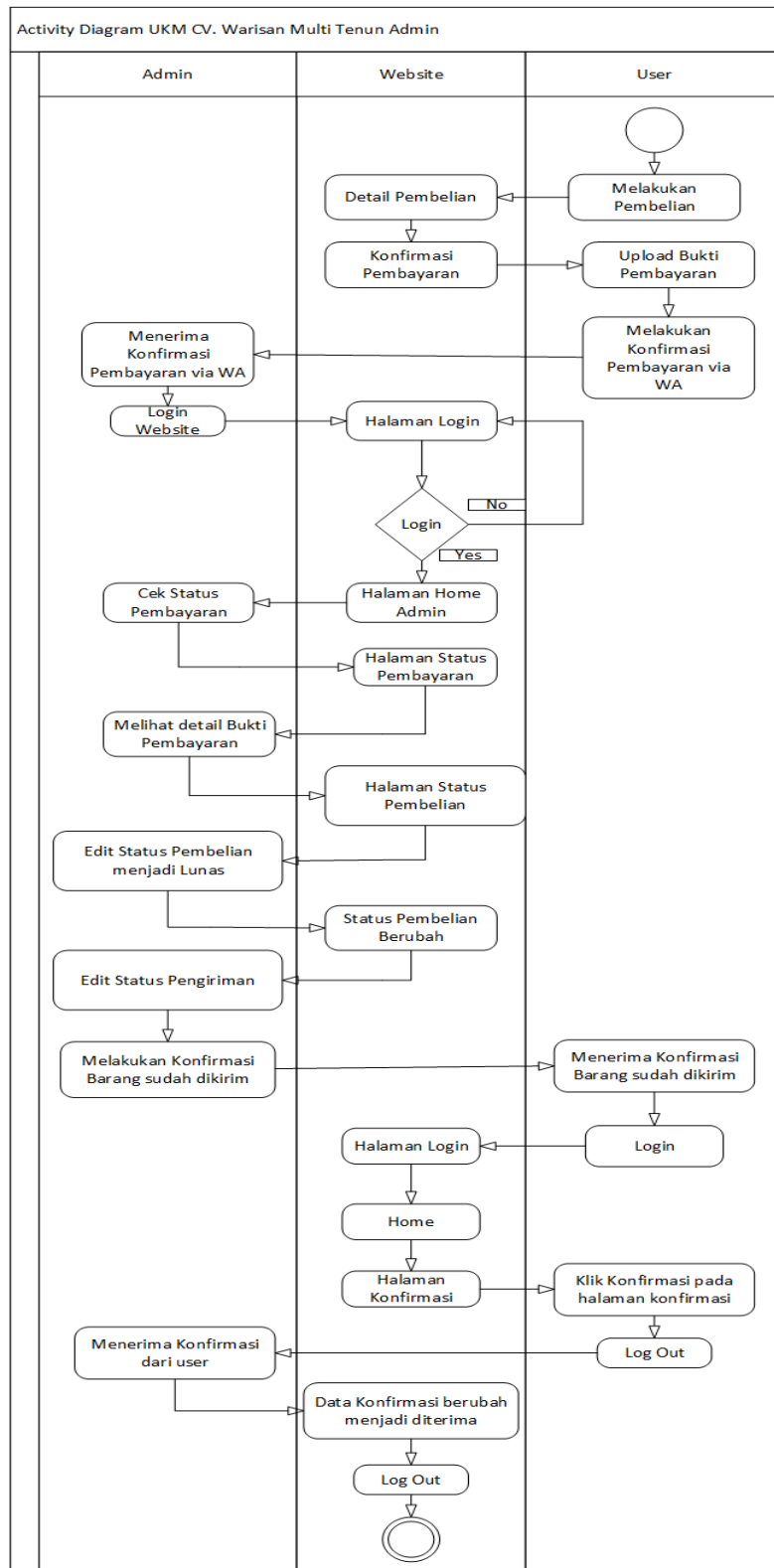
Gambar *activity* diagram yang digunakan dalam aplikasi dapat dilihat pada Gambar 3.4, Gambar 3.5.

#### **Penjelasan Gambar 3.4.**

Alur kegiatan dari *user* dimulai dengan *login* kedalam *website E-commerce* UKM CV. Warisan Multi Tenun namun sebelumnya jika *user* belum memiliki akun dari *website* maka *user* bisa membuat akun terlebih dahulu setelah itu *user* dapat login. Setelah itu *user* dapat melakukan pembelian produk dengan memilih produk dengan cara melihat detail produk dan menambahkannya pad keranjang sampai melakukan proses penginputan form pada halaman *checkout* sesuai dengan alamat pengiriman hingga melakukan proses konfirmasi pembayaran dengan mengunggah bukti transfer pembayaran yang telah dilakukan oleh *user*, pada setelah itu jika barang sudah sampai maka *user* harus melakukan konfirmasi lagi agar admin dapat melihat laporan jika barang sudah diterima diterima *user*.



Gambar 3.4 Activity Diagram User



Gambar 3. 5 Activity Diagram Admin

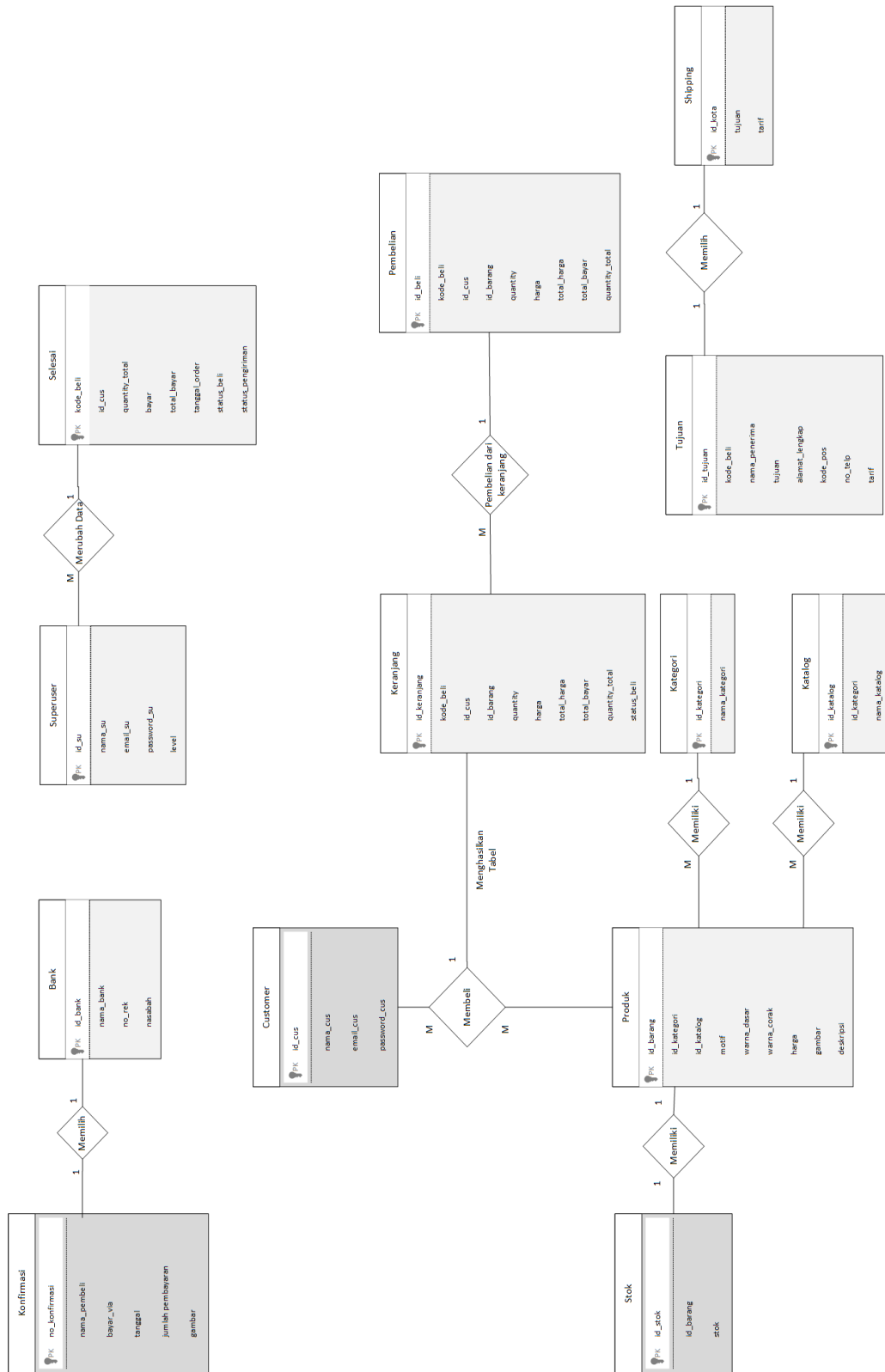
Pada Gambar 3.5 menunjukkan *Activity Diagram* pada kegiatan tersebut admin melakukan proses pengolahan data pada *website E-commerce* UKM CV. Warisan Multi Tenun. Admin diberikan hak akses penuh dalam pengelolaan data seperti:

1. Data Produk
2. Data Shipping
3. Data Konfirmasi Pembayaran
4. Data Pembelian
5. Mengelola data status transaksi
6. Mengelola data pengiriman barang

Pada gambar 3.6 terdapat 13 data entitas:

1. Produk
2. Kategori
3. Katalog
4. Stok
5. *Shipping*
6. Keranjang
7. Selesai
8. *Customer*
9. Superuser
10. Pembelian
11. Tujuan
12. Konfirmasi
13. Bank

### 3.4.3 ER Diagram



Gambar 3. 6 ER Diagram

Berikut penjelasan relasi antar entitas pada gambar 3.6:

Pada penjelasan relasi entitas konfirmasi tidak memiliki relasi dengan tabel manapun.

1. Pada entitas stok memiliki relasi one-to-many dengan entitas superuser, dikarenakan superuser hanya mengolah data stok pada sebuah produk.
2. Pada entitas superuser memiliki relasi one-to-many pada entitas stok dan entitas selesai artinya entitas superuser yang akan mengisi data pada entitas stok dan mengupdate entitas selesai.
3. Relasi dengan entitas produk, entitas superuser memiliki relasi many-to-many dikarenakan banyak data yang diolah oleh superuser.
4. Begitu juga dengan entitas *customer*, entitas superuser memiliki relasi many-to-many dengan entitas *customer*. Dikarenakan entitas superuser nantinya mengolah data dari entitas *customer*.
5. Entitas *Customer* sendiri memiliki relasi many-to-many dengan entitas keranjang, entitas tujuan, serta dengan entitas produk dikarenakan untuk entitas *customer* nantinya akan mengisi dari entitas keranjang dan entitas tujuan serta entitas *customer* juga akan memiliki dari entitas produk.
6. Entitas produk memiliki relasi dengan entitas kategori yaitu many-to-many karena pada entitas produk terdapat jenis kategori.
7. Entitas superuser memiliki relasi many-to-one dengan entitas shipping, artinya dari entitas tujuan dapat mengolah dan mengambil data dari entitas tujuan.
8. Entitas keranjang memiliki relasi many-to-one dengan entitas pembelian, artinya adanya data yang diisi pada entitas keranjang akan masuk juga pada entitas pembelian.
9. Entitas bank memiliki relasi many-to-one dengan entitas konfirmasi artinya data yang terdapat pada entitas bank akan terinput pada entitas konfirmasi
10. Entitas Kategori memiliki relasi many-to-one dengan entitas katalog artinya data pada entitas kategori akan terinput pada entitas katalog.

11. Entitas Keranjang memiliki relasi many to many dengan entitas pembelian artinya jika data keranjang terisi data maka pada entitas pembelian juga akan terisi data yang sama pada entitas keranjang.
12. Entitas tujuan memiliki relasi many-to-one dengan entitas shipping dikarenakan setiap data pada tabel shipping hanya diambil satu kolom saja untuk melengkapi data entitas tujuan.
13. Entitas produk memiliki relasi many to one dengan entitas kategori dan entitas katalog dikarenakan setiap data dari entitas kategori dan katalog dan sebagai inisialisasi pada data entitas produk.

Gambar 3.6 Menunjukkan rancangan tabel *Entity Relationship Diagram* sebagai berikut:

#### 1. Produk

Tabel Produk berisikan data-data dari produk yang akan ditampilkan pada *website*. Pada tabel ini yang disimpan berupa *id\_barang*, *id\_katalog*, *id\_kategori*, *motif*, *warna\_dasar*, *warna\_corak*, *harga*, *deskripsi*, *gambar*.

**Tabel 3. 1** Produk

No	Nama	Tipe	Keterangan
1	<i>Id_barang</i>	Int	Inisialisasi setiap barang yang diinput
2	<i>Id_katalog</i>	Int	Inisialisasi katalog barang
3	<i>Id_kategori</i>	Int	Inisialisasi kategori barang
4	Motif	Varchar	Nama motif dari barang
5	<i>Warna_dasar</i>	Varchar	Warna dari barang
6	<i>Warna_corak</i>	Varchar	Warna corak barang
7	Harga	Varchar	Harga Barang
8	Deskripsi	Varchar	Deskripsi pada setiap barang seperti ukuran dan lain-lain
9	Gambar	Varchar	Gambar barang



## 2. Kategori

Tabel Kategori berisikan data-data kategori dari produk. Data – data tersebut dipilih saat penginputan pada produk. Tabel kategori berisikan: id\_kategori, kategori. Berikut tabel kategori:

**Tabel 3. 2** Kategori

No	Nama	Tipe	Keterangan
1	Id_kategori	Int	Inisialisasi kategori
2	Kategori	Varchar	Nama dari kategori

## 3. Katalog

Tabel katalog berisikan data-data katalog yang akan ditampilkan pada produk. Tabel Katalog berisikan: id\_katalog, id\_kategori, katalog. Berikut Tabel Katalog:

**Tabel 3. 3** Katalog

No	Nama	Tipe	Keterangan
1	Id_katalog	Int	Inisialisasi data katalog
2	Id_kategori	Int	Inisialisasi data kategori diambil dari tabel kategori
3	Katalog	Varchar	Nama katalog

## 4. Stok

Tabel yang berisikan stok barang yang terhubung dengan tabel produk dengan inisialisasi id\_barang sebagai relasi. Tabel ini berisikan id\_stok, Id\_barang, stok. Berikut isi pada Tabel Stok:

**Tabel 3. 4 Stok**

No	Nama	Tipe	Keterangan
1	Id_stok	Int	Inisialisasi data stok
2	Id_barang	Int	Inisialisasi data barang yang mengambil data relasi dengan tabel produk
3	Stok	Varchar	stok

### 5. Shipping

Tabel shipping berisi data tarif pengiriman barang ke tiap kota. Data yang terdapat pada tabel ini seperti: id\_kota, tujuan, tarif. Berikut isi tabel shipping:

**Tabel 3. 5 Shipping**

No	Nama	Tipe	Keterangan
1	Id_kota	Int	Inisialisasi data kota
2	Tujuan	varchar	Kolom yang berisikan data kota
3	Tarif	Varchar	Kolom yang berisikan tarif dari setiap pengiriman barang ke kota tersebut

### 6. Keranjang

Tabel keranjang memiliki peranan penting pada setiap dilakukannya transaksi pada *customer*. Dikarenakan tabel ini berisi: id\_keranjang, kode beli, id\_cus, id\_barang, qty, harga, total\_harga, total\_bayar, qty\_total, status beli. Berikut rincian tabelnya.

**Tabel 3. 6 Keranjang**

No	Nama	Tipe	Keterangan
1	Id_keranjang	Int	Inisialisasi no keranjang
2	Kode_beli	varchar	Sebagai kode inisialisasi pembelian
3	Id_cus	int	Id yang diambil dari tabel <i>customer</i>
4	Id_barang	int	Id yang diambil dari tabel produk
5	qty	Varchar	Jumlah barang yang di beli
6	Harga	Varchar	Harga barang
7	Total Harga	Varchar	Harga total barang yang dibeli
8	Total Bayar	Varchar	Harga yang harus dibayarkan
9	Qty_total	Varchar	Jumlah seluruh barang yang dibeli
10	Status_beli	Varchar	Status pembeliannya

7. Selesai

Tabel yang berisi data status pembelian dari *customer*. Yang dimaksudkan adalah status pembelian dan status pengiriman barang. Berikut tabel selesai:

**Tabel 3. 7 Selesai**

No	Nama	Tipe	Keterangan
1	Kode_beli	varchar	Inisialisasi kode pembelian
2	Id_cus	int	Id <i>customer</i> diambil dari tabel <i>customer</i>

No	Nama	Tipe	Keterangan
3	Qty_total	Varchar	Jumlah total barang yang dibeli
4	Bayar	varchar	Harga yang harus dibayar
5	Total_bayar	varchar	Total pembayaran
6	Tgl_order	text	Tanggal order
7	Status_beli	Enum	Status pembelian
8	Status_pengiriman	Enum	Status pengiriman barang

#### 8. *Customer*

Terdapat data setiap *customer* yang telah melakukan registrasi. Isi tabelnya berupa *id\_cus*, *nama\_cus*, *email\_cus*, *password\_cus*. Berikut tabel *customer*:

**Tabel 3. 8** Customer

No	Nama	Tipe	Keterangan
1	<i>Id_cus</i>	Int	Inisialisasi <i>customer</i>
2	<i>Nama Cus</i>	varchar	Nama <i>customer</i>
3	<i>Email</i>	Varchar	Email <i>Customer</i>
4	<i>Password_cus</i>	varchar	Password <i>Customer</i>

#### 9. *Superuser*

Tabel yang berisi data admin. Berikut tabel superuser:

**Tabel 3. 9 Superuser**

No	Nama	Tipe	Keterangan
1	Id_su	Int	Inisialisasi admin
2	Nama_su	varchar	Nama admin
3	Email_su	Varchar	Email dari admin
4	Password_su	varchar	Password admin
5	Level	Enum	Berisi pilihan owner ataupun admin

#### 10. Pembelian

Tabel yang berisi kurang lebih sama dengan tabel keranjang namun tabel ini dimaksudkan sebagai data yang akan masuk jika *customer* selesai melakukan pembelian.

**Tabel 3. 10 Pembelian**

No	Nama	Tipe	Keterangan
1	Id_beli	Int	Inisialisasi id pembelian
2	Kode_beli	varchar	Inisialisasi kode pembelian
3	Id_cus	int	Inisialisasi <i>customer</i>
4	Id_barang	int	Id yang diambil dari tabel produk
5	qty	Varchar	Jumlah barang yang di beli
6	harga	Varchar	Harga barang

No	Nama	Tipe	Keterangan
7	Total harga	Varchar	Harga total barang yang dibeli
8	Total bayar	Varchar	Harga yang harus dibayarkan
9	Qty total	Varchar	Jumlah seluruh barang yang dibeli

### 11. Tujuan

Berisi data *customer* setelah melakukan proses checkout. Data tersebut digunakan untuk mengetahui alamat *customer* yang bertujuan sebagai pendataan pengiriman barang yang telah dibeli oleh *customer*.

**Tabel 3. 11 Tujuan**

No	Nama	Tipe	Keterangan
1	Id_tujuan	Int	Inisialisasi id tujuan pada tabel shipping
2	Kode_beli	varchar	Inisialisasi kode pembelian
3	Nama penerima	varchar	Nama penerima barang yang dikirimkan
4	Tujuan	varchar	Kota pengiriman barang
5	Alamat lengkap	varchar	alamat
6	Kode pos	Varchar	Kode pos
7	No telpon	Varchar	Nomor telpon yang bisa dihubungi
8	Tarif	Varchar	Harga total barang yang dibeli

## 12. Konfirmasi

Berisi data Konfirmasi Pembayaran dari User:

**Tabel 3. 12** Konfirmasi

No	Nama	Tipe	Keterangan
1	No_konfirmasi	Int	Inisialisasi id konfirmasi pada tabel konfirmasi
2	Nama Pembeli	varchar	Nama Pembeli
3	Bayar via	varchar	Pembayaran melalui transfer ke bank
4	Tanggal	varchar	Tanggal Pengiriman Bukti
5	Jumlah	varchar	Jumlah Pembayaran
6	Gambar	Varchar	Bukti Transfer

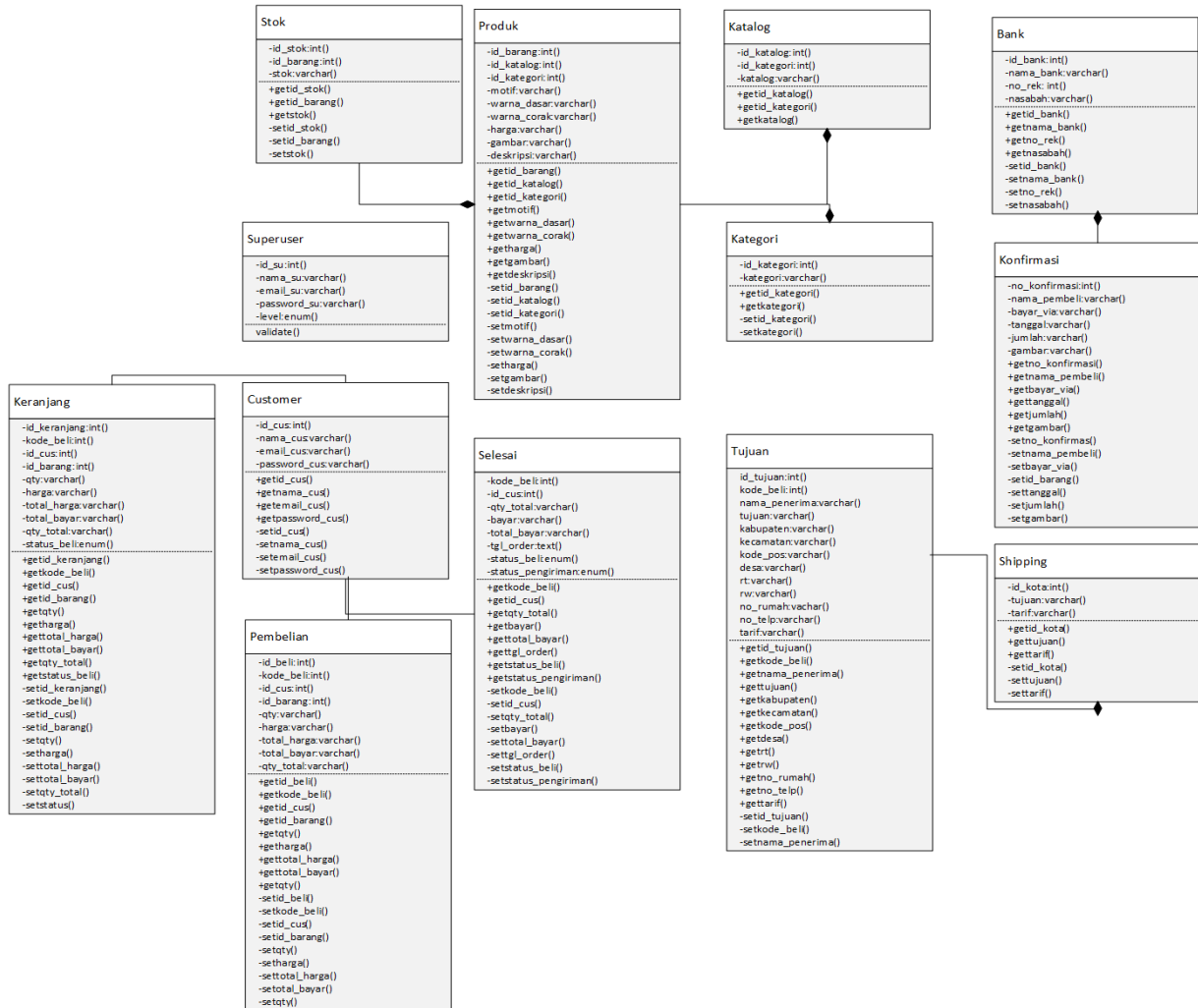
## 13. Bank

Tabel Berisi data bank dari CV. Warisan Multi Tenun.

**Tabel 3. 13** Bank

No	Nama Kolom	Tipe	Keterangan
1	Id_bank	<i>Int</i>	<i>Id bank</i>
2	Nama_bank	<i>Varchar</i>	Nama Bank
3	No_rek	<i>Varchar</i>	Nomor rekening
4	Nasabah	<i>Varchar</i>	Nama Nasabah

### 3.4.4 Class Diagram



Gambar 3.7 Class Diagram

Penjelasan fungsi:

1. Class Produk  
Berfungsi menampilkan data produk dan memasukkan data kategori dan katalog pada Class Kategori dan Class Katalog.
2. Class Katalog  
Berfungsi memasukkan data dari class kategori dan menampilkan data katalog.
3. Class Kategori  
Berfungsi menampilkan data dari kategori.



4. Class Keranjang  
Berfungsi memasukkan data dari class customer dan class pembelian serta menampilkan data pada class selesai.
5. Class Stok  
Berfungsi memasukkan stok yang terhubung dengan stok pada class produk.
6. Class Tujuan  
Berfungsi memasukkan data dari class selesai dan menampilkan data dari class *shipping*.
7. Class Superuser  
Berisi data admin yang telah di buat.
8. Class Pembelian  
Berisi data dari pembelian yang dilakukan oleh customer.
9. Class *Shipping*  
Berisi data shipping berupa kota dan harga pengiriman barang yang akan di tampilkan pada class selesai dan class tujuan.
10. Class Selesai  
Menampilkan data yang telah di inputkan oleh class *customer* dengan isi status pengiriman dan status beli.
11. Class *Customer*  
Menampilkan data customer yang telah melakukan registrasi.
12. Class Konfirmasi  
Berisi data bukti transfer dari user yang telah di *upload*
13. Class Bank  
Berisi data bank sebagai tujuan pembayaran user nantinya.

Berikut penjelasan untuk *class* diagram pada gambar 3.7:

1. Terdapat kelas pada diagram yang digunakan untuk inialisasi ke dalam tabel dan basis data aplikasi.
2. Kelas produk memiliki *asosiasi* dengan kelas kategori dan kelas katalog artinya setiap data kelas barang dapat memilih satu kategori dan satu katalog.

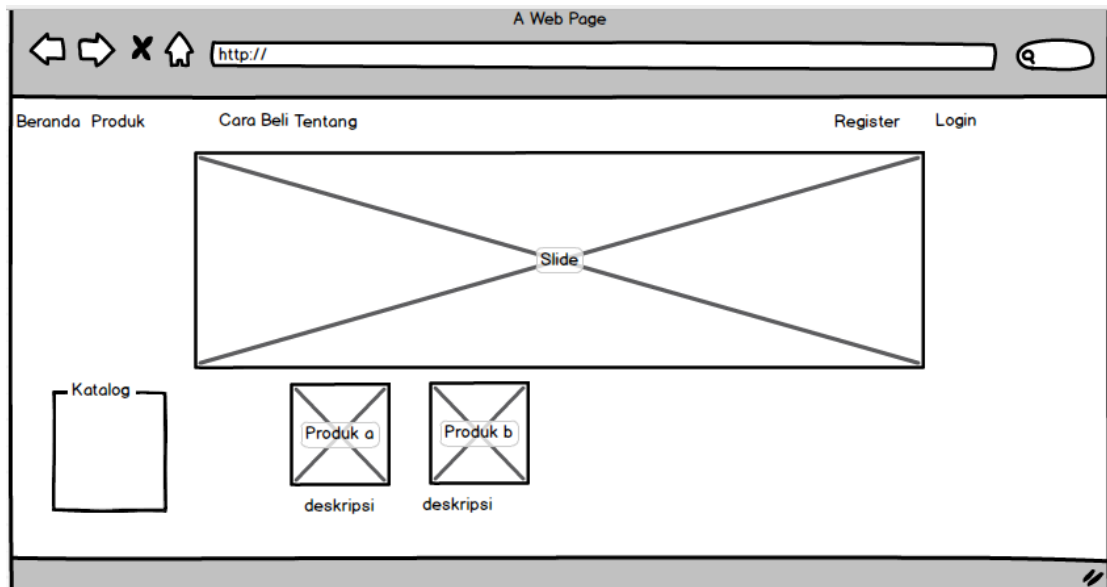
3. Kelas keranjang, kelas pembelian dan kelas selesai memiliki *composition* dengan kelas *customer*, artinya setiap kelas keranjang, kelas pembelian dan kelas selesai tidak dapat diisi jika kelas *customer* tidak melakukan transaksi.
4. Kelas tujuan memiliki *composition* dengan kelas *customer*, artinya jika kelas *customer* tidak mengisi atau tidak melakukan transaksi kelas tujuan tidak akan terisi.
5. Kelas *customer* memiliki *assosiasi* dengan kelas shipping, artinya setiap kelas *customer* dapat memilih data dari kelas shipping.
6. Kelas tujuan memiliki *composition* dengan kelas shipping, artinya jika data kelas tujuan tidak diisi maka data kelas shipping kosong.
7. Kelas Katalog memiliki *assosiasi* dengan kelas kategori, artinya setiap data katalog dapat memilih satu kategori.
8. Kelas pembelian memiliki *composition* dengan kelas keranjang, artinya jika tidak ada pembelian kelas keranjang kosong.
9. Kelas produk memiliki *assosiasi* dengan kelas selesai artinya jika tidak ada produk yang dipilih maka kelas selesai kosong.
10. Kelas stok memiliki *assosiasi* dengan kelas produk, artinya setiap data produk dapat menambah data stok
11. Kelas selesai memiliki *composition* dengan kelas *customer* artinya jika data kelas selesai tidak dapat diisi apabila data kelas *customer* kosong.
12. Kelas konfirmasi memiliki *assosiasi* dengan kelas bank artinya setiap kelas konfirmasi dapat memilih data dari kelas bank.

## **1.5 Rancangan Antar Muka**

Setiap aplikasi harus memiliki desain antarmuka. Rancangan sistem atau aplikasi pada umumnya terdapat sarana berinteraksi antara pengguna dengan sistem atau aplikasi. Pada setiap rancangan otomatis nantinya akan ditampilkan kepada customer CV. Warisan Multi Tenun yang bertujuan untuk mempermudah pemasaran serta penjualan produk dari CV. Warisan Multi Tenun.

### 3.5.1 Rancangan Antar Muka Halaman Utama

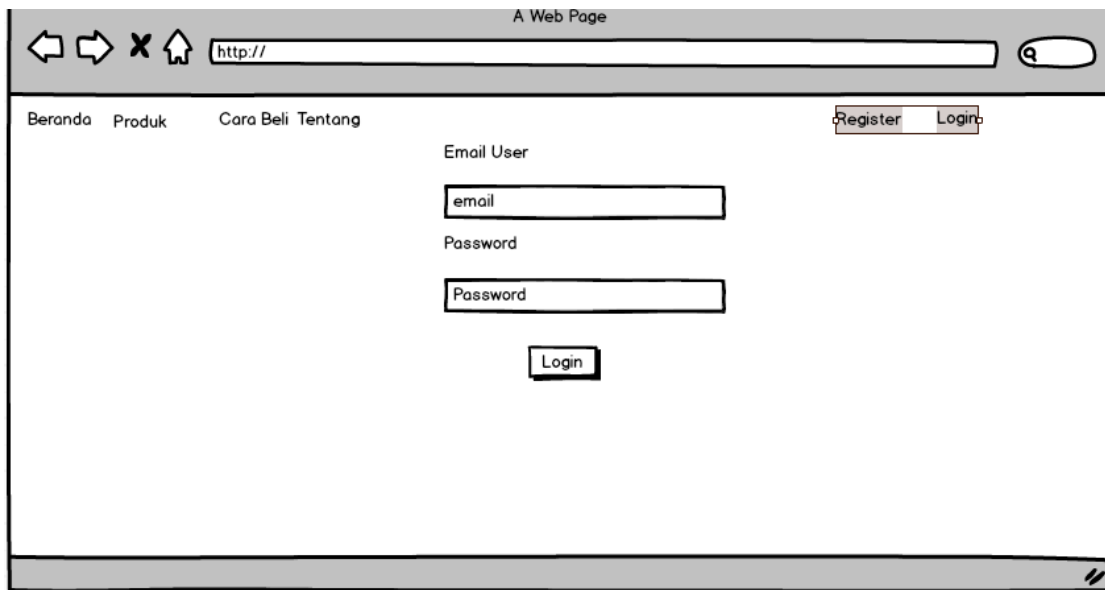
Rancangan antarmuka halaman awal merupakan tampilan awal saat aplikasi dijalankan. Gambaran rancangan antarmuka halaman awal dapat dilihat pada Gambar 3.8. Pada Gambar 3.8 menampilkan halaman home atau lebih tepatnya halaman utama.



**Gambar 3. 8** Halaman Awal

### 3.5.2 Rancangan Antar Muka Halaman *Login*

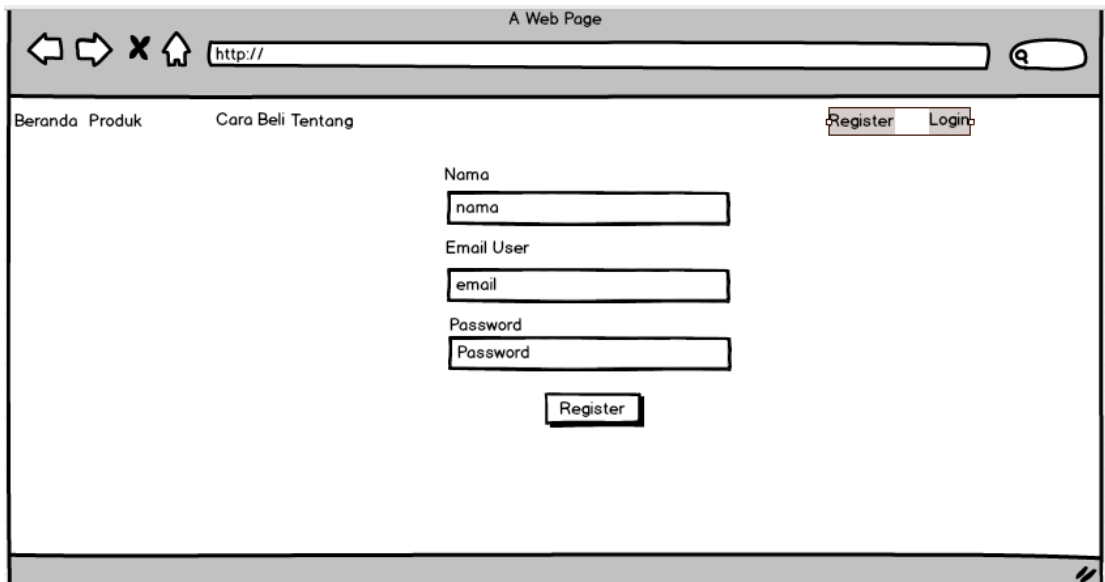
Rancangan antar muka halaman login menampilkan halaman dimana *customer* dan admin melakukan login namun dari kedua user tersebut akan menuju halaman yang berbeda jika telah melakukan login. Semisal admin melakukan login maka admin akan masuk pada halaman beranda admin sedangkan *customer* melakukan *login* maka *customer* akan masuk pada beranda *customer*.



Gambar 3. 9 Halaman Login

### 3.5.3 Rancangan Antar Muka Halaman Registrasi

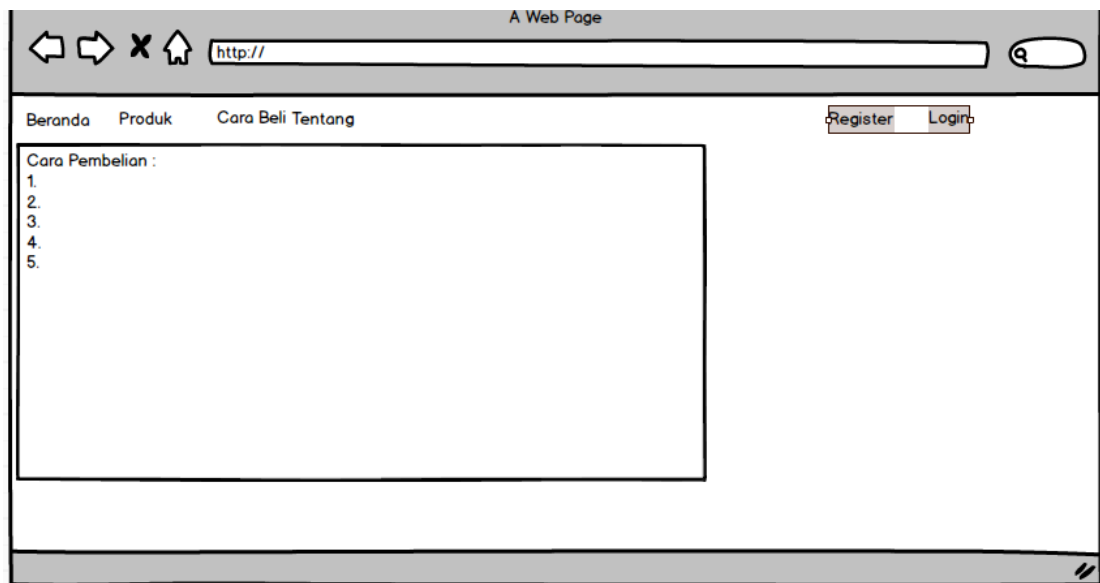
Rancangan antar muka halaman registrasi menampilkan halaman registrasi dimana jika *customer* belum memiliki akun pada *website* ini maka *customer* sendiri tidak dapat membeli produk atau melakukan transaksi



Gambar 3. 10 Halaman Registrasi

### 3.5.4 Rancangan Antar Muka Halaman Cara Pembelian

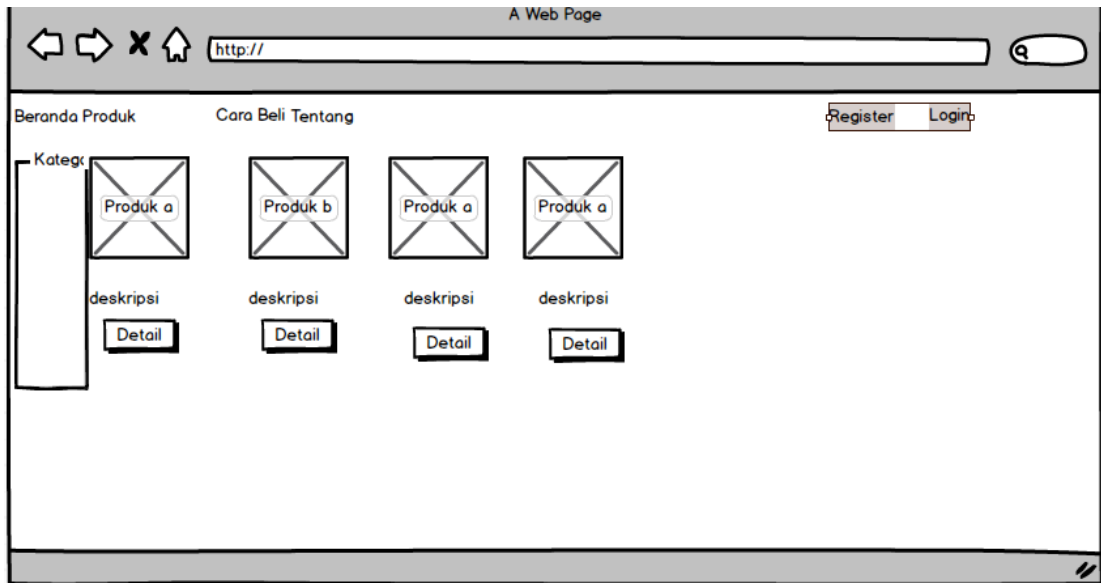
Halaman pada rancangan antar muka ini lebih menunjukkan halaman-halaman umum yang dilihat oleh user sebelum memiliki akun atau belum registrasi. Halaman –halaman berikut ada pada Gambar 3.11. Pada gambar 3.11 menunjukan halaman cara pembelian, user tanpa login ataupun tidak memiliki akun dapat melihat halaman tersebut.



**Gambar 3. 11** Halaman Cara Beli

### 3.5.5 Rancangan Antar Muka Halaman Produk

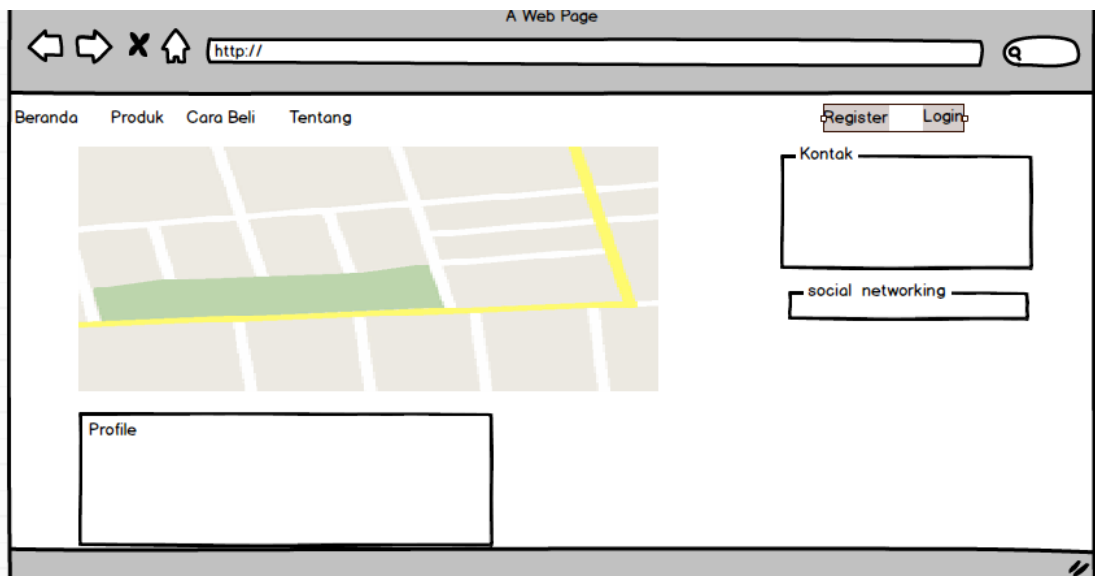
Pada Gambar 3.12 menampilkan halaman produk. Halaman tersebut dapat diakses oleh user yang telah login maupun yang belum memiliki akun. Isi halaman produk tidak lain adalah produk yang telah diunggah oleh admin agar dipasarkan kepada *customer* agar *customer* dapat mengetahui produk yang tersedia pada *website*. Pada setiap produk yang ditampilkan kepada *customer*, admin memberikan stok pada setiap produk yang dipasarkan.



**Gambar 3. 12** Halaman Produk

### 3.5.6 Rancangan Antar Muka Halaman Tentang Kami (Kontak)

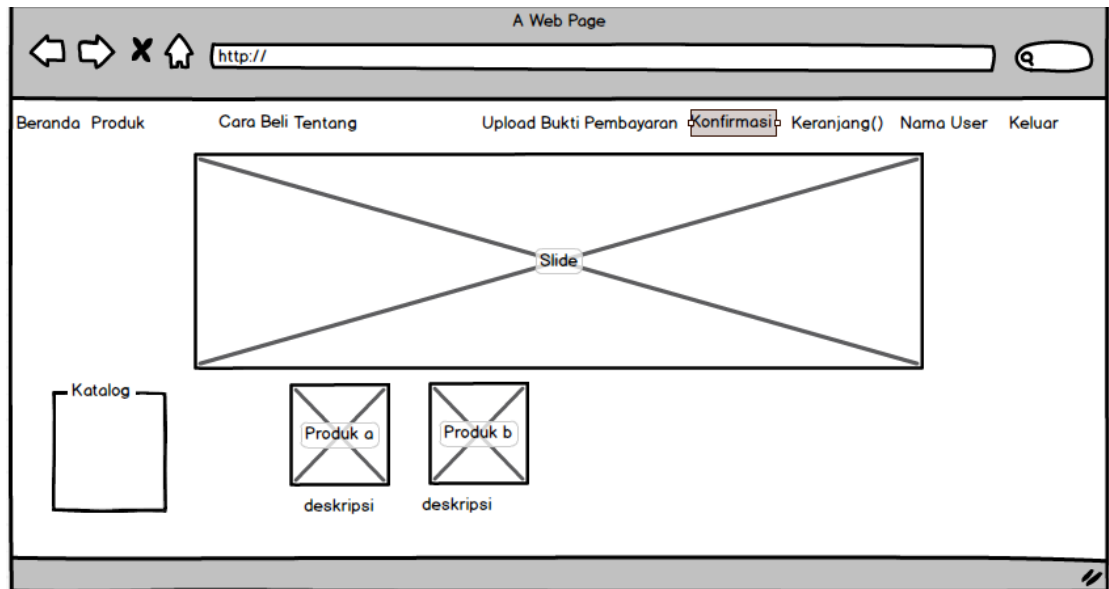
Halaman pada Gambar 3.13 menampilkan halaman Tentang Kami. Yang berisi kontak yang bisa dihubungi *customer*. Pada halaman kontak terdapat kontak yang dapat dihubungi oleh *customer*, serta map untuk mengetahui alamat produksi dari CV. Warisan Multi Tenun. Pada halaman kontak admin juga menyisipkan profil dari CV. Warisan Multi Tenun.



**Gambar 3. 13** Halaman Tentang Kontak

### 3.5.7 Rancangan Antar Muka Halaman User

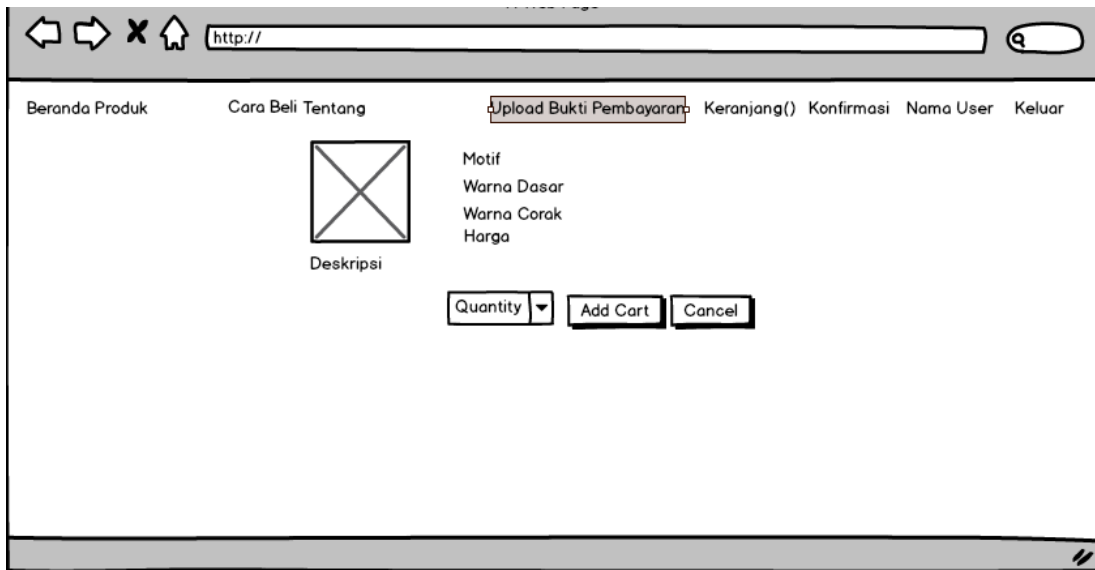
Rancangan antar muka pada halaman user lebih menunjukkan halaman setelah user login sebagai contoh pada Gambar 3.14 lebih menunjukkan halaman setelah user melakukan login.



Gambar 3. 14 Halaman User Home

### 3.5.8 Rancangan Antar Muka Halaman Detail

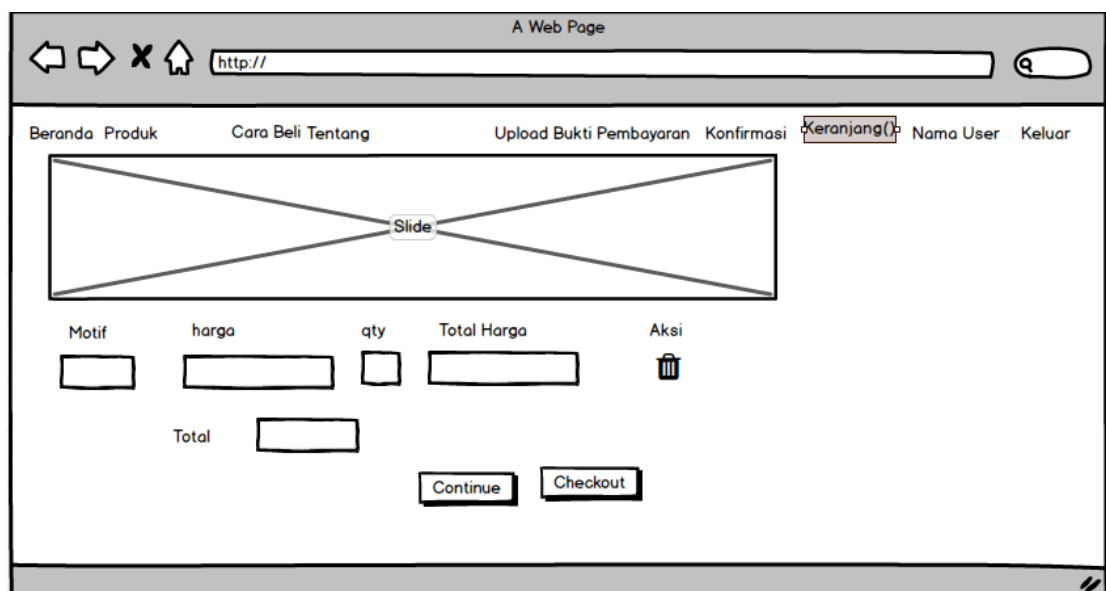
Pada Gambar 3.15 menunjukkan halaman detail produk. Detail produk sendiri memberikan informasi tentang produk yang ingin dilihat oleh *customer*. Pada halaman detail produk *customer* diberikan informasi berupa gambar produk, nama produk, harga, jenis kategori produk, deskripsi produk. Jika *customer* telah melakukan login pada *website*, *customer* dapat langsung melakukan pembelian dengan menekan tombol *add to cart* pada halaman detail sedangkan jika *customer* belum melakukan login maka secara otomatis tombol *add to cart* tidak akan muncul pada halaman detail.



**Gambar 3. 15** Halaman Add Cart

### 3.5.9 Rancangan Antar Muka Halmaan Keranjang

*Customer* akan dihadapkan dengan halaman yang berisi apapu yang telah di pilih customer melalui halaman detail dengan menekan tombol *add to cart* pada halaman tersebut. Setelah barang sdah dipilih maka otomatis data barang tersebut akan masuk dalam halaman keranjang.

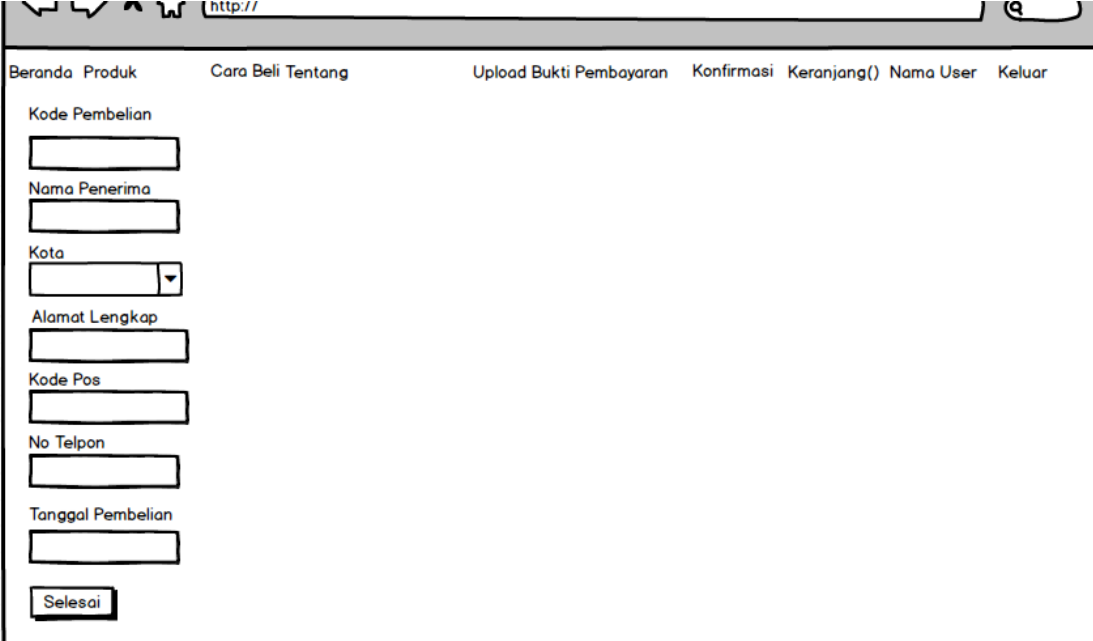


**Gambar 3. 16** Halaman Keranjang



### 3.5.10 Rancangan Antar Muka Halaman Checkout

Halaman selanjutnya setelah halaman keranjang adalah halaman checkout, yaitu halaman *website* dimana *customer* harus mengisi data diri untuk pengiriman barang. Mulai dari nama penerima, kota tujuan, alamat lengkap, kode pos, dan nomor telepon.



The image shows a web browser window displaying a checkout form. The browser's address bar shows "http://". The page has a navigation menu with links: Beranda, Produk, Cara Beli Tentang, Upload Bukti Pembayaran, Konfirmasi, Keranjang(), Nama User, and Keluar. The form contains the following fields:

- Kode Pembelian:
- Nama Penerima:
- Kota:
- Alamat Lengkap:
- Kode Pos:
- No Telpn:
- Tanggal Pembelian:
- Selesai:

Gambar 3. 17 Halaman Checkout

### 3.5.11 Rancangan Antar Muka Halaman Konfirmasi


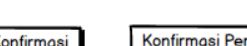
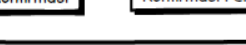
Halaman ini akhir dari proses transaksi, yaitu *customer* akan mendapatkan keterangan data pengiriman barang yang telah tadi diisi oleh *customer* serta dengan total harga yang harus dibayarkan. *Customer* terlebih dahulu melakukan *upload* bukti pembayaran pada tombol konfirmasi peembayaran yang telah tersedia. Jika sudah mengirim bukti dan belum diberikan pemberitahuan tentang pengiriman barang, *customer* dapat menghubungi WhatsApp ataupun dengan email. Setelah itu jika user telah menerima barang harap melakukan konfirmasi jika barang sudah diterima melalui tombol konfirmasi barang diterima.

[Beranda](#) [Produk](#) [Cara Beli Tentang](#) [Upload Bukti Pembayaran](#) [Konfirmasi](#) [Keranjang\(\)](#) [Nama User](#) [Keluar](#)

Kode Pembelian  
 Nama Penerima  
 Kota  
 Alamat Lengkap  
 Kode Pos  
 No Telpon

Motif	harga	qty	Sub Tot
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Tarif Pengiriman   
 Tarif Pembayaran

[Beranda](#) [Produk](#) [Cara Beli Tentang](#) [Upload Bukti Pembayaran](#) [Konfirmasi](#) [Keranjang\(\)](#) [Nama User](#) [Keluar](#)

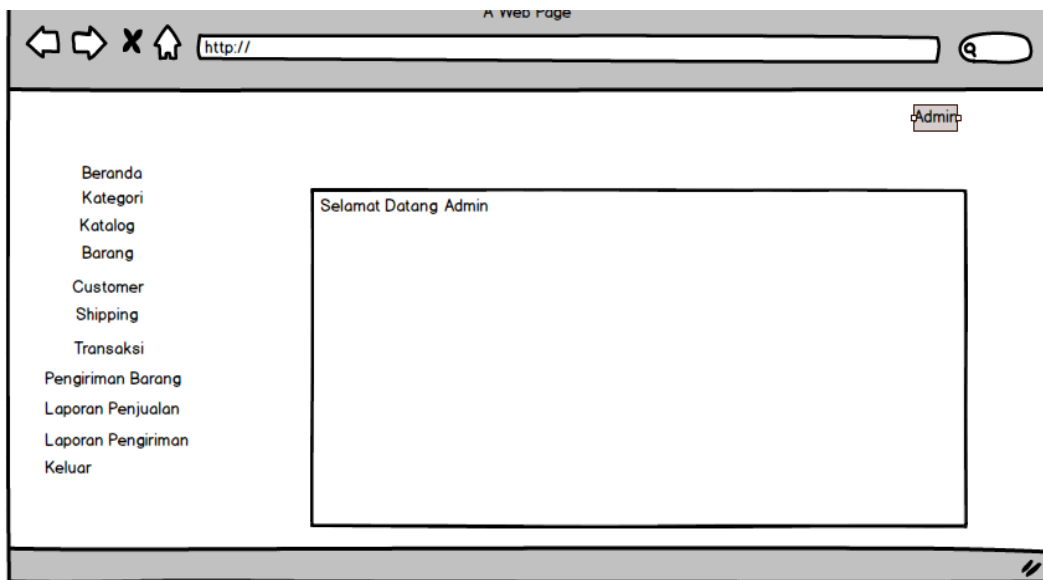
Kategori   
 Nama Pembeli   
 Pembayaran   
 Tanggal   
 Jumlah   
 Gambar

**Gambar 3. 18** Halaman Konfirmasi

### 3.5.12 Rancangan Antar Muka Halaman Admin

Rancangan pada halaman admin memiliki beberapa tab halaman, dimana admin dapat mengolah data dari halaman tersebut. Seperti pada Gambar 3.19, Gambar 3.20, Gambar 3.21, Gambar 3.22, Gambar 3.23, Gambar 3.24, Gambar 3.25, Gambar 3.26, Gambar 3.27, Gambar 3.28, Gambar 3.29, Gambar 3.30.

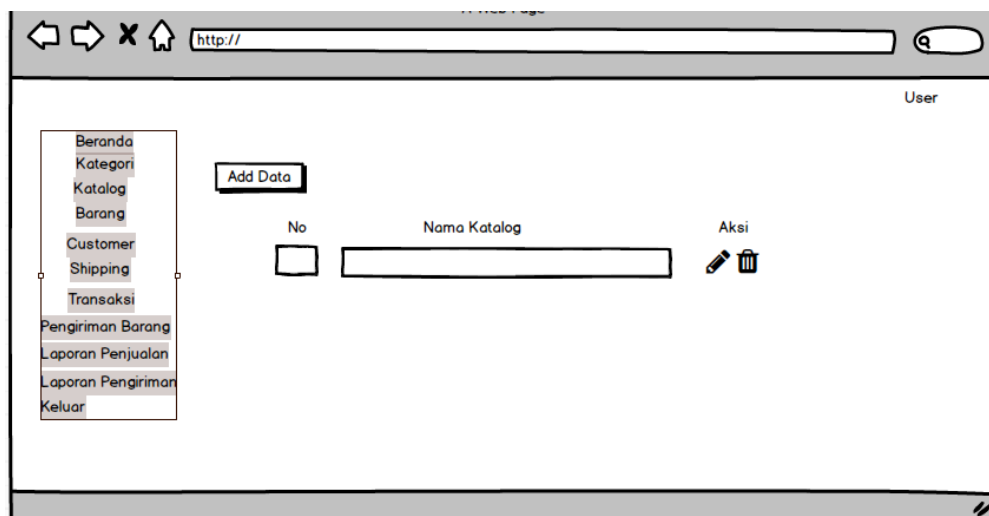
Namun pada halaman Laporan Penjualan dan Laporan Pengiriman hanya sebagai tab link download pdf yang berisi data penjualan dan data pengiriman. Berikut pada Gambar 3.19 menampilkan halaman utama admin jika admin telah melakukan proses login.



**Gambar 3. 19** Halaman Utama Admin

### 3.5.13 Rancangan Antar Muka Halaman Admin Kategori

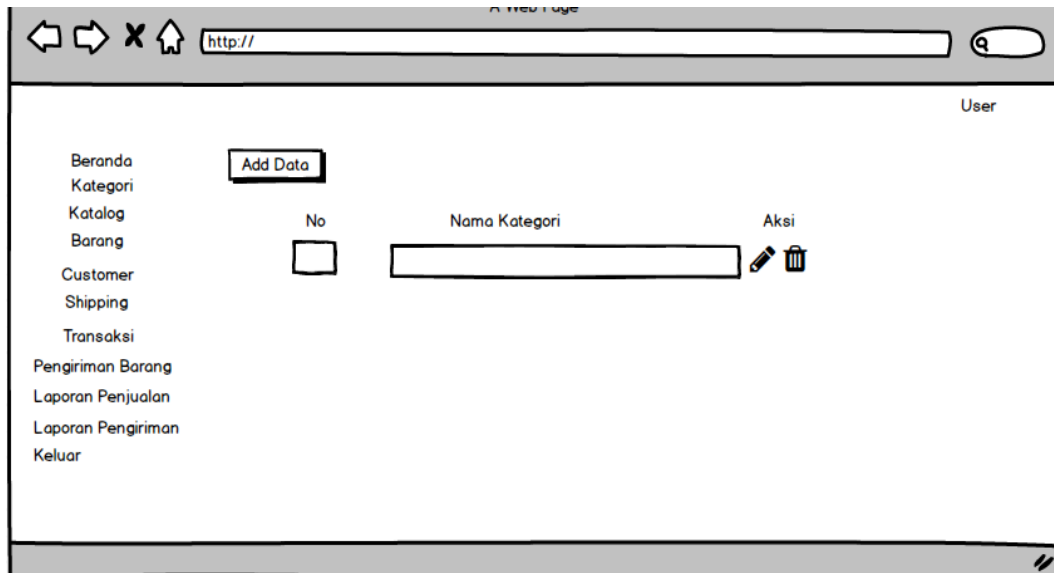
Pada gambar 3.20 menunjukkan halaman kategori dimana, admin dapat melakukan proses insert, update, dan delete pada data kategori.



**Gambar 3. 20** Halaman Kategori

### 3.5.14 Rancangan Antar Muka Halaman Tambah Kategori

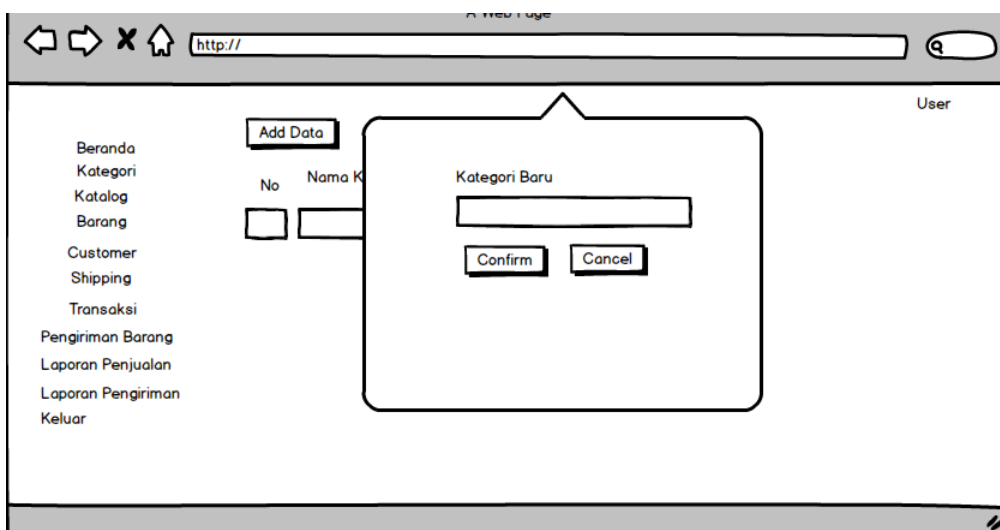
Gambar 3.21 menampilkan *pop up* pada proses *Add Data*. Admin melakukan proses memasukkan (*insert*) data baru pada tab kategori.



Gambar 3. 21 Halaman Tambah Kategori

### 3.5.15 Rancangan Antar Muka Halaman Katalog

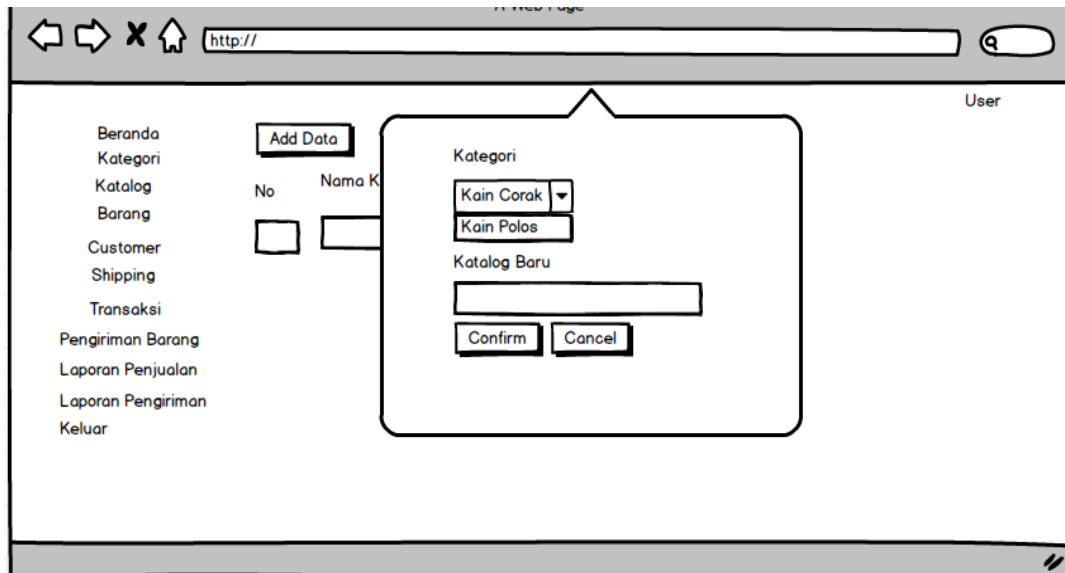
Berikut Gambar 3.22 dengan menampilkan halaman pada tab katalog. Sama seperti halaman kategori, halaman ini memberikan akses admin untuk memasukkan data *input*, *edit*, dan *delete*.



Gambar 3. 22 Halaman Katalog

### 3.5.16 Rancangan Antar Muka Halaman Tambah Katalog

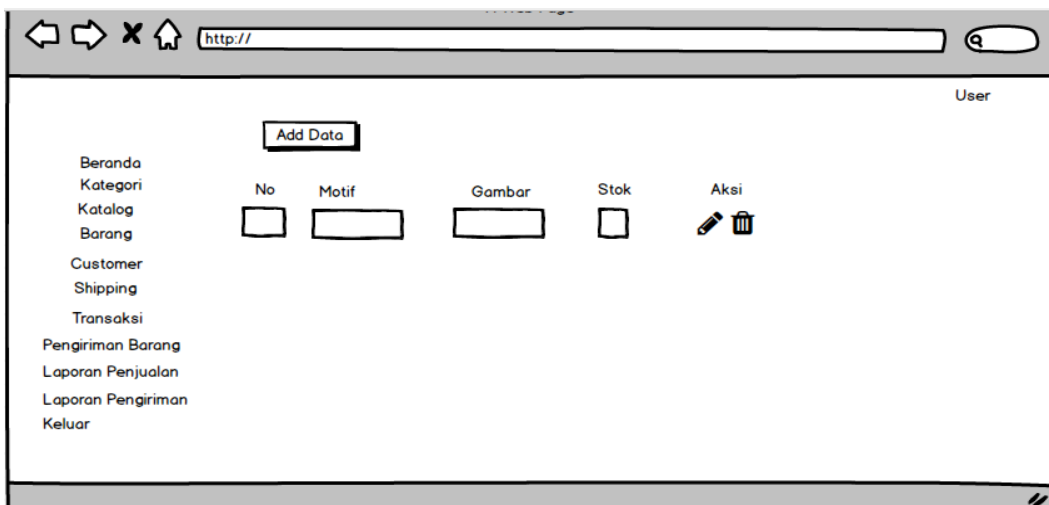
Gambar 3.23 menampilkan *pop up* pada proses *Add Data*. Admin melakukan proses memasukkan (*insert*) data baru pada tab kategori.



Gambar 3. 23 Halaman Tambah Katalog

### 3.5.17 Rancangan Antar Muka Halaman Barang

Halaman pada Gambar 3.24 adalah halaman *Barang* menunjukkan data yang berisi barang-barang yang dimasukkan oleh admin. Pada halaman ini admin diberikan hak akses untuk *input*, *edit*, *delete*.



Gambar 3. 24 Halaman Barang

### 3.5.18 Rancangan Antar Muka Halaman Tambah Barang

Pada gambar 3.25 menunjukkan *pop up* untuk menambah data pada halaman barang.

The image shows a web application interface with a sidebar menu on the left and a main content area. The sidebar menu includes: Beranda, Kategori, Katalog, Barang, Customer, Shipping, Transaksi, Pengiriman Barang, Laporan Penjualan, Laporan Pengiriman, and Keluar. The main content area displays a table with columns 'No' and 'Nama Bara'. A 'pop up' window titled 'Add Data' is overlaid on the table. This window contains the following fields: 'Kategori' (dropdown menu with 'Batik' selected), 'Katalog' (dropdown menu with 'Polos' selected), 'Motif' (text input), 'Warna Dasar' (text input), 'Warna Corak' (text input), 'Harga' (text input), 'Deskripsi' (text input), and 'Gambar' (with a 'Browse' button). At the bottom of the pop-up are 'Confirm' and 'Cancel' buttons. The word 'User' is visible in the top right corner of the main content area.

Gambar 3. 25 Halaman Add Barang

### 3.5.19 Rancangan Antar Muka Halaman *Customer* pada Admin

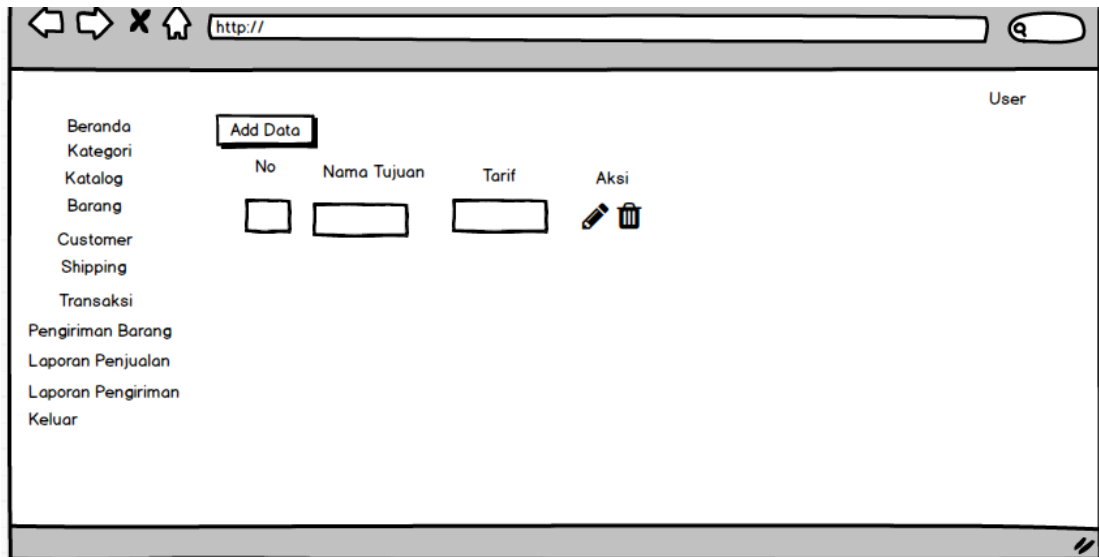
Halaman *customer* menampilkan data *customer* yang telah melakukan registrasi, disini admin hanya dapat menghapus akun *customer* saja.

The image shows a browser window displaying the 'Customer' page. The browser's address bar shows 'http://'. The page layout is consistent with the previous image, featuring the same sidebar menu. The main content area displays a table with columns 'No', 'Nama Customer', and 'Aksi'. The 'Aksi' column contains a trash can icon, indicating a delete function. The word 'User' is visible in the top right corner of the main content area.

Gambar 3. 26 Halaman *Customer*

### 3.5.20 Rancangan Antar Muka Halaman Shipping

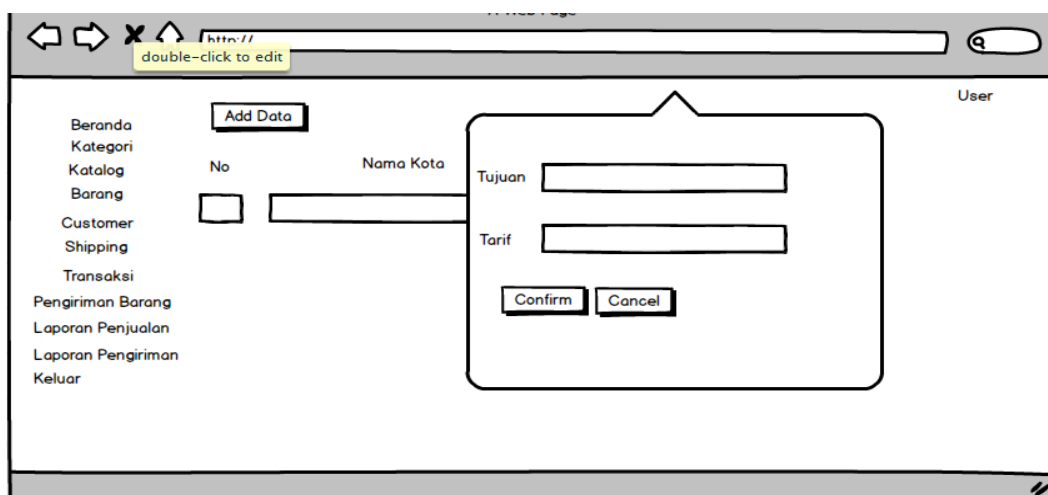
Gambar 3.27 pada rancangan antar muka menunjukkan halaman shipping dimana admin dapat memasukkan, mengedit dan menghapus data yang ada pada halaman shipping.



Gambar 3. 27 Halaman Shipping

### 3.5.21 Rancangan Antar Muka Halaman Tambah Shipping

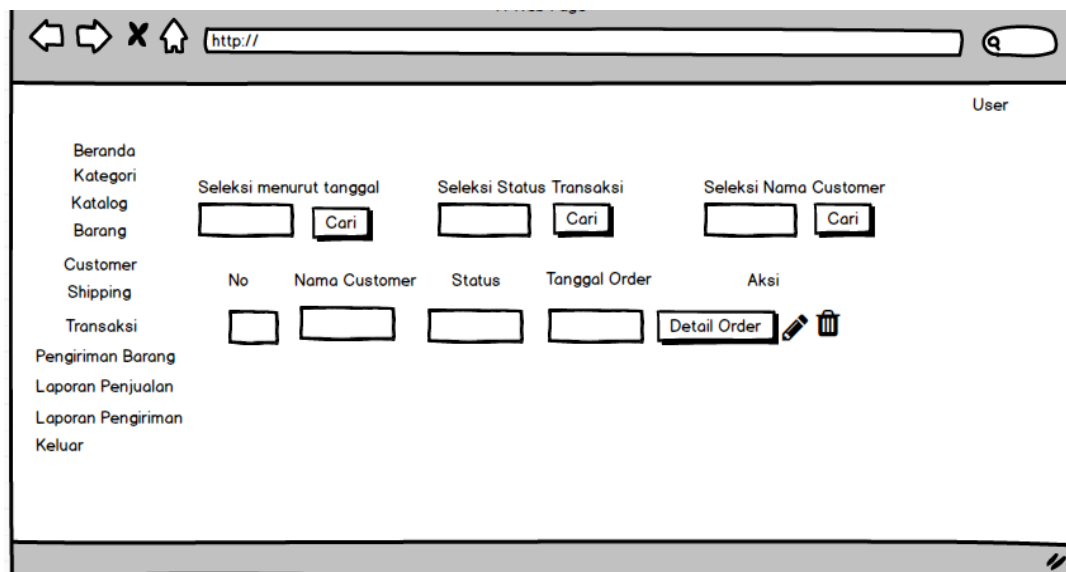
Pada gambar 3.28 menunjukkan *pop up* untuk menambah data pada halaman shipping.



Gambar 3. 28 Halaman Add Data Shipping

### 3.5.22 Rancangan Antar Muka Halaman Transaksi

Rancangan Antar muka halaman pada gambar 3.29 menunjukkan halaman status transaksi dimana halaman tersebut bertujuan untuk mengubah status dari pembelian yang dilakukan oleh *customer*. Jika *customer* telah membayar maka status transaksi akan diubah oleh admin menjadi lunas. Admin dapat mengubahnya serta dapat melihat detail order serta menghapus data tersebut.

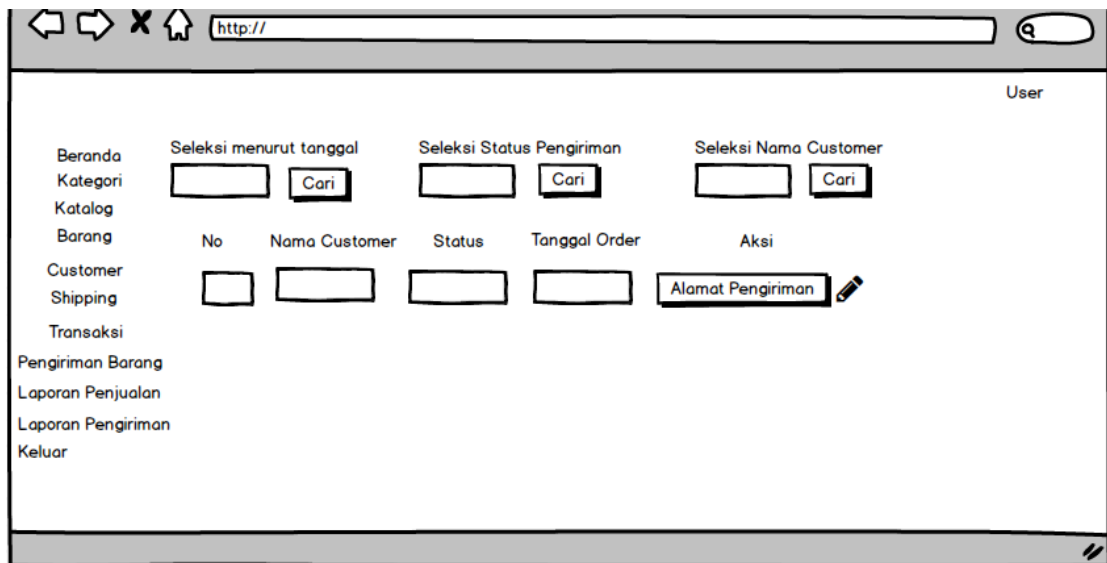


Gambar 3. 29 Halaman Transaksi

### 3.5.23 Rancangan Antar Muka Halaman Status Pengiriman

Gambar 3.30 pada rancangan Antar Muka Halaman Status pengiriman menunjukkan data pengiriman barang kepada *customer*. Admin dengan melihat alamat pengiriman akan mengetahui alamat *customer*. Serta dapat mengubah status pengiriman menjadi “dikirm” sehingga *customer* dapat melakukan konfirmasi melalui halaman konfirmasi yang tersedia pada *customer*.

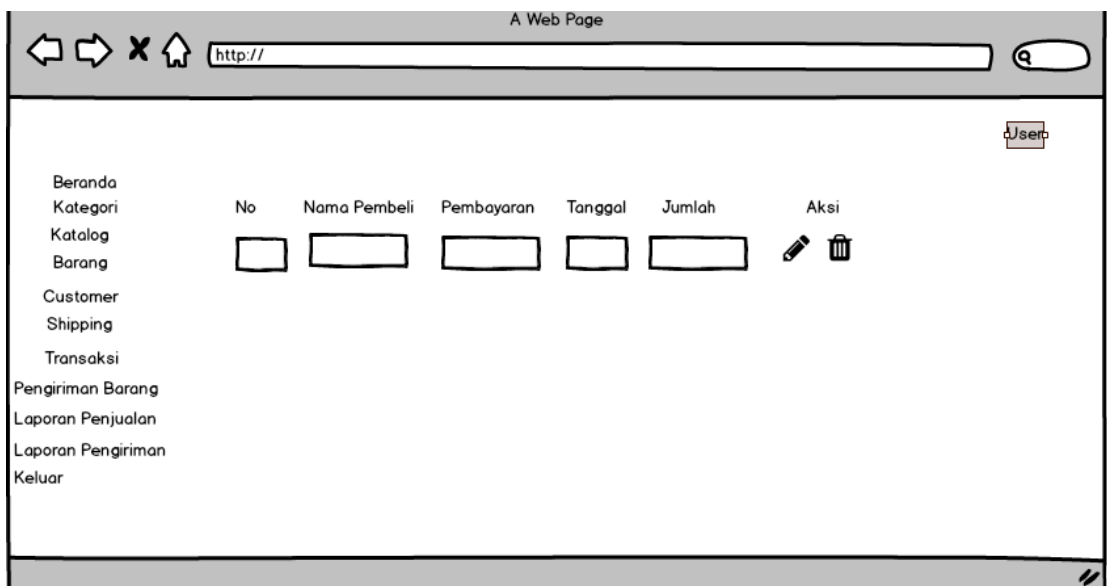




**Gambar 3. 30** Halaman Pengiriman Barang

### 3.5.24 Rancangan Antar Muka Halaman Pembayaran

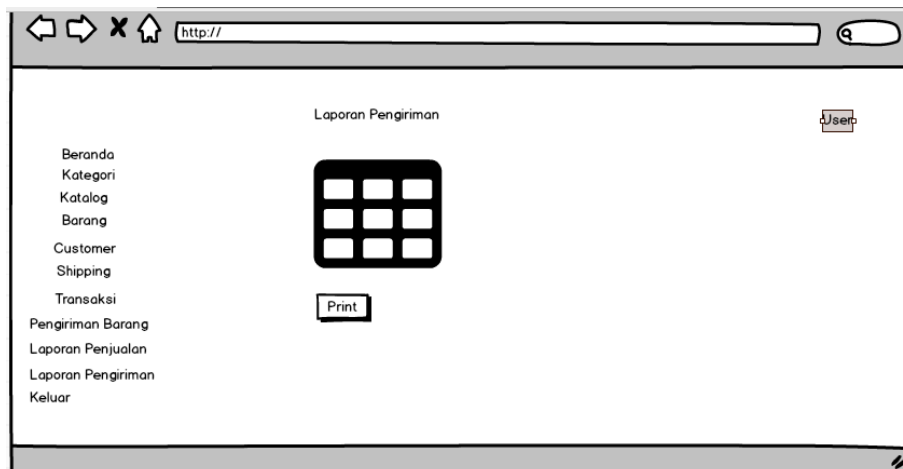
Halaman pembayaran data bukti transfer dari *customer*.



**Gambar 3. 31** Halaman Pembayaran

### 3.5.25 Rancangan Antar Muka Halaman Laporan Penjualan

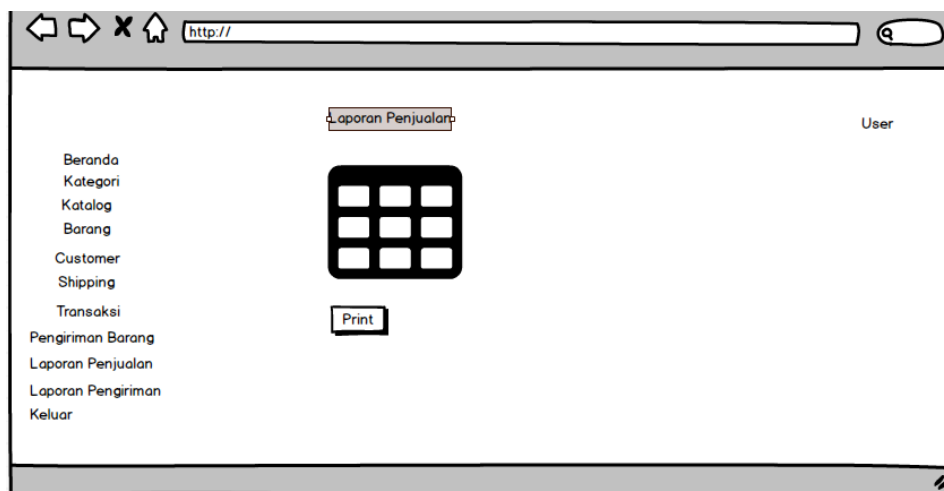
Halaman Laporan Penjualan berisi data laporan penjualan yang telah tercatat pada halaman ini terdapat tombol print untuk mencetak laporan penjualan.



**Gambar 3. 32** Rancangan Halaman Laporan Penjualan

### 3.5.26 Rancangan Antar Muka Halaman Laporan Pengiriman

Halaman Laporan Pengiriman berisi data laporan pengiriman pada *customer*. Sehingga admin dapat melihat data untuk pengiriman barang. Serta terdapat tombol print untuk mencetak laporan pengiriman.



**Gambar 3. 33** Rancangan Halaman Laporan Pengiriman