BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. Implementasi

Untuk membangun perangkat lunak ini, penulis menggunakan Google Chrome 27.0.1453.12, XAMPP 1.7.0, PHP dan MySQL. Google Chrome digunakan sebagai *web browser*, XAMPP versi 1.7.0 dipilih sebagai perangkat lunak *web server*, PHP dipilih sebagai perangkat lunak pengembang karena menyediakan fasilitas yang memadai dan membuat perangkat lunak berbasis web, sementara MySQL digunakan sebagai perangkat lunak dalam pembuatan basis data *(database).*

5.1.1. Batasan Implementasi

Berikut ini merupakan batasan implementasi dalam perangkat lunak sistem informasi *e-commerce* ini. Hanya subsistem-subsistem yang mendukung khususnya bagian pengolahan data dan pendaftaran calon pelanggan.

5.1.2. Implementasi Perangkat Lunak

Untuk mendukung sistem yang diusulkan berjalan dengan optimal, dibutuhkan *software* pengolahan data, adapun perangkat lunak yang digunakan untuk mendukung pembuatan program aplikasi ini sebagai berikut :

1. Sistem Operasi

   Sistem operasi yang digunakan dalam menjalankan aplikasi ini adalah Miscrosoft Windows 7 Ultimate. Digunakannya sistem operasi ini

dikarenakan sistem operasi ini banyak digunakan oleh para pengguna komputer di Indonesia.

2. Bahasa Pemrograman

   Bahasa pemrograman dalam perancangan perancangan aplikasi ini menggunakan PHP, sedangkan program yang digunakan untuk merancang aplikasi ini adalah Macromedia Dreamweaver 8.

3. Database

   Media penyimpanan data dalam aplikasi ini menggunakan MySql.

4. Web Server

   Web server yang digunakan dalam aplikasi ini adalah XAMPP 1.7.0.

5. Web Browser

   Web browser yang digunakan dalam aplikasi ini adalah Google Chrome.

5.1.3. Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi antara lain :

1. Spesifikasi hardware untuk admin.

   a. Processor : Intel Pentium IV atau yang sekelasnya.

   b. Memori (RAM) : minimal 512 MB.

   c. Monitor : Resolusi 1024 X 768 pixel.

   d. Hard Disk : minimal 80 GB.

   e. *Mouse, Keyboard* dan *Monitor* sebagai peralatan antar muka.

   f. Modem internet.

2. Spesifikasi hardware untuk pelanggan dan pengunjung.

a. Processor : Intel Pentium IV atau yang sekelasnya.

b. Memori (RAM) : minimal 256 MB.

c. Monitor : Resolusi 1024 X 768 pixel.

d. Hard Disk : minimal 40 GB.

e. *Mouse, Keyboard* dan *Monitor* sebagai peralatan antar muka.

f. Modem internet.

5.1.4. Implementasi Basis Data

Pembuatan basis data dilakukan dengan menggunakan bahasa SQL, dimana DBMS yang digunakan adalah MySQL. Implemantasi basis datanya dalam bahasa SQL adalah sebagai berikut :

1. Tabel Admin

```
CREATE TABLE IF NOT EXISTS
`admin` (
`username` varchar(10) NOT NULL DEFAULT '',
`pass` varchar(41) NOT NULL,
`nama` varchar(20) NOT NULL,
`level` enum('admin','superadmin') NOT NULL DEFAULT
'admin',
`status_hapus` enum('Y','T') NOT NULL DEFAULT
'T', PRIMARY KEY (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. Tabel Bank

```
CREATE TABLE IF NOT EXISTS
`bank` (
`id_bank` varchar(10) NOT NULL,
`nama_bank` varchar(20) DEFAULT NULL,
`no_rek` varchar(20) DEFAULT NULL,
`atas_nama` varchar(30) DEFAULT NULL,
`cabang` varchar(30) DEFAULT NULL,
`gambar` varchar(100) DEFAULT NULL,
`direktori` varchar(100) DEFAULT NULL, PRIMARY KEY
(`id_bank`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Tabel Barang

```
CREATE TABLE IF NOT EXISTS `barang` (
`kode_barang` varchar(20) NOT NULL,
```

```
`nama_barang` varchar(50) NOT NULL,
`deskripsi` text NOT NULL,
`tggl` date NOT NULL,
`id_kategori` int(11) NOT NULL,
`username` varchar(10) DEFAULT NULL,
`reting` int(11) NOT NULL,
`harga` int(11) NOT NULL,
`diskon` int(11) NOT NULL, PRIMARY KEY (`kode_barang`),
KEY `id_kategori` (`id_kategori`),
KEY `username` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER
TABLE `barang`
ADD CONSTRAINT `barang_ibfk_1` FOREIGN KEY (`id_kategori`)
REFERENCES `kategori` (`id_kategori`),
ADD CONSTRAINT `barang_ibfk_2` FOREIGN KEY
(`username`) REFERENCES `admin` (`username`);
```

4. Tabel Biaya Kirim

```
CREATE TABLE IF NOT EXISTS
`biaya_kirim` (
`id_biayakirim` int(11) NOT NULL AUTO_INCREMENT,
`biayakirim` int(11) NOT NULL,
`lama_kirim` int(2) NOT NULL,
`id_jeniskirim` int(3) DEFAULT NULL,
`id_kota` int(11) NOT NULL,
`username` varchar(10)DEFAULT NULL, PRIMARY KEY
(`id_biayakirim`),
KEY `id_kota` (`id_kota`), KEY `username` (`username`),
KEY `id_jeniskirim` (`id_jeniskirim`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ; ALTER TABLE `biaya_kirim`
ADD CONSTRAINT `biaya_kirim_ibfk_1` FOREIGN KEY
(`id_jeniskirim`) REFERENCES `jenis_kirim`
(`id_jeniskirim`), ADD CONSTRAINT `biaya_kirim_ibfk_2`
FOREIGN KEY (`username`) REFERENCES `admin`
(`username`);
```

5. Tabel Detail Pemesanan

```
CREATE TABLE IF NOT EXISTS `detailpemesanan` (
`id_pesandetail` int(11) NOT NULL AUTO_INCREMENT,
`id_pesanan` varchar(20) DEFAULT NULL,
`kode_barang` varchar(20) DEFAULT NULL,
`qty` int(11) DEFAULT NULL,
`berat` float DEFAULT NULL,
`diskon` float DEFAULT NULL,
`harga` int(11) DEFAULT NULL,
`ukuran` varchar(10) DEFAULT NULL,
`id_detailbarang` int(11) DEFAULT NULL, PRIMARY KEY
(`id_pesandetail`),
KEY `id_detailbarang` (`id_detailbarang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
ALTER TABLE `detailpemesanan`
ADD CONSTRAINT `detailpemesanan_ibfk_1` FOREIGN KEY
```

```
(`id_detailbarang`) REFERENCES `detail_barang`
(`id_detailbarang`);
```

6. Tabel Detail Barang

```
CREATE TABLE IF NOT EXISTS `detail_barang` (
`id_detailbarang` int(11) NOT NULL AUTO_INCREMENT,
`ukuran` varchar(15) NOT NULL,
`berat` float NOT NULL,
`harga` int(9) NOT NULL,
`diskon` float NOT NULL,
`stok` int(5) NOT NULL,
`total_pemesanan` int(5) NOT NULL,
`kode_barang` varchar(11) NOT NULL,
`username` varchar(10) DEFAULT NULL PRIMARY KEY
(`id_detailbarang`),
KEY `kode_barang` (`kode_barang`), KEY `username`
(`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
ALTER TABLE
`detail_barang`
ADD CONSTRAINT `detail_barang_ibfk_2` FOREIGN KEY
(`username`) REFERENCES `admin` (`username`),
ADD CONSTRAINT `detail_barang_ibfk_1` FOREIGN
KEY (`kode_barang`) REFERENCES `barang`
(`kode_barang`);
```

7. Tabel Gambar

```
CREATE TABLE IF NOT EXISTS `gambar` (
`no_gbr` int(11) NOT NULL AUTO_INCREMENT,
`nama_gbr` varchar(225) NOT NULL,
`kode_barang` varchar(11) NOT NULL,
`username` varchar(10) DEFAULT NULL,
`keterangan` text NOT NULL, PRIMARY KEY (`no_gbr`),KEY
 `kode_barang` (`kode_barang`), KEY `username` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
ALTER TABLE `gambar`
ADD CONSTRAINT `gambar_ibfk_1` FOREIGN KEY
(`kode_barang`) REFERENCES `barang`
(`kode_barang`),
ADD CONSTRAINT `gambar_ibfk_2` FOREIGN KEY
(`username`) REFERENCES `admin` (`username`);
```

8. Tabel Jenis Kirim

```
CREATE TABLE IF NOT EXISTS `jenis_kirim` (
`id_jeniskirim` int(3) NOT NULL AUTO_INCREMENT,
`nama_jeniskirim` varchar(30) NOT NULL,
`id_perusahaankirim` int(11) NOT NULL,
`username` varchar(10)
DEFAULT NULL, PRIMARY KEY
(`id_jeniskirim`),
KEY `id_perusahaankirim` (`id_perusahaankirim`),
KEY `username` (`username`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ; ALTER TABLE `jenis_kirim`
ADD CONSTRAINT `jenis_kirim_ibfk_1` FOREIGN
KEY (`id_perusahaankirim`) REFERENCES
`perusahaan_kirim` (`id_perusahaankirim`),
ADD CONSTRAINT `jenis_kirim_ibfk_2` FOREIGN KEY
(`username`) REFERENCES `admin` (`username`);
```

9. Tabel Kategori

```
CREATE TABLE IF NOT EXISTS `kategori` (
`id_kategori` int(11) NOT NULL AUTO_INCREMENT,
`nama_kategori` varchar(30) NOT NULL,
`username` varchar(10) DEFAULT NULL, PRIMARY KEY
(`id_kategori`), KEY `username` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ; ALTER TABLE `kategori`
ADD CONSTRAINT `kategori_ibfk_1` FOREIGN KEY (`username`)
REFERENCES `admin` (`username`);
```

10. Tabel Kota

```
CREATE TABLE IF NOT EXISTS `kota` (
`id_kota` int(11) NOT NULL AUTO_INCREMENT,
`nama_kota` varchar(30) NOT NULL,
`id_prov` int(11) NOT NULL,
`username` varchar(10) DEFAULT NULL, PRIMARY KEY (`id_kota`),
KEY `username` (`username`), KEY `FK_id_prov` (`id_prov`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
ALTER TABLE `kota` ADD CONSTRAINT `kota_ibfk_1` FOREIGN KEY
(`id_prov`) REFERENCES `provinsi` (`id_prov`),
ADD CONSTRAINT `kota_ibfk_2` FOREIGN KEY (`username`)
REFERENCES `admin` (`username`);
```

11. Tabel Pelanggan

```
CREATE TABLE IF NOT EXISTS `pelanggan` (
`id_pelanggan` int(11) NOT NULL AUTO_INCREMENT,
`email` varchar(30) NOT NULL,
`pass` varchar(41) NOT NULL,
`ndepan` varchar(25) NOT NULL,
`nbelakang` varchar(25) DEFAULT NULL,
`jk` enum('L','P') NOT NULL,
`alamat` text NOT NULL,
`kd_pos` int(7) NOT NULL,
`tlp` varchar(20) DEFAULT NULL,
`tanya_aman` varchar(40) NOT NULL,
`jawab_aman` varchar(30) NOT NULL,
`id_kota` int(11) NOT NULL,
`status_hapus` enum('Y','T') NOT NULL DEFAULT 'T',
`is_active` int(1) NOT NULL, PRIMARY KEY(`id_pelanggan`), KEY
`id_kota` (`id_kota`) ) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ; ALTER TABLE `pelanggan`
ADD CONSTRAINT `pelanggan_ibfk_1` FOREIGN KEY (`id_kota`)
```

```
REFERENCES `kota` (`id_kota`);
```

12. Tabel Pembayaran

```
CREATE TABLE IF NOT EXISTS `pembayaran` (
`id_bayar` int(11) NOT NULL AUTO_INCREMENT,
`nama_pengirim` varchar(30) NOT NULL,
`totalbayar` float NOT NULL,
`tipe_bayar` varchar(10) NOT NULL,
`bank` varchar(10) NOT NULL,
`tggl_bayar` date NOT NULL,
`no_rek` varchar(21) NOT NULL,
`status_bayar` enum('dibayar','lunas','ditunda') NOT NULL,
`kode_verifikasi` varchar(30) NOT NULL,
`keterangan` varchar(30) DEFAULT NULL,
`id_pesanan` int(11) NOT NULL,
`username` varchar(11) DEFAULT NULL,
PRIMARY KEY (`id_bayar`),
KEY `username` (`username`),
KEY `id_pesanan` (`id_pesanan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ; ALTER TABLE `pembayaran`
ADD CONSTRAINT `pembayaran_ibfk_2` FOREIGN KEY (`username`)
REFERENCES `admin` (`username`),
ADD CONSTRAINT `pembayaran_ibfk_3` FOREIGN KEY
(`id_pesanan`) REFERENCES `pemesanan` (`id_pesanan`);
```

13. Tabel Pemesanan

```
CREATE TABLE IF NOT EXISTS
`pemesanan` (
`id_pesanan` int(11) NOT NULL AUTO_INCREMENT,
`ongkos_kirim` int(11) NOT NULL,
`total_berat` float NOT NULL,
`total_qty` int(11) NOT NULL,
`total_kirim` int(11) NOT NULL,
`total_harga` int(11) NOT NULL,
`total_grand` int(11) NOT NULL,
`tgl_pesan` datetime DEFAULT NULL,
`tgl_expired` datetime DEFAULT NULL,
`id_pelanggan` int(11) DEFAULT NULL,
`status_pesan` int(1) NOT NULL,
`status_kirim` varchar(20) NOT NULL, PRIMARY KEY
(`id_pesanan`),
KEY `id_pelanggan` (`id_pelanggan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
ALTER TABLE `pemesanan`
ADD CONSTRAINT `pemesanan_ibfk_1` FOREIGN KEY
(`id_pelanggan`) REFERENCES `pelanggan`(`id_pelanggan`);
```

14. Tabel Pemesanan Konfirmasi

```
CREATE TABLE IF NOT EXISTS `pemesanan_konfirmasi` (
`id_konfirmasi` int(11) NOT NULL AUTO_INCREMENT,
```

```
`id_bank` varchar(10) DEFAULT NULL,
`id_pesanan` int(11) DEFAULT NULL,
`bank_tujuan` varchar(20) DEFAULT NULL,
`norek_asal` varchar(20) DEFAULT NULL,
`atas_nama` varchar(30) DEFAULT NULL,
`tgl_konfirmasi` datetime DEFAULT NULL,
`total_bayar` int(11) DEFAULT NULL, PRIMARY KEY
(`id_konfirmasi`),
KEY `id_bank` (`id_bank`),
KEY `id_pesanan` (`id_pesanan`)) ENGINE=InnoDB DEFAULT
CHARSET=latin1 AUTO_INCREMENT=1 ; ALTER TABLE
`pemesanan_konfirmasi`
ADD CONSTRAINT `pemesanan_konfirmasi_ibfk_1`FOREIGN KEY
(`id_bank`) REFERENCES `bank` (`id_bank`),
ADD CONSTRAINT `pemesanan_konfirmasi_ibfk_2` FOREIGN KEY
(`id_pesanan`) REFERENCES `pemesanan` (`id_pesanan`);
```

15. Tabel Pengiriman

```
CREATE TABLE IF NOT EXISTS `pengiriman` (
`id_pengiriman` int(11) NOT NULL AUTO_INCREMENT,
`id_pesanan` int(11) DEFAULT NULL,
`nama` varchar(50) DEFAULT NULL,
`alamat` text,
`id_kota` int(11) DEFAULT NULL,
`tlp` varchar(12) DEFAULT NULL,
`kodepos` varchar(5) DEFAULT NULL,
`nama_perusahaankirim` varchar(20) DEFAULT NULL,
`nama_jeniskirim` varchar(20) DEFAULT NULL,
`no_resi` varchar(20) DEFAULT NULL,
`tgl_kirim` datetime DEFAULT NULL,
PRIMARY KEY (`id_pengiriman`),
KEY `id_pesanan` (`id_pesanan`),
KEY `id_kota` (`id_kota`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ; ALTER TABLE `pengiriman`
ADD CONSTRAINT `pengiriman_ibfk_1` FOREIGN KEY
(`id_kota`) REFERENCES `kota` (`id_kota`),
ADD CONSTRAINT `pengiriman_ibfk_2` FOREIGN KEY (`id_pesanan`)
REFERENCES `pemesanan` (`id_pesanan`);
```

16. Tabel Perusahaan Kirim

```
CREATE TABLE IF NOT EXISTS `perusahaan_kirim` (
`id_perusahaankirim` int(11) NOT NULL AUTO_INCREMENT,
`nama_perusahaankirim` varchar(20) NOT NULL,
`website` varchar(30) NOT NULL,
`username` varchar(10) DEFAULT NULL,
PRIMARY KEY (`id_perusahaankirim`), KEY `username`
(`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
ALTER TABLE `perusahaan_kirim`
ADD CONSTRAINT `perusahaan_kirim_ibfk_1` FOREIGN KEY
(`username`) REFERENCES `admin` (`username`);
```

17. Tabel Provinsi

```
CREATE TABLE IF NOT EXISTS `provinsi` (
`id_prov` int(11) NOT NULL AUTO_INCREMENT,
`nama_prov` varchar(21) NOT NULL,
`username` varchar(10) DEFAULT
NULL, PRIMARY KEY (`id_prov`),
KEY `username` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ; ALTER TABLE `provinsi`
ADD CONSTRAINT `provinsi_ibfk_1` FOREIGN KEY (`username`)
REFERENCES `admin` (`username`);
```

18. Tabel Temp

```
CREATE TABLE IF NOT EXISTS `temp` (
`id_temp` int(11) NOT NULL AUTO_INCREMENT,
`id_session` varchar(50) DEFAULT NULL,
`kode_barang` varchar(20) DEFAULT NULL,
`nama_barang` varchar(25) DEFAULT NULL,
`qty` int(11) DEFAULT NULL,
`harga` int(11) DEFAULT NULL,
`ukuran` varchar(10) DEFAULT NULL,
`id_detailbarang` int(11) DEFAULT NULL, PRIMARY KEY
(`id_temp`),
KEY `kode_barang` (`kode_barang`),
KEY `id_detailbarang` (`id_detailbarang`) ) ENGINE=InnoDB
DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ; ALTER TABLE `temp`
ADD CONSTRAINT `temp_ibfk_1` FOREIGN KEY (`kode_barang`)
REFERENCES `barang` (`kode_barang`),
ADD CONSTRAINT `temp_ibfk_2` FOREIGN KEY (`id_detailbarang`)
REFERENCES `detail_barang` (`id_detailbarang`);
```

19. Tabel Ukuran

```
CREATE TABLE IF NOT EXISTS `ukuran` (
`id_ukuran` int(11) NOT NULL AUTO_INCREMENT,
`ukuran` varchar(10) DEFAULT NULL,
`username` varchar(20) DEFAULT NULL,
PRIMARY KEY (`id_ukuran`), KEY `username` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
ALTER TABLE `ukuran`
ADD CONSTRAINT `ukuran_ibfk_1` FOREIGN KEY (`username`)
REFERENCES `admin` (`username`);
```

20. Tabel Next Number

```
CREATE TABLE IF NOT EXISTS `next_number` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`run_num` int(4) unsigned zerofill DEFAULT NULL,
`ket` varchar(10) DEFAULT NULL,
`code` char(2) DEFAULT NULL,
PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

```
data for table `next_number`
INSERT INTO `next_number` (`id`, `run_num`, `ket`, `code`)
VALUES (1, 0004, 'Barang', 'KB');
```

21. Tabel Setting

```
CREATE TABLE IF NOT EXISTS `setting` (
`id_setting` int(11) NOT NULL AUTO_INCREMENT,
`link` varchar(225) DEFAULT NULL,
PRIMARY KEY (`id_setting`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
-- -- Dumping data for table `setting`--
INSERT INTO `setting` (`id_setting`, `link`) VALUES
(1, 'http://localhost/moveon/admin/');
```

22. Tabel Jenis

```
CREATE TABLE IF NOT EXISTS `jenis` (
`id_jenis` int(11) NOT NULL AUTO_INCREMENT,
`id_kategori` int(11) NOT NULL,
`jenis` varchar(30) NOT NULL,
PRIMARY KEY (`id_jenis`))
ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

5.1.5. Implementasi Antar Muka

Implementasi antar muka dilakukan dengan sebuah *webpage* yang berektensi PHP. Setiap halaman dalam perangkat lunak dibuat *file program* yang ditulis dengan berektensi *.php*. Berikut ini adalah implementasi dari setiap halaman yang dibuat.

5.1.5.1.Implementasi Antar Muka Admin

Tabel 5.1 Implementasi antar muka admin

Menu	Deskripsi	Nama File
Login Admin	Digunakan sebagai halaman login Admin	Admin/index.php
Home	Digunakan sebagai tampilan utama halaman Admin	Admin/beranda.php
Pengolahan Data Pelanggan	Digunakan untuk melakukan pengolahan data Pelanggan	Admin/member.php
Pengolahan Data Kategori	Digunakan untuk melakukan pengolahan data Kategori	Admin/kategori.php

Pengolahan Data Ukuran	Digunakan untuk melakukan pengolahan data ukuran	Admin/ukuran.php
Pengolahan Data Barang	Digunakan untuk melakukan pengolahan data barang	Admin/produk.php
Pengolahan Data Provinsi	Digunakan untuk melakukan pengolahan data provinsi	Admin/provinsi.php
Pengolahan Data Kota	Digunakan untuk melakukan pengolahan data kota	Admin/kota.php
Pengolahan Jasa Pengiriman	Digunakan untuk pengolahan data jasa pengiriman	Admin/kurir.php
Pengolahan Biaya Kirim	Digunakan untuk pengolahan data biaya kirim	Admin/ongkos.php
Pengolahan Pesanan Baru	Digunakan untuk pengolahan data pesanan baru	Admin/transaksi.php
Pengolahan Pesanan Konfirmasi	Digunakan untuk pengolahan data pesanan konfirmasi	Admin/konfirmasi.php
Pengolahan Pesanan Bayar	Digunakan untuk Pengolahan data pesanan bayar	Admin/bayar.php
Pengolahan Pesanan Kirim	Digunakan untuk pengolahan data pesanan kirim	Admin/kirim.php
Pengolahan Laporan Pesanan Baru	Digunakan untuk pembuatan laporan pesanan baru	Admin/pesananbaru.php
Pengolahan Laporan Pesanan Konfirmasi	Digunakan untuk pembuatan laporan pesanan konfirmasi	Admin/pesanankonfirmasi.php
Pengolahan Laporan Pesanan Bayar	Digunakan untuk pembuatan laporan pesanan bayar	Admin/pesananbayar.php
Backup database	Digunakan untuk membackup semua database	Admin/backup.php
Backup file	Digunakan untuk membackup semua file	Admin/backupFile.php
Jenis	Digunakan untuk melakukan pengolahan data jenis	Admin/jenis.php

5.1.5.2.Implementasi Antar Muka Pelanggan

Tabel 5.2 Implementasi antar muka pelanggan

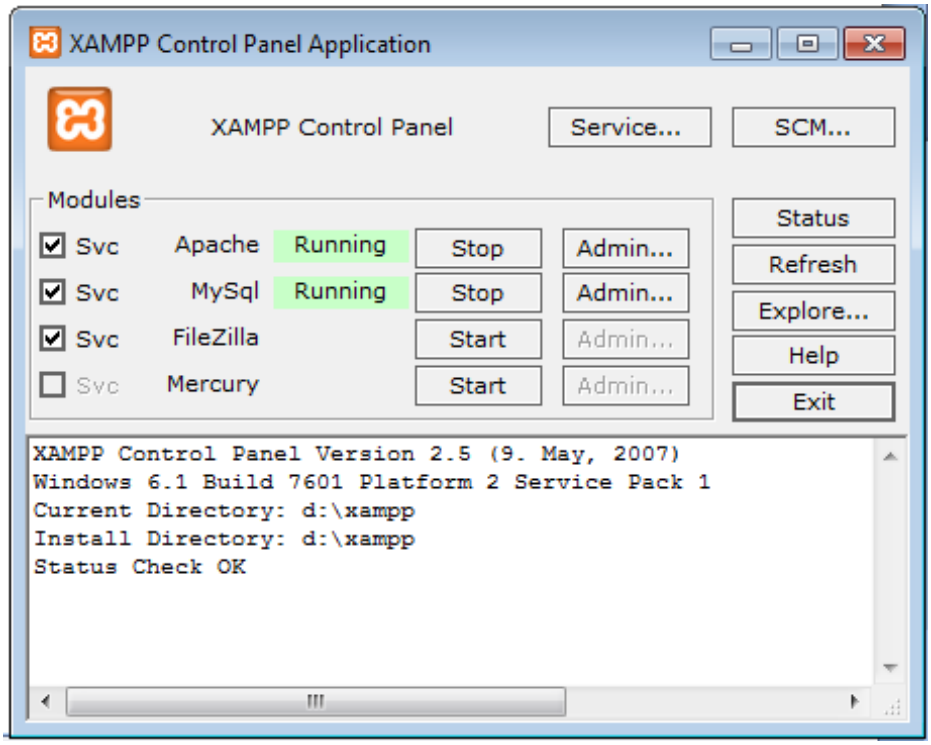
Menu	Deskripsi	Nama File
Login Pelanggan	Digunakan untuk *login* pelanggan	login.php
Daftar Pelanggan	Digunakan untuk melakukan pendaftaran	Register.php
Home	Digunakan untuk menampilkan halaman utama pelanggan	Home.php
Detail Barang	Digunakan untuk melihat detail barang	detailproduk.php
Keranjang Belanja	Digunakan untuk melihat merubah, menghapus daftar pesanan yang akan dilakukan	cart.php
Cara Pemesanan	Digunakan untuk melihat cara melakukan pemesanan	cpemensanan.php
Cara Pembayaran	Digunakan untuk melihat cara melakukan pembayaran	cpembayaran.php
Cara Pengiriman	Digunakan untuk melihat cara melakukan pengiriman	cpengiriman.php
Data Pengiriman	Digunakan untuk melihat, merubah data pengiriman yang akan dilakukan	pengiriman.php
Layanan Pengiriman	Digunakan untuk melihat, merubah layanan pengiriman yang akan dilakukan	kurir.php
Konfirmasi Pesanan	Digunakan untuk melihat konfirmasi pesanan yang telah dilakukan	confirm.php
Riwayat Belanja	Digunakan untuk melihat riwayat belanja yang telah dilakukan	histori.php
Riwayat Belanja setelah Konfirmasi	Digunakan untuk melihat riwayat belanja yang telah dikonfirmasi	histori.php

Riwayat Belanja setelah Barang Diterima	Digunakan untuk melihat riwayat belanja yang telah dilakukan setelah barang diterima	histori.php

5.1.6. Implementasi Instalasi Program

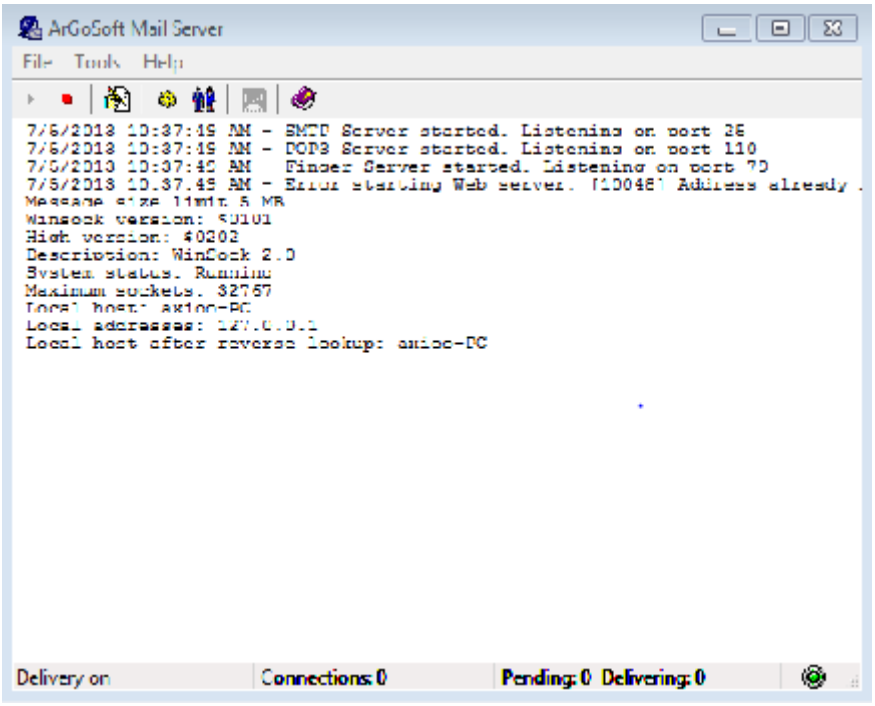
Berikut ini adalah langkah-langkah implementasi instalasi program untuk menggunakan program sistem informasi *e-commerce* pada Move On Event Organizer :

1. Aktifkan XAMPP Control Panel Aplication



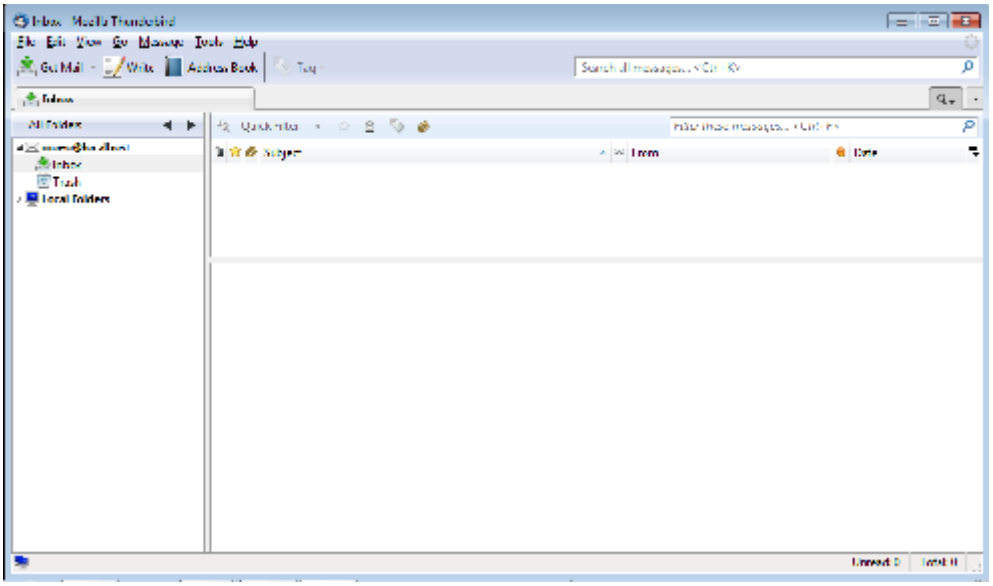
Gambar 5.1 Tampilan XAMPP

2. Aktifkan ArgoSoft Mail Server



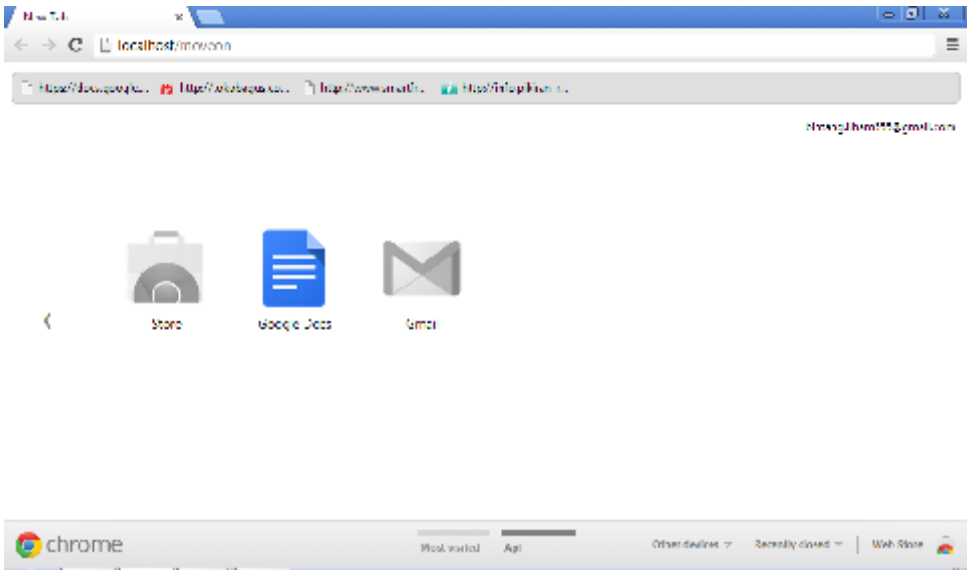
Gambar 5.2 Tampilan ArgoSoft

3. Aktifkan Mozilla Thunderbird



Gambar 5.3 Tampilan Mozilla Thunderbird

4. Buka aplikasi *browser*, lalu ketikan alamat *website*.



Gambar 5.4 Tampilan Browser Google Chrome

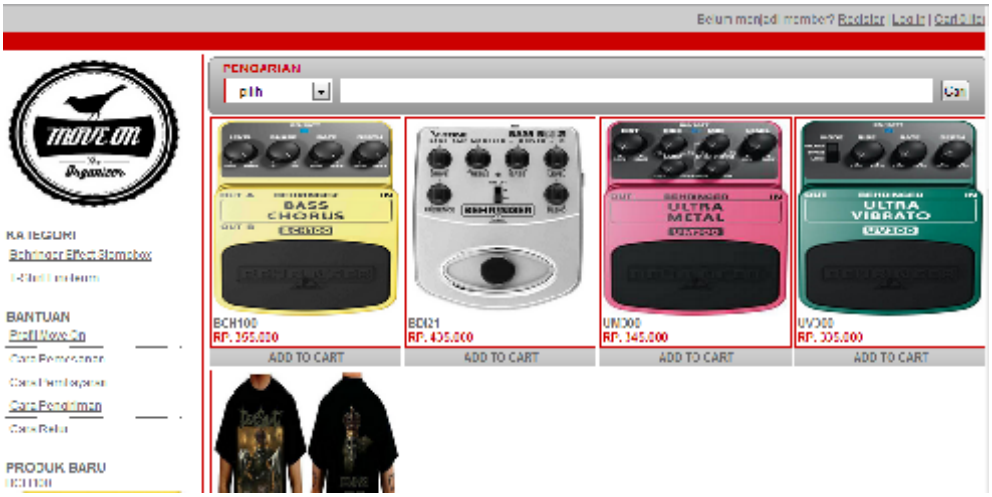
5.1.7. Penggunaan Program

Penggunaan program bertujuan untuk menjelaskan tahapan-tahapan yang dapat dilakukan dalam menggunakan program sistem informasi *e-commerce* pada Move On Event Organizer. Berikut ini akan dijelaskan cara penggunaan program beserta tampilannya :

5.1.7.1. Penggunaan Halaman Pengunjung Dan Pelanggan

1. Halaman Utama Website

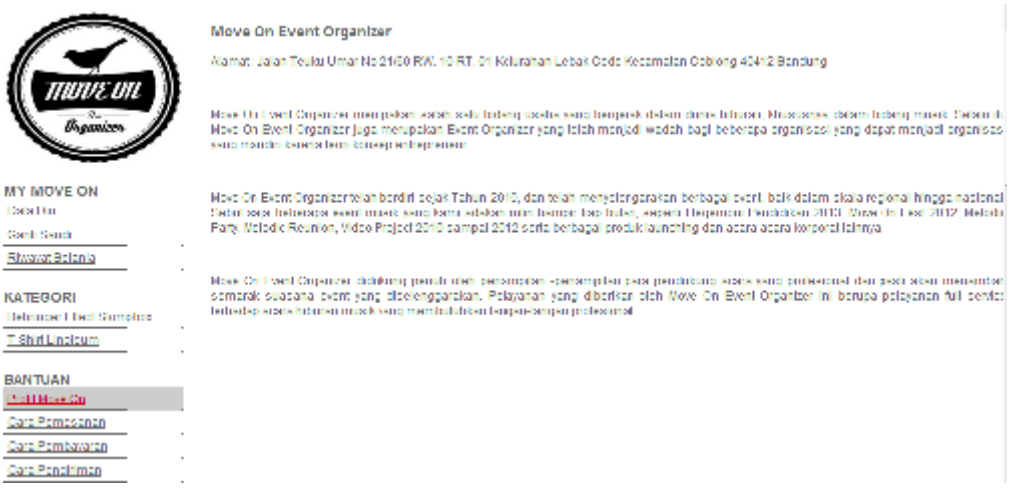
Ketika pertama kali masuk kedalam aplikasi, maka muncul tampilan halaman utama dari *website*.



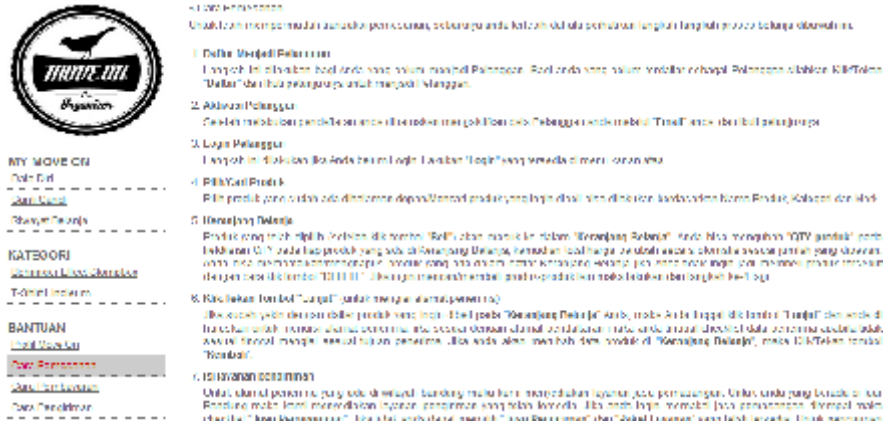
Gambar 5.5 Tampilan Halaman Utama

2. Halaman Bantuan

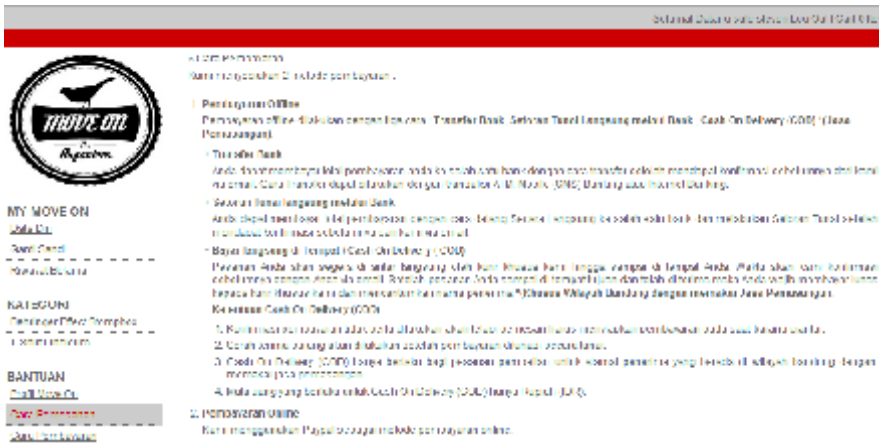
Halaman bantuan merupakan halaman informasi untuk pengunjung yang terdiri dari profil Move On, cara pemesanan, cara pembayaran, cara pengiriman.



Gambar 5.6 Tampilan Halaman Profil



Gambar 5.7 Tampilan Halaman Cara Pemesanan



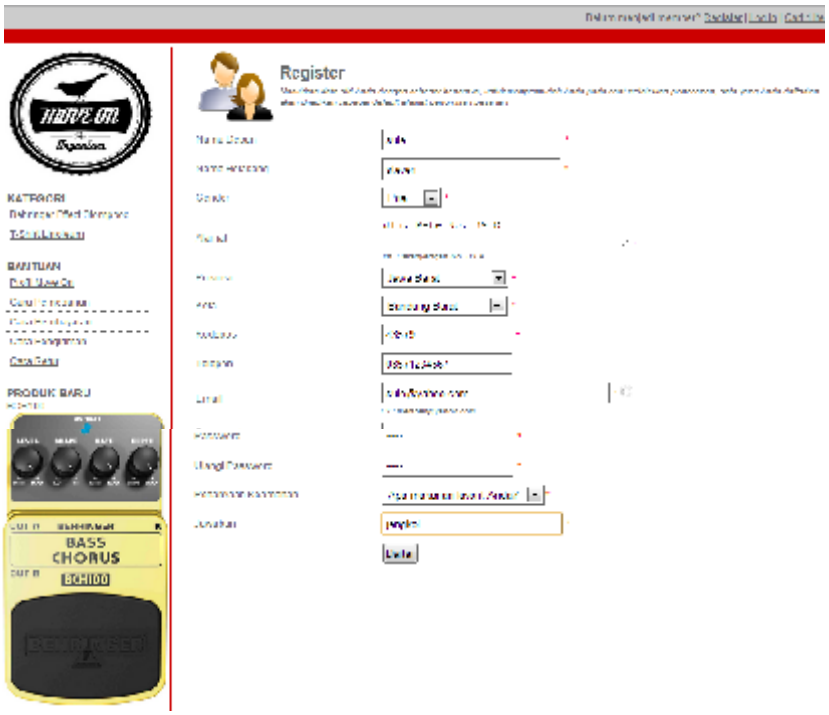
Gambar 5.8 Tampilan Halaman Cara Pembayaran



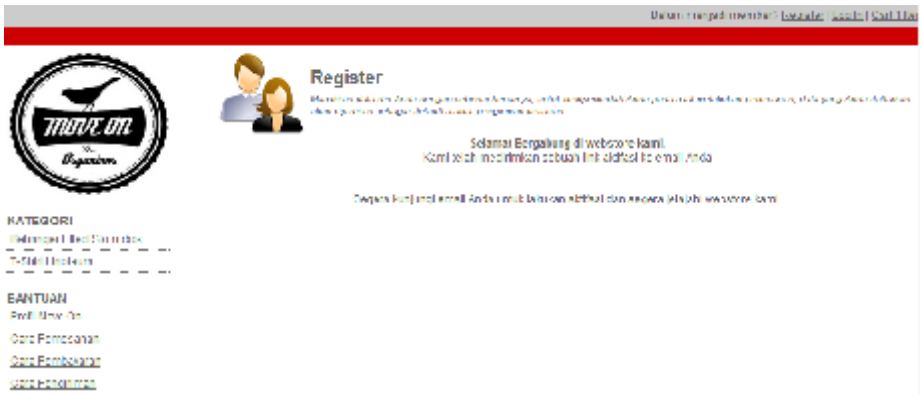
Gambar 5.9 Tampilan Halaman Cara Pengiriman

3. Halaman Daftar Menjadi Pelanggan

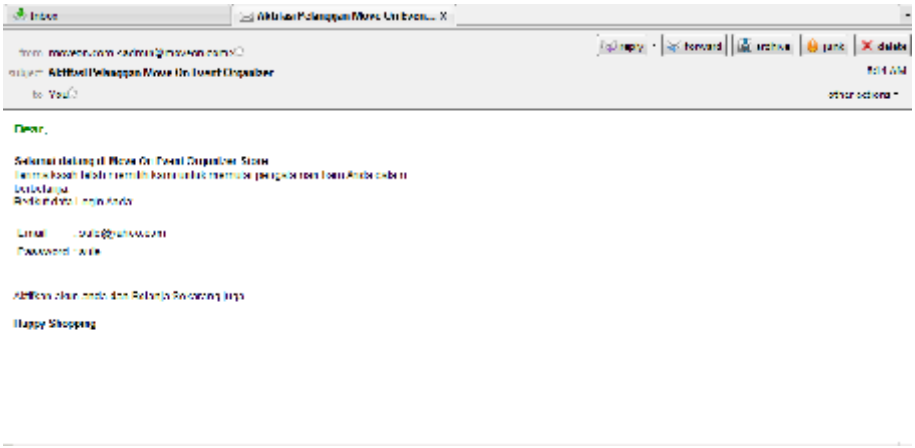
Untuk dapat menjadi pelanggan/*member*, maka pengunjung harus melakukan pendaftaran terlebih dahulu. Pertama mengisi form pendaftaran, lalu membuka *email* untuk melakukan aktivasi akun pelanggan.



Gambar 5.10 Tampilan Halaman Pendaftaran



Gambar 5.11 Tampilan Halaman Pemberitahuan Pendaftaran



Gambar 5.12 Tampilan Halaman Aktivasi Akun Pada Email



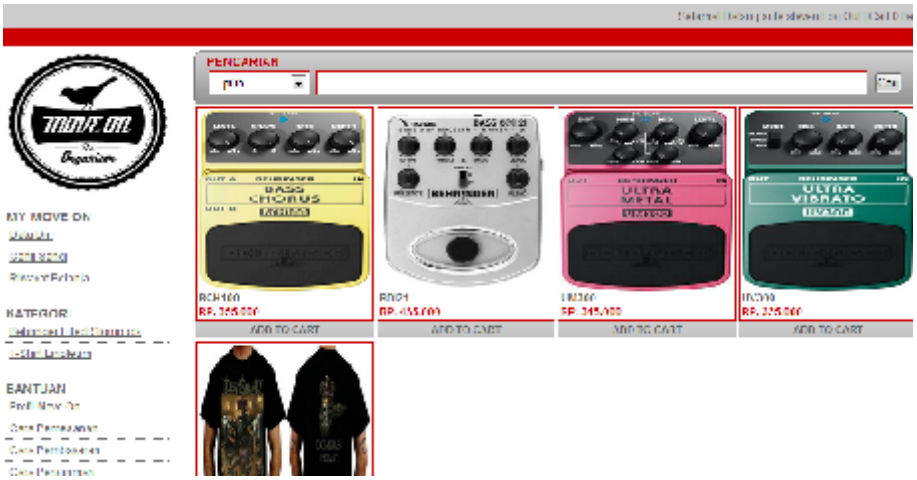
Gambar 5.13 Tampilan Halaman Aktivasi Akun Aktif

4. Halaman Login Pelanggan

Untuk dapat melakukan transaksi pemesanan, pengunjung harus melakukan *login* terlebih dahulu.



Gambar 5.14 Tampilan Halaman Login



Gambar 5.15 Tampilan Halaman Berhasil Login

5. Halaman Pemesanan Barang

Pelanggan melakukan pemesanan barang dengan memilih barang, lalu memilih ukuran serta jumlah barang yang akan dipesan.



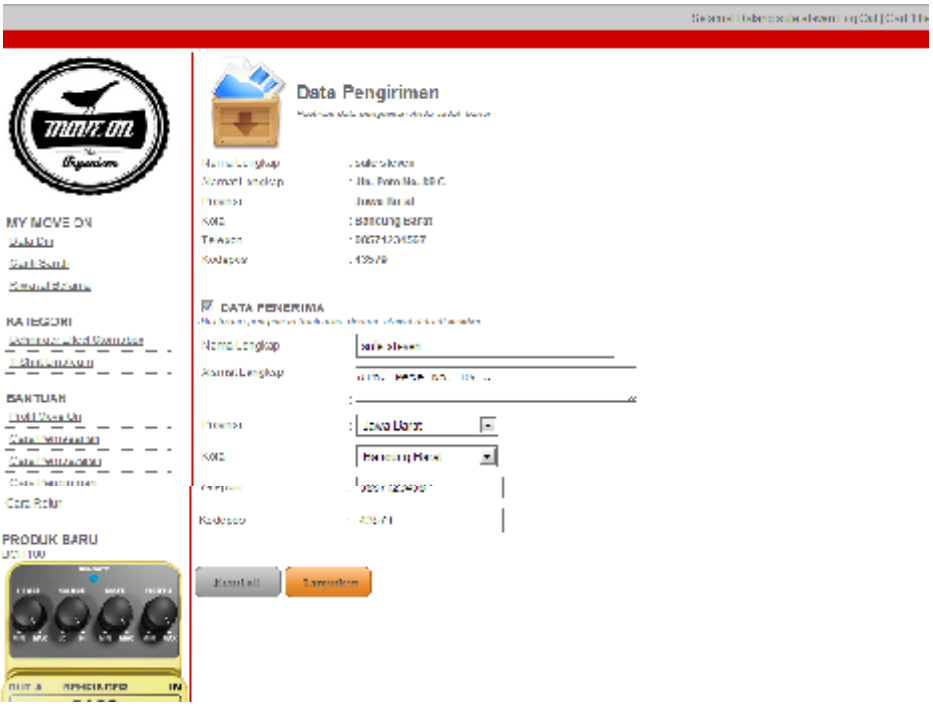
Gambar 5.16 Tampilan Halaman Pemesanan Barang



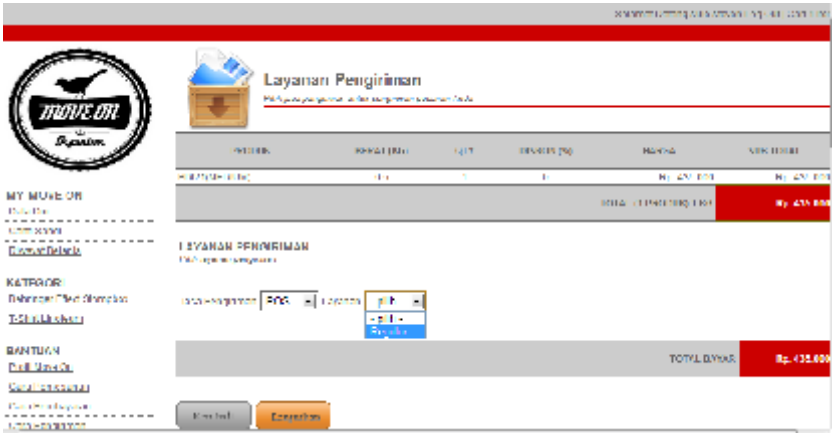
Gambar 5.17 Tampilan Halaman Keranjang Belanja Setelah Memilih Barang Pesanan

6. Halaman Transaksi Pengiriman

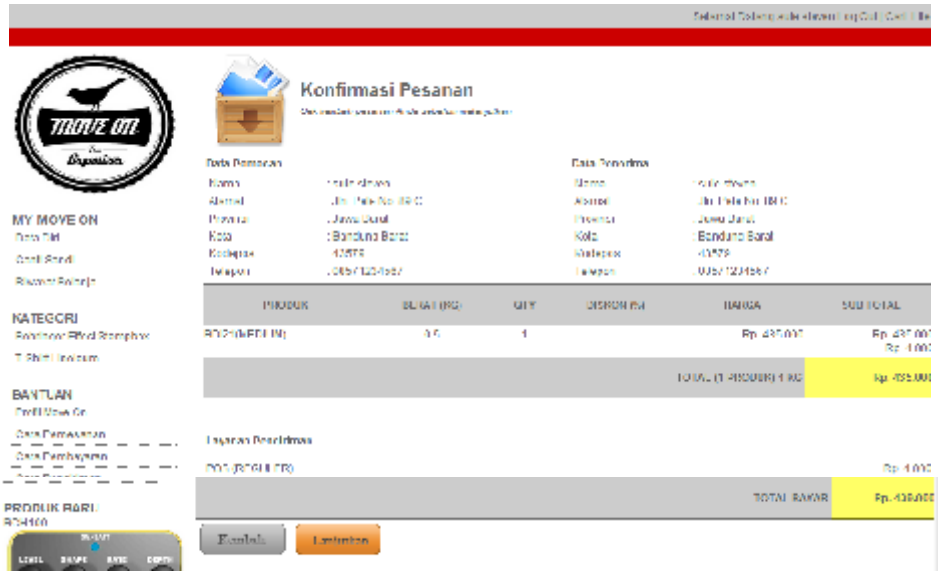
Pada halaman ini. pelanggan memasukkan data penerima setelah itu melakukan pemilihan layana pengiriman. Pelanggan akan dapat melihat detail pesanan dan juga dikirimkan detail pesanan melalui *email* untuk selanjutnya dikonfirmasi kembali.



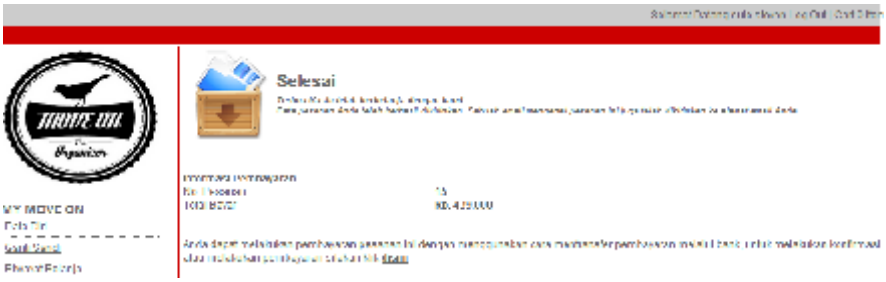
Gambar 5.18 Tampilan Halaman Transaksi Pengiriman



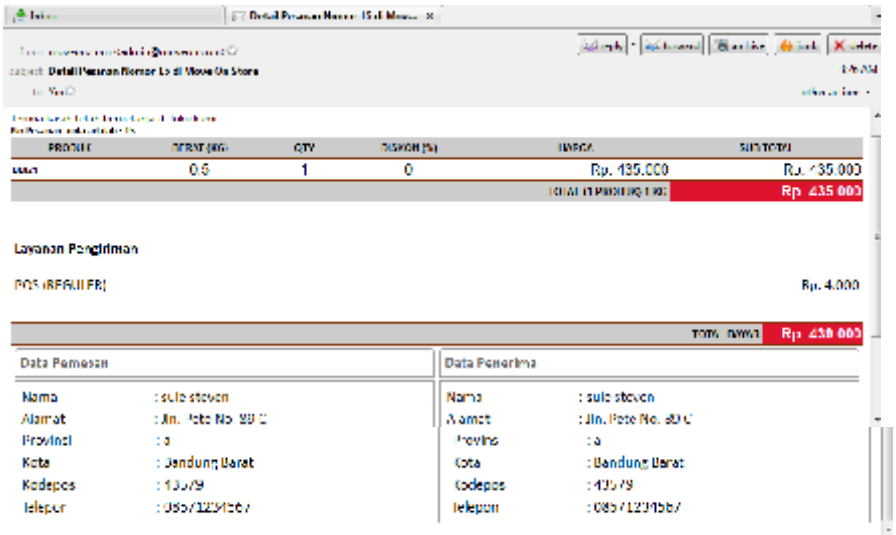
Gambar 5.19 Tampilan Halaman Pemilihan Layanan Pengiriman



Gambar 5.20 Tampilan Halaman Konfirmasi Pemesanan



Gambar 5.21 Tampilan Halaman Informasi Pemesanan



Gambar 5.22 Tampilan Halaman Detail Pesanan Pada Email

7. Halaman Konfirmasi Transfer

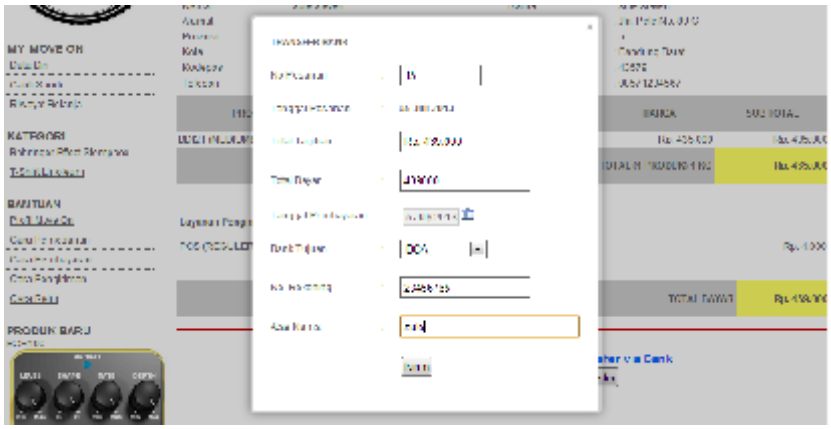
Pelanggan memilih riwayat belanja, lalu melakukan konfirmasi bahwa telah melakukan transaksi pembayaran melalui bank.



Gambar 5.23 Tampilan Halaman Riwayat Belanja



Gambar 5.24 Tampilan Halaman Detail Pesanan Untuk Transfer Via Bank

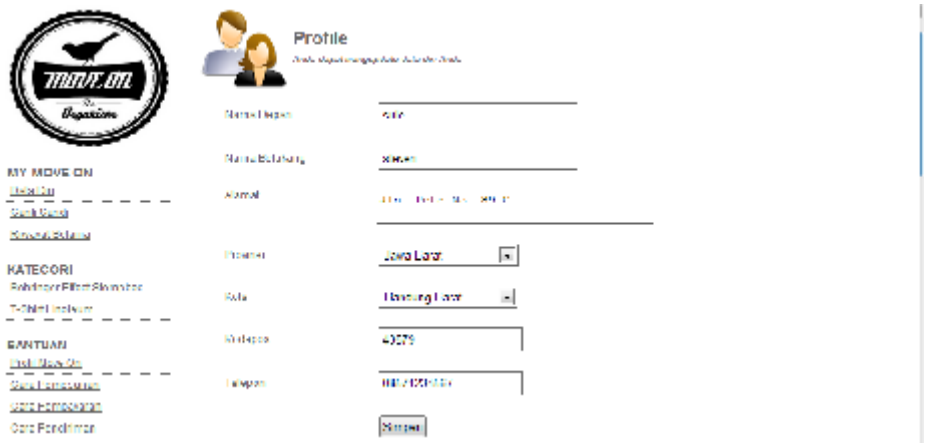


Gambar 5.25 Tampilan Halaman Konfirmasi Transfer Via Bank

8. **Halaman Ubah Password Dan Ubah Profile**



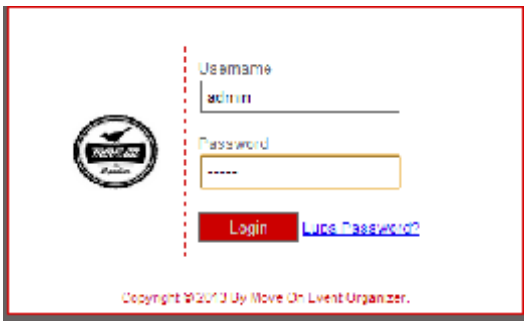
Gambar 5.26 Tampilan Halaman Ubah Password



Gambar 5.27 Tampilan Halaman Ubah Profile

5.1.7.2. Penggunaan Halaman Admin

1. **Halaman Login Admin**



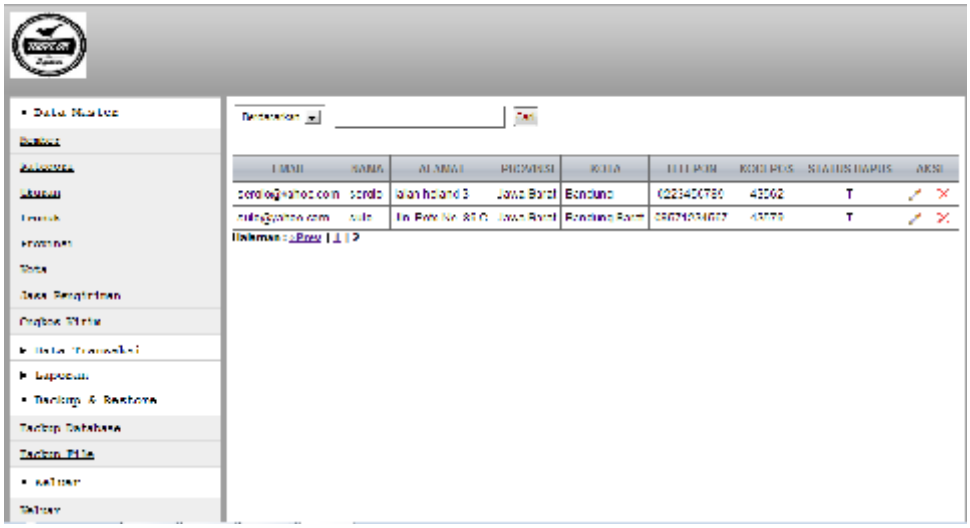
Gambar 5.28 Tampilan Halaman Login Admin



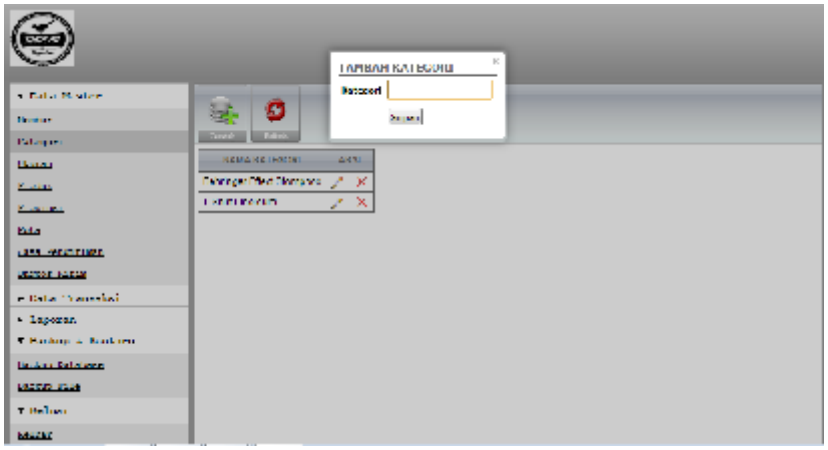
Gambar 5.29 Tampilan Halaman Home Admin

2. Halaman Data Master

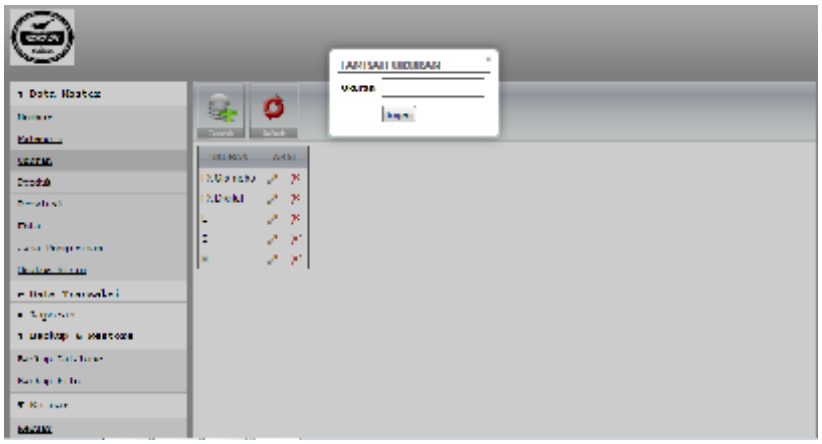
Halaman data master terdiri dari 8 sub-kategori, yaitu member, kategori, ukuran, produk, provinsi, kota, jasa pengiriman, dan ongkos kirim. Pada data master ini, admin dapat memasukan semua data berdasarkan sub-kategorinya, kecuali di data master member karena data master member hanya dimasukan datanya oleh member itu sendiri. Semua data master dapat dihapus, dinonaktifkan dan diedit.



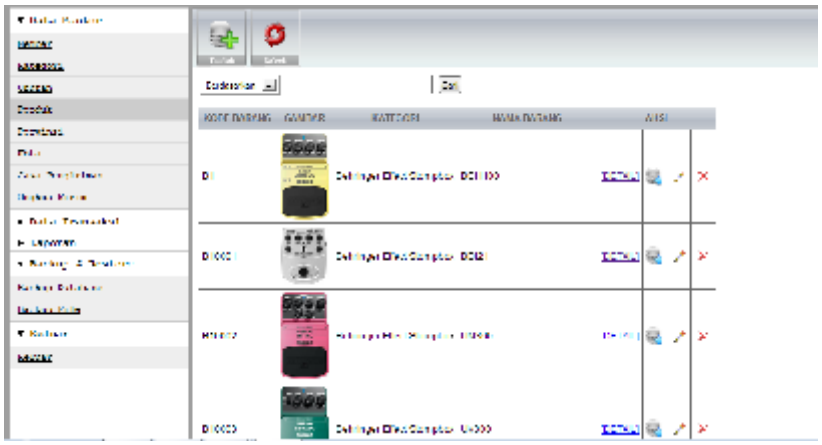
Gambar 5.30 Tampilan Halaman Data Master Member



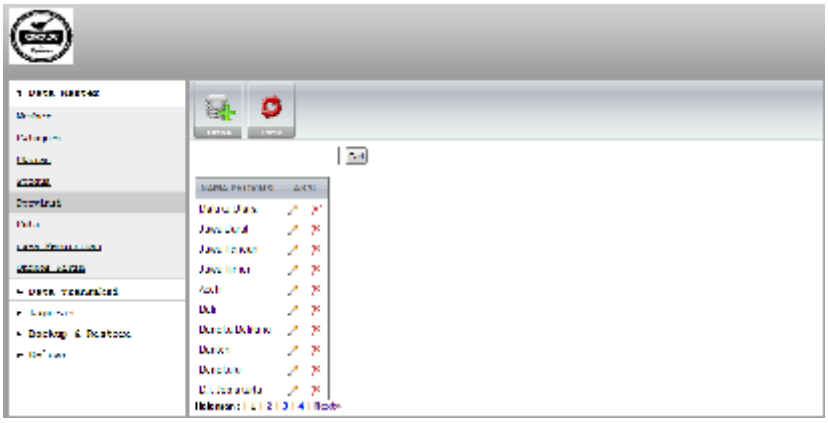
Gambar 5.31 Tampilan Halaman Data Master Kategori



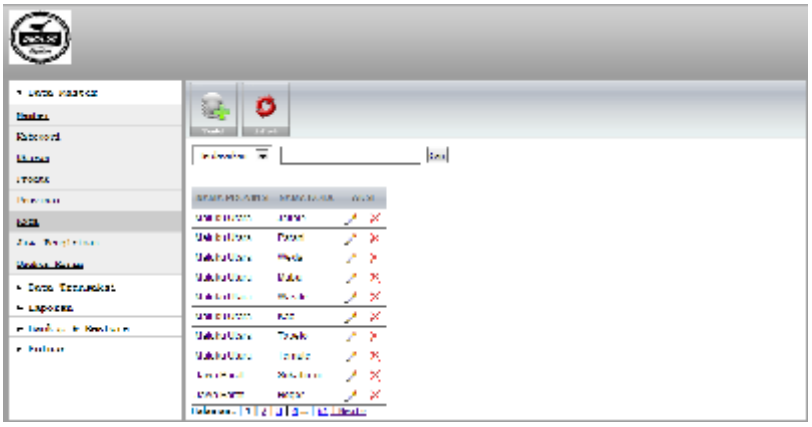
Gambar 5.32 Tampilan Halaman Data Master Ukuran



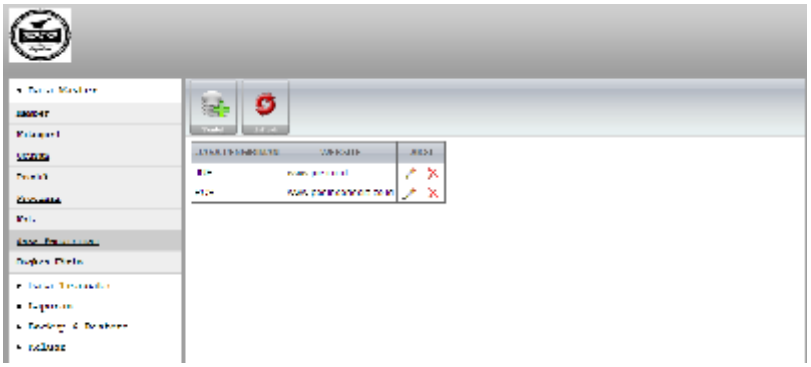
Gambar 5.33 Tampilan Halaman Data Master Barang



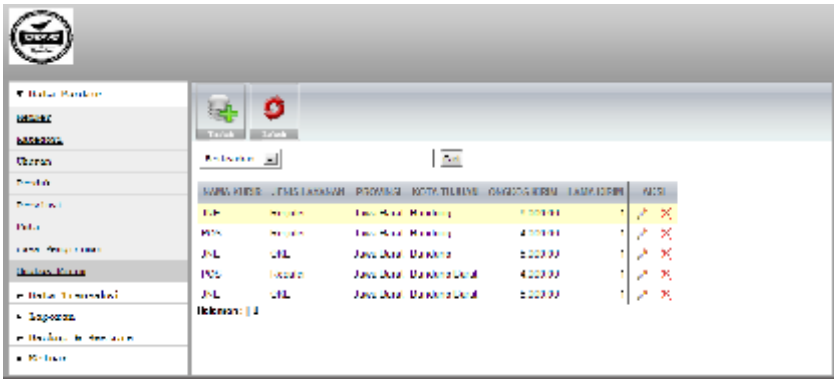
Gambar 5.34 Tampilan Halaman Data Master Provinsi



Gambar 5.35 Tampilan Halaman Data Master Kota



Gambar 5.36 Tampilan Halaman Data Master Jasa Pengiriman

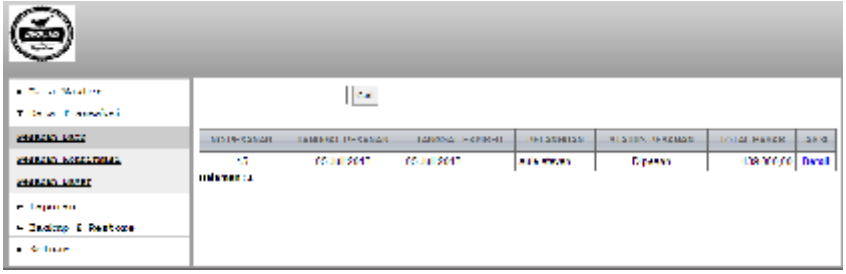


Gambar 5.37 Tampilan Halaman Data Master Ongkos Kirim

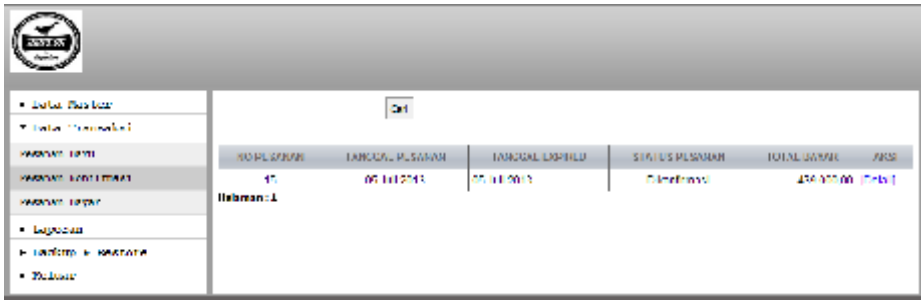
3. Halaman Data Transaksi

Halaman data transaksi terdiri dari 3 bagian, yaitu bagian pesanan baru, pesanan konfirmasi, dan pesanan bayar. Pesanan baru merupakan data transaksi pesanan yang baru dilakukan oleh pelanggan. Pesanan konfirmasi merupakan data transaksi yang telah dikonfirmasi oleh pelanggan, sehingga secara otomatis data pesanan di pesanan baru akan terhapus. Sedangkan pesanan bayar merupakan data transaksi yang telah dikonfirmasi pembayarannya oleh admin.

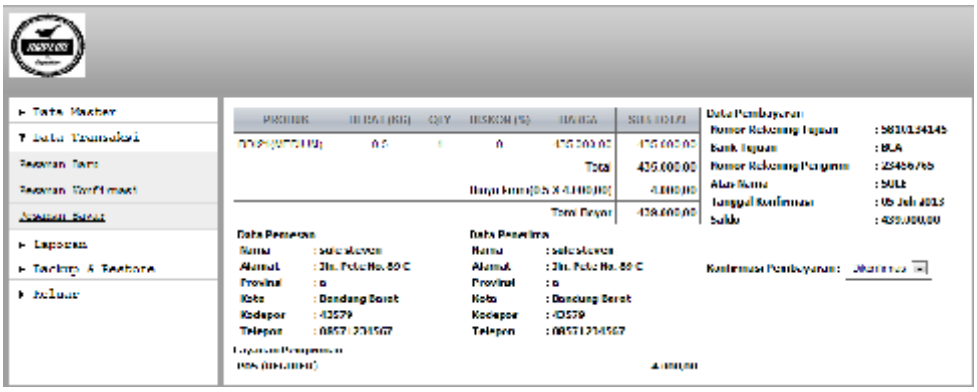
Dalam pesanan bayar, admin telah mengkonfirmasi pesanan pelanggan yang telah melakukan *transfer*. Maka dalam pesanan bayar akan ada 3 *button* aksi, yaitu cetak, kirim, dan detail. Tombol cetak untuk mencetak faktur, tombol kirim untuk membuat nomor resi dan mengirim bukti transaksi, sedangkan tombol detail berfungsi untuk melihat detail transaksi.



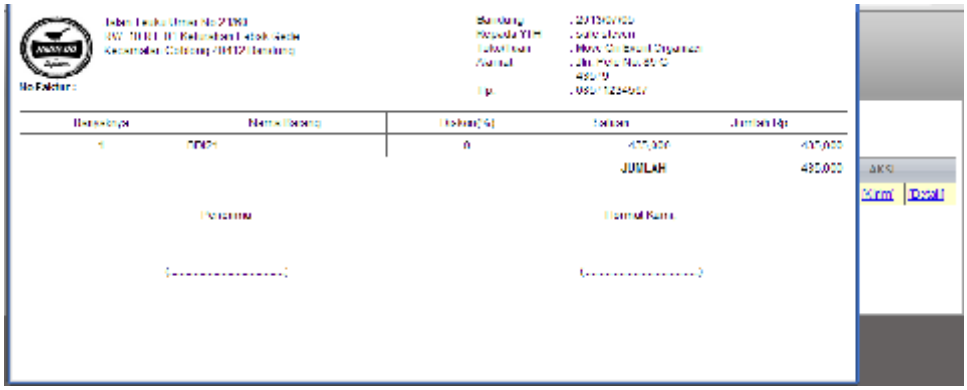
Gambar 5.38 Tampilan Halaman Data Transaksi Pesanan Baru



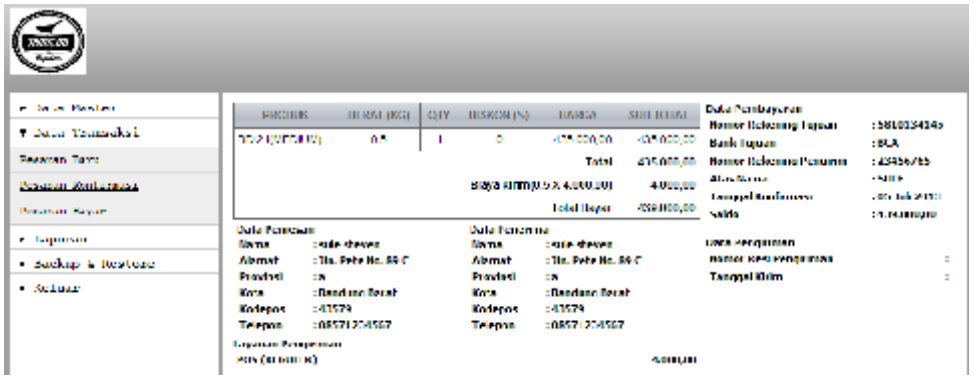
Gambar 5.39 Tampilan Halaman Data Transaksi Pesanan Konfirmasi



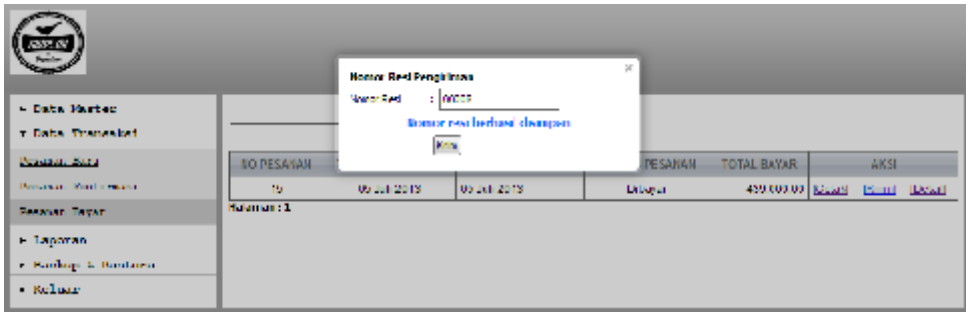
Gambar 5.40 Tampilan Halaman Data Transaksi Detail Pesanan Konfirmasi



Gambar 5.41 Tampilan Halaman Data Transaksi Pesanan Bayar Cetak



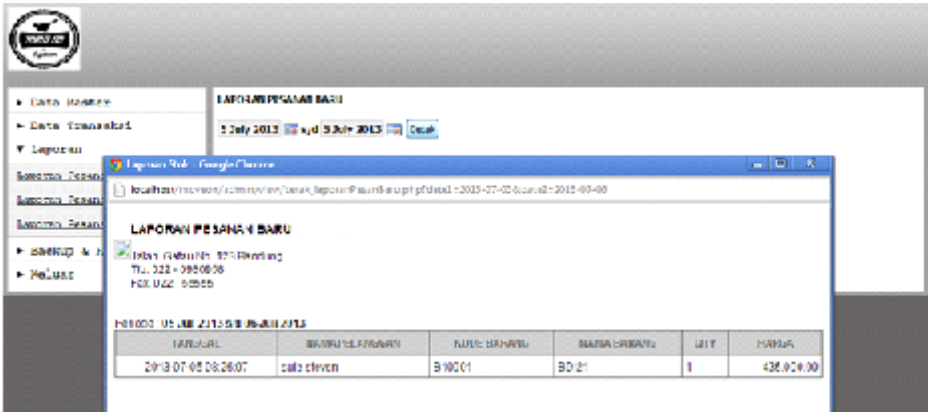
Gambar 5.42 Tampilan Halaman Data Transaksi Pesanan Bayar Detail



Gambar 5.43 Tampilan Halaman Data Transaksi Pesanan Bayar Kirim

4. Halaman Laporan

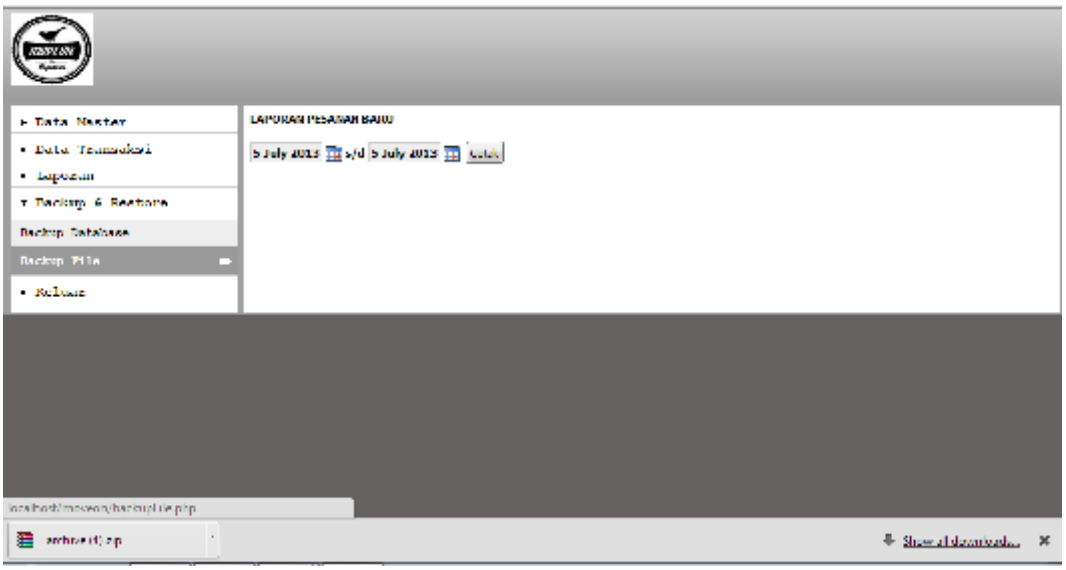
Halaman laporan merupakan halaman yang dapat mencetak laporan berdasarkan tanggal dan dapat langsung dicetak atau disimpan.



Gambar 5.44 Tampilan Halaman Laporan

5. Halaman Backup Dan Restore

Backup dan *restore* merupakan halaman yang dipergunakan untuk mem-*backup* seluruh data dan me-*restore* semua program.



Gambar 5.45 Tampilan Halaman Backup Dan Restore

5.2. Pengujian

Pengujian merupakan salah satu bagian terpenting dalam siklus pengembangan perangkat lunak. Metode pengujian dilakukan untuk menjamin kualitas, untuk menguji apakah perangkat lunak tersebut bebas dari kesalahan serta untuk mengetahui kelemahan dari perangkat lunak tersebut. Pengujian perangkat lunak ini menggunakan metode pengujian *black box.* Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak yang dibuat.

5.2.1. Rencana Pengujian

Pengujian perangkat lunak sistem informasi *e-commerce* ini menggunakan data uji berupa data input dari pengguna perangkat lunak yang telah dibuat.

Berikut adalah rencana pengujian pada sistem informasi *e-commerce* tersebut yang akan dilakukan :

Tabel 5.3 Rencana pengujian halaman admin

Item Uji	Detail Pengujian
Login	Isi data *login*
	Verifikasi *e-mail* dan *password*
Data Kategori	Menampilkan data kategori
	Menambah data kategori
	Mengedit data kategori
	Menghapus data kategori
Data Barang	Menampilkan data barang
	Menambah data barang
	Mengedit data barang
	Menghapus data barang
	Masuk ke proses data detail barang
	Masuk ke proses data gambar
Data Jasa Pengiriman	Menampilkan data pengiriman
	Mencetak label
	Mengupdate status pengiriman

Tabel 5.4 Rencana pengujian halaman pengunjung

Item Uji	Detail Pengujian
Daftar	Menampilkan *form* daftar pelanggan
	Isi *form* daftar pelanggan
	Isi *form* email pelanggan
	Simpan data pelanggan baru

Tabel 5.5 Rencana pengujian halaman pelanggan

Item Uji	Detail Pengujian
Login	Isi data *login*
	Verifikasi *e-mail* dan *password*
Lupa Password	Menampilkan *form* lupa *password*
	Mengirim informasi password ke email pelanggan
Data Diri	Menampilkan *form* data diri
	Ubah data profile
	Simpan perubahan
Ganti Sandi	Menampilkan form ganti password
	Isi form ganti password
	Simpan data baru

Keranjang belanja	Menampilkan data pesanan
	Mengubah data pesanan
	Menghapus data pesanan
	Menampilkan konfirmasi pesanan
Data pengiriman	Menampilkan data pemesan
	Mengubah data penerima
	Menyimpan data penerima baru
Layanan pengiriman	Menampilkan pesanan
	Memilih layanan pengiriman
	Menyimpan data pengiriman
Konfirmasi pesanan	Menampilkan konfirmasi pesanan
Riwayat belanja	Menampilkan riwayat belanja pelanggan
	Mengubah konfirmasi pesanan
Pembayaran	Menampilkan data pesanan
	Memilih sistem pembayaran
	Menampilkan form pembayaran transfer bank
	Isi form tranfer bank
	Menyimpan data pembayaran

5.2.2. Kasus dan Hasil Pengujian

Kasus dan hasil pengujian merupakan tindak lanjut dari rencana pengujian yang telah direncanakan. Pada bagian ini kita dapat mengetahui kekurangan dan kelebihan dari perangkat lunak yang telah dibuat. Berikut merupakan kasus dan hasil pengujian dari *web e-commerce* Move On Event Organizer :

1. **Pengujian Login Admin**

Tabel 5.6 Pengujian login admin

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Username : admin *Password* : admin	Bisa masuk ke halaman admin	Berhasil masuk ke halaman admin	Diterima
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Username : <kosong> *Password* : <kosong>	Tidak dapat *login*	Tidak dapat masuk ke halaman admin. Menampilkan pesan ”*username* tidak boleh kosong”	Diterima
Username : admin *Password* : <kosong>	Tidak dapat *login*	Tidak dapat masuk ke halaman admin. Menampilkan pesan “*password* tidak boleh kosong”	Diterima
Username : coba *Password* : admin	Tidak dapat *login*	Tidak dapat masuk ke halaman admin. Menampilkan pesan “*password* tidak boleh kosong”	Diterima

2. Pengujian Data Kategori

Tabel 5.7 Pengujian data kategori (tambah dan edit)

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Kategori : T-shirt	Data berhasil disimpan ke database	Data berhasil disimpan	Diterima
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Kategori : <kosong>	Data tidak tersimpan	Data tidak tersimpan. Menampilkan pesan “Nama Kategori belum diisi”	Diterima
Kategori : Kemeja	Data tidak tersimpan	Data tidak tersimpan. Menampilkan pesan “Nama Kategori sudah terdaftar”	Diterima

3. Pengujian Data Barang

Tabel 5.8 Pengujian data barang (tambah dan edit)

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Kategori : T-shirt Nama Barang : Lino T-shirt Blue Harga : 50000 Diskon : 10% Deskripsi : abc	Data berhasil Disimpan ke database	Sistem menyimpan data barang ke dalam database	Diterima
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Kategori : <kosong> Nama Barang : <kosong> Harga : <kosong> Diskon : <kosong> Deskripsi : kosong>	Data tidak tersimpan	Sistem menampilkan pesan "Pilih Kategori"	Diterima
Kategori : T-shirt Nama Barang : <kosong> Harga : <kosong> Diskon : <kosong> Deskripsi : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan "Nama Barang belum diisi"	Diterima
Kategori : T-shirt Nama Barang : Lino T-shirt Harga : <kosong> Diskon : <kosong> Deskripsi : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan "Harga belum diisi"	Diterima
Kategori : T-shirt Nama Barang : Lino T-shirt Harga : 50000 Diskon : <kosong> Deskripsi : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan "Diskon tidak valid"	Diterima
Kategori : T-shirt Nama Barang : Lino T-shirt Red Harga : 50000 Diskon : 10% Deskripsi : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan "Barang sudah terdaftar"	Diterima

4. Pengujian Data Detail Barang

Tabel 5.9 Pengujian data detail barang (tambah dan edit)

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpul
Ukuran : XL Berat : 0.5 Stok : 12	Data berhasil disimpan ke database	Sistem menyimpan data detail barang ke dalam database	Diterima
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpul
Ukuran : <kosong> Berat : <kosong> Stok : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan "Pilih Ukuran"	Diterima
Ukuran : XL Berat : <kosong> Stok : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan "Berat belum diisi"	Diterima
Ukuran : XL Berat : 0.5 Stok : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan "Stok belum diisi"	Diterima

5. Pengujian Data Jasa Pengiriman

Tabel 5.10 Pengujian data jasa pengiriman

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Jasa Pegiriman : JNE Website : www.jne.co.id	Data berhasil disimpan ke database	Sistem menyimpan data jasa pengiriman ke dalam database dan menampilkan pesan "Tersimpan"	Diterima
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Jasa Pegiriman : <kosong> Website : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan"Jasa Pengiriman belum diisi"	Diterima

Jasa Pegiriman : JNE Website : <kosong>	Data tidak tersimpan	Sistem menampilkan pesan “Tidak tersimpan”	Diterima

6. Pengujian Daftar Pengunjung

Tabel 5.11 Pengujian daftar pengunjung

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama Depan : Ilham Nama Belakang : Bintang Gender : Pria Alamat : Jl. Pasteur No. 35 Provinsi : Jawa Barat Kota : Bandung Kodepos : 40162 Telepon : 085720319519 Email : bintang.ilham@gmail.com Password : ilham Ulangi Password : ilham Pertanyaan Keamanan : Apa makananan favorit anda? Jawaban : Jengkol	Data berhasil disimpan ke database	Form pendaftaran terisi dan benar semua field, maka akan dapat melanjutkan proses daftar dan ada konfirmasi email ke email pelanggan	Diterima
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Nama Depan : <kosong> Nama Belakang : <kosong> Gender : <kosong> Alamat : <kosong> Provinsi : <kosong> Kota : <kosong> Kodepos : <kosong> Telepon : <kosong> Email : <kosong> Password : <kosong> Ulangi Password : <kosong> Pertanyaan Keamanan : <kosong> Jawaban : <kosong>	Data tidak tersimpan	Form pendaftaran tidak terisi dan tidak benar pada semua field, maka akan menampilkan pesan "Nama masih kosong, Pilih gender, Alamat masih kosong, Provinsi belum dipilih, Kota belum dipilih, Kodepos tidak valid, Email belum diisi, Password masih kosong, Pilih pertanyaan keamanan, jawaban keamanan masih kosong"	Diterima

7. Pengujian Login Pelanggan

Tabel 5.12 Pengujian login pelanggan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Email : bintang.ilham@gmail.com Password : ilham	Menuju *home* pelanggan	Dapat menuju ke *home* pelanggan	Diterima
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Email : <kosong> Password : <kosong>	Menampilkan pesan kesalahan, "Masukan Email"	Dapat menampilkan pesan kesalahan "Masukan Email"	Diterima
Email : bintang.ilham@gmail.com Password : <kosong>	Menampilkan pesan kesalahan, "Email tidak valid"	Dapat menampilkan pesan kesalahan "Email tidak valid"	Diterima

Email : bintang.ilham@gmail.com Password : <kosong>	Menampilkan pesan kesalahan, "Masukan password"	Dapat menampilkan pesan kesalahan "Masukan password"	Diterima

8. Pengujian Lupa Password Pelanggan

Tabel 5.13 Pengujian lupa password pelanggan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukkan email yang benar dan menekan tombol Kirim. Contoh masukan: Email : bintang.ilham@gmail.com	Sistem akan mengirimkan data pelanggan ke email pelanggan	Data pelanggan Terkirim ke email pelanggan	Diterima
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukkan email yang salah dan menekan tombol Kirim. Contoh masukan: Email :	Menampilkan pesan kesalahan, "Masukan Email"	Dapat menampilkan pesan kesalahan "Masukan Email"	Diterima

9. Pengujian Data Diri Pelanggan

Tabel 5.14 Pengujian data diri pelanggan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengubah data pelanggan Contoh : Nama Depan : Ilham Nama Belakang : Bintang Alamat : Jln Pasteur No. 35 Provinsi : Jawa Barat Kota : Bandung Kodepos : 40162 Telepon : 085720319519	Sistem dapat menyimpan data pelanggan yang sudah diubah dan memunculkan pesan "Profil berhasil diupdate"	Data pelanggan yang sudah diubah disimpan ke dalam database dan memunculkan pesan "Profil berhasil diupdate"	Diterima

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengubah data pelanggan Contoh : Nama Depan : <kosong> Nama Belakang : <kosong> Alamat : <kosong> Provinsi : <kosong> Kota : <kosong> Kodepos : <kosong> Telepon : <kosong>	Jika semua *field* tidak terisi dengan benar dalam pengisian, maka akan menampilkan pesan "Nama Depan masih kosong"	Form profile tidak terisi dan tidak benar pada semua field, maka akan menampilkan pesan "Nama Depan masih kosong"	Diterima
Mengubah data pelanggan Contoh : Nama Depan : Ilham Nama Belakang : <kosong> Alamat : <kosong> Provinsi : <kosong> Kota : <kosong> Kodepos : <kosong> Telepon : <kosong>	Jika semua *field* tidak terisi dengan benar dalam pengisian, maka akan menampilkan pesan "Nama Belakang masih kosong"	Form profile tidak terisi dan tidak benar pada semua field, maka akan menampilkan pesan "Nama Belakang masih kosong"	Diterima

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan pengujian yang telah dilakukan, maka penulis mendapatkan hasil bahwa sistem informasi *e-commerce* ini telah berjalan dengan baik. Namun demikian pengujian tersebut dapat dikatakan belum sempurna, dikarenakan masih adanya batasan pengujian yang telah dilakukan. Dan semua pengujian yang dilakukan dalam pengujian ini diharapkan dapat mewakili pengujian fungsi yang lainnya dalam perancangan sistem informasi *e-commerce* pada Move On Event Organizer.