BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

Tahap implementasi system merupakan tahap penerapan sistem yang telah didesain atau dirancang, sehingga system yang telah dibuat dapat dioperasikan dan digunakans ecara optimal dan sesuai dengan kebutuhan.

5.1 Implementasi

Implementasi sistem adalah tahap penerapan system yang akan dilakukan jika system disetujui termasuk program yang telah dibuat pada tahap perancangan system agar siap untuk dioperasikan. Implementasi system informasi penjualan dan pembelian di Toko Tas Yulia dilakukan menggunakan bahasa pemrograman PHP dengan basis data yang digunakan adalah MySQL. Aplikasi PHP tersebut dapat dijalankan pada berbagai *platform* sistem operasi dan perangkat keras, tetapi implementasi dan pengujian sepenuhnya hanya dilakukan pada perangkat keras PC (Personal Computer) dengan system operasi Microsoft Windows XP SP3.

5.1.1. Batasan Implementasi

Pengimpelementasian perangkat lunak pemantauan dan pengendalian aplikasi ini ada beberapa hal yang menjadin batasan impelementasi, yaitu :

1. Tidak semua kebutuhan sistem pengelolaan sisteminformasi *penjualan dan pembelian* dikerjakan, tetapi hanya subsistem-subsistem yang mendukung

yaitu penjualan, tambah, delete dan edit produk, sertalaporan penjualan dan pembelian.

2. Basis data yang digunakan dalam pengimplementasian ini adalah MySQL.
3. Tampilan untuk web based menggunakan bahasa Indonesia.

5.1.2. Implementasi Perangkat Lunak

Untuk pengembangan perangkat lunak ini digunakan Xampp, MySQL, PHP dan adobe dreamweaver CS5.Apache digunakan sebagai web server, MySQL sebagai database, PHP sebagai bahasa pemrograman dan Adobe Dreamweaver CS5 sebagai aplikasi editor.

5.1.3. Implementasi Perangkat Keras

Untuk menjalankan website yang dirancang maka dibutuhkan perangkat keras sebagai penunjangnya. Adapun perangkat keras yang dibutuhkan adalah :

1. Server
   a. Processor minimal Pentium IV dan sekelasnya.
   b. Menggunakan minimal RAM 1GB.
   c. Memory VGA Onboard atau Minimal 128MB
   d. Hardisk 80GB
   e. Keyboard, monitor dan mouse.
2. Client
   a. Processor minimal Pentium IV dan sekelasnya.
   b. Menggunakan minimal RAM 1GB.

c. Hardisk 80GB.

d. Memory VGA Onboard atau 128MB.

e. Keyboard, monitor dan mouse.

5.1.4. Implementasi Basis Data

DBMS (Data Base Management System) yang digunakan untuk mengimplementasikan aplikasi collaborative learning ini menggunakan Adobe Dreamweaver CS5 sebagai aplikasi pembangun dan MySQL. Berikut adalah implementasi basis data yang terdapat dalam system :

```
--
-- Table structure for table `admin`
--

CREATE TABLE IF NOT EXISTS `admin` (
  `id_admin` int(10) NOT NULL AUTO_INCREMENT,
  `user_admin` varchar(25) NOT NULL,
  `pass_admin` int(25) NOT NULL,
  `nama_admin` varchar(25) NOT NULL,
  `email_admin` varchar(30) NOT NULL,
  `tgl_daftar` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_admin`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;

--
-- Table structure for table `bank`
--

CREATE TABLE IF NOT EXISTS `bank` (
  `id_bank` int(10) NOT NULL AUTO_INCREMENT,
  `nama_bank` varchar(50) NOT NULL,
  `no_rekening` bigint(20) NOT NULL,
  PRIMARY KEY (`id_bank`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;
```

```
--
-- Table structure for table `bulan`
--

CREATE TABLE IF NOT EXISTS `bulan` (
  `bulan` text NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;


--
-- Table structure for table `carapesan`
--

CREATE TABLE IF NOT EXISTS `carapesan` (
  `id_cara` int(5) NOT NULL AUTO_INCREMENT,
  `judul_cara` varchar(300) NOT NULL,
  `isi_cara` varchar(1000) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_cara`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;


--
-- Table structure for table `cara_pembelian`
--

CREATE TABLE IF NOT EXISTS `cara_pembelian` (
  `id_carapembelian` int(10) NOT NULL AUTO_INCREMENT,
  `judul` varchar(150) NOT NULL,
  `isi` varchar(1000) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_carapembelian`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;


--
-- Table structure for table `detail_pesanan`
--

CREATE TABLE IF NOT EXISTS `detail_pesanan` (
  `id_dpesan` int(10) NOT NULL AUTO_INCREMENT,
  `id_pemesanan` int(10) NOT NULL,
  `id_produk` int(10) NOT NULL,
  `jumlah` int(2) NOT NULL,
```

```
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `id_member` int(15) NOT NULL,
  `nama_produk` varchar(50) NOT NULL,
  `harga` int(15) NOT NULL,
  PRIMARY KEY (`id_dpesan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;


--
-- Table structure for table `download_katalog`
--

CREATE TABLE IF NOT EXISTS `download_katalog` (
  `id_katalog` int(15) NOT NULL AUTO_INCREMENT,
  `judul_katalog` varchar(100) NOT NULL,
  `nama_file` varchar(100) NOT NULL,
  `info_katalog` varchar(500) NOT NULL,
  `tanggal` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_katalog`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;


--
-- Table structure for table `jenis_kelamin`
--

CREATE TABLE IF NOT EXISTS `jenis_kelamin` (
  `jenis_kelamin` varchar(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;


--
-- Table structure for table `kategori`
--

CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` int(10) NOT NULL AUTO_INCREMENT,
  `nama_kategori` varchar(30) NOT NULL,
  `waktu` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `kode_kategori` varchar(15) NOT NULL,
  `deskripsi` varchar(500) NOT NULL,
  `keyword` varchar(150) NOT NULL,
  PRIMARY KEY (`id_kategori`)
```

```
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;


--
-- Table structure for table `keranjang`
--

CREATE TABLE IF NOT EXISTS `keranjang` (
  `id_keranjang` int(5) NOT NULL AUTO_INCREMENT,
  `id_produk` int(10) NOT NULL,
  `jumlah` int(3) NOT NULL,
  `id_member` int(10) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `harga` int(10) NOT NULL,
  `total` int(10) NOT NULL,
  `nama_produk` varchar(50) NOT NULL,
  PRIMARY KEY (`id_keranjang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;


--
-- Table structure for table `kontak`
--

CREATE TABLE IF NOT EXISTS `kontak` (
  `id_kontak` int(15) NOT NULL AUTO_INCREMENT,
  `nama_kontak` varchar(30) NOT NULL,
  `email_kontak` varchar(50) NOT NULL,
  `pesan` varchar(250) NOT NULL,
  `tanggal` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `no_hp` bigint(15) NOT NULL,
  PRIMARY KEY (`id_kontak`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;


--
-- Table structure for table `kota`
--

CREATE TABLE IF NOT EXISTS `kota` (
  `id_kota` int(15) NOT NULL AUTO_INCREMENT,
  `nama_kota` varchar(100) NOT NULL,
  `kodepos` int(20) NOT NULL,
  `ongkos_kirim` int(10) NOT NULL,
```

```
  PRIMARY KEY (`id_kota`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;


--
-- Table structure for table `manufaktur`
--

CREATE TABLE IF NOT EXISTS `manufaktur` (
  `id_manufaktur` int(10) NOT NULL AUTO_INCREMENT,
  `nama_manufaktur` varchar(30) NOT NULL,
  `kode_manufaktur` varchar(10) NOT NULL,
  `tanggal` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_manufaktur`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=9 ;


--
-- Table structure for table `member`
--

CREATE TABLE IF NOT EXISTS `member` (
  `id_member` int(10) NOT NULL AUTO_INCREMENT,
  `user_member` varchar(25) NOT NULL,
  `pass_member` int(25) NOT NULL,
  `tgl_daftar` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `info` varchar(250) NOT NULL,
  `alamat` varchar(250) NOT NULL,
  `kota` varchar(50) NOT NULL,
  `kodepos` int(10) NOT NULL,
  `no_rekening` int(30) NOT NULL,
  `bank` varchar(50) NOT NULL,
  `telepon` bigint(15) NOT NULL,
  `id_kota` int(10) NOT NULL,
  `negara` varchar(20) NOT NULL,
  `email` varchar(50) NOT NULL,
  `kelamin` varchar(50) NOT NULL,
  `no_fax` int(15) NOT NULL,
  `status_pesan` varchar(30) NOT NULL,
  `tgl_lahir` int(3) NOT NULL,
  `bulan` text NOT NULL,
  `tahun` int(5) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `provinsi` varchar(25) NOT NULL,
  PRIMARY KEY (`id_member`)
```

```
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;


--
-- Table structure for table `metode_pembayaran`
--

CREATE TABLE IF NOT EXISTS `metode_pembayaran` (
  `id_metode` int(10) NOT NULL AUTO_INCREMENT,
  `jenis_pembayaran` varchar(50) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_metode`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;


--
-- Table structure for table `pembelian`
--

CREATE TABLE IF NOT EXISTS `pembelian` (
  `id_pembelian` int(15) NOT NULL AUTO_INCREMENT,
  `id_supplier` int(15) NOT NULL,
  `jenis_pembayaran` varchar(15) NOT NULL,
  `status_pembelian` varchar(15) NOT NULL DEFAULT 'Pending',
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `nama_produk` varchar(50) NOT NULL,
  `jumlah` int(3) NOT NULL,
  `harga` int(15) NOT NULL,
  `manufaktur` varchar(25) NOT NULL,
  `nama_file` varchar(30) NOT NULL,
  PRIMARY KEY (`id_pembelian`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;


--
-- Table structure for table `pemesanan`
--

CREATE TABLE IF NOT EXISTS `pemesanan` (
  `id_pemesanan` int(8) NOT NULL AUTO_INCREMENT,
  `id_member` int(6) NOT NULL,
  `status_bayar` varchar(20) NOT NULL DEFAULT 'Pending',
  `jenis_trf` varchar(30) NOT NULL,
```

```
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `jumlah` int(2) NOT NULL,
  `id_produk` int(11) NOT NULL,
  `harga` int(12) NOT NULL,
  `nama_produk` varchar(50) NOT NULL,
  PRIMARY KEY (`id_pemesanan`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=85 ;


--
-- Table structure for table `privasipolis`
--

CREATE TABLE IF NOT EXISTS `privasipolis` (
  `id_polis` int(15) NOT NULL AUTO_INCREMENT,
  `judul_polis` varchar(300) NOT NULL,
  `isi_polis` varchar(1000) NOT NULL,
  `penulis_polis` varchar(50) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_polis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;


--
-- Table structure for table `produk`
--

CREATE TABLE IF NOT EXISTS `produk` (
  `id_produk` int(10) NOT NULL AUTO_INCREMENT,
  `nama_produk` varchar(150) NOT NULL,
  `kategori` varchar(50) NOT NULL,
  `manufaktur` varchar(50) NOT NULL,
  `harga` int(10) NOT NULL,
  `deskripsi_produk` varchar(1000) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `nama_file` text NOT NULL,
  `stok` int(10) NOT NULL,
  `diskon` int(3) NOT NULL,
  PRIMARY KEY (`id_produk`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=18 ;


--
```

```
-- Table structure for table `provinsi`
--

CREATE TABLE IF NOT EXISTS `provinsi` (
  `id_provinsi` int(10) NOT NULL AUTO_INCREMENT,
  `nama_provinsi` varchar(25) NOT NULL,
  `ongkos_kirim` int(10) NOT NULL,
  PRIMARY KEY (`id_provinsi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;


--
-- Table structure for table `rekeningtoko`
--

CREATE TABLE IF NOT EXISTS `rekeningtoko` (
  `id_rekening` int(5) NOT NULL AUTO_INCREMENT,
  `nama_bank` varchar(15) NOT NULL,
  `no_rekening` int(20) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_rekening`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;


--
-- Table structure for table `stats`
--

CREATE TABLE IF NOT EXISTS `stats` (
  `no` bigint(20) NOT NULL AUTO_INCREMENT,
  `ip` varchar(15) NOT NULL,
  `tanggal` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `hits` int(11) NOT NULL,
  `online` varchar(255) NOT NULL,
  PRIMARY KEY (`no`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=94 ;


--
-- Table structure for table `supplier`
--

CREATE TABLE IF NOT EXISTS `supplier` (
  `id_supplier` int(10) NOT NULL AUTO_INCREMENT,
```

```
  `nama_supplier` varchar(15) NOT NULL,
  `email` varchar(25) NOT NULL,
  `alamat` varchar(100) NOT NULL,
  `no_telp` bigint(15) NOT NULL,
  `nama_usaha` varchar(100) NOT NULL,
  `deskripsi_usaha` varchar(500) NOT NULL,
  `handphone` int(13) NOT NULL,
  `username` varchar(20) NOT NULL,
  `password` int(20) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_supplier`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;


--
-- Table structure for table `syaratketentuan`
--

CREATE TABLE IF NOT EXISTS `syaratketentuan` (
  `id_syarat` int(5) NOT NULL AUTO_INCREMENT,
  `judul_syarat` varchar(150) NOT NULL,
  `isi_syarat` varchar(1000) NOT NULL,
  `penulis` varchar(50) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_syarat`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;


--
-- Table structure for table `tahun`
--

CREATE TABLE IF NOT EXISTS `tahun` (
  `tahun` int(5) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;



--
-- Table structure for table `tambahstokproduk`
--

CREATE TABLE IF NOT EXISTS `tambahstokproduk` (
  `id_tambahstok` int(15) NOT NULL AUTO_INCREMENT,
```

```
  `nama_produk` varchar(50) NOT NULL,
  `jumlah` int(3) NOT NULL,
  `harga` int(10) NOT NULL,
  `manufaktur` varchar(50) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `id_supplier` int(10) NOT NULL,
  `status_pembelian` varchar(25) NOT NULL DEFAULT 'Pending',
  `id_pembelian` int(10) NOT NULL,
  PRIMARY KEY (`id_tambahstok`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;


--
-- Table structure for table `tanggal`
--

CREATE TABLE IF NOT EXISTS `tanggal` (
  `tanggal` int(3) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;


--
-- Table structure for table `tentangkami`
--

CREATE TABLE IF NOT EXISTS `tentangkami` (
  `id_tentang` int(15) NOT NULL AUTO_INCREMENT,
  `judul_inform` varchar(150) NOT NULL,
  `deskripsi` varchar(300) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_tentang`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;


--
-- Table structure for table `testimonial`
--

CREATE TABLE IF NOT EXISTS `testimonial` (
  `id_testi` int(10) NOT NULL AUTO_INCREMENT,
  `nama` varchar(25) NOT NULL,
  `email` varchar(50) NOT NULL,
  `pesan` varchar(255) NOT NULL,
```

```
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_testi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

5.1.5. Implementasi Antar Muka

Dalam PHP implementasi antarmuka dilakukan dengan sebuah webpage yang berekstensi PHP. Setiap halaman dalam perangkat lunak dibuat file program yang ditulis dengan berekstensi .php. Berikut ini adalah implementasi dari setiap halaman yang dibuat dan dibedakan berdasarkan member, supplier dan admin.

5.1.5.1. Implementasi Halaman Utama

Implementasi halaman yang terdapat pada tampilan awal yang dibuat bentuk file program dengan berekstensi PHP dapat dilihat pada table 5.1 berikut :

Table 5.1

Implementasi Sub Menu Utama

Menu	**Sub Menu**	**Sub-sub Menu**	**Deskripsi**	**Nama File**
Home	-	-	File program untuk menangani halaman utama.	Index.php
Tentang Kami	-	-	File program untuk menangani halaman tentang perusahaan (profil perusahaan).	tentangkami.php

Menu	**Sub Menu**	**Sub-sub Menu**	**Deskripsi**	**Nama File**
Cara Pembelian	-	-	File program untuk menangani halaman cara pembelian produk	Carapemesanan.php
Semua Produk	-	-	File program untuk menangani tampil produk yang ada.	katalog.php
Keranjang Belanja	-	-	File program untuk menangani halaman keranjang belanja member.	keranjangbelanja.php
Testimonials	-	-	File program untuk menangani halaman kesan dan pesan dari pengunjung (testimonial).	testimonial.php
Kontak	-	-	File program untuk menangani halaman kontak admin atau	Kontak.php

			bagian penjualan.	

5.1.5.2. Implementasi Halaman Utama Admin

Implementasi halaman yang terdapat pada admin yang dibuat bentuk file program dengan berekstensi PHP dapat dilihat pada table 5.2 berikut :

Table 5.2

Implementasi Sub Menu Admin

Menu	**Sub Menu**	**Sub-sub Menu**	**Deskripsi**	**Nama File**
Dashboard	-	-	File program untuk menangani halaman utama admin.	Index.php
Produk	-	-	File program untuk menangani halaman tentang produk.	produk.php
Penjualan	-	-	File program untuk menangani halaman detail penjualan.	penjualan.php
Pembelian	Lihat Delete Tambah	-	File program untuk menangani halaman data produk dari supplier.	pembelian.php
Laporan	Cetak	-	File program untuk menangani halaman cetak laporan	laporan.php

			penjualan dan pembelian.	
Sistem	-	-	File program untuk menangani halaman system yang ada pada website.	sistem.php

5.1.5.3. Implementasi Halaman Utama Member

Implementasi halaman yang terdapat pada member yang dibuat bentuk file program dengan berekstensi PHP dapat dilihat pada table 5.3 berikut :

Table 5.3

Implementasi Sub Menu Member

Menu	**Sub Menu**	**Sub-sub Menu**	**Deskripsi**	**Nama File**
Dashboard	-	-	File program untuk menangani halaman utama member.	Index.php
Edit Profil	-	-	File program untuk menangani halaman edit data profil.	profil.php
Histori Pemesanan	-	-	File program untuk menangani halaman histori pemesanan member.	historipemesanan.php

5.1.5.4. Implementasi Halaman Utama Supplier

Implementasi halaman yang terdapat pada member yang dibuat bentuk file program dengan berekstensi PHP dapat dilihat pada table 5.4 berikut :

Tabel 5.4

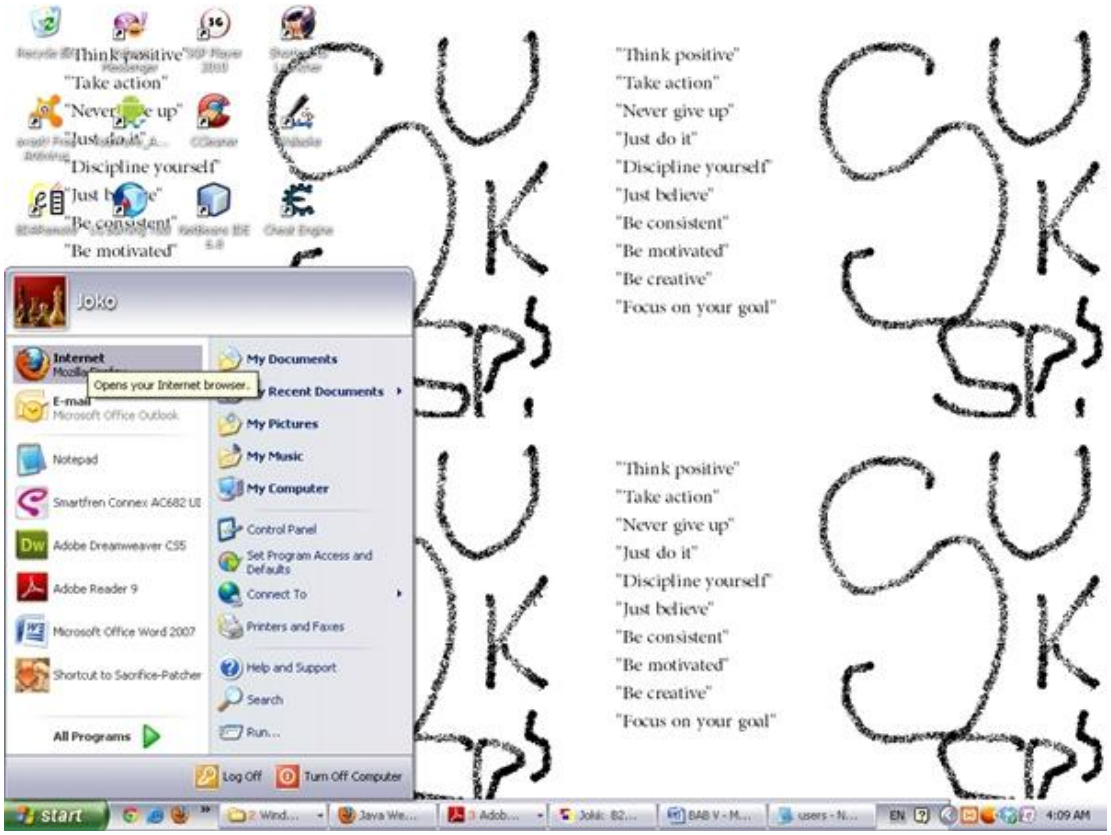
Implementasi Sub Menu Supplier

Menu	**Sub Menu**	**Sub-sub Menu**	**Deskripsi**	**Nama File**
Dashboard	Lihat	-	File program untuk menangani halaman utama Supplier.	Index.php
Pengajuan Produk	-	-	File program untuk menangani halaman pengajuan produk.	pengajuanproduk.php

5.1.6. Implementasi Instalasi Program

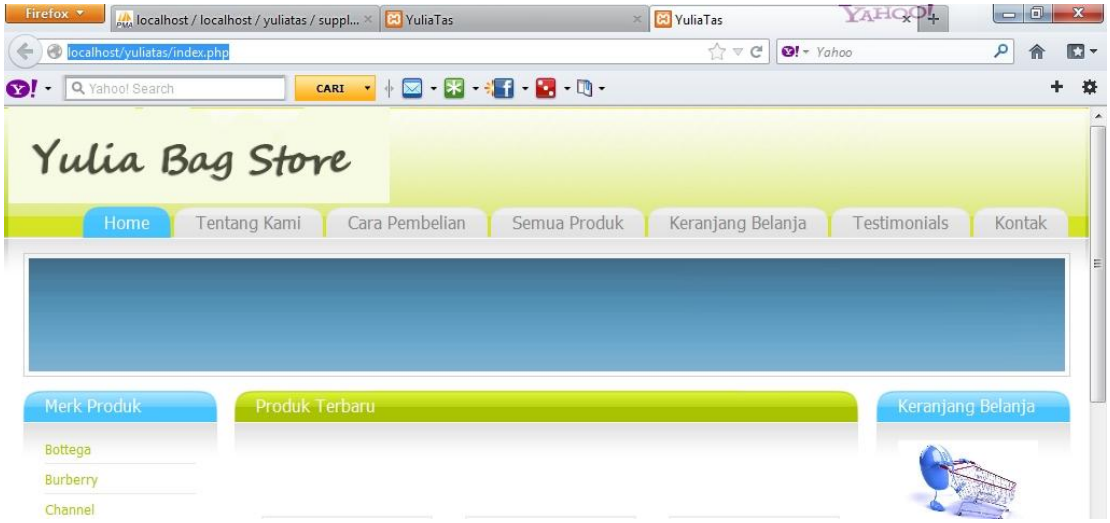
Berikjut ini adalah implementasi instalasi program untuk menggunakan program sistem penjualan dan pembelian, langkah-langkahnya adalah sebagai berikut:

1. Klik menu start kemudian klik Mozila Firefox.



Gambar 5.1. Membuka Mozila Firefox

2. Setelah jendela Mozila Firefox keluar, ketikkan alamat localhost/yuliatas/index.php di url form Mozila Firefox.



Gambar 5.2. Memasukkan alamat url.

3. Aplikasi web akan muncul dalam beberapa detik.

5.1.7. Penggunaan Program

Penggunaan program bertujuan untuk memberikan informasi tentang penggunaan program penjualan dan pembelian di toko yulia tas bagi member, supplier dan bagian penjualan (administrator). Dalam perancangan antarmuka aplikasi penjualan dan pembelian ini terdiri dari beberapa menu dan form yaitu :

1. Halaman utama pengunjung biasa (non member)

   a. Menu utama ini berfungsi untuk proses-proses yang ada di halaman utama ketika memasukin website, menu utama ini mempunyai beberapa fasilitas yaitu lihat data tentang kami, cara pembelian, semua produk, keranjang belanja, testimonial, kontak dan layanan bagi member dan supplier.Fasilitas dalam menu utama ini dikhususkan untuk pengunjung biasa yang tidak

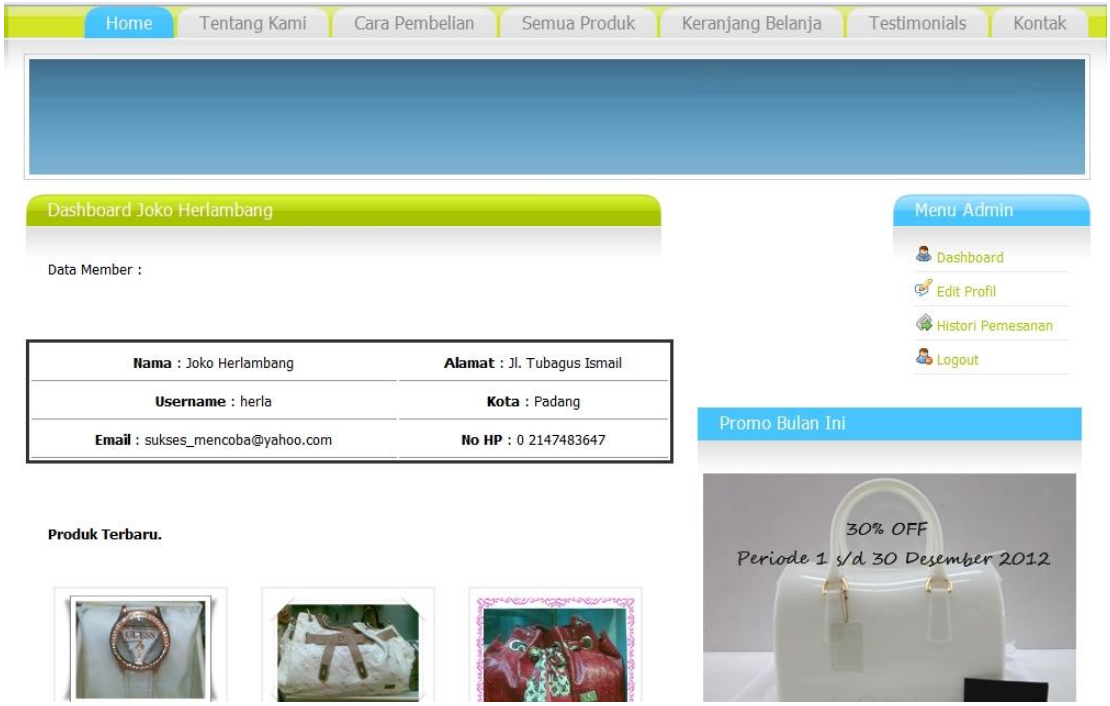
melakukan login. Dibawah ini adalah tampilan layar menu utama aplikasi penjualan dan pembelian di yulia tas.



Gambar 5.3. Halaman Menu Utama.

b. Menu Utama Member

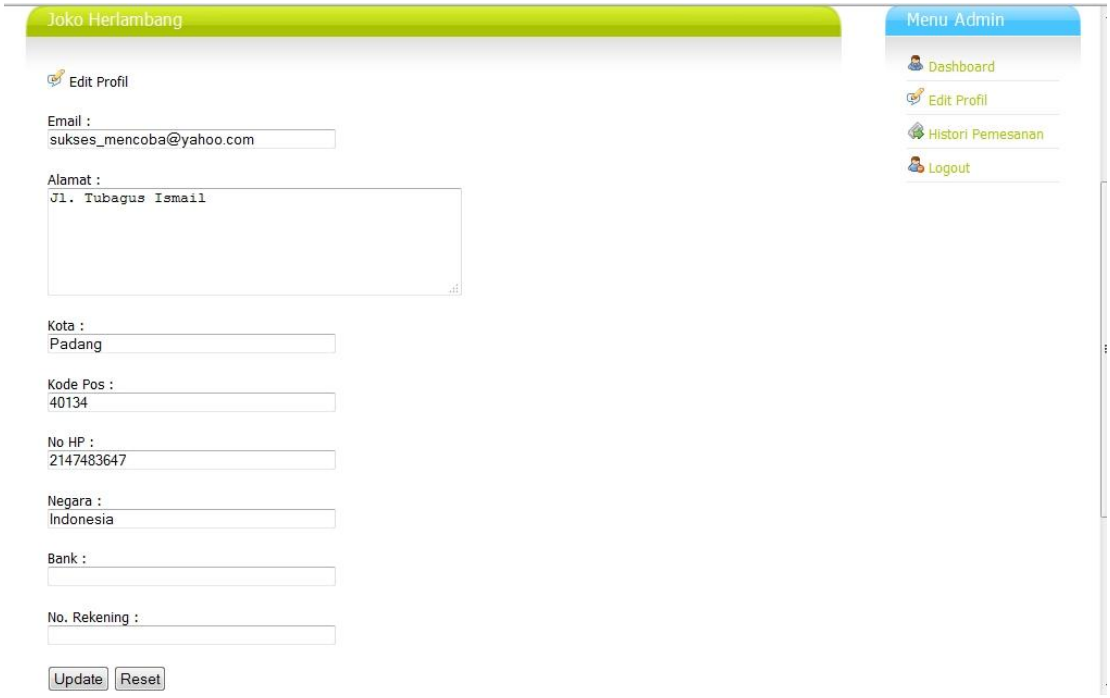
Menu Utama member berfungsi sebagai halaman utama member ketika melakukan login. Disini member bias bernavigasi untuk melakukan edit profil dan meliah histori pemesanan.



Gambar 5.4.Menu Utama Member.

c. Menu Edit Profil Member.

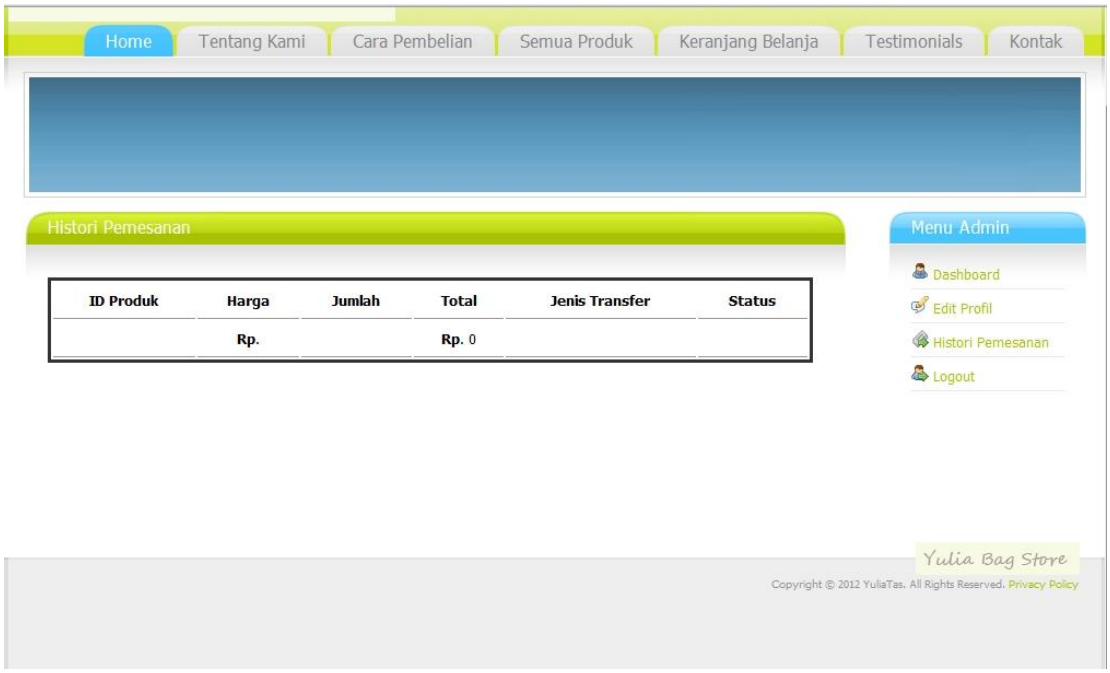
Menu Edit Profil berfungsi untuk mengubah data member,berikut merupakan tampilannya.



Gambar 5.5.Menu Edit Profil.

d. Menu Histori Pemesanan

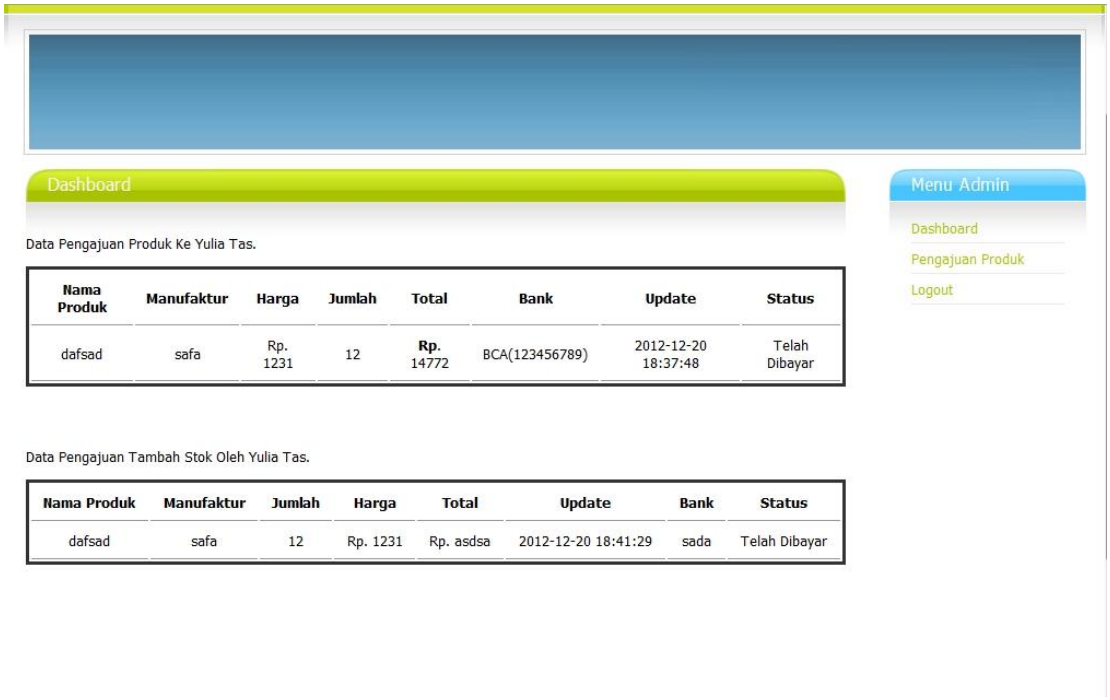
Menu Histori Pemesanan berfungsi untuk mengakses data pemesanan produk yang dilakukan oleh member, berikut merupakan tampilannya.



Gambar 5.6. Menu Histori Pemesanan Member.

e. Menu Utama Supplier

Menu Utama Supplier merupakan halaman utama ketika supplier berhasil login. Pada menu utama ini akan di tampilkan data pengajuan produk oleh supplier dan tambah stok produk oleh admin, berikut merupakan tampilannya.



Gambar 5.7.Menu Utama Supplier.

f. Menu Pengajuan Produk Supplier

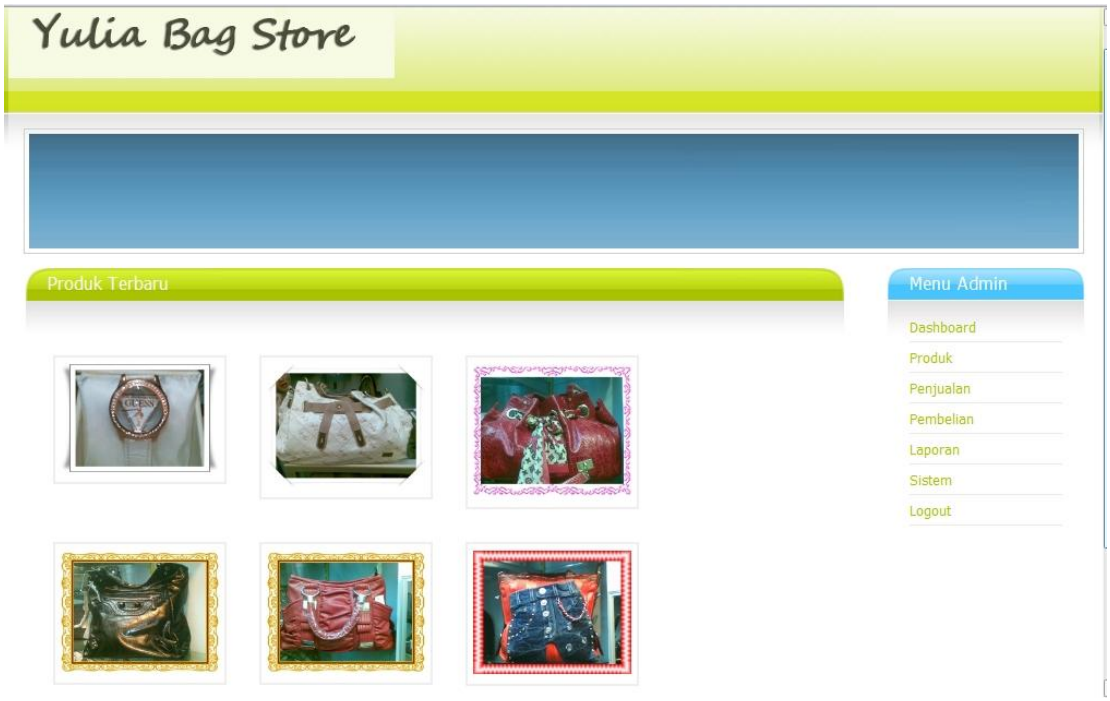
Menu Pengajuan Produk berfungsi untuk melakukan pengajuan atau mendaftarkan produk baru kepada admin, berikut merupakan tampilannya.



Gambar 5.8 Menu Pengajuan Produk

g. Menu Utama Admin

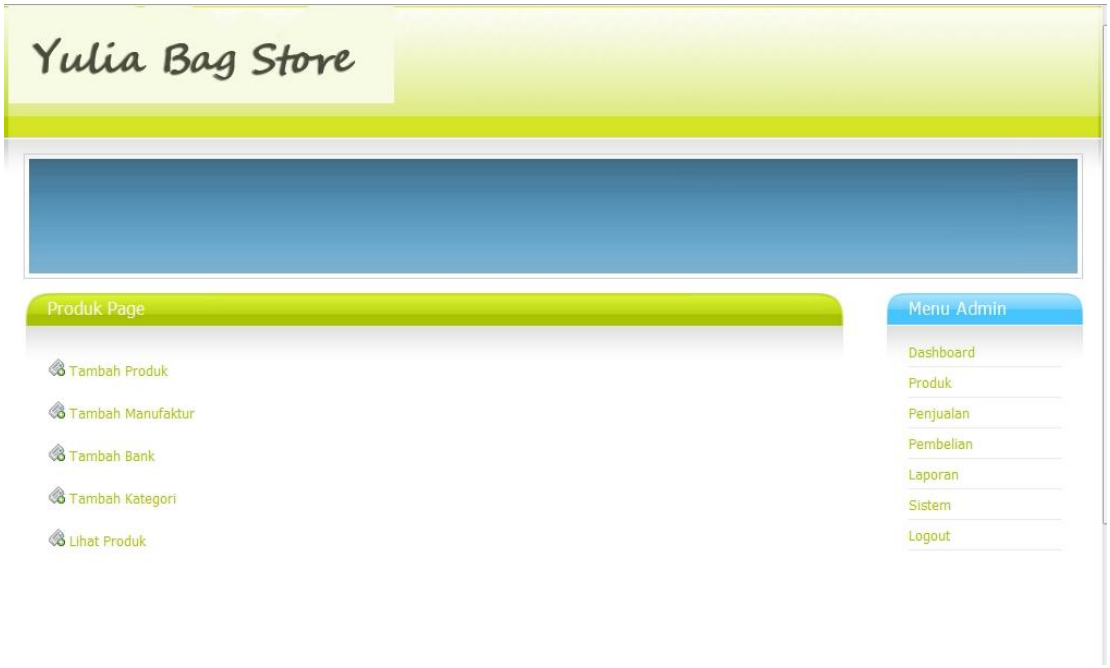
Menu Utama Admin merupakan menu yang akan di akses admin ketika berhasil melakukan login, berikut merupakan tampilannya.



Gambar 5.9, Menu Utama Admin

h. Menu Produk Admin

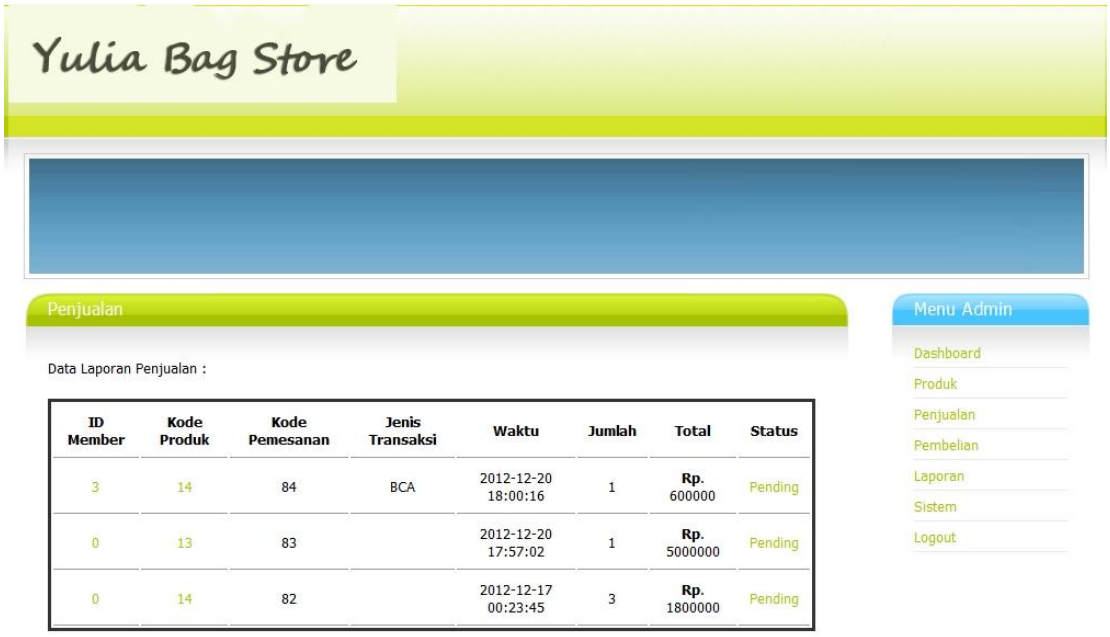
Menu Produk adalah tempat admin mengakses data produk mulai dari tambah, edit, delete, produk, kategori, manufaktur, dan bank, berikut merupakan tampilannya.



Gambar 5.10.Menu Produk Admin.

i. MenuPenjualan Admin

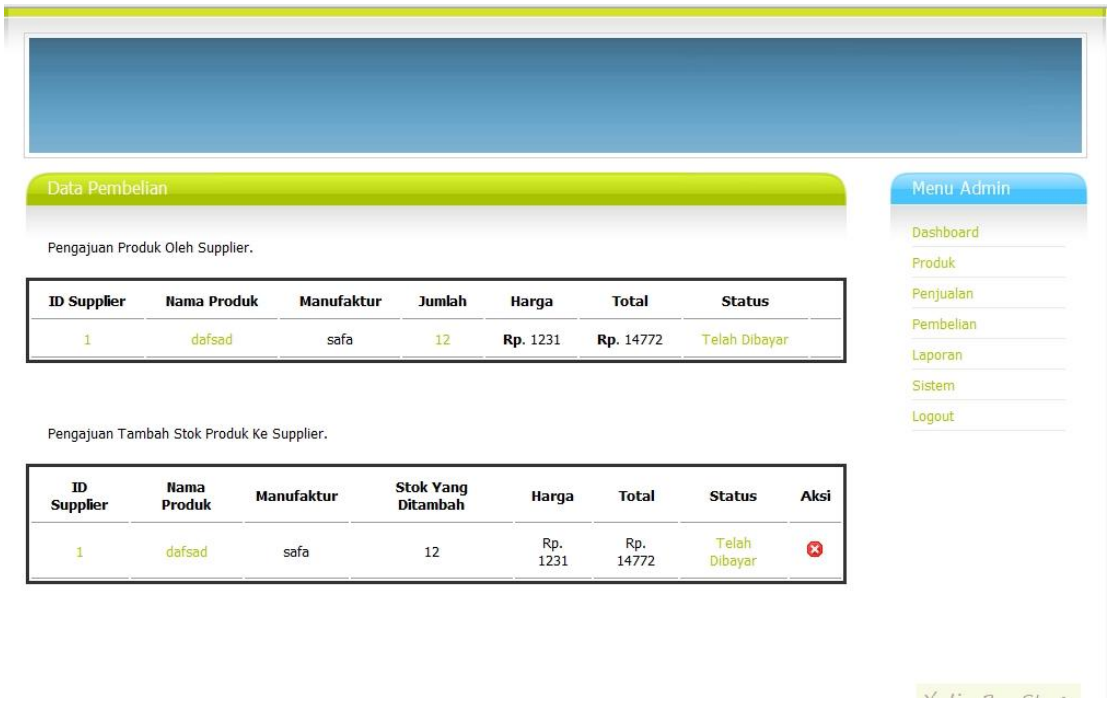
MenuPenjualan merupakan halaman dimana admin dapat melihat daftar produk yang dipesan oleh member, berikut merupakan tampilannya.



Gambar 5.11. Menu Penjualan Admin.

j. Menu Pembelian Admin

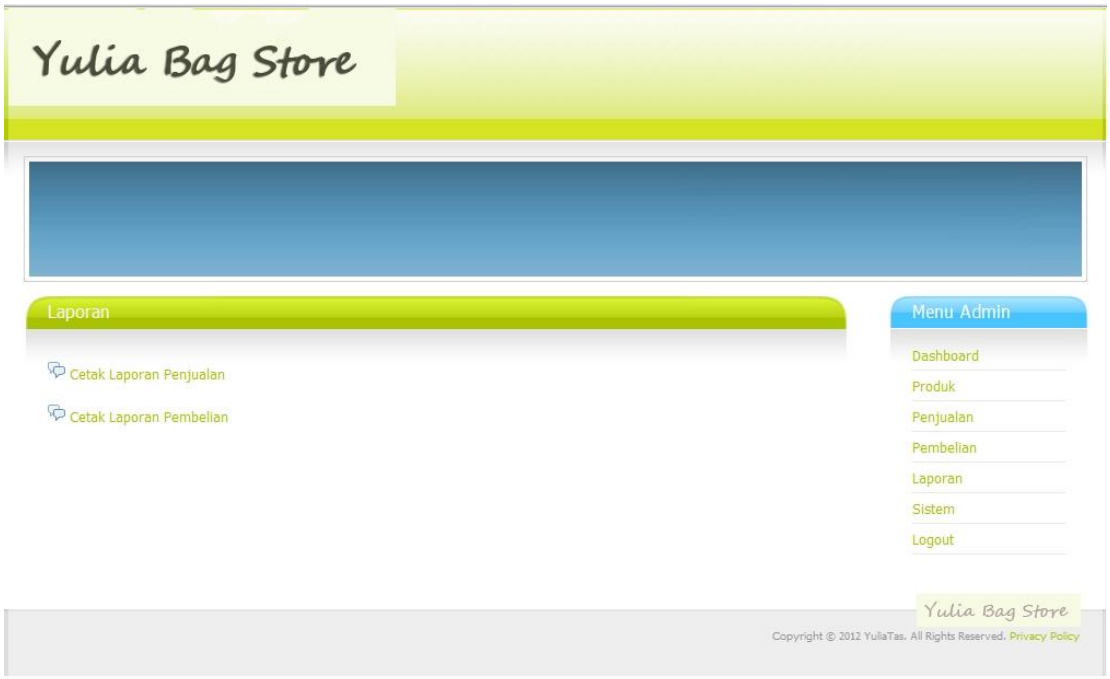
Menu Pembelian merupakan halaman dimana admin dapat melihat daftar pengajuan produk oleh supplier dan juga admin dapat memesan untuk menambah stok produk ke supplier, berikut merupakan tampilannya.



Gambar 5.12. Menu Pembelian Admin

k. Menu Laporan Admin

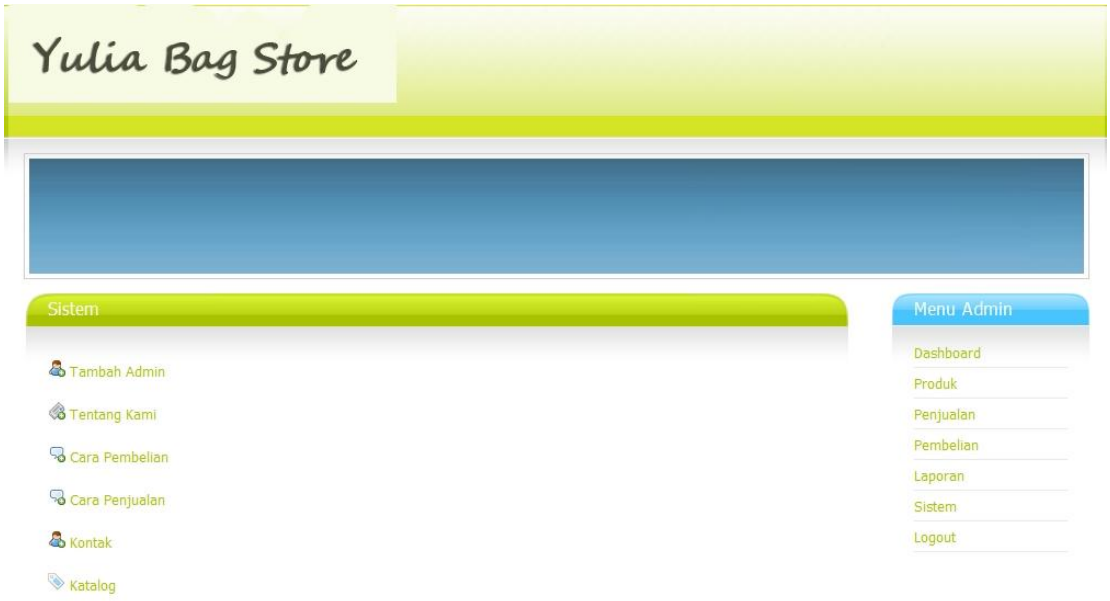
Menu Laporan merupakan halaman dimana admin dapat mencetak laporan penjualan dan pembelian, berikut merupakan tampilannya.



Gambar 5.13. Menu Laporan Admin.

l. Menu Sistem Admin

Menu Sistem admin merupakan menu dimana admin dapat melakukan konfigurasi website untuk menambah, mengubah data-data yang ada pada website penjualan dan pembelian ini, berikut merupakan tampilannya.



Gambar 5.14.Menu Sistem Admin.

5.2 Pengujian

Pengujian adalah bagian yang penting dalam siklus pembangunan perangkat lunak.Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak.Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal yaitu mampu mempresentasikan kajian pokok dari spesifikasi, analisis, perangcangan dan pengkodean dari perangkat lunak itu sendiri.

5.2.1. Rencana Pengujian

Pengujian perangkat lunak pada aplikasi penjualan dan pembelianini menggunakan pengujian *blackbox*. Pengujian*blackbox* berfokus pada persyaratan fungsional perangkat lunak yang dibuat.

Tabel 5.5

Rencana Pengujian

Kelas Uji	**Butir Uji**	**Tingkat Pengujian**	**Jenis Pengujian**
Login member, supplier, admin	Pengecekan data login admin	System	*Black Box*
	Pengecekan data login supplier	Sistem	*Black Box*
	Pengecekan data login member	System	*Black Box*
Kelas Uji	**Butir Uji**	**Tingkat Pengujian**	**Jenis Pengujian**
Pengujian pengisian data penjualan, pembelian, produk	Pengisian data produk.	Modul	*Black Box*
	Pengisian data penjualan dan pembelian	Modul	*Black Box*

5.2.2. Kasus dan Hasil Pengujian

Kasus pengujian ini, penulis akan menguji beberapa *form* diantaranya pengujian login admin, login member, login supplier, pengisian data penjualan dan pembelian produk dan tambah produk.

1. Pengujian Login Admin

Tabel 5.6

Pengujian Login Admin

Kasus dan Hasil Uji Login Admin (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data login admin Username : herla Password : 63136899	Tercantum pada textbox username dan textbox password	Dapat mengisi login admin sesuai yang diharapkan	[x] diterima [ ] ditolak
Klik login	Dapat masuk ke dashboard admin	Tombol login dapat berfungsi sesuai yang diharapkan	[x] diterima [ ] ditolak
Kasus dan Hasil Uji Login Admin (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data login admin dengan username salah.	Gagal melakukan login ke dashboard admin	Admin tidak dapat login dan menampilkan pesan “Maaf, kombinasi	[x] diterima [ ] ditolak

Username : testing Password : apasaja63158		username dan password anda salah. Silahkan coba kembali”	

2. Pengujian Login Member

Tabel 5.7

Pengujian Login Member

Kasus dan Hasil Uji Login Member (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data login member Username : fitri Password : 63136899	Tercantum pada textbox username dan textbox password	Dapat mengisi login member sesuai yang diharapkan	[x] diterima [ ] ditolak
Klik login	Dapat masuk ke dashboard member	Tombol login dapat berfungsi sesuai yang diharapkan	[x] diterima [ ] ditolak
Kasus dan Hasil Uji Login Member (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data login member dengan username salah.	Gagal melakukan login ke dashboard member	member tidak dapat login dan menampilkan pesan “Maaf, kombinasi	[x] diterima [ ] ditolak

Username : testing Password : 500687		username dan password anda salah. Silahkan coba kembali”	
Login Member dengan Password salah Username : bejo Password : sembarang6584	Tidak dapat login ke dashboard Member	Member tidak dapat login dan menampilkan pesan “Maaf, kombinasi username dan password anda salah. Silahkan coba kembali”	[x] diterima [ ] ditolak

3. Pengujian Login Supplier

Tabel 5.8

Pengujian Login Supplier

Kasus dan Hasil Uji Login Supplier (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data login supplier Username : herla Password : 63136899	Tercantum pada textbox username dan textbox password	Dapat mengisi login supplier sesuai yang diharapkan	[x] diterima [ ] ditolak
Klik login	Dapat masuk ke dashboard supplier	Tombol login dapat berfungsi sesuai yang	[x] diterima [ ] ditolak

		diharapkan	
Kasus dan Hasil Uji Login Supplier (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data login supplier dengan username salah. Username : testing Password : 63136899	Gagal melakukan login ke dashboard supplier	supplier tidak dapat login dan menampilkan pesan "Maaf, kombinasi username dan password anda salah. Silahkan coba kembali"	[x] diterima [ ] ditolak
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Login supplier dengan Password salah Username : herla Password : sembarang6584	Tidak dapat login ke dashboard supplier	supplier tidak dapat login dan menampilkan pesan "Maaf, kombinasi username dan password anda salah. Silahkan coba kembali"	[x] diterima [ ] ditolak

4. Pengujian Pengisian Data Pengajuan Produk Dari Supplier (Pembelian)

Tabel 5.9

Pengujian Pengisian Data Data Pengajuan Produk Dari Supplier (Pembelian)

Kasus dan Hasil Uji Data Pengajuan Produk (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih menu pengajuan produk dan mengisi data dengan benar	Form menampilkan data pengajuan produk	Dapat melakukan pengisian data pengajuan produk sesuai yang diharapkan	[x] diterima [ ] ditolak
Klik kirim	Data tersimpan di file pembelian	Data pengisian pengajuan produk tersimpan	[x] diterima [ ] ditolak
Kasus dan Hasil Uji Uji Data Pengajuan Produk(Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data pengajuan produk yang tidak lengkap	Tidak dapat menyimpan data pengajuan produk	Mengeluarkan pesan “data cannot be null”	[x] diterima [ ] ditolak

5. Pengujian Pengisian Produk.

Tabel 5.10

Pengujian Pengisian Data Produk

Kasus dan Hasil Uji Pengisian Data Produk (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih menu tambah produk dan mengisi lengkap data produk	Form menampilkan data produk	Dapat melakukan pengisian data produk sesuai yang diharapkan	[x] diterima [ ] ditolak
Klik submit	Data tersimpan di file produk	Data pengisian produk tersimpan	[x] diterima [ ] ditolak
Klik tombol hapus	Data yang telah terisi telah dihapus	Data telah terhapus sesuai yang diharapkan	[x] diterima [ ] ditolak
Kasus dan Hasil Uji Pengisian Data Produk(Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data produk yang tidak lengkap	Tidak dapat menyimpan data produk	Mengeluarkan pesan “data cannot be null”	[x] diterima [ ] ditolak

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus uji sampel diatas, dapat ditarik kesimpulan bahwa perangkat lunak bebas dari kesalahan sintaks dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.