BAB V

IMPLEMENTASI DAN PENGUJIAN PROGRAM

Pada bab ini akan dilakukan implementasi dan pengujian terhadap sistem yang baru. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemrograman. Setelah implementasi maka dilakukan pengujian terhadap sistem yang baru dan akan dilihat kekurangan-kekurangan pada aplikasi yang baru untuk pengembangan sistem selanjutnya.

5.1. Implementasi

Implementasi adalah suatu bentuk kegiatan yang merupakan rangkaian lanjutan dari kegiatan perancangan perangkat lunak. Implementasi dimaksudkan sebagai usaha untuk mewujudkan hasil dari perancangan perangkat lunak. Hasil ini berarti hasil dari kegiatan rangkaian implementasi perancangan perangkat lunak adalah proses aplikasi yang telah dibuat dan sudah dapat digunakan serta berfungsi dengan baik.Jadi secara garis besar implementasi merupakan proses penerapan rancangan program yang telah dibuat pada bab sebelumnya atau aplikasi dalam melaksanakan sisten informasi pemograman yang dibuat.

5.1.1. Batasan Implementasi (Optional)

Dalam mengimplementasikan perangkat lunak Website Penjualan di Distro Birmingham ini dibatasi oleh penggunaan Software (perangkat lunak) dan Hardware (perangkat keras) minimum.

5.1.2. Implementasi Perangkat Lunak

Untuk implementasi perangkat lunak dibagi menjadi 3 sudut pandang, yaitu:

1. Implementasi Perangkat Lunak untuk pengembangan, antara lain :

a. Apache (3.5.2.2), sebagai web server.

b. PHP (5.4.7) sebagai bahasa pemograman.

c. MySQL(5.5.27), untuk pengembangan dalam pembuatan database.

d. Adobe Dreamweaver CS5, sebagai editor dalam pengembangan desain web.

2. Implementasi Perangkat Lunak untuk admin, antara lain :

a. Apache, sebagai web server dalam keadaaan offline.

b. Web hosting dan domain, dalam keadaan online.

c. Browser, hasil terbaik menggunakan Google Chrome.

d. PDF Reader.100

5.1.3. Implementasi Perangkat Keras

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

1. Untuk admin.

a. Processor dengan kecepatan minimal 2.0 GHz.

b. RAM minimal 1 Gb.

c. Hard Drive, minimal 160 Gb sebagai media penyimpanan data.

d. Mouse, Keyboard, dan Monitor sebagai peralatan antarmuka.

e. Modem, sebagai media untuk koneksi ke internet.

2. Untuk user.

a. Processor dengan kecepatan minimal 1.2 GHz.

b. RAM minimal 1 Gb.

c. Hard Drive, minimal 80 Gb sebagai media penyimpanan data.

d. Mouse, Keybord, dan Monitor sebagai peralatan antarmuka.

e. Modem, sebagai media untuk koneksi ke internet.

5.1.4 Implementasi Basis Data

Pembuatan basis data dilakukan dengan menggunakan bahasa SQL, dimana DBMS yang digunakan adalah MySQL. Adapun implementasi basis datanya dalam

SQL adalah sebagi berikut :

```
-- phpMyAdmin SQL Dump

-- version 3.5.2.2

-- http://www.phpmyadmin.net

-- Host: 127.0.0.1

-- Generation Time: Jun 17, 2013 at 07:08 AM

-- Server version: 5.5.27

-- PHP Version: 5.4.7

Database: `sipenjualan`
```

Table structure for table `admin`

```
CREATE TABLE IF NOT EXISTS `admin` (

    `id_admin` varchar(100) NOT NULL,

    `nama_admin` varchar(25) NOT NULL,

    `email_admin` varchar(50) NOT NULL,
```

```
`no_telepon` varchar(13) NOT NULL,
`username` varchar(25) NOT NULL,
`password` varchar(100) NOT NULL,
`tgl_daftar` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
PRIMARY KEY (`id_admin`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Table structure for table `cara_pembelian`

```
CREATE TABLE IF NOT EXISTS `cara_pembelian` (
`id_carapembelian` int(2) NOT NULL AUTO_INCREMENT,
`judul` varchar(150) NOT NULL,
`isi` varchar(10000) NOT NULL,
`update` timestamp NOT NULL
DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
PRIMARY KEY (`id_carapembelian`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2
```

Table structure for table `detail_pesanan`

```
CREATE TABLE IF NOT EXISTS `detail_pesanan` (
  `id_detailpesan` int(15) NOT NULL
AUTO_INCREMENT,
  `id_pemesanan` int(15) NOT NULL,
  `id_produk` int(15) NOT NULL,
  `tgl_pesan` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `status` varchar(25) NOT NULL DEFAULT 'Pending',
  `username` varchar(25) NOT NULL,
  `nama_produk` varchar(25) NOT NULL,
`harga` int(20) NOT NULL,
  `jumlah` int(20) NOT NULL,
  `total` int(20) NOT NULL,
  `id_member` int(15) NOT NULL,
  PRIMARY KEY (`id_detailpesan`),
  KEY `id_produk` (`id_produk`),
  KEY `id_produk_2` (`id_produk`),
```

```
  KEY `id_pemesanan` (`id_pemesanan`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1
AUTO_INCREMENT=9
```

Table structure for table `info_bank`

```
CREATE TABLE IF NOT EXISTS `info_bank` (
  `id_bank` int(5) NOT NULL AUTO_INCREMENT,
  `nama_bank` varchar(10) NOT NULL,
  `atas_nama` varchar(25) NOT NULL,
  `no_rekening` int(25) NOT NULL,
  PRIMARY KEY (`id_bank`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2
Table structure for table `info_kontak`
CREATE TABLE IF NOT EXISTS `info_kontak` (
  `id_kontak` int(15) NOT NULL
AUTO_INCREMENT,
  `namapengirim_kontak` varchar(25) NOT NULL,
  `emailpengirim_kontak` varchar(35) NOT NULL,
  `pesan` varchar(255) NOT NULL,
```

```
  `no_hp` varchar(13) NOT NULL,
  `tanggal_kirim` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_kontak`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1
```

Table structure for table `info_perusahaan`

```
CREATE TABLE IF NOT EXISTS `info_perusahaan` (
  `nama_perusahaan` varchar(50) NOT NULL,
  `owner` varchar(25) NOT NULL,
  `email_owner` varchar(25) NOT NULL,
  `telepon` varchar(13) NOT NULL,
  `alamat` varchar(150) NOT NULL,
  `update` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Table structure for table `kategori_produk`

```
CREATE TABLE IF NOT EXISTS `kategori_produk` (
```

```
  `id_ketegori` int(15) NOT NULL
AUTO_INCREMENT,
  `kode_kategori` varchar(10) NOT NULL,
  `nama_kategori` varchar(40) NOT NULL,
`meta_deskripsi` varchar(200) NOT NULL,
  `tag` varchar(200) NOT NULL,
  `induk_kategori` varchar(25) NOT NULL,
  PRIMARY KEY (`id_ketegori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=18 ;
```

Table structure for table `keranjang`

```
CREATE TABLE IF NOT EXISTS `keranjang` (
  `id_keranjang` int(10) NOT NULL AUTO_INCREMENT,
`id_produk` int(10) NOT NULL,
  `jumlah` int(15) NOT NULL,
  `total` int(20) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `ukuran` varchar(10) NOT NULL,
```

```
  `warna` varchar(25) NOT NULL,
  `username` varchar(25) NOT NULL,
  `nama_produk` varchar(100) NOT NULL,
  `harga` int(20) NOT NULL,
  `status` varchar(25) NOT NULL,
  `id_member` int(15) NOT NULL,
 PRIMARY KEY (`id_keranjang`),
  KEY `id_produk` (`id_produk`)
) ENGINE=InnoDB  DEFAULT
CHARSET=latin1 AUTO_INCREMENT=80
```

Table structure for table `kontak`

```
CREATE TABLE IF NOT EXISTS `kontak` (
  `id_kontak` int(10) NOT NULL AUTO_INCREMENT,
  `nama` varchar(25) NOT NULL,
  `email` varchar(35) NOT NULL,
  `web` varchar(25) DEFAULT NULL,
  `pesan` varchar(255) NOT NULL,
```

```
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE
CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_kontak`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1
AUTO_INCREMENT=4 ;
```

Table structure for table `kota`

```
CREATE TABLE IF NOT EXISTS `kota` (
  `id_kota` int(10) NOT NULL AUTO_INCREMENT,
  `nama_kota` varchar(50) NOT NULL,
  PRIMARY KEY (`id_kota`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1
AUTO_INCREMENT=34 ;
```

Table structure for table `member`

```
CREATE TABLE IF NOT EXISTS `member` (
  `id_member` int(50) NOT NULL
AUTO_INCREMENT,
  `username` varchar(25) NOT NULL,
```

```
  `password` varchar(50) NOT NULL,
  `nama` varchar(25) NOT NULL,
  `email` varchar(35) NOT NULL,
  `telepon` varchar(13) NOT NULL,
`jenis_kelamin` varchar(25) NOT NULL,
  `alamat` varchar(150) NOT NULL,
  `kota` varchar(25) NOT NULL,
  `kodepos` int(7) NOT NULL,
  `provinsi` varchar(25) NOT NULL,
  `tgl_lahir` date NOT NULL,
`tgl` int(3) NOT NULL,
  `bulan` int(3) NOT NULL,
  `tahun` int(5) NOT NULL,
  `tgl_daftar` timestamp
NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_member`)
)
```

```
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

Table structure for table `menu`

```
CREATE TABLE IF NOT EXISTS `menu` (
  `kode_menu` varchar(100) NOT NULL,
  `nama_menu` varchar(50) NOT NULL,
  `id_menu` int(5) NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`id_menu`)
) ENGINE=InnoDB
DEFAULT CHARSET=latin1 AUTO_INCREMENT=24 ;
```

Table structure for table `provinsi`

```
CREATE TABLE IF NOT EXISTS `provinsi` (
  `id_provinsi` int(3) NOT NULL AUTO_INCREMENT,
`nama_provinsi` varchar(50) NOT NULL,
  `ongkos_kirim` int(10) NOT NULL,
  PRIMARY KEY (`id_provinsi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=34 ;
```

Table structure for table `tb_event`

```
CREATE TABLE IF NOT EXISTS `tb_event` (
  `id_event` int(15) NOT NULL
AUTO_INCREMENT,
  `judul` varchar(255) NOT NULL,
  `isi` varchar(15000) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `nama_file` varchar(100) NOT NULL,
  `tahun` int(6) NOT NULL,
  PRIMARY KEY (`id_event`)
) ENGINE=InnoDB  DEFAULT
CHARSET=latin1 AUTO_INCREMENT=4 ;
```

Table structure for table `tb_konfirmasi`

```
CREATE TABLE IF NOT EXISTS `tb_konfirmasi` (
  `id_konfirmasi` int(10) NOT NULL AUTO_INCREMENT,
  `id_member` int(10) NOT NULL,
  `an_asal` varchar(65) NOT NULL,
```

```
`bank_asal` varchar(65) NOT NULL,
  `rek_asal` int(35) NOT NULL,
  `an_tujuan` varchar(35) NOT NULL,
`bank_tujuan` varchar(35) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_konfirmasi`),
  KEY `id_member` (`id_member`)
)
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

Table structure for table `tb_produk`

```
CREATE TABLE IF NOT EXISTS `tb_produk` (
  `id_produk` int(15) NOT NULL AUTO_INCREMENT,
`nama_produk` varchar(25) NOT NULL,
  `kategori_produk` varchar(25) NOT NULL,
  `harga_produk` int(15) NOT NULL,
  `diskon_produk` int(3) NOT NULL,
  `stok_produk` int(5) NOT NULL,
```

```
`deskripsi_produk` varchar(255) NOT NULL,
  `tag_keyword` varchar(255) NOT NULL,
  `nama_file` varchar(100) NOT NULL,
  `jenis_pemakai` varchar(25) NOT NULL,
  `nama_file2` varchar(100) DEFAULT NULL,
  `nama_file3` varchar(100) DEFAULT NULL,
  `nama_file4` varchar(100) DEFAULT NULL,
  `ukuran` varchar(5) NOT NULL,
  `warna` varchar(25) NOT NULL,
  `nama_file5` varchar(50) DEFAULT NULL,
`nama_file6` varchar(50) DEFAULT NULL,
  `feature` varchar(25) DEFAULT 'Tidak',
  PRIMARY KEY (`id_produk`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=15 ;
```

Table structure for table `tb_slide`

```
CREATE TABLE IF NOT EXISTS `tb_slide` (
  `id_slide` int(5) NOT NULL AUTO_INCREMENT,
```

```
  `title` varchar(25) NOT NULL,
  `nama_file` varchar(55) NOT NULL,
  `link` varchar(100) NOT NULL,
  `update` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_slide`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

Table structure for table `tb_warna`

```
CREATE TABLE IF NOT EXISTS `tb_warna` (
  `id_warna` int(5) NOT NULL AUTO_INCREMENT,
  `warna` varchar(25) NOT NULL,
  PRIMARY KEY (`id_warna`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1
AUTO_INCREMENT=15 ;
```

Table structure for table `tentang_kami`

```
CREATE TABLE IF NOT EXISTS `tentang_kami` (
  `id_tentang` int(2) NOT NULL AUTO_INCREMENT,
  `judul` varchar(150) NOT NULL,
```

```
  `isi` varchar(10000) NOT NULL,

`tgl_pembuatan` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
ON UPDATE CURRENT_TIMESTAMP,

  PRIMARY KEY (`id_tentang`)

) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

5.1.5. Implementasi antar muka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibuat dan pengkodeannya dalam bentuk file program.Berikut ini adalah implementasi antarmuka yang dibuat dan dibedakan antara antarmuka untuk user dengan administrator.

5.1.5.1 Implementasi Antarmuka Untuk Konsumen / Member

Berikut ini adalah implementasi antarmuka berdasarkan user.

1.Halaman Utama

Berikut ini merupakan deskripsi dari halaman utama dari web Birmingham.

Tabel 5.1. Implementasi Antarmuka Konsumen

Menu	Deskripsi	Nama File
Index	Menu ini berisikan tentang tampilan utama web Birmingham	Index.php
Tentang Kami	Menu ini berisi tentang info atau profil Birmingham	About.php
Produk	Menu ini menampilkan kategori yang disediakan	Katalog_produk.php
Event	Menu ini menampilkan beberapa informasi seputar musik dari bebera video di youtube	Event.php
Cara Pembelian	Menu ini berisikan langkahlangkah tentang bagaimana carabertransaksi di website Birmingham	Howtoorder.php
Kontak	Menu ini berisikan tentang kontak dari Birmingham yang bisa dihubungi	Contact.php
Login	Menu ini berfungsi untuk konsumen yang akan melakukan login	Login.php
Edit Profile	Menu ini digunakan bagi konsumen jika ingin mengedit profil	Index.php
Ganti password	Menu ini digunakan konsumen jika ingin mengganti password	Index.php
Keranjang Belanja	Menu ini menampilkan detail produk yang dipesan konsumen	Index.php
Histori Transaksi	Menu ini menampilkan histori transaksi dari produk yang telah dipesan oleh konsumen	Index.php
Konfirmasi Pembayaran	Menu ini digunakan oleh konsumen untuk malakukan konfirmasi pembayaran	Index.php
Ubah alamat pengiriman	Menu ini digunakan oleh konsumen untuk merubah alamat pengiriman yang di inginkan	Index.php

Nota	Menu ini dibuat agar konsumen dapat mencetak invoice tentang pemesanan yang telah dilakukan	Suskes.php
Logout	Menu ini dibuat untuk konsumen yang telah selesai dan ingin keluar dari program	Index.php

5.1.5.2 Implementasi Antar Muka untuk Admin

Halaman ini berisi mengenai file-file yang dapat diakses oleh Admin, mengenai pengolahan data master seperti data kategori produk, data produk, data pengiriman, kemudian untuk pengolahan data transaksi seperti data pemesanan masuk, data pemberitauan masuk, dan laporan penjualan,laporan minimum stok,dll. Untuk lebih jelasnya akan dijelaskan pada tabel berikut:

Tabel 5.2. Implementasi Antar Muka untuk Admin

Menu	Deskripsi	Nama File
Login	Menu ini berfungsi untuk login Admin	Login.php
Dashboard	Menu ini terdapat informasi admin untuk menjalankan program ini dengan baik.	Index.php
Profile	Menu ini berfungsi untuk menampilkan profile admin,edit profile dang anti password	Index.php
Produk	Menu ini berfungsi untuk menampilkan data produk, tambah produk, kategori produk, stok prduk minimum serta fitur slide	Index.php
Event	Menu ini berfungsi untuk melihat data event dan menambahkan Event.	Index.php
Penjualan	Menu ini berfungsi untuk menampilkan data hasil transaksi yang telah di konfirmasi dan untuk mecetak laporan penjualan.	Index.php

Informasi Bank	Menu ini berfungsi untuk melihat data rekening dan menambahkan rekening Birmingham	Index.php
Sistem pengaturan	Menu ini berfungsi untuk mengontrol hak akses website Birmingham diantara lain seperti (Data Admin, tambah admin, menginput atau mengedit cara pembelian, menginput atau mengedit cara tentang kami, dan melihat kontak distro yang bekerja sama dengan disrto Birmingham.	Index.php

5.1.6 Penggunaan Program

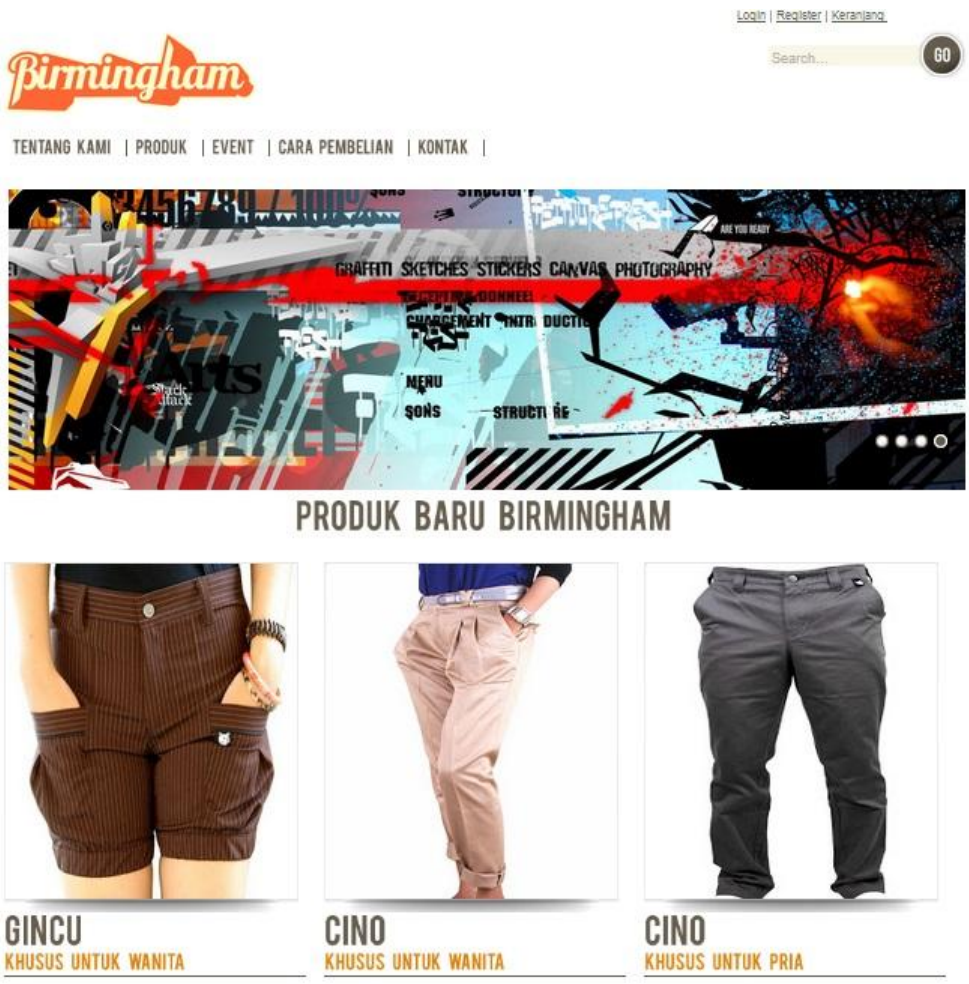
Penggunaan program ini bertujuan untuk mempermudah user dalam menjalankan / aplikasi Butik Shop ini, yaitu dengan mengikuti langkah-langkah yang akan dijelaskan. Dimana akan dijelaskan mulai dari tampilan utama ketika user menggunakan program, kemudian login user, melakukan proses transaksi dan terakhir logout.

5.1.6.1 Penggunaan Program untuk Konsumen

Berikut ini langkah-langkah penggunaan program aplikasi Distro BirminghamUntuk konsumen :

1.1. Halaman Index

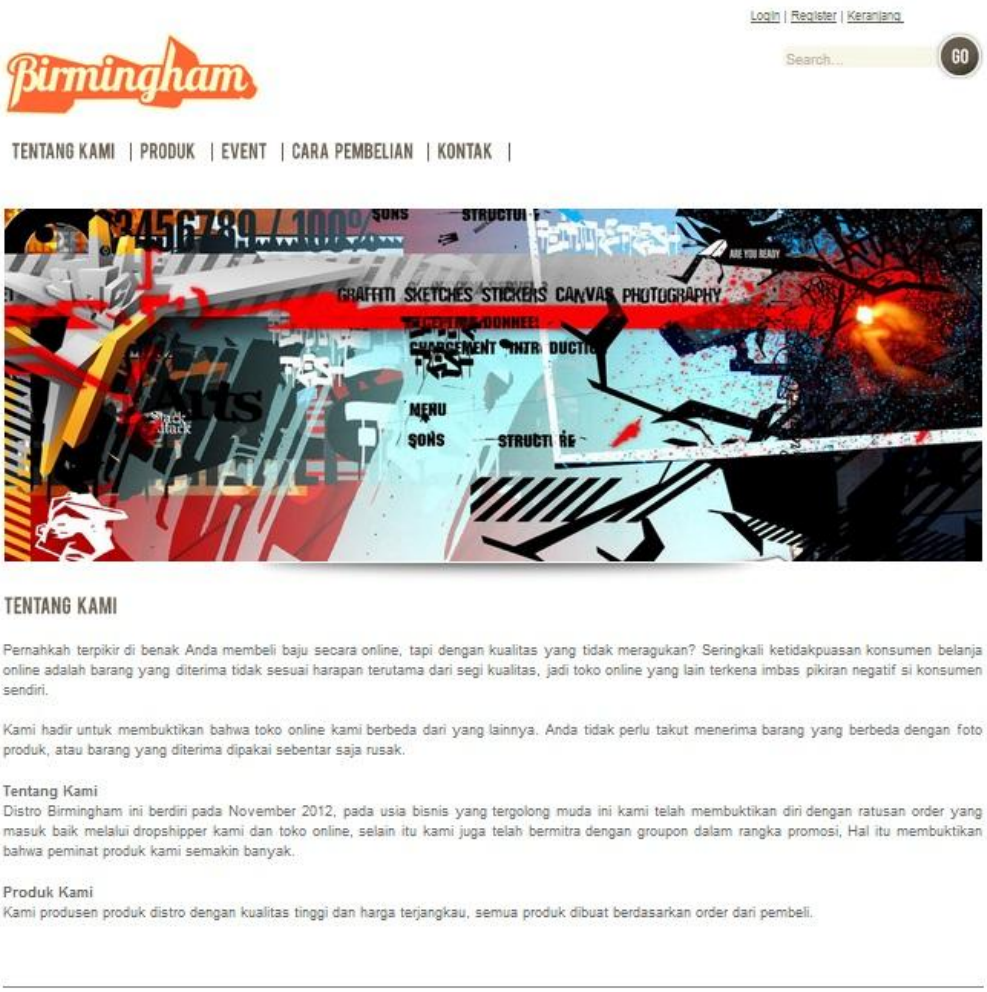
Pada halaman index ini berupa slideshow banner sebagai tampilan depan dari website Birmingham dan pengguna dapat melihat keseluruhan menu website Birmingham.



Gambar 5.1 Halaman Depan Website Birmingham

1.2. Halaman Tentang kami

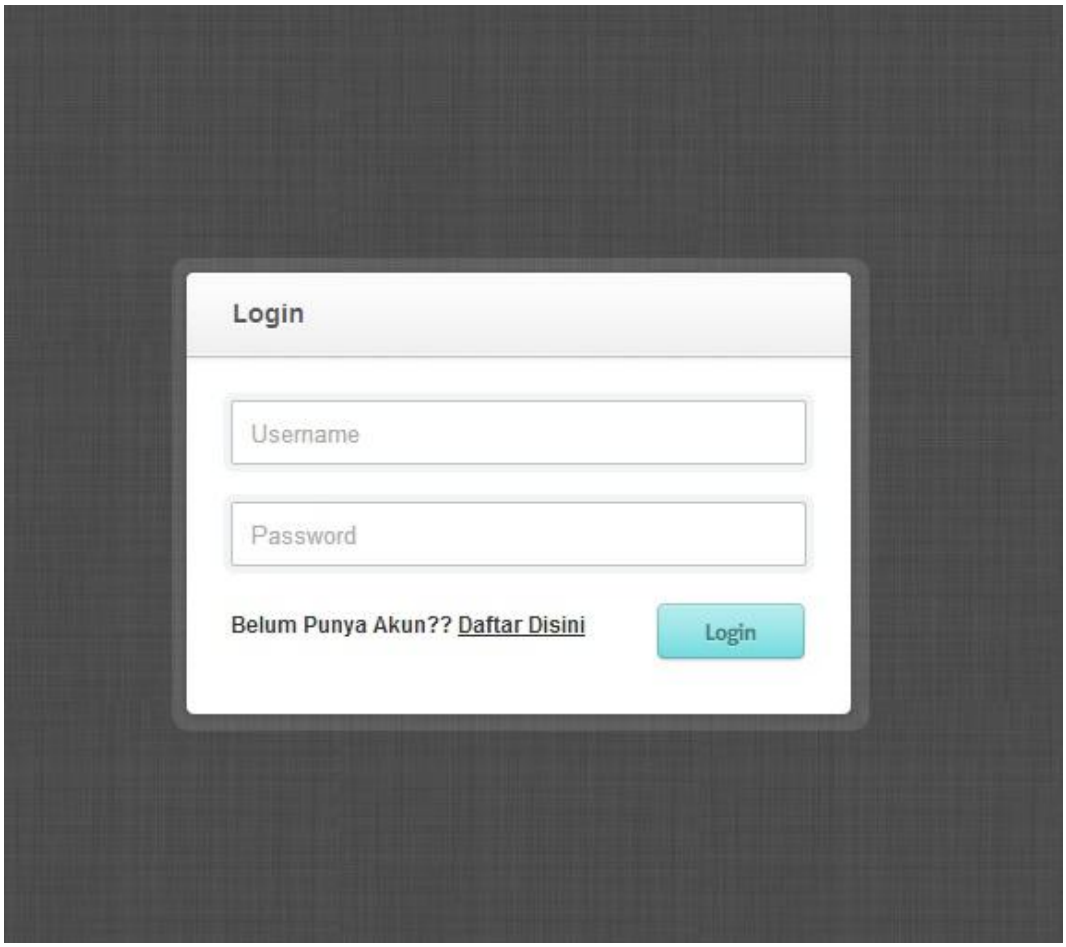
Pada halaman beranda ini terdapat beberapa Informasi Distro Birmingham untuk para konsumen.



Gambar 5.2 Halaman Tentang Kami

1.3. Form Login Konsumen

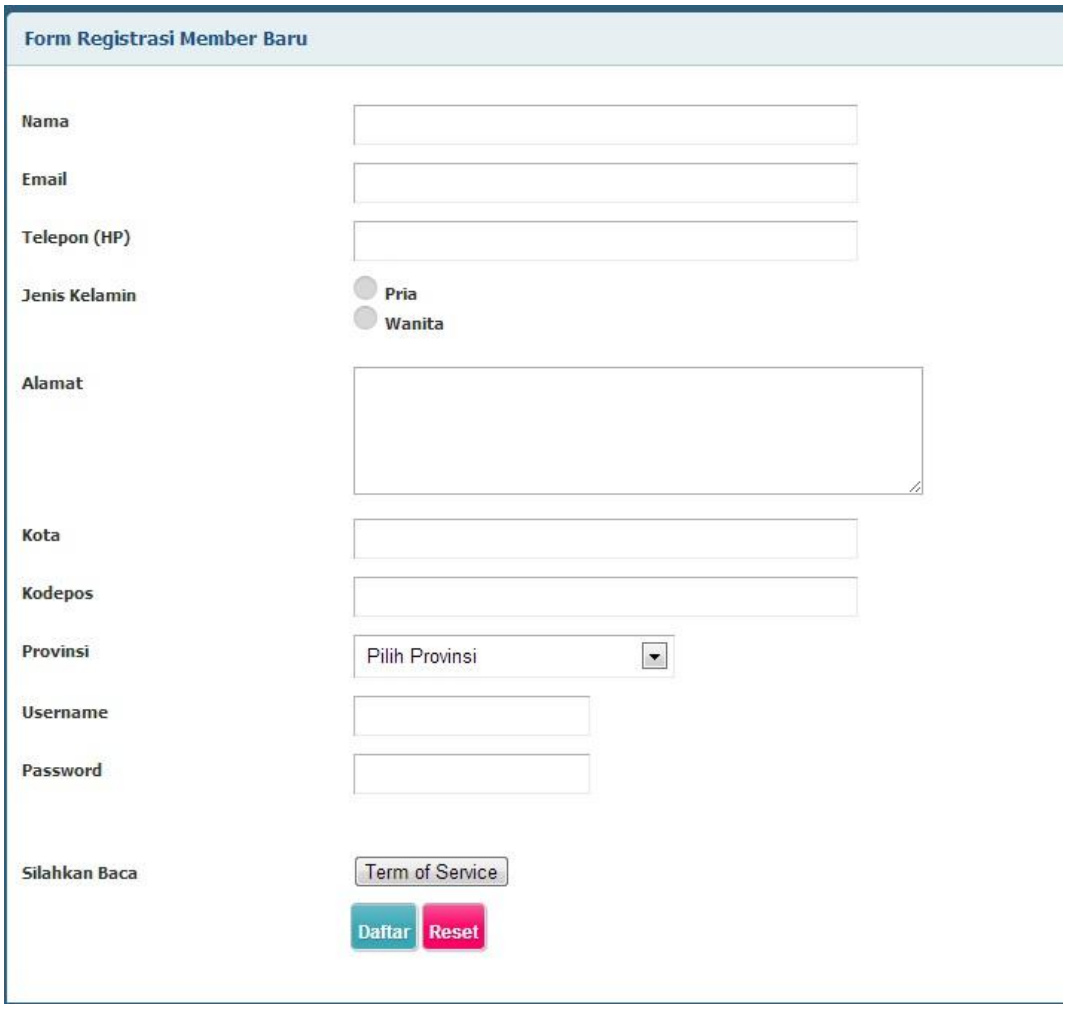
Form dimana jika konsumen ingin melakukan pembelian, maka pengguna diwajibkan mengisi login Pengguna, dan jika belum terdaftar maka pengguna diberikan fasilitas pendaftaran yang ditujukan untuk menjadi pengguna.



Gambar 5.3 Halaman Login User

1.4. Tampilan Menu Pendaftaran

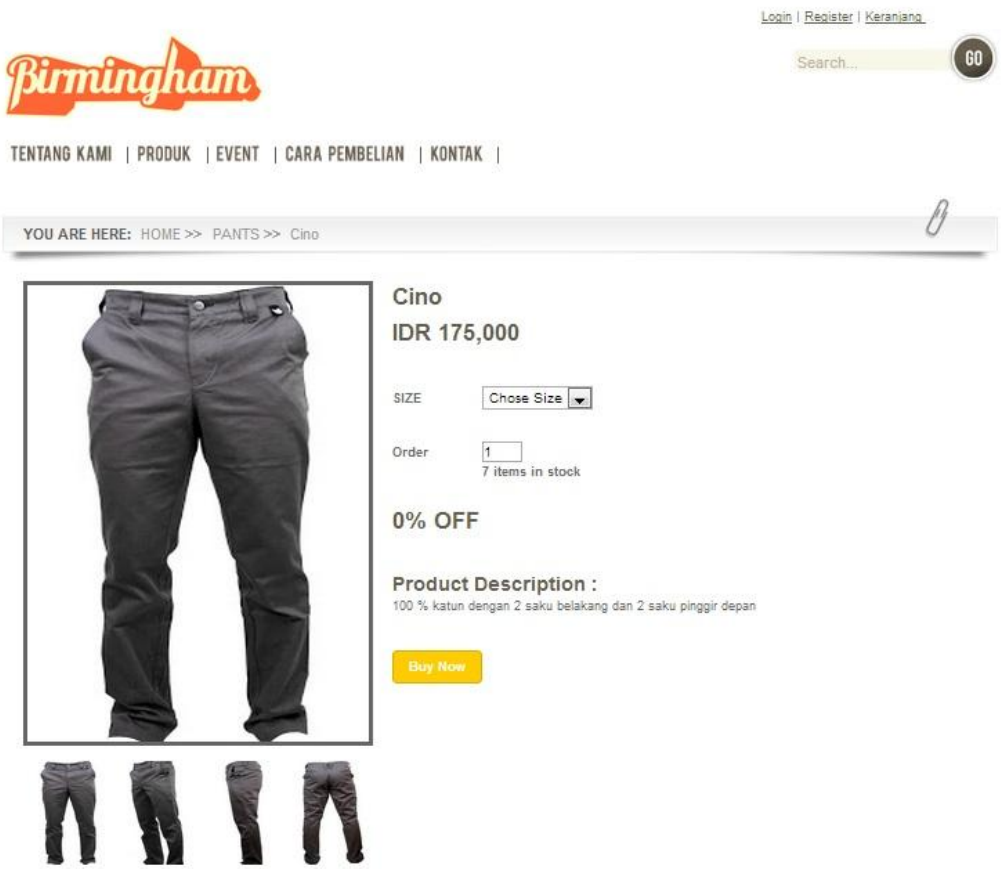
Untuk melakukan transaksi, jika konsumen belum memiliki akun dapat mendaftar terlebih dahulu. Form pendaftaran bisa dilihat pada gambar di bawah ini.



Gambar 5.4 Halaman Form Pendaftaran

1.5. Menu Produk

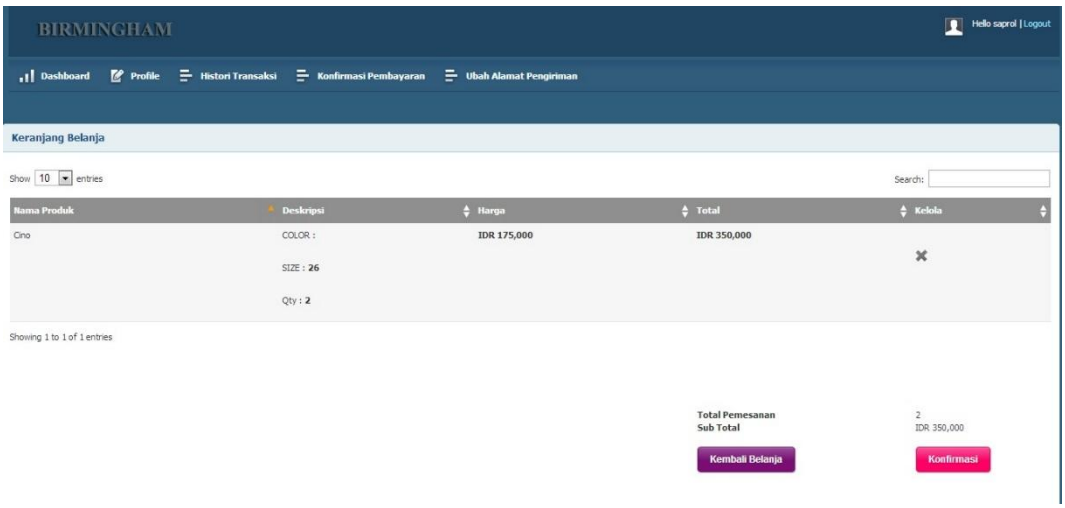
Tampilan dimana konsumen akan memilih produk yang di inginkan dan dapat memilih produk yang ingin dipesan, apabila ada produk yang ingin dipesan, maka pengguna dapat mengklik kolom “Size” Masukan jumlah produk yang ingin dipesan pada kolom “Order” dan order pengguna akan ditampilkan pada Keranjang Belanja.



Gambar 5.5 Halaman Produk

1.6. Menu Keranjang

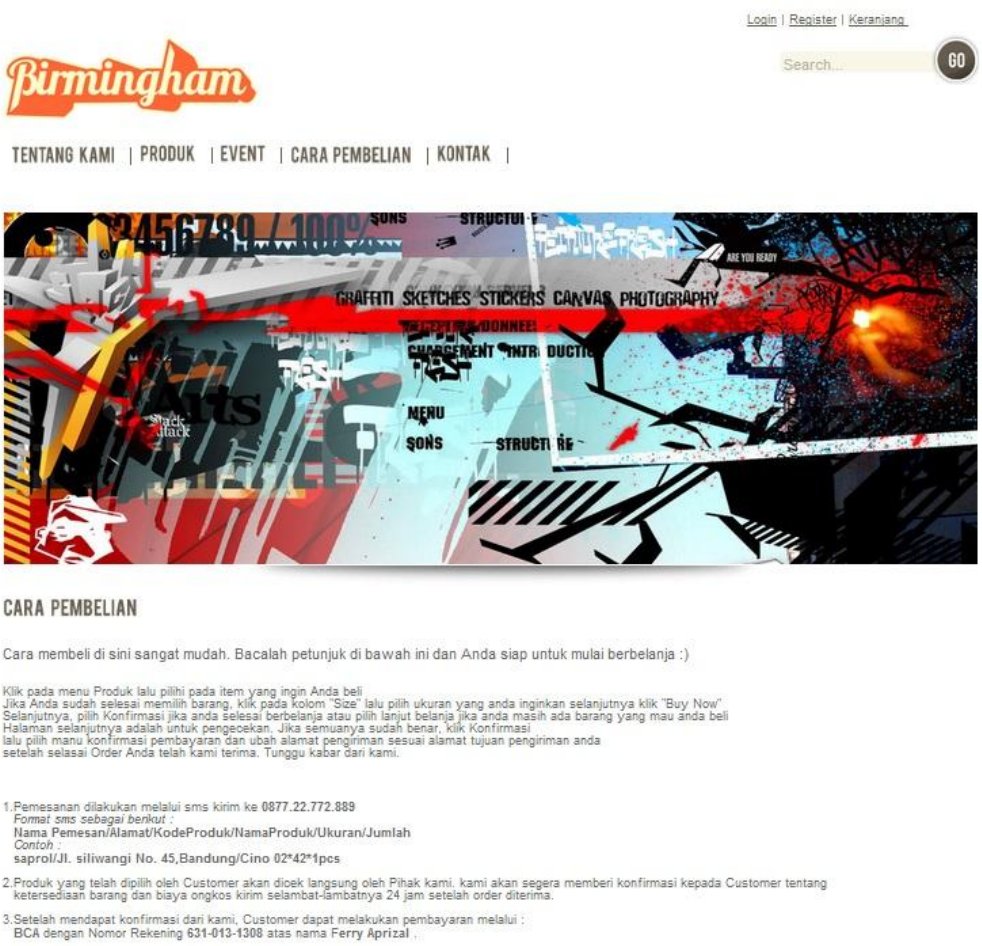
Tampilan dimana Konsumen sudah melakukan pemesanan dengan memilih produk terlebih dahulu dan produk yang dipesan masuk pada Keranjang Belanja untuk mengetahui jenis dan total pesanan, dan Apabila konsumen ingin mengganti cukup menekan tombol "X(delete)" dan apalabila ingin memilih produk lagi, maka cukup menekan tombol “kembali Belanja” apabila selesai konsumen dapat menekan tombol "Konfirmasi".



Gambar 5.6 Halaman Keranjang

1.7. Menu Cara Pembelian

Menu ini berisi mengenai tata cara melakukan transaksi/pemesanan pada Website Birmingham.



Gambar 5.7 Halaman Cara Pembelian

1.8. Menu Event

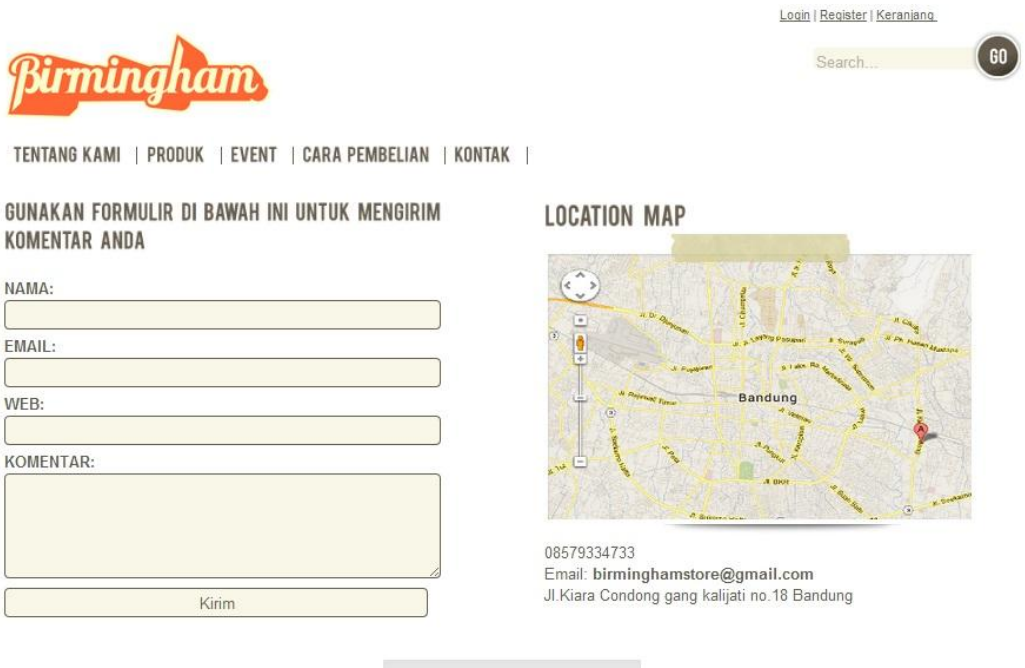
Menu ini berfungsi untuk mengetahui beberapa informasi seputar musik dari bebera video di youtube.



Gambar 5.8 Halaman Event

1.9. Menu Kontak

Menu ini berfungsi untuk memudahkan pengguna agar dapat berkomentar tentang kelebihan dan kekurangan web Birmingham dengan melakukan pengisian form komentar yang telah disediakan dan melihat lewat Map lokasi Store Distro Birmingham.



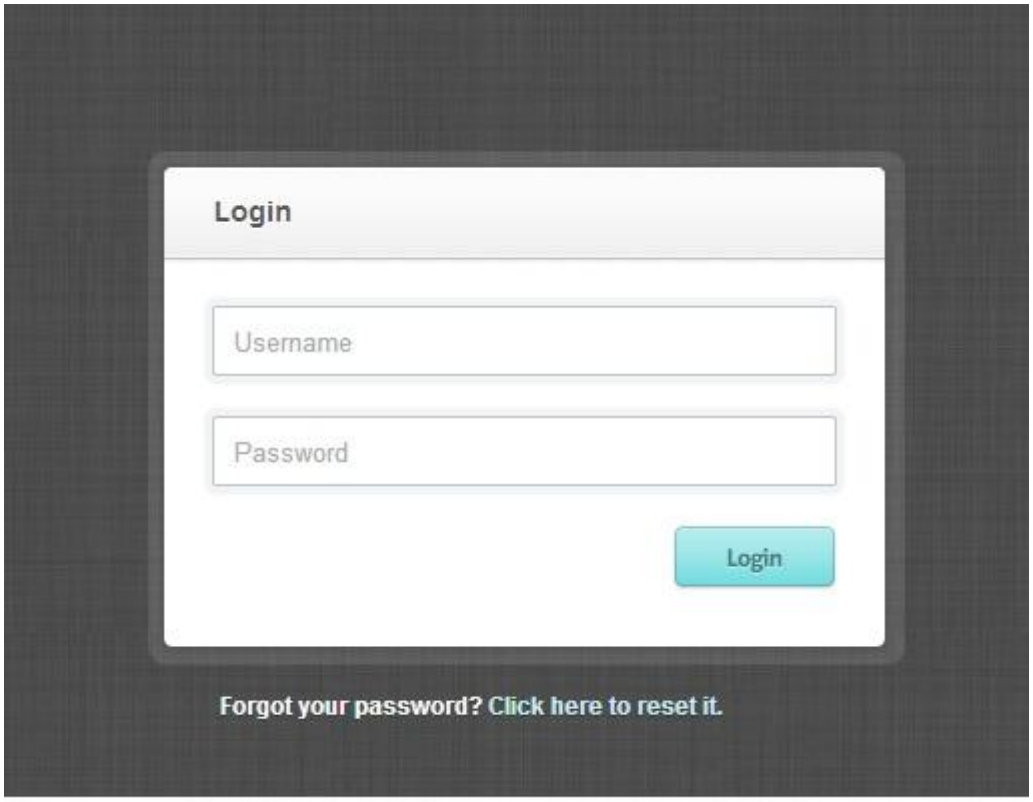
Gambar 5.9 Halaman Kontak

2. Tampilan Administrator

Tampilan administrator merupakan tampilan yang digunakan oleh admin untuk melakukan kontrol terhadap data-data yang akan ditampilkan pada Website Birmingham. Selain itu admin juga dapat melakukan kontrol terhadap data produk, data pemesanan, serta mencetak laporan stock barang dan laporan penjualan. Berikut menu-menu yang terdapat di dalam tampilan admin.

2.1. Log in Administrator

Untuk dapat masuk ke halaman utama administrator, terlebih dahulu harus melakukan proses login dengan memasukkan username dan password admin.



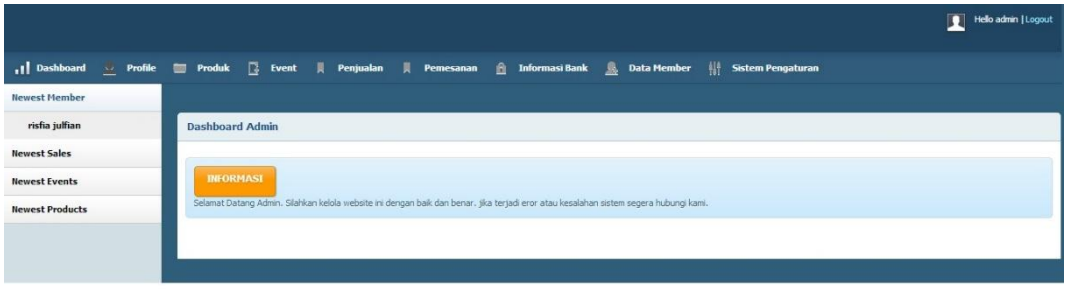
Gambar 5.10 Halaman Login Admin

2.2. Tampilan Utama Administrator

Pada halaman utama administrator ini terdapat sembilan menu diantaranya (Dasboard, Profile, Produk, Event, Penjualan, Pemesanan, Informasi Bank, Data Member, Sistem Pengaturan.

1. Dashboard Admin

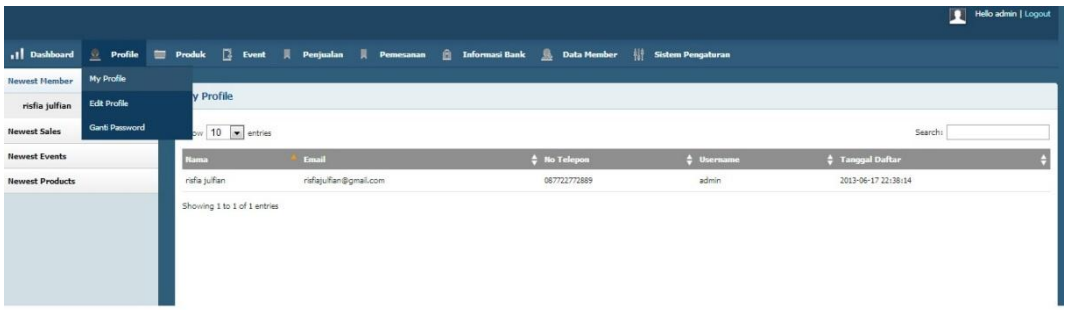
Tampilan Utama Pada halaman Dashboard ini terdapat informasi admin untuk menjalankan program ini dengan baik.



Gambar 5.11 Halaman Awal Admin

2. Profile Admin

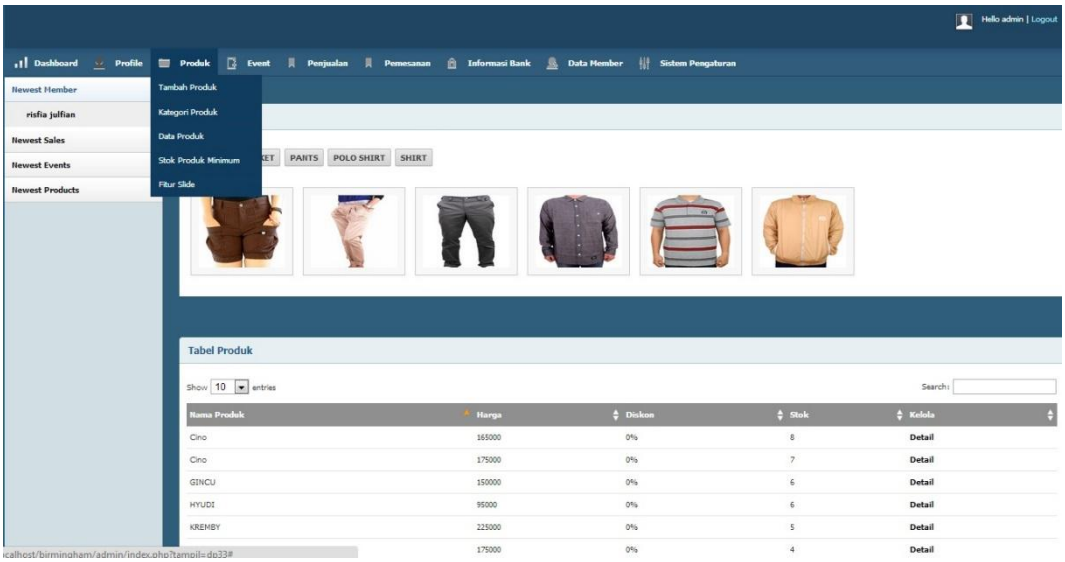
Menu ini berfungsi untuk melihat profile admin, edit profile, dan ganti password.



Gambar 5.12 Halaman Profile Admin

3. Menu Produk Admin

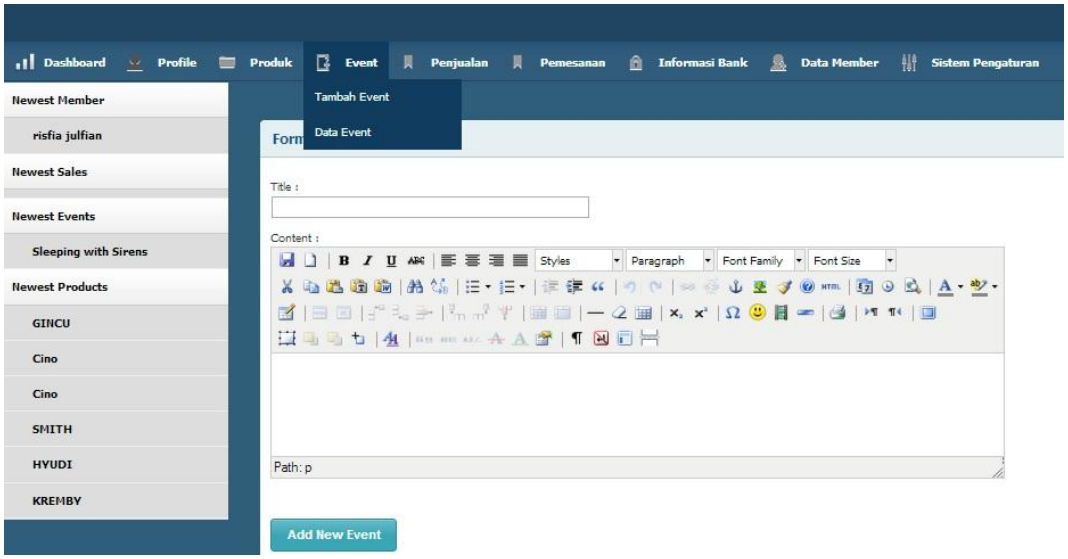
Menu ini berfungsi untuk melihat data produk, tambah produk, kategori produk, stok produk data minimum, fitur slide.



Gambar 5.13 Halaman Produk Admin

4. Menu Event

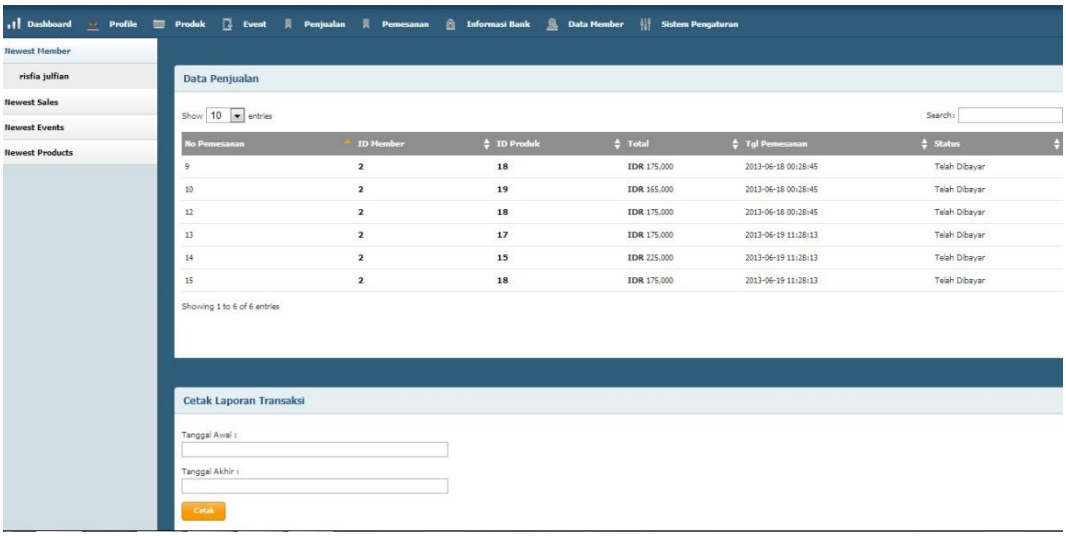
Menu ini berfungsi untuk melihat data Event, dan tambah Event.



Gambar 5.14 Halaman Event

5. Menu Penjualan

Menu ini berisi tentang data penjualan yang sudah di konfirmasi

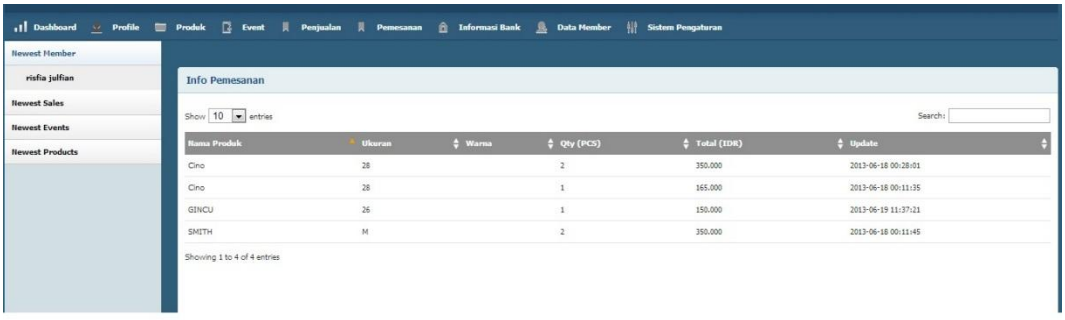


oleh konsumen dan form untuk mencetak Laporan penjualan.

Gambar 5.15 Halaman Penjualan

6. Pemesanan

Menu ini berisi tentang data pemesanan yang telah dipesan

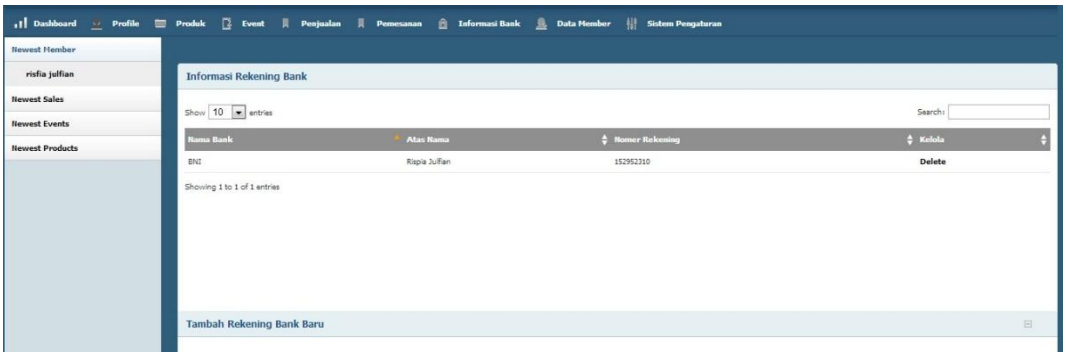


konsumen dan belum di konfirmasi.

Gambar 5.16 Halaman Pemesanan

7. Informasi Bank

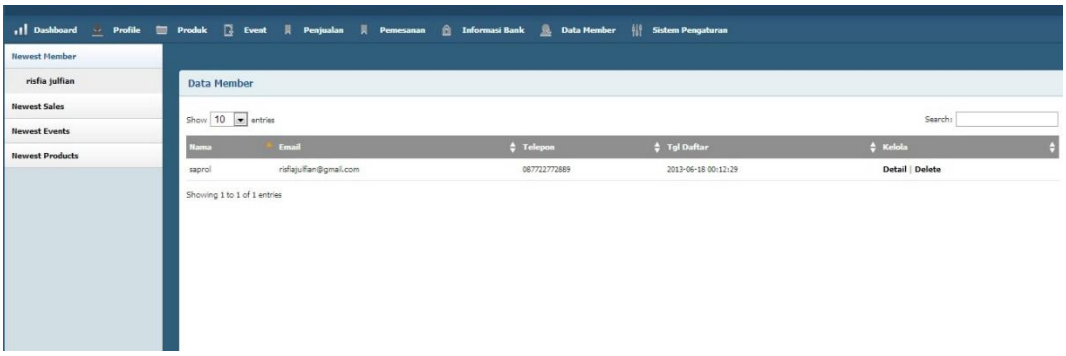
Menu ini berisi informasi rekening bank distro Birmingham dan untuk menambahkan rekening bank distro Birmingham lainnya.



Gambar 5.17 Halaman Informasi Bank

8. Data member

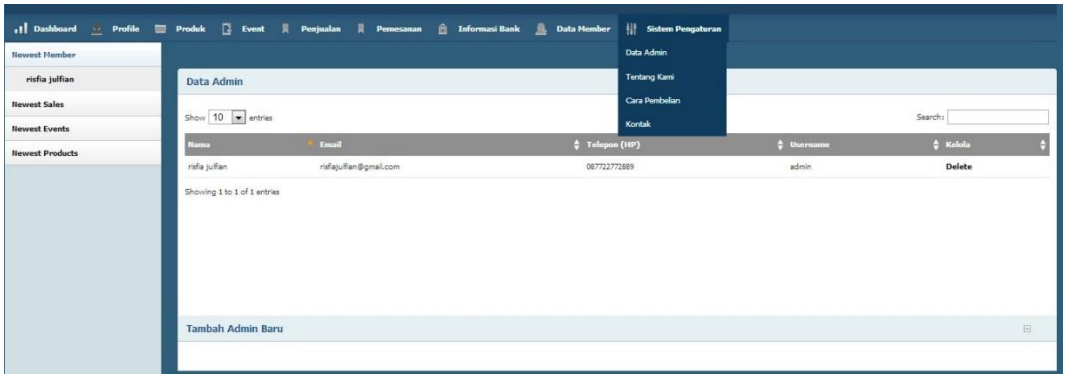
Menu ini berisi tentang data konsumen yang sudah menjadi member di website Birmingham.



Gambar 5.18 Halaman Data Member

9. Sistem Pengaturan

Menu ini berisi tentang pengaturan dari hak akses website Birmingham diantara lain seperti (Data Admin, tambah admin, menginput atau mengedit cara pembelian, menginput atau mengedit cara tentang kami, dan melihat kontak distro yang bekerja sama dengan disrto Birmingham.



Gambar 5.19 Halaman Sistem Pengaturan Admin

5.2 Pengujian Program

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak .Pengujian dilakukan untik 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 merepresentasikan kajian pokok dari spesifikasi, analis perancangan dan pengkodean dari perangkat lunak itu sendiri.

5.2.1. Rencana Pengujian

Pengujian perangkat lunak pada aplikasi Sistem Informasi Penjualan dan persediaan barang Berbasis Web pada Distro Birmingham yaitu dengan menggunakan pengujian black box. Pengujian black box berfokus pada persyaratan fungsional perangkat lunak yang dibuat. Berikut adalah rencana pengujian aplikasi aplikasi Sistem Informasi Penjualan dan persediaan barang Berbasis Web pada Distro Birmingham.

1.Pengujian unit yaitu pengujian ini difokuskan pada suatu unit dari program secara sendiri.

2.Pengujian penerimaan, yaitu pengujian yang meyakinkan bahwa sistem telah sesuai dengan kebutuhan organisasi.

5.2.2 Kasus dan Hasil Pengujian

Berikut adalah beberapa pengujian yang telah dilakukan, yaitu:

1. Login Admin

Tabel 5.3 Pengujian *login* Admin

Kasus dan Hasil Uji (Data normal)			
Data Masukan	Data diharapkan	Pengamatan	Kesimpulan
Username: admin Password: 772289	Muncul halaman utama admin.	Dapat masuk ke tampilan halaman utama admin.	[x] Diterima [ ] Ditolak
Kasus dan Hasil U ji (Data salah)			
Data kosong atau tidak terdaftar.	Tidak dapat login dan menampilkan pesan “Username atau Password” salah.	Admin tidak dapat melakukan proses login.	[ ] Diterima [x] Ditolak

2. Login Admin

Tabel 5.4 Pengujian *login* konsumen

Kasus Dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Login: Risfia julfian Password: 772289	Masuk ke halaman konsumen dengan benar	Dapat mengakses halaman konsumen sesuai dengan yang diharapkan	[x] Diterima [ ] Ditolak
Klik *Login*	Dapat melanjutkan ke form selanjutnya untuk konsumen	Tombol *login* / Enter dapat berfungsi sesuai dengan yang diharapkan	[x] Diterima [ ] Ditolak
Kasus Dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data *login* konsumen dengan password salah Login: Risfia julfian Password: abc	Tidak dapat *login* dan menampilkan pesan "*login* gagal, username dan password tidak benar"	Konsumen tidak dapat *login* dan memberikan pesan kode salah.	[x] Diterima [ ] Ditolak

3. Registrasi Member

Tabel 5.5. Pengujian Registrasi *User*

Kasus dan Hasil Uji (Data normal)			
Data Masukan	Data diharapkan	Pengamatan	Kesimpulan

Menginput nama *email* yang berbeda dengan *user* lain yang telah terdaftar sebagai *member*.	Berhasil menjadi *member*.	Berhasil menjadi *member* dan dapat melakukan proses *login*.	[x] Diterima [ ] Ditolak
Kasus dan Hasil U ji (Data salah)			
Menginput nama	Gagal disimpan dan	*User* tidak terdaftar	[ ] Diterima
email yang sama dengan *user* lain yang telah terdaftar sebagai *member*.	akan muncul peringatan : "Email sudah terdaftar".	menjadi *member* dan tidak dapat melakukan proses *login*.	[x] Ditolak

4. Inputan produk

Tabel 5.6. Pengujian Input Data Produk

Kasus dan Hasil Uji (Data normal)			
Data Masukan	Data diharapkan	Pengamatan	Kesimpulan
Nama Produk : Cino Ukuran :S Harga : 175.000 Stok : 4 Gambar : Cino.jpg	Produk tampil pada halaman utama *web*.	Data produk dapat masuk ke tampilan halaman utama *web*.	[x] Diterima [ ] Ditolak
Kasus dan Hasil U ji (Data salah)			
Data kosong atau tidak terdaftar.	Tidak menampilkan data barang.	Data barang yang dimasukkan gagal.	[ ] Diterima [x] Ditolak

5. Cetak laporan

Tabel 5.7. Pengujian Cetak Laporan

Kasus dan Hasil Uji (Data normal)			
Data Masukan	Data diharapkan	Pengamatan	Kesimpulan

Jenis Laporan: Penjualan Periode: 1 juni 2013 – 30 juni 2013	Muncul halaman laporan dalam bentuk file pdf.	Dapat masuk ke tampilan laporan dalam bentuk file pdf.	[x] Diterima [ ] Ditolak
Kasus dan Hasil U ji (Data salah)			
Data kosong atau tidak terdaftar	Tidak ada data penjualan.	*Admin* tidak dapat Mencetak laporan.	[ ] Diterima [x] Ditolak

6. Konfirmasi pembayaran

Tabel 5.8. Pengujian Konfirmasi Pembayaran

Kasus dan Hasil Uji (Data normal)			
Data Masukan	Data diharapkan	Pengamatan	Kesimpulan
Menginput data pada form konfirmasi pembayaran diisi lengkap.	Data berhasil dikonfirmasi.	Dapat masuk ke tampilan konfirmasi data dan dapat melakukan transaksi.	[x] Diterima [ ] Ditolak
Kasus dan Hasil U ji (Data salah)			
Data tidak lengkap.	Gagal disimpan dan akan muncul peringatan pending	Data belum dikonfirmasi dan akan memberikan pesan status“Pending”	[ ] Diterima [x] Ditolak

5.2.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian yang dilakukan diatas, yaitu pengujian data login konsumen, login admin,pengujian pendaftaran member, pengujian proses input produk,cetak laporan,konfirmasi pembayaran bahwa sistem yang diusulkan ini telah memenuhi standar pengujian. Dengan itu dapat diambil kesimpulan bahwa perangkat lunak bebas dari kesalahan sintaks dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.