BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Implementasi *prototype* sistem informasi Distro Heyshirt Clothing dilakukan menggunakan bahasa dan lingkungan pemrograman PHP dengan adobe dreamweaver 8 sebagai perangkat lunaknya, dan juga bahasa SQL sebagai basis data dengan MySQL sebagai perangkat lunaknya. Bahasa pemograman PHP tersebut dapat dijalankan pada berbagai platform sistem operasi dan perangkat keras, tetapi implementasi dan pengujian sepenuhnya hanya dilakukan di perangkat keras *PC (personal computer)* dengan system operasi Microsoft Windows Seven.

5.1.1 Batasan Implementasi

Dalam mengimplementasikan perangkat lunak, batasan sebagai pengendalian aplikasi ini ada beberapa hal yang perlu dibatasai dalam implementasi ini yaitu :

1. Perangkat lunak yang digunakan dalam pengimplementasian basis data ini adalah MySQL yang telah terkompilasi kedalam sebuah perangkat lunak bebas yaitu xampp.
2. Tampilan untuk *web based* menggunakan bahasa Indonesia dan Inggris.

5.1.2 Implementasi Perangkat Lunak

Untuk implementasi perangkat lunak ini menggunakan Apachefriends Xampp 1.6.8 WIN 32, Adobe dreamweaver 8 dan MySQL 1.4. Apachefriends Xampp 1.6.8 WIN 32 dipilih sebagai perangkat lunak untuk *webserver* karena faktor kecepatan, kinerja yang stabil, dan performansi PHP dan adobe dreamweaver 8 dipilih sebagai perangkat lunak pengembang karena menyediakan fasilitas yang memadai dan membuat perangkat lunak yang berbasis web. Sementara itu MySQL digunakan sebagai pengembang dalam pembuatan basis data.

5.1.3 Implementasi Perangkat Keras

Untuk dapat menjalankan *website* yang dirancang maka dibutuhkan suatu perangkat keras sebagai penunjangnya. Adapun perangkat kebutuhan perangkat keras yang dibutuhkan tersebut adalah sebagai berikut :

1. *Server* :
    a. *Processor* Minimal Pentium atau yang sekelasnya.
    b. Menggunakan minimal RAM 512 MB.
    c. *Harddisk* 500 Gb.
    d. *Keyboard*, *Monitor* dan *Mouse.*
2. *Client*
    a. *Processor* Minimal *Pentium* atau yang sekelasnya.
    b. Menggunakan minimal RAM 256 MB.

c. *Harddisk* 40 Gb.

d. *Keyboard*, *Monitor* dan *Mouse.*

5.1.4 Implementasi Basis Data (Sintaks SQL)

Implementasi basis data dilakukan dengan menggunakan bahasa SQL, dimana aplikasi pemrograman yang digunakan adalah MySQL, implementasi basis datanya dalam bahas SQL adalah sebagai berikut :

1. **Tabel Admin**

```
CREATE TABLE IF NOT EXISTS `admin` (
  `username` varchar(15) NOT NULL,
  `password` varchar(32) NOT NULL,
  `email` varchar(40) NOT NULL,
  `hak_akses` enum('admin','keuangan','marketing') NOT NULL,
  PRIMARY KEY (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. **Tabel Buku Tamu**

```
CREATE TABLE IF NOT EXISTS `buku_tamu` (
  `id_tamu` int(5) NOT NULL AUTO_INCREMENT,
  `username` varchar(15) NULL,
  `nama` varchar(30) NOT NULL,
  `komentar` longtext NOT NULL,
  `tgl_input` datetime NOT NULL,
  `email` varchar(40) NOT NULL,
  PRIMARY KEY (`id_tamu`)
 CONSTRAINT `buku_tamu_ibfk_1` FOREIGN KEY (`username`)
 REFERENCES `admin` (`username`) ON UPDATE CASCADE;
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

3. **Tabel Detail Pembelian**

```
CREATE TABLE IF NOT EXISTS `detail_pembelian` (
  `id_detailbeli` int(11) NOT NULL AUTO_INCREMENT,
  `id_pembelian` int(11) NOT NULL,
  `id_detailp` int(11) NOT NULL,
  `username` varchar(15) NULL,
  `hargabeli` int(20) NOT NULL,
  `qty` int(11) NOT NULL,
  `diskon` int(11) NOT NULL,
  PRIMARY KEY (`id_detailbeli`)
CONSTRAINT `detail_pembelian_ibfk_3` FOREIGN KEY (`id_pembelian`)
REFERENCES `pembelian` (`id_pembelian`) ON UPDATE CASCADE,
CONSTRAINT `detail_pembelian_ibfk_2` FOREIGN KEY (`id_detailp`)
REFERENCES `detail_produk` (`id_detailp`) ON UPDATE CASCADE,
```

```
CONSTRAINT `detail_pembelian_ibfk_1` FOREIGN KEY (`username`)

REFERENCES `admin` (`username`) ON UPDATE CASCADE;

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

4. **Tabel Detail Produk**

```
CREATE TABLE IF NOT EXISTS `detail_produk` (

  `id_detailp` int(11) NOT NULL AUTO_INCREMENT,

  `id_produk` int(8) NOT NULL,

  `id_ukuran` int(11) NOT NULL,

  `stok` int(11) NOT NULL,

  `berat` int(11) NOT NULL,

  PRIMARY KEY (`id_detailp`)

CONSTRAINT `detail_produk_ibfk_2` FOREIGN KEY (`id_produk`)
REFERENCES `produk` (`id_produk`) ON UPDATE CASCADE,

CONSTRAINT `detail_produk_ibfk_1` FOREIGN KEY (`id_ukuran`)
REFERENCES `ukuran` (`id_ukuran`) ON UPDATE CASCADE;

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

5. **Tabel Detail Retur**

```
CREATE TABLE IF NOT EXISTS `detail_retur` (

  `id_detail_retur` int(11) NOT NULL AUTO_INCREMENT,

  `qty` int(11) NOT NULL,

  `ket` varchar(255) NOT NULL,

  `id_retur` int(11) NOT NULL,

  `id_detailbeli` int(11) NOT NULL,

  PRIMARY KEY (`id_detail_retur`)

CONSTRAINT `detail_retur_ibfk_2` FOREIGN KEY (`id_detailbeli`)
REFERENCES `detail_pembelian` (`id_detailbeli`) ON UPDATE
```

```
CASCADE,

CONSTRAINT `detail_retur_ibfk_1` FOREIGN KEY (`id_retur`)
REFERENCES `retur` (`id_retur`) ON UPDATE CASCADE,

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

6. **Tabel Jasa Pengiriman**

```
CREATE TABLE IF NOT EXISTS `jasapengiriman` (

  `id_jasapengiriman` int(11) NOT NULL AUTO_INCREMENT,

  `nama_jasapengiriman` varchar(20) NOT NULL,

  `deskripsi_jasapengiriman` text NOT NULL,

  PRIMARY KEY (`id_jasapengiriman`),

  UNIQUE KEY `nama_jasapengiriman` (`nama_jasapengiriman`)

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

7. **Tabel Jenis Pengiriman**

```
CREATE TABLE IF NOT EXISTS `jenispengiriman` (

  `id_jenispengiriman` int(11) NOT NULL AUTO_INCREMENT,

  `id_jasapengiriman` int(11) NOT NULL,

  `nama_jenispengiriman` varchar(20) NOT NULL,

  `deskripsi_jenispengiriman` text NOT NULL,

  PRIMARY KEY (`id_jenispengiriman`)

CONSTRAINT `jenispengiriman_ibfk_1` FOREIGN KEY
(`id_jasapengiriman`) REFERENCES `jasapengiriman`
(`id_jasapengiriman`) ON UPDATE CASCADE;

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

8. **Tabel Kategori**

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` int(11) NOT NULL AUTO_INCREMENT,
  `kategori` varchar(15) NOT NULL,
  `h_dasar` int(11) DEFAULT NULL,
  PRIMARY KEY (`id_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

9. **Tabel Kurs**

```
CREATE TABLE IF NOT EXISTS `kurs` (
  `id_kurs` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(15) NULL,
  `nilai` int(11) NOT NULL,
  `tanggal` date NOT NULL,
  PRIMARY KEY (`id_kurs`)
CONSTRAINT `kurs_ibfk_1` FOREIGN KEY (`username`) REFERENCES
`admin` (`username`) ON UPDATE CASCADE;
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

10. **Tabel Member**

```
CREATE TABLE IF NOT EXISTS `member` (
  `id_member` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) NOT NULL,
  `alamat` text NOT NULL,
  `id_kabkota` int(10) NOT NULL,
  `kodepos` varchar(10) NOT NULL,
```

```
  `telepon` varchar(20) NOT NULL,

  `email` varchar(40) NOT NULL,

  `password` varchar(32) NOT NULL,

  `tgl_reg` datetime NOT NULL,

  `status` int(11) NOT NULL,

  PRIMARY KEY (`id_member`),

  UNIQUE KEY `email` (`email`)

CONSTRAINT `member_ibfk_1` FOREIGN KEY (`id_kabkota`) REFERENCES
`member` (`id_kabkota`) ON UPDATE CASCADE;

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

11. **Tabel Ongkos Kirim**

```
CREATE TABLE IF NOT EXISTS `ongkoskirim` (

  `id_ongkoskirim` int(11) NOT NULL AUTO_INCREMENT,

  `id_kabkota` int(11) NOT NULL,

  `id_jenispengiriman` int(11) NOT NULL,

  `harga_ongkoskirim` int(11) NOT NULL,

  PRIMARY KEY (`id_ongkoskirim`)

CONSTRAINT `ongkoskirim_ibfk_2` FOREIGN KEY
(`id_jenispengiriman`) REFERENCES `jenispengiriman`
(`id_jenispengiriman`) ON UPDATE CASCADE,

CONSTRAINT `ongkoskirim_ibfk_1` FOREIGN KEY (`id_kabkota`)
REFERENCES `t_kabkota` (`id_kabkota`) ON UPDATE CASCADE;

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

12. **Tabel Pembelian**

```
CREATE TABLE IF NOT EXISTS `pembelian` (
  `id_pembelian` int(11) NOT NULL AUTO_INCREMENT,
  `session_id` varchar(32) NOT NULL,
  `tgl_beli` datetime NOT NULL,
  `tgl_bayar` datetime NOT NULL,
  `status` enum('pesan','bayar','konfirmasi','kirim','selesai')
NOT NULL,
  `id_member` int(11) NOT NULL,
  `pembayaran` enum('rekening','paypal') NOT NULL,
  `kirim_nama` varchar(30) NOT NULL,
  `kirim_alamat` text NOT NULL,
  `kirim_telp` varchar(12) NOT NULL,
  `id_ongkoskirim` int(11) NOT NULL,
  `kirim_kota` int(11) NOT NULL,
  `kirim_kdpos` varchar(6) NOT NULL,
  `kirim_ongkos` int(10) NOT NULL,
  `kirim_resi` varchar(15) NOT NULL,
  `transfer_jumlah` int(10) NOT NULL,
  `transfer_tgl` date NOT NULL,
  `penerima_bank` varchar(15) NOT NULL,
  `penerima_rekening` varchar(30) NOT NULL,
  `totalbayar` float(12,2) NOT NULL,
  `id_kurs` int(11) NOT NULL,
  PRIMARY KEY (`id_pembelian`)
CONSTRAINT `pembelian_ibfk_3` FOREIGN KEY (`id_ongkoskirim`)
REFERENCES `ongkoskirim` (`id_ongkoskirim`) ON UPDATE CASCADE,
CONSTRAINT `pembelian_ibfk_2` FOREIGN KEY (`id_kurs`)
```

```
REFERENCES `kurs` (`id_kurs`) ON UPDATE CASCADE,

CONSTRAINT `pembelian_ibfk_1` FOREIGN KEY (`id_member`)
REFERENCES `member` (`id_member`) ON UPDATE CASCADE;

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

13. **Tabel Retur**

```
CREATE TABLE IF NOT EXISTS `retur` (

  `id_retur` int(11) NOT NULL AUTO_INCREMENT,

  `id_pembelian` int(10) unsigned NOT NULL,

  `id_member` int(11) NOT NULL,

  `tgl_retur` date DEFAULT NULL,

  `status_retur` varchar(11) NOT NULL,

  `tgl_kirim` date DEFAULT NULL,

  `resi_kirim` varchar(20) DEFAULT NULL,

  `resi_retur` int(11) DEFAULT NULL,

  `norek` varchar(20) DEFAULT NULL,

  `bank` varchar(20) DEFAULT NULL,

  `namarek` varchar(50) DEFAULT NULL,

  PRIMARY KEY (`id_retur`)

  CONSTRAINT `retur_ibfk_2` FOREIGN KEY (`id_member`) REFERENCES
  `member` (`id_member`) ON UPDATE CASCADE,

  CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`id_pembelian`)
  REFERENCES `pembelian` (`id_pembelian`) ON UPDATE CASCADE;

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

14. **Tabel Kabkota**

```
CREATE TABLE IF NOT EXISTS `t_kabkota` (

  `id_kabkota` int(10) NOT NULL AUTO_INCREMENT,

  `id_prov` int(10) DEFAULT NULL,

  `nama` varchar(30) DEFAULT NULL,

  PRIMARY KEY (`id_kabkota`)

CONSTRAINT `t_kabkota_ibfk_1` FOREIGN KEY (`id_prov`)
REFERENCES `member` (`id_kabkota`) ON UPDATE CASCADE;

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

15. **Tabel Provinsi**

```
CREATE TABLE IF NOT EXISTS `t_provinsi` (

  `id_prov` int(10) NOT NULL AUTO_INCREMENT,

  `nama` varchar(30) DEFAULT NULL,

  PRIMARY KEY (`id_prov`)

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

16. **Tabel ukuran**

```
CREATE TABLE IF NOT EXISTS `ukuran` (

  `id_ukuran` int(11) NOT NULL AUTO_INCREMENT,

  `nama_ukuran` varchar(10) NOT NULL,

  PRIMARY KEY (`id_ukuran`)

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

17. **Tabel produk**

```
CREATE TABLE IF NOT EXISTS `produk` (
  `id_produk` int(11) NOT NULL AUTO_INCREMENT,
  `id_kategori` int(11) NOT NULL,
  `produk` varchar(30) NOT NULL,
  `image` varchar(100) DEFAULT NULL,
  `harga` int(20) NOT NULL,
  `diskon` int(3) unsigned NOT NULL DEFAULT '0',
  `tgl_input` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_produk`),
  UNIQUE KEY `produk` (`produk`),
  KEY `id_kategori` (`id_kategori`)
  CONSTRAINT `produk_ibfk_1` FOREIGN KEY (`id_kategori`)
  REFERENCES `kategori` (`id_kategori`) ON UPDATE CASCADE;
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.5 Implementasi Antar Muka

Dalam bahasa pemrograman 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 Kebutuhan Pengguna dan Petugas Distro Heyshirt Clothing.

5.1.5.1 Implementasi Halaman Utama

Implementasi halaman yang terdapat pada pengguna yang dibuat bentuk *File* program dengan berekstensi (.php) dapat dilihat pada tabel 5.1.

Tabel 5.1 Implementasi antar muka

Menu	Deskripsi	Nama File
Login admin	Digunakan sebagai halaman login admin	login.php
Member	Digunakan untuk melihat, menambah, dan merubah data member	member.php
Kota	Digunakan untuk melihat, menambah, dan merubah data kota	kota. php
Provinsi	Digunakan untuk melihat, menambah, dan merubah data provinsi	provinsi. php
Produk	Digunakan untuk melihat, menambah, dan merubah data produk	produk. php
Ongkos kirim	Digunakan untuk melihat, menambah, dan merubah data ongkos kirim	ongkoskirim. php
Kategori	Digunakan untuk melihat, menambah, dan merubah data kategori produk	kategori.php
Retur	Digunakan untuk melihat data retur dan mengubah status retur	admin_retur.php
Laporan produk	Digunakan untuk melihat laporan produk	laporan.php

Laporan penjualan	Digunakan untuk melihat laporan penjualan	laporan.php
Laporan retur barang	Digunakan untuk melihat laporan retur	laporan.php
Halaman Depan	Digunakan untuk menampilkan halaman utama	lib_func.php
Registrasi *member*	Digunakan untuk melakukan pendaftaran *member*	register.php
About	Digunakan untuk menampilkan halaman about	about.php
Hubungi kami	Digunakan untuk menampilkan halaman hubungi kami	kontak.php
Keranjang belanja	Digunakan untuk melihat, merubah, menghapus detail transaksi yang akan dilakukan	pemesanan.php

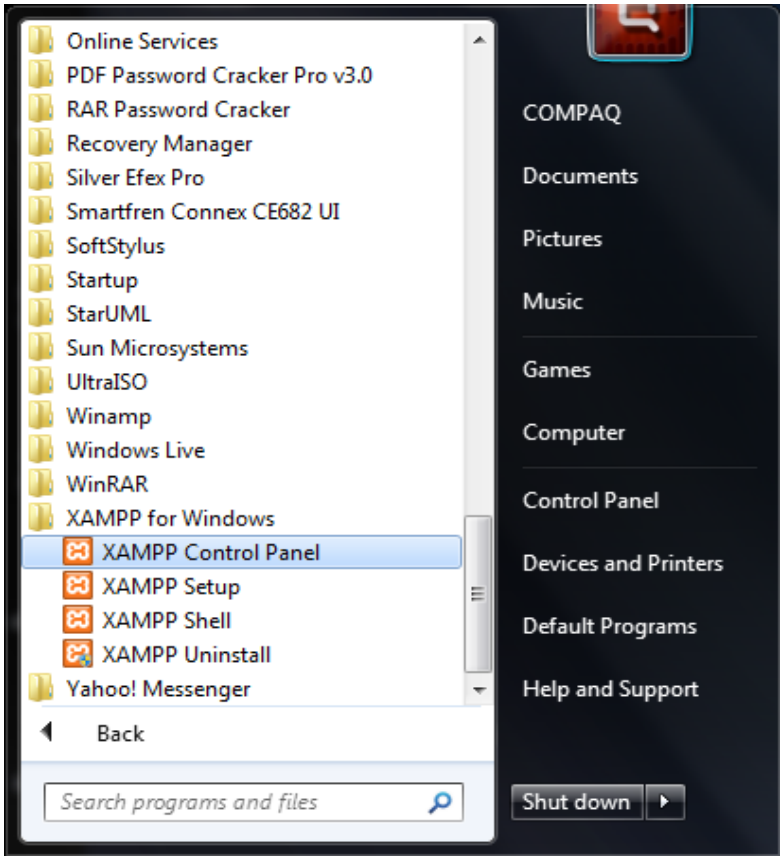
5.1.5.2 Implementasi Halaman *Admin* (Petugas)

Implementasi halaman yang terdapat *admin* yang dibuat bentuk *File* program dengan berekstensi (.php).

5.1.6 Implementasi Instalasi Program

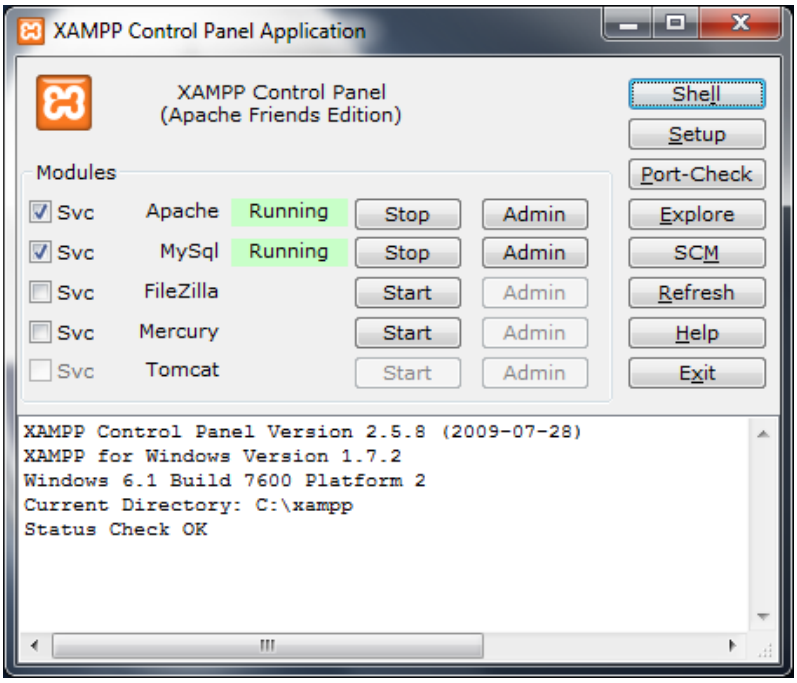
Berikut merupakan implementasi instalasi program bagaimana cara menggunakan program Sistem Informasi Penjualan Berbasis Web pada Distro Heyshirt Clothing. Langkah-langkahnya adalah sebagai berikut :

1. Klik menu start kemudian XAMPP for windows, XAMPP *Control PanelApplication.*



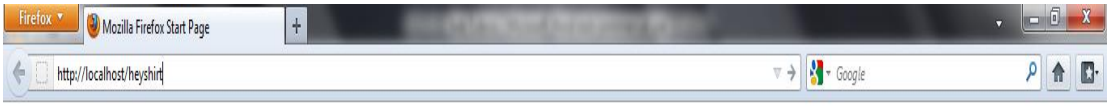
Gambar 5.1 Mengaktifkan XAMPP *Control Panel Application*

2. Aktifkan XAMPP *Control Panel Application* seperti pada gambar di bawah.



Gambar 5.2 XAMPP *Control Panel Application.*

3. Buka Mozilla Firefox, lalu ketikan alamat *http://localhost/heyshirt*



Gambar 5.3 Tampilan URL.

5.1.7 Implementasi Hosting Program

Pada implementasi ini penulis mencoba menjelaskan bagaimana caranya menghosting sebuah aplikasi berbasis web yang sudah dirancang sehingga dapat diakses melalui internet. Dimana aplikasi yang penulis rancang telah dihosting dengan nama domain aktif http://heyshirtclothing.com melalui situs resmi yang memberikan pelayanan jasa hosting yaitu hostinggokil.com.

Adapun langkah-langkah yang penulis lakukan dalam menghosting aplikasi yang

telah dirancang adalah sebagai berikut :

1. Buka situs hosting gokil di http://hostinggokil.com
2. Pilih salah satu paket yang tersedia (paket yang digunakan penulis adalah paket iseng)
3. Klik order disalah satu paket yang ada
4. Masukan daftar domain yang ingin anda gunakan (yang penulis pesan heyshirtclothing.com) sesuai dengan kebutuhan, jika ingin memesan space hosting saja bisa pilih cekbox yang ke- 3
5. Pilih berapa lama akan kontrak di server kami (penulis memilih per 1 tahun saja).
6. Tampilan bahwa order anda sudah berada pada keranjang belanja
7. Mengisi data diri untuk kelegalitasan sebuah situs
8. Hasil tagihan anda
9. Lakukan pembayaran ke salah satu account bank kami, kemudian lakukan konfirmasi di
10. Jawaban konfirmasi akan dikirim lewat email yang telah didaftarkan pada poin 7
11. Melakukan login pada situs yang didaftarkan domainnya dengan ditambah slash cpanel dibelakangnya (situs yang penulis gunakan http://heyshirtclothing/cpanel) untuk meng upload file dan pembuatan database.

5.1.8 Penggunaan Program

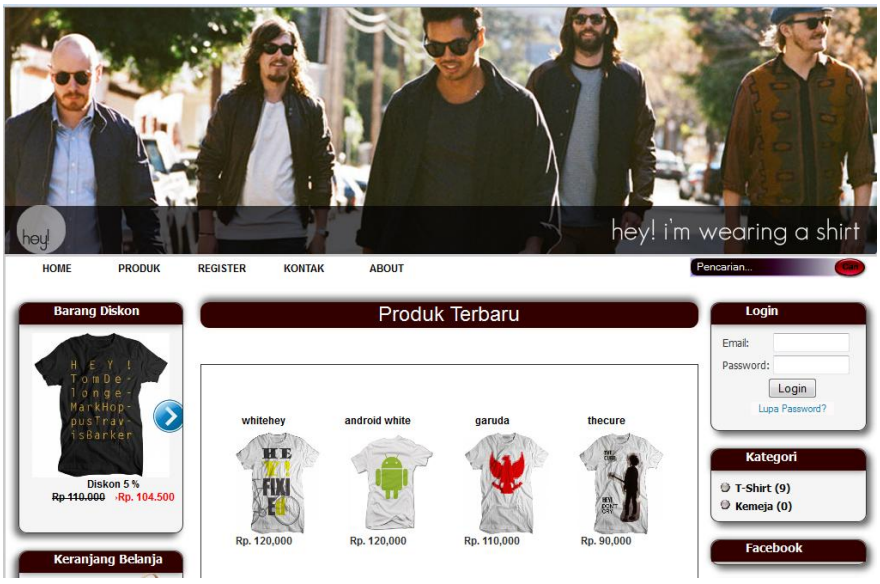
Penggunaan program bertujuan untuk memberikan informasi tentang bagaimana tata cara dalam menjalankan dan menggunakan program Sistem Penjualan dan Promosi Berbasis Web pada Distro Heyshirt Clothing bagi pengguna maupun *admin*.

5.1.8.1 Penggunaan Program Anggota

Berikut adalah petunjuk penggunaan program Sistem Informasi Penjualan dan Promosi Online Distro Heyshirt Clothing bagi anggota

1. Home

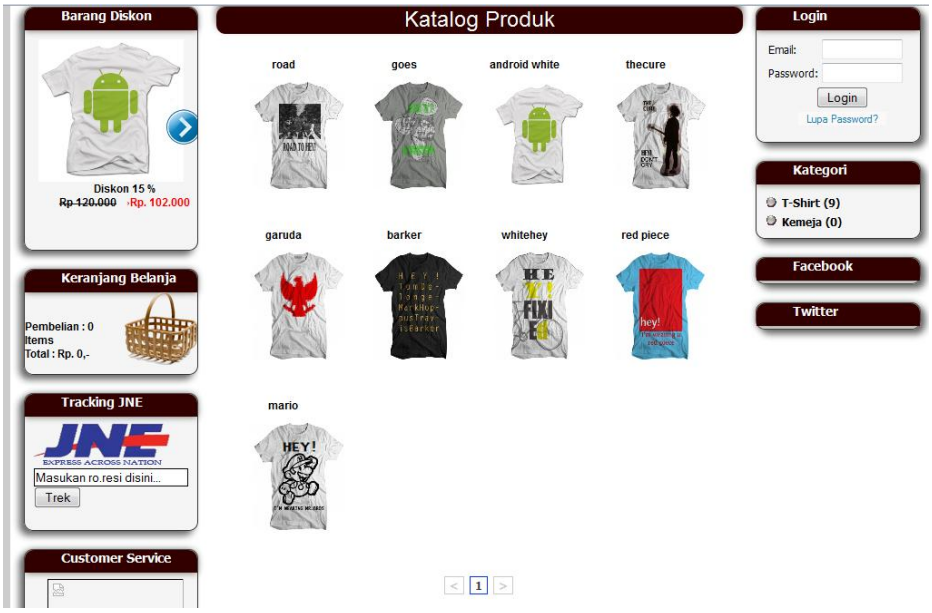
Merupakan tampilan utama pada *website* dimana saat pertama kali pengguna membuka alamat *website* heyshirt dan juga tampilan dari *login* anggota.



Gambar 5.4 Tampilan Halaman Utama *website* Distro Heyshirt Clothing.

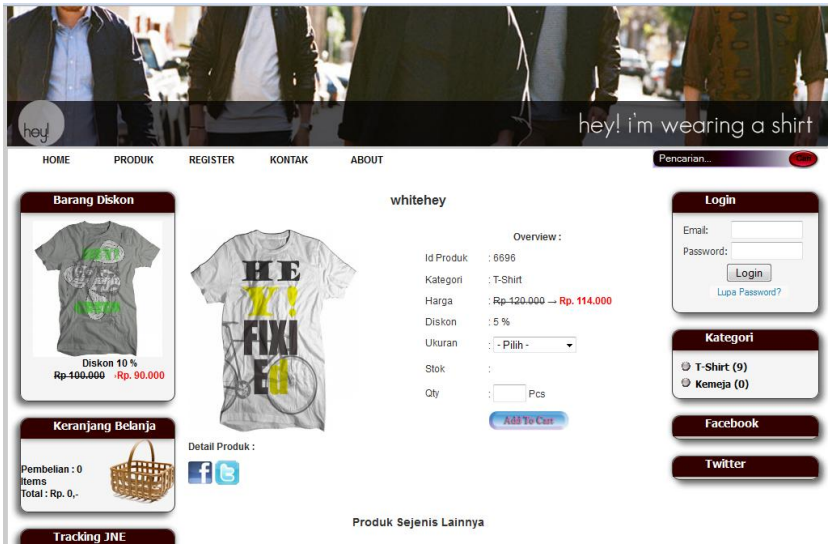
2. Produk

Tampilan *website* yang menampilkan informasi tentang dafar produk, kategori dan detail produk yang ditawarkan pada Distro Heyshirt Clothing.



Gambar 5.5 Tampilan Daftar Produk dan Menu Kategori Produk pada *website* Distro Heyshirt Clothing.

Apabila anggota mengklik salah satu menu kategori produk yang tertera di layar, maka akan tampil form daftar produk berdasarkan kategori yang di pilih saja seperti gambar dibawah ini :

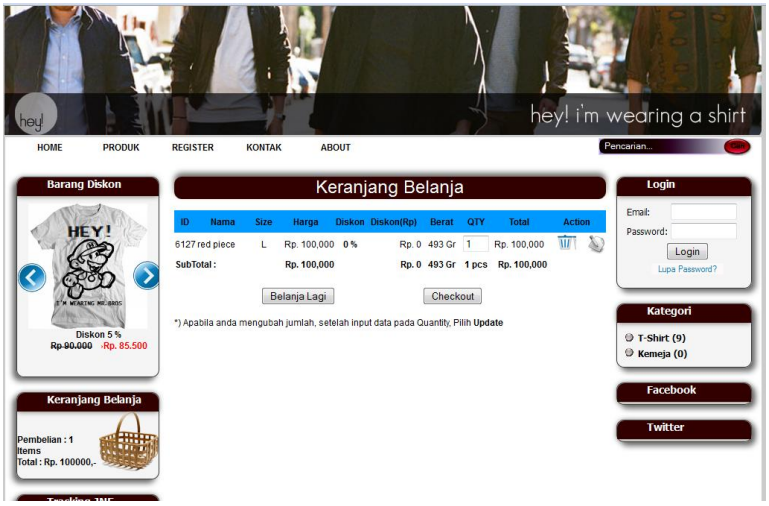


Gambar 5.6 Tampilan Daftar Produk Berdasarkan Kategori pada *website* DistroHeyshirt Clothing.

Apabila anggota ingin membeli produk tersebut tinggal memilih ukuran yang telah tersedia di kolom ukuran dan selanjutnya tinggal klik pada bagian *add to cart*.

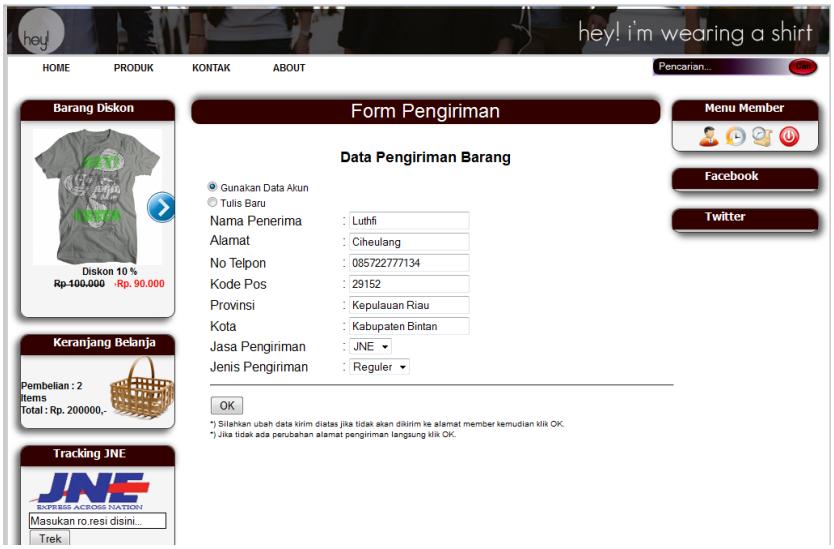
3. Keranjang Belanja (Pesanan)

Menu ini berperan untuk menampilkan informasi daftar pesanan yang telah dilakukan oleh anggota sebelum dilakukan proses konfirmasi pemesanan pada menu konfirmasi. Berikut merupakan tampilan informasi daftar produk yang telah dipesan oleh anggota.



Gambar 5.7 Tampilan Daftar Detail Pemesanan Produk pada *website* Distro Heyshirt Clothing.

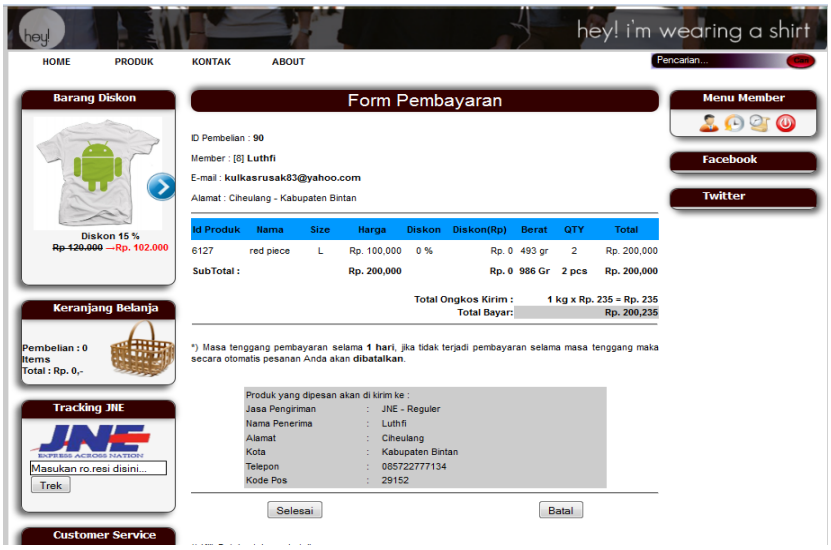
Apabila anggota mengklik tombol *Checkout* maka akan ditampilkan informasi form pengiriman beserta alamat lengkap data pribadi anggota yang selanjutnya akan masuk ke database admin untuk melakukan konfirmasi kepada anggota yang akan membeli produk tersebut.



Gambar 5.8 Tampilan Form Pengiriman Produk pada *website* Distro Heyshirt Clothing.

4. Konfirmasi Pembayaran

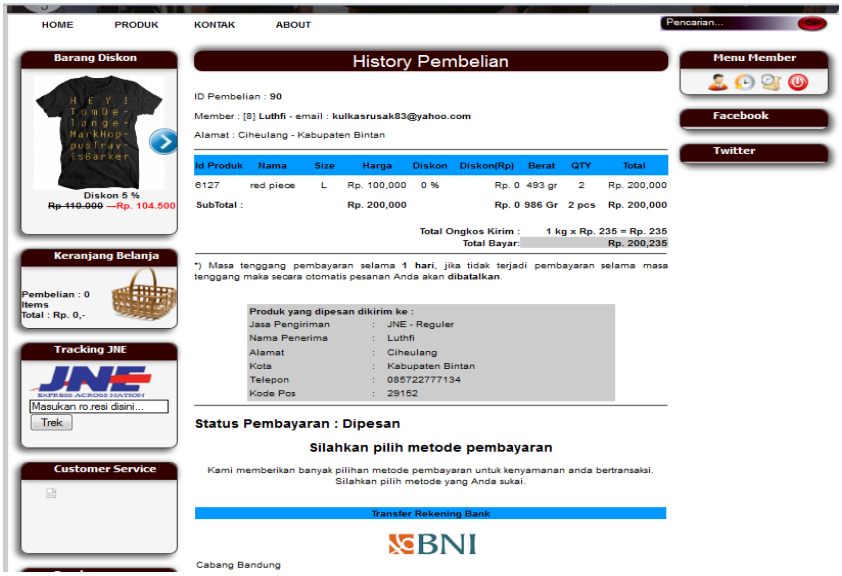
Menu ini bertujuan untuk menampilkan form konfirmasi yang telah ditampilkan setelah proses pemesanan yang dilakukan oleh anggota dan kemudian setelah proses transfer pembayaran.



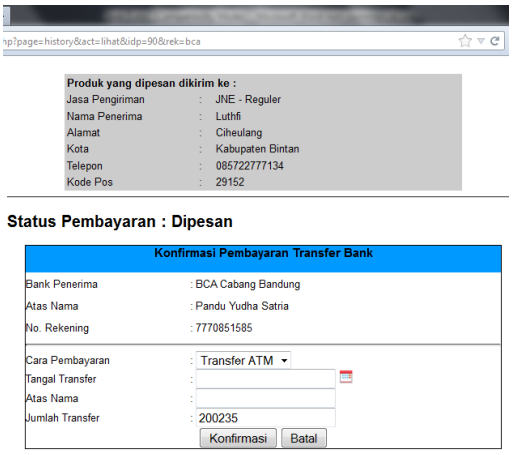
Gambar 5.9 Tampilan Form Konfirmasi Pembayaran pada *website* Distro Heyshirt Clothing.

5. History Pembayaran

Menu ini bertujuan untuk menampilkan form untuk mengetahui history pembelian produk dan melihat status pembelian produk sekaligus konfirmasi pembayaran melalui cara transfer bank yang telah disediakan oleh admin.



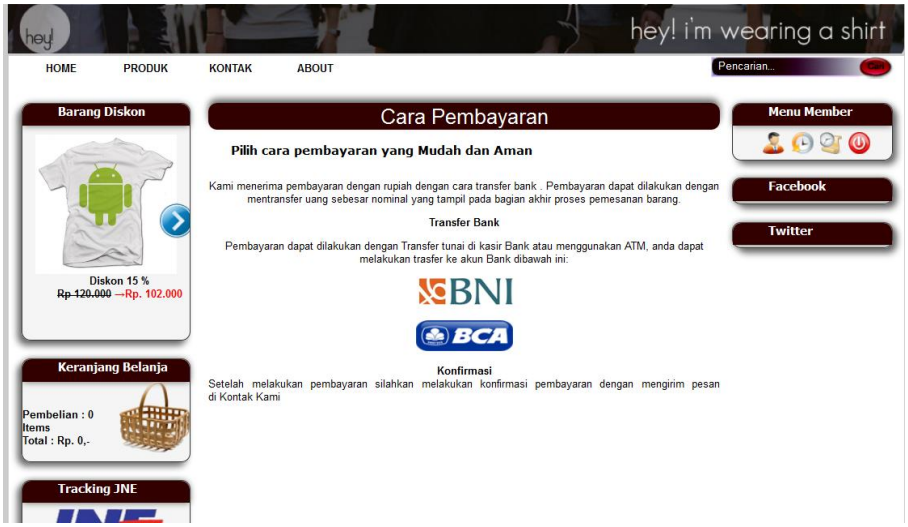
Gambar 5.10 Tampilan Form History Pembelian & Status pembayaran pada *website* Distro Heyshirt Clothing.



Gambar 5.11 Tampilan Form Konfirmasi Pembayaran Transfer Bank pada *website* Distro Heyshirt Clothing.

6. Cara pembayaran

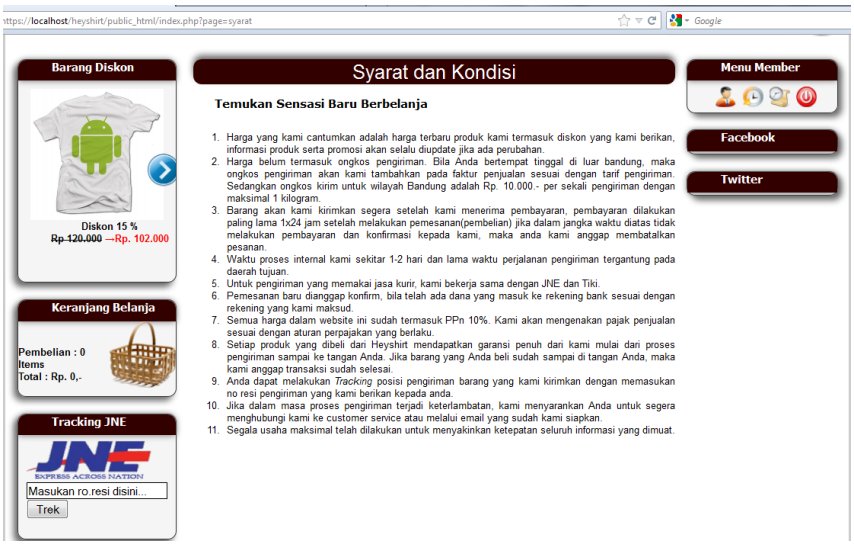
Menu ini menampilkan informasi bagaimana tata cara ataupun prosedur dalam melakukan pemesanan hingga proses konfirmasi pembayaran.



Gambar 5.12 Tampilan Form Cara pembayaran pada *website* Distro Heyshirt Clothing.

7. Syarat dan Kondisi

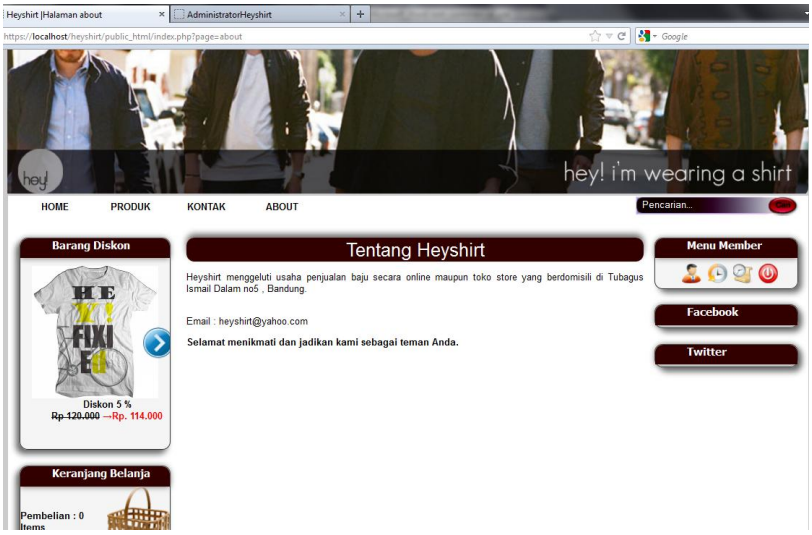
Menu ini menampilkan informasi bagaimana syarat – syarat dan kondisi yang berlaku untuk melakukan pemesanan sekaligus penjelasan pembayaran atau sejenisnya.



Gambar 5.13 Tampilan menu syarat dan kondisi pada *website* Distro Heyshirt Clothing.

8. Tentang Perusahaan/*About*

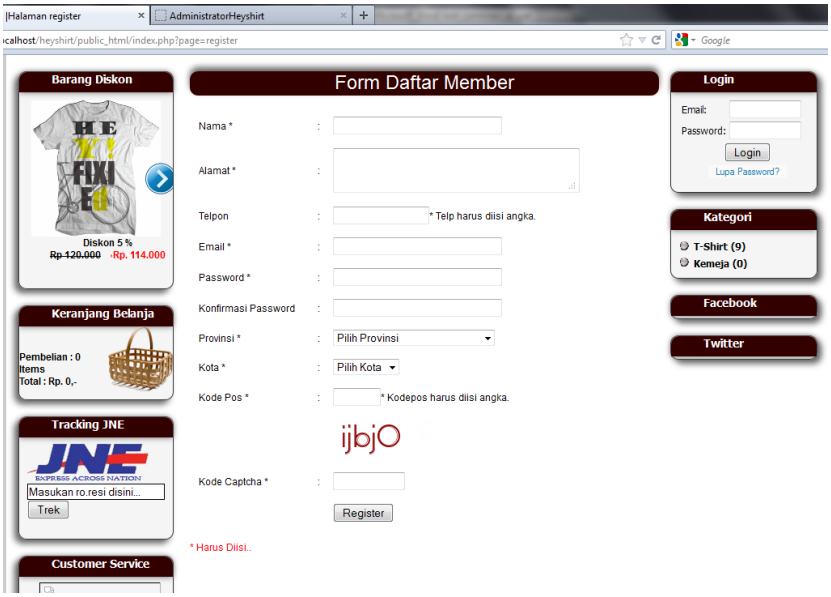
Menu ini menampilkan informasi – informasi sedikit tentang perusahaan heyshirt yang sedang berjalan.



Gambar 5.14 Menu Tampilan about pada *website* Distro Heyshirt Clothing.

9. Register

Menu ini menampilkan form untuk melakukan registrasi menjadi member atau anggota pada heyshirt clothing yang akan nantinya melakukan proses pembelian pada heyshirt clothing tersebut.



Gambar 5.15 Menu Tampilan *Register* pada *website* Distro Heyshirt Clothing.

5.1.8.2 Penggunaan Program Admin

Berikut adalah petunjuk penggunaan program Sistem Informasi Penjualan *Online* pada Distro Heyshirt Clothing bagi *Admin*.

1. Form Login Akun Admin

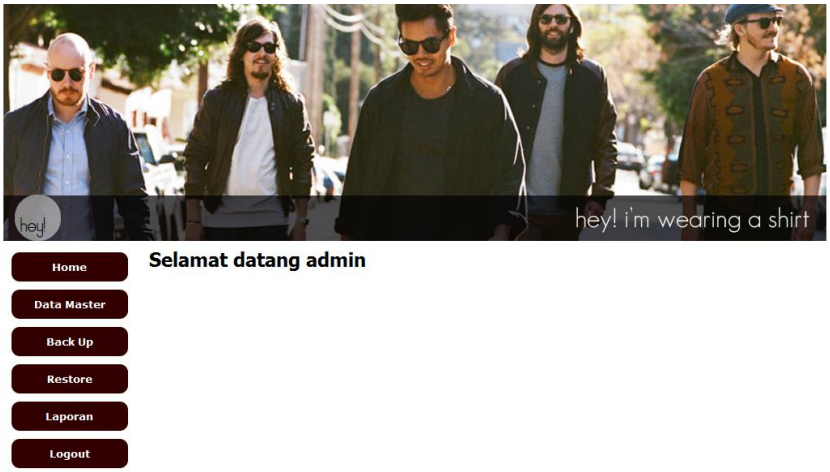
Tampilan *website* dimana petugas harus mengisikan *username* dan *password* untuk dapat masuk ke menu utama *admin*. Menu admin terbagi 3 yaitu menu admin utama, Keuangan, Marketing.



Gambar 5.16 Tampilan Form *Login* Akun *admin* pada *website* Distro Heyshirt Clothing.

2. Halaman Utama *Admin*/ *Home*

Merupakan tampilan utama *admin* dimana terdapat menu Home, *Data Master*, *Backup*, *Restore*, Laporan dan *logout*.

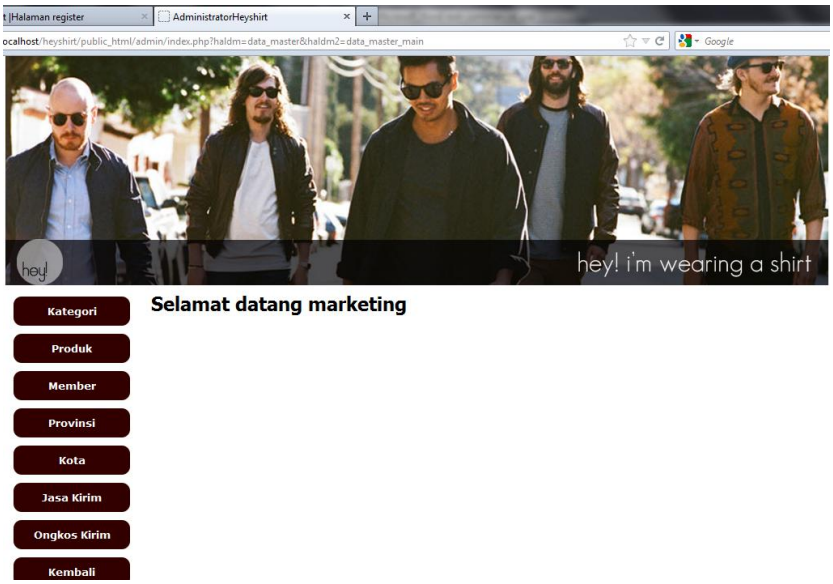


Gambar 5.17 Tampilan Halaman Utama admin utama pada *website* Distro Heyshirt Clothing.

.

3.Halaman Utama *Admin* Marketing

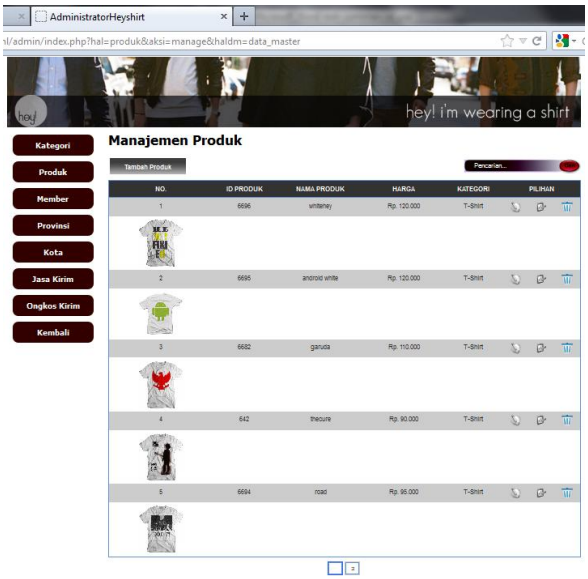
Menu utama admin marketing adalah untuk menangani segala kegiatan yang berhubungan dengan persediaan stok barang/produk dan berhubungan dengan data – data member yang telah melakukan registrasi yang mencakup antara lain kategori, produk, member , provinsi , kota, jasa kirim , ongkos kirim dan *logout*.



Gambar 5.18 Tampilan Halaman Utama *admin marketing* pada *website* Distro Heyshirt Clothing.

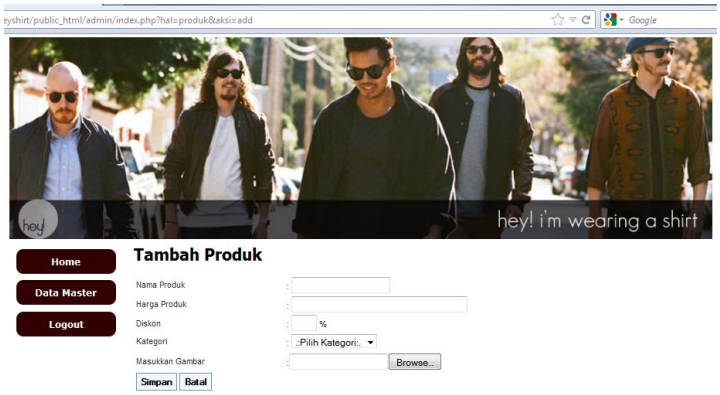
4. Menu produk

Menu produk pada halaman website ini bertujuan menampilkan daftar produk dan kategori produk, tambah produk dan mengecek persediaan stok produk yang masih tersisa.



Gambar 5.19 Tampilan Daftar produk pada akun *admin marketing* pada *website* Distro Heyshirt Clothing.

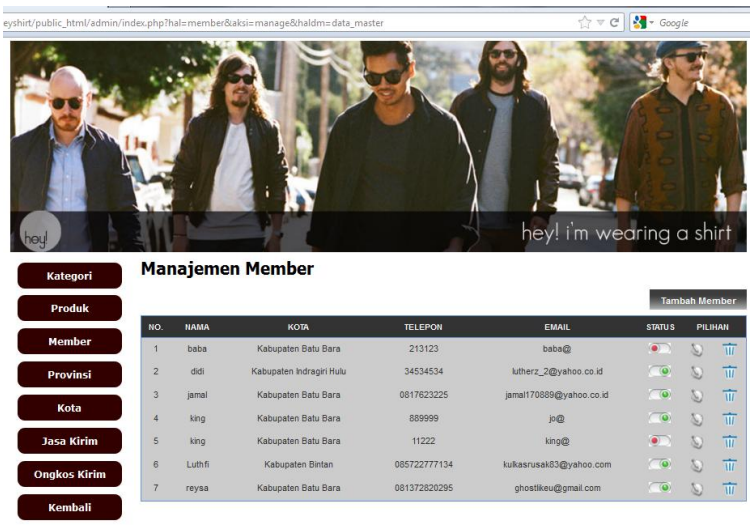
Pada halaman daftar produk diatas, apabila admin mengklik *link* untuk tambah produk maka sistem akan menampilkan halaman sebagai berikut.



Gambar 5.20 Tampilan tambah produk pada akun *admin marketing* pada *website* Distro Heyshirt Clothing.

5. Manajemen Member

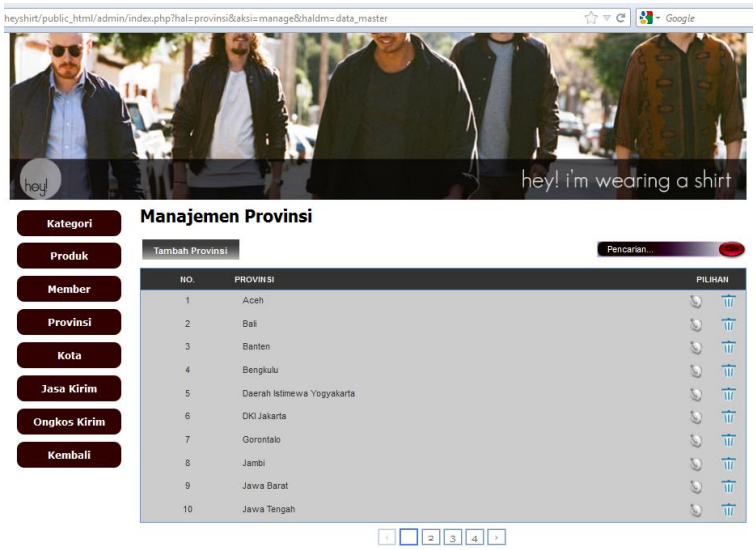
Menu manajemen member pada halaman website ini bertujuan menampilkan daftar atau data pribadi member dan akun member yang telah aktif dan belum aktif.



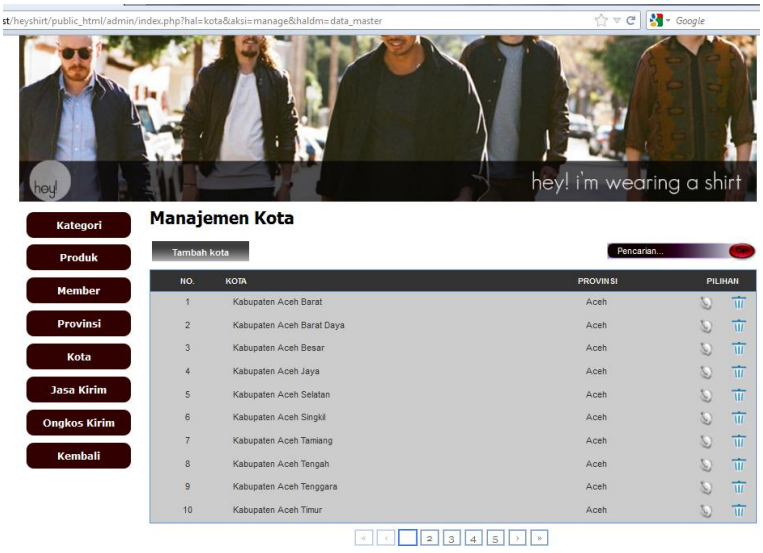
Gambar 5.21 Tampilan manajemen member pada akun *admin marketing* pada *website* Distro Heyshirt Clothing.

6. Manajemen Provinsi dan Kota

Menu provinsi dan kota pada halaman website ini bertujuan menampilkan provinsi dan kota tujuan pembeli yang akan membeli produk pada toko heyshirt clothing ini dan sekaligus menambah data provinsi dan kota untuk kelengkapan pengiriman pada daerah tertentu.



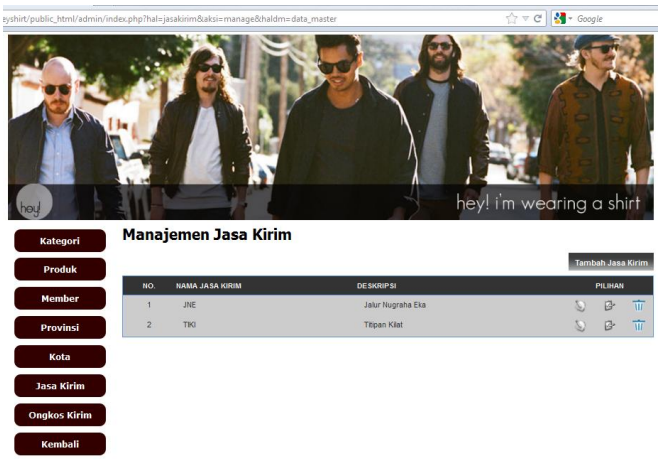
Gambar 5.22 Tampilan manajemen provinsi pada akun *admin marketing* pada *website* Distro Heyshirt Clothing.



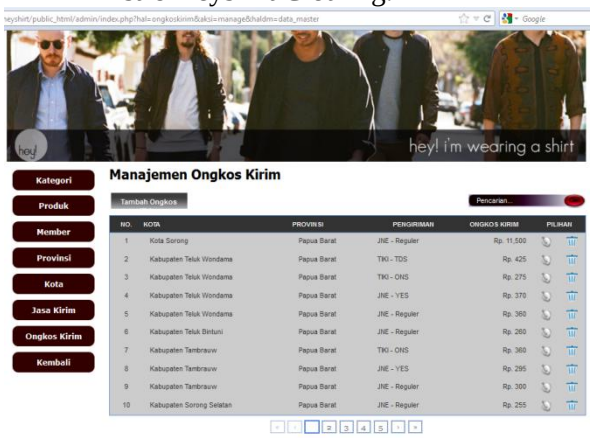
Gambar 5.23 Tampilan manajemen kota pada akun *admin marketing* pada *website* Distro Heyshirt Clothing.

6. Manajemen Jasa Kirim dan Ongkos Kirim

Menu manajemen jasa kirim dan ongkos kirim pada halaman website ini bertujuan menampilkan jasa apa saja yang di pakai untuk pengiriman produk yang akan di pakai oleh *admin* untuk mengirim barang, sekaligus ongkos kirim yang akan dikenakan pembeli sesuai daerah dan tempat dimana pembeli berada yang langsung di kalkulasikan atau ditambah ke pembayaran produk yang dibeli.



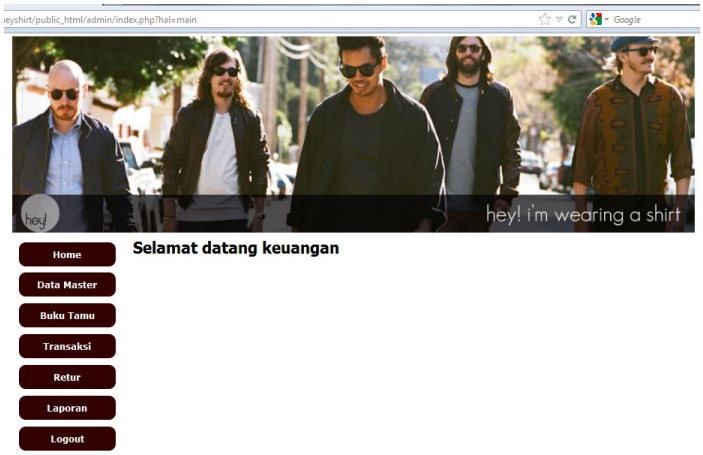
Gambar 5.24 Tampilan manajemen Jasa kirim pada akun *admin marketing* pada *website* Distro Heyshirt Clothing.



Gambar 5.25 Tampilan manajemen Ongkos kirim pada akun *admin marketing* pada *website* Distro Heyshirt Clothing.

7. Halaman Utama *Admin* Keuangan

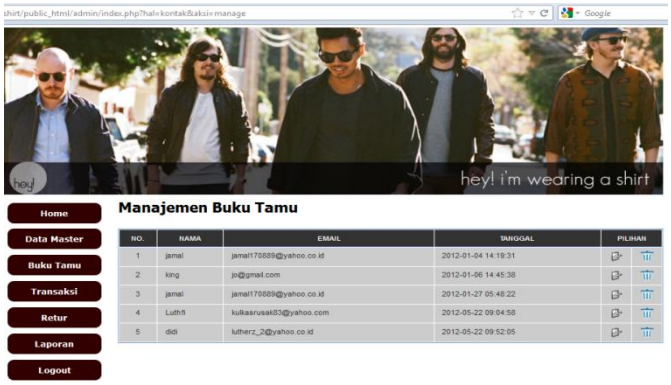
Menu utama admin keuangan adalah untuk menangani segala kegiatan yang berhubungan dengan transaksi, keuangan, buku tamu, retur, dan laporan.



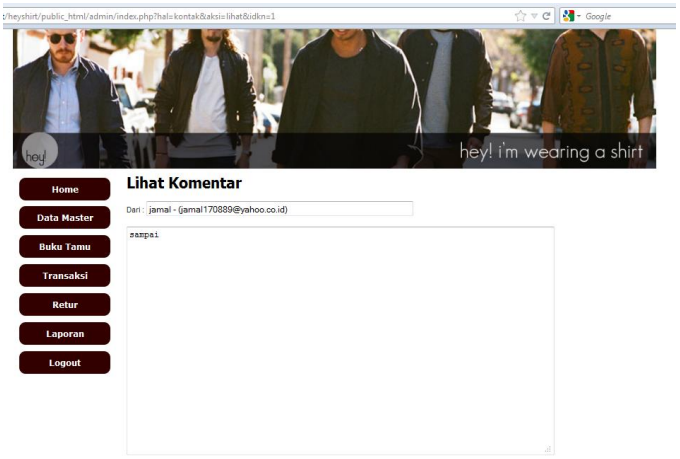
Gambar 5.26 Tampilan *admin* keuangan pada *website* Distro Heyshirt Clothing.

7. Manajemen Buku Tamu

Menu manajemen buku tamu pada halaman website ini bertujuan menampilkan testimonial anggota yang membeli barang setalah anggota melakukan pembayaran.



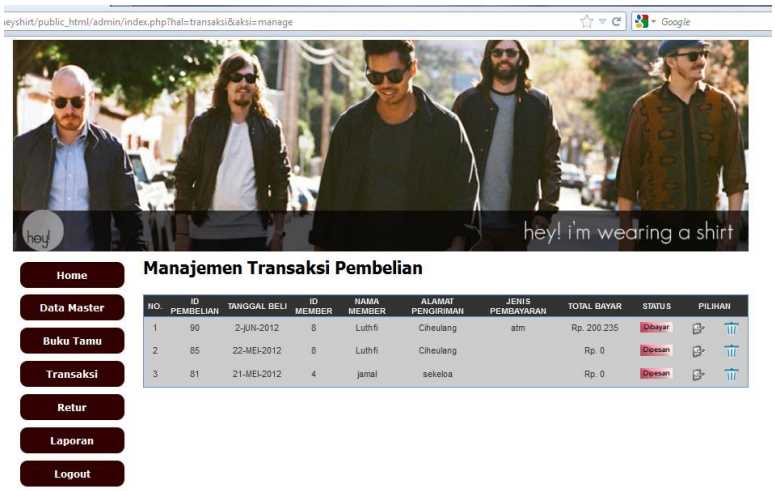
Gambar 5.27 Tampilan manajemen buku tamu pada akun *admin keuangan* pada *website* Distro Heyshirt Clothing.



Gambar 5.28 Tampilan komentar buku tamu pada akun *admin keuangan* pada *website* Distro Heyshirt Clothing.

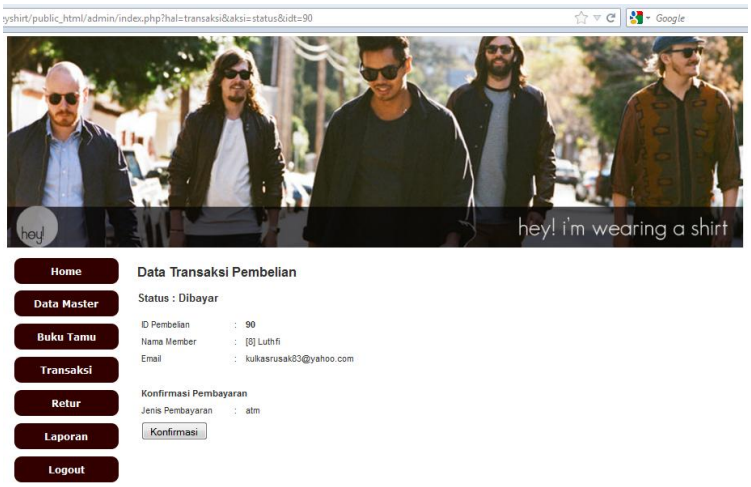
8. Manajemen Transaksi pembelian

Menu manajemen transaksi pembelian pada halaman website ini bertujuan menampilkan data pemesanan, pembelian produk dari anggota yang membeli produk untuk dilakukan konfirmasi selanjutnya oleh *admin*.



Gambar 5.29 Tampilan manajemen transaksi pembelian pada akun *admin keuangan* pada *website* Distro Heyshirt Clothing.

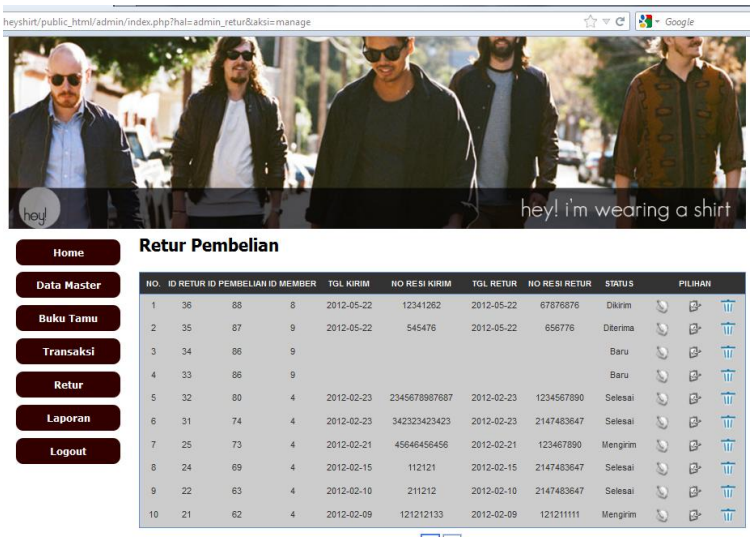
Setelah admin melihat status yang telah dibayar oleh anggota, *admin* segera melakukan konfirmasi untuk memberikan konfirmasi kepada anggota bahwa barang siap untuk dikirim seperti contoh gambar dibawah ini :



Gambar 5.30 Tampilan status pembelian pada akun *admin keuangan* pada *website* Distro Heyshirt Clothing.

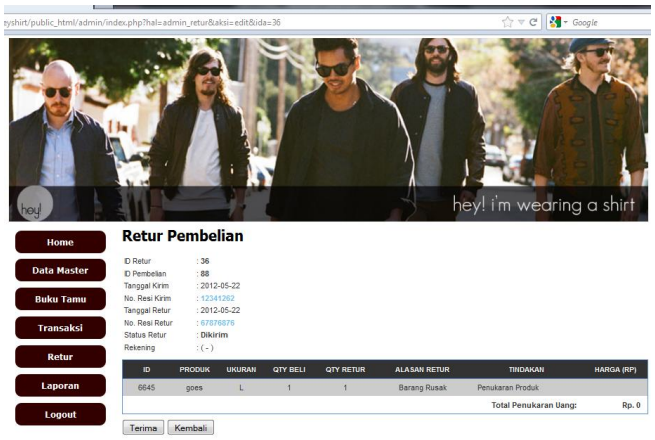
9. Manajemen Retur

Menu manajemen retur pada halaman website ini bertujuan menampilkan data retur atau data pengembalian produk dari anggota yang mendapatkan barang atau produk tidak sesuai dengan apa yg dipesan sebelumnya.



Gambar 5.31 Tampilan retur pembelian pada akun *admin keuangan* pada *website* Distro Heyshirt Clothing.

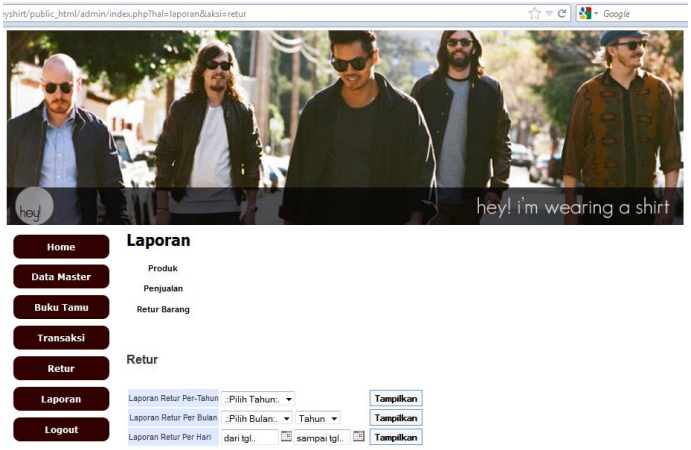
Dibawah ini tampilan menu retur pembelian setelah admin melakukan pengecekan atau konfirmasi retur pembelian anggota :



Gambar 5.32 Tampilan retur pembelian pada akun *admin keuangan* pada *website* Distro Heyshirt Clothing.

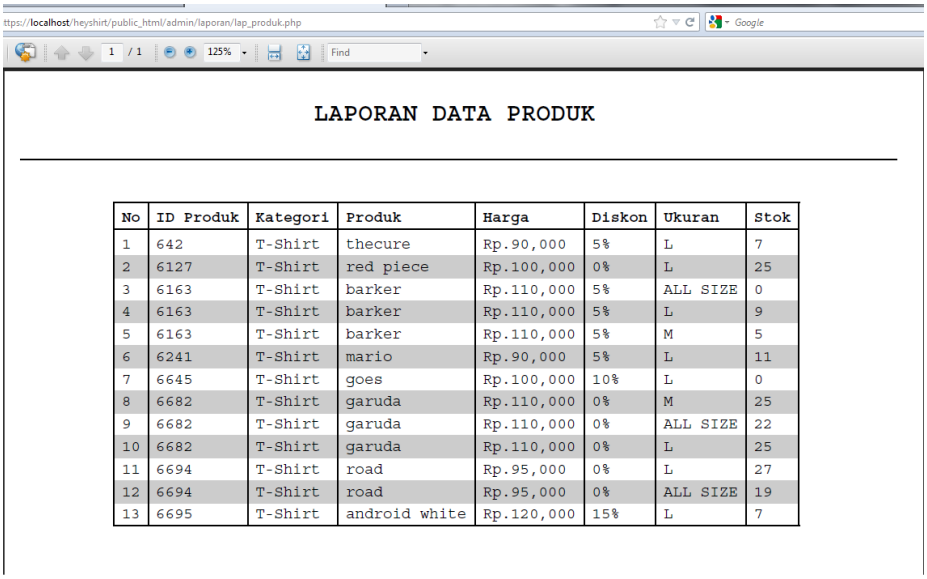
10. Manajemen Laporan

Menu manajemen laporan pada halaman website ini bertujuan menampilkan data penjualan, produk dan data retur pembelian setalah distro heyshirt melakukan transaksi kepada pelanggan guna mempermudah pegawai distro heyshirt melakukan rekapitulasi data.



Gambar 5.33 Tampilan Laporan pada akun *admin keuangan* pada *website* Distro Heyshirt Clothing.

Berikut merupakan tampilan dari laporan persediaan dan data produk, apabila petugas mengklik *link* cetak laporan yang ada pada halaman daftar produk.

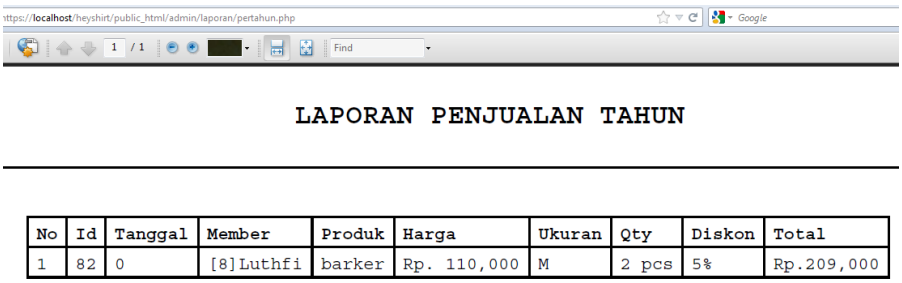


LAPORAN DATA PRODUK

No	ID Produk	Kategori	Produk	Harga	Diskon	Ukuran	Stok
1	642	T-Shirt	thecure	Rp.90,000	5%	L	7
2	6127	T-Shirt	red piece	Rp.100,000	0%	L	25
3	6163	T-Shirt	barker	Rp.110,000	5%	ALL SIZE	0
4	6163	T-Shirt	barker	Rp.110,000	5%	L	9
5	6163	T-Shirt	barker	Rp.110,000	5%	M	5
6	6241	T-Shirt	mario	Rp.90,000	5%	L	11
7	6645	T-Shirt	goes	Rp.100,000	10%	L	0
8	6682	T-Shirt	garuda	Rp.110,000	0%	M	25
9	6682	T-Shirt	garuda	Rp.110,000	0%	ALL SIZE	22
10	6682	T-Shirt	garuda	Rp.110,000	0%	L	25
11	6694	T-Shirt	road	Rp.95,000	0%	L	27
12	6694	T-Shirt	road	Rp.95,000	0%	ALL SIZE	19
13	6695	T-Shirt	android white	Rp.120,000	15%	L	7

Gambar 5.34 Tampilan Laporan data produk pada akun *admin keuangan* pada *website* Distro Heyshirt Clothing.

Berikut merupakan tampilan dari laporan penjualan pertahun dan bisa di sesuaikan dengan penjualan perbulan atau perhari.



LAPORAN PENJUALAN TAHUN

No	Id	Tanggal	Member	Produk	Harga	Ukuran	Qty	Diskon	Total
1	82	0	[8]Luthfi	barker	Rp. 110,000	M	2 pcs	5%	Rp.209,000

Gambar 5.35 Tampilan Laporan Penjualan produk pada akun *admin keuangan* pada *website* Distro Heyshirt Clothing.

5.2 Pengujian Perangkat Lunak

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan kesalahan – kesalahan atau kekurangan – kekurangan pada perangkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

Pengujian perangkat lunak ini menggunakan pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

5.2.1 Rencana Pengujian

Pengujian perangkat lunak Sistem Informasi Penjualan dan Promosi Berbasis *Web* ini menggunakan metode *black box.* Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak yang dibuat.

Tabel 5.1 menjelaskan skenario pengujian aplikasi *member.*

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Registrasi *member*	Menampilkan *form* user baru	*Black Box*
	Isi *form* pendaftaran	*Black Box*
	Simpan data baru	*Black Box*
Login	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Logout	Menghapus *session user*	*Black Box*
Edit profil	Menampilkan *form* ubah profil	*Black Box*
	Ubah data *costumer*	*Black Box*

	Simpan perubahan	*Black Box*
Ganti *password*	Menampilkan *form* ubah *password*	*Black Box*
	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verifikasi konfirmasi *password*	*Black Box*
Menu produk	Menampilkan data barang	*Black Box*
	Menampilkan data detail barang	*Black Box*
	Cari barang	*Black Box*
Menu about	Menampilkan profil perusahaan	*Black Box*
Menu cara pembayaran	Menampilkan petunjuk pembayaran	*Black Box*
Menu keranjang belanja	Menampilkan data detail transaksi	*Black Box*

Tabel 5.2 menjelaskan skenario pengujian aplikasi admin.

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Login	Isi data login	
	Verifikasi *username*	*Black Box*
	Verivikasi *password*	*Black Box*
Logout	Menghapus *session user*	*Black Box*
Ganti P*assword*	Menampilkan *form* ganti *password*admin	*Black Box*
	Verifikasi *password*lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verfikasi konfirmasi *password*	*Black Box*
Member	Tampilkan data member	*Black Box*
Kota	Tampilkan data kota	*Black Box*
	Tambah data kota	*Black Box*

	Ubah data kota	*Black Box*
Provinsi	Tampilkan data Provinsi	*Black Box*
	Tambah data Provinsi	*Black Box*
	Ubah data Provinsi	*Black Box*
Kategori	Tampilkan data Kategori	*Black Box*
	Tambah data Kategori	*Black Box*
	Ubah data Kategori	*Black Box*
Retur	Tampilkan data Retur	*Black Box*
	Tambah data Retur	*Black Box*
	Ubah data Retur	*Black Box*
Laporan Penjualan	Tampilkan Laporan Penjualan	*Black Box*
Laporan Produk	Tampilkan Laporan Produk	*Black Box*

5.2.2 Kasus dan Hasil Pengujian

Kasus dan hasil pengujian dibuat untuk mengetahui apakah Sistem Informasi Penjualan dan Promosi Berbasis Web pada Distro Heyshirt Clothing dapat berjalan dengan baik atau tidak.

5.2.2.1 Kelas Uji Login

Berikut ini adalah tabel pengujian *login* Akun *Admin* untuk pengecekkan *user* yang telah terdaftar :

1. Pengujian Registrasi

Tabel 5.3 Pengujian Registrasi *Member* (Data Benar)

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data pendaftaran, contoh masukan: Nama: Luthfi pratama Alamat : Jl. Ciheulang no. 53c Telpon: 085722777134 Email : kulkasrusak83@yahoo.com Password : 123456 Konfirmasi password : 123456 Provinsi : Jawa Barat Kota : Bandung Kodepos : 40134 KodeCaptcha: hay61	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses registrasi.	Form registrasi terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses registrasi.	[√] Diterima [ ] Ditolak

Tabel 5.4 Pengujian Registrasi *Member* (Data Salah)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama: (kosong) Alamat : (kosong) Telpon: (kosong) Email : (kosong) Password : (kosong) Konfirmasi password : (kosong) Provinsi : (kosong) Kota : (kosong) Kodepos : (kosong) KodeCaptcha: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan bahwa beberapa kolom yang tersedia harus terisi.	Pesan kesalahan dapat tampil dan muncul pesan kesalahan bahwa beberapa kolom yang tersedia harus terisi.	[√] Diterima [ ] Ditolak

2. *Pengujian Login* Member

Berikut dapat dilihat pengujian login *member* pada tabel 4.6 dibawah ini:

Tabel 5.5 Pengujian Login *Member* (Data Benar)

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data login, contoh masukan: *email* : her@yahoo.co.id password: qwerty	Mengisikan data alamat *email* pada *textbox* email dan password	Dapat mengisi *textbox* email dan password	[√] Diterima [ ] Ditolak

Tabel 5.6 Pengujian Login *Member* (Data Salah)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data login, contoh masukan: *email* : (kosong) password: (kosong)	Tampilkan pesan kesalahan ”Maaf, Akun Tidak Ditemukan Masukan Email dan password dengan benar”	Dapat menampilkan pesan kesalahan ”Maaf, Akun Tidak Ditemukan Masukan Email dan password dengan benar”	[√] Diterima [ ] Ditolak

3. *Pengujian Edit Profile* Member

Berikut dapat dilihat pengujian dari *edit profile* member pada tabel 5.7 dibawah ini:

Tabel 5.7 Pengujian *Edit Profil Member* (Data Benar)

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data *edit profile*, contoh masukan: Nama Lengkap: Luthfi Email: kulkasrusak83@yahoo.com Password Lama: qwerty Password Baru: asdfgh Konfirmasi Password: asdfgh Alamat: ciheulang Provinsi: Jawa Barat Kota: Bandung Kode Pos: 45122 Telepon: 081234567	Dapat mengubah isi *text box edit profil* yang akan diubah dan dapat melanjutkan proses *edit profile*	Form *edit profile* terisi atau benar dalam pengisian, maka akan dapat melanjutkan *edit profile*.	[√] Diterima [ ] Ditolak

Tabel 5.8 Pengujian *Edit Profil Member* (Data Salah)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data *edit profile*, contoh masukan: Nama Lengkap: (kosong) Email: (kosong) Password Lama: (kosong) Password Baru: (kosong) Konfirmasi Password: (kosong) Alamat: (kosong) Provinsi: (kosong) Kota: (kosong) Kode Pos: (kosong) Telepon: (kosong)	Muncul pesan kesalahan "kolom yang tersedia tidak boleh kosong"	Dapat menampilkan pesan kesalahan kolom yang tersedia tidak boleh kosong	[√] Diterima [ ] Ditolak

4. Pengujian Menu Kontak

Berikut dapat dilihat pengujian dari menu kontak pada tabel 5.9 dibawah ini:

Tabel 5.9 Pengujian Menu Kontak (Data Benar)

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data kontak kami, contoh masukan: Nama: Luthfi Email: kulkasrusak83@yahoo.com Komentar: Terima kasih Kode Captcha: 1Ofd0	Mengisikan data kontak kami pada *textbox*	Dapat mengisi *textbox*	[√] Diterima [ ] Ditolak

Tabel 5.10 Pengujian Menu Kontak (Data Salah)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data kontak kami, contoh masukan: Nama: (kosong) Email: (kosong) Komentar: (kosong) Kode Captcha: (kosong)	Muncul pesan kesalahan "kolom yang tersedia tidak boleh kosong"	Dapat menampilkan pesan kesalahan kolom yang tersedia tidak boleh kosong	[√] Diterima [ ] Ditolak

5. Pengujian Login Admin

Berikut dapat dilihat pengujian login admin pada tabel 5.11 dibawah ini:

Tabel 5.11 Pengujian Login Admin (Data Benar)

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data login admin, contoh masukan: *username* : admin password: qwerty	Mengisikan data *username* pada *textbox* *username* dan password	Dapat mengisi *textbox* *username* dan password	[√] Diterima [ ] Ditolak

Tabel 5.12 Pengujian Login Admin (Data Salah)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data login admin, contoh masukan: *username* : (kosong) password: (kosong)	Tampilkan pesan kesalahan ”Username atau Password Salah”	Dapat menampilkan pesan kesalahan ” Username atau Password Salah”	[√] Diterima [ ] Ditolak

6. Pengujian Edit Admin

Berikut dapat dilihat pengujian dari *edit* admin pada tabel 5.13 dibawah ini:

Tabel 5.13 Pengujian *Edit* Admin (Data Benar)

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data *edit* admin, contoh masukan: *username* : admin Email: kulkasrusak83@yahoo.com Password Lama: qwerty Password: asdfgh Confirm Password: asdfgh	Dapat mengubah isi *text box edit admin* yang akan diubah dan dapat melanjutkan proses *edit* admin	Form *edit* admin terisi atau benar dalam pengisian, maka akan dapat melanjutkan *edit* admin.	[√] Diterima [ ] Ditolak

Tabel 5.14 Pengujian *Edit* Admin (Data Salah)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data *edit* admin, contoh masukan: *Username* : (kosong) Email: (kosong) Password Lama: (kosong) Password: (kosong) Confirm Password: (kosong)	Muncul pesan kesalahan “kolom yang tersedia tidak boleh kosong”	Dapat menampilkan pesan kesalahan kolom yang tersedia tidak boleh kosong	[√] Diterima [ ] Ditolak

7. Pengujian Data Tambah kategori

Berikut dapat dilihat pengujian data tambah kategori pada tabel 5.15 dibawah ini:

Tabel 5.15 Pengujian Data Tambah Kategori (Data Benar)

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data tambah Kategori, contoh masukan : Kategori : kemeja	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah kategori	Jika semua form terisi atau benar dalam pengisian, dapat melanjutkan proses tambah Kategori	[√] Diterima [ ] Ditolak

Tabel 5.16 Pengujian Data Tambah Kategori(Data Salah)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data tambah Kategori, contoh masukan: Kategori: (kosong)	Tampilkan pesan kesalahan "Kategori tidak boleh kosong"	Dapat menampilkan pesan kesalahan "Kategori tidak boleh kosong"	[√] Diterima [ ] Ditolak

8. Pengujian Data Edit Kategori

Berikut dapat dilihat pengujian data tambah kategori pada tabel 5.17 dibawah ini:

Tabel 5.17 Pengujian Data Edit Kategori (Data Benar)

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data edit Kategori, contoh masukan: Kategori: jas	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses edit Kategori	Jika semua form terisi atau benar dalam pengisian, dapat melanjutkan	[√] Diterima [ ] Ditolak

		proses edit Kategori	

Tabel 5.18 Pengujian Data Edit Kategori (Data Salah)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data edit Kategori, contoh masukan: Kategori: (kosong)	Tampilkan pesan kesalahan "Kategori tidak boleh kosong"	Dapat menampilkan pesan kesalahan "Kategori tidak boleh kosong"	[√] Diterima [ ] Ditolak

9. Pengujian Tambah Data Provinsi

Berikut dapat dilihat pengujian dari tambah data provinsi pada tabel 5.19 dibawah ini:

Tabel 5.19 Pengujian Tambah Provinsi(Data Benar)

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data tambah provinsi, contoh masukan: Provinsi: Jawa Barat	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah provinsi	Jika semua form terisi atau benar dalam pengisian, dapat melanjutkan proses tambah provinsi	[√] Diterima [ ] Ditolak

Tabel 5.20 Pengujian Tambah Provinsi (Data Salah)

Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data tambah provinsi, contoh masukan: Provinsi: (kosong)	Tampilkan pesan kesalahan "Provinsi tidak boleh kosong"	Dapat menampilkan pesan kesalahan "Provinsi tidak boleh kosong"	[√] Diterima [ ] Ditolak

10. Pengujian Edit Data Provinsi

Berikut dapat dilihat pengujian dari edit data provinsi pada tabel 5.21 dibawah ini:

Tabel 5.21 Pengujian Edit Data Provinsi(Data Benar)

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data edit provinsi, contoh masukan: Provinsi : Jawa Barat	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses edit provinsi	Jika semua form terisi atau benar dalam pengisian, dapat melanjutkan proses edit provinsi	[√] Diterima [ ] Ditolak

Tabel 5.22 Pengujian Edit Data Provinsi(Data Salah)

Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data edit provinsi, contoh masukan: Provinsi : (kosong)	Tampilkan pesan kesalahan "Provinsi tidak boleh kosong"	Dapat menampilkan pesan kesalahan "Provinsi tidak boleh kosong"	[√] Diterima [ ] Ditolak

11. Pengujian Tambah Data Kota

Berikut dapat dilihat pengujian data kota pada tabel 5.23 dibawah ini:

Tabel 5.23 Pengujian Tambah Data Kota

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data tambah kota, contoh masukan: Nama Kota: Cirebon Provinsi: Jawa Barat Kab/Kota: Kota	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah kota	Jika semua form terisi atau benar dalam pengisian, dapat melanjutkan proses tambah kota	[√] Diterima [ ] Ditolak

Tabel 5.24 Pengujian Tambah Data Kota

Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data tambah kota, contoh masukan: Nama Kota: (kosong) Provinsi: (kosong) Kab/Kota: (kosong)	Tampilkan pesan kesalahan "Kota, Provinsi dan Kab/Kota tidak boleh kosong"	Dapat menampilkan pesan kesalahan " Kota, Provinsi dan Kab/Kota tidak boleh kosong"	[√] Diterima [ ] Ditolak

12. Pengujian Edit Data Kota

Berikut dapat dilihat pengujian dari edit data kota pada tabel 5.25 dibawah ini:

Tabel 5.25 Pengujian Edit Data Kota (Data Benar)

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data edit kota, contoh masukan: Nama Kota: Cirebon	Jika form terisi atau benar dalam pengisian, maka	Jika semua form terisi atau benar dalam	[√] Diterima [ ] Ditolak

Provinsi: Jawa Barat Kab/Kota: Kota	akan dapat melanjutkan proses edit kota	pengisian, dapat melanjutkan proses edit kota	

Tabel 5.26 Pengujian Edit Data Kota (Data Salah)

Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data edit kota, contoh masukan: Nama Kota: (kosong) Provinsi: (kosong) Kab/Kota: (kosong)	Tampilkan pesan kesalahan "Kota, Provinsi dan Kab/Kota tidak boleh kosong"	Dapat menampilkan pesan kesalahan " Kota, Provinsi dan Kab/Kota tidak boleh kosong"	[√] Diterima [ ] Ditolak

13. Pengujian Data Tambah Produk

Berikut dapat dilihat pengujian data kategori pada tabel 5.27 dibawah ini:

Tabel 5.27 Pengujian Data Tambah Produk (Data Benar)

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data tambah produk, contoh masukan: Nama Produk: road Harga Produk: 48000 Diskon: 10% Kategori: T-Shirt Masukan Gambar: C../road.jpg	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah produk	Jika semua form terisi atau benar dalam pengisian, dapat melanjutkan proses tambah produk	[√] Diterima [ ] Ditolak

Tabel 5.28 Pengujian Data Tambah Produk (Data Salah)

Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data tambah produk, contoh masukan: Nama Produk: (kosong) Harga Produk: (kosong) Diskon: (kosong)	Tampilkan pesan kesalahan "Data tidak boleh kosong"	Dapat menampilkan pesan kesalahan "Data tidak	[√] Diterima [ ] Ditolak

Kategori: (kosong) Masukan Gambar: (kosong)		boleh kosong”	

14. Pengujian Edit Data Produk

Tabel 5.29 Pengujian Data Tambah Produk (Data Benar)

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data edit produk, contoh masukan: Nama Produk: road Harga Produk: 48000 Diskon: 10% Kategori: T-Shirt Masukan Gambar: C../road.jpg	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses edit produk	Jika semua form terisi atau benar dalam pengisian, dapat melanjutkan proses edit produk	[√] Diterima [ ] Ditolak

Tabel 5.30 Pengujian Data Tambah Produk (Data Salah)

Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data edit produk, contoh masukan: Nama Produk: (kosong) Harga Produk: (kosong) Diskon: (kosong) Kategori: (kosong) Masukan Gambar: (kosong)	Tampilkan pesan kesalahan “Data tidak boleh kosong”	Dapat menampilkan pesan kesalahan “Data tidak boleh kosong”	[√] Diterima [ ] Ditolak

15. Pengujian Tambah Data Detail Produk

Berikut dapat dilihat pengujian data retur pada tabel 5.31 dibawah ini:

Tabel 5.31 Pengujian Tambah Data Detail Produk (Data Benar)

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data tambah detail produk, contoh masukan:	Jika form terisi atau benar dalam pengisian, maka	Jika semua form terisi atau benar dalam	[√] Diterima [ ] Ditolak

Ukuran: L Berat: 320gr Stok: 10pcs	akan dapat melanjutkan proses tambah detail produk	pengisian, dapat melanjutkan proses tambah detail produk	

Tabel 5.32 Pengujian Tambah Data Detail Produk (Data Salah)

Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data tambah detail produk, contoh masukan: Ukuran: (kosong) Berat: (kosong) Stok: (kosong)	Tampilkan pesan kesalahan "Data tidak boleh kosong"	Dapat menampilkan pesan kesalahan "Data tidak boleh kosong"	[√] Diterima [ ] Ditolak

16. Pengujian Edit Data Detail Produk

Tabel 5.33 Pengujian Edit Data Detail Produk (Data Benar)

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data edit detail produk, contoh masukan: Ukuran: L Berat: 320gr Stok: 10pcs	Jika form terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses edit detail produk	Jika semua form terisi atau benar dalam pengisian, dapat melanjutkan proses edit detail produk	[√] Diterima [ ] Ditolak

Tabel 5.34 Pengujian Edit Data Detail Produk (Data Benar)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data edit detail produk, contoh masukan: Ukuran: (kosong) Berat: (kosong) Stok: (kosong)	Tampilkan pesan kesalahan "Data tidak boleh kosong"	Dapat menampilkan pesan kesalahan "Data tidak boleh kosong"	[√] Diterima [ ] Ditolak

5.2.3 Kesimpulan Hasil Pengujian

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