BAB IV
IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Tahap implementasi sistem merupakan tahap menerjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dapat dimengerti oleh mesin, serta penerapan perangkat lunak pada keadaan yang sesungguhnya. Tahap ini merupakan tahap dimana sistem siap untuk dioperasikan, yang terdiri dari penjelasan mengenai lingkungan implementasi, dan implementasi program.

4.1.1 Perangkat Keras Yang Digunakan

Spesifikasi perangkat keras yang digunakan dalam pembangunan aplikasi *E-Commerce* pada Dela Jaya Variasi adalah tercantum pada tabel 4.1 sebagai berikut :

Tabel 4.1. Perangkat keras yang digunakan

Perangkat	Spesifikasi
Processor	Kecepatan 2.4 GHz
RAM	1,5 GHz
Harddisk	120 Gb
Monitor	Resolusi 1280 x 800
VGA	128 Mb

4.1.2 Perangkat Lunak Yang Digunakan

Spesifikasi perangkat lunak yang digunakan dalam membangun aplikasi *E-Commerce* pada Dela Jaya Variasi adalah sebagai berikut:

Tabel 4.2. Perangkat Lunak yang digunakan

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows XP SP 2
2	Bahasa Pemrograman	PHP
3	*DBMS*	MySQL 5.0.27
4	*Web browser*	Mozilla firefox 3.5.5
5	*Code editor*	Macromedia Dreamweaver 8

6	DFD modeler	Microsoft Visio 2007

4.2 Implementasi database

Pembuatan *database* dilakukan dengan menggunakan aplikasi *DBMS MySQL*. Implementasi *database* dalam bahasa *SQL* adalah sebagai berikut:

1. Tabel Admin

```
CREATE TABLE IF NOT EXISTS `admin`
(
   `username` char(10) NOT NULL DEFAULT,
   `email` varchar(100) DEFAULT NULL,
   `nama_admin` varchar(100) DEFAULT NULL,
   `password` char(100) DEFAULT NULL,
   `status` char(10) DEFAULT 'User',
   `ym` char(100) DEFAULT NULL,
   PRIMARY KEY (`username`),
   UNIQUE KEY `email` (`email`),
   UNIQUE KEY `ym` (`ym`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1  ;
```

2. Tabel Bentuk

```
CREATE TABLE IF NOT EXISTS `bentuk`
(
   id_bentuk` char(5) NOT NULL DEFAULT '',
   `username` char(10) DEFAULT NULL,
   `id_kategori` char(5) DEFAULT NULL,
   `bentuk` char(25) DEFAULT NULL,
   PRIMARY KEY (`id_bentuk`),
   KEY `FK_bentuk_admin` (`username`),
   KEY `FK_bentuk_kategori` (`id_kategori`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Tabel Produk

```
CREATE TABLE IF NOT EXISTS `produk`
(
   `id_produk` char(5) NOT NULL DEFAULT '',
   `username` char(10) DEFAULT NULL,
```

```
   `id_kategori` char(5) DEFAULT NULL,
   `id_bentuk` char(5) DEFAULT NULL,
   `id_merek` char(10) DEFAULT NULL,
   `nama_produk` varchar(50) DEFAULT NULL,
   `harga` int(10) DEFAULT '0',
   `harga2` int(10) DEFAULT '0',
   `diskon` int(3) DEFAULT '0',
   `stok` int(5) DEFAULT '0',
   `berat` int(3) DEFAULT '0',
   PRIMARY KEY (`id_produk`),
   KEY `FK_produk_admin` (`username`),
   KEY `FK_produk_kategori` (`id_kategori`),
   KEY `FK_produk_bentuk` (`id_bentuk`),
   KEY `FK_produk_merek` (`id_merek`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4. Tabel Detail_pembelian

```
CREATE TABLE IF NOT EXISTS `detail_pembelian`
(
   `id_detail` int(10) NOT NULL AUTO_INCREMENT,
   `no_pesan` int(11) DEFAULT NULL,
   `id_produk` char(5) DEFAULT NULL,
   `jumlah` int(5) DEFAULT '0',
   `total_berat` int(5) DEFAULT '0',
   `rupiah` int(9) DEFAULT '0',
   `dollar` int(9) DEFAULT '0',
   PRIMARY KEY (`id_detail_pembelian`),
   KEY `FK_detail_pembelian_produk` (`id_produk`),
   KEY `FK_detail_pembelian_transaksi` (`no_pesan`)
)
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

5. Tabel Gambar

```
CREATE TABLE IF NOT EXISTS `id_gambar`
(
   `id_gambar` int(7) NOT NULL AUTO_INCREMENT,
   `username` char(10) DEFAULT NULL,
   `id_produk` char(5) DEFAULT NULL,
   `gambar` varchar(100) DEFAULT NULL,
   `status` int(1) DEFAULT '0',
   PRIMARY KEY (`id_gambar`),
```

```
  UNIQUE KEY `gambar` (`gambar`),
  KEY `FK_id_gambar_admin` (`username`),
  KEY `FK_id_gambar_produk` (`id_produk`)
)
ENGINE=InnoDB   DEFAULT  CHARSET=latin1  AUTO_INCREMENT=20
;
```

6. Tabel Info

```
CREATE TABLE IF NOT EXISTS `info`
(
   `id_info` int(2) NOT NULL AUTO_INCREMENT,
   `username` char(10) DEFAULT NULL,
   `isi` text,
   PRIMARY KEY (`id_info`),
   KEY `FK__admin` (`username`)
)
ENGINE=InnoDB   DEFAULT  CHARSET=latin1  ROW_FORMAT=COMPACT
AUTO_INCREMENT=2 ;
```

7. Tabel Jasa Pengiriman

```
CREATE TABLE IF NOT EXISTS `jasa pengiriman`
(
   `id_jasa` char(4) NOT NULL DEFAULT '',
   `username` char(10) DEFAULT NULL,
   `nm_jasa` char(15) DEFAULT NULL,
   `link` varchar(150) DEFAULT NULL,
   `gambar` varchar(100) DEFAULT NULL,
   PRIMARY KEY (`id_jasa`),
   UNIQUE KEY `nm_jasa` (`nm_jasa`),
   UNIQUE KEY `link` (`link`),
   KEY `FK_jasa_admin` (`username`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

8. Tabel Kategori

```
CREATE TABLE IF NOT EXISTS `kategori`
(
   `id_kategori` char(5) NOT NULL DEFAULT '',
   `username` char(10) DEFAULT NULL,
   `nm_kategori` char(30) DEFAULT NULL,
```

```
   PRIMARY KEY (`id_kategori`),
   UNIQUE KEY `nm_kategori` (`nm_kategori`),
   KEY `FK_kategori_admin` (`username`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

9. Tabel Keranjang

```
CREATE TABLE IF NOT EXISTS `keranjang`
(
   `id_keranjang` int(10) NOT NULL AUTO_INCREMENT,
   `kd_plg` char(7) NOT NULL,
   `id_produk` char(5) NOT NULL,
   `tgl` date NOT NULL,
   `jam` time NOT NULL,
   `jumlah` int(5) NOT NULL DEFAULT '0',
   `total_berat` int(5) NOT NULL DEFAULT '0',
   `bayar` int(9) NOT NULL DEFAULT '0',
   PRIMARY KEY (`id_keranjang`),
   KEY `FK_keranjang_pelanggan` (`kd_plg`),
   KEY `FK_keranjang_produk` (`id_produk`)
   )
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

10. Tabel Kontak

```
CREATE TABLE IF NOT EXISTS `kontak`
(
   `id_kontak` int(2) NOT NULL AUTO_INCREMENT,
   `username` char(10) DEFAULT NULL,
   `isi` text,
   PRIMARY KEY (`id_kontak`),
   KEY `FK__admin` (`username`)
)
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

11. Tabel Kota

```
CREATE TABLE IF NOT EXISTS `kota`
(
  `id_kota` char(5) NOT NULL DEFAULT '',
  `username` char(10) DEFAULT NULL,
  `id_prop` char(5) DEFAULT NULL,
```

```
 `kota` char(30) DEFAULT NULL,
 PRIMARY KEY (`id_kota`),
 UNIQUE KEY `nm_kota` (`kota`),
 KEY `FK_kota_admin` (`username`),
 KEY `FK_kota_propinsi` (`id_prop`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

12. Tabel Merek

```
CREATE TABLE IF NOT EXISTS `merek`
(
 `id_merek` char(10) NOT NULL DEFAULT '',
 `username` char(10) NOT NULL DEFAULT '',
 `merek` char(30) DEFAULT NULL,
 PRIMARY KEY (`id_merek`),
 UNIQUE KEY `merek` (`merek`),
 KEY `FK_merek_admin` (`username`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

13. Tabel Rating

```
CREATE TABLE IF NOT EXISTS `rating`
(
      `id` int(5) NOT NULL AUTO_INCREMENT,
      `username` char(10) NOT NULL,
      `id_produk` char(5) DEFAULT NULL,
      `view` int(5) DEFAULT '0',
      `best` int(5) DEFAULT '0',
      `rate` int(5) DEFAULT NULL,
      PRIMARY KEY (`id`),
      KEY `FK_rating_produk` (`id_produk`),
      KEY `FK_rating_admin` (`username`)
)
ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4
;
```

14. Tabel Member

```
CREATE TABLE IF NOT EXISTS `member`
(
      `kd_member` char(6) NOT NULL DEFAULT '',
      `nama` varchar(50) DEFAULT NULL,
```

```
      `alamat` text,
      `id_prop` char(5) DEFAULT NULL,
      `id_kota` char(5) DEFAULT NULL,
      `kdpos` int(5) unsigned DEFAULT NULL,
      `email` varchar(50) DEFAULT NULL,
      `tlp` varchar(15) DEFAULT NULL,
      `password` varchar(50) DEFAULT NULL,
      `status` int(1) unsigned DEFAULT '0',
      PRIMARY KEY (`kd_member`),
      UNIQUE KEY `email` (`email`),
      KEY `FK_member_kota` (`id_kota`),
      KEY `FK_member_propinsi` (`id_prop`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

15. Tabel Pembayaran

```
CREATE TABLE IF NOT EXISTS `pembayaran`
(
      `id_pembayaran` char(3) NOT NULL DEFAULT '',
      `username` char(10) DEFAULT NULL,
      `nm_pembayaran` char(20) DEFAULT NULL,
      `ket` text,
      `gambar` varchar(100) DEFAULT NULL,
      PRIMARY KEY (`id_pembayaran`),
      UNIQUE KEY `nm_pembayaran` (`nm_pembayaran`),
      KEY `FK_pembayaran_admin` (`username`)
)
ENGINE=InnoDB          DEFAULT          CHARSET=latin1
ROW_FORMAT=COMPACT;
```

16. Tabel Provinsi

```
CREATE TABLE IF NOT EXISTS `propinsi`
(
      `id_prop` char(5) NOT NULL DEFAULT '',
      `username` char(10) DEFAULT NULL,
      `propinsi` char(25) DEFAULT NULL,
      PRIMARY KEY (`id_prop`),
      UNIQUE KEY `propinsi` (`propinsi`),
      KEY `FK_propinsi_admin` (`username`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

17. Tabel Retur

```
CREATE TABLE IF NOT EXISTS `retur`
(
      `id_retur` char(8) NOT NULL DEFAULT '',
      `no_pesan` int(11) DEFAULT NULL,
      `id_produk` char(5) DEFAULT NULL,
      `kd_member` char(6) DEFAULT NULL,
      `jumlah_retur` int(5) DEFAULT '0',
      `tgl_retur` date DEFAULT NULL,
      `foto` varchar(100) DEFAULT NULL,
      `ket_retur` text,
      `status_retur` char(10) DEFAULT 'Menunggu',
      PRIMARY KEY (`id_retur`),
      KEY `FK_retur_produk` (`id_produk`),
      KEY `FK_retur_pelanggan` (`kd_member`),
      KEY `FK_retur_pembelian` (`no_pesan`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

18. Tabel Detail Pengiriman

```
CREATE TABLE IF NOT EXISTS `detail pengiriman`
(
 `id_shipp` char(5) NOT NULL DEFAULT '',
 `username` char(10) NOT NULL DEFAULT '',
 `id_jasa` char(4) NOT NULL DEFAULT '',
 `id_subjasa` char(4) NOT NULL DEFAULT '',
 `id_prop` char(5) DEFAULT NULL,
 `id_kota` char(5) DEFAULT NULL,
 `biaya` int(9) NOT NULL DEFAULT '0',
 PRIMARY KEY (`id_shipp`),
 KEY `FK_detail pengiriman_admin` (`username`),
 KEY `FK_detail pengiriman_jasa` (`id_jasa`),
 KEY     `FK_detail     pengiriman_paket     pemgiriman`
(`id_paket`),
 KEY `FK_detail pengiriman_propinsi` (`id_prop`),
 KEY `FK_detail pengiriman_kota` (`id_kota`)
)
ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

19. Tabel Detail Transaksi

```
CREATE TABLE IF NOT EXISTS `detail transaksi`
```

```
(
 `no_pesan` int(11) NOT NULL AUTO_INCREMENT,
 `kd_member` char(6) DEFAULT NULL,
 `id_detail pengiriman` char(5) DEFAULT NULL,
 `id_pembayaran` char(3) DEFAULT NULL,
 `tgl` date DEFAULT NULL,
 `lim_tgl` date DEFAULT NULL,
 `jam` time DEFAULT NULL,
 total_rupiah` int(9) DEFAULT NULL,
 `total_dollar` int(9) DEFAULT NULL,
 `total_item` int(5) DEFAULT NULL,
 `total_berat` int(5) DEFAULT NULL,
 `biaya_kirim` int(9) DEFAULT NULL,
 `no_resi` varchar(50) DEFAULT NULL,
 `nama_tujuan` varchar(100) DEFAULT NULL,
 `alamat_tujuan` text,
 `tlp_tujuan` int(15) DEFAULT NULL,
 `rekening` char(50) DEFAULT NULL,
 `no_transfer` varchar(50) DEFAULT NULL,
 `tgl_transfer` date DEFAULT NULL,
 `jum_transfer` int(9) DEFAULT NULL,
 `status` char(20) DEFAULT 'Pesan',
 `retur` char(5) DEFAULT 'Tidak',
 `ex_retur` date DEFAULT NULL,
  PRIMARY KEY (`no_pesan`),
 KEY `FK_detail transaksi_member` (`kd_member`),
 KEY     `FK_detail     transaksi_detail     pengiriman`
(`id_detail pengiriman`)
)
ENGINE=InnoDB   DEFAULT  CHARSET=latin1  AUTO_INCREMENT=2
;
```

20. Tabel Paket pengiriman

```
CREATE TABLE IF NOT EXISTS `paket pengiriman`
(
 `id_paket pengiriman` char(4) NOT NULL DEFAULT '',
 `username` char(10) NOT NULL DEFAULT '',
 `id_jasa` char(4) DEFAULT NULL,
 `nm_paket pengiriman` char(50) DEFAULT NULL,
 PRIMARY KEY (`id_paket pengiriman`),
 KEY `FK_paket pengiriman_admin` (`username`),
 KEY `FK_paket pengiriman_jasa` (`id_jasa`)
)
```

```
ENGINE=InnoDB            DEFAULT            CHARSET=latin1
ROW_FORMAT=COMPACT;
```

4.3 Implementasi Antar Muka

Implementasi antar muka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna. Adapun untuk tampilan implementasi aplikasi *E-commerce* pada Dela Jaya Variasi akan dilampirkan pada halaman lampiran.

Tabel 4.3 Implementasi antar muka

Menu	Deskripsi	Nama File
Login administrator	Digunakan sebagai halaman login administrator	Index.php
ganti *password* administrator	Digunakan jika administrator ingin merubah *password*	pass.php
bentuk	Digunakan untuk melihat, menambah, dan merubah data bentuk	bentuk. php
produk	Digunakan untuk melihat, menambah, dan merubah data produk	produk. php
kategori	Digunakan untuk melihat, menambah, dan merubah data kategori	kategori. php
info	Digunakan untuk melihat, menambah, dan merubah data info	info. php
merek	Digunakan untuk melihat, menambah, dan merubah data merek	Merek.php
kontak	Digunakan untuk melihat, dan merubah data kontak	kontak. php
Detail_pengiriman	Digunakan untuk melihat, dan merubah data detail_pengiriman	detail_pengiriman. php

Registrasi *member*	Digunakan untuk melakukan pendaftaran *member*	pendaftaran.php
Login *member*	Digunakan sebagai halaman login *member*	login.php
Ganti *password member*	Digunakan jika *member* ingin merubah *password*	password.php
Ubah Profil	Digunakan untuk mengubah profil *member*	profil. Php
Produk	Digunakan untuk melihat seluruh data barang yang ada	Produk.php
Riwayat	Digunakan untuk melihat data riwayat transaksi member	riwayat.php
Keranjang belanja	Digunakan untuk melihat, merubah, menghapus detail transaksi yang akan dilakukan	cart.php
Logout	Digunakan User (administrator, *member*)Untuk dapat keluar dari aplikasi	Logout.php

4.4 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.

4.5 Pengujian Alpa

Pengujian alpha dilakukan dengan menggunakan metode *black box*. Untuk pengujian alpha ini yaitu pada pengujian sebagai pengguna

4.5.1 Skenario Pengujian Alpa

Tabel 4.4. menjelaskan skenario pengujian aplikasi *member*.

Tabel 4.4. Skenario pengujian aplikasi *member*

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Registrasi	Menampilkan *form* pendaftaran *member* baru	*Black Box*
	Isi *form* pendaftaran *member*	*Black Box*
	Simpan data baru	*Black Box*
Login *member*	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Logout	Menghapus *session user*	*Black Box*
My account	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 tentang kami	Menampilkan profil perusahaan	*Black Box*
Menu cara belanja	Menampilkan cara belanja	*Black Box*
Menu pengiriman	Menampilkan cara pengiriman	*Black Box*
Menu cara membayar	Menampilkan cara membayar	*Black Box*
Menu keranjang belanja	Menampilkan data detail transaksi	*Black Box*
	Menampilkan konfirmasi	*Black Box*

	transaksi	
	Menampilkan form pengiriman	*Black Box*
Form konfirmasi	Isi form pengiriman	*Black Box*
Form pengiriman	Menampilkan cara pembayaran	*Black Box*
	Menampilkan data transaksi yang pernah dilakukan	*Black Box*
Form cara pembayaran	Menampilkan data detail tiap transaksi	*Black Box*

Tabel 4.5. Skenario pengujian aplikasi administrator

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 *password*	Menampilkan *form* ganti *password*	*Black Box*
	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verfikasi konfirmasi *password*	*Black Box*
pengiriman	Tampilkan data pengiriman	*Black Box*
	Tambah data pengiriman	*Black Box*
	Ubah data pengiriman	*Black Box*
provinsi	Tampilkan data provinsi	*Black Box*
	Tambah data provinsi	*Black Box*
	Ubah data provinsi	*Black Box*
kota	Tampilkan data kota	*Black Box*
	Tambah data kota	*Black Box*
	Ubah data kota	*Black Box*
bentuk	Tampilkan data bentuk	*Black Box*
	tambah data bentuk	*Black Box*
	ubah data bentuk	*Black Box*

Data transaksi	Tampilkan data transaksi	*Black Box*
	Ubah status data transaksi	*Black Box*
	Lihat data detail transaksi	*Black Box*
Data kategori	Tampilkan data kategori	*Black Box*
	Tambah data kategori	*Black Box*
	Ubah data kategori	*Black Box*
Data produk	Tampilkan data produk	*Black Box*
	Tambah data produk	*Black Box*
	Ubah data produk	*Black Box*
Data produk	Tampilkan data produk	*Black Box*
	hapus data produk	*Black Box*

4.5.2 Kasus dan Hasil Pengujian

4.5.2.1 Pengujian Registrasi member

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

Tabel 4.6. Pengujian Registrasi *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik link registrasi	Tampil form pendaftaran *member*	Dapat menampilkan form pendaftaran *member*	[ ] Diterima [ ] Ditolak
Memasukan data pendaftaran, contoh masukan : User Naeme : herman Password :22051988 Nama : herman Alamat : Jl. Sapujagat Provinsi : bandung Kota : Bandung Kodepos : 40123 No Tlp : 0222507480 Email :phedot_munadi@yahoo.co.id	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut tidak boleh kosong.	Dapat memunculkan pesan kesalahan bahwa field tersebut tidak boleh kosong, jika salah satu field tidak terisi.	[ ] Diterima [ ] Ditolak
Klik tombol simpan	Jika salah satu field tidak terisi atau salah	Tampil pesan “Anda telah berhasil	[ ] Diterima [ ] Ditolak

	dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut tidak boleh kosong. Jika data masukan benar muncul pesan "Anda telah berhasil melakukan registrasi"	melakukan registrasi" jika data masukan benar dan dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	
Memasukan data kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[ ] Diterima [ ] Ditolak

4.5.2.2 Pengujian Login *Member*

Berikut dapat dilihat pengujian login *member* pada tabel 4.7 dibawah ini:

Tabel 4.7 Pengujian login *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan alamat *email* : phedot_munadi@yahoo.co.id	Mengisikan data alamat *email* pada *field* alamat *email*. Jika salah maka akan muncul pesan "invalid alamat email"	Dapat mengisi *field* alamat *email* dan muncul pesan "invalid alamat email" jika salah dalam pengisian data	[ ] Diterima [ ] Ditolak
Memasukan *Password* : 22051988	Mengisikan data *password* pada *field password*. Jika salah maka	Dapat mengisi *field password* dan muncul pesan "diantara 6	[√] Diterima [ ] Ditolak

	akan muncul pesan "diantara 6 sampai 11 karakter diperbolehkan"	sampai 11 karakter diperbolehkan" jika salah dalam pengisian data	
Klik tombol login	Data *email* dan data *password* dicari	Tombol login dapat berfungsi	[√] Diterima [ ] Ditolak
Memasukan data *email* atau *password* yang tidak terdaftar	Tidak dapat login	Muncul pesan "*email* atau *password* salah"	[√] Diterima [ ] Ditolak
mengosongkan data *email* atau *password*	Tidak dapat login	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

4.5.2.3 Pengujian Logout

Berikut dapat dilihat pengujian dari logout pada tabel 4.8. dibawah ini:

Tabel 4.8. Pengujian logout

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik logout	Dapat keluar dan menghapus session pengguna	*User* keluar dari aplikasi	[√] Diterima [ ] Ditolak

4.5.2.4 Pengujian Ubah profil member

Berikut dapat dilihat pengujian dari ubah profil member pada tabel 4.9. dibawah ini:

Tabel 4.9. Pengujian ubah profil member

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik ubah profil member	Menampilkan *form* ubah profil	Dapat menampilkan *form* ubah profil	[√] Diterima [ ] Ditolak
Data profil yang akan diubah	Tercantum pada text box data yang akan diubah	Dapat mengisi *field* data yang akan diubah dengan benar	[√] Diterima [√] Ditolak
Klik ubah	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[√] Diterima [ ] Ditolak
Mengosongkan salah satu fied	Menampilkan pesan kesalahan bahwa field tersebut harus terisi.	Dapat menampilkan pesan kesalahan bahwa field tersebut harus terisi.	[√] Diterima [ ] Ditolak
Mengosongkan semua field	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

4.5.2.5 Pengujian ganti *password member*

Berikut dapat dilihat pengujian dari ganti *password member* pada tabel 4.10. dibawah ini:

Tabel 4.10. Pengujian ganti *password member*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik ubah *password*	Menampilkan *form* ubah *password*	Dapat menampilkan *form* ubah *password*	[√] Diterima [ ] Ditolak
Memasukan data yang	*Password* baru	Dapat	[√] Diterima

benar	berhasil disimpan di *database*	menyimpan data *password* yang baru dengan muncul pesan “ Data berhasil diubah”	[ ] Ditolak
Mengosongkan salah satu field	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi” bila ada field yang belum diisi	[√] Diterima [ ] Ditolak
Data *password* lama salah	muncul pesan "password salah"	Dapat menampilkan pesan kesalahan ”*Password* salah”	[√] Diterima [ ] Ditolak
Data *password* baru dah konfirmasi *password* baru tidak sama	Muncul pesan "password baru dan konfirmasi password harus sama"	Dapat menampilkan pesan "password baru dan konfirmasi password harus sama"	[√] Diterima [ ] Ditolak
Mengosongkan semua field	Muncul pesan kesalahan field harus diisi	Dapat menampilkan pesan kesalahan field harus diisi	√] Diterima [ ] Ditolak

4.5.2.6 Pengujian menu produk

Berikut dapat dilihat pengujian dari menu produk pada tabel 4.11. dibawah ini:

Tabel 4.11. Pengujian menu produk

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu produk	Menampilkan data produk yang ada di *database*	Dapat menampilkan data produk yang ada di *data base*	[√] Diterima [ ] Ditolak
Klik detail pada data produk	Menampilkan detail produk	Dapat menampilkan	[√] Diterima [ ] Ditolak

		detail produk	
Klik pesan	Menuju ke keranjang belanja	Dapat menampilkan keranjang belanja	[√] Diterima [ ] Ditolak

4.5.2.7 Pengujian menu profil

Berikut dapat dilihat pengujian dari menu profil pada tabel 4.12. dibawah ini:

Tabel 4.12. Pengujian menu profil

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Profil	Menampilkan menu profil	Tampil menampilkan menu profil	[√] Diterima [ ] Ditolak

4.5.2.8 Pengujian menu info

Berikut dapat dilihat pengujian dari menu info pada tabel 4.13 dibawah ini:

Tabel 4.13. Pengujian menu info

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik info	Menampilkan menu info	Tampil menampilkan menu info	[√] Diterima [ ] Ditolak
Klik selangkapnya	Menampilkan menu info selangkapnya	Menampilkan menu info selengkapnya	[√] Diterima [ ] Ditolak

4.5.2.9 Pengujian menu kontak

Berikut dapat dilihat pengujian dari menu info pada tabel 4.14 dibawah ini:

Tabel 4.14. Pengujian menu kontak

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan

Klik kontak	Menampilkan menu kontak	Tampil menampilkan menu kontak	[√] Diterima [ ] Ditolak

4.5.2.10 Pengujian menu cara pemesanan

Berikut dapat dilihat pengujian dari menu cara pemesanan pada tabel 4.15. dibawah ini:

Tabel 4.15. Pengujian menu cara pemesanan

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik cara pemesanan	Menampilkan menu cara pemesanan	Dapat menampilkan form cara pemesanan	[√] Diterima [ ] Ditolak

4.5.2.11 Pengujian menu kategori

Berikut dapat dilihat pengujian dari menu kategori pada tabel 4.16. dibawah ini:

Tabel 4.16. Pengujian menu kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu kategori	Menampilkan data produk yang ada di *database*	Dapat menampilkan data produk yang ada di*data base*	[√] Diterima [ ] Ditolak
Klik detail pada data produk	Menampilkan detail produk	Dapat menampilkan detail produk	[√] Diterima [ ] Ditolak
Klik pesan	Menuju ke keranjang belanja	Dapat menampilkan keranjang belanja	[√] Diterima [ ] Ditolak

4.5.2.12 Pengujian menu keranjang belanja

Berikut dapat dilihat pengujian dari menu keranjang belanja pada tabel 4.17. dibawah ini:

Tabel 4.17. Pengujian meu *shopping cart*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu keranjang belanja	Menampilkan *form* keranjang belanja	Dapat menampilkan *form* keranjang belanja	[√] Diterima [ ] Ditolak
Menampilkan data jumlah pesanan	Tercantum pada text box data yang akan diubah	Dapat mengisi data jumlah dengan benar	[√] Diterima [ ] Ditolak
Klik tombol *edit*	Data jumlah yang diubah tersimpan *didatabase*	Dapat menampilkan perubahan data jumlah	[√] Diterima [ ] Ditolak
Klik tombol hapus	Data yang dihapus terhapus dari keranjang belanja	Dapat menghapus data pada keranjang belanja	[√] Diterima [ ] Ditolak
Klik *checkout*	Menampilkan form pengiriman	Dapat menampilkan form pengiriman	[√] Diterima [ ] Ditolak

4.5.2.13 Pengujian *form* pengiriman

Berikut dapat dilihat pengujian dari form pengiriman pada tabel 4.18. dibawah ini:

Tabel 4.18. Pengujian form pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data yang benar lalu mengklik tombol ganti	Data pengirimah baru akan disimpan di dalam *database*	Dapat data pengirimah baru akan disimpan di	[√] Diterima [ ] Ditolak

		dalam *database*	
Mengosongkan salah satu *field*	Menampilkan pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi”	[√] Diterima [ ] Ditolak
Klik tombol batal	Kembali ke menu sebelumnya	Dapat kembali ke menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.14 Pengujian *form checkout*

Berikut dapat dilihat pengujian dari form *checkout* pada tabel 4.19 dibawah ini:

Tabel 4.19 Pengujian *form checkout*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik *checkout*	Data transaksi dan detail transaksi tersimpan di *database*, Menampilkan *form* terimakasih	Dapat menyimpan data transaksi, dapat menampilkan *form* terima kasih	[√] Diterima [ ] Ditolak

4.5.2.15 Pengujian *form* konfirmasi pembayaran

Berikut dapat dilihat pengujian dari *form* konfirmasi pembayaran pada tabel 4.20 dibawah ini:

Tabel 4.20 Pengujian *form* konfirmasi pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Masukan data konfirmasi pembayaran, contoh No. invoice : inv-123321456 No. transfer : 24 Nama Bank : BCA Atas Nama : rahman Jumlah Transfer : 100000	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Mengosongkan semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi”	[√] Diterima [ ] Ditolak

4.5.2.16 Pengujian menu riwayat

Berikut dapat dilihat pengujian dari riwayat pada tabel 4.21. dibawah ini:

Tabel 4.21. Pengujian menu riwayat

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu riwayat	Menampilkan riwayat	Dapat menampilkan form riwayat	[√] Diterima [ ] Ditolak
Klik detail	Menampilkan	Dapat	[√] Diterima

	detail transaksi	menampilkan detail transaksi	[ ] Ditolak

4.5.2.17 Pengujian Login Administrator

Berikut dapat dilihat pengujian login administrator pada tabel 4.22 dibawah ini:

Tabel 4.22 Pengujian login administrator

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
username : garuci	Tercantum pada text box *username*	Dapat mengisi field *username* dengan benar	[√] Diterima [ ] Ditolak
Password : garuci	Tercantum pada text box *password*	Dapat mengisi field *password* dengan benar	[√] Diterima [ ] Ditolak
Klik tombol login	Data *username* dan data *password* dicari	Tombol login dapat berfungsi	[√] Diterima [ ] Ditolak
Memasukan data *username* atau *password* yang tidak terdaftar	Tidak dapat login	Muncul pesan "Tidak Sama Untuk Username Dan/Atau Password."	[√] Diterima [ ] Ditolak
Tidak memasukan data *username* atau *password*	Tidak dapat login	Dapat menampilkan pesan kesalahan "Tidak Sama Untuk Username Dan/Atau Password."	[√] Diterima [ ] Ditolak

4.5.2.18 Pengujian ganti *password* administrator

Berikut dapat dilihat pengujian dari ganti *password* administrator pada tabel 4.23 dibawah ini:

Tabel 4.23 Pengujian ganti *password* administrator

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik ubah *password*	Menampilkan *form* ubah *password*	Dapat menampilkan *form* ubah *password*	[√] Diterima [ ] Ditolak
Memasukan data yang benar	*Password* baru berhasil disimpan di *database*	Dapat menyimpan data *password* yang baru dengan muncul pesan “ Data berhasil diubah”	[√] Diterima [ ] Ditolak
Mengosongkan salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi” bila ada field yang belum diisi	[√] Diterima [ ] Ditolak
Memasukan *password* yang salah	muncul pesan "password salah"	Dapat menampilkan pesan kesalahan ”*Password* salah”	[√] Diterima [ ] Ditolak
Memasukan data *password* baru dan konfirmasi *password* baru yang tidak sama	Muncul pesan "password baru dan konfirmasi password harus sama"	Dapat menampilkan pesan "password baru dan konfirmasi password harus sama"	[√] Diterima [ ] Ditolak
Mengosongkan semua *field*	Muncul pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan *field* harus diisi	√] Diterima [ ] Ditolak

4.5.2.19 Pengujian data pengiriman

Berikut dapat dilihat pengujian data pengiriman pada tabel 4.24 dibawah ini:

Tabel 4.24 Pengujian data pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik pengiriman	Menampilkan data kota pengiriman	Dapat menampilkan data kota pengiriman	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah kota pengiriman	Dapat menampilkan *form* kota pengiriman	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.20 Pengujian tambah pengiriman

Berikut dapat dilihat pengujian dari tambah pengiriman pada tabel 4.25 dibawah ini:

Tabel 4.25 Pengujian tambah pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik baru	Menampilkan *form* tambah pengiriman	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data pengiriman Contoh: Provinsi : Jawa Barat Kota : Cimahi harga: 1000	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.21 Pengujian data provinsi

Berikut dapat dilihat pengujian dari data provinsi pada tabel 4.26 dibawah ini:

Tabel 4.26 Pengujian data provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik provinsi	Menampilkan data provinsi	Dapat menampilkan data provinsi	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah provinsi	Dapat menampilkan *form* ubah provinsi	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.22 Pengujian tambah provinsi

Berikut dapat dilihat pengujian dari tambah provinsi pada tabel 4.27 dibawah ini:

Tabel 4.27 Pengujian tambah provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah provinsi	Menampilkan *form* tambah provinsi	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data provinsi Contoh: Nama Provinsi : Jawa Barat	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.23 Pengujian data kota

Berikut dapat dilihat pengujian dari data kota pada tabel 4.28 dibawah ini:

Tabel 4.28 Pengujian data kota

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik kota	Menampilkan data kota	Dapat menampilkan data kota	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah kota	Dapat menampilkan *form* ubah kota	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.24 Pengujian tambah kota

Berikut dapat dilihat pengujian dari tambah kota pada tabel 4.29 dibawah ini:

Tabel 4.29 Pengujian tambah kota

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik baru	Menampilkan *form* tambah kota	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data kota Contoh: Provinsi : Jawa Barat Nama Kota : Subang	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.25 Pengujian data bentuk

Berikut dapat dilihat pengujian dari data bentuk pada tabel 4.30 dibawah ini:

Tabel 4.30 Pengujian data bentuk

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik bentuk	Menampilkan data bentuk	Dapat menampilkan data bentuk	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah bentuk	Dapat menampilkan *form* ubah bentuk	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.26 Pengujian tambah bentuk

Berikut dapat dilihat pengujian dari tambah bentuk pada tabel 4.31 dibawah ini:

Tabel 4.31 Pengujian tambah bentuk

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah bentuk	Menampilkan *form* tambah bentuk	Dapat menampilkan *form* tambah bentuk	[√] Diterima [ ] Ditolak
Mengisi data bentuk Contoh: Kode : S Bentuk : Small	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.27 Pengujian data kategori

Berikut dapat dilihat pengujian dari lihat kategori pada tabel 4.32 dibawah ini:

Tabel 4.32 Pengujian data kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik kategori	Menampilkan *form* lihat kategori	Dapat menampilkan *form* lihat kategori	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah kategori	Dapat menampilkan *form* ubah kategori	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.28 Pengujian tambah kategori

Berikut dapat dilihat pengujian dari tambah kategori pada tabel 4.33 dibawah ini:

Tabel 4.33 Pengujian tambah kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik baru	Menampilkan *form* tambah kategori	Dapat menampilkan *form* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data kategori Contoh: Nama kategori: Baju	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.29 Pengujian data produk

Berikut dapat dilihat pengujian dari data produk pada tabel 4.34 dibawah ini:

Tabel 4.34 Pengujian data produk

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik produk	Menampilkan data produk	Dapat menampilkan data produk	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah produk	Dapat menampilkan *form* ubah produk	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.30 Pengujian tambah produk

Berikut dapat dilihat pengujian dari tambah produk pada tabel 4.35 dibawah ini:

Tabel 4.35 Pengujian tambah produk

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik baru	Menampilkan *form* tambah produk	Dapat menampilkan *form* tambah produk	[√] Diterima [ ] Ditolak
Mengisi data produk Contoh: kategori: T-Shirt kode : B-020 Nama : Isthisit Blank T-Shirt Keterangnan : baju polos Bentuk : S Jumlah : 20 Berat : 700 Harga : 80000 Gambar :	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.31 Pengujian data member

Berikut dapat dilihat pengujian dari data member pada tabel 4.36 dibawah ini:

Tabel 4.36 Pengujian lihat member

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik member	Menampilkan data member	Dapat menampilkan data member	[√] Diterima [ ] Ditolak
Klik *check box* pada data yang akan dihapus hapus	Memilih data yang akan dihapus	Data yang akan dihapus telah dipilih	[√] Diterima [ ] Ditolak
Klik hapus	Data akan dihapus dari *database*. Jika tidak ada data yang dipilih akan muncul pesan kesalahan	Data telah dihapus dari database. Muncul pesan "pilih data yang akan dihapus" bila data belum dipilih	[√] Diterima [ ] Ditolak

4.5.2.32 Pengujian data info

Berikut dapat dilihat pengujian dari data info pada tabel 4.37 dibawah ini:

Tabel 4.37 Pengujian lihat info

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik info	Menampilkan *form* lihat info	Dapat menampilkan *form* lihat info	[√] Diterima [ ] Ditolak
Klik *check box* pada data yang akan dihapus hapus	Memilih data yang akan dihapus	Data yang akan dihapus telah dipilih	[√] Diterima [ ] Ditolak
Klik hapus	Data akan dihapus dari *database*. Jika tidak ada data yang	Data telah dihapus dari database.	[√] Diterima [ ] Ditolak

	dipilih akan muncul pesan kesalahan	Muncul pesan "pilih data yang akan dihapus" bila data belum dipilih	
Klik ubah	Menampilkan *form* ubah info	Dapat menampilkan *form* ubah info	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "ada kesalahan didalam pengisian data, silahkan cek kembali"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.33 Pengujian tambah info

Berikut dapat dilihat pengujian dari tambah info pada tabel 4.38 dibawah ini:

Tabel 4.38 Pengujian tambah info

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik baru	Menampilkan *form* tambah info	Dapat menampilkan *form* tambah	[√] Diterima [ ] Ditolak

		info	
Mengisi data info Contoh: judul : diskon ringkasan : diskon kecil kecilan isi : saat ini ada diskon lho	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.34 Pengujian data transaksi

Berikut dapat dilihat pengujian dari lihat transaksi pada tabel 4.39 dibawah ini:

Tabel 4.39 Pengujian lihat transaksi

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang	Yang diharapkan	Pengamatan	Kesimpulan

dilakukan			
Klik transaksi	Menampilkan *form* transaksi	Dapat menampilkan *form* lihat transaksi	[√] Diterima [ ] Ditolak
Klik tabel transaksi pemesanan baru	Menampilkan *form* transaksi pemesanan baru	Dapat menampilkan *form* transaksi pemesanan baru	[√] Diterima [ ] Ditolak
Klik tabel transaksi pembayaran	Menampilkan *form* transaksi pembayaran	Dapat menampilkan *form* transaksi pembayaran	[√] Diterima [ ] Ditolak
Klik tabel transaksi pengiriman	Menampilkan *form* transaksi pengiriman	Dapat menampilkan *form* transaksi pengiriman	[√] Diterima [ ] Ditolak
Klik tabel transaksi berhasil	Menampilkan *form* transaksi berhasil	Dapat menampilkan *form* transaksi berhasil	[√] Diterima [ ] Ditolak
Klik tabel transaksi gagal	Menampilkan *form* transaksi gagal	Dapat menampilkan *form* transaksi gagal	[√] Diterima [ ] Ditolak
Klik tombol pembayaran pada tab transaksi pemesanan baru	Ubah status transaksi pada tabel transaksi di *database* menjadi telah dibayar	Data status transaksi telah diubah di *database* menjadi telah dibayar	[√] Diterima [ ] Ditolak
Klik tombol detail pada tab transaksi pemesanan baru	Menampilkan *form* detail transaksi	Dapat menampilkan *form* detail transaksi	[√] Diterima [ ] Ditolak
Klik tombol pengiriman pada tab transaksi pembayaran	Ubah status transaksi pada tabel transaksi di *database* menjadi telah dikirim	Data status transaksi telah diubah di *database* menjadi telah dikirim	[√] Diterima [ ] Ditolak
Klik tombol detail pada tab transaksi pembayaran	Menampilkan *form* detail transaksi	Dapat menampilkan *form* detail transaksi	[√] Diterima [ ] Ditolak

Klik tombol berhasil pata tab transaksi pengiriman	Ubah status transaksi pada tabel transaksi di *database* menjadi telah terkirim	Data status transaksi telah diubah di *database* menjadi telah terkirim	[√] Diterima [ ] Ditolak
Klik tombol detail pata tab transaksi pengiriman	Menampilkan *form* detail transaksi	Dapat menampilkan *form* detail transaksi	[√] Diterima [ ] Ditolak

4.5.3 Kesimpulan Pengujian Alpha

Berdasarkan hasi pengujian alpha (fungsional) dengan kasus uji diatas dapat ditarik kesimpulan bahwa pembangunan aplikasi *e-commerce* berbasis web pada Dela Jaya Variasi bebas kesalahan penulisan dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

4.6 Pengujian Betha

Pengujian betha merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan membuat kuisoner mengenai kepuasan *user* dengan kandungan poin syarat *user friendly* untuk selanjutnya dibagikan kepada *user* dengan mengambil *sample* sebanyak 5 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penelitian penerapan sistem yang baru.

Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung di tempat penelitian. Penelitian ini dilakukan di dua lingkungan :

1. Lingkungan Penjual di Dela jaya variasi

   Penelitian dilingkungan penjual di Dela jaya variasi dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan diawal.

2. Lingkungan Masyarakat Umum

   Penelitian di lingkungan pada masyarakat umum khususnya yang sudah terbiasa dengan teknologi *internet*, dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat memberikan kemudahan masyarakat umum untuk melakukan transaksi pembelian barang secara *online*.

4.6.1 Kuesioner Pengguna

Kuesioner disebarkan menggunakan teknik sampling yaitu *Sample Random Sampling* yang disebarkan kepada 10 orang pengguna. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner ini terdiri dari 4 pertanyaan (contoh kuesioner dapat dilihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 10, dengan ketentuan skala untuk tiap pertanyaan berikut :

4.6.1.1 Kuesioner

1. Apakah tampilan web yang dibangun menarik?

   A. Sangat Menarik
   B. Menarik
   C. Cukup Menarik
   D. Biasa saja
   E. Kurang menarik
   F. Tidak menarik
   G. Sangat tidak menarik

2. Apakah tampilan web yang dibangun mudah dipergunakan?

   A. Sangat mudah digunakan
   B. Mudah digunakan
   C. Agak mudah digunakan
   D. Biasa saja
   E. Agak sulit digunakan
   F. Sulit digunakan
   G. Sangat sulit digunakan

3. Apakah intruksi yang disediakan mudah dipelajari?

   A. Sangat mudah dipelajari
   B. Mudah dipelajari
   C. Agak mudah dipelajari
   D. Biasa saja
   E. Agak sulit dipelajari
   F. Sulit dipelajari
   G. Sangat sulit dipelajari

4. Apakah informasi yang dihasilkan akurat sesuai dengan yang diinginkan?

   A. Sangat akurat
   B. Akurat
   C. Cukup akurat
   D. Biasa saja
   E. Agak kurang akurat
   F. Kurang akurat
   G. Sangat kurang akurat

Berdasarkan data hasil kuesioner tersebut, dapat dicari prosentasi masing-masing jawaban dengan menggunakan rumus : Y = P/Q * 100%

Keterangan :

P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai persentase

Berikut ini adalah hasil peresentasi masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuesinoer ini diujikan kepada 2 orang.

1. Bagaiman tampilan perangkat administrator untuk pengolahan data?

Tabel 4.40 Hasil Pengujian Kuesioner Soal Nomor 1

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Menarik	0	0
	2	Menarik	1	50
	3	Cukup Menarik	1	50
	4	Biasa saja	0	0
	5	Kurang menarik	0	0
	6	Tidak menarik	0	0
	7	Sangat tidak menarik	0	0
Jumlah			**2**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 1 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat bagus dan 1 orang atau 50% mengatakan perangkat lunak tersebut cukup bagus sehingga secara keseluruhan tampilan perangkat lunak tersebut cukup bagus.

2. Apakah anda mengerti dengan intruksi yang terdapat dalam *website* tersebut?

Tabel 4.41 Hasil Pengujian Kuesioner Soal Nomor 2

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat mudah digunakan	0	0
	2	Mudah digunakan	1	50
	3	Agak mudah digunakan	1	50
	4	Biasa saja	0	0
	5	Agak sulit digunakan	0	0
	6	Sulit digunakan	0	0

	7	Sangat sulit digunakan	0	0
Jumlah			**2**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 1 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat mudah digunakan, 1 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat cukup mudah digunakan. Sehinggan keseluruhan struktur perangkat lunak tersebut cukup mudah digunakan.

3. Apakah *website* ini dapat membantu dalam pengolahan data petugas?

Tabel 4.42 Hasil Pengujian Kuesioner Nomor 3

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat mudah dipelajari	0	0
	2	Mudah dipelajari	2	100
	3	Agak mudah dipelajari	0	0
	4	Biasa saja	0	0
	5	Agak sulit dipelajari	0	0
	6	Sulit dipelajari	0	0
	7	Sangat sulit dipelajari	0	0
Jumlah			**2**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 100% mengatakan bahwa perangkat lunak yang dibuat membantu. Sehingga secara keseluruhan perangkat lunak tersebut membantu.

4. Apakah *website* ini dapat membantu dalam pengolahan *member*?

Tabel 4.43 Hasil Pengujian Kuesioner Nomor 4

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
4	1	Sangat akurat	0	0
	2	Akurat	0	0
	3	Cukup akurat	2	100
	4	Biasa saja	0	0
	5	Agak kurang akurat	0	0
	6	Kurang akurat	0	0
	7	Sangat kurang akurat	0	0
Jumlah			**2**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 100% mengatakan bahwa perangkat lunak yang dibuat cukup membantu. Sehingga secara keseluruhan perangkat lunak tersebut cukup membantu.

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 100% mengatakan bahwa perangkat lunak yang dibuat membantu. Sehingga secara keseluruhan perangkat lunak tersebut membantu.

4.6.2 Kesimpulan Hasil Pengujian Beta

Berdasarkan hasil prosentase diatas yang didapatkan dari pengujian beta pengguna administrator, dan member, maka sistem penerapan *e-commerce* untuk sistem penjualan ini sudah sesuai dengan tujuan yaitu toko Dela Jaya Variasi dapat mempermudah dalam proses promosi dan penjualan produk yang ditawarkan, pelanggan dapat melakukan transaksi pembelian secara online, tampilannya *user friendly,* memudahkan dalam pengelolaan data barang dan memudahkan dalam membuat laporan penjualan.