BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Tahap implementasi sistem merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan menerjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dapat dimengerti oleh mesin serta penerapan perangkat lunak pada keadaan yang sesungguhnya.

4.1.1 Perangkat keras yang digunakan

Spesifikasi perangkat keras (*hardware*) yang digunakan dalam pembangunan aplikasi *E-Commerce* di toko Jellybox adalah tercantum pada tabel 4.1 sebagai berikut:

Tabel 4.1 Perangkat keras yang digunakan

Perangkat Keras	Spesifikasi
Processor	Dual Core T42000 @2.20GHZ
RAM	2.00 Gb
Harddisk	160 Gb
Monitor	LCD 14”

4.1.2 Perangkat Lunak yang digunakan

Spesifikasi perangkat lunak (*Software*) yang dipasang pada sistem komputer yang digunakan untuk mengembangkan sistem penerapan *E-commerce* untuk sistem penjualan ini adalah tercantum pada tabel 4.2 sebagai berikut :

Tabel 4.2 Perangkat lunak yang digunakan

Perangkat Lunak	Keterangan
Sistem Operasi	Windows XP profesional
PHP	Bahasa Pemrograman
MySQL	Penyimpanan *database*
PhpMyAdmin	*interface database*
XAMPP	*web server*
Macromedia Dreamweaver 8	Kode dan perancangan tampilan
Mozilla Firefox	*Web browser*

4.2 Implementasi Basis Data

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` (
  `id_admin` int(11) NOT NULL AUTO_INCREMENT,
  `nama_admin` varchar(50) NOT NULL,
  `alamat_admin` text NOT NULL,
  `email_admin` varchar(50) NOT NULL,
  `telp_admin` varchar(20) NOT NULL,
  `password_admin` varchar(32) NOT NULL,
  `verification_admin` varchar(32) NOT NULL,
  PRIMARY KEY (`id_admin`),
```

```
  UNIQUE KEY `email_admin` (`email_admin`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Administrator' AUTO_INCREMENT=2 ;
```

2. Tabel Barang

```
CREATE TABLE IF NOT EXISTS `barang` (
  `id_barang`  int(4)  unsigned  zerofill  NOT  NULL
AUTO_INCREMENT,
  `id_kategori` int(11) NOT NULL,
  `nama_barang` varchar(50) NOT NULL,
  `kode_barang` varchar(4) NOT NULL,
  `harga_barang` int(11) NOT NULL,
  `hargacustom_barang` int(11) NOT NULL,
  `hargagrosir_barang` float NOT NULL,
  `mingrosir_barang` int(11) NOT NULL,
  `deskripsi_barang` text NOT NULL,
  `diskon_barang` float NOT NULL,
  `rating_barang` float NOT NULL,
  `voterrating_barang` int(11) NOT NULL,
  `viewcounter_barang` int(11) NOT NULL,
  `type_barang` enum('standard','custom') NOT NULL,
  PRIMARY KEY (`id_barang`),
  KEY `id_kategori` (`id_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Barang' AUTO_INCREMENT=33 ;
ALTER TABLE `barang`
  ADD  CONSTRAINT  `barang_ibfk_1`  FOREIGN  KEY
(`id_kategori`) REFERENCES `kategori` (`id_kategori`);
```

3. Tabel Detail Barang

```
CREATE TABLE IF NOT EXISTS `detailbarang` (
  `id_detailbarang` int(11) NOT NULL AUTO_INCREMENT,
  `id_barang` int(3) unsigned zerofill NOT NULL,
  `id_warna` int(11) DEFAULT NULL,
  `id_ukuran` int(11) NOT NULL,
```

```
  `tanggal_detailbarang` datetime NOT NULL,
  `berat_detailbarang` float NOT NULL,
  `promo_detailbarang` enum('0','1') NOT NULL,
  PRIMARY KEY (`id_detailbarang`),
  KEY `id_barang` (`id_barang`),
  KEY `id_warna` (`id_warna`),
  KEY `id_ukuran` (`id_ukuran`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Detail Barang' AUTO_INCREMENT=60 ;
ALTER TABLE `detailbarang`
  ADD  CONSTRAINT  `detailbarang_ibfk_1`  FOREIGN  KEY
(`id_barang`) REFERENCES `barang` (`id_barang`),
  ADD  CONSTRAINT  `detailbarang_ibfk_2`  FOREIGN  KEY
(`id_warna`) REFERENCES `warna` (`id_warna`),
  ADD  CONSTRAINT  `detailbarang_ibfk_3`  FOREIGN  KEY
(`id_ukuran`) REFERENCES `ukuran` (`id_ukuran`);
```

4. Tabel Detail Pembelian

```
CREATE TABLE IF NOT EXISTS `detailpembelian` (
  `id_detailpembelian` int(11) NOT NULL AUTO_INCREMENT,
  `kodeunik_detailpembelian` int(11) NOT NULL,
  `tanggal_detailpembelian` datetime NOT NULL,
  `tanggalditerima_detailpembelian` datetime NOT NULL,
  `status_pengiriman`
enum('dipesan','dikonfirmasi','dibayar','dikirim','dite
rima') NOT NULL,
  `keterangan_detailpembelian` text NOT NULL,
  `nama_pemesan` varchar(50) NOT NULL,
  `alamat_pemesan` text NOT NULL,
  `email_pemesan` varchar(50) NOT NULL,
  `no_telp_pemesan` varchar(20) NOT NULL,
  `kodepos_pemesan` varchar(6) NOT NULL,
  `kota_pemesan` varchar(100) NOT NULL,
  `provinsi_pemesan` varchar(100) NOT NULL,
```

```
  `no_resi_pemesan` varchar(50) NOT NULL,
  `nama_penerimabarang` varchar(100) NOT NULL,
  `tanggalditerima_barang` varchar(100) NOT NULL,
  `totalpembayaran_detailpembelian` float NOT NULL,
  `pengiriman_detailpembelian` varchar(200) NOT NULL,
  `jenispengiriman_detailpembelian`  varchar(100)  NOT
NULL,
  `hargaongkir_detailpembelian` float NOT NULL,
  `status_pemesanan` enum('ok','cancel') NOT NULL,
  `jenis_detailpembelian`  enum('normal','custom')  NOT
NULL,
  `session_id` varchar(32) NOT NULL,
  PRIMARY KEY (`id_detailpembelian`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
detailpembelian' AUTO_INCREMENT=54 ;
```

5. Tabel Gambar

```
CREATE TABLE IF NOT EXISTS `gambar` (
  `id_gambar` int(11) NOT NULL AUTO_INCREMENT,
  `id_barang` int(4) unsigned zerofill NOT NULL,
  `path_gambar` varchar(200) NOT NULL,
  `profile_gambar` enum('0','1') NOT NULL,
  PRIMARY KEY (`id_gambar`),
  KEY `id_barang` (`id_barang`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Gambar' AUTO_INCREMENT=31 ;
ALTER TABLE `gambar`
  ADD   CONSTRAINT   `gambar_ibfk_1`   FOREIGN   KEY
(`id_barang`) REFERENCES `barang` (`id_barang`);
```

6. Tabel Gambar Upload

```
CREATE TABLE IF NOT EXISTS `gambarupload` (
  `id_gambarupload` int(11) NOT NULL AUTO_INCREMENT,
  `id_pembelian` int(11) DEFAULT NULL,
```

```
  `path_gambarupload` varchar(200) NOT NULL,
  `session_gambarupload` varchar(50) NOT NULL,
  PRIMARY KEY (`id_gambarupload`),
  KEY `id_pembelian` (`id_pembelian`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel Upload Gambar' AUTO_INCREMENT=5 ;

ALTER TABLE `gambarupload`
  ADD CONSTRAINT `gambarupload_ibfk_1` FOREIGN KEY (`id_pembelian`) REFERENCES `pembelian` (`id_pembelian`);
```

7. Tabel Hubungi

```
CREATE TABLE IF NOT EXISTS `hubungi` (
  `id_hubungi` int(11) NOT NULL AUTO_INCREMENT,
  `id_admin` int(11) DEFAULT NULL,
  `nama_hubungi` varchar(100) NOT NULL,
  `email_hubungi` varchar(50) NOT NULL,
  `telp_hubungi` varchar(20) NOT NULL,
  `isi_hubungi` text NOT NULL,
  `tanggal_hubungi` datetime NOT NULL,
  PRIMARY KEY (`id_hubungi`),
  KEY `id_admin` (`id_admin`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Tabel Hubungi Kami' AUTO_INCREMENT=1 ;

ALTER TABLE `hubungi`
  ADD CONSTRAINT `hubungi_ibfk_1` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
```

8. Tabel Jasa Pengiriman

```
CREATE TABLE IF NOT EXISTS `jasapengiriman` (
  `id_jasapengiriman` int(11) NOT NULL AUTO_INCREMENT,
  `nama_jasapengiriman` varchar(100) NOT NULL,
  `deskripsi_jasapengiriman` text NOT NULL,
  PRIMARY KEY (`id_jasapengiriman`)
```

```
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Jasa Pengiriman' AUTO_INCREMENT=4 ;
```

9. 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(50) NOT NULL,
  `deskripsi_jenispengiriman` text NOT NULL,
  PRIMARY KEY (`id_jenispengiriman`),
  KEY `id_jasapengiriman` (`id_jasapengiriman`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Jenis Pengiriman' AUTO_INCREMENT=6 ;
ALTER TABLE `jenispengiriman`
  ADD CONSTRAINT `jenispengiriman_ibfk_1` FOREIGN KEY
(`id_jasapengiriman`)    REFERENCES    `jasapengiriman`
(`id_jasapengiriman`);
```

10. Tabel Kategori

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` int(11) NOT NULL AUTO_INCREMENT,
  `nama_kategori` varchar(50) NOT NULL,
  `deskripsi_kategori` text NOT NULL,
  PRIMARY KEY (`id_kategori`),
  UNIQUE KEY `nama_kategori` (`nama_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Kategori' AUTO_INCREMENT=5 ;
```

11. **Tabel Komplain**

```
CREATE TABLE IF NOT EXISTS `komplain` (
  `id_komplain` int(11) NOT NULL AUTO_INCREMENT,
  `nama_komplain` text NOT NULL,
  `deskripsi_komplain` text NOT NULL,
  PRIMARY KEY (`id_komplain`)
```

```
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Komplain' AUTO_INCREMENT=3 ;
```

12. Tabel Konfirmasi Pembayaran

```
CREATE TABLE IF NOT EXISTS `konfirmasipembayaran` (
  `id_konfirmasipembayaran`     int(11)     NOT     NULL
AUTO_INCREMENT,
  `id_detailpembelian` int(11) NOT NULL,
  `payment_gateway` enum('TRANSFER','PAYPAL','COD') NOT
NULL,
  `bank_rekening` varchar(50) DEFAULT NULL,
  `no_rekening` varchar(50) DEFAULT NULL,
  `instansi_pembayaran` varchar(25) NOT NULL,
  `no_transaksi` varchar(25) NOT NULL,
  `totalbayar_konfirmasipembayaran` int(11) NOT NULL,
  `kurs_konfirmasipembayaran` float DEFAULT NULL,
  `tanggal_konfirmasipembayaran` datetime NOT NULL,
  PRIMARY KEY (`id_konfirmasipembayaran`),
  KEY `id_detailpembelian` (`id_detailpembelian`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Konfirmasi' AUTO_INCREMENT=31 ;
```

13. Tabel Kota

```
CREATE TABLE IF NOT EXISTS `kota` (
  `id_kota` int(10) NOT NULL AUTO_INCREMENT,
  `id_provinsi` int(10) DEFAULT NULL,
  `nama_kota` varchar(50) DEFAULT NULL,
  `kabkota` varchar(20) DEFAULT NULL,
  `cod_kota` enum('0','1') NOT NULL,
  UNIQUE KEY `kota#PX` (`id_kota`),
  KEY `id_provinsi` (`id_provinsi`)
)     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=425 ;
```

ALTER TABLE `kota`

```
 ADD CONSTRAINT `kota_ibfk_1` FOREIGN KEY (`id_provinsi`) REFERENCES `provinsi` (`id_provinsi`);
```

14. Tabel Member

```
CREATE TABLE IF NOT EXISTS `member` (
  `id_member` int(11) NOT NULL AUTO_INCREMENT,
  `id_kota` int(11) NOT NULL,
  `nama_member` varchar(50) NOT NULL,
  `alamat_member` text NOT NULL,
  `ttl_member` date NOT NULL,
  `telp_member` varchar(20) NOT NULL,
  `kodepos_member` varchar(6) NOT NULL,
  `email_member` varchar(50) NOT NULL,
  `password_member` varchar(32) NOT NULL,
  `verificationcode_member` varchar(32) NOT NULL,
  `status_member` enum('0','1') NOT NULL,
  PRIMARY KEY (`id_member`),
  UNIQUE KEY `email_member` (`email_member`),
  KEY `id_kota` (`id_kota`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel Member' AUTO_INCREMENT=9 ;

ALTER TABLE `member`
  ADD CONSTRAINT `member_ibfk_1` FOREIGN KEY (`id_kota`) REFERENCES `kota` (`id_kota`);
```

15. Tabel Ongkir

```
CREATE TABLE IF NOT EXISTS `ongkir` (
  `id_ongkir` int(11) NOT NULL AUTO_INCREMENT,
  `id_kota` int(11) NOT NULL,
  `id_jenispengiriman` int(11) NOT NULL,
  `harga_ongkir` int(11) NOT NULL,
  PRIMARY KEY (`id_ongkir`),
  KEY `id_kota` (`id_kota`),
```

```
  KEY `id_jenispengiriman` (`id_jenispengiriman`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Ongkos Kirim' AUTO_INCREMENT=32 ;
ALTER TABLE `ongkir`
  ADD   CONSTRAINT   `ongkir_ibfk_1`   FOREIGN   KEY
(`id_kota`) REFERENCES `kota` (`id_kota`),
  ADD   CONSTRAINT   `ongkir_ibfk_2`   FOREIGN   KEY
(`id_jenispengiriman`)  REFERENCES  `jenispengiriman`
(`id_jenispengiriman`);
```

16. Tabel Pembelian

```
CREATE TABLE IF NOT EXISTS `pembelian` (
  `id_pembelian` int(11) NOT NULL AUTO_INCREMENT,
  `id_detailpembelian` int(11) DEFAULT NULL,
  `id_member` int(11) DEFAULT NULL,
  `id_detailbarang` int(11) DEFAULT NULL,
  `harga_temp` int(11) NOT NULL,
  `stok_temp` int(11) NOT NULL,
  `berat_temp` float NOT NULL,
  `diskon_temp` float NOT NULL,
  `typebarang_pembelian`  enum('normal','custom')  NOT
NULL DEFAULT 'normal',
  `typepesanan_pembelian`  enum('normal','grosir')  NOT
NULL DEFAULT 'normal',
  `session_id` varchar(32) NOT NULL,
  PRIMARY KEY (`id_pembelian`),
  KEY `id_detailpembelian` (`id_detailpembelian`),
  KEY `id_member` (`id_member`),
  KEY `id_detailbarang` (`id_detailbarang`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Pembelian' AUTO_INCREMENT=74 ;
ALTER TABLE `pembelian`
  ADD   CONSTRAINT   `pembelian_ibfk_1`   FOREIGN   KEY
(`id_detailpembelian`)  REFERENCES  `detailpembelian`
(`id_detailpembelian`),
```

```
  ADD   CONSTRAINT   `pembelian_ibfk_2`   FOREIGN   KEY
(`id_member`) REFERENCES `member` (`id_member`),
  ADD   CONSTRAINT   `pembelian_ibfk_3`   FOREIGN   KEY
(`id_detailbarang`)      REFERENCES      `detailbarang`
(`id_detailbarang`);
```

17. Tabel Provinsi

```
CREATE TABLE IF NOT EXISTS `provinsi` (
  `id_provinsi` int(10) NOT NULL AUTO_INCREMENT,
  `nama_provinsi` varchar(30) DEFAULT NULL,
  KEY `id_prov` (`id_provinsi`)
)     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=34 ;
```

18. Tabel Retur

```
CREATE TABLE IF NOT EXISTS `retur` (
CREATE TABLE IF NOT EXISTS `retur` (
  `id_retur` int(11) NOT NULL AUTO_INCREMENT,
  `id_pembelian` int(11) DEFAULT NULL,
  `id_komplain` int(11) DEFAULT NULL,
  `status_retur`
enum('dikonfirmasi','diterima','ditolak') NOT NULL,
  `tanggal_retur` datetime NOT NULL,
  PRIMARY KEY (`id_retur`),
  KEY `id_pembelian` (`id_pembelian`),
  KEY `id_komplain` (`id_komplain`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Retur Barang' AUTO_INCREMENT=3 ;
ALTER TABLE `retur`
  ADD    CONSTRAINT    `retur_ibfk_2`    FOREIGN    KEY
(`id_komplain`) REFERENCES `komplain` (`id_komplain`),
  ADD    CONSTRAINT    `retur_ibfk_3`    FOREIGN    KEY
(`id_pembelian`)         REFERENCES         `pembelian`
(`id_pembelian`);
```

19. Tabel Testimonial

```
CREATE TABLE IF NOT EXISTS `testimonial` (
  `id_testimonial` int(11) NOT NULL AUTO_INCREMENT,
  `id_member` int(11) NOT NULL,
  `id_pembelian` int(11) NOT NULL,
  `isi_testimonial` text NOT NULL,
  `tanggal_testimonial` datetime NOT NULL,
  `publish_testimonial` enum('0','1') NOT NULL,
  PRIMARY KEY (`id_testimonial`),
  KEY `id_member` (`id_member`),
  KEY `id_pembelian` (`id_pembelian`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Tabel
Testimonial' AUTO_INCREMENT=1 ;
```

20. Tabel Ukuran

```
CREATE TABLE IF NOT EXISTS `ukuran` (
  `id_ukuran` int(11) NOT NULL AUTO_INCREMENT,
  `nama_ukuran` varchar(10) NOT NULL,
  `deskripsi_ukuran` text NOT NULL,
  PRIMARY KEY (`id_ukuran`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Ukuran' AUTO_INCREMENT=7 ;
```

21. Tabel Warna

```
CREATE TABLE IF NOT EXISTS `warna` (
  `id_warna` int(11) NOT NULL AUTO_INCREMENT,
  `nama_warna` varchar(50) NOT NULL,
  `format_warna` varchar(10) DEFAULT NULL,
  PRIMARY KEY (`id_warna`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Warna' AUTO_INCREMENT=11 ;
```

22. Tabel Detail Retur

```
CREATE TABLE IF NOT EXISTS `detailretur` (
  `id_detailretur` int(11) NOT NULL AUTO_INCREMENT,
  `id_detailpembelian` int(11) NOT NULL,
  `no_resi_baru` varchar(100) NOT NULL,
  `nama_penerima` varchar(100) NOT NULL,
  `tanggalditerima_retur` varchar(100) NOT NULL,
  PRIMARY KEY (`id_detailretur`),
  KEY `id_detailpembelian` (`id_detailpembelian`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=4 ;
ALTER TABLE `detailretur`
  ADD CONSTRAINT `detailretur_ibfk_1` FOREIGN KEY
(`id_detailpembelian`) REFERENCES `detailpembelian`
(`id_detailpembelian`);
```

4.3 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna. Adapun untuk tampilan implementasi pembangunan aplikasi *E-commerce* di jellybox berbasis *Web* akan dilampirkan pada halaman lampiran.

Tabel 4.3 Implementasi antarmuka Pengunjung dan Member

Menu	Deskripsi	Nama File
Registarsi member	Digunakan untuk melakukan pendaftaran *member* baru.	Register.html
Login *member*	Digunakan Digunakan sebagai halaman login *member*	Account.html

Lupa *password*	Digunakan jika *member* lupa dengan *password*	forgotpassword.html
Cara Pembelian	Digunakan untuk melihat cara pembelian	Carapembelian.html
Setting Akun	Digunakan untuk mengubah profil *member.*	editakun.html
Ganti password	Digunakan jika *member* ingin merubah *password*	ubahpass.php
Kategori	Digunakan untuk melihat barang perkategori	kategori. Php
Home	Digunakan untuk melihat keseluruhan produk	Home.html
Lihat History	digunakan untuk melihat history pembelian,pemesanan	History.html
Hubungi Kami	Digunakan untuk melakukan tanya jawab secara langsung	hubungi.html
Keranjang belanja	Digunakan untuk melihat, merubah, menghapus data pembelian yang akan dilakukan.	cart.html
Konfirmasi pembayaran	Digunakan untuk melihat status pembayaran,dan membayar produk	Konfirmasi.html
Retur	Digunakan member untuk mengembalikan barang dalam kondisi rusak,sobek	Retur.html
Logout	Digunakan member untuk keluar dari akun	Logout.html

Tabel 4.4 Implementasi Antarmuka Admin

Menu	Deskripsi	Nama File
Login administrator	Digunakan sebagai halaman login administrator	index.php
Lupa Password	Digunakan jika admin lupa terhadap passwordnya	Forgotpassword.php
Pengaturan Akun	Digunakan jika admin ingin merubah profil	Editprofile.php
Pengolahan Data Admin	Digunakan untuk melihat,menambah dan menghapus admin	Admin.Php
Ubah Kata Sandi Admin	Digunakan jika administrator ingin merubah *password*	editpassword.Php
Kota	Digunakan untuk melihat, menambah, menghapus dan merubah data kota	Kota.Php
Provinsi	Digunakan untuk melihat, menambah, menghapus dan merubah data provinsi	provinsi.Php
Kategori	Digunakan untuk melihat, menambah, menghapus dan merubah data kategori	kategori.Php
Barang	Digunakan untuk melihat, menambah, menghapus dan merubah data barang	barang.Php
Jasa Pengiriman	Digunakan untuk melihat, menambah, menghapus dan merubah data Jasa Pengiriman	Jasapengiriman.Php
Jenis Pengiriman	Digunakan untuk melihat, menambah, menghapus dan merubah data Jenis Pengiriman	Jenispengiriman.Php

Member	Digunakan untuk melihat, menghapus, dan mengaktifkan status member	Member.Php
Rekening	Digunakan untuk melihat, menambah, menghapus dan merubah data rekening	Rekening.Php
Detail barang	Digunakan untuk melihat,menambah,memberikan warna dan ukuran barang	Detailbarang.php
Warna	Digunakan untuk menambah dan mengahapus,edit warna	Warna.php
Kurs	Digunakan untuk melihat, menambah, menghapus, merubah dan mengaktifkan status data kurs	Currency.Php
Ukuran	Digunakan untuk melihat,menambah,mengedit,menghapus ukuran	Ukuran.php
Data Transaksi	Digunakan untuk melihat data transaksi yang masuk dan merubah statusnya.	datatransaksi. Php
Konfirmasi Pembayaran	Digunakan untuk melihat data konfirmasi pembayaran	Konfirmasipembayaran.Php
Ongkos Kirim	Digunakan untuk melihat data onkos kirim, merubah, menghapus onkos kirim	ongkoskirim. Php
Laporan	Digunakan untuk melihat data laporan harian,bulanan maupun tahunan kemudian dicetak ke pdf	laporan.php
Hubungi	Digunakan untuk melihat data hubungi yang akan dibaca.	hubungi.php
Data Retur	Digunakan untuk menerima retur	Dataretur.php
Logout	Digunakan administrator Untuk dapat keluar dari aplikasi	logout.php

4.4 Pengujian Perangkat Lunak

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan kesalahan-kesalahan aatau 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 Alpha

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

4.5.1 Skenario Pengujian Alpha

Tabel 4.5 akan menjelaskan skenario pengujian aplikasi *member*

Tabel 4.5 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 Account	Menampilkan *form* ubah profil	*Black Box*
	Ubah data *costumer*	*Black Box*
	Simpan perubahan	*Black Box*

	Menampilkan *form* ubah *password*	*Black Box*
	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verifikasi konfirmasi *password*	*Black Box*
Menu Semua Produk	Menampilkan semua produk	*Black Box*
Menu kategori	Menampilkan produk sesuai pilihan kategori	*Black Box*
Menu Produk Terbaru	Menampilkan data produk terbaru	*Black Box*
Menu Produk Rekomendasi	Menampilkan data produk-produk rekomendasi	*Black Box*
Menu Produk Diskon	Menampilkan data produk-produk yang didiskon	*Black Box*
Menu Hubungi kami	Menampilkan kontak perusahaan	*Black Box*
Menu Cara Pembayaran	Menampilkan informasi cara pembayaran	*Black Box*
Menu Cara Pembelian	Menampilkan informasi cara pembelian	*Black Box*
Menu Cara Kontak	Menampilkan informasi cara kontak	*Black Box*
Menu Cara Upload Gambar	Menampilkan informasi cara upload gambar	*Black Box*
Menu Keranjang Belanja	Menampilkan data detail transaksi	*Black Box*
	Mengubah data detail transaksi	*Black Box*
	Menghapus data detail transaksi	*Black Box*
Selesai belanja	Menampilkan form *checkout*	*Black Box*
Form *checkout*	Menampilkan form *checkout*	*Black Box*
	Isi form pengiriman	*Black Box*
Menu Konfirmasi Pembayaran	Menampilkan konfirmasi pembayaran	*Black Box*
	Isi form konfimasi pembayaran	*Black Box*

Menu Lihat History	Menampilkan data transaksi yang pernah dilakukan	*Black Box*
	Menampilkan data detail tiap transaksi	*Black Box*
	Isi testimoni bila barang telah diterima	*Black Box*
Menu Retur	Menampilkan data transaksi yang telah diterima barangnya	*Black Box*
	Isi form barang yang akan di retur	*Black Box*

Tabel 4.6 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*
Edit Password	Menampilkan *form* ganti *password*	*Black Box*
	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verfikasi konfirmasi *password*	*Black Box*
	Simpan Perubahan	*Black Box*
Menu Pengaturan Akun	Menampilkan *form* ganti *profil*	*Black Box*
	Ubah data *user*	*Black Box*
	Simpan Perubahan	*Black Box*
Menu Ongkos Kirim	Tampilkan data harga kirim	*Black Box*
	Tambah data harga kirim	*Black Box*
	Ubah data harga kirim	*Black Box*
	Hapus data harga kirim	*Black Box*
Menu Provinsi	Tampilkan data provinsi	*Black Box*

	Tambah data provinsi	*Black Box*
	Ubah data provinsi	*Black Box*
	Hapus data provinsi	*Black Box*
Menu Kota	Tampilkan data kota	*Black Box*
	Tambah data kota	*Black Box*
	Ubah data kota	*Black Box*
	Hapus data kota	*Black Box*
Menu Ukuran	Tampilkan data ukuran	*Black Box*
	tambah data ukuran	*Black Box*
	ubah data ukuran	*Black Box*
	Hapus data ukuran	*Black Box*
Menu Jenis Pengiriman	Tampilkan data jenis pengiriman	*Black Box*
	Tambah data jenis pengiriman	*Black Box*
	Ubah data jenis pengiriman	*Black Box*
	Hapus data jenis pengiriman	*Black Box*
Menu Member	Tampilkan data Member	*Black Box*
	Ubah status member	*Black Box*
	Hapus data member	*Black Box*
Menu Kategori	Tampilkan data kategori	*Black Box*
	Tambah data kategori	*Black Box*
	Ubah data kategori	*Black Box*
Menu Warna	Tampilkan data warna	*Black Box*
	Tambah data warna	*Black Box*
	Ubah data warna	*Black Box*
	Hapus data warna	*Black Box*
Menu Barang	Tampilkan data barang	*Black Box*
	Tambah data barang	*Black Box*
	Ubah data barang	*Black Box*
	Hapus data produk	*Black Box*

Menu Detail Barang	Tampilkan data detail barang	*Black Box*
	Tambah data detail barang	*Black Box*
	Ubah data detail barang	*Black Box*
	Hapus Data detail barang	*Black Box*
Menu Retur	Tampilkan data barang yang diretur	*Black Box*
	Ubah status retur barang	*Black Box*
	Hapus data retur barang	*Black Box*
Menu Pengolahan Data Transaksi	Tampilkan data barang yang dipesan	*Black Box*
	Ubah status barang yang dipesan	*Black Box*
	Print data barang yang dipesan	*Black Box*
	Hapus data barang yang dipesan	*Black Box*
Menu Laporan Penjualan	Tampilkan data laporan penjualan	*Black Box*
	Tampilkan data detail laporan penjualan	*Black Box*
	Hapus data laporan penjualan	*Black Box*
	Print data laporan penjualan	*Black Box*
Menu Laporan Barang	Tampilkan data laporan barang	*Black Box*
	Tampilkan data detail laporan barang	*Black Box*
	Hapus data laporan barang	*Black Box*
	Print data laporan barang	*Black Box*
Menu Laporan Retur	Tampilkan data laporan retur	*Black Box*
	Tampilkan data detail laporan retur	*Black Box*
	Hapus data laporan retur	*Black Box*
	Print data laporan retur	*Black Box*

4.5.2 Kasus dan Hasil Pengujian

4.5.2.1 Pengujian Registrasi Member

Berikut dapat dilihat dari pengujian dari registrasi *member* yang terdapat pada tabel 4.7 :

Tabel 4.7 Pengujian Registrasi *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik menu registrasi *member*	Tampil form registrasi *member*	Dapat menampilkan form registrasi *member*	[√] Diterima [ ] Ditolak
Memasukan data pendaftaran, contoh masukan : Nama Saya : M. Kharisma Alamat Email : kharisma_semesta@yahoo.com Provinsi : DKI Jakarta Kota : Jakarta Utara Alamat : Jl. H Suit No. 35A Kodepos : 14130 Telepon : 082119959810 Password : jellybox Konfirmasi password : jellybox	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan 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
Klik tombol simpan	Jika salah satu field yang wajib di isi tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan	Tampil pesan "Pendaftaran berhasil, silahkan cek email Anda untuk mengkonfirmasi pendaftaran."	[√] Diterima [ ] Ditolak

	bahwa field tersebut harus terisi. Jika data masukan benar muncul pesan "Pendaftaran berhasil, silahkan cek email Anda untuk mengkonfirmasi pendaftaran"		
Memasukan data kosong selain di field yang wajib diisi	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 4.8 :

Tabel 4.8 Pengujian Login *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan alamat *email* : kharisma_semesta@yahoo.com	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* : jellybox	Mengisikan data *password* pada *field password*. Jika salah maka akan muncul pesan "Maaf, username dan password salah!!"	Dapat mengisi *field password* dan muncul pesan "Maaf, username dan password salah!!" jika salah dalam pengisian data	[√] Diterima [ ] Ditolak
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 "Maaf, Username dan Password Salah!!!"	[√] Diterima [ ] Ditolak
mengosongkan data *email* atau *password*	Tidak dapat login	Dapat menampilkan pesan kesalahan " Maaf, Username dan Password Salah!!!"	[√] Diterima [ ] Ditolak

4.5.2.3 Pengujian Logout

Berikut dapat dilihat pengujian dari logout pada tabel 4.9 :

Tabel 4.9 Pengujian Logout

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

	pengguna		

4.5.2.4 Pengujian Setting Akun

Berikut dapat dilihat pengujian dari ubah profil *member* pada tabel 4.10:

Tabel 4.10 Pengujian Setting Akun

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Klik edit account	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 "Akun berhasil diubah"	[√] Diterima [ ] Ditolak
Mengosongkan salah satu fied yang wajib di isi	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 4.11:

Tabel 4.11 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 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

Data *password* lama salah	muncul pesan "password salah"	Dapat menampilkan pesan kesalahan ”*Password* salah”	[√] Diterima [ ] Ditolak
Data *password* baru dan 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 menu produk pada tabel 4.12:

Tabel 4.12 Pengujian Menu Produk

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

	terbaru yang ada di *database*	data produk terbaru yang ada di *data base*	
Klik menu produk rekomendasi	Menampilkan data produk rekomendasi yang ada di *database*	Dapat menampilkan data produk rekomendasi yang ada di *data base*	[√] Diterima [ ] Ditolak
Klik menu produk diskon	Menampilkan data produk yang didiskon yang ada di *database*	Dapat menampilkan data produk yang didiskon yang ada di *data base*	[√] Diterima [ ] Ditolak
Klik gambar atau nama produk	Menampilkan detail produk	Dapat menampilkan detail produk	[√] Diterima [ ] Ditolak
Klik add to cart	Menuju ke keranjang belanja	Dapat menampilkan keranjang belanja	[√] Diterima [ ] Ditolak

4.5.2.7 Pengujian Menu Cara Pembelian

Berikut dapat dilihat pengujian dari cara pembelian pada tabel 4.13 :

Tabel 4.13 Pengujian Cara Pembelian

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

4.5.2.8 Pengujian Menu Hubungi

Berikut dapat dilihat pengujian dari menu hubungi kami pada tabel 4.14 :

Tabel 4.14 Pengujian Menu Hubungi

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik hubungi kami	Menampilkan form yang harus diisi untuk hubungi kami	Tampil menampilkan form yang harus di isi untuk hubungi kami	[√] Diterima [ ] Ditolak
Mengisikan alamat email yang tidak terdaftar	Muncul pesan "gagal mengirim, pastikan email Anda valid"	Dapat menampilakan pesan "gagal mengirim, pastikan email Anda valid"	[√] Diterima [ ] Ditolak
Mengosongkan semua field	Muncul pesan "Harus diisi"	Dapat menampilkan pesan "Harus diisi"	[√] Diterima [ ] Ditolak
Klik kirim	Data berhasil dikirim ke admin dan tampil pesan "Data berhasil dikirim"	Data berhasil dikirim ke admin dan menampilkan pesan "Data berhasil dikirim"	

4.5.2.9 Pengujian Menu Retur

Berikut dapat dilihat pengujian dari menu retur pada tabel 4.15 :

Tabel 4.15 Pengujian Menu Retur

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

4.5.2.10 Pengujian Menu Cara Pembayaran

Berikut dapat dilihat pengujian dari menu cara pembayaran pada tabel 4.16 :

Tabel 4.16 Pengujian Menu Cara Pembayaran

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

4.5.2.11 Pengujian Menu Kategori

Berikut dapat dilihat pengujian dari menu kategori pada tabel 4.17 :

Tabel 4.17 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 4.18:

Tabel 4.18 Pengujian Menu Keranjang Belanja

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik 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 ubah	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 tombol lanjutkan	Kembali ke	Dapat kembali	[√] Diterima

belanja	halaman home	ke halaman home	[ ] Ditolak
Klik selesai belanja	Menampilkan form pengiriman	Dapat menampilkan form *checkout*	[√] Diterima [ ] Ditolak

4.5.2.13 Pengujian *Form* Isi Data Pengiriman

Berikut dapat dilihat pengujian dari *form* isi data pengiriman pada tabel 4.19:

Tabel 4.19 Pengujian *Form* Isi Data Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data yang benar lalu mengklik tombol finish	Data pemesanan baru akan disimpan di dalam *database* dan menampilkan halaman konfirmasi pesanan	Dapat menampilkan halaman konfirmasi pesanan dan menyimpan data pesanan baru di dalam *database*	[√] Diterima [ ] Ditolak
Mengosongkan salah satu *field*	Menampilkan pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi”	[√] Diterima [ ] Ditolak
Klik tombol *edit cart*	Kembali ke menu sebelumnya	Dapat kembali ke menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.14 Pengujian *Form* Konfirmasi Pembayaran

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

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 : 24 No. transaksi : 24 Nama Bank : BCA Atas Nama : jellybox 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

4.5.2.15 Pengujian Menu *History*

Berikut dapat dilihat pengujian dari menu *history* pada tabel 4.21 :

Tabel 4.21 Pengujian Menu *History*

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

Klik menu *history*	Menampilkan *history* transaksi	Dapat menampilkan form *history* transaksi	[√] Diterima [ ] Ditolak
Klik *view*	Menampilkan detail transaksi yang telah dilakukan	Dapat menampilkan detail transaksi yang telah dilakukan	[√] Diterima [ ] Ditolak
Klik *view* pada saat status diterima	Menampilkan field testimoni untuk di isi oleh member	Dapat menampilkan field testimoni untuk di isi oleh member	[√] Diterima [ ] Ditolak

4.5.2.16 Pengujian Menu Retur

Berikut dapat dilihat pengujian dari menu retur pada tabel 4.22 :

Tabel 4.22 Pengujian Menu Retur

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu retur	Menampilkan transaksi dengan status diterima	Dapat menampilkan transaksi dengan status diterima	[√] Diterima [ ] Ditolak
Klik tombol *view*	Menampilkan data transaksi yang telah diterima dan menampilkan form barang yang akan di retur	Dapat menampilkan data transaksi yang telah diterima dan menampilkan form barang yang akan di retur	[√] Diterima [ ] Ditolak
Mengosongkan salah			

satu field			

4.5.2.17 Pengujian Ganti *Password Administrator*

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

Tabel 4.23 Pengujian Ganti *Password Administrator*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik edit *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 Maaf pasword tidak cocok dengan password lama	Dapat menampilkan pesan kesalahan ”*Maaf, Password Tidak Cocok dengan Password Lama !!*” bila ada field yang belum diisi	[√] Diterima [ ] Ditolak
Memasukan *password* yang salah	muncul pesan " *Maaf, Password Tidak Cocok dengan Password Lama !!*"	Dapat menampilkan pesan kesalahan ”*Maaf, Password Tidak Cocok dengan Password Lama !!*”	[√] Diterima [ ] Ditolak

Memasukan data *password* baru dan konfirmasi *password* baru yang tidak sama	Muncul pesan "*Maaf, Password Tidak Cocok dengan Password Lama !!*"	Dapat menampilkan pesan "*Maaf, Password Tidak Cocok dengan Password Lama !!*"	[√] Diterima [ ] Ditolak
Mengosongkan semua *field*	Muncul pesan kesalahan Maaf, Password Tidak Cocok dengan Password Lama.	Dapat menampilkan pesan"*Maaf, Password Tidak Cocok dengan Password Lama !!"*	√] Diterima [ ] Ditolak
Klik tombol update	Data profil baru tersimpan dalam database	Dapat menyimpan data perubahan profil ke dalam database	

4.5.2.18 Pengujian Pengolahan Akun

Berikut dapat dilihat pengujian dari pengolahan akun pada tabel 4.24 :

Tabel 4.24 Pengujian Pengolahan Akun

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik edit profil	Menampilkan form ubah profil	Dapat menampilkan formubah profil	[√] Diterima [ ] Ditolak
Mengosongkan semua field	Menampilkan Pesan kesalahan "Maaf, Form yang anda isi kurang lengkap".	Dapat menampilkan pesan kesalahan "Maaf, Form yang anda isi kurang lengkap".	[√] Diterima [ ] Ditolak

4.5.2.19 Pengujian Pengolahan Data *Admin*

Berikut dapat dilihat pengujian dari pengolahan data admin pada tabel 4.25:

Tabel 4.25 Pengujian pengolahan data admin

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu pengolahan data admin	Menampilkan *form* pengolahan data admin	Dapat menampilkan *form* pengolahan data admin	[√] Diterima [ ] Ditolak
Klik tombol tambah	Menampilkan *form* tambah admin	Dapat menampilkan *form* tambah admin	
Mengisi tambah admin Contoh: Nama admin : jellybox email : m.kharisma.semesta@gmail.com Password : jellybox Telephone :0229278123 Alamat : Parahyangan Plaza	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
Klik tombol simpan	Data administrator akan tersimpan didalam *database*	Data operator akan tersimpan didalam *database* dan muncul pesan ”data berhasil ditambah”	[√] Diterima [ ] Ditolak
Mengosongkan semua	Tampilkan pesan	Dapat	[√] Diterima

field	kesalahan	menampilkan pesan kesalahan ”wajib diisi”	[ ] Ditolak
Klik batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.20 Pengujian Data Ongkos Kirim

Berikut dapat dilihat pengujian data ongkos kirim pada tabel 4.26 :

Tabel 4.26 Pengujian data ongkos kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu ongkos kirim	Menampilkan data ongkos kirimnya	Dapat menampilkan ongkos kirimnya pengiriman	[√] Diterima [ ] Ditolak
Klik *edit*	Menampilkan *form* ubah ongkos kirim	Dapat menampilkan *form* ubah ongkos kirim	[√] 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 update	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak
Klik tombol *delete*	Data akan dihapus dari *database*	Dapat menghapus data dari *database*	[√] Diterima [ ] Ditolak

4.5.2.21 Pengujian Tambah Ongkos Kirim

Berikut dapat dilihat pengujian dari tambah ongkos kirim pada tabel 4.27 :

Tabel 4.27 Pengujian Tambah Ongkos Kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tombol tambah ongkos kirim	Menampilkan *form* tambah ongkos kirim	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data ongkos kirim Contoh: Provinsi : Jawa Barat Kota : Cimahi Jenis Pengiriman : Regular harga: 10000	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 terisijika salah satufield tidak terisiataualah dalampengisian.	[√] 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

4.5.2.22 Pengujian Data Provinsi

Berikut dapat dilihat pengujian dari data provinsi pada tabel 4.28:

Tabel 4.28 Pengujian Data Provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu provinsi	Menampilkan data provinsi	Dapat menampilkan data provinsi	[√] Diterima [ ] Ditolak
Klik *edit*	Menampilkan *form* ubah provinsi	Dapat menampilkan *form* ubah provinsi	[√] Diterima [ ] Ditolak
Mengisi data sama seperti data yang telah ada	Muncul pesan kesalahan Provinsi Sudah Ada Sebelumnya	Dapat muncul pesan kesalahan “Provinsi Sudah Ada Sebelumnya”	[√] Diterima [ ] Ditolak
Klik tombol save	Data berhasil diubah di*database*	Data yang diubah berhasil disimpan didalam database	[√] Diterima [ ] Ditolak
Klik tombol *delete*	Data akan dihapus	Dapat	[√] Diterima

	dari *database*	menghapus data dari *database*	[ ] Ditolak

4.5.2.23 Pengujian Tambah Provinsi

Berikut dapat dilihat pengujian dari tambah provinsi pada tabel 4.29 :

Tabel 4.29 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 pengisian data sama dengan data sebelumnya atau sudah ada datanya akan muncul pesan Provinsi Sudah Ada Sebelumnya	Dapat memunculkan pesan kesalahan "Provinsi Sudah Ada Sebelumnya"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data provinsi akan tersimpan didalam *database*	Data provinsi akan tersimpan didalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.24 Pengujian Data Kota

Berikut dapat dilihat pengujian dari data kota pada tabel 4.30:

Tabel 4.30 Pengujian Data Kota

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu kota	Menampilkan data kota	Dapat menampilkan data kota	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah kota	Dapat menampilkan *form* ubah kota	[√] Diterima [ ] Ditolak
Mengisi data sama persis dengan data yang ada sebelumnya	Jika pengisian data sama dengan data sebelumnya atau sudah ada datanya akan muncul pesan Kota Sudah Ada Sebelumnya	Dapat memunculkan pesan kesalahan "Kota Sudah Ada Sebelumnya"	[√] Diterima [ ] Ditolak
Klik tombol update	Data berhasil diubah di*database*	Data kota akan tersimpan didalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak
Klik tombol *delete*	Data akan dihapus dari *database*	Dapat enghapus data dari *database*	[√] Diterima [ ] Ditolak

4.5.2.25 Pengujian Tambah Kota

Berikut dapat dilihat pengujian dari tambah kota pada tabel 4.31:

Tabel 4.31 Pengujian Tambah Kota

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tombol tambah kota	Menampilkan *form* tambah kota	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data kota Contoh : Nama Provinsi : Jawa Barat Nama Kota : Bandung Jenis Kota : Kota	Jika pengisian data sama dengan data sebelumnya atau sudah ada datanya akan muncul pesan Kota Sudah Ada Sebelumnya	Dapat memunculkan pesan kesalahan "Kota Sudah Ada Sebelumnya"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data kota akan tersimpan didalam *database*	Data kotaakan tersimpan didalam *database*.	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.26 Pengujian Data Ukuran

Berikut dapat dilihat pengujian dari data ukuran pada tabel 4.32 :

Tabel 4.32 Pengujian Data Ukuran

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

Klik tombol *edit*	Menampilkan *form* ubah ukuran	Dapat menampilkan *form* ubah ukuran	[√] Diterima [ ] Ditolak
Tidak mengisi field yang wajib diisi	Jika field nama ukuran tidak diisi maka akan tampil pesan kesalah field ini wajib di isi	Dapat menampilkan pesan kesalahan apabila field nama ukuran tidak di isi "Field ini wajib di isi"	
Klik tombol update	Data berhasil diubah di *database*	Dapat menyimpan data yang dirubah di *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak
Klik tombol *delete*	Data akan dihapus dari *database*	Dapat menghapus data dari *database*	[√] Diterima [ ] Ditolak

4.5.2.27 Pengujian Tambah Ukuran

Berikut dapat dilihat pengujian dari tambah ukuran pada tabel 4.33:

Tabel 4.33 Pengujian Tambah Ukuran

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

Mengisi data ukuran Contoh : Nama Ukuran : 38 Deskripsi Ukuran : ukuran 38 setara dengan xx cm	Jika field nama ukuran tidak diisi maka akan tampil pesan kesalah field ini wajib di isi	Dapat menampilkan pesan kesalahanapabila field nama ukuran tidak di isi “Fieldini wajib di isi”	[√] Diterima [ ] Ditolak
Klik tombol save	Data ukuran berhasil diubah didalam *database*	Dapat menyimpan data yang dirubah di *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.28 Pengujian Data Kategori

Berikut dapat dilihat pengujian dari lihat kategori pada tabel 4.34:

Tabel 4.34 Pengujian Data Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu 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 yang wajib di isi	Jika field parent dan nama kategori tidak diisi maka akan tampil pesan kesalah field ini wajib di isi	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
Klik tombol *delete*	Data akan dihapus dari *database*	Dapat menghapus data dari *database*	[√] Diterima [ ] Ditolak

4.5.2.29 Pengujian Tambah Kategori

Berikut dapat dilihat pengujian dari tambah kategori pada tabel 4.35:

Tabel 4.35 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 ukuran Contoh : Parent : Sepatu Nama Kategori : Heels Deskripsi Kategori :	Jika field parent dan nama kategori tidak diisi maka akan tampil pesan kesalah field ini wajib di isi dan jika pengisian sama datanya seperti data yang ada saebelumya akan tampil pesan "Maaf,Kategori tidak boleh sama"	Dapat menampilkan pesan kesalahan apabila field nama ukuran tidak di isi "Field ini wajib di isi" dan "Maaf,Kategori tidak boleh sama "	[√] 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

4.5.2.30 Pengujian Menu Barang

Berikut dapat dilihat pengujian dari data barang pada tabel 4.36 :

Tabel 4.36 Pengujian Data Barang

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu barang	Menampilkan data barang	Dapat menampilkan data barang	[√] Diterima [ ] Ditolak
Klik *edit*	Menampilkan *form* ubah produk	Dapat menampilkan *form* ubah	[√] Diterima [ ] Ditolak

		produk	
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan ” *field* harus diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Dapat menyimpan data yang diubah ke dalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak
Klik tombol *delete*	Data akan dihapus dari *database*	Dapat menghapus data dari *database*	[√] Diterima [ ] Ditolak

4.5.2.31 Pengujian Tambah Barang

Berikut dapat dilihat pengujian dari tambah barang pada tabel 4.37:

Tabel 4.37 Pengujian Tambah Barang

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: Heels nama barang : Heels01 Harga : 13000 Diskon : 0 %	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus	Dapat menampilkan pesan kesalahan apabila field nama ukuran tidak di isi	[√] Diterima [ ] Ditolak

Promo : Ya Deskripsi : Sepatu dengan hak tinggi Gambar :	terisi. jika pengisian sama datanya seperti data yang ada saebelumya akan tampil pesan "Maaf,nama barang tidak boleh sama"	"Field ini wajib di isi" dan "Maaf,nama barang tidak boleh sama "	
Tidak mengisi semua *field*	Tampilkan pesan kesalahan "ada kesalahan didalam pengisian data, silahkan cek kembali"	Dapat menampilkan pesan kesalahan "ada kesalahan didalam pengisian data, silahkan cek kembali"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data barang akan tersimpan didalam *database*	Data barang akan tersimpan didalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.32 Pengujian Menu Member

Berikut dapat dilihat pengujian dari data member pada tabel 4.38:

Tabel 4.38 Pengujian Menu 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 tombol status	Mengaktifkan atau me *non*-aktif kan *account* member	Dapat mengaktifkan atau me *non-aktif* kan	[√] Diterima [ ] Ditolak
Klik tombol *delete*	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

4.5.2.33 Pengujian Menu Jenis Pengiriman

Berikut dapat dilihat pengujian dari menu jasa pengiriman pada tabel 4.39:

Tabel 4.39 Pengujian Menu Jasa Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu	Menampilkan data jasa pengiriman	Dapat menampilkan data jasa pengiriman	[√] Diterima [ ] Ditolak
Klik tombol *edit*	Menampilkan *form* ubah jasa pengiriman	Dapat menampilkan *form* ubah jasa pengiriman	[√] Diterima [ ] Ditolak

Klik hapus	Data akan dihapus dari *database*.	Data telah dihapus dari database.	[√] 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 “ada kesalahan didalam pengisian data, silahkan cek kembali”	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Data berhasil diubah ke dalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak
Klik tombol *delete*	Data akan dihapus dari *database*	Dapat menghapus data dari *database*	[√] Diterima [ ] Ditolak

4.5.2.34 Pengujian Tambah Jasa Pengiriman

Berikut dapat dilihat pengujian dari tambah jasa pengiriman pada tabel 4.40:

Tabel 4.40 Pengujian Tambah Jasa Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah jenis pengiriman	Menampilkan *form* tambah jenis pengiriman	Dapat menampilkan *form* tambah jenis pengiriman	[√] Diterima [ ] Ditolak
Mengisi data jenis pengiriman Contoh: Nama jenis pengiriman : YES deskripsi :	Jika *field* nama jenis pengiriman tidak di isi akan tampil pesan "field ini wajib di isi" dan jika isi nama jenis pengiriman sama dengan data yang sudah ada sebelumnya akan tampil pesan "Maaf,nama jenis pengiriman tidak boleh sama"	Dapat memunculkan pesan kesalahan "*field* ini wajib di isi" dan "Maaf,nama jenis pengiriman tidak boleh sama".	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data jenis pengiriman akan tersimpan didalam *database*	Data jenis pengiriman akan tersimpan didalam *database*.	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.35 Pengujian Menu Detail Barang

Berikut dapat dilihat pengujian dari menu detail barang pada tabel 4.41:

Tabel 4.41 Pengujian Menu Detail Barang

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu detail barang	Menampilkan data detail barang	Dapat menampilkan data detail barang	[√] Diterima [ ] Ditolak
Klik tombol *edit*	Menampilkan *form* ubah detail barang	Dapat menampilkan *form* ubah detail barang	[√] Diterima [ ] Ditolak
Jika salah satu *field* tidak di isi	Akan tampil pesan kesalahan "*field* wajib diisi"	Dapat menampilkan pesan kesalahan "*field* wajib diisi"	[√] Diterima [ ] Ditolak
Klik tombol *update*	Data yang diubah disimpan ke dalam *database*	Dapat menyimpan data yang telah diubah ke dalam *database*	[√] Diterima [ ] Ditolak
Klik tombol *delete*	Data akan dihapus dari *database*	Dapat menghapus data dari *database*	[√] Diterima [ ] Ditolak

4.5.2.36 Pengujian Tambah Detail Barang

Berikut dapat dilihat pengujian dari tambah detail barang pada tabel 4.42:

Tabel 4.42 Pengujian Tambah Detail Barang

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah detail barang	Menampilkan *form* tambah detail barang	Dapat menampilkan *form* tambah detail barang	[√] Diterima [ ] Ditolak
Mengisi data detail barang Contoh: Nama barang : Flat01 Warna : Merah Ukuran : 38 Berat : 0,6 Kg	Jika pengisian data sama dengan data yang ada sebelumnya akan tampil pesan kesalahan "Data ini sudah ada"	Dapat menampilkan pesan kesalahan " Data ini sudah ada"	[√] Diterima [ ] Ditolak
Jika salah satu *field* tidak di isi	Akan tampil pesan kesalahan "*field* wajib diisi"	Dapat menampilkan pesan kesalahan "*field* wajib diisi"	[√] Diterima [ ] Ditolak
Klik tombol *save*	Data disimpan ke dalam *database*	Dapat menyimpan data ke dalam *database*	[√] Diterima [ ] Ditolak

4.5.2.37 Pengujian Menu Rekening

Berikut dapat dilihat pengujian dari menu rekening pada tabel 4.43:

Tabel 4.43 Pengujian Menu Rekening

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu rekening	Menampilkan data rekening	Dapat menampilkan data rekening	[√] Diterima [ ] Ditolak
Klik tombol *edit*	Menampilkan *form* ubah rekening	Dapat menampilkan *form* ubah rekening	[√] Diterima [ ] Ditolak
Jika salah satu *field* tidak di isi	Akan tampil pesan kesalahan "*field* wajib diisi"	Dapat menampilkan pesan kesalahan "*field* wajib diisi"	[√] Diterima [ ] Ditolak
Klik tombol *update*	Data yang diubah disimpan ke dalam *database*	Dapat menyimpan data yang telah diubah ke dalam *database*	[√] Diterima [ ] Ditolak
Klik tombol *delete*	Data akan dihapus dari *database*	Dapat menghapus data dari *database*	[√] Diterima [ ] Ditolak

4.5.2.38 Pengujian Tambah Rekening

Berikut dapat dilihat pengujian dari tambah rekening pada tabel 4.44:

Tabel 4.44 Pengujian Tambah Rekening

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tombol tambah rekening	Menampilkan *form* tambah rekening	Dapat menampilkan *form* tambah rekening	[√] Diterima [ ] Ditolak
Mengisi data rekening Contoh: Nama Bank : BCA Nama Pemilik : Ridwan Arifin Nomor Rekening : 010256521 Cabang : Cikampek Gambar :	Jika pengisian data sama dengan data yang ada sebelumnya akan tampil pesan kesalahan "Data ini sudah ada"	Dapat menampilkan pesan kesalahan " Data ini sudah ada"	[√] Diterima [ ] Ditolak
Jika salah satu *field* yang wajib di isi tidak di isi	Akan tampil pesan kesalahan "*field* wajib diisi"	Dapat menampilkan pesan kesalahan "*field* wajib diisi"	[√] Diterima [ ] Ditolak
Klik tombol *save*	Data disimpan ke dalam *database*	Dapat menyimpan data ke dalam *database*	[√] Diterima [ ] Ditolak

4.5.2.39 Pengujian Menu Warna

Berikut dapat dilihat pengujian dari menu warna pada tabel 4.45:

Tabel 4.45 Pengujian Menu Warna

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu warna	Menampilkan data warna	Dapat menampilkan data warna	[√] Diterima [ ] Ditolak
Klik tombol *edit*	Menampilkan *form* ubah warnaa	Dapat menampilkan *form* ubah warna	[√] Diterima [ ] Ditolak
Jika salah satu *field* tidak di isi	Akan tampil pesan kesalahan "*field* wajib diisi"	Dapat menampilkan pesan kesalahan "*field* wajib diisi"	[√] Diterima [ ] Ditolak
Klik tombol *update*	Data yang diubah disimpan ke dalam *database*	Dapat menyimpan data yang telah diubah ke dalam *database*	[√] Diterima [ ] Ditolak
Klik tombol *delete*	Data akan dihapus dari *database*	Dapat menghapus data dari *database*	[√] Diterima [ ] Ditolak

4.5.2.40 Pengujian Tambah Warna

Berikut dapat dilihat pengujian dari tambah warna pada tabel 4.46:

Tabel 4.46 Pengujian Tambah Warna

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tombol tambah warna	Menampilkan *form* tambah warna	Dapatmenampilkan*form*tambah warna	[√] Diterima [ ] Ditolak
Mengisi data rekening Contoh: Nama Bank : ISBC	Jika pengisian data sama dengan data yang ada sebelumnya akan tampil pesan kesalahan “Data ini sudah ada”	Dapat menampilkan pesan kesalahan “ Data ini sudah ada”	[√] Diterima [ ] Ditolak
Jika salah satu *field* yang wajib di isi tidak di isi	Akan tampil pesan kesalahan “*field* wajib diisi”	Dapat menampilkan pesan kesalahan ”*field* wajib diisi”	[√] Diterima [ ] Ditolak
Klik tombol *save*	Data disimpan ke dalam *database*	Dapat menyimpan data ke dalam *database*	[√] Diterima [ ] Ditolak

4.5.2.41 Pengujian Menu Pengolahan Data Transaksi

Berikut dapat dilihat pengujian dari menu pengolahan data Transaksi pada tabel 4.47:

Tabel 4.47 Pengujian Menu Pengolahan Data Transaksi

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu data pesanan	Menampilkan data pesanan	Dapat menampilkan data pesanan	[√] Diterima [ ] Ditolak
Klik tombol *print*	Menampilkan *form* data pesanan yang akan sebagai tanda kirim	Dapat menampilkan *form* data pesanan yang akan sebagai tanda kirim	[√] Diterima [ ] Ditolak
Klik tombol ubah	Menampilkan data pesanan dan status pembayaran	Dapat menampilkan data pesanan dan status pembayaran	[√] Diterima [ ] Ditolak
Mengisi nomor resi(*field* ini ada jika member telah melakukan pembayaran dan konfirmasi pembayaran) Contoh: No Resi : 12345678	Jika *field* tidak di isi akan tampil pesan kesalahan "*field* ini wajib diisi"	Dapat menampilkan pesan kesalahan "*field* ini wajib diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan(tombol ini ada jika member telah melakukan pembayaran dan konfirmasi pembayaran)	Nomor resi yang telah di inputkan akan terkirim ke email member dan menu history pemesanan member dan mengubah	Dapat menyimpan dan mengirimkan ke email member dan history	[√] Diterima [ ] Ditolak

	status pesanan yang tadinya “dibayar” menjadi “dikirim”	pemesanan member dan mengubah status pesanan yang tadinya “dibayar” menjadi “dikirim”	
Klik tombol batal	Kembali ke menu sebelumnya	Dapat kembali ke menu sebelumnya	[√] Diterima [ ] Ditolak
Klik tombol konfirmasi barang diterima(tombol ini ada setelah admin mengirimkan no resi ke member)	Mengubah status pesanan yang tadinya “dikirim” menjadi “diterima”	Dapat mengubah status pesanan yang tadinya “dikirim” menjadi “diterima”	[√] Diterima [ ] Ditolak
Klik tombol hapus	Data akan dihapus dari *database*	Dapat menghapus data dari *database*	[√] Diterima [ ] Ditolak

4.5.3 Kesimpulan Hasil Pengujian Alpha

Berdasarkan hasil pengujian sistem yang telah dilakukan secara keseluaruhan memberikan kesimpulan bahwa pada proses pada situs *e-commerce* telah melalui tahap perbaikan dan sudah dimaksimalkan terhaap proses-proses tersebut dan secara fungsional sistem sudah dapat digunakan dan menghasilkan *output* yang diharapkan.

4.6 Pengujian Beta

Pengujian beta merupakan yang dilakukan secara objektif dimana diuji secara langsung kelapangan yaitu instansi yang bersangkutan dengan membuat

kuesioner mengenai kepuasan *user* dengan kandungan point syarat *user friendly* untuk selanjutnya dibagikan kepada sebagian *user* dengan mengambil sample sebanyak 5 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

Dan untuk dilingkungan jellybox hanya menggunakan metode wawancara terstruktur karena aplikasi ini hanya digunakan oleh satu *user* saja yaitu pemilik toko (*owner*).

Kemudian penelitian di lingkungan pada masyarakat umum khususnya kisaran umur 17 hingga yang sudah terbiasa dengan *e-commerce*, 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 *Simple Random Sampling* yang disebarkan kepada 5 pengguna. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner ini terdiri dari 5 pertanyaan (contoh kuesioner dapat diliihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 5, dengan ketentuan skala untuk tiap pertanyaan sebagai berikut :

4.6.1.1 Pengujian Beta untuk Admin

Pada pengujian beta ini saya melakukan tahapan wawancara berstruktur yang diajukan pada owner toko jellybox, berikut adalah butir-butir wawancara yang saya ajukan pada owner jellybox :

Penulis : Apakah anda setuju tampilan perangkat lunak untuk pengolahan data jellybox sudah bagus?

Owner : Setuju, karena perpaduan warna tidak membuat mata sakit.

Penulis : Apakah anda setuju instruksi yang terdapat dalam *website* tersebut mudah dimengerti?

Owner : Setuju, karena indtruksi-instruksi yang ada mudah dipahami

Penulis : Apakah anda setuju *website* ini membantu dalam pengolahan data barang?

Owner : Setuju, karena saya lebih mudah memberikan penamaan atau list barang.

Penulis : Apakah anda setuju website ini membantu dalam pengolahan data penjualan secara *online?*

Owner : Tidak, karena web ini masih dalam tahap proses pembangunan mungkin beberapa bulan kedepan baru bisa dilihat hasil dari web ini.

Penulis : Apakah anda setuju *website* ini dapat membantu dalam membuat laporan penjualan ?

Owner : Setuju, karena web ini memfasilitasi adanya pembuatan laporan baik harian maupun barang.

Menurut hasil wawancara secara keseluruhan untuk *website* yang dibangun *owner* merasa puas dengan adanya *website* jellybox, dilihat dari segi tampilan *website* jellybox menarik dan tidak membosankan. Dari segi instruksi *owrner* juga mengakui bahwa instruksi yang digunakan pada *website* memang mudah dipahami, bila bicara dari segi pemasaran *owner* pun menjawab dengan serius

bahwa mungkin beberapa bulan kedepan baru dapat dilihat apakah benar *website* ini membantu dalam proses pengolahan data barang dalam hal ini penjulan barang.

4.6.1.2 Kuesioner Pengujian Beta Untuk Pengguna

Berdasarkan data hasil kusioner tersebut (terlampir), dapat dicari rata-rata hasil setiap jawaban dengan menggunakan rumus : Y = P/R

Keterangan :

P = Jumlah point jawaban responden tiap soal.

R = Jumlah responden

Y = Nilai rata-rata

Dengan beberapa aturan pembulatan data statistika sebagai berikut :

- Angka lebih dari 5 dibulatkan jadi 10 pada tempatnya. Artinya angka yang mendahuluinya ditambah 1 pada tempatnya.
- Angka kurang dari 5 dibulatkan jadi 0 pada tmpatnya. Artinya angka yang mendahuluinya tetap.
- Angka sama dengan 5 dibulatkan jadi 0 jika angka yang mendahuluinya genap, dan dibulatkan jadi 10 jika angka yang mendahuluinya ganjil. Dengan demikian hasil pembulatan selalu genap, aturan terakhir ini disebut aturan genap terdekat.

Berikut ini adalah hasil analisis pengujian pengguna terhadap 10 responden hasil rata-rata setiap jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 10 orang.

No	Responden	Tanya 1					Tanya 2					Tanya 3					Tanya 4					Tanya 5					Tanya 6					Tanya 7				
		1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
1	Kirana C				√					√		√							√			√								√		√				
2	Rudi				√					√						√					√				√					√						√
3	Reiwanti				√					√				√						√					√					√						√
4	Septya Y				√						√					√			√							√					√						√
5	Siti				√					√					√				√						√						√				√		
6	Ditha					√				√				√						√					√					√					√		
7	Uswatun				√						√			√						√					√						√						√
8	Atha					√				√					√					√					√					√					√		
9	Putri					√				√					√				√						√					√							√
10	Reni					√				√				√					√					√						√					√		
	Jumlah Point	44					42					36					36					37					42					43				
	Y=P/R	4,4					4,2					3,6					3,6					3,7					4,2					4,3				

Gambar 4.1 Hasil Kuesioner

Untuk pertanyaan ke 1 responden rata-rata memilih bahwa tampilan *website* jellybox menarik dan mudah untuk dipahami/*user friendly*, pertanyaan ke 2 responden pun sejutu dengan adanya *website e-commerce* konsumen mudah mendapatkan informasi tebaru tentang produk yang ada di jellybox. Untuk pertanyaan ke 3 responden dengan adanya *website* ini akan lebih mudah dalam melakukan transaksi pembelian di bandingkan transaksi secara manual dengan cara pergi ke toko langsung. Pertanyaan ke 4 responden setuju dengan adanya *website* ini pemilik toko dapat memperluas pansa pasar dengan adanya integrasi dengan situs-situs *social network*. Pertanyaan ke 5 responden pun setuju bahwa dengan adanya sistem pembayaran secara *online* akan lebih memudahkan dalam

bertransaksi, responden juga menyetujui bahwa dengan adanya *website* ini pengguna dapat dengan mudah mengetahui produk-produk jellybox. Dan yang terakhir untuk pertanyaan ke 7 responden setuju bahwa dengan adanya *website* ini pengguna dapat dengan mudah memperoleh barang yang diinginkan tanpa harus pergi ke toko.