BAB IV
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.

4.2 Kebutuhan Sistem

Kebutuhan sistem terdiri atas spesifikasi perangkat keras dan spesifikasi perangkat lunak.

4.2.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras (*hardware*) yang digunakan dalam pembangunan aplikasi *E-Commerce* di Butik De'Smoothy ini adalah :

1. *Processor AMD Athlon (tm)* 64 x2 *Dual Core Processor* 4200++
2. *RAM* 2046 MB
3. *Harddisk* 250 GB
4. *Keyboard*
5. *Monitor*

4.2.2 Spesifikasi Perangkat Lunak

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

1. Sistem Operasi menggunakan Windows 7
2. Bahasa Pemograman menggunakan PHP
3. *MySQL* sebagai penyimpanan *database*
4. *PhpMyAdmin* sebagai *interface database*
5. *WampServer* 2.0 sebagai *web server*
6. Macromedia Dreamweaver sebagai kode dan perancangan tampilan
7. *Mozilla Firefox* sebagai *web browser*

4.3 Implementasi Basis Data

a. Tabel Admin

```
CREATE TABEL 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=7
```

b. Tabel Barang

```
CREATE TABEL IF NOT EXISTS `barang` (
  `id_barang` int(11) NOT NULL AUTO_INCREMENT,
  `id_kategori` int(11) NOT NULL,
  `nama_barang` varchar(50) NOT NULL,
  `harga_barang` int(11) NOT NULL,
  `warna_barang` varchar(100) NOT NULL,
  `deskripsi_barang` text NOT NULL,
```

```
  `diskon_barang` int(11) NOT NULL,
  `rating_barang` float NOT NULL,
  `voterrating_barang` int(11) NOT NULL,
  `viewcounter_barang` int(11) NOT NULL,
  PRIMARY KEY (`id_barang`),
  KEY `id_kategori` (`id_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Barang' AUTO_INCREMENT=26
ALTER TABLE `barang`
  ADD CONSTRAINT `barang_ibfk_1` FOREIGN KEY (`id_kategori`)
REFERENCES `kategori` (`id_kategori`);
```

c. Tabel Config

```
CREATE TABEL IF NOT EXISTS `config` (
  `id_config` int(11) NOT NULL AUTO_INCREMENT,
  `nama_config` varchar(100) NOT NULL,
  `content_config` text NOT NULL,
  `publish_config` float NOT NULL,
  PRIMARY KEY (`id_config`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Config' AUTO_INCREMENT=2
```

d. Tabel Detail Barang

```
CREATE TABEL IF NOT EXISTS `detailbarang` (
  `id_detailbarang` int(11) NOT NULL auto_increment,
  `id_barang` int(11) NOT NULL,
  `id_ukuran` int(11) default NULL,
  `tanggal_detailbarang` datetime NOT NULL,
  `stok_detailbarang` int(11) 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_ukuran` (`id_ukuran`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Detail Barang' AUTO_INCREMENT=76 ;
ALTER TABLE `detailbarang`
  ADD CONSTRAINT `detailbarang_ibfk_1` FOREIGN KEY
(`id_barang`) REFERENCES `barang` (`id_barang`);
ADD CONSTRAINT 'detailbarang_ibfk_2` FOREIGN KEY
(`id_ukuran`) REFERENCES `ukuran` (`id_ukuran`);
```

e. Tabel Gambar

```
CREATE TABEL IF NOT EXISTS `gambar` (
  `id_gambar` int(11) NOT NULL auto_increment,
  `id_barang` int(11) 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=60 ;
ALTER TABLE `gambar`
  ADD CONSTRAINT `detailbarang_ibfk_1` FOREIGN KEY
(`id_barang`) REFERENCES `barang` (`id_barang`);
```

f. Tabel Hubungi

```
CREATE TABEL 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,
  `status` varchar(20) NOT NULL default 'BELUM DIBACA',
  PRIMARY KEY  (`id_hubungi`)
  KEY `id_admin` (`id_admin`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Hubungi Kami' AUTO_INCREMENT=12 ;
ALTER TABLE `hubungi`
  ADD CONSTRAINT `hubungi_ibfk_1` FOREIGN KEY (`id_dmin`)
REFERENCES `admin` (`id_admin`);
```

g. Tabel Invoice

```
CREATE TABEL IF NOT EXISTS `invoice` (
  `id_detailpembelian` int(11) NOT NULL auto_increment,
  `tanggal_ detailpembelian ` datetime NOT NULL,
  `jenis_pembayaran` enum('paypal','rekening','cod') NOT
NULL,
  `status_pengiriman`
enum('dipesan','dikonfirmasi','dibayar','dikirim','diterima'
) 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,
  `nama_bank_pemesan` varchar(50) NOT NULL,
  `no_rekening_pemesan` varchar(20) NOT NULL,
  `no_resi_pemesan` varchar(32) NOT NULL,
  `id_rekening` int(11) NOT NULL,
  `id_ongkoskirim` int(11) NOT NULL,
  `status_pemesanan` enum('ok','cancel') NOT NULL,
  `session_id` varchar(32) NOT NULL,
  PRIMARY KEY  (`id_invoice`)
  KEY `id_rekening` (`id_rekening`)
  KEY `id_ongkoskirim` (`id_ongkoskirim`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Invoice' AUTO_INCREMENT=136 ;
```

```
   ALTER TABLE `invoice`
     ADD CONSTRAINT `invoice_ibfk_1` FOREIGN KEY
   (`id_rekening`) REFERENCES `rekening` (`id_rekening`);
   ADD CONSTRAINT `invoice_ibfk_2` FOREIGN KEY
   (`id_ongkoskirin`) REFERENCES `ongkoskirim`
   (`id_ongkoskirim`);
```

h. Tabel Jasa Pengiriman

```
CREATE TABEL 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`),
  UNIQUE KEY `nama_jasapengiriman` (`nama_jasapengiriman`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

i. Tabel Jenis Pengiriman

```
CREATE TABEL 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`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 COMMENT='Tabel Jenis
Pengiriman' AUTO_INCREMENT=3
ALTER TABLE `jenispengiriman`
  ADD CONSTRAINT `jenispengiriman_ibfk_1` FOREIGN KEY
(`id_jasapengiriman`) REFERENCES `jasapengiriman`
(`id_barang`);
```

j. Tabel Kategori

```
CREATE TABEL IF NOT EXISTS `kategori` (
  `id_kategori` int(11) NOT NULL AUTO_INCREMENT,
 `kode_kategori` varchar(3) NOT NULL,
  `nama_kategori` varchar(50) NOT NULL,
  PRIMARY KEY (`id_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Kategori' AUTO_INCREMENT=22
```

k. Tabel Kota

```
CREATE TABEL IF NOT EXISTS `kota` (
  `id_kota` int(11) NOT NULL AUTO_INCREMENT,
  `id_provinsi` int(11) NOT NULL,
  `nama_kota` varchar(50) NOT NULL,
  `kabkota` varchar(15) NOT NULL,
  `cod` enum('0','1') NOT NULL,
  PRIMARY KEY (`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`);
```

l. Tabel Member

```
CREATE TABEL 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,
  `tanggal_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`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Member' AUTO_INCREMENT=22 ;
ALTER TABLE `member`
  ADD CONSTRAINT `member_ibfk_1` FOREIGN KEY (`id_kota`)
REFERENCES `kota` (`id_kota`);
```

m. Tabel Ongkir

```
CREATE TABEL 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_ongkoskirim`),
  KEY `id_kota` (`id_kota`),
  KEY `id_jenispengiriman` (`id_jenispengiriman`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Ongkos Kirim' AUTO_INCREMENT=13 ;
ALTER TABLE `ongkir`
  ADD CONSTRAINT `ongkir_ibfk_1` FOREIGN KEY (`id_ongkir`)
REFERENCES `ongkir` (`id_ongkir`);
  ADD CONSTRAINT `kota_ibfk_2` FOREIGN KEY (`id_kota`)
REFERENCES `kota` (`id_kota`);
```

n. Tabel Pembelian

```
CREATE TABEL 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) NOT NULL,
  `harga_temp` int(11) NOT NULL,
  `stok_temp` int(11) NOT NULL,
  `berat_temp` float NOT NULL,
  `status_retur` enum('0','1') NOT NULL,
  `session_id` varchar(32) NOT NULL,
  PRIMARY KEY (`id_pembelian`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Pembelian' AUTO_INCREMENT=69 ;
ALTER TABLE `pembelian`
 ADD CONSTRAINT  `kota_ibfk_1`  FOREIGN KEY (`id_invoice`)
REFERENCES `invoice` (`id_invoice`);
ADD CONSTRAINT  `kota_ibfk_2`  FOREIGN KEY (`id_member`)
REFERENCES `member` (`id_member`);
ADD CONSTRAINT  `kota_ibfk_3`  FOREIGN KEY
(`id_detailbarang`) REFERENCES `detailbarang`
(`id_detailbarang`);
```

o. Tabel provinsi

```
CREATE TABEL IF NOT EXISTS `provinsi` (
  `id_provinsi` int(11) NOT NULL AUTO_INCREMENT,
  `nama_provinsi` varchar(50) NOT NULL,
  PRIMARY KEY (`id_provinsi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=35
```

p. Tabel rekening

```
CREATE TABEL IF NOT EXISTS `rekening` (
  `id_rekening` int(11) NOT NULL AUTO_INCREMENT,
  `nama_rekening` varchar(50) NOT NULL,
  `bank_rekening` varchar(50) NOT NULL,
  `cabang_rekening` varchar(100) NOT NULL,
  `no_rekening` varchar(32) NOT NULL,
  `gambar_rekening` varchar(200) NOT NULL,
  PRIMARY KEY (`id_rekening`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Rekening' AUTO_INCREMENT=6 ;
```

q. Tabel retur

```
CREATE TABEL IF NOT EXISTS `retur` (
  `id_retur` int(11) NOT NULL AUTO_INCREMENT,
  `id_admin` int(11) DEFAULT NULL,
  `id_detailpembelian` int(11) NOT NULL,
  `id_detailbarang` int(11) NOT NULL,
  `stok_retur` int(11) NOT NULL,
  `tanggal_retur` datetime NOT NULL,
  `keterangan_retur` text NOT NULL,
  `status_retur` enum('dikonfirmasi','diterima','ditolak')
NOT NULL,
  PRIMARY KEY (`id_retur`),
```

```
  KEY `id_admin` (`id_admin`),
  KEY `id_invoice` (`id_invoice`),
  KEY `id_detaibarang` (`id_detailbarang`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel Retur
Barang' AUTO_INCREMENT=6 ;
ALTER TABLE `retur`
ADD CONSTRAINT `admin_ibfk_1`   FOREIGN KEY (`id_admin`)
REFERENCES `admin` (`id_admin`);
ADD CONSTRAINT  `detailpembelian_ibfk_2`  FOREIGN KEY
(`id_invoice`) REFERENCES `detailpembelian`
(`id_detailpembelian`);
ADD CONSTRAINT  `detailbarang_ibfk_1`  FOREIGN KEY
(`id_detaibarang`) REFERENCES `detailbarang`
(`id_detailbarang`);
```

r. Tabel ukuran

```
CREATE TABEL 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=26 ;
```

s. Tabel komplain

```
CREATE TABLE IF NOT EXISTS `komplain` (
  `id_komplain` int(11) NOT NULL AUTO_INCREMENT,
  `nama_komplain` varchar(200) NOT NULL,
  `deskripsi_komplain` text NOT NULL,
  PRIMARY KEY (`id_komplain`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='Tabel
Komplain' AUTO_INCREMENT=3 ;
```

t. 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) NOT NULL,
  PRIMARY KEY (`id_warna`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 COMMENT='Tabel
Warna' AUTO_INCREMENT=4 ;
ADD CONSTRAINT `warna_ibfk_1`   FOREIGN KEY
(`id_detailpembelian`) REFERENCES ` detailpembelian ` (`id_
detailpembelian `);
```

4.4 Implementasi Antar Muka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Implementasi sistem

dimaksudkan untuk menjelaskan secara singkat hasil transformasi dari hasil perancangan kepada program aplikasi. Berikut ini adalah implementasi antarmuka yang dibuat dan dibedakan antara antarmuka untuk pengunjung, member dan admin.

4.4.1 Implementasi Antar Muka Pengunjung & Member

Berikut ini adalah gambar implementasi antar muka pengunjung dan member pada gambar dibawah ini :

Tabel 4.1 Implementasi antarmuka Pengunjung dan Member

Menu	Deskripsi	Nama File
Registarsi member	Digunakan untuk melakukan pendaftaran *member* baru.	Register.php
Login *member*	Digunakan Digunakan sebagai halaman login *member*	Index.php
Lupa *password*	Digunakan jika *member* lupa dengan *password*	lupapassword.php
Cara Pembelian	Digunakan untuk melihat cara pembelian	Carapembelian.php
Setting Akun	Digunakan untuk mengubah profil *member.*	Edit.account.php
Ganti password	Digunakan jika *member* ingin merubah *password*	ubahpass.php
Kategori	Digunakan untuk melihat barang perkategori	kategori. php
Home	Digunakan untuk melihat keseluruhan produk	Index.php

Lihat History	digunakan untuk melihat history pembelian,pemesanan	History.php
Hubungi Kami	Digunakan untuk melakukan tanya jawab secara langsung	hubungi.php
Keranjang belanja	Digunakan untuk melihat, merubah, menghapus data pembelian yang akan dilakukan.	checkout.php
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 4.2 Implementasi Antarmuka Admin

Menu	**Deskripsi**	**Nama File**
Login administrator	Digunakan sebagai halaman login administrator	index.php
Lupa Password	Digunakan jika admin lupa terhadap passwordnya	lupapassword.php
Pengaturan Akun	Digunakan jika admin ingin merubah profil	ubahprofile.php
Pengolahan Data Admin	Digunakan untuk melihat,menambah dan menghapus admin	Admin.Php

Ubah Kata Sandi Admin	Digunakan jika administrator ingin merubah *password*	ubahpassword.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	kurs.Php

	status data kurs	
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	laporanbarang.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

4.5 Pengujian Sistem

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.

4.6 Pengujian Alpa

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

4.6.1 Skenario Pengujian Alpa

Tabel 4.3 Skenario Pengujian Aplikasi Member

Item Uji	Detail Pengujian	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 *Email*	*Black Box*
	Verifikasi *password*	*Black Box*
Ubah profil	Menampilkan *form* ubah profil	*Black Box*
	Ubah data *member*	*Black Box*
	Simpan perubahan	*Black Box*
ubah *password*	Menampilkan *form* ubah *password*	*Black Box*
	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verifikasi konfirmasi *password*	*Black Box*
Menu Barang	Menampilkan data barang	*Black Box*
	Menampilkan data barang	*Black Box*

	Cari barang	*Black Box*
Menu Hubungi	Menampilkan form	*Black Box*
	Isi form hubungi	*Black Box*
Menu keranjang belanja	Menampilkan data pembelian	*Black Box*
	Mengupdate data pembelian	*Black Box*
	Menghapus data pembelian	*Black Box*
Form konfirmasi	Menampilkan konfirmasi transaksi	*Black Box*
Form pengiriman	Menampilkan form pengiriman	*Black Box*
	Pilih jenis pengiriman	*Black Box*
Logout	Menghapus *session user*	*Black Box*

Tabel 4.4 Skenario Pengujian Aplikasi Admin

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Lupa *password*	Menampilkan *form* Lupa *password*	*Black Box*
Data Transaksi	Tampilkan data transaksi	*Black Box*
	Ubah status transaksi	*Black Box*
Kategori	Tampilkan data kategori	*Black Box*

	Tambah Data katgori	*Black Box*
	Ubah data kategori	*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*
Ukuran	Tampilkan data Ukuran	*Black Box*
	Tambah data Ukuran	*Black Box*
	Ubah data ukuran	*Black Box*
Retur	Tampilkan data retur	*Balck Box*
	Ubah data Retur	*Black Box*
Data Laporan	Tampilkan data laporan	*Black Box*
	Cari data laporan	*Black Box*
	Cetak data laporan	*Black Box*
Data Transaksi	Tampilkan data Transaksi	*Black Box*
	Ubah data transaksi	*Black Box*
Member	Tampilan Data Member	*Black Box*
Data kategori	Tampilkan data kategori	*Black Box*

	Tambah data kategori	*Black Box*
	Ubah data kategori	*Black Box*
Detail Barang	Tampilkan detail barang	*Black Box*
	Tambah detail barang	*Black Box*
	Ubah detail barang	*Black Box*
Data Barang	Tampilkan data barang	*Black Box*
	Tambah data barang	*Black Box*
	Ubah data barang	*Black Box*
Logout	Menghapus *session user*	*Black Box*

4.6.2 Kasus dan Hasil Pengujian

Berdasarakan rencana pengujian yang telah disusun, maka dapat dilakukan pengujian sebagai berikut:

4.6.2.1 Pengujian Registrasi

Berikut ini adalah pengujian registrasi sebagai berikut :

Tabel 4.5 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 : bugi Alamat Email : bugi@yahoo.com Provinsi : DKI Jakarta Kota : Jakarta Utara Alamat : Jl. gunungbatu Kodepos : 14130 Telepon : 085689785645 Password : bugi123 Konfirmasi password :bugi123	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 bahwa field tersebut harus terisi. Jika data masukan benar muncul pesan "Pendaftaran berhasil, silahkan cek email Anda untuk mengkonfirmasi pendaftaran"	Tampil pesan "Pendaftaran berhasil, silahkan aktivasi akun anda."	[√] Diterima [ ] Ditolak
Memasukan data kosong selain di field	Tampilkan pesan	Dapat menampilkan	[√] Diterima

yang wajib diisi	kesalahan	pesan kesalahan ”*field* ini harus diisi”	[ ] Ditolak

4.6.2.2 Pengujian Login

Tabel 4.6 Pengujian Login Member

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan alamat *email* : bugi@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* : bugi123	Mengisikan data *password* pada *field password*. Jika salah maka akan muncul pesan “Maaf, username dan password salah!!”	Dapat mengisi *fieldpassword* dan muncul pesan “Maaf, email atau 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, email atau Password Salah!!!”	[√] Diterima [ ] Ditolak

mengosongkan data *email* atau *password*	Tidak dapat login	Dapat menampilkan pesan kesalahan ”Maaf, email atau Password Salah!!!”	[√] Diterima [ ] Ditolak

4.6.2.3 Pengujian Logut

Tabel 4.7 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

4.6.2.4 Pengujian Setelan Akun

Tabel 4.8 Pengujian Setelan 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

4.6.2.4 Pengujian Ganti Password *Member*

Berikut dapat dilihat pengujian dari ganti password *member* pada tabel 4.6:

Tabel 4.9 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	Dapat menyimpan data *password* yang baru dengan	[√] Diterima [ ] Ditolak

	database	muncul pesan “ Data berhasil diubah”	
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

4.6.2.5 Pengujian Keranjang Belanja

Tabel 4.10 Pengujian 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 update	Data jumlah yang diubah tersimpan *didatabase*	Dapat menampilkan perubahan data jumlah	[√] Diterima [ ] Ditolak
Klik tombol lanjutkan belanja	Kembali ke halaman home	Dapat kembali ke halaman home	[√] Diterima [ ] Ditolak
Klik selesai belanja	Menampilkan form pengiriman	Dapat menampilkan form *checkout*	[√] Diterima [ ] Ditolak

4.6.2.6 Pengujian Menu Produk

Tabel 4.11 Pengujian Menu Barang

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih menu barang	Menampilkan data barang yang ada di *database*	menampilkan data barang yang ada di *data base*	[√] Diterima [ ] Ditolak
Klik nama barang	Menampilkan detail barang	Dapat menampilkan detail barang	[√] Diterima [ ] Ditolak
piliht *add to chart*	Menuju ke keranjang belanja	Dapat menampilkan keranjang belanja	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
ukuran tidak dipilih	Menampilkan pesan “ukuran	Pesan “ ukuran barang belum	[√] Diterima

	barang belum dipilih”	dipilih ”	[ ] Ditolak
Memasukan jumlah (qty) barang lebih dari stok	Menampilkan pesan “stok tidak mencukupi”	Dapat menampilkan pesan ““stok tidak mencukupi”	[√] Diterima [ ] Ditolak

4.6.2.7 Pengujian Menu Kategori

Tabel 4.12 Pengujian Menu Kategori

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pilih menu kategori	Menampilkan data kategori yang ada di *database*	Dapat menampilkan data kategori yang ada di*data base*	[√] Diterima [ ] Ditolak
pilih detail pada data produk	Menampilkan detail produk	Dapat menampilkan detail produk	[√] Diterima [ ] Ditolak
pilih *add to chart*	Menuju ke keranjang belanja	Dapat menampilkan keranjang belanja	[√] Diterima [ ] Ditolak

4.6.2.8 Pengujian *Form CheckOut*

Tabel 4.13 Pengujian CheckOut

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pilih tombol *checkout*	Data pembelian yang disimpan di dalam *databese*	Dapat menyimpan data pembelian yang sisimpan di dalam *database*.	[√] Diterima [ ] Ditolak

4.6.2.9 Pengujian Form Pengiriman

Tabel 4.14 Pengujian Pengiriman

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pilih jasa pengiriman	Menampilkan form jasa pengiriman	Dapat menampilkan jasa pengiriman	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Jenis pengiriman tidak diisi	Menampilkan pesan kesalahan "Anda belum memilih Jenis pengiriman!"	Dapat menampilkan pesan kesalahan "Anda belum memilih Jenis pengiriman!"	[√] Diterima [ ] Ditolak

4.6.2.10 Pengujian Konfirmasi Pembayaran

Tabel 4.15 Pengujian Pembayaran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data konfirmasi pembayaran, contoh No. invoice : 2 Nama Bank : Mandiri	Menampilkan *form* konfirmasi	Dapat memunculkan pesan form konfirmasi.	[√] Diterima [ ] Ditolak

Atas Nama : demo Jumlah Transfer : 90			
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Masukan data konfirmasi pembayaran no transaksi yang sama, contoh Nama Bank : Mandiri No. transaksi : 2 Atas Nama : demo Jumlah Transfer : 90	Ada pesan error bahwa no transaksi tersebut telah digunakan	Terdapat pesan error "no transaksi sudah pernah tercatat pada database kami"	[√] Diterima [ ] Ditolak

4.6.2.11 Pengujian Login Admin

Tabel 4.16 Pengujian Login Admin

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : bugi@yahoo.com *Password* : admin	Login berhasil	Nama *email, password* sesuai dengan tabel admin Login berhasil sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**

Email : bugi@yahoo.com *password* : bugi	Tidak dapat melakukan login dan menampilkan pesan “ username or password incorect”	User tidak dapat login dan menampilakn pesan “username or password incorect ” sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Email : bugi@gmail.com *password* : bugi	Tidak dapat melakukan login dan menampilkan pesan “ username or password incorect”	Tidak dapat melakukan login dan menampilkan pesan “ username or password incorect”	[√] Diterima [ ] Ditolak

4.6.2.12 Pengujian Ubah Password Admin

Tabel 4.17 Pengujian Ubah Password Admin

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data yang benar	*Password* baru berhasil disimpan di *database*	Dapat menyimpan data *password* yang baru dengan muncul pesan “ Data berhasil diubah”	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
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*	Dapat menampilkan pesan kesalahan *field*	[√] Diterima [ ] Ditolak

	harus diisi	harus diisi	

4.6.2.13 Pengujian Ubah Profil Admin

Tabel 4.18Pengujian Ubah Profil Admin

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data profil yang akan diubah	Tercantum pada text box data yang akan diubah	Dapat mengisi *field* data yang akan diubah dengan benar	[√] Diterima [ ] Ditolak
pilih tombol ubah	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[√] Diterima [ ] Ditolak

4.6.2.14 Pengujian Pengolahan Data Kategori

Tabel 4.19 Pengujian Kelola Kategori

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pilih tombol Edit	Menampilkan *form* edit kategori	Dapat menampilkan *form* ubah kategori	[√] Diterima [ ] Ditolak
pilih tombol simpan	Tampil pesan "Data berhasil disimpan"	Tampil pesan "Data berhasil disimpan"	[√] Diterima [ ] Ditolak
pilih tombol batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input kategori yang	Tampilkan pesan kesalahan	Dapat menampilkan	[√] Diterima

sama	"kategori sudah ada"	pesan kesalahan "kategori sudah ada"	[ ] Ditolak

4.6.2.15 Pengujian Tambah Kategori

Tabel 4.20 Pengujian Tambah Kategori

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Kategori : Sepatu	Penyimpanan berhasil	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah kategori	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input Kategori yang sama	Muncul pesan kesalahan "kategori gagal disimpan, nama kategori tidak boleh sama"	Dapat muncul pesan kesalahan "kategori gagal disimpan, nama kategori tidak boleh sama"	[√] Diterima [ ] Ditolak

4.6.2.16 Pengujian Tambah Data Barang

Tabel 4.21 Pengujian Tambah Data Barang

Kasus dan Hasil Uji (Data Normal)

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Kategori : Baju Wanita Nama Barang : Baju maroon Harga : 50 Warna : Maroon Diskon : 0 Masukan gambar : 	Barang berhasil disimpan	Barang masuk ke database sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukkan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data barang sama	Tampilkan pesan kesalahan "*field* tersebut sudah tersedia"	Dapat menampilkan pesan kesalahan" *field* sudah tersedia"	[√] Diterima [ ] Ditolak

4.6.2.17 Pengujian Tambah Provinsi

Tabel 4.22 Pengujian Tambah Provinsi

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Provinsi : Jawa Barat	Provinsi berhasil disimpan	Nama provinsi berhasil masuk ke database sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama provinsi yang sudah ada	Tampilkan pesan kesalahan "Provinsi gagal disimpan, nama provinsi tidak boleh sama"	Dapat menampilkan pesan kesalahan "Provinsi gagal disimpan, nama provinsi tidak boleh sama"	[√] Diterima [ ] Ditolak

4.6.2.18 Pengujian Tambah Detail Barang

Tabel 4.23 Pengujian Detail Barang

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Barang : BAju Maroon Ukuran : M Stok : 5 Berat : 0.7 Promo : ya Tanggal masuk :	Barang berhasil disimpan	Barang masuk ke database sesuai yang diharapkan	[√] Diterima [ ] Ditolak

01/01/2011			
Kasus dan Hasil Uji (Data salah)			
Data Masukkan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan Data barang yang sama	Tampilkan pesan kesalahan "barang tersebut sudah tersedia"	Dapat menampilkan pesan kesalahan barang tersebut sudah tersedia"	[√] Diterima [ ] Ditolak

4.6.2.19 Pengujian Tambah Rekening

Tabel 4.24 Pengujian Tambah Rekening

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Bank : BNI Nama Pemilik Rekening : Bugi Ramadhan Cabang : Bandung No Rekening : 106008996746 Masukan Gambar : 	No Rekening berhasil disimpan	No Rekening masuk ke database sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukkan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan no rekening yang sama	Tampilkan pesan kesalahan "no rekening tersebut sudah tersedia"	Dapat menampilkan pesan kesalahan no rekening tersebut sudah tersedia"	[√] Diterima [ ] Ditolak

4.6.2.20 Pengujian Tambah Ongkos Kirim

Tabel 4.25 Pengujian Tambah Ongkos Kirim

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Provinsi : Jawa Barat Nama Kota : CIMAHI Jasa Pengiriman : JNE Jenis Pengiriman : Reguler Biaya Kirim : 50	Ongkos Kirim berhasil disimpan	Ongkos Kirim masuk ke database sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukkan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data sama	Tampilkan pesan kesalahan “maaf, pengisian form tidak benar”	Dapat menampilkan pesan kesalahan” maaf, pengisian form tidak benar”	[√] Diterima [ ] Ditolak

4.6.2.21 Pengujian Kelola Member

Tabel 4.26 Pengujian Kelola Member

Kasus dan Hasil Uji (Data Normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pilih *check box* pada data yang akan dihapus hapus	Memilih data yang akan dihapus	Data yang akan dihapus telah dipilih	[√] Diterima [ ] Ditolak
pilih hapus	Data akan dihapus dari	Data telah dihapus dari database.	[√] Diterima

	database. Jika tidak ada data yang dipilih akan muncul pesan kesalahan	Muncul pesan "pilih data yang akan dihapus" bila data belum dipilih	[ ] Ditolak

4.6.2.22 Pengujian Tambah Jasa Pengiriman

Tabel 4.27 Pengujian Tambah Jasa Pengiriman

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Jasa Pengirman : JNE	Jasa pengiriman berhasil disimpan	Jasa pengiriman masuk ke database sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukkan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input data masukan yang sama	Tampilkan pesan kesalahan "maaf, isikan form dengan benar"	Dapat menampilkan pesan kesalahan" maaf, isikan form dengan benar"	[√] Diterima [ ] Ditolak

4.6.2.23 Pengujian Tambah Ukuran

Tabel 4.28 Pengujian Tambah Ukuran

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Pemilik Ukuran : S Deskripsi Ukuran : Small	Ukuran berhasil disimpan	Ukuran masuk ke database sesuai yang diharapkan	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data salah)			
Data Masukkan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Input Data yang sama sama	Tampilkan pesan kesalahan " ukuran gagal disimpan, ukuran tidak boleh sama"	Dapat menampilkan pesan kesalahan " ukuran gagal disimpan, ukuran tidak boleh sama"	[√] Diterima [ ] Ditolak

4.6.2.24 Pengujian Menu Warna

Tabel 4.29 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 datawarna	[√] Diterima [ ] Ditolak
Klik tombol *edit*	Menampilkan *form*ubahwarnaa	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

4.6.2.25 Pengujian Tambah Warna

Tabel 4.30 Pengujian Tambah Warna

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tombol tambah warna	Menampilkan *form*tambahwarna	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

4.6.2.26 Pengujian Kelola Data Kurs

Tabel 4.31 Pengujian Kelola Data Kurs

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kurs : 9025	Kurs berhasil disimpan	Kurs masuk ke database sesuai yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukkan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data Kurs : bugi	Tampilkan pesan kesalahan "*fiel* harus berupa angka"	Dapat menampilkan pesan kesalahan" *fiel* harus berupa angka"	[√] Diterima [ ] Ditolak

4.6.2.27 Pengujian Kelola Data Transaksi

Tabel 4.32 Pengujian Kelola Data Transaksi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pilih pengeolahan data transaksi	Menampilkan *form* transaksi	Dapat menampilkan *form* lihat transaksi	[√] Diterima [ ] Ditolak
pilih edit pada tabel pilihan	Menampilkan edit transaksi	Dapat menampilkan *form edit* transaksi	[√] Diterima [ ] Ditolak
pilih konfirmasi	Mengubah status pengiriman	Dapat menampilkan no resi	[√] Diterima [ ] Ditolak
Isi No Resi : 123456789	No resi berhasil disimpan	No resi berhasil di simpan didalam database	[√] Diterima [ ] Ditolak

4.6.2.28 Pengujian Kelola Data Retur

Tabel 4.33 Pengujian Kelola Data Retur

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
pilih pengeolahan data retur	Menampilkan *form* retur	Dapat menampilkan *form* lihat retur	[√] Diterima [ ] Ditolak
pilih edit pada tabel pilihan	Menampilkan edit retur	Dapat menampilkan *form edit* retur	[√] Diterima [ ] Ditolak
pilih diterima	Mengubah status pengiriman	Mengubah status retur	[√] Diterima [ ] Ditolak

4.6.3 Kesimpulan Hasil Pengujian Alpa

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

4.7 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan membuat kuesioner mengenai kepuasan user dengan kandungan poin syarat user friendly untuk selanjutnya dibagikan kepada sebagian user dengan mengambil sample sebanyak 10 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

Kuesioner disebarkan menggunakan teknik sampling yaitu *Simple Random Sampling* yang disebarkan kepada 10 pengguna. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner ini terdiri dari 7 pertanyaan (contoh kuesioner dapat diliihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 5, dimana daftar pertanyaan yang diberikan adalah sebagai berikut

4.7.1 Kuesioner Pengujian Beta Untuk Admin

Pada pengujian beta ini saya melakukan tahapan wawancara berstruktur yang diajukan pada pemilik Butik De'Smoothy, berikut adalah butir-butir wawancara yang saya ajukan pada pemilik Butik De'Smoothy :

Penulis	Apakah anda setuju tampilan perangkat lunak untuk pengolahan data Butik De'Smoothy sudah bagus?
Owner	Setuju, warnanya cukup enak dilihat.
Penulis	Apakah anda setuju instruksi yang terdapat dalam *website* tersebut mudah dimengerti?
Owner	Awalnya sempat bingung karena belum terbiasa,tapi setelah diterangkan,ternyata mudah.
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	Untuk pengujian di lapangan, kami pihak De'Smoothy belum merasakan efek dari penjualan secara online. Semoga dlm beberapa waktu kedepan bisa terlihat.
Penulis	Apakah anda setuju *website* ini dapat membantu dalam membuat laporan penjualan ?
Owner	Setuju karena disediakan fasilitas laporan.

Menurut hasil wawancara secara keseluruhan untuk *website* yang dibangun *owner* merasa cukup puas dengan adanya *website* Butik De'Smoothy, dilihat dari segi tampilan *website* Butik De'Smoothy menarik dan tidak membosankan. Dari segi instruksi *owner* 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.

4.7.2 Kuesioner Pengujian Beta Untuk Pengunjung dan Member

Berikut adalah hasil dari kuesioner pengujian beta bagi pengunjung dan Member :

1. Apakah anda setuju bahwa tampilan *website* butikdesmoothy.com cukup menarik dan mudah dipahami ?

No	Keterangan	Responden	Presentase (%)

1	Sangat setuju		
2	Setuju	3	40%
3	Cukup Setuju	5	50%
4	Kurang Setuju	2	10%
5	Tidak Setuju		

2. Apakah Anda setuju bahwa *website* E-commerce ini dapat membantu Anda untuk mendapatkan informasi terbaru tentang produk yang ada di butikdesmoothy?

No	Keterangan	Responden	Presentase (%)
1	Sangat setuju	3	30%
2	Setuju	4	40%
3	Cukup Setuju	5	20%
4	Kurang Setuju	1	10%
5	Tidak Setuju		

3. Apakah Anda setuju bahwa *website* ini akan lebih mempermudah Anda dalam melakukan transaksi pembelian dibandingkan transaksi secara manual?

No	Keterangan	Responden	Presentase (%)
1	Sangat setuju	1	10%
2	Setuju	6	70%
3	Cukup Setuju	2	20%
4	Kurang Setuju	1	
5	Tidak Setuju		

4. Apakah Anda setuju bahwa *website* ini pemilik dapat memperluas pangsa pasar?

No	Keterangan	Responden	Presentase (%)
1	Sangat Setuju	2	20%
2	Setuju	4	40%
3	Cukup Setuju	3	30%
4	Kurang Setuju	1	10%
5	Tidak Setuju		

5. Apakah Anda setuju sistem pembayaran yang digunakan akan lebih memudahkan anda?

No	Keterangan	Responden	Presentase (%)
1	Sangat Setuju	2	20%
2	Setuju	3	30%
3	Cukup Setuju	4	40%
4	Kurang Setuju	1	10%
5	Tidak Setuju		

6. Apakah Anda setuju bahwa *website* ini menyediakan informasi bagi konsumen untuk melihat produk yang ditawarkan?

No	Keterangan	Responden	Presentase (%)
1	Sangat Setuju	2	20%
2	Setuju	4	40%
3	Cukup Setuju	3	30%
4	Kurang Setuju	1	10%
5	Tidak Setuju		

7. Apakah Anda setuju bahwa *website* ini mempermudah pembeli untuk mendapatkan barang yang diinginkan?

No	Keterangan	Responden	Presentase (%)
1	Sangat Setuju	2	20%
2	Setuju	1	10%
3	Cukup Setuju	4	40%
4	Kurang Setuju	3	30%
5	Tidak Setuju		

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

Keterangan:

P = Banyaknya jawaban responden tiap soal

Q = Jumlah responden

Y = Nilai Presentase

No resp	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				√					√			√				√					√					√					√				
2				√					√						√			√			√						√					√			
3				√					√				√					√				√							√		√				
4			√							√					√		√					√						√					√		
5			√						√					√			√						√				√							√	
6		√							√				√					√				√					√						√		
7			√							√			√				√						√			√							√		
8			√						√					√					√				√				√						√		
9		√							√					√			√						√					√						√	
10		√							√				√			√								√		√								√	
Point	30					42					36					23					24					19					28				
Y= P/R	3					4.2					3.6					2.3					2.4					1.9					2.8				

Kesimpulan :

Untuk pertanyaan ke 1 responden rata-rata memilih bahwa tampilan *website* Butik De'Smoothy *user friendly*, pertanyaan ke 2 responden pun setuju dengan adanya *website e-commerce* konsumen mudah mendapatkan informasi tebaru tentang produk yang ada di Butik De'Smoothy. Untuk pertanyaan ke 3

dengan adanya *website* ini konsumen belum yakin 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 pangsa 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 Butik De'Smoothy. 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.