BAB 4
IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini akan dijelaskan mengenai implementasi dan pengujian sistem yang dibangun. Hasil dari perancangan pada tahap sebelumnya diimplementasikan dalam bahasa pemrograman. Setelah tahap implementasi selesai, dilanjutkan dengan tahap pengujian terhadap sistem yang dibangun.

4.1 Implementasi Sistem

Tahap implementasi sistem merupakan tahap menterjemahkan perancangan, berdasarkan hasil analisis dalam bahasa pemrograman serta penerapan perangkat lunak pada keadaan yang sesungguhnya. Wujud dari hasil tahap implementasi ini nantinya adalah sebuah sistem yang siap untuk diuji dan digunakan.

Untuk mendukung aplikasi yang akan diterapkan pada lingkungan implementasi, maka dalam hal ini menggunakan perangkat keras dan perangkat lunak yang menunjang dalam pembangunan aplikasi *e-commerce* pada Grutty Shoes & Bags.

4.1.1 Perangkat Lunak Yang Digunakan

Perangkat lunak yang digunakan dalam membangun aplikasi *e-commerce* pada Grutty Shoes & Bags adalah sebagai berikut :

Tabel 4.1 Perangkat Lunak Yang Digunakan

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows 7 Ultimate
2	Bahasa Pemrograman	PHP
3	*DBMS*	MySQL 5.0.41
4	*Web browser*	Mozilla firefox 3.6
5	*Code editor*	Adobe Dreamweaver CS5
6	DFD modeler	Microsoft Visio 2007

4.1.2 Perangkat Keras Yang Digunakan

Perangkat keras yang digunakan dalam membangun aplikasi *e-commerce* pada Grutty Shoes & Bags adalah sebagai berikut:

Tabel 4.2 Perangkat Keras Yang Digunakan

Perangkat Keras	Spesifikasi
Processor	Prosesor Intel Core 2 Duo T5870
RAM	1Gb
Harddisk	250Gb
Monitor	Resolusi 1366x768 pixel
VGA	Intel GMA X3100 256 MB

4.2 Implementasi *Database*

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

1. Tabel Barang

```
CREATE TABLE IF NOT EXISTS `barang` (
  `id_barang` bigint(20) NOT NULL auto_increment,
  `id_induk_barang` bigint(20) NOT NULL default '0',
  `id_kategori` bigint(11) NOT NULL,
  `id_pegawai` int(11) NOT NULL default '2',
  `tanggal_masuk` date NOT NULL,
  `terakhir_diperbarui` date NOT NULL,
  `kode_barang` varchar(255) NOT NULL,
  `warna` varchar(255) NOT NULL,
  `ukuran` varchar(255) NOT NULL,
  `deskripsi` text NOT NULL,
  `berat` double NOT NULL default '0',
  `stok` int(11) NOT NULL default '0',
  `harga` double NOT NULL default '0',
  `diskon` double NOT NULL default '0',
  `url_gambar` longtext character set latin1 collate
latin1_general_ci,
  `ditampilkan` enum('true','false') default 'true',
  PRIMARY KEY  (`id barang`),
  KEY `id_kategori` (`id_kategori`),
  KEY `id_induk_barang` (`id_induk_barang`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=42 ;
ALTER TABLE `barang`
  ADD CONSTRAINT `barang_ibfk_1` FOREIGN KEY (`id_induk_barang`)
REFERENCES `barang` (`id_induk_barang`),
  ADD CONSTRAINT `barang_ibfk_2` FOREIGN KEY (`id_kategori`)
REFERENCES `kategori` (`id_kategori`);
```

2. Tabel Detail_pesanan

```
CREATE TABLE IF NOT EXISTS `detail_pesanan` (
  `id_detail_pesanan` bigint(20) NOT NULL auto_increment,
  `id_pesanan` bigint(20) NOT NULL,
  `id_barang` bigint(20) NOT NULL,
  `rincian_barang` text collate latin1_general_ci NOT NULL,
  `berat` double NOT NULL default '0',
  `harga` double NOT NULL default '0',
  `diskon` double NOT NULL default '0',
  `jumlah` int(11) NOT NULL default '0',
  `status` enum('normal','stock-limit','stock-less') collate
latin1_general_ci NOT NULL default 'normal',
  `retur` enum('true','false') collate latin1_general_ci NOT NULL
default 'false',
  `jumlah_retur` int(11) NOT NULL,
```

```
  `alasan` text collate latin1_general_ci NOT NULL,
  `catatan` text collate latin1_general_ci NOT NULL,
  `jumlah_pengganti_retur` int(11) NOT NULL,
  `alasan2` text collate latin1_general_ci NOT NULL,
  `catatan2` text collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`id_detail_pesanan`),
  KEY `id_barang` (`id_barang`),
  KEY `id_pesanan` (`id_pesanan`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
AUTO_INCREMENT=73 ;
ALTER TABLE `detail_pesanan`
  ADD CONSTRAINT `detail_pesanan_ibfk_2` FOREIGN KEY (`id_barang`)
REFERENCES `barang` (`id_barang`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `detail_pesanan_ibfk_1` FOREIGN KEY
(`id_pesanan`) REFERENCES `pesanan` (`id_pesanan`) ON DELETE
CASCADE ON UPDATE CASCADE;
```

3. Tabel Halaman_web

```
CREATE TABLE IF NOT EXISTS `halaman_web` (
  `id_halaman_web` int(11) NOT NULL auto_increment,
  `id_pegawai` int(11) NOT NULL default '1',
  `tanggal_input` date NOT NULL,
  `terakhir_diperbarui` date NOT NULL,
  `judul` varchar(255) NOT NULL,
  `isi` longtext NOT NULL,
  `urutan` varchar(5) NOT NULL,
  `ditampilkan` enum('true','false') NOT NULL default 'true',
  PRIMARY KEY  (`id_halaman_web`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;
```

4. Tabel Jasa_pengiriman

```
CREATE TABLE IF NOT EXISTS `jasa_pengiriman` (
  `id_jasa_pengiriman` int(11) NOT NULL auto_increment,
  `nama_perusahaan` varchar(30) NOT NULL,
  `website` varchar(255) NOT NULL,
  `link_tracking` varchar(255) NOT NULL,
  PRIMARY KEY  (`id_jasa_pengiriman`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

5. Tabel Kategori

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` bigint(20) NOT NULL auto_increment,
  `nama_kategori` varchar(255) character set latin1 collate
latin1_general_ci NOT NULL,
  `deskripsi` text character set latin1 collate latin1_general_ci,
  `url_gambar` text character set latin1 collate
latin1_general_ci,
  `ditampilkan` enum('true','false') character set latin1 collate
latin1_general_ci NOT NULL default 'true',
  PRIMARY KEY  (`id_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;
```

6. Tabel Kota

```
CREATE TABLE IF NOT EXISTS `kota` (
  `id_kota` int(20) NOT NULL auto_increment,
  `id_provinsi` int(11) NOT NULL,
  `nama_kota` varchar(50) NOT NULL,
  PRIMARY KEY  (`id_kota`),
  KEY `id_provinsi` (`id_provinsi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=63 ;
ALTER TABLE `kota`
  ADD CONSTRAINT `kota_ibfk_1` FOREIGN KEY (`id_provinsi`)
REFERENCES `provinsi` (`id_provinsi`) ON DELETE CASCADE ON UPDATE
CASCADE;
```

7. Tabel Ongkos_Kirim

```
CREATE TABLE IF NOT EXISTS `ongkos_kirim` (
  `id_ongkos_kirim` bigint(20) NOT NULL auto_increment,
  `id_kota` int(11) NOT NULL default '1',
  `id_provinsi` int(11) NOT NULL default '1',
  `id_jasa_pengiriman` int(11) NOT NULL default '1',
  `id_tipe_pengiriman` int(11) NOT NULL default '1',
  `biaya` double NOT NULL default '0',
  `cod` enum('true','false') NOT NULL default 'false',
  `ditampilkan` enum('true','false') default 'true',
  PRIMARY KEY  (`id_ongkos_kirim`),
  KEY `id_kota` (`id_kota`),
  KEY `id_provinsi` (`id_provinsi`),
  KEY `id_jasa_pengiriman` (`id_jasa_pengiriman`),
  KEY `id_tipe_pengiriman` (`id_tipe_pengiriman`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=59 ;
ALTER TABLE `ongkos_kirim`
  ADD CONSTRAINT `ongkos_kirim_ibfk_8` FOREIGN KEY
(`id_tipe_pengiriman`) REFERENCES `tipe_pengiriman`
(`id_tipe_pengiriman`),
  ADD CONSTRAINT `ongkos_kirim_ibfk_6` FOREIGN KEY (`id_provinsi`)
REFERENCES `provinsi` (`id_provinsi`),
```

```
  ADD CONSTRAINT `ongkos_kirim_ibfk_7` FOREIGN KEY
(`id_jasa_pengiriman`) REFERENCES `jasa_pengiriman`
(`id_jasa_pengiriman`);
```

8. Tabel Pegawai

```
CREATE TABLE IF NOT EXISTS `pegawai` (
  `id_pegawai` int(11) NOT NULL auto_increment,
  `daftar` date NOT NULL,
  `username` varchar(20) NOT NULL,
  `password` varchar(64) NOT NULL,
  `email` varchar(255) NOT NULL,
  `akses` enum('0','1','2') NOT NULL default '0',
  `nama_pegawai` varchar(255) NOT NULL,
  `aktif` enum('true','false') NOT NULL default 'false',
  PRIMARY KEY  (`id_pegawai`),
  UNIQUE KEY `username` (`username`),
  UNIQUE KEY `email` (`email`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

9. Tabel Pelanggan

```
CREATE TABLE IF NOT EXISTS `pelanggan` (
  `id_pelanggan` bigint(20) NOT NULL auto_increment,
  `daftar` date NOT NULL,
  `email` varchar(100) character set latin1 collate
latin1_general_ci NOT NULL,
  `password` varchar(64) character set latin1 collate
latin1_general_ci NOT NULL,
  `nama_pelanggan` varchar(50) character set latin1 collate
latin1_general_ci NOT NULL,
  `alamat` text character set latin1 collate latin1_general_ci NOT
NULL,
  `kota` varchar(50) character set latin1 collate
latin1_general_ci NOT NULL,
  `provinsi` varchar(30) character set latin1 collate
latin1_general_ci NOT NULL,
  `kode_pos` varchar(10) character set latin1 collate
latin1_general_ci NOT NULL,
  `tlp` varchar(20) character set latin1 collate latin1_general_ci
NOT NULL,
  `hp` varchar(20) character set latin1 collate latin1_general_ci
default NULL,
  `fax` varchar(20) character set latin1 collate latin1_general_ci
default NULL,
  `aktif` enum('true','false') character set latin1 collate
latin1_general_ci NOT NULL default 'false',
  PRIMARY KEY  (`id_pelanggan`),
  UNIQUE KEY `email` (`email`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=13 ;
```

10. Tabel Pengaturan_web

```
CREATE TABLE IF NOT EXISTS `pengaturan_web` (
  `id_pengaturan_web` int(11) NOT NULL auto_increment,
  `id_pegawai` int(11) NOT NULL default '1',
  `kode` varchar(255) collate latin1_general_ci NOT NULL,
  `jenis`
enum('text','textarea','chechkbox','radio','select','multicheck','
integer','decimal','email','truefalse') collate latin1_general_ci
NOT NULL default 'text',
  `nama` varchar(255) collate latin1_general_ci NOT NULL,
  `anggapan` longtext collate latin1_general_ci,
  `isi` longtext collate latin1_general_ci,
  `urutan` int(11) NOT NULL default '0',
  `keterangan` longtext collate latin1_general_ci,
  `ditampilkan` enum('true','false') collate latin1_general_ci NOT
NULL default 'true',
  PRIMARY KEY  (`id_pengaturan_web`),
  UNIQUE KEY `kode` (`kode`),
  KEY `id_pegawai` (`id_pegawai`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
AUTO_INCREMENT=64 ;
ALTER TABLE `pengaturan_web`
  ADD CONSTRAINT `pengaturan_web_ibfk_1` FOREIGN KEY
(`id_pegawai`) REFERENCES `pegawai` (`id_pegawai`);
```

11. Tabel Pesanan

```
CREATE TABLE IF NOT EXISTS `pesanan` (
  `id_pesanan` bigint(20) NOT NULL auto_increment,
  `id_pelanggan` bigint(20) NOT NULL,
  `id_ongkos_kirim` bigint(20) NOT NULL,
  `tanggal_pesanan` date NOT NULL,
  `tanggal_kadaluwarsa` date NOT NULL,
  `nama_pengirim` varchar(255) character set latin1 collate
latin1_general_ci NOT NULL,
  `nama_penerima` varchar(255) character set latin1 collate
latin1_general_ci NOT NULL,
  `alamat` text character set latin1 collate latin1_general_ci NOT
NULL,
  `kota` varchar(50) character set latin1 collate
latin1_general_ci NOT NULL,
  `provinsi` varchar(50) character set latin1 collate
latin1_general_ci NOT NULL,
  `kode_pos` varchar(10) character set latin1 collate
latin1_general_ci NOT NULL,
  `tlp` varchar(20) character set latin1 collate latin1_general_ci
default NULL,
  `hp` varchar(20) character set latin1 collate latin1_general_ci
default NULL,
  `fax` varchar(20) character set latin1 collate latin1_general_ci
default NULL,
  `total_barang` int(11) NOT NULL,
```

```
  `subtotal_bayar` double NOT NULL,
  `ongkos_kirim` double NOT NULL,
  `total_bayar` double NOT NULL,
  `bisa_cod` enum('true','false') default 'true',
  `mata_uang` enum('rp','pp','$') character set latin1 collate
latin1_general_ci NOT NULL default 'rp',
  `harga_paypal` double NOT NULL default '10000',
  `tipe_pembayaran` varchar(50) character set latin1 collate
latin1_general_ci default NULL,
  `kode_pembayaran` varchar(50) character set latin1 collate
latin1_general_ci default NULL,
  `tanggal_pembayaran` date default NULL,
  `catatan_pembayaran` text character set latin1 collate
latin1_general_ci NOT NULL,
  `lunas` enum('true','false') character set latin1 collate
latin1_general_ci default 'false',
  `kode_pengiriman` varchar(50) character set latin1 collate
latin1_general_ci default NULL,
  `tanggal_pengiriman` date default NULL,
  `tanggal diterima` date default NULL,
  `catatan_penerimaan` longtext NOT NULL,
  `status` varchar(255) character set latin1 collate
latin1_general_ci default 'cart',
  PRIMARY KEY  (`id_pesanan`),
  KEY `id pelanggan` (`id pelanggan`),
  KEY `id_ongkos_kirim` (`id_ongkos_kirim`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=426 ;
ALTER TABLE `pesanan`
  ADD CONSTRAINT `pesanan_ibfk_2` FOREIGN KEY (`id_ongkos_kirim`)
REFERENCES `ongkos_kirim` (`id_ongkos_kirim`) ON DELETE CASCADE ON
UPDATE CASCADE,
  ADD CONSTRAINT `pesanan_ibfk_1` FOREIGN KEY (`id_pelanggan`)
REFERENCES `pelanggan` (`id_pelanggan`) ON DELETE CASCADE ON
UPDATE CASCADE;
```

12. Tabel Provinsi

```
CREATE TABLE IF NOT EXISTS `provinsi` (
  `id_provinsi` int(11) NOT NULL auto_increment,
  `nama_provinsi` varchar(30) NOT NULL,
  PRIMARY KEY  (`id_provinsi`),
  UNIQUE KEY `nama_provinsi` (`nama_provinsi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=95 ;
```

13. Tabel Riwayat_stok

```
CREATE TABLE IF NOT EXISTS `riwayat_stok` (
  `id_riwayat_stok` int(11) NOT NULL auto_increment,
  `id_barang` bigint(11) NOT NULL,
  `tanggal` varchar(25) NOT NULL,
  `keterangan` varchar(255) NOT NULL,
```

```
  `stok_awal` int(11) NOT NULL,
  `perubahan_stok` int(11) NOT NULL,
  `stok_akhir` int(11) NOT NULL,
  PRIMARY KEY  (`id_riwayat_stok`),
  KEY `id_barang` (`id_barang`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=42 ;
ALTER TABLE `riwayat_stok`
  ADD CONSTRAINT `riwayat_stok_ibfk_1` FOREIGN KEY (`id_barang`)
REFERENCES `barang` (`id_barang`) ON DELETE CASCADE ON UPDATE
CASCADE;
```

14. Tabel Tipe_pengiriman

```
CREATE TABLE IF NOT EXISTS `tipe_pengiriman` (
  `id_tipe_pengiriman` int(11) NOT NULL auto_increment,
  `nama_tipe` varchar(255) NOT NULL,
  PRIMARY KEY  (`id_tipe_pengiriman`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

15. Tabel Umpan_balik

```
CREATE TABLE IF NOT EXISTS `umpan_balik` (
  `id_umpan_balik` bigint(20) NOT NULL auto_increment,
  `id_pelanggan` bigint(20) NOT NULL,
  `id_barang` bigint(20) NOT NULL default '0',
  `tanggal_input` date NOT NULL,
  `terakhir_diperbarui` date NOT NULL,
  `jenis`
enum('comment','rating','view','like','recommended','wish','order'
,'buy') default 'comment',
  `isi` longtext character set latin1 collate latin1_general_ci
NOT NULL,
  `ditampilkan` enum('true','false') default 'true',
  PRIMARY KEY  (`id_umpan_balik`),
  KEY `id_pelanggan` (`id_pelanggan`),
  KEY `id_barang` (`id_barang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
ALTER TABLE `umpan_balik`
  ADD CONSTRAINT `umpan_balik_ibfk_2` FOREIGN KEY (`id_barang`)
REFERENCES `barang` (`id_barang`) ON DELETE CASCADE ON UPDATE
CASCADE,
  ADD CONSTRAINT `umpan_balik_ibfk_1` FOREIGN KEY (`id_pelanggan`)
REFERENCES `pelanggan` (`id_pelanggan`) ON DELETE CASCADE ON
UPDATE CASCADE;
```

4.3 Implementasi Antar Muka

Implementasi antar muka dilakukan pada setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna. Adapun untuk tampilan implementasi akan dilampirkan pada halaman lampiran.

Tabel 4.3 Implementasi Antar Muka

Halaman Login Admin	Index.php
Halaman Utama Admin	Index2.php
Halaman Profil Admin	Profil Admin.php
Halaman Edit Profil Admin	Edit Profil Admin.php
Halaman Pengolahan Petugas	Pengolahan Petugas.php
Halaman Tambah Petugas	Tambah Petugas.php
Halaman Edit Petugas	Edit Petugas.php
Halaman Laporan Pesanan	Laporan Pesanan.php
Halaman Laporan Pengiriman	Laporan Pengiriman.php
Halaman Laporan Produk	Laporan Produk.php
Halaman Laporan Retur	Laporan Retur.php
Halaman Login Operator	Index.php
Halaman Utama Operator	Index2.php
Halaman Profil Operator	Profil Operator.php
Halaman Edit Profil Operator	Edit Profil Operator.php
Halaman Pengolahan Data Pemesan	Pengolahan Data Pemesan.php
Halaman Edit Data Pemesan	Edit Data Pemesan.php
Halaman Pengolahan Data Kategori	Pengolahan Data Kategori.php
Halaman Tambah Kategori	Tambah Kategori.php
Halaman Edit Kategori	Edit Kategori.php
Halaman Pengolahan Data Barang	Pengolahan Data Barang.php
Halaman Tambah Barang	Tambah Barang.php
Halaman Edit Barang	Edit Barang.php
Halaman Pengolahan Data Kota	Pengolahan Data Kota.php
Halaman Tambah Kota	Tambah Kota.php
Halaman Edit Kota	Edit Kota.php
Halaman Pengolahan Ongkos Kirim	Pengolahan Ongkos Kirim.php
Halaman Tambah Ongkos Kirim	Tambah Ongkos Kirim.php
Halaman Edit Ongkos Kirim	Edit Ongkos Kirim.php
Halaman Pengolahan Jasa Pengiriman	Pengolahan Jasa Pengiriman.php
Halaman Tambah Jasa Pengiriman	Tambah Jasa Pengiriman.php

Halaman Pengolahan Data Barang	Pengolahan Data Barang.php
Halaman Tambah Barang	Tambah Barang.php
Halaman Edit Barang	Edit Barang.php
Halaman Pengolahan Data Kota	Pengolahan Data Kota.php
Halaman Tambah Kota	Tambah Kota.php
Halaman Edit Kota	Edit Kota.php
Halaman Pengolahan Ongkos Kirim	Pengolahan Ongkos Kirim.php
Halaman Tambah Ongkos Kirim	Tambah Ongkos Kirim.php
Halaman Edit Ongkos Kirim	Edit Ongkos Kirim.php
Halaman Pengolahan Jasa Pengiriman	Pengolahan Jasa Pengiriman.php
Halaman Tambah Jasa Pengiriman	Tambah Jasa Pengiriman.php
Halaman Edit Jasa Pengiriman	Edit Jasa Pengiriman.php
Halaman Utama Pengunjung	index.php
Halaman Profil	Profil.php
Halaman Cara Belanja	Cara Belanja.php
Halaman Daftar	Daftar.php
Halaman Lupa Password	Lupa Password.php
Halaman Utama Member	Index.php

4.4 Pengujian Perangkat Lunak

Agar hasil implementasi benar-benar memenuhi kebutuhan maka dilakukan pengujian terhadap sistem. Pengujian dilakukan dengan metode pengujian *black box* yang berfokus pada persyaratan fungsional perangkat lunak.

4.4.1 Pengujian Alpha

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

Berikut dapat dilihat Skenario pengujian aplikasi *pengunjung* pada tabel dibawah ini:

Tabel 4.4 Skenario pengujian aplikasi *pengunjung*

Kelas Uji	Butir Uji	Jenis Pengujian
Pendaftaran	Menampilkan *form* pendaftaran	*Black Box*
	Isi *form* pendaftaran	*Black Box*

	Simpan data pendaftaran	*Black Box*
Aktivasi	Buka email	*Black Box*
	Aktivasi pendaftaran	*Black Box*

Berikut dapat dilihat pengujian dari Skenario pengujian aplikasi *pemesan* pada tabel dibawah ini:

Tabel 4.5 Skenario pengujian aplikasi *pemesan*

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Logout	Menghapus *session user*	*Black Box*
Lupa *Password*	Menampilkan *form* lupa password	*Black Box*
	Masukkan data *password* baru	*Black Box*
	Kirim *password* baru ke email pemesan	*Black Box*
Profil	Menampilkan *form* profil pemesan	*Black Box*
Edit Profil	Menampilkan *form* edit profil	*Black Box*
	Ubah data pemesan	*Black Box*
	Simpan perubahan	*Black Box*
Keranjang Belanja	Menampilkan data detail pesanan	*Black Box*
	Mengubah data detail pesanan	*Black Box*
	Menghapus data detail pesanan	*Black Box*
Form pengiriman barang	Menampilkan form pengiriman barang	*Black Box*
	Isi form pengiriman barang	*Black Box*
Form konfirmasi pesanan	Menampilkan konfirmasi pesanan barang, total pembayaran dan data pengiriman barang	*Black Box*

Form pembayaran	Menampilkan total bayar dan cara pembayaran	*Black Box*
	Isi data pembayaran	*Black Box*
	Menyimpan data pembayaran	*Black Box*
Pesanan	Menampilkan data pesanan yang pernah dilakukan	*Black Box*
	Menampilkan data detail pesanan	*Black Box*
	Menampilkan pembayaran pesanan	*Black Box*
	Menampilkan pesanan yang bisa di retur	*Black Box*
Form retur	Menampilkan pesanan dan produk yang akan di kembalikan	*Black Box*

Berikut dapat dilihat pengujian dari Skenario pengujian aplikasi *administrator* pada tabel dibawah ini:

Tabel 4.6 Skenario pengujian aplikasi administrator

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Logout	Menghapus *session user*	*Black Box*
Profil Anda	Menampilkan *form* profil administrator	*Black Box*
Edit Profil	Menampilkan *form* edit profil	*Black Box*
	Ubah data administrator	*Black Box*
	Simpan perubahan	*Black Box*
Pengolahan petugas	Tampilkan data petugas	*Black Box*
	Ubah status petugas	*Black Box*
	Tambah data petugas	*Black Box*
Laporan	Menampilkan *form* cari laporan produk,	*Black Box*

	penjualan, pengiriman, retur barang	
	Menampilkan cetak laporan, download pdf	*Black Box*

Berikut dapat dilihat pengujian dari Skenario pengujian aplikasi *Operator* pada tabel dibawah ini:

Tabel 4.7 Skenario pengujian aplikasi *Operator*

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Verivikasi *password*	*Black Box*
Logout	Menghapus *session user*	*Black Box*
Profil Anda	Menampilkan *form* profil petugas	*Black Box*
Edit Profil	Menampilkan *form* edit profil	*Black Box*
	Ubah data petugas	*Black Box*
	Simpan perubahan	*Black Box*
Pengolahan Data Pemesan	Tampilkan data pemesan	*Black Box*
	Cari data pemesan	*Black Box*
	Ubah status pemesan	*Black Box*
Pengolahan Data Kategori	Tampilkan data kategori	*Black Box*
	Cari data kategori	*Black Box*
	Tambah data kategori	*Black Box*
	Ubah data kategori	*Black Box*
	Hapus kategori	*Black Box*
Pengolahan Data barang	Tampilkan data barang	*Black Box*
	Cari data barang	*Black Box*
	Tambah data barang	*Black Box*
	Ubah data barang	*Black Box*
	hapus barang	*Black Box*
Pengolahan Data Kota	Tampilkan data kota	*Black Box*
	Cari data kota	*Black Box*
	Tambah data kota	*Black Box*

	Ubah data kota	*Black Box*
	Hapus data kota	*Black Box*
Pengolahan Jasa pengiriman	Tampilkan data jasa pengiriman	*Black Box*
	Cari data jasa pengiriman	*Black Box*
	Tambah data jasa pengiriman	*Black Box*
	Ubah data jasa pengiriman	*Black Box*
	Hapus data jasa pengiriman	*Black Box*
Pengolahan Data ongkos kirim	Tampilkan data ongkos kirim	*Black Box*
	Cari data ongkos kirim	*Black Box*
	Tambah data ongkos kirim	*Black Box*
	Ubah data ongkos kirim	*Black Box*
	Hapus data ongkos kirim	*Black Box*
Data pesanan	Tampilkan data pesanan, retur	*Black Box*
	Ubah status data pesanan, data retur	*Black Box*
	Tampilkan data detail transaksi	*Black Box*
	Tampilkan detail retur	*Black Box*
Pengolahan Komentar	Tampilkan data komentar	*Black Box*
	Hapus komentar	*Black Box*
Backup database	Menampilkan *form* backup database	*Black Box*
	Melakukan backup database	*Black Box*
Restore database	Menampilkan *form* restore database	*Black Box*
	Mengubah database lama dengan database baru	*Black Box*
Data laporan produk	Tampilkan form cari laporan produk	*Black Box*
	Tampilkan data laporan produk	*Black Box*
Data laporan pesanan	Tampilkan form cari laporan pesanan	*Black Box*
	Tampilkan data laporan pesanan	*Black Box*
Data laporan pengiriman	Tampilkan form cari laporan pengiriman	*Black Box*

	Tampilkan data laporan pengiriman	*Black Box*
Data laporan retur	Tampilkan form cari laporan retur	*Black Box*
	Tampilkan data laporan retur	*Black Box*

4.4.2 Kasus dan Hasil Pengujian

4.4.2.1 Pengujian Pendaftaran

Berikut dapat dilihat pengujian dari pendaftaran pada tabel dibawah ini:

Tabel 4.8 Pengujian Pendaftaran

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data pendaftaran, contoh masukan : Nama lengkap : rickar dsiadari Alamat : jl. Sukasari II No.272 Provinsi : Jawa Barat Kota : Bandung Kode Pos : 40134 Email : rajanihuta@gmail.com Password : xxxxxxx Ulangi Password : xxxxxxx	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan "Ini dibutuhkan" Jika data masukan benar muncul pesan "Pendaftaran Berhasil. Silahkan cek email anda untuk mengaktifkan account anda".	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian. Dan dapat menampilkan pesan "Pendaftaran Berhasil. Silahkan cek email anda untuk mengaktifkan account anda". jika data masukan benar dan dapat memunculkan pesan "Ini dibutuhkan" jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak

4.4.2.2 Pengujian Login Pemesan

Berikut dapat dilihat pengujian dari login pemesan pada tabel dibawah ini:

Tabel 4.9 Pengujian Login Pemesan

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan alamat *email* : rajanihuta@gmail.com	Mengisikan data *username* pada *field username*. Jika salah maka akan muncul pesan "Ini dibutuhkan, Email tidak valid"	Dapat mengisi *field* alamat *email* dan muncul pesan "Ini dibutuhkan, Email tidak valid" jika salah dalam pengisian data	[√] Diterima [ ] Ditolak
Memasukan *Password* : xxxxxxx	Mengisikan data *password* pada *field password*. Jika salah maka akan muncul pesan "Ini Dibutuhkan"	Dapat mengisi *field password* dan muncul pesan "Ini Dibutuhkan"	[√] Diterima [ ] Ditolak

4.4.2.3 Pengujian Ubah Profil Pemesan

Berikut dapat dilihat pengujian dari ubah profil pemesan pada tabel dibawah ini:

Tabel 4.10 Pengujian Ubah Profil Pemesan

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Merubah kode pos: Nama lengkap : rickar dsiadari Alamat : jl. Sukasari II No.272 Provinsi : Jawa Barat Kota : Bandung Kode Pos : 40134 TLP : 2531667 HP : 081375540205 FAX :2531222 Email : rajanihuta@gmail.com Password : xxxxxxx	Dapat merubah data "Profil dan Alamat telah dirubah"	Dapat merubah data dengan benar. Tampil pesan "Profil dan Alamat telah dirubah"	[√] Diterima [ ] Ditolak

Ulangi Password : xxxxxxx			

4.4.2.4 Pengujian Lupa *Password* Pemesan

Berikut dapat dilihat pengujian dari lupa *password* pemesan pada tabel dibawah ini:

Tabel 4.11 Pengujian Lupa *Password* Pemesan

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan *email*: rajanihuta@gmail.com	Mengirim *password* baru ke email dan menyimpan di *database.* Jika mengosongkan salah satu atau semua *field* muncul pesan kesalahan “Ini dibutuhkan Email tidak valid” dan jika memasukan penanda spam yang salah muncul pesan "Ini dibutuhkan Format bilangan integer"	Dapat mengirim data *password* yang baru dan menyimpan ke *database* dengan muncul pesan “Silahkan periksa *email* anda untuk mengetahui *password* baru anda.”. Dapat menampilkan pesan kesalahan “Ini dibutuhkan Email tidak valid” bila ada field yang belum diisi. Dapat menampilkan pesan kesalahan “Ini dibutuhkan Email tidak valid” "Ini dibutuhkan Format bilangan integer"	[√] Diterima [ ] Ditolak

4.4.2.5 Pengujian Menu Keranjang Belanja

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

Tabel 4.12 Pengujian Menu Keranjang Belanja

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Merubah data jumlah barang yang dibeli	Tercantum pada text box data yang akan diubah. Data jumlah yang diubah tersimpan di *database*	Dapat mengisi data jumlah dengan benar. Dapat menampilkan perubahan data jumlah barang.	[√] Diterima [ ] Ditolak

4.4.2.6 Pengujian *Form* Pengiriman

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

Tabel 4.13 Pengujian *Form* Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data pengiriman : Nama lengkap : rickar dsiadari Alamat : jl. Sukasari II No.272 Provinsi : Jawa Barat Kota : Bandung Kode Pos : 40134 Email : rajanihuta@gmail.com	Data pengiriman baru akan disimpan di dalam *database* dan jika mengosongkan salah satu *field* pesan kesalahan “Ini dibutuhkan”	Dapat menyimpan data pengiriman. Dapat menampilkan pesan kesalahan “Ini dibutuhkan”	[√] Diterima [ ] Ditolak

4.4.2.7 Pengujian *form* Pembayaran

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

Tabel 4.14 Pengujian *form* Pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih cara pembayaran: Paypal	Jika cara pembayaran paypal maka akan membuka halaman paypal dan menampilkan jumlah yang harus dibayar. Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan "Ini dibutuhkan". Data konfirmasi pembayaran akan disimpan di dalam *database* dan menampilkan pesan "Terimakasih telah melakukan pembayaran, pesanan anda akan kami proses"	Dapat membuka halaman paypal dan menampilkan jumlah pembayaran. Dapat memunculkan pesan kesalahan "Ini dibutuhkan" jika salah satu field tidak terisi atau salah dalam pengisian. Dapat menyimpan data pembayaran ke dalam *database* dan menampilkan pesan "Terimakasih telah melakukan pembayaran, pesanan anda akan kami proses"	[√] Diterima [ ] Ditolak

4.4.2.8 Pengujian *form* Retur

Berikut dapat dilihat pengujian dari pengujian form retur pada tabel dibawah ini:

Tabel 4.15 Pengujian *form* Retur

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data retur: No pesanan 4001 Nama barang: GR 21-011 Alasan: Kerusakan Fungsional	Menyimpan data retur ke dalam *database*	Dapat menyimpan data retur ke dalam *database*	[√] Diterima [ ] Ditolak

4.4.2.9 Pengujian Login Administrator

Berikut dapat dilihat pengujian login administrator pada tabel dibawah ini:

Tabel 4.16 Pengujian Login Administrator

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
username : admin *Password* : xxxxxxx	Berhasil login. Jika memasukan data *username* atau *password* yang tidak terdaftar Muncul pesan "ERROR:Masukkan username dan password anda" "Ini dibutuhkan".	Berhasil login. Tidak dapat login dan Muncul pesan "ERROR:Masukkan username dan password anda" "Ini dibutuhkan".	[√] Diterima [ ] Ditolak

4.4.2.10 Pengujian ubah profil administrator

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

Tabel 4.17 Pengujian ubah profil administrator

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Merubah alamat: username : admin password : xxxxxxx ulang password : administrator nama lengkap : petugas jenis kelamin: operator email : operator@grutty.com	Data berhasil diubah di *database.* Jika Mengosongkan salah satu *field* Menampilkan pesan kesalahan "Ini dibutuhkan".	Tampil pesan "Data berhasil disimpan". Dapat menampilkan pesan kesalahan "Ini dibutuhkan".	[√] Diterima [ ] Ditolak

4.4.2.11 Pengujian Tambah Petugas

Berikut dapat dilihat pengujian dari tambah petugas pada tabel dibawah ini:

Tabel 4.18 Pengujian Tambah Petugas

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi tambah Petugas, Contoh: nama lengkap : operator email : operator@grutty.com Username : operator Password : xxxxxxx Ulangi Password : xxxxxxx Aktifkan : Ya / Tidak	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan "Ini dibutuhkan". Data petugas akan tersimpan didalam *database.*	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian. Data petugas	[√] Diterima [ ] Ditolak

		akan tersimpan didalam *database* dan muncul pesan ” Petugas baru telah disimpan”	

4.4.2.12 Pengujian Data Laporan Produk

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

Tabel 4.19 Pengujian Data Laporan Produk

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan tanggal 1 juni 2011	Menampilkan data produk berdasarkan tanggal yang di inputkan. Jika Memasukan tanggal yang salah Tampilkan pesan kesalahan “data tidak ditemukan”	Dapat menampilkan data produk berdasarkan tanggal yang diinputkan. Dapat menampilkan pesan kesalahan ”data tidak ditemukan”	[√] Diterima [ ] Ditolak

4.4.2.13 Pengujian Data Laporan Penjualan

Berikut dapat dilihat pengujian dari data laporan penjualan pada tabel dibawah ini:

Tabel 4.20 Pengujian Data Laporan Penjualan

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan tanggal 1	Menampilkan	Dapat	[√] Diterima

Juni 2011	data penjualan berdasarkan tanggal yang di inputkan. Jika Memasukan tanggal yang salah Tampilkan pesan kesalahan “data tidak ditemukan”	menampilkan data penjualan berdasarkan tanggal yang diinputkan. Dapat menampilkan pesan kesalahan ”data tidak ditemukan”	[ ] Ditolak

4.4.2.14 Pengujian Data Laporan Pengiriman

Berikut dapat dilihat pengujian dari data laporan pengiriman pada tabel dibawah ini:

Tabel 4.21 Pengujian Data Laporan Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan tanggal 1 februari 2011	Menampilkan data pengiriman berdasarkan tanggal yang di inputkan. Jika Memasukan tanggal yang salah Tampilkan pesan kesalahan “data tidak ditemukan”	Dapat menampilkan data pengiriman berdasarkan tanggal yang diinputkan. Dapat menampilkan pesan kesalahan ”data tidak ditemukan”	[√] Diterima [ ] Ditolak

4.4.2.15 Pengujian Data Laporan Retur

Berikut dapat dilihat pengujian dari data laporan retur pada tabel dibawah ini:

Tabel 4.22 Pengujian Data Laporan Retur

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan tanggal 1 Juni 2011	Menampilkan data retur berdasarkan tanggal yang di inputkan. Jika Memasukan tanggal yang salah Tampilkan pesan kesalahan "data tidak ditemukan"	Dapat menampilkan data retur berdasarkan tanggal yang diinputkan. Dapat menampilkan pesan kesalahan "data tidak ditemukan"	[√] Diterima [ ] Ditolak

4.4.2.16 Pengujian *Login* Operator

Berikut dapat dilihat pengujian dari data laporan retur pada tabel dibawah ini:

Tabel 4.23 Pengujian *Login* Operator

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
username : operator *Password* : xxxxxxx	Login berhasil. Jika Memasukan data *username* atau *password* yang tidak terdaftar Muncul pesan "ERROR:Masukkan username dan password anda" "Ini dibutuhkan"	Tombol login dapat berfungsi. Dapat menampilkan pesan kesalahan "ERROR:Masukkan username dan password anda" "Ini dibutuhkan"	[√] Diterima [ ] Ditolak

4.4.2.17 Pengujian Ubah Profil Operator

Berikut dapat dilihat pengujian dari ubah profil petugas pada tabel dibawah ini :

Tabel 4.24 Pengujian Ubah Profil Operator

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Merubah nama lengkap: nama lengkap : operator email : operator@grutty.com	Data berhasil diubah di *database.* Jika Mengosongkan salah satu *field* Menampilkan pesan kesalahan "Ini dibutuhkan".	Tampil pesan "Data berhasil diubah". Dapat menampilkan pesan kesalahan "Ini dibutuhkan".	[√] Diterima [ ] Ditolak

4.4.2.18 Pengujian Pengolahan Data Pemesan

Berikut dapat dilihat pengujian dari pengolahan data pemesan pada tabel dibawah ini:

Tabel 4.25 Pengujian Pengolahan Data Pemesan

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Merubah status pemesan : Status : Aktif / Tidak	Merubah status pemesan menjadi Aktif / Tidak	Dapat merubah status pemesan menjadi Aktif / Tidak	[√] Diterima [ ] Ditolak

4.4.2.19 Pengujian Pengolahan Data Kategori

Berikut dapat dilihat pengujian dari pengolahan data kategori pada tabel dibawah ini:

Tabel 4.26 Pengujian Pengolahan Data Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data kategori : Nama Kategori : Sepatu Casual Gambar : Sepatu Casual.jpg Deskripsi : Sangat menarik	Data berhasil diubah di *database.* Jika tidak mengisi *field* kategori muncul pesan "Ini dibutuhkan".	Tampil pesan "Data berhasil disimpan". Dapat muncul pesan kesalahan "Ini dibutuhkan".	[√] Diterima [ ] Ditolak

4.4.2.20 Pengujian Tambah Kategori

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

Tabel 4.27 Pengujian Tambah Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data kategori Contoh: Nama Kategori : Sepatu Casual Gambar : Sepatu Casual.jpg Deskripsi : Sangat menarik	Data kategori akan tersimpan didalam *database*. Jika field tidak terisi maka akan muncul pesan kesalahan "Ini dibutuhkan"	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi. Data kategori tersimpan didalam *database* dan muncul pesan " Kategori berhasil disimpan"	[√] Diterima [ ] Ditolak

4.4.2.21 Pengujian Pengolahan Data Barang

Berikut dapat dilihat pengujian dari pengolahan data barang pada tabel dibawah ini:

Tabel 4.28 Pengujian Pengolahan Data Barang

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data : Gambar : Sepatu Casual.jpg Kategori : Formal Induk : - Kode : GR 21-011 Warna : Coklat+hitam Ukuran : 32-44 Berat : 1 Kg Stok : 25 Harga : 345000 Diskon : 25 % Deskripsi : Kualitas Terbaik Publikasikan : Ya / Tidak	Data berhasil diubah di *database.* Jika tidak mengisi salah satu *field* muncul pesan kesalahan "Ini dibutuhkan" "Format bilangan integer"	Tampil pesan "Data Barang berhasil disimpan.". Dapat muncul pesan kesalahan "Ini dibutuhkan" "Format bilangan integer"	[√] Diterima [ ] Ditolak

4.4.2.22 Pengujian Tambah Barang

Berikut dapat dilihat pengujian dari tambah barang pada tabel dibawah ini :

Tabel 4.29 Pengujian Tambah Barang

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data barang Contoh: Gambar : Kategori : Formal Induk : - Kode : GR 21-011 Warna : Coklat+hitam Ukuran : 32-44 Berat : 1 Kg Stok : 25 Harga : 345000 Diskon : 25 % Deskripsi : Kualitas Terbaik Publikasikan : Ya / Tidak	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan "Ini dibutuhkan" "Format bilangan integer". Data barang akan tersimpan	Dapat memunculkan pesan kesalahan "Ini dibutuhkan". Data barang akan tersimpan didalam *database* dan muncul pesan " Barang berhasil disimpan"	[√] Diterima [ ] Ditolak

	didalam *database*		

4.4.3 Kesimpulan Pengujian Alpha

Berdasarkan hasil pengujian alpha (fungsional) dengan kasus uji diatas dapat ditarik kesimpulan bahwa pembangunan aplikasi *e-commerce* pada Grutty Shoes & Bags bebas dari kesalahan penulisan dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

4.4.4 Pengujian *Betha*

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana pengujian dengan membuat kuisioner mengenai penilaian pengguna mengenai aplikasi yang dibangun untuk selanjutnya ditanyakan langsung kepada responden. Hasil pengujian *betha* disertakan dalam lampiran.

Berdasarkan hasil pengujian *betha*, dicari persentase masing-masing jawaban dengan menggunakan rumus:

Y=P/Q*100%

Keterangan:

P = banyaknya jawaban responden tiap soal

Q = jumlah responden

Y = nilai prosentase

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 10 orang.

1. Apakah anda setuju bahwa tampilan *website* cukup menarik dan mudah dipahami *user friendly*?

Tabel 4.30 Hasil Pengujian Kuesioner Soal Nomor 1

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	7	70
	2	Cukup Setuju	2	20
	3	Kurang Setuju	0	0
	4	Biasa Saja	1	10
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase tabel diatas maka dapat disimpulkan sebanyak 7 orang atau 70% menyatakan sangat setuju, 2 orang atau 20% menyatakan cukup setuju, dan 1 orang atau 10% menyatakan biasa saja

2. Apakah anda setuju bahwa informasi produk yang diberikan oleh *website* Grutty Shoes & Bags ini sesuai dengan informasi yang anda butuhkan?

Tabel 4.31 Hasil Pengujian Kuesioner Soal Nomor 2

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	7	70
	2	Cukup Setuju	3	30
	3	Kurang Setuju	0	0
	4	Biasa Saja	0	0
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase tabel diatas maka dapat disimpulkan sebanyak 7 orang atau 70 % menyatakan sangat setuju, 3 orang atau 30%

menyatakan sukup setuju bahwa informasi yang diberikan sesuai dengan informasi yang dibutuhkan *costumer*.

3. Apakah anda setuju bahwa *website* Grutty Shoes & Bags akan lebih memudahkan pelanggan dalam melakukan transaksi pemesanan dan pembelian produk?

Tabel 4.32 Hasil Pengujian Kuesioner Soal Nomor 3

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	2	20
	2	Cukup Setuju	7	70
	3	Kurang Setuju	0	0
	4	Biasa Saja	1	10
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 orang atau 20 % menyatakan sangat setuju, 7 atau 70 % menyatakan cukup setuju, dan 1 orang atau 10% menyatakan Biasa saja, bahwa *website* Grutty Shoes & Bags ini akan lebih memudahkan pelanggan dalam melakukan transaksi pembelian barang.

4. Apakah anda setuju bahwa dengan adanya *website* ini pihak Grutty Shoes & Bags akan lebih mudah mengelola data *master* dan data transaksi pemesanannya?

Tabel 4.33 Hasil Pengujian Kuesioner Soal Nomor 4

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)

4	1	Sangat Setuju	5	50
	2	Cukup Setuju	5	50
	3	Kurang Setuju	0	0
	4	Biasa Saja	0	0
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase tablel diatas maka dapat disimpulkan sebanyak 5 orang atau 50 % menyatakan sangat setuju, 5 orang atau 50 % menyatakan cukup setuju bahwa dengan adanya *website* ini pihak Grutty Shoes & Bags akan lebih mudah mengelola data master dan data transaksi penjualannya.

5. Apakah anda setuju bahwa sistem pembayaran yang digunakan *website* Grutty Shoes & Bags akan lebih memudahkan anda?

Tabel 4.34 Hasil Pengujian Kuesioner Soal Nomor 5

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
5	1	Sangat Setuju	6	60
	2	Cukup Setuju	3	30
	3	Kurang Setuju	0	0
	4	Biasa Saja	1	10
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 6 orang atau 60 % menyatakan sangat setuju, 3 orang atau 30% menyatakan cukup setuju, dan, 1 orang atau 10% menyatakan biasa saja bahwa sistem pembayaran yang digunakan akan lebih memudahkan.

6. Apakah anda setuju dengan adanya *website* ini Grutty Shoes & Bags dapat memperluas pemasarannya?

Tabel 4.35 Hasil Pengujian Kuesioner Soal Nomor 6

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
6	1	Sangat Setuju	8	80
	2	Cukup Setuju	2	20
	3	Kurang Setuju	0	0
	4	Biasa Saja	0	0
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 8 orang atau 80% menyatakan sangat setuju dan 2 orang atau 20% menyatakan cukup setuju bahwa *website* ini akan memperluas pemasaran Grutty Shoes & Bags.

7. Apakah anda setuju bahwa *website* Grutty Shoes & Bags ini telah layak untuk di-*online*-kan?

Tabel 4.36 Hasil Pengujian Kuesioner Soal Nomor 7

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
7	1	Sangat Setuju	6	60
	2	Cukup Setuju	4	40
	3	Kurang Setuju	0	0
	4	Biasa Saja	0	0
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 6 orang atau 60% menyatakan sangat setuju, dan 4 orang atau 40% menyatakan cukup setuju bahwa *website* ini sudah layak untuk di-*online*-kan.

4.1.1 Kesimpulan Pengujian Beta

Berdasarkan pengujian beta diatas, dapat diambil kesimpulan bahwa :

1. Aplikasi ini memudahkan pelanggan dalam mendapatkan informasi produk yang tersedia pada Grutty Shoes & Bags.
2. Proses pembayaran yang digunakan cukup mudah digunakan bagi para pelanggan.
3. Aplikasi ini juga memudahkan Grutty Shoes & Bags dalam mendokumentasikan data master dan data transaksi penjualan barangnya.
4. Aplikasi ini juga dapat memperluas jaringan pemasaran Grutty Shoes & Bags.