BAB 4

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan implementasi dan pengujian terhadap sistem yang baru. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemograman yang akan digunakan. Setelah implementasi maka dilakukan pengujian sistem yang baru dimana akan dilihat kekurangan - kekurangan pada aplikasi yang baru untuk pengembangan sistem selanjutnya.

4.1 Implementasi

Setelah sistem dianalisis dan didesain secara rinci, maka selanjutnya akan menuju tahap implementasi. Tujuan Implementasi adalah untuk mengkonfirmasikan modul program perancangan pada para pelaku sistem sehingga pengguna dapat *member*i masukan kepada pembangun sistem.

4.1.1 Implementasi Perangkat Keras

Perangkat keras yang diperlukan untuk membangun Aplikasi *e-commerce* di toko Ramdhan Distro adalah sebagai berikut :

Tabel 4. 1 Spesifikasi Perangkat Keras

No	Perangkat Keras	Spesifikasi
1	*Processor*	2.0 GHz
2	VGA	VGA *On-Board* 64MB
3	*Memory*/RAM	DDR2 1GB
4	Harddisk	160GB
5	*Optical Drive*	DVD-R 16x
6	*Keyboard*	*Keyboard Standard*
7	*Mouse*	*Mouse Optic Standard*
8	Printer	Printer
9	Monitor	19” LCD Monitor
10	Modem	*Wired Internet*

4.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang digunakan untuk mengembangkan Aplikasi *e-commerce* di toko Ramdhan Distro adalah sebagai berikut:

Tabel 4. 2 Spesifikasi Perangkat Lunak

Perangkat Lunak	Keterangan
Sistem Operasi	Windows XP
PHP	Bahasa Pemograman
WAMP	*Web Server*
MySQL	*Database Server*
Mozilla firefox, Chrome, Internet Explorer	*Web Browser*
Adobe dreamweaver CS4	*Code Editor*

4.1.3 Implimentasi Basis Data

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

1. Tabel Petugas

```
CREATE TABLE IF NOT EXISTS `petugas` (
  `username` varchar(25) NOT NULL,
  `password` varchar(32) NOT NULL,
  `email` varchar(100) DEFAULT NULL,
  `nama` varchar(150) NOT NULL,
  `no_telp` varchar(15) DEFAULT NULL,
  `level` enum('admin','petugas','gudang') NOT NULL DEFAULT
'petugas',
  PRIMARY KEY (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. Tabel Ongkos Kirim

```
CREATE TABLE IF NOT EXISTS `ongkos_kirim` (
  `id_kota` int(11) NOT NULL,
  `id_jenis_paket` int(11) NOT NULL,
  `ongkos` decimal(10,0) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `ongkos_kirim`
  ADD CONSTRAINT FOREIGN KEY (id_kota) REFERENCES kota (id_kota),
  ADD CONSTRAINT FOREIGN KEY (id_jenis_paket) REFERENCES
jenis_paket (id_jenis_paket);
```

3. Tabel Provinsi

```
CREATE TABLE IF NOT EXISTS `provinsi` (
  `id_provinsi` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(30) NOT NULL,
  `dihapus` varchar(1) NOT NULL DEFAULT 'T',
  PRIMARY KEY (`id_provinsi`),
  UNIQUE KEY `nama` (`nama`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=35 ;
```

4. Tabel Kota

```
CREATE TABLE IF NOT EXISTS `kota` (
  `id_kota` int(11) NOT NULL AUTO_INCREMENT,
  `id_provinsi` int(11) NOT NULL,
  `nama` varchar(30) NOT NULL,
  `dihapus` varchar(1) NOT NULL DEFAULT 'T',
  PRIMARY KEY (`id_kota`),
  UNIQUE KEY `nama` (`nama`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=98 ;
ALTER TABLE `kota`
  ADD CONSTRAINT FOREIGN KEY (id_provinsi) REFERENCES provinsi
(id_provinsi),
```

5. Tabel Kategori

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(30) NOT NULL,
  `dihapus` char(1) NOT NULL DEFAULT 'T',
  PRIMARY KEY (`id_kategori`),
  UNIQUE KEY `nama` (`nama`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=9 ;
```

6. Tabel Barang

```
CREATE TABLE IF NOT EXISTS barang (
  id_barang int(11) NOT NULL AUTO_INCREMENT,
  id_kategori int(11) NOT NULL,
  ukuran varchar(100) NOT NULL,
  nama varchar(100) NOT NULL,
  harga decimal(10,0) NOT NULL,
  berat float NOT NULL DEFAULT '0' COMMENT 'Dalam KG',
  diskon float NOT NULL,
  stok int(11) NOT NULL DEFAULT '0',
  tgl_release date NOT NULL,
  dijual char(1) NOT NULL DEFAULT 'Y',
  deskripsi text NOT NULL,
  filegambar varchar(100) DEFAULT NULL,
  dihapus char(1) NOT NULL DEFAULT 'T',
  PRIMARY KEY (id_barang),
  KEY id_kategori (id_kategori),
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;
ALTER TABLE `barang`
```

```
  ADD CONSTRAINT barang_ibfk_1 FOREIGN KEY (id_kategori)
REFERENCES kategori (id_kategori),
```

7. Tabel *Member*

```
CREATE TABLE IF NOT EXISTS `member` (
  `id_member` int(11) NOT NULL AUTO_INCREMENT,
  `email` varchar(100) NOT NULL,
  `password` varchar(32) NOT NULL,
  `nama` varchar(150) NOT NULL,
  `no_telp` varchar(15) NOT NULL,
  `id_kota` int(11) NOT NULL,
  `alamat` varchar(250) NOT NULL,
  `kode_pos` varchar(10) NOT NULL,
  `status` enum('Y','T') NOT NULL DEFAULT 'T',
  PRIMARY KEY (`id_member`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=36 ;
ALTER TABLE `member`
  ADD CONSTRAINT FOREIGN KEY (id_kota) REFERENCES kota (id_kota),
```

8. Tabel Pembelian

```
CREATE TABLE IF NOT EXISTS `pembelian` (
  `id_pembelian` int(11) NOT NULL AUTO_INCREMENT,
  `id_jenis_paket` int(11) NOT NULL,
  `tgl_beli` date NOT NULL,
  `tgl_bayar` date DEFAULT NULL,
  `tgl_sampai` date DEFAULT NULL,
  `status_bayar` varchar(1) NOT NULL DEFAULT 'T',
  `status_kirim` varchar(1) NOT NULL DEFAULT 'T',
  `nama` varchar(100) NOT NULL,
  `alamat` varchar(250) NOT NULL,
  `username` varchar(25) DEFAULT NULL,
  `kode_pos` varchar(10) NOT NULL,
  `no_telp` varchar(15) NOT NULL,
  `email` varchar(100) NOT NULL,
  `id_member` int(11) NOT NULL,
  `jenis_bayar` varchar(20) NOT NULL,
  `no_resi` varchar(50) DEFAULT NULL,
  `ongkos` decimal(10,0) NOT NULL,
  `mata_uang` varchar(5) NOT NULL DEFAULT 'IDR',
  `nilai_mata_uang` decimal(10,2) NOT NULL DEFAULT '1.00',
  `paypal_id` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`id_pembelian`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=30 ;
ALTER TABLE `pembelian`
  ADD CONSTRAINT FOREIGN KEY (username) REFERENCES petugas
(username);
  ADD CONSTRAINT FOREIGN KEY (id_member) REFERENCES member
(id_member);  `mata_uang` varchar(5) NOT NULL DEFAULT 'IDR',
  `nilai_mata_uang` decimal(10,2) NOT NULL DEFAULT '1.00',
  PRIMARY KEY (`id_pembelian`)
ALTER TABLE `pembelian`
```

```
  ADD CONSTRAINT `id_pembelian` FOREIGN KEY (`id_jenis_paket`)
REFERENCES `jenis_paket` (`id_jenis_paket`),
  ADD CONSTRAINT `id_pembelian` FOREIGN KEY (`id_member`)
REFERENCES `member` (`id_member`),
  ADD CONSTRAINT `id_pembelian` FOREIGN KEY (`id_kota`)
REFERENCES `kota` (`id_kota`),
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

9. Tabel Detail Pembelian

```
CREATE TABLE IF NOT EXISTS `detail_pembelian` (
  `id_pembelian` int(11) NOT NULL,
  `id` int(11) NOT NULL,
  `tipe` varchar(6) NOT NULL COMMENT 'barang , paket',
  `jumlah` int(11) NOT NULL,
  `berat` float NOT NULL,
  `diskon` int(11) NOT NULL,
  `harga` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `detail_pambelian`
  ADD CONSTRAINT  FOREIGN KEY (id_pambelian) REFERENCES pembelian
(id_pembelian);
```

10. Tabel Komentar

```
CREATE TABLE IF NOT EXISTS `komentar` (
  `id_komentar` int(11) NOT NULL AUTO_INCREMENT,
  `id_barang` int(11) NOT NULL,
  `id_member` int(11) NOT NULL,
  `komentar` text NOT NULL,
  `dibaca` enum('Y','T') NOT NULL DEFAULT 'T',
  PRIMARY KEY (`id_komentar`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=12 ;
ALTER TABLE `komentar`
  ADD CONSTRAINT FOREIGN KEY (id_barang) REFERENCES barang
(id_barang),
  ADD CONSTRAINT FOREIGN KEY (id_member) REFERENCES member
(id_member);
```

11. Tabel Kurs

```
CREATE TABLE IF NOT EXISTS `kurs` (
  `id_kurs` int(11) NOT NULL AUTO_INCREMENT,
  `mata_uang` varchar(5) NOT NULL,
  `simbol` varchar(5) NOT NULL,
  `nilai_beli` decimal(10,2) NOT NULL DEFAULT '0.00',
  `nilai_jual` decimal(10,2) NOT NULL DEFAULT '0.00',
  `last_update` datetime NOT NULL,
  PRIMARY KEY (`id_kurs`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=15 ;
```

12. Tabel Rating

```
CREATE TABLE IF NOT EXISTS `rating` (
  `id_member` int(11) NOT NULL,
  `id_barang` int(11) NOT NULL,
  `nilai` tinyint(4) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `rating`
  ADD CONSTRAINT FOREIGN KEY (id_member) REFERENCES member
(id_member),
  ADD CONSTRAINT FOREIGN KEY (id_barang) REFERENCES barang
(id_barang);
```

13. Tabel Jasa Pengiriman

```
CREATE TABLE IF NOT EXISTS `jasa_pengiriman` (
  `id_jasa` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(20) NOT NULL,
  `dihapus` varchar(1) NOT NULL DEFAULT 'T',
  PRIMARY KEY (`id_jasa`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;;
```

14. Tabel Retur

```
CREATE TABLE IF NOT EXISTS `retur` (
  `id_retur` int(11) NOT NULL AUTO_INCREMENT,
  `id_pembelian` int(11) NOT NULL,
  `tgl_retur` date NOT NULL,
  `id_jasa_1` int(11) DEFAULT NULL,
  `id_jasa_2` int(11) DEFAULT NULL,
  `no_resi_1` varchar(50) DEFAULT NULL,
  `no_resi_2` varchar(50) DEFAULT NULL,
  `tgl_sampai_1` date DEFAULT NULL,
  `tgl_sampai_2` date DEFAULT NULL,
  `status` varchar(1) NOT NULL DEFAULT '0' COMMENT '0 = menunggu
konfirmasi petugas, 1 = menunggu user input no_resi, 2=menunggu
petugas input_no_resi, 3=retur berhasil',
  `catatan` text,
  PRIMARY KEY (`id_retur`),
  KEY `id_retur` (`id_retur`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;
ALTER TABLE `retur`
  ADD CONSTRAINT FOREIGN KEY (id_pembelian) REFERENCES pembelian
(id_pembelian);
```

15. Tabel Jenis Paket

```
CREATE TABLE IF NOT EXISTS `jenis_paket` (
  `id_jenis_paket` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(20) NOT NULL,
  `dihapus` varchar(1) NOT NULL DEFAULT 'T',
  `id_jasa` int(11) NOT NULL,
  PRIMARY KEY (`id_jenis_paket`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;
ALTER TABLE `jenis_paket`
  ADD CONSTRAINT FOREIGN KEY (id_jasa) REFERENCES jasa_pengiriman
(id_jasa);
```

16. Tabel Detail Retur

```
CREATE TABLE IF NOT EXISTS `detail_retur` (
  `id_retur` int(11) NOT NULL,
  `id` int(11) NOT NULL,
  `tipe` varchar(6) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `alasan` text NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `detail_retur`
  ADD CONSTRAINT FOREIGN KEY (id_retur) REFERENCES retur
(id_retur),
ADD CONSTRAINT FOREIGN KEY (id) REFERENCES detail_pembelian (id);
```

17. Tabel Konfirmasi_pembayaran

```
CREATE TABLE IF NOT EXISTS `konfirmasi_pembayaran` (
  `id_pembelian` int(11) NOT NULL,
  `transfer_ke` enum('BCA','MANDIRI') NOT NULL,
  `jumlah` decimal(10,0) NOT NULL,
  `tgl_bayar` date NOT NULL,
  `status` varchar(1) NOT NULL DEFAULT '0' COMMENT '0 = pending, 1
= diterima, 2 = ditolak',
  `catatan` text
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `konfirmasi_pembayaran`
  ADD CONSTRAINT  FOREIGN KEY (id_pembelian) REFERENCES pembelian
(id_pembelian);
```

4.1.4 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap halaman program yang dibuat dalam bentuk file program. Berikut ini adalah implementasi antarmuka yang dibuat dan dibedakan antara antarmuka untuk petugas, admin, gudang, pengunjung dan *member*s.

4.1.4.1 Implementasi Antarmuka Petugas

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditunujukan untuk petugas dapat dilihat pada tabel 4.3 berikut ini :

Tabel 4. 3 Antarmuka Petugas

Menu	Deskripsi	Nama File
Login petugas	Digunakan sebagai halaman login petugas	login.php
Home	Digunakan untuk melihat *home* petugas	index.php
Pengolahan kategori	Digunakan untuk proses pengolahan data kategori	kategori.php
Pengolahan provinsi	Digunakan untuk proses pengolahan data provinsi	provinsi.php
Pengolahan kota	Digunakan untuk proses pengolahan data kota	kota.php
Pengolahan Jasa Pengiriman	Digunakan untuk proses pengolahan data jasa pengiriman	jasa_kirim.php
Pengolahan Jenis pengiriman	Digunakan untuk proses pengolahan data jenis pengiriman	jenis_kirim.php
Pengolahan Ongkos Kirim	Digunakan untuk proses pengolahan data ongkos kirim	ongkir.php
Pengolahan Pembelian	Digunakan untuk proses pengolahan data pembelian	pembelian.php
Pengolahan *Member*	Digunakan untuk proses pengolahan data pembelian	*member*.php

4.1.4.2 Implementasi Antarmuka *Member*

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditunujukan untuk *member* dapat dilihat pada tabel 4.4 berikut ini :

Tabel 4. 4 Antarmuka *Member*

Menu	Deskripsi	Nama File
Login *member*	Digunakan sebagai halaman login *member*	login.php
Daftar	Untuk mendaftar sebagai *member*	Daftar.php
Home	Digunakan untuk melihat *home member*	index.php
Kategori	Digunakan untuk menampilkan semua kategori	kategori.php
Kota	Digunakan untuk menampilkan semua kota	kota.php
Keranjang Belanja	Digunakan untuk menampilkan semua daftar kerjang belanja	keranjang.php
Histori Pembelian	Digunakan untuk menampilkan semua daftar histori pembelian	Transaksi.php
Pencarian	Digunakan untuk menampilkan semua pancarian	search.php
Retur	Digunakan untuk mengolah dan menampilkan data retur	retur.php
Member	Digunakan untuk menampilkan semua data *member*	*member*.php
konfirmasi	Digunakan untuk proses konfirmasi pembelian	konfirmasi.php
Invoice	Digunakan untuk proses menampilkan invoice pembelian	invoice.php
Detail_pembelian	Digunakan untuk proses menapilkan detail pembelian	Detail_transaksi.php
Cara membeli	Digunakan untuk proses menampilkan cara belanja	Carabenlanja.php

Menu	Deskripsi	Nama File
Tentang Kami	Digunakan untuk menampilkan tentang kami	AboutUs.php
Checkout	Digunakan untuk proses checkout pembelian	Checkout.php

4.1.4.3 Implementasi Antarmuka Pengunjung

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditunujkan untuk pengunjung dapat dilihat pada tabel 4.5 berikut ini :

Tabel 4. 5 Antarmuka Pengunjung

Menu	Deskripsi	Nama File
Daftar	Untuk mendaftar sebagai *member*	Daftar.php
Home	Digunakan untuk melihat *home member*	index.php
Kategori	Digunakan untuk menampilkan semua kategori	kategori.php
Kota	Digunakan untuk menampilkan semua kota	kota.php
Keranjang Belanja	Digunakan untuk menampilkan semua daftar kerjang belanja	keranjang.php
Pencarian	Digunakan untuk menampilkan semua pancarian	search.php
Tentang Kami	Digunakan untuk menampilkan tentang kami	AboutUs.php
Cara membeli	Digunakan untuk proses menampilkan cara belanja	Carabenlanja.php

4.1.4.4 Implementasi Antarmuka Admin

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditunujkan untuk admin dapat dilihat pada tabel 4.6 berikut ini :

Tabel 4. 6 Antarmuka Admin

Menu	Deskripsi	Nama File
Login admin	Digunakan sebagai halaman login petugas	login.php
Home	Digunakan untuk melihat *home* petugas	index.php
Pengolahan petugas	Digunakan untuk proses pengolahan data petugas	petugas.php
Pengolahan backup	Digunakan untuk proses pengolahan data backup	backup.php
Pengolahan restore	Digunakan untuk proses pengolahan data restore	restore.php

4.1.4.5 Implementasi Antarmuka Gudang

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditunujkan untuk gudang dapat dilihat pada tabel 4.7 berikut ini :

Tabel 4. 7 Antarmuka Gudang

Menu	Deskripsi	Nama File
Login gudang	Digunakan sebagai halaman login petugas	login.php
Home	Digunakan untuk melihat *home* petugas	index.php
Pengolahan barang	Digunakan untuk proses pengolahan data stok kota	barang.php

4.2 Pengujian Perangkat Lunak

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

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

4.3 Pengujian Alpha

Pengujian yang dilakukan yaitu pengujian secara fungsional *(alpha)* dan *betha*. Metode yang digunakan dalam pengujian ini adalah pengujian *black box* yang berfokus pada persyaratan fungsional dari sistem yang dibangun.

4.3.1 Perancangan Pengujian Sistem

Rencana pengujian yang akan dilakukan dengan menguji sistem menggunakan metode *black box* untuk mengetahui apakah fungsi tersebut berfungsi sesuai yang diharapkan atau tidak. Berikut ini tabel 4.8 rencana pengujian dari sistem yang dibangun :

Tabel 4. 8 Rencana Pengujian Petugas

Item Uji	Detail Pengujian	Jenis Pengujian
Login petugas	Isi data *login*	*Black Box*
	Pengecekan u*sername* dan *password*	*Black Box*
Menu *home*	Menampilkan data *home*	*Black Box*
Kategori	Tampil Kategori	*Black Box*
	Tambah kategori	*Black Box*
	Ubah kategori	*Black Box*
	Hapus kategori	*Black Box*
Barang	Tampil barang	*Black Box*
	Ubah barang	*Black Box*
	Hapus barang	*Black Box*
	Tambah barang	*Black Box*
Provinsi	Tampil provinsi	*Black Box*
	Tambah provinsi	*Black Box*

Item Uji	Detail Pengujian	Jenis Pengujian
	Ubah provinsi	*Black Box*
	Hapus provinsi	*Black Box*
Kota	Tampil kota	*Black Box*
	Tambah kota	*Black Box*
	Ubah kota	*Black Box*
	Hapus kota	*Black Box*
Jasa Pengiriman	Tampil jasa pengiriman	*Black Box*
	Tambah jasa pengiriman	*Black Box*
	Ubah jasa pengiriman	*Black Box*
	Hapus jasa pengiriman	*Black Box*
Jenis Pengiriman	Tampil jenis pengiriman	*Black Box*
	Tambah jenis pengiriman	*Black Box*
	Ubah jenis pengiriman	*Black Box*
	Hapus jenis pengiriman	*Black Box*
Ongkos Kirim	Atur Ongkos Kirim	*Black Box*
Pembelian	Tampil data pembelian	*Black Box*
	Detail	*Black Box*
	Retur	*Black Box*
Member	Tampil data *member*	*Black Box*
	Ubah status *member*	*Black Box*
logout	Menghapus *session*	*Black Box*

Tabel 4. 9 Rencana Pengujian *Member*

Item Uji	Detail Pengujian	Jenis Pengujian
Login Member	Isi data *login*	*Black Box*
	Pengecekan u*sername* dan *password*	*Black Box*
Lupa Password	Pengecekan *email*	*Black Box*
Menu Beranda	Menampilkan data *home*	*Black Box*
Kategori	Menampilkan kota berdasarkan kategori	*Black Box*
Kota	Menampilkan kota kota	*Black Box*
Keranjang Belanja	Tampil Keranjang Belanja	*Black Box*
	Update jumlah kota yang berada di keranjang belanja	*Black Box*
	Hapus kota yang ada di keranjang	*Black Box*
	Checkout	*Black Box*
	Belanja Lagi	*Black Box*
Histori Pembelian	Menampilkan data histori pembelian	*Black Box*
Pencarian	Menampilkan hasil	*Black Box*

Item Uji	Detail Pengujian	Jenis Pengujian
	Pencarian	
Retur	Menampilkan Data Retur	*Black Box*
Jasa Pengiriman	Tampil jasa pengiriman	*Black Box*
	Tambah jasa pengiriman	*Black Box*
	Ubah jasa pengiriman	*Black Box*
	Hapus jasa pengiriman	*Black Box*
Jenis Pengiriman	Tampil jenis pengiriman	*Black Box*
	Tambah jenis pengiriman	*Black Box*
	Ubah jenis pengiriman	*Black Box*
	Hapus jenis pengiriman	*Black Box*
Ongkos Kirim	Atur Ongkos Kirim	*Black Box*
Pembelian	Tampil data pembelian	*Black Box*
	Detail	*Black Box*
	Retur	*Black Box*
Member	Tampil data *member*	*Black Box*
	Ubah status *member*	*Black Box*
Tentang Kami	Menampilkan tentang perusahaan	*Black Box*
Cara membeli	Menampilkan cara membeli	*Black Box*
logout	Menghapus *session*	*Black Box*

Tabel 4. 10 Rencana Pengujian Gudang

Item Uji	Detail Pengujian	Jenis Pengujian
Login	Isi data login	*Black Box*
	Pengecekan *Email* dan *password*	*Black Box*
Menu *home*	Menampilkan data *home*	*Black Box*
Pengolahan data stok	Pengolahan stok	*Black Box*
logout	Menghapus *session*	*Black Box*

Tabel 4. 11 Rencana Pengujian Admin

Item Uji	Detail Pengujian	Jenis Pengujian
Login	Isi data login	*Black Box*
	Pengecekan *Email* dan *password*	*Black Box*
Menu *home*	Menampilkan data *home*	*Black Box*
Backup	Pengolahan data backup	*Black Box*
Restore	Pengolahan data restore	*Black Box*
logout	Menghapus *session*	*Black Box*

4.3.2 Kasus dan Hasil Pengujian

Berdasarkan rencana pengujian yang disusun, maka dilakukan pengujian sebagai berikut :

4.3.2.1 Pengujian Login Petugas

Untuk penjelasan labih lanjut tentang sekenario pengujian login petugas dapat dilihat pada tabel 4.12 sebagai berikut :

Tabel 4. 12 Pengujian Login Petugas

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Petugas Contoh *Username* : petugas Password : petugas	Menampilkan menu utama petugas	Dapat menampilkan menu utama petugas	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Petugas Contoh *Username* : ‘kosong’ Password : ‘kosong’	Tidak dapat masuk ke halaman utama petugas	Tidak dapat masuk ke halaman utama petugas	[√] Diterima [ ] Ditolak

4.3.2.2 Pengujian Tambah Kategori

Untuk penjelasan lebih lanjut tentang sekenario tambah data kategori dapat dilihat pada tabel 4.13 sebagai berikut :

Tabel 4. 13 Pengujian Tambah Kategori

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Kategori Contoh Kategori : Sweater	Menampilkan pesan “data kategori telah disimpan”	Dapat Menampilkan pesan “data kategori telah disimpan”	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data katagori diisi dengan nama yang sudah ada (duplikasi) Contoh Kategori : Sweater	Menampilkan pesan “data kategori gagal disimpan”	Dapat menampilkan pesan “data kategori gagal disimpan”	[√] Diterima [ ] Ditolak

4.3.2.3 Pengujian Ubah Kategori

Untuk penjelasan lebih lanjut tentang sekenario ubah data kategori dapat dilihat pada tabel 4.14 sebagai berikut :

Tabel 4. 14 Pengujian Ubah Data Kategori

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategori Ubah dengan kategori baru Contoh Kategori baru : Baju	Menampilkan pesan "data kategori telah di *update"*	Dapat menampilkan pesan "data kategori telah di *update"*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama kategori sudah ada Contoh Data awal Nama Kategori : Kaos Data baru Nama Kategori : Kaos	Menampilkan pesan "data kategori gagal disimpan"	Dapat menampilkan pesan "data kategori gagal disimpan"	[√] Diterima [ ] Ditolak

4.3.2.4 Pengujian Tambah Barang

Untuk penjelasan lebih lanjut tentang sekenario tambah data barang dapat dilihat pada tabel 4.15 sebagai berikut :

Tabel 4. 15 Pengujian Tambah Barang

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Barang Contoh Nama : T-Shirt Gooners Ukuran : M Kategori : T-Shirt Harga : 50000 Berat :1 Diskon : 0 Stok : 5 Tanggal Release : 30/11/2012 Dijual : ya Gambar : gooners.jpg	Menampilkan pesan "data barang telah disimpan"	Dapat Menampilkan pesan "data barang telah disimpan"	[√] Diterima [ ] Ditolak

Deskripsi : T-Shrit dengan bertemakan gooners atau supporter sejati Arsenal FC.			
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Barang tidak diisi Contoh Nama : ‘kosong’ Ukuran : ‘kosong’ Kategori : ‘kosong’ Harga : ‘kosong’ Berat : ‘kosong’ Diskon : ‘kosong’ Stok : ‘kosong’ Tanggal Release : ‘kosong’ Dijual : ‘kosong’ Gambar : ‘kosong’ Deskripsi : ‘kosong’	Menampilkan pesan “nama barang harus diisi”	Dapat menampilkan pesan “nama barang harus diisi”	[√] Diterima [ ] Ditolak
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Barang sudah ada Contoh Nama : T-Shirt Gooners Ukuran : M Kategori : T-Shirt Harga : 50000 Berat :1 Diskon : 0 Stok : 5 Tanggal Release : 30/11/2012 Dijual : ya Gambar : gooners.jpg Deskripsi : T-Shrit dengan bertemakan gooners atau supporter sejati Arsenal FC.	Menampilkan pesan “data barang gagal disimpan”	Dapat menampilkan pesan “menampilkan data barang gagal disimpan”	[√] Diterima [ ] Ditolak

4.3.2.5 Pengujian Ubah Barang

Untuk penjelasan lebih lanjut tentang sekenario ubah data barang dapat dilihat pada tabel 4.16 sebagai berikut :

Tabel 4. 16 Pengujian Ubah Barang

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Barang Awal Contoh Nama : T-Shirt Gooners Ukuran : M Kategori : T-Shirt Harga : 50000 Berat :1 Diskon : 0 Stok : 5 Tanggal Release : 30/11/2012 Dijual : ya Gambar : gooners.jpg Deskripsi : T-Shrit dengan bertemakan gooners atau supporter sejati Arsenal FC. Data barang baru Contoh Data Barang Contoh Nama : SUJU Ukuran : All Size Kategori : Sweater Harga : 100000 Berat :1 Diskon : 0 Stok : 5 Tanggal Release : 21/12/2012 Dijual : ya Gambar : suju.jpg Deskripsi : Chace 8MB	Menampilkan pesan ”data barang telah di *update”*	Dapat menampilkan pesan ”data barang telah di *update”*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Barang Awal	Menampilkan	Dapat	[√] Diterima

Contoh Nama : T-Shirt Gooners Ukuran : M Kategori : T-Shirt Harga : 50000 Berat :1 Diskon : 0 Stok : 5 Tanggal Release : 30/11/2012 Dijual : ya Gambar : gooners.jpg Deskripsi : T-Shrit dengan bertemakan gooners atau supporter sejati Arsenal FC. Data barang baru Contoh Data Barang Contoh Nama : ‘kosong’ Ukuran : ‘kosong’ Kategori : ‘kosong’ Harga : ‘kosong’ Berat : ‘kosong’ Diskon : ‘kosong’ Stok : ‘kosong’ Tanggal Release : ‘kosong’ Dijual : ‘kosong’ Gambar : suju.jpg Deskripsi : ‘kosong’	pesan “data barang gagal disimpan”	menampilkan pesan “data barang gagal disimpan”	[ ] Ditolak
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Barang Awal Contoh Nama : T-Shirt Gooners Ukuran : M Kategori : T-Shirt Harga : 50000 Berat :1 Diskon : 0 Stok : 5 Tanggal Release : 30/11/2012 Dijual : ya	Menampilkan pesan “data barang gagal disimpan”	Dapat menampilkan pesan “data barang gagal disimpan”	[√] Diterima [ ] Ditolak

Gambar : gooners.jpg Deskripsi : T-Shrit dengan bertemakan gooners atau supporter sejati Arsenal FC. Data barang baru (Duplikasi) Contoh Nama : T-Shirt Gooners Ukuran : M Kategori : T-Shirt Harga : 50000 Berat :1 Diskon : 0 Stok : 5 Tanggal Release : 30/11/2012 Dijual : ya Gambar : gooners.jpg Deskripsi : T-Shrit dengan bertemakan gooners atau supporter sejati Arsenal FC.			

4.3.2.6 Pengujian Tambah Provinsi

Untuk penjelasan lebih lanjut tentang sekenario tambah data provinsi dapat dilihat pada tabel 4.17 sebagai berikut :

Tabel 4. 17 Pengujian Tambah Provinsi

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Provinsi Contoh Provinsi : Jawa Barat	Menampilkan pesan "data provinsi telah disimpan"	Dapat Menampilkan pesan "data provinsi telah disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan

Data Provinsi diisi dengan nama yang sudah ada (duplikasi) Contoh Provinsi : 'Jawa Barat'	Menampilkan pesan "data provinsi gagal disimpan"	Dapat menampilkan pesan "data provinsi gagal disimpan"	[√] Diterima [ ] Ditolak

4.3.2.7 Pengujian Ubah Provinsi

Untuk penjelasan lebih lanjut tentang sekenario ubah data provinsi dapat dilihat pada tabel 4.18 sebagai berikut :

Tabel 4. 18 Pengujian Ubah Provinsi

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data provinsi Ubah dengan provinsi baru Contoh Provinsi baru : Jawa Tengah	Menampilkan pesan "data provinsi telah di *update"*	Dapat Menampilkan pesan "data provinsi telah di *update"*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama provinsi sudah ada Contoh Data awal Nama provinsi : Jawa barat Data baru Nama provinsi : Jawa barat	Menampilkan pesan "data provinsi gagal disimpan"	Dapat Menampilkan pesan "data provinsi gagal disimpan"	[√] Diterima [ ] Ditolak

4.3.2.8 Pengujian Tambah Kota

Untuk penjelasan lebih lanjut tentang sekenario tambah data kota dapat dilihat pada tabel 4.19 sebagai berikut :

Tabel 4. 19 Pengujian Tambah Kota

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Kota Contoh Kota : Kabupaten Cirebon	Menampilkan pesan "data kota telah disimpan"	Dapat Menampilkan pesan "data kota telah disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Kota diisi dengan nama yang sudah ada (duplikasi) Contoh Kota : 'Kabupaten Cirebon'	Menampilkan pesan "data kota gagal disimpan"	Dapat menampilkan pesan "data kota gagal disimpan"	[√] Diterima [ ] Ditolak

4.3.2.9 Pengujian Ubah Kota

Untuk penjelasan lebih lanjut tentang sekenario ubah data kota dapat dilihat pada tabel 4.20 sebagai berikut :

Tabel 4. 17 Pengujian Ubah Kota

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kota Ubah dengan kota baru Contoh Kota baru : Bandung	Menampilkan pesan "data kota telah di *update"*	Dapat Menampilkan pesan "data kota telah di *update"*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama kota tidak diisi Contoh Data awal Nama kota : Bandung Data baru Nama kota : 'kosong'	Menampilkan pesan "data kota gagal disimpan"	Dapat Menampilkan pesan "data kota gagal disimpan"	[√] Diterima [ ] Ditolak

4.3.2.10 Pengujian Tambah Jasa Pengiriman

Untuk penjelasan lebih lanjut tentang sekenario tambah data jasa pengiriman dapat dilihat pada tabel 4.21 sebagai berikut :

Tabel 4. 18 Pengujian Tambah Jasa Pengiriman

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Jasa Pengiriman Contoh Jasa Pengiriman : JNE	Menampilkan pesan "data jasa pengiriman telah disimpan"	Dapat Menampilkan pesan "data jasa pengiriman telah disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Jasa Pengiriman diisi dengan nama yang sudah ada (duplikasi) Contoh Jasa Pengiriman : 'JNE'	Menampilkan pesan "data jasa pengiriman gagal disimpan"	Dapat menampilkan pesan "data jasa pengiriman gagal disimpan"	[√] Diterima [ ] Ditolak

4.3.2.11 Pengujian Ubah Jasa Pengiriman

Untuk penjelasan lebih lanjut tentang sekenario ubah data jasa pengiriman dapat dilihat pada tabel 4.22 sebagai berikut :

Tabel 4. 19 Pengujian Ubah Jasa Pengiriman

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data jasa pengiriman Ubah dengan jasa pengiriman baru Contoh Jasa pengiriman baru : TIKI	Menampilkan pesan "data jasa pengiriman telah di *update"*	Dapat Menampilkan pesan "data jasa pengiriman telah di *update"*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama jasa pengiriman diisi dengan nama yang sudah ada (duplikasi) Contoh Data awal	Menampilkan pesan "data jasa pengiriman gagal disimpan"	Dapat Menampilkan pesan "data jasa pengiriman gagal disimpan"	[√] Diterima [ ] Ditolak

Nama jasa pengiriman : JNE Data baru Nama jasa pengiriman : JNE			

4.3.2.12 Pengujian Tambah Jenis Pengiriman

Untuk penjelasan lebih lanjut tentang sekenario tambah data jenis pengiriman dapat dilihat pada tabel 4.23 sebagai berikut :

Tabel 4. 20 Pengujian Tambah Jenis Pengiriman

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Jenis Pengirman Contoh Jenis Pengiriman : REG	Menampilkan pesan "data jenis pengiriman telah disimpan"	Dapat Menampilkan pesan "data jenis pengiriman telah disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Jenis Pengiriman tidak diisi sudah ada Contoh Data awal Nama Jenis Pengiriman : REG Data baru Nama Jenis Pengiriman : 'kosong'	Menampilkan pesan "data jenis pengiriman gagal disimpan"	Dapat menampilkan pesan "data jenis pengiriman gagal disimpan"	[√] Diterima [ ] Ditolak

4.3.2.13 Pengujian Ubah Jenis Pengiriman

Untuk penjelasan lebih lanjut tentang sekenario ubah data jenis pengiriman dapat dilihat pada tabel 4.24 sebagai berikut :

Tabel 4. 21 Pengujian Ubah Jenis Pengirman

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data jenis pengiriman Ubah dengan jenis	Menampilkan pesan "data jenis	Dapat Menampilkan	[√] Diterima

pengiriman baru Contoh Jenis pengiriman baru : YES	pengiriman telah di *update"*	pesan "data jenis pengiriman telah di *update"*	[ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama jenis pengiriman tidak diisi Contoh Data awal Nama jenis pengiriman : YES Data baru Nama jenis pengiriman : 'kosong'	Menampilkan pesan "data jenis pengiriman gagal disimpan"	Dapat Menampilkan pesan "data jenis pengiriman gagal disimpan"	[√] Diterima [ ] Ditolak

4.3.2.14 Pengujian Konfirmasi Pembayaran

Untuk penjelasan lebih lanjut tentang sekenario konfimrasi pembayaran dapat dilihat pada tabel 4.25 sebagai berikut :

Tabel 4. 22 Pengujian Konfirmasi Pembayaran

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data pembelian Contoh Transfer ke bank : BCA Jumlah Pembayaran : 200 Tgl pembayaran : 21/12/2012 Catatan : Sudah dibayar	Menampilkan pesan "data anda sudah kami terima, silahkan tunggu 1x24 jam untuk di konfirmasi oleh petugas"	Dapat menampilkan pesan "data anda sudah kami terima, silahkan tunggu 1x24 jam untuk di konfirmasi oleh petugas"	[√] Diterima [ ] Ditolak
Pengisian nama ongkos kirim dengan data yang belum ada kemudian klik tombol *update*	Menampilkan pesan "jumlah belum diisi tanggal bayar belum diisi"	Dapat Menampilkan pesan "jumlah belum diisi tanggal bayar belum diisi"	[√] Diterima [ ] Ditolak

4.3.2.15 Pengujian Retur

Untuk penjelasan lebih lanjut tentang sekenario retur dapat dilihat pada tabel 4.26 sebagai berikut :

Tabel 4. 26 Pengujian Retur

Kasus dan Hasil Uji (Data Normal)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data retur Contoh Pilih jasa pengiriman : JNE Masukan nomor resi : 123456	Menampilkan pesan “menunggu barang dikirm petugas”	Dapat menampilkan pesan “menunggu barang dikirm petugas”	[√] Diterima [ ] Ditolak
Data retur Contoh Pilih jasa pengiriman : ‘kosong’ Masukan nomor resi : ‘kosong’	Menampilkan pesan “anda belum memilih jasa pengiriman dan nomor resi”	Dapat menampilkan pesan “anda belum memilih jasa pengiriman dan nomor resi”	[√] Diterima [ ] Ditolak

4.3.2.16 Pengujian Daftar *Member*

Untuk penjelasan lebih lanjut tentang sekenario daftar *member* dapat dilihat pada tabel 4.27 sebagai berikut :

Tabel 4. 27 Pengujian Daftar *Member*

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *Member* Contoh Nama: Ferlian Juniardi Email: arsevo21@gmail.com Password : 123456 Provinsi : Jawa Barat Kota : Bandung Alamat : Jl. Jabung C2 No 18 Kode Pos : 40534 Telepon : 085720225818	Menampilkan pesan “Terima kasih telah menjadi *member* kami sialahkan cek email untuk mengaktifkan akun anda”	Dapat menampilkan pesan “Terima kasih telah menjadi *member* kami sialahkan cek email untuk mengaktifkan akun anda”	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *Member* Contoh Nama: 'kosong' Email: 'kosong' Password : 'kosong' Provinsi : 'kosong' Kota : 'kosong' Alamat : 'kosong' Kode Pos : 'kosong' Telepon : 'kosong'	Menampilkan pesan "nama belum diisi email belum diisi Periksa kembali alamat email Password belum diisi Password minimal 6 dan maksimal 32 digit Ulangi password belum diisi provinsi belum diisi kota belum diisi alamat belum diisi kode pos belum diisi kode pos harus berupa angka"	Dapat Menampilkan pesan "nama belum diisi email belum diisi Periksa kembali alamat email Password belum diisi Password minimal 6 dan maksimal 32 digit Ulangi password belum diisi provinsi belum diisi kota belum diisi alamat belum diisi kode pos belum diisi kode pos harus berupa angka"	[√] Diterima [ ] Ditolak

4.3.2.17 Pengujian *Login Member*

Untuk penjelasan lebih lanjut tentang sekenario login *member* dapat dilihat pada tabel 4.28 sebagai berikut :

Tabel 4. 28 Pengujian *Login Member*

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Member Contoh masukan : Email : arsevo21@gmail.com Password : 123	Menampilkan menu utama *member*	Dapat menampilkan menu utama *member*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Member Contoh masukan : Email : 'kosong'	Menampilkan pesan kesalahan "*Email* atau	Tidak dapat *login* dan muncul pesana *email* dan *password*	[√] Diterima [ ] Ditolak

Password : ‘kosong’	*password* anda salah”	tidak valid, coba lagi dengan menggunakan *email* Anda yang lengkap	

4.3.3 Kesimpulan Pengujian Alpha

Berdasarkan hasil pengujian alpha (fungsional) dengan kasus uji diatas dapat ditarik kesimpulan sebagai berikut :

1. Jika data yang dimasukan benar maka sistem secara fungsional akan mengeluarkan keluran sesuai dengan harapan.
2. Jika data yang dimasukan salah maka sistem akan menunjukan data yang salah dan tombol simpan tidak dapat berfungsi.

Dengan kata lain bahwa pembangunan sistem ini bebas kesalahan dan secara fisik fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

4.4 Pengujian Beta (Kuesioner)

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan mengenai kepuasan user dengan kandungan poin yaitu pemenuhan kebutuhan dari tujuan awal pembangunan *website e-commerce* dan tampilan antarmuka dari *website e-commerce* tersebut. Pengujian beta dilakukan melalui kuesioner.

Pengujian beta melalui kuesioner dilakukan dengan membuat kuesioner mengenai kepuasan user dengan kandungan poin syarat *user friendly* untuk selanjutnya dibagikan kepada sebagian user dengan mengambil sample sebanyak 10 orang *member* dan pengunjung. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

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

1. Lingkungan penjualan di Ramdhan Distro

Penelitian di lingkungan penjual di Ramdhan Distro dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan di awal.

2. Lingkungan masyarakat umum

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

4.4.1 Wawancara Pengguna

Wawancara dilakukan dengan menggunakan teknik kualitatif melalui pertanyaan terbuka. Wawancara mengenai halaman admin dilakukan kepada 1 orang yaitu pemilik Ramdhan Distro, sedangkan untuk wawancara mengenai halaman petugas dilakukan terhadap 1 orang yaitu petugas pemasaran Ramdhan Distro. Dari hasil wawancara tersebut akan dilakukan pengambilan kesimpulan terhadap penilaian penerapan sistem yang baru. Wawancara terdiri dari 5 pertanyaan terbuka yang akan ditanyakan kepada narasumber.

4.4.1.1 Wawancara Pengujian Beta Untuk Admin

Wawancara dilakukan untuk pengujian beta terhadap *website e-commerce* Ramdhan Distro kepada pemilik toko dengan total narasumber sebanyak 1 orang. Berikut ini adalah masing-masing jawaban dari pertanyaan hasil wawancara tersebut :

1. Menurut Anda, apakah halaman admin pada *website e-commerce* Ramdhan Distro ini sudah sesuai dengan kebutuhan anda dalam mengolah data ?

Jawaban: *Website E-Commerce* ini sudah mengurangi beban kerja jika dibandingkan sebelumnya.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman admin pada *website e-commerce* Ramdhan Distro sudah mampu memenuhi kebutuhan admin dalam mengolah data.

2. Menurut Anda, apakah cara pemasaran barang yang diterapkan di *website e-commerce* ini dapat memperluas cakupan pemasaran Ramdhan Distro ?
   Jawaban: Bisa saja, karena dengan situs *online* ini konsumen yang berasal dari jauh pun bisa mengakses situs ini karena bisa dicari melalui mesin pencari.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa cara pemasaran barang yang diterapkan di *website e-commerce* ini dapat memperluas cakupan pemasaran Ramdhan Distro dibandingkan sebelumnya.

3. Menurut Anda, apakah halaman admin dalam *website* ini dapat membantu Anda dalam mengelola *restore* dan *backup* data ?
   Jawaban: Sangat membantu kami dalam menyimpan data yang selama ini kami belum miliki.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman admin dalam *website* ini dapat membantu admin dalam mengelola data.

4. Menurut Anda, apakah halaman admin dalam *website* ini dapat membantu Anda dalam mengelola data petugas ?
   Jawaban: Ya, dengan *website* ini saya dapat dengan mudah mengelola data petugas yang menangani penjualan melalui *website*.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman admin dalam *website* ini dapat membantu admin dalam mengelola data petugas.

5. Menurut Anda, apakah fungsionalitas dari halaman admin ini mudah dipahami dan digunakan ?
   Jawaban : Setelah saya coba, fungsionalitas yang ada cukup simpel dan mudah dipahami.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa fungsionalitas dari halaman admin ini mudah dipahami digunakan.

4.4.1.2 Wawancara Pengujian Beta Untuk Petugas

Wawancara dilakukan untuk pengujian beta terhadap *website e-commerce* Ramdhan Distro kepada petugas toko dengan total narasumber sebanyak 1 orang Berikut ini adalah masing-masing jawaban dari pertanyaan hasil wawancara tersebut.

1. Menurut anda apakah halaman petugas pada *website e-commerce* Ramdhan Distro ini sudah mampu memenuhi kebutuhan Anda dalam pengelolaan data ?
   Jawaban: Ya, dengan *website* ini saya dapat mengelola data dengan lebih mudah.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman petugas pada *website e-commerce* Ramdhan Distro ini sudah mampu memenuhi kebutuhan petugas dalam pengelolaan data.

2. Menurut pendapat Anda, apakah halaman petugas dalam *website* ini dapat membantu Anda dalam mengelola data barang ?
   Jawaban: Ya, Saya merasa lebih mudah dan cepat lagi dalam mendata barang-barang yang ada di toko ini.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman petugas dalam *website* inidapat membantu petugas dalam mengelola data barang.

3. Menurut pendapat Anda, apakah halaman petugas dalam *website* ini dapat membantu Anda dalam mengelola data penjualan secara *online*?
   Jawaban: Ya, halaman petugas dalam *website* ini sangat membantu dalam mengelola data penjualan secara *online*.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman petugas dalam *website* inidapat membantu petugas dalam mengelola data penjualan secara *online*.

4. Menurut pendapat Anda, apakah halaman petugas dalam *website* ini dapat membantu Anda dalam pembuatan laporan penjualan ?
   Jawaban: Ya, saya tidak perlu lagi mencatat kedalam berkas laporan setiap hari.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman petugas dalam *website* ini dapat membantu petugas dalam pembuatan laporan penjualan.

5. Menurut pendapat Anda, apakah fungsionalitas dari halaman petugas ini mudah dipahami dan digunakan ?

   Jawaban: Ya, menurut saya sudah sangat mudah untuk dipahami.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa fungsionalitas dari halaman petugas ini mudah dipahami dan digunakan.

4.4.2 Kuesioner Untuk Pengguna

Kuesioner disebarkan menggunakan teknik *quota sampling* dimana pemilihan sampel dengan memilih sejumlah tertentu (kuota) dan paling mudah diperoleh seperti yang dikehendaki oleh peneliti. Kuesioner disebarkan kepada 30 orang, dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat mengambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner teridiri dari 5 pertanyaan dengan menggunakan skala 1 sampai 5 (skala likert).

Tabel 4. 29 Tabel Bobot

Keterangan	**Bobot**
Sangat Setuju	5
Cukup Setuju	4
Setuju	3
Kurang Setuju	2
Tidak Setuju	1

Untuk mencari nilai persentase dari masing – masing jawaban kuesioner digunakan rumus skala likert sebagai berikut :

(4.1)

$$Y = \frac{S}{skorideal} \times 100$$

Keterangan :

Y = Nilai persentase yang dicari

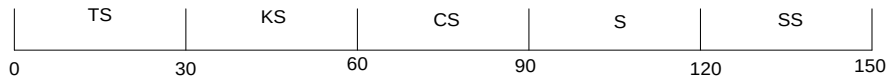
S =Total yang diperoleh dari S =$\sum_{i=1}^{5} n_i . r_i$

r_i =Banyak responden yang menjawab jawaban ke-i

n_i = Nilai jawaban ke-i

Skorideal = Nilai tertinggi dikalikan dengan jumlah sampel atau responden

Dibawah ini merupakan skala untuk mengmbil kesimpulan dari setiap petanyaan



Berikut ini adalah hasil persentase hasil masing – masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuesioner ini diuji kepada 30 orang.

1. Apakah Anda setuju *website* penjualan *online* Toko Ramdhan Distro mudah untuk dipahami?

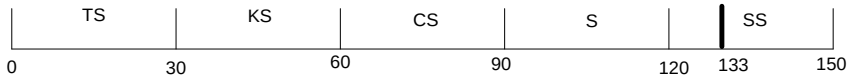
Tabel 4. 30 Hasil Pengujian Kuesioner soal nomor 1 (Situs *Frontend*)

Pertanyaan	Keterangan	Nilai	Responden	$n_i.r_i$
1	Sangat Setuju	5	16	80
	Setuju	4	11	44
	Cukup Setuju	3	3	9
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			**30**	**133**

Berdasarkan data dari hasil kuesioner tersebut, dapat dicari prosentase jawaban dengan menggunakan rumus :

$$Y = \frac{133}{30\ x\ 5} \times 100 = \ 88{,}6\ \%$$

Skala



Berdasarkan hasil persentase skala diatas maka dapat disimpulkan 82% menyatakan sangat setuju bahwa tampilan *website* penjualan *online* Toko Ramdhan Distro sudah cukup bagus.

2. Apakah *website* penjualan *online* Toko Ramdhan Distro dapat membantu Anda untuk mendapatkan informasi terbaru tentang produk Toko Ramdhan Distro?

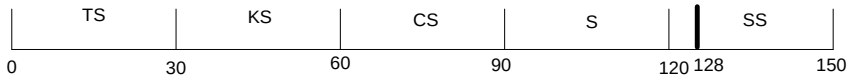
Tabel 4. 23 Hasil Pengujian Kuesioner soal nomor 2 (Situs *Frontend*)

Pertanyaan	**Keterangan**	**Nilai**	**Responden**	**$n_i.r_i$**
2	Sangat Membantu	5	12	60
	Membantu	4	15	60
	Cukup Membantu	3	2	6
	Kurang Membantu	2	1	2
	Tidak Membantu	1	0	0
Jumlah			**30**	**128**

Berdasarkan data dari hasil kuesioner tersebut, dapat dicari prosentase jawaban dengan menggunakan rumus :

$$Y = \frac{128}{30\ x\ 5} \times 100 = \ 85{,}3\ \%$$

Skala



Berdasarkan hasil persentase skala diatas maka dapat disimpulkan 76% menyatakan *website* penjualan *online* Toko Ramdhan Distro dapat membantu untuk mendapatkan informasi terbaru tentang produk Toko Ramdhan Distro.

3. Apakah Anda setuju informasi yang ada pada *website* penjualan *online* Toko Ramdhan Distro sudah cukup lengkap ?

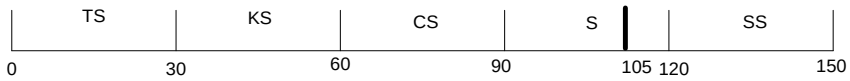
Tabel 4. 24 Hasil Pengujian Kuesioner soal nomor 3 (Situs *Frontend*)

Pertanyaan	**Keterangan**	**Nilai**	**Responden**	**$n_i.r_i$**
3	Sangat Setuju	5	13	45
	Setuju	4	10	40
	Cukup Setuju	3	6	18
	Kurang Setuju	2	1	2
	Tidak Setuju	1	0	0
Jumlah			**30**	**105**

Berdasarkan data dari hasil kuesioner tersebut, dapat dicari prosentase jawaban dengan menggunakan rumus :

$$Y = \frac{105}{30\ x\ 5} \times 100 = \ 70\ \%$$

Skala



Berdasarkan hasil presentase skala diatas maka dapat disimpulkan 78% menyatakan setuju informasi yang ada pada *website* penjualan *online* Toko Ramdhan Distro mudah sudah cukup lengkap.

4. Apakah Anda setuju *website* penjualan *online* Toko Ramdhan Distro mempermudah anda dalam melakukan transaksi pembelian produk ?

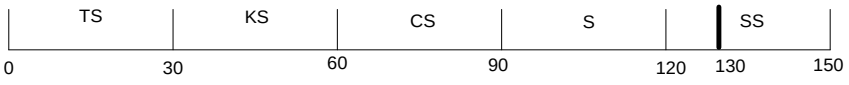
Tabel 4. 25 Hasil Pengujian Kuesioner soal nomor 4 (Situs *Frontend*)

Pertanyaan	**Keterangan**	**Nilai**	**Responden**	$\mathbf{n_i.r_i}$
4	Sangat Setuju	5	15	75
	Setuju	4	10	40
	Cukup Setuju	3	5	15
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			**30**	**130**

Berdasarkan data dari hasil kuesioner tersebut, dapat dicari prosentase jawaban dengan menggunakan rumus :

$$Y = \frac{130}{30\ x\ 5} \times 100 = \ 86{,}6\ \%$$

Skala



Berdasarkan hasil persentase skala diatas maka dapat disimpulkan 88% menyatakan sangat setuju proses pembayaran yang ada pada *website* penjualan *online* Toko Ramdhan Distro aman dilakukan.

5. Apakah Anda setuju jika website Toko Ramdhan Distro mudah dicari di search engine ?

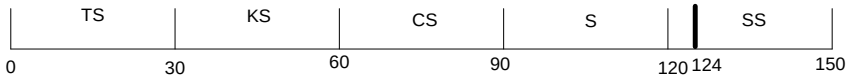
Tabel 4. 26 Hasil Pengujian Kuesioner soal nomor 5 (Situs *Frontend*)

Pertanyaan	**Keterangan**	**Nilai**	**Responden**	**$n_i.r_i$**
5	Sangat Setuju	5	10	50
	Setuju	4	14	56
	Cukup Setuju	3	6	18
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			**30**	**124**

Berdasarkan data dari hasil kuesioner tersebut, dapat dicari prosentase jawaban dengan menggunakan rumus :

$$Y = \frac{124}{30\ x\ 5} \times 100 = \ 82{,}6\ \%$$

Skala



Berdasarkan hasil persentase skala diatas maka dapat disimpulkan 92% menyatakan sangat setuju bahwa *website* penjualan *online* Toko Ramdhan Distro dapat dengan mudah di cari melalui *search engine*.

4.4.3 Kesimpulan Pengujian Beta

Berdasarkan pengujian beta yang telah dilakukan, dapat diambil kesimpulan bahwa:

1. *Website* ini mempermudah toko Ramdhan Distro dalam melakukan pengolahan data barang dan pembuatan laporan.
2. *Website* ini memberikan sarana promosi produk-produk Ramdhan Distro.
3. *Website* ini memberikan kemudahan dalam melakukan transaksi pembelian produk.
4. *Website* ini memberikan informasi produk yang disukai di toko Ramdhan Distro.
5. *Website* ini mudah dicari melalui *search engine.*