BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi

Analisis dan perancangan sistem secara terperinci telah dilakukan, maka langkah selanjutnya adalah menuju tahap implementasi. Tujuan implementasi adalah untuk menjelaskan tentang manual modul kepada semua *user* yang akan menggunakan sistem. Sehingga *user* tersebut dapat merespon apa yang ditampilkan di sistem dan memberikan masukan kepada pembuat sistem untuk dilakukan perbaikan agar sistem lebih baik lagi.

4.1.1 Lingkungan Implementasi

Untuk mendukung aplikasi yang akan diterapkan pada lingkungan implementasi, maka dalam hal ini harus memperhatikan perangkat keras dan perangkat lunak yang menunjang dalam Pembangunan Aplikasi *E-commerce* pada toko Mix.

4.1.2 Implementasi Perangkat Keras

Spesifikasi perangkat keras minimal yang dibutuhkan untuk mengimplementasikan aplikasi yang di bangun antara lain terdapat pada tabel 4.1.

Tabel 4.1 Perangkat Keras yang dibutuhkan

No	Perangkat Keras	Spesifikasi
1	Prosessor	Kecepatan 2.0 GHz
2	Monitor	Ukuran 15” Resolusi 1024x768 px
3	VGA	Onboard 64 Mb
4	Memori	512 MB
5	Hard Disk	40 GB (ruang kosong 2 GB)
6	Keyboard	Keyboard
7	Mouse	Mouse
8	Printer	Hitam putih dan warna
9	Jaringan Internet	LAN dan Internet

4.1.3 Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang dibutuhkan untuk mengimplementasikan perangkat lunak antara lain terdapat pada tabel 4.2.

Tabel 4.2 Perangkat Lunak yang digunakan

No	Perangkat Keras	Spesifikasi
1	Sistem operasi	Windows Xp
2	Web Browser	Mozilla firefox, opera, internet explorer, google chrome

4.1.4 Implementasi Basis Data

Implementasi data dalam aplikasi *E-Commerce* pada toko Mix ini menggunakan satu buah DBMS (*Database Management System*) dan sebuah GUI *database* untuk memudahkan tugas *administrator*. DBMS yang digunakan untuk menimplentasikan datanya adalah MySQL. Rancangan data beserta nama file query yang dilakukan adalah sebagai berikut:

1. Tabel agama

   Tabel agama berfungsi untuk menyimpan data agama.

```
CREATE TABLE IF NOT EXISTS `agama` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) DEFAULT NULL,
  `date_created` datetime DEFAULT NULL,
  `date_updated` datetime DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

2. Tabel cart_temp

   Tabel Cart_temp digunakan untuk menyimpan data produk yang ada pada shoping cart.

```
CREATE TABLE IF NOT EXISTS `cart_temp` (
  `rowid` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `session_rowid` text NOT NULL,
  `detail_produk_id` varchar(20) NOT NULL,
  `member_id` varchar(100) NOT NULL,
  `qty` int(11) NOT NULL,
  `cart_date` datetime NOT NULL,
  `daftar_harga_id` int(11) NOT NULL,
  `harga` int(11) NOT NULL,
  `gambar` text NOT NULL,
  `name` varchar(100) NOT NULL,
  `diskon_id` int(11) NOT NULL,
  `diskon` int(11) NOT NULL,
  `min_quantity` int(11) NOT NULL,
  `stok` int(11) NOT NULL,
  PRIMARY KEY (`rowid`)
  KEY `FK_reference_20` (`detail_produk_id`)
  KEY `FK_member_cart_temp` (`member_id`)
  KEY `FK_daftar_harga_cart_temp` (`daftar_harga_id`)
  KEY `FK_diskon_cart_temp` (`diskon_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

3. Tabel config

Tabel config digunakan untuk menyimpan data konfigurasi pada *web*.

```
CREATE TABLE IF NOT EXISTS `config` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `users_id` int(11) NOT NULL,
  `nama` text NOT NULL,
  `slogan` text NOT NULL,
  `email` text NOT NULL,
  `facebook` text NOT NULL,
  `twiter` text NOT NULL,
  `telp` int(11) NOT NULL,
  `alamat` text NOT NULL,
  PRIMARY KEY (`id`)
  KEY `FK_users_config` (`user_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

4. Tabel daftar_harga

Tabel daftar_harga digunakan untuk menyimpan data harga produk.

```
CREATE TABLE IF NOT EXISTS `daftar_harga` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `detail_produk_id` varchar(10) DEFAULT NULL,
  `harga` int(11) DEFAULT NULL,
  `date_created` datetime DEFAULT NULL,
  `date_updated` datetime DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `FK_reference_20` (`detail_produk_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;
```

5. Tabel detail_pembelian

Tabel detail_pembelian digunakan untuk menyimpan data detail pembelian yang dilakukan oleh pelanggan.

```
CREATE TABLE IF NOT EXISTS `detail_pembelian` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `pembelian_id` int(11) DEFAULT NULL,
  `detail_produk_id` varchar(10) DEFAULT NULL,
  `diskon_id` int(11) DEFAULT NULL,
  `daftar_harga_id` int(11) DEFAULT NULL,
  `quantity` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `FK_daftar_harga_detail_pembelian` (`daftar_harga_id`),
```

```
  KEY `FK_diskon_detail_pembelian` (`diskon_id`),
  KEY `FK_pembelian_detail` (`pembelian_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=36 ;
```

6. Tabel detail_produk

Tabel detail_produk digunakan untuk menyimpan data detail produk.

```
CREATE TABLE IF NOT EXISTS `detail_produk` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) DEFAULT NULL,
  `warna` varchar(50) DEFAULT NULL,
  `stok` int(11) DEFAULT NULL,
  `gambar` varchar(50) DEFAULT NULL,
  `bentuk` varchar(50) DEFAULT NULL,
  `produk_kode` varchar(5) DEFAULT NULL,
  `ukuran` varchar(5) DEFAULT NULL,
  `status_produk` enum('preorder','stok') DEFAULT NULL,
  `date_created` datetime DEFAULT NULL,
  `date_updated` datetime DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `FK_produk_detail_produk` (`produk_kode`)
```

```
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=9 ;
```

7. Tabel detail_retur

Tabel detail_retur digunakan untuk menyimpan informasi data kurs.

```
CREATE TABLE IF NOT EXISTS `detail_retur` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `retur_pembelian_id` int(11) NOT NULL,
  `detail_pembelian_id` int(11) NOT NULL,
  `retur_detail_produk_id` int(11) NOT NULL,
  `jumlah` varchar(45) DEFAULT NULL,
  `keterangan_retur` varchar(45) DEFAULT NULL,
  `status` enum('baru','ditolak', 'disetujui', 'cofirm',
  'accept_confirm')  DEFAULT NULL,
  `gambar` text,
  `jumlah_pengganti` text,
  `keterangan_verifikasi` text,
  PRIMARY KEY (`id`)
  KEY `FK_retur_detail_retur` (`retur_id`)
  KEY `FK_detail_pembelian_detail_retur`
  (`detail_pembelian_id`)
```

```
  KEY `FK_detail_produk_detail_retur` (`detail_produk_id`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

8. Tabel diskon

   Tabel diskon digunakan untuk menyimpan data diskon.

```
CREATE TABLE IF NOT EXISTS `diskon` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `produk_kode` varchar(5) DEFAULT NULL,

  `diskon` int(11) DEFAULT NULL,

  `berawal` datetime DEFAULT NULL,

  `berakhir` datetime DEFAULT NULL,

  `min_quantity` int(11) DEFAULT NULL,

  PRIMARY KEY (`id`),

  KEY `FK_produk_diskon` (`produk_kode`)

) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COMMENT='daftar
produk yang sedang memiliki diskon' AUTO_INCREMENT=8 ;
```

9. Tabel groups

   Tabel groups digunakan untuk menyimpan data groups *user*.

```
CREATE TABLE IF NOT EXISTS `groups` (

  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `nama` varchar(50) DEFAULT NULL,
  `status` enum('aktif','nonaktif') DEFAULT NULL,
  `deskripsi` text,
  `date_created` datetime DEFAULT NULL,
  `date_updated` datetime DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=100 ;
```

10. Tabel kategori

Tabel kategori digunakan untuk menyimpan data kategori.

```
 CREATE TABLE IF NOT EXISTS `kategori` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) DEFAULT NULL,
  `deskripsi` text,
  `date_created` datetime DEFAULT NULL,
  `date_updated` datetime DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

11. Tabel kecamatan

Tabel kecamatan digunakan untuk menyimpan data kecamatan.

```
CREATE TABLE IF NOT EXISTS `kecamatan` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) DEFAULT NULL,
  `kota_id` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `FK_kecamatan_kota` (`kota_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

12. Tabel kelurahan

Tabel kelurahan digunakan untuk menyimpan data kelurahan.

```
CREATE TABLE IF NOT EXISTS `kelurahan` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) DEFAULT NULL,
  `kecamatan_id` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `FK_kelurahan_kecamatan` (`kecamatan_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

13. Tabel kota

Tabel kota digunakan untuk menyimpan data kota.

```
`CREATE TABLE IF NOT EXISTS `kota` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) DEFAULT NULL,
  `propinsi_id` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `FK_propinsi_kota` (`propinsi_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

14. Tabel member

Tabel member digunakan untuk menyimpan data member.

```
CREATE TABLE IF NOT EXISTS `member` (
  `no_ktp` varchar(30) NOT NULL,
  `nama` varchar(50) DEFAULT NULL,
  `agama_id` int(11) DEFAULT NULL,
  `jenis_kelamin` enum('pria','wanita') DEFAULT NULL,
  `alamat` varchar(80) DEFAULT NULL,
  `kelurahan_id` int(11) DEFAULT NULL,
  `telp` varchar(20) DEFAULT NULL,
```

```
  `email` text,
  `password` varchar(100) NOT NULL,
  `aktivasi_key` varchar(100) NOT NULL,
  `status` enum('aktif','nonaktif','new') NOT NULL,
  PRIMARY KEY (`no_ktp`),
  KEY `FK_agama_member` (`agama_id`),
  KEY `FK_kelurahan_member` (`kelurahan_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

15. Tabel pages

Tabel pages digunakan untuk menyimpan data pages.

```
CREATE TABLE IF NOT EXISTS `pages` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `name` varchar(100) NOT NULL,
  `intitle` varchar(100) NOT NULL,
  `category` enum('static','dinamic') NOT NULL DEFAULT
  'static',
  `incontent` text NOT NULL,
  `entitle` varchar(100) NOT NULL,
  `encontent` text NOT NULL,
```

```
  `date_created` datetime NOT NULL,
  `date_updated` timestamp NOT NULL DEFAULT
  CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;
```

16. Tabel pembayaran

Tabel pembayaran digunakan untuk menyimpan data pembayaran.

```
CREATE TABLE IF NOT EXISTS `pembayaran` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `pembelian_id` int(11) DEFAULT NULL,
  `tanggal` datetime DEFAULT NULL,
  `jenis_bayar` varchar(16) DEFAULT NULL,
  `no_rekening` varchar(16) DEFAULT NULL,
  `no_rek_tujuan` varchar(16) DEFAULT NULL,
  `rekening_id` int(11) NOT NULL,
  `total` int(11) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `FK_pembayaran_pembelian` (`pembelian_id`)
  KEY `FK_pembayaran_rekening` (`rekening_id`)
```

```
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;
```

17. Tabel pembelian

Tabel pembelian digunakan untuk menyimpan data pembelian.

```
CREATE TABLE IF NOT EXISTS `pembelian` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `rekening_id` int(11) NOT NULL,


  `tanggal` datetime DEFAULT NULL,
  `member_no_ktp` varchar(30) DEFAULT NULL,
  `subtotal` int(11) DEFAULT NULL,
  `total` int(11) DEFAULT NULL,
  `status_pembayaran` enum('sudah','blom') DEFAULT NULL,
  `status_barang` enum('stok','preorder') DEFAULT NULL,
  `type_pembayaran` enum('paypall','transfer','cod') DEFAULT
   NULL,
  `invoice` varchar(5) DEFAULT NULL,
  `currency` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `FK_pembelian_member` (`member_no_ktp`)
```

```
  KEY `FK_pembayaran_rekening` (`rekening_id`)

) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=24 ;
```

18. Tabel pengiriman

Tabel pengiriman digunakan untuk menyimpan data pengiriman.

```
CREATE TABLE IF NOT EXISTS `pengiriman` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `nama` varchar(100) DEFAULT NULL,

  `pembelian_id` int(11) DEFAULT NULL,

  `vendor_id` int(11) DEFAULT NULL,

  `tarif_id` int(11) DEFAULT NULL,

  `kode_pengiriman` text,

  `total_biaya` int(11) DEFAULT NULL,

  `tanggal` datetime DEFAULT NULL,

  `alamat` varchar(45) DEFAULT NULL,

  `no_telp` varchar(45) DEFAULT NULL,

  `kelurahan_id` int(11) NOT NULL,

  `status_pengiriman` enum('belum','mengirim','terkirim')

  DEFAULT 'belum',

  PRIMARY KEY (`id`,`kelurahan_id`),
```

```
  KEY `FK_pengiriman_pembelian` (`pembelian_id`),
  KEY `FK_tarif_pengiriman` (`tarif_id`),
  KEY `FK_vendor_pengiriman` (`vendor_id`),
  KEY `fk_pengiriman_kelurahan1` (`kelurahan_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=21 ;
```

19. Tabel produk

Tabel produk digunakan untuk menyimpan data produk.

```
CREATE TABLE IF NOT EXISTS `produk` (
  `kode` varchar(5) NOT NULL,
  `nama` varchar(50) DEFAULT NULL,
  `deskripsi` text,
  `gambar` varchar(100) DEFAULT NULL,
  `kategori_id` int(11) DEFAULT NULL,
  `status_produk` enum('preorder','stok') DEFAULT NULL,
  `date_created` datetime DEFAULT NULL,
  `date_updated` datetime DEFAULT NULL,
  `users_id` int(11) NOT NULL,
  PRIMARY KEY (`kode`,`users_id`),
  KEY `kategori_id` (`kategori_id`),
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

20. Tabel propinsi

Tabel propinsi digunakan untuk menyimpan data propinsi.

```
CREATE TABLE IF NOT EXISTS `propinsi` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

21. Tabel rating

Tabel rating digunakan untuk menyimpan data rating.

```
CREATE TABLE IF NOT EXISTS `rating` (
  `detail_produk_id` int(11) NOT NULL,
  `member_email` text NOT NULL,
  `rating` int(11) NOT NULL,
  `tanggal` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
   PRIMARY KEY (`member_email`,`detail_produk_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

22. Tabel rekening

Tabel rekening digunakan untuk menyimpan data rekening.

```
CREATE TABLE IF NOT EXISTS `rekening` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `pembayaran_id` int(11) DEFAULT NULL,
  `member_no_ktp` varchar(30) NOT NULL,
  `bank` varchar(80) NOT NULL,
  `rekening` bigint(20) NOT NULL,
  PRIMARY KEY (`id`)
  KEY `pembayaran_rekening` (`pembayaran_id`)
  KEY `member_rekening` (`member_no_ktp`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

23. Tabel retur

Tabel retur digunakan untuk menyimpan data retur.

```
CREATE TABLE IF NOT EXISTS `retur` (
  `pembelian_id` int(11) NOT NULL,
  `tanggal_retur` datetime DEFAULT NULL,
  PRIMARY KEY (`pembelian_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

24. Tabel tarif

Tabel tarif digunakan untuk menyimpan data tarif.

```
CREATE TABLE IF NOT EXISTS `tarif` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `vendor_id` int(11) DEFAULT NULL,
  `kode` varchar(5) DEFAULT NULL,
  `nama` varchar(30) DEFAULT NULL,
  `asal` varchar(40) DEFAULT NULL,
  `tujuan` varchar(50) DEFAULT NULL,
  `tarif` int(11) DEFAULT NULL,
  `satuan` varchar(5) DEFAULT NULL,
  `pola_perhitungan` enum('satuan','all') DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `FK_tarif_vendor` (`vendor_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

25. Tabel testimonial

Tabel testimonial digunakan untuk menyimpan data testimonial.

```
CREATE TABLE IF NOT EXISTS `testimonial` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `email` text NOT NULL,
  `nama` varchar(50) NOT NULL,
  `deskripsi` text NOT NULL,
  `pages` varchar(50) NOT NULL,
  `tanggal` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `member_no_ktp` varchar(30) NOT NULL,
  PRIMARY KEY (`id`)
  KEY `fk_testimonial_member` (`member_no_ktp`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

26. Tabel users

Tabel users digunakan untuk menyimpan data users.

```
CREATE TABLE IF NOT EXISTS `users` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(50) DEFAULT NULL,
  `email` varchar(30) DEFAULT NULL,
  `password` varchar(50) DEFAULT NULL,
  `group_id` int(11) DEFAULT NULL,
  `status` text NOT NULL,
  PRIMARY KEY (`id`),
```

```
  KEY `FK_group_users` (`group_id`)

) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

27. Tabel vendor

Tabel vendor digunakan untuk menyimpan data vendor.

```
CREATE TABLE IF NOT EXISTS `vendor` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `nama` varchar(50) DEFAULT NULL,

  `deskripsi` text,

  `kode` varchar(5) DEFAULT NULL,

  `site` text NOT NULL,

  `date_created` datetime DEFAULT NULL,

  `date_updated` datetime DEFAULT NULL,

  PRIMARY KEY (`id`)

) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

4.1.5 Implementasi Antarmuka

Implementasi antar muka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk file program. Berikut ini adalah implementasi antarmuka untuk pengguna. Implementasi antarmuka admin pada aplikasi *E-Commerce* pada toko mix dapat dilihat pada tabel 4.3.

Tabel 4.3 Implementasi Antarmuka Super Admin

Menu	Deskripsi	Nama File
Login *Users*	Digunakan sebagai halaman login *users.*	login_backend/login.php
Halaman Utama *Users*	Halaman utama ketika *users* berhasil login yang memperlihatkan *dashboard management sistem* dan menu pengolahan *users*.	backend/index.php
Groups	Halaman *groups* digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data *groups*.	groups/groups.php
Users	Halaman *users* digunakan untuk melihat, menambah, mencari, dan mengubah data *users*.	users/users.php
Member	Halaman *member* digunakan untuk melihat, menambah, mencari, dan mengubah data *member*.	members/member.php
Web_config	Halaman *web_config* digunakan untuk melihat, mencari, mengubah dan menghapus data *web_config*.	pages/web_config.php
About_us	Halaman *about_us* digunakan untuk melihat, dan mengubah data *about_us*.	pages/about_us.php
Delivery	Halaman *delivery* digunakan untuk melihat, dan mengubah data *delivery*.	pages/delivery.php

Payment	Halaman *payment* digunakan untuk melihat, dan mengubah data *payment*.	pages/payment.php
FAQ	Halaman *FAQ* digunakan untuk melihat, dan mengubah data *FAQ*.	pages/faq.php
Cara Pembayaran	Halaman cara pembayaran digunakan untuk melihat, dan mengubah data cara pembayaran.	pages/cara_menjadi_anggota.php
Cara Menjadi Anggota	Halaman cara menjadi anggota digunakan untuk melihat, dan mengubah data cara menjadi anggota.	pages/cara_menjadi_anggota.php
Cara Konfirmasi Pengiriman	Halaman cara konfirmasi pengiriman digunakan untuk melihat, dan mengubah data cara konfirmasi pengiriman.	pages/cara_konfirmasi_pengiriman.php
Kategori	Halaman kategori digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data kategori.	category/kategori.php
Produk	Halaman produk digunakan untuk melihat detail produk, menambah produk, menambah detail produk, mencari, dan mengubah data produk.	products/produk.php
Diskon	Halaman diskon digunakan untuk melihat, menambah, mencari, dan mengubah data diskon.	discounts/diskon.php

Daftar Harga	Halaman daftar harga digunakan untuk melihat, dan mencari data daftar harga.	price_lists/daftar_harga.php
Penjualan	Halaman penjualan digunakan untuk melihat detail penjualan, dan mencari data penjualan.	sells/penjualan.php
Pembayaran	Halaman pembayaran digunakan untuk melihat detail pembayaran, dan mencari data pembayaran.	payment/pembayaran.php
Pengiriman	Halaman pengiriman digunakan untuk melihat, menambah, dan mencari data pengiriman.	delivers/pengiriman.php
Retur	Halaman retur digunakan untuk melihat detail retur, dan mencari data retur.	returns/retur.php
Laporan Penjualan	Halaman laporan penjualan digunakan untuk melihat, mencari, dan mencetak data laporan penjualan.	laporan/laporan_penjualan.php
Laporan Pengiriman	Halaman laporan pengiriman digunakan untuk melihat, mencari, dan mencetak data laporan pengiriman.	laporan/laporan_pengiriman.php
Laporan Stok	Halaman laporan stok digunakan untuk melihat, mencari, dan mencetak data laporan stok.	laporan/laporan_stok.php
Laporan Retur	Halaman laporan retur digunakan untuk melihat, mencari, dan	laporan/laporan_retur.php

	mencetak data laporan retur.	
Grafik Penjualan	Halaman grafik penjualan digunakan untuk melihat grafik penjualan.	grafik/grafik_penjualan.php
Grafik Stok	Halaman grafik stok digunakan untuk melihat grafik stok.	grafik/grafik_stok.php
Data Master: - Master Agama - Master Propinsi - Master Kota/Kab - Master Kecamatan - Master Kelurahan	Halaman yang digunakan untuk mengolah data master Halaman master agama digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data agama. Halaman master propinsi digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data propinsi. Halaman master kota/kab digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data kota/kab. Halaman master kecamatan digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data kecamatan. Halaman master kelurahan digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data kelurahan.	 agama/agama.php propinsi/propinsi.php kota/kota.php kecamatan/kecamatan.php kelurahan/kelurahan.php

Vendor Pengiriman	Halaman vendor pengiriman digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data vendor pengiriman.	vendors/vendor_pengiriman.php
Tarif Pengiriman	Halaman tarif pengiriman digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data tarif pengiriman.	fares/tarif_pengiriman.php

Tabel implementasi antarmuka *Owner* dapat dilihat seperti pada tabel 4.4 berikut ini:

Tabel 4.4 Implementasi Antarmuka *Owner*

Menu	**Deskripsi**	**Nama File**
Login Users	Digunakan sebagai halaman login *users.*	login_backend/login.php
Halaman Utama *Users*	Halaman utama ketika *users* berhasil login yang memperlihatkan *dashboard management sistem* dan menu pengolahan *users*.	backend/index.php
Daftar Harga	Halaman daftar harga digunakan untuk melihat, dan mencari data	price_lists/daftar_harga.php

	daftar harga.	
Retur	Halaman retur digunakan untuk melihat detail retur, dan mencari data retur.	returns/retur.php
Pembayaran	Halaman pembayaran digunakan untuk melihat detail pembayaran, dan mencari data pembayaran.	payment/pembayaran.php
Pengiriman	Halaman pengiriman digunakan untuk melihat, menambah, dan mencari data pengiriman.	delivers/pengiriman.php
Penjualan	Halaman penjualan digunakan untuk melihat detail penjualan, dan mencari data penjualan.	sells/penjualan.php

Tabel implementasi antarmuka keuangan dapat dilihat seperti pada tabel 4.5 berikut ini:

Tabel 4.5 Implementasi Antarmuka Keuangan

Menu	**Deskripsi**	**Nama File**
Login *Users*	Digunakan sebagai halaman login *users.*	Login Users
Halaman Utama *Users*	Halaman utama ketika *users* berhasil login yang memperlihatkan *dashboard management sistem* dan menu pengolahan *users*.	Halaman Utama *Users*
Daftar Harga	Halaman daftar harga digunakan	Daftar Harga

	untuk melihat, dan mencari data daftar harga.	
Retur	Halaman retur digunakan untuk melihat detail retur, dan mencari data retur.	Retur
Pembayaran	Halaman pembayaran digunakan untuk melihat detail pembayaran, dan mencari data pembayaran.	Pembayaran
Pengiriman	Halaman pengiriman digunakan untuk melihat, menambah, dan mencari data pengiriman.	Pengiriman
Penjualan	Halaman penjualan digunakan untuk melihat detail penjualan, dan mencari data penjualan.	Penjualan
Login *Users*	Digunakan sebagai halaman login *users.*	Login *Users*
Halaman Utama *Users*	Halaman utama ketika *users* berhasil login yang memperlihatkan *dashboard management sistem* dan menu pengolahan *users*.	Halaman Utama *Users*
Daftar Harga	Halaman daftar harga digunakan untuk melihat, dan mencari data daftar harga.	Daftar Harga
Retur	Halaman retur digunakan untuk melihat detail retur, dan mencari data retur.	Retur
Pembayaran	Halaman pembayaran digunakan untuk melihat detail pembayaran,	Pembayaran

	dan mencari data pembayaran.	
Pengiriman	Halaman pengiriman digunakan untuk melihat, menambah, dan mencari data pengiriman.	Pengiriman
Penjualan	Halaman penjualan digunakan untuk melihat detail penjualan, dan mencari data penjualan.	Penjualan
Login *Users*	Digunakan sebagai halaman login *users.*	Login Users

Tabel implementasi antarmuka petugas input data dapat dilihat seperti pada tabel 4.6 berikut ini:

Tabel 4.6 Implementasi Antarmuka Petugas Input Data

Menu	**Deskripsi**	**Nama File**
Login *Users*	Digunakan sebagai halaman login *users.*	login_backend/login.php
Halaman Utama *Users*	Halaman utama ketika *users* berhasil login yang memperlihatkan *dashboard management sistem* dan menu pengolahan *users*.	backend/index.php
About_us	Halaman *about_us* digunakan untuk melihat, dan mengubah data *about_us*.	pages/about_us.php
Delivery	Halaman *delivery* digunakan untuk melihat, dan mengubah data	pages/delivery.php

	delivery.	
Payment	Halaman *payment* digunakan untuk melihat, dan mengubah data *payment.*	pages/payment.php
Kategori	Halaman kategori digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data kategori.	category/kategori.php
Produk	Halaman produk digunakan untuk melihat detail produk, menambah produk, menambah detail produk, mencari, dan mengubah data produk.	products/produk.php
Diskon	Halaman diskon digunakan untuk melihat, menambah, mencari, dan mengubah data diskon.	discounts/diskon.php
Daftar Harga	Halaman daftar harga digunakan untuk melihat, dan mencari data daftar harga.	price_lists/daftar_harga.php
Penjualan	Halaman penjualan digunakan untuk melihat detail penjualan, dan mencari data penjualan.	sells/penjualan.php
Pembayaran	Halaman pembayaran digunakan untuk melihat detail pembayaran, dan mencari data pembayaran.	payment/pembayaran.php
Pengiriman	Halaman pengiriman digunakan untuk melihat, menambah, dan mencari data pengiriman.	delivers/pengiriman.php

Retur	Halaman retur digunakan untuk melihat detail retur, dan mencari data retur.	returns/retur.php
Data Master: - Master Agama - Master Propinsi - Master Kota/Kab - Master Kecamatan - Master Kelurahan	Halaman yang digunakan untuk mengolah data master Halaman master agama digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data agama. Halaman master propinsi digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data propinsi. Halaman master kota/kab digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data kota/kab. Halaman master kecamatan digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data kecamatan. Halaman master kelurahan digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data kelurahan.	agama/agama.php propinsi/propinsi.php kota/kota.php kecamatan/kecamatan.php kelurahan/kelurahan.php
Vendor Pengiriman	Halaman vendor pengiriman digunakan untuk melihat,	vendors/vendor_pengiriman.php

	menambah, mencari, mengubah dan menghapus data vendor pengiriman.	
Tarif Pengiriman	Halaman tarif pengiriman digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data tarif pengiriman.	fares/tarif_pengiriman.php

Tabel implementasi antarmuka *member* dapat dilihat seperti pada tabel 4.7 berikut ini:

Tabel 4.7 Implementasi Antarmuka *Member*

Menu	**Deskripsi**	**Nama File**
Beranda	Digunakan sebagai halaman utama pengunjung dan member berisi informasi mengenai produk terbaru, produk terlaris, dan kategori produk.	index.php
Semua Produk	Halaman yang menampilkan semua produk dan semua kategori.	frontend/products.php
Tentang Kami	Halaman yang berisi tentang profil, sejarah, dan penjelasan lainnya mengenai perusahaan.	frontend/static_pages/1.php
Kontak Kami	Halaman untuk menampilkan data kontak perusahaan yang dapat dihubungi oleh pengunjung atau member, disertai dengan alamat	frontend/contactus.php

	lengkap perusahaan.	
FAQ	Halaman yang berisi mengenai penjelasan aturan-aturan yang terkait dengan proses transaksi, seperti cara pembelian, pembayaran, pengiriman, dll.	frontend/static_pages/10.php
History Belanja	Halaman untuk menampilkan data pembelian yang pernah dilakukan oleh *member*.	frontend/history_wish.php
Testimonial	Halaman member yang fungsinya untuk memberikan testimonial.	frontend/testimonial.php
Data Akun	Halaman untuk menampilkan data akun *member*.	frontend/registrasi_member.php
Login	Halaman login untuk *member* yang sudah terdaftar dan aktif.	frontend/login.php
Register	Halaman untuk pengunjung yang ingin mendaftar menjadi *member* baru.	frontend/register.php
Checkout	Halaman *member* yang berfungsi untuk konfirmasi pembelian terhadap barang yang dipilih di dalam *shoping cart.*	frontend/checkout.php

Tabel implementasi antarmuka pengunjung dapat dilihat seperti pada tabel 4.8 berikut ini:

Tabel 4.8 Implementasi Antarmuka Pengunjung

Menu	Deskripsi	Nama File
Beranda	Digunakan sebagai halaman utama pengunjung dan *member* berisi informasi mengenai produk terbaru, produk terlaris, dan kategori produk.	index.php
Semua Produk	Halaman yang menampilkan semua produk dan semua kategori.	frontend/products.php
Tentang Kami	Halaman yang berisi tentang profil, sejarah, dan penjelasan lainnya mengenai perusahaan.	frontend/static_pages/1.php
Kontak Kami	Halaman untuk menampilkan data kontak perusahaan yang dapat dihubungi oleh pengunjung atau *member*, disertai dengan alamat lengkap perusahaan.	frontend/contactus.php
FAQ	Halaman yang berisi mengenai penjelasan aturan-aturan yang terkait dengan proses transaksi, seperti cara pembelian, pembayaran, pengiriman, dll.	frontend/static_pages/10.php
Beranda	Digunakan sebagai halaman utama pengunjung dan *member* berisi informasi mengenai produk terbaru, produk terlaris, dan kategori produk.	index.php
Semua Produk	Halaman yang menampilkan semua	frontend/products.php

	produk dan semua kategori.	
Tentang Kami	Halaman yang berisi tentang profil, sejarah, dan penjelasan lainnya mengenai perusahaan.	frontend/static_pages/1.php
Kontak Kami	Halaman untuk menampilkan data kontak perusahaan yang dapat dihubungi oleh pengunjung atau *member*, disertai dengan alamat lengkap perusahaan.	frontend/contactus.php
FAQ	Halaman yang berisi mengenai penjelasan aturan-aturan yang terkait dengan proses transaksi, seperti cara pembelian, pembayaran, pengiriman, dll.	frontend/static_pages/10.php
Login	Halaman login untuk *member* yang sudah terdaftar dan aktif.	frontend/login.php
Register	Halaman untuk pengunjung yang ingin mendaftar menjadi *member* baru.	frontend/register.php
Checkout	Halaman member yang berfungsi untuk konfirmasi pembelian terhadap barang yang dipilih di dalam *shoping cart.*	frontend/checkout.php

4.2 Pengujian Sistem

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

Pengujian yang digunakan untuk menguji sistem yang baru adalah metode pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

4.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara *alpha* dan *beta*. Pengujian pembangunan *E-Commerce* di toko Mix menggunakan data uji berdasarkan data dari toko Mix. Rencana pengujian selengkapnya terlihat pada tabel 4.9.

Tabel 4.9 Rencana Pengujian Sistem

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Login	Validasi *email* and password	*Black Box*
Pengolahan *Groups*	Tambah *Groups*	*Black Box*
	Edit *Groups*	*Black Box*
	Hapus *Groups*	*Black Box*

	Cari *Groups*	*Black Box*
Pengolahan Users	Tambah *Users*	*Black Box*
	Edit *Users*	*Black Box*
	Cari *Users*	*Black Box*
Pengolahan *Member*	Tambah *Member*	*Black Box*
	Edit *Member*	*Black Box*
	Cari *Member*	*Black Box*
Web_config	Edit *Web_config*	*Black Box*
	Hapus *Web_config*	*Black Box*
	Cari *Web_config*	*Black Box*
About Us	Edit *About_us*	*Black Box*
Delivery	Edit *Delivery*	*Black Box*
Payment	Edit *Payment*	*Black Box*
FAQ	Edit *FAQ*	*Black Box*
Cara Pembayaran	Edit Cara Pembayaran	*Black Box*
Cara Menjadi Anggota	Edit Cara Menjadi Anggota	*Black Box*
Cara Konfimasi Pengiriman	Edit Cara Konfirmasi Pengiriman	*Black Box*
Kategori	Tambah Kategori	*Black Box*
	Edit Kategori	*Black Box*

	Hapus Kategori	*Black Box*
	Cari Kategori	*Black Box*
Produk	Tambah Produk	*Black Box*
	Edit Produk	*Black Box*
	Tambah Detail Produk	*Black Box*
	Detail Produk	*Black Box*
Diskon	Tambah Diskon	*Black Box*
	Edit Diskon	*Black Box*
	Cari Diskon	*Black Box*
Daftar Harga	Cari Daftar Harga	*Black Box*
Penjualan	Detail Penjualan	*Black Box*
	Cari Penjualan	*Black Box*
Pembayaran	Detail Pembayaran	*Black Box*
	Cari Pembayaran	*Black Box*
Pengiriman	Tambah Pengiriman	*Black Box*
	Cari Pengiriman	*Black Box*
Retur	Detail Retur	*Black Box*
	Cari Retur	*Black Box*
Laporan Penjualan	Pembuatan Laporan Penjualan	*Black Box*
Laporan Pengiriman	Pembuatan Laporan Pengiriman	*Black Box*

Laporan Stok	Pembuatan Laporan Stok	*Black Box*
Laporan Retur	Cetak Laporan Retur	*Black Box*
	Cari Laporan Retur	*Black Box*
Master Agama	Tambah Agama	*Black Box*
	Edit Agama	*Black Box*
	Hapus Agama	*Black Box*
	Cari Agama	*Black Box*
Master Propinsi	Tambah Propinsi	*Black Box*
	Edit Propinsi	*Black Box*
	Hapus Propinsi	*Black Box*
	Cari Propinsi	*Black Box*
Master Kota/Kab	Tambah Kota/Kab	*Black Box*
	Edit Kota/Kab	*Black Box*
	Hapus Kota/Kab	*Black Box*
	Cari Kota/Kab	*Black Box*
Master Kecamatan	Tambah Kecamatan	*Black Box*
	Edit Kecamatan	*Black Box*
	Hapus Kecamatan	*Black Box*
	Cari Kecamatan	*Black Box*
Master Kelurahan	Tambah Kelurahan	*Black Box*

	Edit Kelurahan	*Black Box*
	Hapus Kelurahan	*Black Box*
	Cari Kelurahan	*Black Box*
Vendor Pengiriman	Tambah Vendor Pengiriman	*Black Box*
	Edit Vendor Pengiriman	*Black Box*
	Hapus Vendor Pengiriman	*Black Box*
	Cari Vendor Pengiriman	*Black Box*
Tarif Pengiriman	Tambah Tarif Pengiriman	*Black Box*
	Edit Tarif Pengiriman	*Black Box*
	Hapus Tarif Pengiriman	*Black Box*
	Cari Tarif Pengiriman	*Black Box*
Register *Member*	Pendaftaran sebagai *member*	*Black Box*
Ubah Akun	Mengubah akun *member*	*Black Box*
Shopping Cart	Tambah data produk ke *Shopping Cart*	*Black Box*
Cari Produk	Menampilkan data produk yang dicari	*Black Box*

4.2.2 Pengujian Alpha

Hasil implementasi perangkat lunak dari aplikasi *E-Commerce* yang dibangun harus diuji terlebih dahulu agar sistem yang dibangun dapat berjalan dengan baik. Pengujian perangkat lunak sistem penerapan *E-Commerce* pada toko Mix menggunakan data uji berdasarkan *form* yang terdapat didalam aplikasi.

4.2.2.1 Pengujian aplikasi

Pengujian yang dilakukan adalah menguji semua proses yang ada di halaman website diantaranya adalah sebagai berikut :

1. Pengujian *login*

   Pengujian *login* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian *login* dapat dilihat pada tabel 4.10.

Tabel 4.10 Pengujian login

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Login berupa *Email* : admin *Password* : admin	Data login dicari dalam *database*, *Email* dan data *password* benar login *valid* dan dapat masuk ke menu utama	*email* dan *password* benar dapat masuk ke menu utama	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Memasukan data *Email* dan	Tidak dapat login/masuk dan	Muncul pesan "*email* atau	[√] Diterima

Password yang tidak sesuai	menampilkan pesan kesalahan	*password* salah"	[ ] Ditolak
Memasukan data *Email* atau *Password* yang tidak terdaftar	Tampilkan pesan kesalahan "*email* atau *password* yang Anda masukkan salah"	Dapat menampilkan pesan kesalahan "*email* atau *password* yang Anda masukkan salah"	[√] Diterima [ ] Ditolak
Memasukkan *Email* atau *Password* yang belum aktif	Tampilkan pesan kesalahan "*email* atau *password* yang Anda masukkan salah"	Dapat menampilkan pesan kesalahan "*email* atau *password* yang Anda masukkan salah"	[√] Diterima [ ] Ditolak

2. Pengujian Tambah *Groups*

Pengujian tambah *groups* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah *groups* dilihat pada tabel 4.11.

Tabel 4.11 Pengujian Tambah *Groups*

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah *Groups*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Tambah *groups*	Menampilkan *form* tambah *groups*.	Sistem berhasil menampilkan *form* tambah *groups*.	[√] Diterima [ ] Ditolak
Mengisi data *groups* contoh : Nama : petugas Deskripsi : petugas 1 Status : aktif	Tercantum dalam masing-masing *field* kemudian data *groups* tersimpan dalam *database*.	Data *groups* berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah *Groups*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data *groups* dengan mengosongkan nama	Data *groups* tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

3. Pengujian Edit *Groups*

   Pengujian edit *groups* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit *groups* dilihat pada tabel 4.12.

Tabel 4.12 Pengujian Edit Groups

Kasus dan Hasil Uji Benar (Data Benar)			
Edit *Groups*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit *Groups*	Menampilkan *form* edit *groups* beserta data yang akan diedit.	Dapat menampilkan *form* edit *groups* beserta data yang akan diedit.	[√] Diterima [ ] Ditolak
Data *groups* lama yang akan diedit contoh : Nama : petugas Deskripsi : petugas 1 Status : aktif	Tercantum dalam *field groups* berupa nama, deskripsi, dan status.	Dapat menampilkan *field* data *groups* yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data *groups* contoh : Nama : petugas 2 Deskripsi : petugas 2 Status : aktif	Dapat mengisi *field* data yang akan diedit dengan benar dan data *groups* berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data *groups* berhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			

Edit *Groups*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data groups yang akan diedit Nama : - Deskripsi : - Status : -	Data *groups* tidak berhasil diedit karena nama, deskripsi dan status belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

4. Pengujian Hapus *Groups*

Pengujian hapus *groups* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus *groups* dilihat pada tabel 4.13.

Tabel 4.13 Pengujian Hapus *Groups*

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus *groups*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *groups* yang akan dihapus contoh : Nama : petugas Deskripsi : petugas Status : aktif	Tercantum data *groups* yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data *groups* yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

5. Pengujian Cari *Groups*

Pengujian cari *groups* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari *groups* dilihat pada tabel 4.14.

Tabel 4.14 Pengujian Cari *Groups*

Kasus dan Hasil Uji Benar (Data Benar)			
Cari *Groups*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *groups*	Menampilkan data *groups*	Berhasil menampilkan data *groups*	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama *groups* contoh : admin	Menampilkan data *groups* berdasarkan nama *groups* yang dicari	Berhasil Menampilkan data *groups* berdasarkan nama *groups* yang dicari	[√] Diterima [ ] Ditolak

6. Pengujian Tambah *Users*

Pengujian tambah *users* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah *users* dilihat pada tabel 4.15.

Tabel 4.15 Tambah *Users*

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah *Users*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah *users*	Menampilkan *form* tambah *users*.	Sistem berhasil menampilkan *form* tambah *users*.	[√] Diterima [ ] Ditolak
Mengisi data *users*	Tercantum dalam masing-masing *field* kemudian data *users* tersimpan dalam *database*.	Data *users* berhasil disimpan dalam *database*.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah *users*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data *users* dengan mengosongkan nama *users*	Data *users* tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data *users* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

7. Pengujian Edit *Users*

Pengujian edit *users* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit *users* dilihat pada tabel 4.16.

Tabel 4.16 Pengujian Edit *Users*

Kasus dan Hasil Uji Benar (Data Benar)			
Edit *Users*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah *users*	Menampilkan *form* ubah *users* beserta data yang akan diubah	Dapat menampilkan *form* ubah users beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data *users* lama yang akan diubah contoh : Nama : Budi	Tercantum dalam *field users* berupa nama.	Dapat menampilkan *field* data *users* yang akan diubah.	[√] Diterima [ ] Ditolak
Mengisi data *users* contoh : Nama : Asep	Dapat mengisi *field* data yang akan diubah dengan benar dan data *users* berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data *users* berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			

Edit *Users*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *users* yang akan diubah Nama : -	Data *users* tidak berhasil diubah karena nama tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

8. Pengujian Cari *Users*

Pengujian cari *users* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari *users* dilihat pada tabel 4.17.

Tabel 4.17 Pengujian Cari *Users*

Kasus dan Hasil Uji Benar (Data Benar)			
Cari *Users*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *users*	Menampilkan data *users*	Berhasil menampilkan data *users*	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama *users* contoh :	Menampilkan data *users* berdasarkan nama *users* yang dicari	Berhasil Menampilkan data *users* berdasarkan nama *users* yang dicari	[√] Diterima [ ] Ditolak

admin			

9. Pengujian Tambah *Member*

Pengujian tambah *member* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah *member* dilihat pada tabel 4.18.

Tabel 4.18 Pengujian Tambah *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah *Member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah *member*	Menampilkan *form* tambah *member*.	Sistem berhasil menampilkan *form* tambah *member*.	[√] Diterima [ ] Ditolak
Mengisi data *users*	Tercantum dalam masing-masing *field* kemudian data *member* tersimpan dalam *database*.	Data *member* berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah *Member*			
Data Masukan	Yang	Pengamatan	Kesimpulan

	diharapkan		
Mengisi data *member* dengan mengosongkan nama *member*	Data *member* tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data *member* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

10. Pengujian Edit *Member*

Pengujian edit *member* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit *member* dilihat pada tabel 4.19.

Tabel 4.19 Pengujian Edit *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Edit *Member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit *member*	Menampilkan *form* edit *member* beserta data yang akan diedit	Dapat menampilkan *form* edit *member* beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data *member* lama yang akan diedit	Tercantum dalam *field member* berupa	Dapat menampilkan *field* data	[√] Diterima [ ] Ditolak

contoh : Nama : Santi	nama.	*member* yang akan diedit.	
Mengisi data *member* contoh : Nama : Sinta	Dapat mengisi *field* data yang akan diedit dengan benar dan data *member* berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data *member* berhasil diedit dan disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit *Member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *member* yang akan diedit Nama : -	Data *member* tidak berhasil diedit karena nama tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

11. Pengujian Cari *Member*

Pengujian cari *member* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari *member* dilihat pada tabel 4.20.

Tabel 4.20 Pengujian Cari *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Cari *Member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *member*	Menampilkan data *member*	Berhasil menampilkan data *member*	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama *member* contoh : admin	Menampilkan data *member* berdasarkan nama *member* yang dicari	Berhasil Menampilkan data *member* berdasarkan nama *member* yang dicari	[√] Diterima [ ] Ditolak

12. Pengujian Edit *Web_Config*

Pengujian edit *web_config* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit *web_config* dilihat pada tabel 4.21.

Tabel 4.21 Pengujian Edit *Web_Config*

Kasus dan Hasil Uji Benar (Data Benar)			
Edit *Web_Config*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit *web_config*	Menampilkan form edit	Dapat menampilkan	[√] Diterima

	web_config beserta data yang akan diedit	form edit *web_config* beserta data yang akan diedit	[ ] Ditolak
Data *web_config* lama yang akan diedit contoh : Nama : Mix Store	Tercantun dalam *field web_config* berupa nama.	Dapat menampilkan field data *web_config* yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data *web_config* contoh : Nama : Mix Store Online	Dapat mengisi *field* data yang akan diedit dengan benar dan data *web_config* berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data *web_config*berhasil diedit dan disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit *Web_Config*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *web_config* yang akan diedit Nama : -	Data *web_config* tidak berhasil diedit karena nama tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

13. Pengujian Hapus *Web_Config*

Pengujian hapus *web_config* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus *web_config* dilihat pada tabel 4.22.

Tabel 4.22 Pengujian Hapus *Web_Config*

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus *Web_Config*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *web_config* yang akan dihapus contoh : Nama : Mix Store Slogan : Menjadi Terbaik Telp : 0853467678	Tercantum data *web_config* yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data *web_config* yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

14. Pengujian Cari *Web_Config*

Pengujian cari *web_config* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari *web_config* dilihat pada tabel 4.23.

Tabel 4.23 Pengujian Cari *Web_Config*

Kasus dan Hasil Uji Benar (Data Benar)			
Cari *Web_Config*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *web_config*	Menampilkan data *web_config*	Berhasil menampilkan data *web_config*	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama *web_config* contoh : *Mix Store*	Menampilkan data *web_config* berdasarkan nama *web_config* yang dicari	Berhasil Menampilkan data *web_config* berdasarkan nama *web_config* yang dicari	[√] Diterima [ ] Ditolak

15. Pengujian Edit *About_Us*

Pengujian edit *about_us* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit *about_us* dilihat pada tabel 4.24.

Tabel 4.24 Pengujian Edit *About_Us*

Kasus dan Hasil Uji Benar (Data Benar)			
Edit *About_Us*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Edit *about_us*	Menampilkan *form* edit *about_us* beserta data yang akan diedit	Dapat menampilkan *form* edit *about_us* beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data *about_us* lama yang akan diedit contoh : *Title* : About_Us	Tercantum dalam *field about_us* berupa *title*.	Dapat menampilkan *field* data *about_us* yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data *about_us* contoh : *Title* : *About_Us Mix Store*	Dapat mengisi *field* data yang akan diedit dengan benar dan data *about_us* berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data *about_us* berhasil diedit dan disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit *About_Us*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *about_us* yang akan diedit *Title* : -	Data *about_us* tidak berhasil diedit karena *title* tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan	[√] Diterima [ ] Ditolak

		kesalahan	

16. Pengujian Edit *Delivery*

Pengujian edit *delivery* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit *delivery* dilihat pada tabel 4.25.

Tabel 4.25 Pengujian Edit *Delivery*

Kasus dan Hasil Uji Benar (Data Benar)			
Edit *Delivery*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit *delivery*	Menampilkan form edit *delivery* beserta data yang akan diedit	Dapat menampilkan form edit *delivery* beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data *delivery* lama yang akan diedit contoh : *Title* : *Delivery*	Tercantum dalam *delivery* berupa *title.*	Dapat menampilkan field data d*elivery* yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data *Delivery* contoh : *Title* : *Delivery Mix*	Dapat mengisi *field* data yang akan diedit dengan benar dan data *delivery*	Dapat mengisi *field* data yang akan diedit dengan benar dan data	[√] Diterima [ ] Ditolak

Store	berhasil diedit dan disimpan dalam *database*.	*delivery* berhasil diedit dan disimpan dalam *database*	
Kasus dan Hasil Uji Salah (Data Salah)			
Edit *Delivery*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *delivery* yang akan diedit *Title* : -	Data *delivery* tidak berhasil diedit karena *title* tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

17. Pengujian Edit *Payment*

Pengujian edit *payment* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit *payment* dilihat pada tabel 4.26.

Tabel 4.26 Pengujian Edit *Payment*

Kasus dan Hasil Uji Benar (Data Benar)			
Edit *Payment*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit *payment*	Menampilkan *form* edit	Dapat menampilkan	[√] Diterima

	payment beserta data yang akan diedit	*form* edit *payment* beserta data yang akan diedit	[ ] Ditolak
Data *payment* lama yang akan diedit contoh : Title : *Payment*	Tercantum dalam *payment* berupa *title*.	Dapat menampilkan field data *payment* yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data *Payment* contoh : title : *Payment Mix Store*	Dapat mengisi *field* data yang akan diedit dengan benar dan data *payment* berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data *payment* berhasil diedit dan disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit *Payment*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *payment* yang akan diedit *Title* : -	Data *payment* tidak berhasil diedit karena *title* tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

18. Pengujian Edit *FAQ*

Pengujian edit *FAQ* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit *FAQ* dilihat pada tabel 4.27.

Tabel 4.27 Pengujian Edit FAQ

Kasus dan Hasil Uji Benar (Data Benar)			
Edit *FAQ*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit *FAQ*	Menampilkan *form* edit *FAQ* beserta data yang akan diedit	Dapat menampilkan *form* edit *FAQ* beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data *FAQ* lama yang akan diedit contoh : *Title* : *FAQ*	Tercantum dalam *FAQ* berupa *title.*	Dapat menampilkan *field* data *FAQ* yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data *FAQ* contoh : *Title* : *FAQ Mix Store*	Dapat mengisi *field* data yang akan diedit dengan benar dan data *FAQ* berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data *FAQ* berhasil diedit dan disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			

Edit *FAQ*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *FAQ* yang akan diedit *Title* : -	Data *FAQ* tidak berhasil diedit karena *title* tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

19. Pengujian Edit Cara Pembayaran

Pengujian edit cara pembayaran dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit cara pembayaran dilihat pada tabel 4.28.

Tabel 4.28 Pengujian Edit Cara Pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Cara Pembayaran			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Cara Pembayaran	Menampilkan *form* edit Cara Pembayaran beserta data yang akan diedit	Dapat menampilkan *form* edit Cara Pembayaran beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data Cara Pembayaran lama	Tercantum dalam Cara	Dapat menampilkan	[√] Diterima

yang akan diedit contoh : *Title* : Cara Pembayaran	Pembayaran berupa *title*.	*field* data Cara Pembayaran yang akan diedit.	[ ] Ditolak
Mengisi data Cara Pembayaran contoh : *Title* : Cara Pembayaran *Mix Store*	Dapat mengisi *field* data yang akan diedit dengan benar dan data Cara Pembayaran berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data Cara Pembayaran berhasil diedit dan disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Cara Pembayaran			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Cara Pembayaran yang akan diedit *Title* : -	Data Cara Pembayaran tidak berhasil diedit karena *title* tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

20. Pengujian Edit Cara Menjadi Anggota

Pengujian edit cara menjadi anggota dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit cara menjadi anggota dilihat pada tabel 4.29.

Tabel 4.29 Pengujian Edit Cara Menjadi Anggota

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Cara Menjadi Anggota			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Cara Menjadi Anggota	Menampilkan *form* edit Cara Menjadi Anggota beserta data yang akan diedit	Dapat menampilkan *form* edit Cara Menjadi Anggota beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data Cara Menjadi Anggota lama yang akan diedit contoh : *Title* : Cara Menjadi Anggota	Tercantum dalam Cara Menjadi Anggota berupa *title*.	Dapat menampilkan *field* data Cara Menjadi Anggota yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data Cara Menjadi Anggota contoh : *Title* : Cara Menjadi	Dapat mengisi *field* data yang akan diedit dengan benar dan data Cara	Dapat mengisi *field* data yang akan diedit dengan benar dan data Cara	[√] Diterima [ ] Ditolak

Anggota *Mix Store*	Menjadi Anggota berhasil diedit dan disimpan dalam *database*.	Menjadi Anggota berhasil diedit dan disimpan dalam *database*	
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Cara Menjadi Anggota			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Cara Menjadi Anggota yang akan diedit *Title* : -	Data Cara Menjadi Anggota tidak berhasil diedit karena title tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

21. Pengujian Edit Cara Konfirmasi Pengiriman

Pengujian edit cara konfirmasi pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit cara konfirmasi pengiriman dilihat pada tabel 4.30.

Tabel 4.30 Pengujian Edit Cara Konfirmasi Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Cara Konfirmasi Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Cara Konfirmasi	Menampilkan	Dapat	[√] Diterima

Pengiriman	*form* edit Cara Konfirmasi Pengiriman beserta data yang akan diedit	menampilkan *form* edit Cara Konfirmasi Pengiriman beserta data yang akan diedit	[ ] Ditolak
Data Cara Konfirmasi Pengiriman lama yang akan diedit contoh : *Title* : Cara Konfirmasi Pengiriman	Tercantum dalam Cara Konfirmasi Pengiriman berupa title.	Dapat menampilkan *field* data Cara Konfirmasi Pengiriman yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data Cara Konfirmasi Pengirman contoh : *Title* : Cara Konfirmasi Pengiriman *Mix Store*	Dapat mengisi *field* data yang akan diedit dengan benar dan data Cara konfirmasi Pengiriman berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data Cara Konfirmasi Pengiriman berhasil diedit dan disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Cara Konfirmasi Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Data Cara Konfirmasi Pengiriman yang akan diedit *Title* : -	Data Cara Konfirmasi Pengiriman tidak berhasil diedit karena title tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

22. Pengujian Tambah Kategori

Pengujian tambah kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah kategori dilihat pada tabel 4.31.

Tabel 4.31 Pengujian Tambah Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Kategori	Menampilkan *form* tambah kategori.	Sistem berhasil menampilkan *form* tambah kategori.	[√] Diterima [ ] Ditolak
Mengisi data kategori contoh : Nama : Raglan Deskripsi : Kaos	Tercantum dalam masing-masing *field* kemudian data kategori tersimpan dalam	Data kategori berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak

Raglan	*database*.		
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori dengan mengosongkan nama	Data *kategori* tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data *kategori* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

23. Pengujian Edit Kategori

Pengujian edit kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit kategori dilihat pada tabel 4.32.

Tabel 4.32 Pengujian Edit kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Kategori	Menampilkan *form* edit kategori beserta data yang akan	Dapat menampilkan *form* edit kategori beserta	[√] Diterima [ ] Ditolak

	diedit	data yang akan diedit	
Data kategori lama yang akan diedit contoh : Nama : Raglan Deskripsi : Kaos Raglan	Tercantum dalam *field* kategori berupa nama, dan deskripsi.	Dapat menampilkan *field* data kategori yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data kategori contoh : Nama : Kaos Kerah Deskripsi : Kaos Kerah Belang	Dapat mengisi *field* data yang akan diedit dengan benar dan data kategori berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data kategori berhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategori yang akan diedit Nama : - Deskripsi : -	Data kategori tidak berhasil diedit karena nama, dan deskripsi belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

24. Pengujian Hapus Kategori

Pengujian hapus kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus kategori dilihat pada tabel 4.33.

Tabel 4.33 Pengujian Hapus Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus Kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategori yang akan dihapus contoh : Nama : Raglan Deskripsi : Kaos Raglan	Tercantum data kategori yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data kategori yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

25. Pengujian Cari Kategori

Pengujian cari kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari kategori dilihat pada tabel 4.34.

Tabel 4.34 Pengujian Cari Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Kategori	Menampilkan data kategori	Berhasil menampilkan data kategori	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama kategori	Menampilkan data kategori berdasarkan nama kategori yang dicari	Berhasil Menampilkan data kategori berdasarkan nama kategori yang dicari	[√] Diterima [ ] Ditolak

26. Pengujian Tambah Produk

Pengujian tambah produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah produk dilihat pada tabel 4.35.

Tabel 4.35 Pengujian Tambah Produk

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Produk	Menampilkan	Sistem berhasil	[√] Diterima

	form tambah produk.	menampilkan *form* tambah produk.	[ ] Ditolak
Mengisi data produk contoh : Kode : 001 Nama : Raglan Abu-Hitam Deskripsi : Kaos Raglan Abu-Hitam Kategori : Raglan Status Produk : Stok	Tercantum dalam masing-masing *field* kemudian data produk tersimpan dalam *database*.	Data produk berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data produk dengan mengosongkan nama	Data produk tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data produk*i* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

27. Pengujian Edit Produk

Pengujian edit produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit Produk dilihat pada tabel 4.36.

Tabel 4.36 Pengujian Edit Produk

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Produk	Menampilkan form edit produk beserta data yang akan diedit	Dapat menampilkan form edit produk beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data produk lama yang akan diedit contoh : Kode : 001 Nama : Raglan Abu-Hitam Deskripsi : Kaos Raglan Abu-Hitam Kategori : Raglan Status Produk : Stok	Tercantum dalam *field* produk berupa kode, nama, deskripsi, dan kategori.	Dapat menampilkan field data produk yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data produk contoh :	Dapat mengisi *field* data yang akan diedit	Dapat mengisi *field* data yang akan diedit	[√] Diterima [ ] Ditolak

Kode : 001 nama : Raglan Abu-Hitam deskripsi : Kaos Raglan Abu-Hitam Kategori : Raglan Status Produk : Stok	dengan benar dan data produk berhasil diedit dan disimpan dalam *database*.	dengan benar dan data produk berhasil diedit dan disimpan dalam *databsase*	
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data produk yang akan diedit Kode : - Nama : - Deskripsi : Kaos Raglan Abu-Hitam Kategori : Raglan Status Produk : Stok	Data produk tidak berhasil diedit karena kode, dan nama belum di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

28. Pengujian Tambah Detail Produk

Pengujian tambah detail produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah detail produk dilihat pada tabel 4.37.

Tabel 4.37 Pengujian Tambah Detail Produk

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Detail Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Detail Produk	Menampilkan *form* tambah Detail produk.	Sistem berhasil menampilkan *form* tambah detail produk.	[√] Diterima [ ] Ditolak
Mengisi data detail produk contoh : Nama : Kaos Euro Warna : Hitam Stok : 12 Status Produk : Stok Ukuran : m Harga : 35000	Tercantum dalam masing-masing *field* kemudian data detail produk tersimpan dalam *database*.	Data detail produk berhasil disimpan dalam *database*.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Detail Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Mengisi data detail produk dengan mengosongkan nama	Data detail produk tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data detail produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

29. Pengujian Detail Produk

Pengujian detail produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian detail produk dilihat pada tabel 4.38.

Tabel 4.38 Pengujian Detail Produk

Kasus dan Hasil Uji Benar (Data Benar)			
Detail Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Produk	Menampilkan Detail Produk yang dipilih	Dapat menampilkan detail produk yang diplih	[√] Diterima [ ] Ditolak

30. Pengujian Tambah Diskon

Pengujian tambah diskon dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah diskon dilihat pada tabel 4.39.

Tabel 4.39 Pengujian Tambah Diskon

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Diskon			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Diskon	Menampilkan *form* tambah diskon.	Sistem berhasil menampilkan *form* tambah diskon.	[√] Diterima [ ] Ditolak
Mengisi data diskon contoh : Kode Produk : 001 Diskon : 10 Tanggal Awal : 12-07-2012 Tanggal Akhir : 19-07-2012 Min-Quantity : 1	Tercantum dalam masing-masing *field* kemudian data diskon tersimpan dalam *database*.	Data diskon berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Diskon			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data diskon dengan mengosongkan kode	Data diskon tidak dapat disimpan dalam	Sistem tidak dapat menyimpan data	[√] Diterima [ ] Ditolak

produk	*database* karena data kode produk belum terisi	diskon ke dalam *database* dan muncul pesan kesalahan	

31. Pengujian Edit Diskon

Pengujian edit diskon dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit diskon dilihat pada tabel 4.40.

Tabel 4.40 Pengujian Edit Diskon

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Diskon			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Diskon	Menampilkan *form* edit diskon beserta data yang akan diedit	Dapat menampilkan *form* edit diskon beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data kategori lama yang akan diedit contoh : Kode Produk : 001 Diskon : 10 Tanggal Awal : 12-07-2012 Tanggal Akhir : 19-	Tercantum dalam *field* diskon berupa kode produk, diskon, tanggal awal, tanggal akhir, dan min quantity.	Dapat menampilkan *field* data diskon yang akan diedit.	[√] Diterima [ ] Ditolak

07-2012 Min-Quantity : 1			
Mengisi data diskon contoh : Kode Produk : 002 Diskon : 20 Tanggal Awal : 12-07-2012 Tanggal Akhir : 19-07-2012 Min-Quantity : 1	Dapat mengisi *field* data yang akan diedit dengan benar dan data diskon berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data diskon berhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Diskon			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategori yang akan diedit Kode Produk : -	Data kategori tidak berhasil diedit karena kode produk belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

32. Pengujian Cari Diskon

Pengujian cari diskon dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari diskon dilihat pada tabel 4.41.

Tabel 4.41 Pengujian Cari Diskon

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Diskon			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Diskon	Menampilkan data diskon	Berhasil menampilkan data diskon	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan kode produk	Menampilkan data diskon berdasarkan kode produk yang dicari	Berhasil Menampilkan data diskon berdasarkan kode produk yang dicari	[√] Diterima [ ] Ditolak

33. Pengujian Cari Daftar Harga

Pengujian cari daftar harga dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari daftar harga dilihat pada tabel 4.42.

Tabel 4.42 Pengujian Cari Daftar Harga

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Daftar Harga			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Daftar	Menampilkan data	Berhasil menampilkan data	[√] Diterima

Harga	daftar harga	daftar harga	[ ] Ditolak
Data yang akan dicari berdasarkan nama produk	Menampilkan data daftar harga berdasarkan nama produk yang dicari	Berhasil Menampilkan data daftar harga berdasarkan nama produk yang dicari	[√] Diterima [ ] Ditolak

34. Pengujian Detail Penjualan

Pengujian detail penjualan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian detail penjualan dilihat pada tabel 4.43.

Tabel 4.43 Pengujian Detail Penjualan

Kasus dan Hasil Uji Benar (Data Benar)			
Detail Penjualan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Penjualan	Menampilkan Detail Penjualan yang dipilih	Dapat menampilkan detail penjualan yang diplih	[√] Diterima [ ] Ditolak

35. Pengujian Cari Penjualan

Pengujian cari penjualan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari penjualan dilihat pada tabel 4.44.

Tabel 4.44 Pengujian Cari Penjualan

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Penjualan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Penjualan	Menampilkan data daftar harga	Berhasil menampilkan data penjualan	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama	Menampilkan data penjualan berdasarkan nama penjualan yang dicari	Berhasil Menampilkan data penjualan berdasarkan nama penjualan yang dicari	[√] Diterima [ ] Ditolak

36. Pengujian Detail Pembayaran

Pengujian detail pembayaran dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian detail pembayaran dilihat pada tabel 4.45.

Tabel 4.45 Pengujian Detail Pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Detail Pembayaran			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Pembayaran	Menampilkan Detail Pembayaran yang dipilih	Dapat menampilkan detail pembayaran yang diplih	[√] Diterima [ ] Ditolak

37. Pengujian Cari Pembayaran

Pengujian cari pembayaran dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari pembayaran dilihat pada tabel 4.46.

Tabel 4.46 Pengujian Cari Pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Pembayaran			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Pembayaran	Menampilkan data pembayaran	Berhasil menampilkan data pembayaran	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan	Menampilkan data pembayaran berdasarkan nama	Berhasil Menampilkan data pembayaran	[√] Diterima [ ] Ditolak

nama	pembayaran yang dicari	berdasarkan nama pembayaran yang dicari	

38. Pengujian Tambah Pengiriman

Pengujian tambah pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah pengiriman dilihat pada tabel 4.47.

Tabel 4.47 Pengujian Tambah Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Pengiriman	Menampilkan *form* tambah pengiriman.	Sistem berhasil menampilkan *form* tambah pengiriman.	[√] Diterima [ ] Ditolak
Mengisi data pengiriman contoh : Kode Pengirman : 001 Tanggal Pengirman : 17-07-2012 Status Pengiriman :	Tercantum dalam masing-masing *field* kemudian data pengiriman tersimpan dalam *database*.	Data pengiriman berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak

mengirim			
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data produk dengan mengosongkan kode pengiriman	Data pengiriman tidak dapat disimpan dalam *database* karena data kode pengiriman belum terisi	Sistem tidak dapat menyimpan data pengiriman ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

39. Pengujian Cari Pengiriman

Pengujian cari pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari pengiriman dilihat pada tabel 4.48.

Tabel 4.48 Pengujian Cari Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Pengiriman	Menampilkan data pengiriman	Berhasil menampilkan data pengiriman	[√] Diterima [ ] Ditolak

Data yang akan dicari berdasarkan nama pengiriman	Menampilkan data pengiriman berdasarkan nama pengiriman yang dicari	Berhasil Menampilkan data pengiriman berdasarkan nama pengirman yang dicari	[√] Diterima [ ] Ditolak

40. Pengujian Detail Retur

Pengujian detail retur dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian detail retur dilihat pada tabel 4.49.

Tabel 4.49 Pengujian Detail Retur

Kasus dan Hasil Uji Benar (Data Benar)			
Detail Retur			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Retur	Menampilkan Detail Retur yang dipilih	Dapat menampilkan detail retur yang diplih	[√] Diterima [ ] Ditolak

41. Pengujian Cari Retur

Pengujian cari retur dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari retur dilihat pada tabel 4.50.

Tabel 4.50 Pengujian Cari Retur

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Retur			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Retur	Menampilkan data retur	Berhasil menampilkan data retur	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan No KTP	Menampilkan data pembayaran berdasarkan No KTP yang dicari	Berhasil Menampilkan data retur berdasarkan No KTP yang dicari	[√] Diterima [ ] Ditolak

42. Pengujian Pembuatan Laporan Penjualan

Pengujian pembuatan laporan penjualan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian pembuatan laporan penjualan dilihat pada tabel 4.51.

Tabel 4.51 Pengujian Pembuatan Laporan Penjualan

Kasus dan Hasil Uji Benar (Data Benar)			
Pembuatan Laporan Penjualan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Cetak laporan Penjualan	Menampilkan *form* laporan penjualan	Dapat menampilkan *form* laporan	[√] Diterima [ ] Ditolak

		penjualan	
Memasukan bulan dan tahun contoh : Bulan : Juli Tahun : 2012	Tercantun dalam *field* laporan berupa bulan dan tahun	Dapat menampilkan data laporan pada periode yang dipilih.	[√] Diterima [ ] Ditolak
Proses laporan	Menampilkan laporan penjualan pada periode tertentu dalam bentuk *file*.	Dapat menampilkan laporan penjualan pada periode tertentu dalam benuk *file*.	[√] Diterima [ ] Ditolak

43. Pengujian Pembuatan Laporan Pengiriman

Pengujian pembuatan laporan pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian pembuatan laporan pengiriman dilihat pada tabel 4.52.

Tabel 4.52 Pengujian Pembuatan Laporan Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Pembuatan Laporan Penjualan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Cetak laporan Pengiriman	Menampilkan *form* laporan penngiriman	Dapat menampilkan *form* laporan pengiriman	[√] Diterima [ ] Ditolak

Memasukan bulan dan tahun contoh : Bulan : Juli Tahun : 2012	Tercantun dalam *field* laporan berupa bulan dan tahun	Dapat menampilkan data laporan pada periode yang dipilih.	[√] Diterima [ ] Ditolak
Proses laporan	Menampilkan laporan pengiriman pada periode tertentu dalam bentuk *file*.	Dapat menampilkan laporan pengiriman pada periode tertentu dalam benuk *file.*	[√] Diterima [ ] Ditolak

44. Pengujian Pembuatan Laporan Stok

Pengujian pembuatan laporan stok dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian pembuatan laporan stok dilihat pada tabel 4.53.

Tabel 4.53 Pengujian Pembuatan Laporan Stok

Kasus dan Hasil Uji Benar (Data Benar)			
Pembuatan Laporan Stok			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Cetak laporan Stok	Menampilkan *form* laporan stok	Dapat menampilkan *form* laporan stok	[√] Diterima [ ] Ditolak
Memasukan bulan dan tahun	Tercantun dalam *field* laporan berupa	Dapat menampilkan data	[√] Diterima

contoh : Bulan : Juli Tahun : 2012	bulan dan tahun	laporan pada periode yang dipilih.	[ ] Ditolak
Proses laporan	Menampilkan laporan stok pada periode tertentu dalam bentuk *file*.	Dapat menampilkan laporan stok pada periode tertentu dalam benuk *file*.	[√] Diterima [ ] Ditolak

45. Pengujian Tambah Agama

Pengujian tambah agama dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah agama dilihat pada tabel 4.54.

Tabel 4.54 Pengujian Tambah Agama

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Agama			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah agama	Menampilkan *form* tambah agama.	Sistem berhasil menampilkan *form* tambah agama.	[√] Diterima [ ] Ditolak
Mengisi data agama contoh :	Tercantum dalam masing-masing *field*	Data agama berhasil disimpan dalam	[√] Diterima [ ] Ditolak

Nama : islam Deskripsi : agama islam	kemudian data agama tersimpan dalam *database*.	*database*.	
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Agama			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data agama dengan mengosongkan nama	Data agama tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data agama ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

46. Pengujian Edit Agama

Pengujian edit agama dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit agama dilihat pada tabel 4.55.

Tabel 4.55 Pengujian Edit Agama

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Agama			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit agama	Menampilkan *form* edit agama beserta data	Dapat menampilkan *form* edit agama	[√] Diterima [ ] Ditolak

	yang akan diedit	beserta data yang akan diedit	
Data agama lama yang akan diedit contoh : Nama : islam Deskripsi : agama islam	Tercantum dalam *field* agama berupa nama, dan deskripsi.	Dapat menampilkan *field* data agama yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data agama contoh : Nama : kristen Deskripsi : agama kristen	Dapat mengisi *field* data yang akan diedit dengan benar dan data agama berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data agama berhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Agama			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data agama yang akan diedit Nama : - Deskripsi : -	Data agama tidak berhasil diedit karena nama, dan deskripsi belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

47. Pengujian Hapus Agama

Pengujian hapus agama dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus agama dilihat pada tabel 4.22.

Tabel 4.56 Pengujian Hapus Agama

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus Agama			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data agama yang akan dihapus contoh : Nama : kristen Deskripsi : agama kristen	Tercantum data agama yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data agama yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

48. Pengujian Cari Agama

Pengujian cari agama dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari agama dilihat pada tabel 4.57.

Tabel 4.57 Pengujian Cari Agama

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Agama			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Agama	Menampilkan data agama	Berhasil menampilkan data agama	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama agama	Menampilkan data agama berdasarkan nama agama yang dicari	Berhasil Menampilkan data agama berdasarkan nama agama yang dicari	[√] Diterima [ ] Ditolak

49. Pengujian Tambah Propinsi

Pengujian tambah propinsi dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah propinsi dilihat pada tabel 4.58.

Tabel 4.58 Pengujian Tambah Propinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Propinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Propinsi	Menampilkan *form* tambah	Sistem berhasil menampilkan	[√] Diterima

	propinsi.	*form* tambah propinsi.	[ ] Ditolak
Mengisi data propinsi contoh : Nama : Jawa Barat	Tercantum dalam masing-masing *field* kemudian data propinsi tersimpan dalam *database*.	Data propinsi berhasil disimpan dalam *database*.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Propinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data propinsi dengan mengosongkan nama	Data propinsi tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data propinsi ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

50. Pengujian Edit Propinsi

Pengujian edit propinsi dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit propinsi dilihat pada tabel 4.59.

Tabel 4.59 Pengujian Edit Propinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Propinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit propinsi	Menampilkan *form* edit propinsi beserta data yang akan diedit	Dapat menampilkan *form* edit propinsi beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data propinsi lama yang akan diedit contoh : Nama : Jawa Barat	Tercantun dalam *field* propinsi berupa nama.	Dapat menampilkan *field* data propinsi yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data propinsi contoh : Nama : Jawa Timur	Dapat mengisi *field* data yang akan diedit dengan benar dan data propinsi berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data propinsi berhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Propinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Data agama yang akan diedit Nama : -	Data propinsi tidak berhasil diedit karena nama, belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

51. Pengujian Hapus Propinsi

Pengujian hapus propinsi dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus propinsi dilihat pada tabel 4.60.

Tabel 4.60 Pengujian Hapus Propinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus Propinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data agama yang akan dihapus contoh : Nama : Jawa Tengah	Tercantum data propinsi yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data propinsi yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

52. Pengujian Cari Propinsi

Pengujian cari propinsi dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari propinsi dilihat pada tabel 4.61.

Tabel 4.61 Pengujian Cari Propinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Propinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Propinsi	Menampilkan data propinsi	Berhasil menampilkan data propinsi	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama propinsi	Menampilkan data propinsi berdasarkan nama propinsi yang dicari	Berhasil Menampilkan data propinsi berdasarkan nama propinsi yang dicari	[√] Diterima [ ] Ditolak

53. Pengujian Tambah Kota/Kab

Pengujian tambah kota/kab dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah kota/kab dilihat pada tabel 4.62.

Tabel 4.62 Pengujian Kota/Kab

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Kota/Kab			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Kota/Kab	Menampilkan *form* tambah kota/kab.	Sistem berhasil menampilkan *form* tambah kota/kab.	[√] Diterima [ ] Ditolak
Mengisi data kota/kab contoh : Nama : Bandung Propinsi : Jawa Barat	Tercantum dalam masing-masing *field* kemudian data kota/kab tersimpan dalam *database*.	Data kota/kab berhasil disimpan dalam *database*.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Kota/Kab			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kota/kab dengan mengosongkan nama	Data kota/kab tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data kota/kab ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

54. Pengujian Edit Kota/Kab

Pengujian edit kota/kab dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit kota/kab dilihat pada tabel 4.63.

Tabel 4.63 Pengujian Edit Kota/Kab

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Kota/Kab			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit kota/kab	Menampilkan *form* edit kota/kab beserta data yang akan diedit	Dapat menampilkan *for*m edit kota/kab beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data kota/kab lama yang akan diedit contoh : Nama : Bandung Propinsi : Jawa Barat	Tercantun dalam *field* kota/kab berupa nama dan propinsi.	Dapat menampilkan *field* data kota/kab yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data kota/kab contoh : Nama : Tasik Propinsi : Jawa Barat	Dapat mengisi *field* data yang akan diedit dengan benar dan data kota/kab berhasil diedit dan	Dapat mengisi *field* data yang akan diedit dengan benar dan data kota/kab berhasil diedit	[√] Diterima [ ] Ditolak

	disimpan dalam *database*.	dan disimpan dalam *databsase*	
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Kota/Kab			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kota/kab yang akan diedit Nama : - Propinsi : Jawa Barat	Data kota/kab tidak berhasil diedit karena nama, belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

55. Pengujian Hapus Kota/Kab

Pengujian hapus kota/kab dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus kota/kab dilihat pada tabel 4.64.

Tabel 4.64 Pengujian Hapus Kota/Kab

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus Kota/Kab			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Data kota/kab yang akan dihapus contoh : Nama : Bandung Propinsi : Jawa Barat	Tercantum data kota/kab yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data kota/kab yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

56. Pengujian Cari Kota/Kab

Pengujian cari kota/kab dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari kota/kab dilihat pada tabel 4.65.

Tabel 4.65 Pengujian Cari Kota/Kab

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Kota/Kab			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Kota/Kab	Menampilkan data kota/kab	Berhasil menampilkan data propinsi	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama kota/kab	Menampilkan data kota/kab berdasarkan nama kota/kab yang dicari	Berhasil Menampilkan data kota/kab berdasarkan nama kota/kab yang	[√] Diterima [ ] Ditolak

		dicari	

57. Pengujian Tambah Kecamatan

Pengujian tambah kecamatan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah kecamatan dilihat pada tabel 4.66.

Tabel 4.66 Pengujian Kecamatan

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Kecamatan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Kecamatan	Menampilkan *form* tambah kecamatan.	Sistem berhasil menampilkan *form* tambah kecamatan.	[√] Diterima [ ] Ditolak
Mengisi data kecamatan contoh : nama : Sukaraya propinsi : Jawa Barat kota : Bandung	Tercantum dalam masing-masing *field* kemudian data kecamatan tersimpan dalam *database*.	Data kecamatan berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Kecamatan			

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kecamatan dengan mengosongkan nama	Data kecamatan tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data kecamatan ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

58. Pengujian Edit Kecamatan

Pengujian edit kecamatan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit kecamatan dilihat pada tabel 4.21.

Tabel 4.67 Pengujian Edit Kecamatan

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Kecamatan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit kecamatan	Menampilkan *form* edit kecamatan beserta data yang akan diedit	Dapat menampilkan *form* edit kecamatan beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data kecamatan lama yang akan diedit	Tercantun dalam *field* kecamatan	Dapat menampilkan	[√] Diterima

contoh : Nama : Sukaraya Propinsi : Jawa Barat Kota : Bandung	berupa nama, propinsi, dan kota.	*field* data kecamatan yang akan diedit.	[ ] Ditolak
Mengisi data kecamatan contoh : Nama : Cimenyan Propisni : Jawa Barat Kota : Bandung	Dapat mengisi *field* data yang akan diedit dengan benar dan data kecamatan berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data kecamatan berhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Kecamatan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kecamatan yang akan diedit Nama : - Propinsi : Jawa Barat Kota : Bandung	Data kecamatan tidak berhasil diedit karena nama, belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

59. Pengujian Hapus Kecamatan

Pengujian hapus kecamatan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus kecamatan dilihat pada tabel 4.68.

Tabel 4.68 Pengujian Hapus Kecamatan

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus Kecamatan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kecamatan yang akan dihapus contoh : Nama : Cimenyan Propinsi : Jawa Barat Kota : Bandung	Tercantum data kecamatan yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data kota/kab yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

60. Pengujian Cari Kecamatan

Pengujian cari kecamatan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari kecamatan dilihat pada tabel 4.69.

Tabel 4.69 Pengujian Cari Kecamatan

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Kecamatan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Kecamatan	Menampilkan data kecamatan	Berhasil menampilkan data kecamatan	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama kecamatan	Menampilkan data kecamatan berdasarkan nama kecamatan yang dicari	Berhasil Menampilkan data kecamatan berdasarkan nama kecamatan yang dicari	[√] Diterima [ ] Ditolak

61. Pengujian Tambah Kelurahan

Pengujian tambah kelurahan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah kelurahan dilihat pada tabel 4.70.

Tabel 4.70 Pengujian Tambah Kelurahan

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Kelurahan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Kelurahan	Menampilkan	Sistem berhasil	[√] Diterima

	form tambah kelurahan.	menampilkan *form* tambah kelurahan.	[ ] Ditolak
Mengisi data kelurahan contoh : Nama : Sari Asih Propinsi : Jawa Barat Kota : Bandung Kecamatan : Cimenyan	Tercantum dalam masing-masing *field* kemudian data kelurahan tersimpan dalam *database*.	Data kelurahan berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Kelurahan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kelurahan dengan mengosongkan nama	Data kelurahan tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data kecamatan ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

62. Pengujian Edit Kelurahan

Pengujian edit kelurahan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit kelurahan dilihat pada tabel 4.71.

Tabel 4.71 Pengujian Edit Kelurahan

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Kelurahan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit kelurahan	Menampilkan *form* edit kelurahan beserta data yang akan diedit	Dapat menampilkan *form* edit kelurahan beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data kelurahan lama yang akan diedit contoh : Nama : Sari Asih Propinsi : Jawa Barat Kota : Bandung Kecamatan : Cimenyan	Tercantun dalam *field* kelurahan berupa nama, propinsi, kota, dan kecamatan.	Dapat menampilkan *field* data kelurahan yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data kelurahan	Dapat mengisi *field* data yang akan diedit	Dapat mengisi *field* data yang akan diedit	[√] Diterima [ ] Ditolak

contoh : Nama : Ciwaruga Propisni : Jawa Barat Kota : Bandung Kecamatan : Cimenyan	dengan benar dan data kelurahan berhasil diedit dan disimpan dalam *database*.	dengan benar dan data kelurahan berhasil diedit dan disimpan dalam *databsase*	
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Kelurahan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kelurahan yang akan diedit Nama : - Propinsi : Jawa Barat Kota : Bandung Kecamatan : Cimenyan	Data kelurahan tidak berhasil diedit karena nama, belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

63. Pengujian Hapus Kelurahan

Pengujian hapus kelurahan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus kelurahan dilihat pada tabel 4.72.

Tabel 4.72 Pengujian Hapus Kelurahan

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus Kelurahan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kelurahan yang akan dihapus contoh : Nama : Ciwaruga Propinsi : Jawa Barat Kota : Bandung Kecamatan : Cimenyan	Tercantum data kelurahan yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data kelurahan yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

64. Pengujian Cari Kelurahan

Pengujian cari kelurahan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari kelurahan dilihat pada tabel 4.73.

Tabel 4.73 Pengujian Cari Kelurahan

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Kelurahan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Data Kelurahan	Menampilkan data kelurahan	Berhasil menampilkan data kelurahan	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama kelurahan	Menampilkan data kelurahan berdasarkan nama kelurahan yang dicari	Berhasil Menampilkan data kelurahan berdasarkan nama kelurahan yang dicari	[√] Diterima [ ] Ditolak

65. Pengujian Tambah Vendor Pengiriman

Pengujian tambah vendor pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah vendor pengiriman dilihat pada tabel 4.74.

Tabel 4.74 Pengujian Tambah Vendor Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Kelurahan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Vendor Pengiriman	Menampilkan *form* tambah vendor pengiriman.	Sistem berhasil menampilkan *form* tambah vendor pengiriman.	[√] Diterima [ ] Ditolak
Mengisi data vendor pengiriman	Tercantum dalam masing-	Data vendor pengiriman	[√] Diterima

contoh : nama : JNE kode : JE01 deskripsi : Jasa Pengiriman JNE	masing *field* kemudian data vendor pengiriman tersimpan dalam *database*.	berhasil disimpan dalam *database.*	[ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Vendor Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data vendor pengiriman dengan mengongkan nama	Data vendor pengiriman tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data vendor pengiriman ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

66. Pengujian Vendor Pengiriman

Pengujian edit vendor pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit vendor pengiriman dilihat pada tabel 4.75.

Tabel 4.75 Pengujian Edit Vendor Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Vendor Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Vendor Pengiriman	Menampilkan *form* edit vendor pengiriman beserta data yang akan diedit	Dapat menampilkan *form* edit vendor pengiriman beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data vendor pengiriman lama yang akan diedit contoh : Nama : JNE Kode : JE01 Deskripsi : Jasa Pengirman JNE	Tercantum dalam *field* vendor pengiriman berupa nama, kode, dan deskripsi.	Dapat menampilkan *field* data vendor pengiriman yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data vendor pengiriman contoh : Nama : Tiki Kode : TK01 Deskripsi : Jasa Pengirman Tiki	Dapat mengisi *field* data yang akan diedit dengan benar dan data vendor pengirman berhasil diedit dan disimpan	Dapat mengisi *field* data yang akan diedit dengan benar dan data vendor pengiriman berhasil diedit dan disimpan	[√] Diterima [ ] Ditolak

	dalam *database*.	dalam *databsase*	
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Vendor Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data vendor pengiriman yang akan diedit Nama : - Kode : TK01 Deskripsi : Jasa Pengirman Tiki	Data vendor pengiriman tidak berhasil diedit karena nama, belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

67. Pengujian Hapus Vendor Pengiriman

Pengujian hapus vendor pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus vendor pengiriman dilihat pada tabel 4.76.

Tabel 4.76 Pengujian Hapus Vendor Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus Vendor Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Data vendor pengiriman yang akan dihapus contoh : Nama : Tiki Kode : TK01 Deskripsi : Jasa Pengirman Tiki	Tercantum data vendor pengiriman yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data vendor pengiriman yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

68. Pengujian Cari Vendor Pengiriman

Pengujian cari vendor pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari vendor pengiriman dilihat pada tabel 4.77.

Tabel 4.77 Pengujian Cari Vendor Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Vendor Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Vendor Pengiriman	Menampilkan data kelurahan	Berhasil menampilkan data kelurahan	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama vendor	Menampilkan data vendor pengiriman berdasarkan nama vendor pengiriman	Berhasil Menampilkan data vendor pengiriman berdasarkan nama	[√] Diterima [ ] Ditolak

pengiriman	yang dicari	vendor pengiriman yang dicari	

69. Pengujian Tambah Tarif Pengiriman

Pengujian tambah tarif pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah tarif pengiriman dilihat pada tabel 4.78.

Tabel 4.78 Pengujian Tambah Tarif Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Tarif Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah Tarif Pengiriman	Menampilkan *form* tambah tarif pengiriman.	Sistem berhasil menampilkan *form* tambah tarif pengiriman.	[√] Diterima [ ] Ditolak
Mengisi data tarif pengiriman pengiriman contoh : Vendor : Tiki Nama : REG Kode : TK01 Tujuan : Bandung -	Tercantum dalam masing-masing *field* kemudian data tarif pengiriman tersimpan dalam *database*.	Data tarif pengiriman berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak

Surabaya Tarif : 11000 Pola Perhitungan : Flat			
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Tarif Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data tarif pengiriman dengan mengosongkan vendor	Data tarif pengiriman tidak dapat disimpan dalam *database* karena data vendor belum terisi	Sistem tidak dapat menyimpan data tarif pengiriman ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

70. Pengujian Edit Tarif Pengiriman

Pengujian edit tarif pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit tarif pengiriman dilihat pada tabel 4.79.

Tabel 4.79 Pengujian Edit Tarif Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Tarif Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Edit Tarif Pengiriman	Menampilkan *form* edit tarif pengiriman beserta data yang akan diedit	Dapat menampilkan *form* edit tarif pengiriman beserta data yang akan diedit	[√] Diterima [ ] Ditolak
Data tarif pengiriman lama yang akan diedit contoh : Vendor : Tiki Nama : REG Kode : TK01 Tujuan : Bandung - Surabaya Tarif : 11000 Pola Perhitungan : Flat	Tercantum dalam *field* tarif pengiriman berupa vendors, nama, kode, tujuan, tarif, dan pola perhitungan.	Dapat menampilkan *field* data tarif pengiriman yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data tarif pengiriman contoh :	Dapat mengisi *field* data yang akan diedit dengan benar	Dapat mengisi *field* data yang akan diedit dengan benar	[√] Diterima [ ] Ditolak

Vendor : JNE Nama : YES Kode : TK01 Tujuan : Bandung - Surabaya Tarif : 11000 Pola Perhitungan : Flat	dan data tarif pengirman berhasil diedit dan disimpan dalam *database*.	dan data tarif pengiriman berhasil diedit dan disimpan dalam *databsase*	
Kasus dan Hasil Uji Salah (Data Salah)			
Edit Vendor Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data tarif pengiriman yang akan diedit Vendor : JNE Nama : - Kode : TK01 Tujuan : Bandung - Surabaya Tarif : 11000 Pola Perhitungan : Flat	Data tarif pengiriman tidak berhasil diedit karena nama, belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

71. Pengujian Hapus Tarif Pengiriman

Pengujian hapus tarif pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus tarif pengiriman dilihat pada tabel 4.80.

Tabel 4.80 Pengujian Hapus Tarif Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus Tarif Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data tarif pengiriman yang akan dihapus contoh : Vendor : Tiki Nama : REG Kode : TK01 Tujuan : Bandung - Surabaya Tarif : 11000 Pola Perhitungan : Flat	Tercantum data tarif pengiriman yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data tarifpengiriman yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

72. Pengujian Cari Tarif Pengiriman

Pengujian cari tarif pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari tarif pengiriman dilihat pada tabel 4.81.

Tabel 4.81 Pengujian Cari Tarif Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Tarif Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Tarif Pengiriman	Menampilkan data kelurahan	Berhasil menampilkan data kelurahan	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama tarif pengiriman	Menampilkan data tarif pengiriman berdasarkan nama tarif pengiriman yang dicari	Berhasil Menampilkan data tarif pengiriman berdasarkan nama vendor pengiriman yang dicari	[√] Diterima [ ] Ditolak

73. Pengujian Register *Member*

Pengujian register *member* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian register *member* dilihat pada tabel 4.82.

Tabel 4.82 Pengujian Register *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Pendaftaran Register *Member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Register *Member*	Menampilkan *form* register *member*	Dapat menampilkan *form* daftar member	[√] Diterima [ ] Ditolak
Masukan data register, contoh masukan : Nama : Budi Santoso *Email* : budisantoso@yahoo.com Password : 12345 Kota : Bandung Telepon : 0856789023 Konfirmasi *Password* : 12345	Sistem menyimpan data register kedalam *database*	Berhasil menyimpan data register member kedalam database	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji Salah (Data Salah)			
Register *member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Masukan data pendaftaran, contoh masukan : Nama : Budi Santoso Email : budisantoso@yahoo.com password : 12345 Kota : Bandung Telepon : 0856789023 Konfirmasi *Password* : 54321	Tampil pesan kesalahan *field password* dengan field *password* konfirmasi tidak sama	Dapat pesan kesalahan *field password* dengan field *password* konfirmasi tidak sama	[√] Diterima [ ] Ditolak
Masukan data pendaftaran *password* tidak diisi	Tampil pesan kesalahan *password* belum diisi	Sistem tidak dapat menyimpan data dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Masukan data pendaftaran nama tidak diisi	Tampil pesan kesalahan nama belum diisi	Sistem tidak dapat menyimpan data dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

74. Pengujian Ubah Akun

Pengujian ubah akun dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah akun dilihat pada tabel 4.83.

Tabel 4.83 Pengujian Ubah Akun

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah akun			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah akun	Menampilkan *form* ubah akun beserta data yang akan diubah	Dapat menampilkan *for*m ubah akun beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data akun lama yang akan diubah No KTP : 12352342 Nama : Budi Santoso	Tercantun dalam *field* akun berupa *username*, *password*, nama, *email*, jenis-kelamin, kota, telepon	Dapat menampilkan *field* data akun yang akan diubah.	[√] Diterima [ ] Ditolak

Agama : Islam Telepon : 0856789023 Alamat: Jl. Setrasari No.1 Propinsi : Jawa Barat Kota : Bandung Kecamatan : Harapan Indah Kelurahan : Cimenyan *Email* : budisantoso@yahoo.com Password : 12345 Konfirmasi *Password* : 12345			
Mengisi data akun contoh : No KTP : 12352342 Nama : Budi	Dapat mengisi *field* data yang akan diubah dengan benar dan data akun berhasil diubah dan disimpan dalam	Dapat mengisi *field* data yang akan diubah dengan benar dan data akun berhasil diubah	[√] Diterima [ ] Ditolak

Santoso Agama : Islam Telepon : 0856789023 Alamat: Jl. Setrasari No.1 Propinsi : Jawa Barat Kota : Bandung Kecamatan : Harapan Indah Kelurahan : Cimenyan *Email* : budisantoso@yahoo.com *Password* : 12345 Konfirmasi *Password* : 12345	*database*.	dan disimpan dalam *databsase*	
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah Akun			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Mengisi data akun contoh : No KTP : 12352342 Nama : Budi Santoso Agama : Islam Telepon : 0856789023 Alamat: Jl. Setrasari No.1 Propinsi : Jawa Barat Kota : Bandung Kecamatan : Harapan Indah Kelurahan : Cimenyan *Email* : budisantoso@yahoo.com Password : 12345 Konfirmasi *Password* :	Data akun tidak berhasil diubah dan menampilkan pesan kesalahan, password dengan password konfirmasi tidak sama	Sistem tidak dapat menyimpan data akun dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

54321			
Masukan data *email* tidak diisi	Tampil pesan kesalahan *email* belum diisi	Sistem tidak dapat menyimpan data dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data nama tidak diisi	Tampil pesan kesalahan nama belum diisi	Sistem tidak dapat menyimpan data dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

75. Pengujian Tambah Produk ke *Shoping Cart*

Pengujian tambah produk ke *shoping cart* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah produk ke *shoping cart* dilihat pada tabel 4.84.

Tabel 4.84 Pengujian Tambah Produk ke *Shoping Cart*

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Produk ke *Shoping Cart*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Detail produk	Menampilkan detail produk yang akan dibeli	Sistem menampilkan detail produk yang	[√] Diterima [ ] Ditolak

		akan dibeli	
membeli produk dengan memasukan data produk contoh : produk : Kaos Polos Hitam Tangan Panjang	produk masuk kedalam keranjang belanja.	Sistem berhasil memasukan data produk kedalam *shoping cart*.	[√] Diterima [ ] Ditolak

76. Pengujian Hapus Data Produk dari *Shopping Cart*

Pengujian hapus data produk dari *shopping cart* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus data produk dari *shoping cart* dilihat pada tabel 4.85.

Tabel 4.85 Pengujian Hapus Data Produk dari *Shoping Cart*

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus Data Produk dari *Shoping Cart*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data produk dalam *shoping cart* yang akan dihapus : contoh : Produk : Kaos	Tercantum data produk dalam *shoping cart* yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan	Tercantum data produk dalam *shoping cart* yang akan dihapus berupa setelah memilih hapus maka tampil pesan	[√] Diterima [ ] Ditolak

Polos Hitam Tangan Panjang	apakah data akan dihapus	peringatan apakah data akan dihapus	

77. Pengujian Cari Produk

Pengujian cari produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari produk dilihat pada tabel 4.86.

Tabel 4.86 Pengujian Cari Produk

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data produk	Menampilkan data produk	Berhasil menampilkan data produk	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama produk contoh : Kaos Kerah	Menampilkan data produk berdasarkan nama produk yang dicari	Berhasil Menampilkan data produk berdasarkan nama produk yang dicari	[√] Diterima [ ] Ditolak

4.2.2.2 Kesimpulan Hasil Pengujian Alpha

Berdasarkan hasi pengujian alpha (fungsional) dengan kasus uji diatas dapat ditarik kesimpulan bahwa pembangunan *E-Commerce* di toko Mix secara fungsional sistem sudah dapat digunakan dan menghasilkan *output* yang diharapkan.

4.2.3 Pengujian Beta

Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung di tempat penelitian dengan membagikan kuesioner kepada 30 orang dari masyarakat umum dan melakukan wawancara kepada pengguna sistem atau petugas di toko Mix. Penelitian ini dilakukan di dua lingkungan yaitu :

1. Lingkungan masyarakat umum

   Penelitian di lingkungan pada masyarakat umum khususnya yang sudah terbiasa dengan teknologi internet, dilakukan agar dapat mengetahui sejauh mana sistem yang dibangun dapat memberikan kemudahan masyarakat umum untuk melakukan transaksi pembelian barang secara online. Penelitian ini ditujukan kepada member dengan menggunakan teknik kuesioner.

2. Lingkungan toko Mix

   Penelitian di lingkungan Toko Mix dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang ada. Penelitian ini ditujukan kepada super admin, *owner*, petugas input data, dan keuangan dengan menggunakan teknik wawancara.

Berdasarkan data hasil kusioner tersebut, dapat dicari presentase 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 *presentase* masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuesioner ini diujikan kepada 10 orang.

4.2.3.1 Kuesioner Member

Kuesioner untuk member terdiri dari 7 pertanyaan (contoh kuesioner dapat diliihat pada lampiran) dengan menggunakan rumus Y=P/Q*100% , dengan pertanyaan sebagai berikut :

1. Apakah anda mudah mendapatkan informasi barang beserta detailnya melalui *website* toko Mix?

Tabel 4.87 Hasil kuesioner pertanyaan nomor 1 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	15	50
	2	Setuju	10	33.33
	3	Cukup Setuju	5	16.66
	4	Biasa Saja	-	-
	5	Tidak Setuju	-	-
Jumlah			30	100

Berdasarkan hasil prosentase pada tabel 4.87 maka dapat disimpulkan sebanyak 15 atau 50% menyatakan sangat mudah, 10 atau 33.33% menyatakan

mudah dan 5 atau 16.66% menyatakan cukup mudah untuk mendapatkan informasi barang beserta detail *website* toko Mix.

2. Apakah anda setuju bahwa *website* ini mudah dioperasikan dan menarik ?

Tabel 4.88 Hasil kuesioner pertanyaan nomor 2 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	18	60
	2	Setuju	8	26.66
	3	Cukup Setuju	4	13.33
	4	Biasa Saja	-	-
	5	Tidak Setuju	-	-
Jumlah			30	100

Berdasarkan hasil prosentase pada tabel 4.88 maka dapat disimpulkan sebanyak 18 orang atau 60% menyatakan sangat setuju, 8 orang atau 26.66% menyatakan setuju dan 4 orang atau 13.33% menyatakan cukup setuju saja bahwa tampilan dari website *E-Commerce* pada toko Mix menarik dan *user friendly*.

3. Apakah anda setuju bahwa *website* ini lebih memudahkan dalam melakukan transaksi jual-beli barang khususnya dari tempat yang berjauhan dengan toko?

Tabel 4.89 Hasil kuesioner pertanyaan nomor 3 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	21	70
	2	Setuju	4	13.33
	3	Cukup Setuju	4	13.33
	4	Biasa Saja	1	3.33
	5	Tidak Setuju	-	-
Jumlah			30	100

Berdasarkan hasil prosentase pada tabel 4.89 maka dapat disimpulkan sebanyak 21 orang atau 70% menyatakan sangat setuju 4 orang atau 13.33% menyatakan setuju, 4 orang atau 13.33% menyatakan cukup setuju dan 1 orang atau 3.33% menyatakan biasa saja bahwa *website* ini lebih memudahkan dalam melakukan transaksi jual-beli barang khususnya dari tempat yang berjauhan dengan toko.

4. Apakah anda setuju bahwa sistem pembayaran *online* yang digunakan lebih memudahkan dalam proses transaksi ?

Tabel 4.90 Hasil kuesioner pertanyaan nomor 4 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
4	1	Sangat Setuju	13	43.33
	2	Setuju	14	46.66
	3	Cukup Setuju	3	10
	4	Biasa Saja	-	-
	5	Tidak Setuju	-	-
Jumlah			30	100

Berdasarkan hasil prosentase pada tabel 4.90 maka dapat disimpulkan sebanyak 13 orang atau 43.33% menyatakan sangat setuju, 14 orang atau 46.66% menyatakan setuju dan 3 orang atau 10% menyatakan cukup setuju bahwa sistem pembayaran *online* yang digunakan lebih memudahkan dalam proses transaksi.

5. Apakah anda setuju bahwa informasi produk yang diberikan oleh *website* toko Mix ini sudah sesuai dengan informasi yang anda butuhkan?

Tabel 4.91 Hasil kuesioner pertanyaan nomor 5 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
5	1	Sangat Setuju	17	56.66
	2	Setuju	2	6.66
	3	Cukup Setuju	11	36.66

	4	Biasa Saja	-	-
	5	Tidak Setuju	-	-
Jumlah			30	100

Berdasarkan hasil prosentase pada tabel 4.91 maka dapat disimpulkan sebanyak 17 orang atau 56.66% menyatakan sangat setuju, 11 orang atau 36.66% menyatakan setuju, 2 orang atau 6.66% menyatakan setuju bahwa informasi produk *website* ini sesuai dengan informasi yang dibutuhkan.

6. Apakah anda setuju sistem pembayaran yang digunakan *website* toko Mix dapat memudahkan anda dalam melakukan transaksi?

Tabel 4.92 Hasil kuesioner pertanyaan nomor 6 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
6	1	Sangat Setuju	16	53.33
	2	Setuju	11	36.66
	3	Cukup Setuju	3	10
	4	Biasa Saja	-	-
	5	Tidak Setuju	-	-
Jumlah			30	100

Berdasarkan hasil prosentase pada tabel 4.92 maka dapat disimpulkan sebanyak 16 orang atau 53.33% menyatakan sangat setuju, 11 orang atau 36.66%

menyatakan setuju, 3 orang atau 10% menyatakan cukup setuju bahwa sistem pembayaran yang digunakan *website* toko Mix dapat memudahkan anda dalam melakukan transaksi.

7. Apakah anda setuju bahwa *website* toko Mix ini telah layak untuk di-*online*-kan?

Tabel 4.93 Hasil kuesioner pertanyaan nomor 7 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
7	1	Sangat Setuju	17	56.66
	2	Setuju	11	36.66
	3	Cukup Setuju	2	6.66
	4	Biasa Saja	-	-
	5	Tidak Setuju	-	-
Jumlah			30	100

Berdasarkan hasil prosentase pada tabel 4.93 maka dapat disimpulkan sebanyak 17 orang atau 56.66% menyatakan sangat setuju, 11 orang atau 36.66% menyatakan setuju, 2 orang atau 6.66% menyatakan setuju *website* toko Mix ini telah layak untuk di-*online*-kan

4.2.3.2 Wawancara Super Admin

Wawancara dilakukan kepada petugas yang telah menggunakan sistem dan melakukan proses-proses yang ada dalam sistem tersebut. Wawancara dilakukan

dengan mengajukan 4 pertanyaan kepada petugas toko atau super admin, hasil dari wawancara dapat dilihat sebagai berikut :

1. Apakah dengan adanya *website e-commerce* ini dapat memudahkan dalam melakukan promosi produk di toko Mix ?

   Jawaban:

   Website yang dibangun dapat membantu dalam mempromosikan produk kepada masyarakat luas khususnya kepada para pengguna internet.

2. Apakah halaman administrator dari *website* ini mudah dipahami dan digunakan ?

   Jawaban:

   Secara keseluruhan, halaman administrator mudah untuk dipahami dan digunakan untuk melakukan proses-proses pengolahan data hingga pembuatan laporan.

3. Apakah *website* ini dapat memudahkan dalam mengelola data transaksional Toko Mix ?

   Jawaban:

   Data transaksional lebih mudah untuk dikelola dan proses produksi yang dilakukan lebih mudah untuk dikontrol.

4. Apakah aplikasi yang dibangun dapat memudahkan dalam membuat laporan penjualan produk ?

   Jawaban:

   Laporan lebih mudah untuk dibuat sehingga mudah untuk dicek dan mengurangi kesalahan yang terjadi.

4.2.3.2 Wawancara Owner

Wawancara dilakukan kepada pemilik toko atau *owner* yang telah menggunakan sistem dan melakukan proses-proses yang ada dalam sistem tersebut. Wawancara dilakukan dengan mengajukan 4 pertanyaan kepada *owner*, hasil dari wawancara dapat dilihat sebagai berikut :

1. Apakah aplikasi yang dibuat mudah digunakan?

   Jawaban:

   Sejauh ini saya tidak mengalami kesulitan dalam menggunakan aplikasi tersebut.

2. Dengan mengguanakan aplikasi ini, apa dalam pemasaran produk memiliki waktu yang lebih efektif?

   Jawaban:

   Ya, karena pemasaran produk dapat dipasarkan dalam waktu 24 jam dikarenakan secara online.

3. Apakah aplikasi yang dibangun sudah layak untuk di *online*-kan?

   Jawaban:

   Sudah layak, karena sejauh ini belum melihat masalah yang berarti pada sistem tersebut.

4. Apakah *website E-Commerce* yang dibuat sudah dapat memenuhi kebutuhan bagi pemilik toko?

Jawaban:

Sudah, karena sistem ini dapat mengelola proses transaksi dari awal pelanggan melakukan pembelian hingga proses akhir yaitu pembuatan laporan.

4.2.3.3 Wawancara Petugas Input Data

Wawancara dilakukan kepada petugas input data yang telah menggunakan sistem dan melakukan proses-proses yang ada dalam sistem tersebut. Wawancara dilakukan dengan mengajukan 4 pertanyaan kepada petugas input data, hasil dari wawancara dapat dilihat sebagai berikut :

1. Apakah tampilan *website E-Commerce* cukup menarik dan *user friendly?*

   Jawaban:

   Tampilan *website* ini sangat user friendly karena tidak berbeda jauh dengan tampilan website *E-Commerce* lainnya.
2. Apakah mengalami kesulitan dalam mengoperasikan *E-Commerce* ini?

   Jawaban:

   Menurut saya pribadi tidak begitu sulit asalkan kita sudah paham mengenai cara pengoperasiannya.
3. Apakah anda setuju bahwa dengan adanya *website E-Commerce* ini akan memudahkan dalam mengelola data *member* dan data transaksi?

   Jawaban:

   Iya, karena menurut saya sistem ini dapat mengelola semua proses tersebut.
4. Menurut anda apakah *website E-Commerce* ini bisa berguna untuk *member* yang ingin membeli produk toko Mix?

Jawaban:

Sangat berguna, terutama untuk member yang berada di luar kota yang ingin membeli produk kami.

4.2.3.4 Wawancara Petugas Keuangan

Wawancara dilakukan kepada petugas keuangan yang telah menggunakan sistem dan melakukan proses-proses yang ada dalam sistem tersebut. Wawancara dilakukan dengan mengajukan 4 pertanyaan kepada petugas keuangan, hasil dari wawancara dapat dilihat sebagai berikut :

1. Apakah aplikasi yang dibangun memiliki tampilan yang menarik bagi penggunanya?

   Jawaban:

   Dari segi tampilan menurut saya cukup menarik dan tampilannya umum seperti *E-Commerce* lainnya, sehingga tidak ada kesulitan dalam menjalankannya.

2. Apakah *website E-Commerce* yang dibuat sudah dapat memenuhi kebutuhan bagi petugas keuangan?

   Jawaban:

   Dari segi laporan sudah cukup baik dan dapat memenuhi kebutuhan bagian keuangan.

3. Apakah anda menemukan kesulitan pada saat menggunakan *website E-Commerce* ini?

Jawaban:

Menurut pendapat saya pribadi tidak begitu sulit asalkan kita paham mengenai cara penggunaannya.

4. Apakah *website E-Commerce* ini dapat dijadikan alternatif sebagai media pemasaran yang baik?

   Jawaban:

   Iya, karena lewat *E-Commerce* biaya pemasarannya tidak begitu mahal jika dibandingkan dengan media pemasaran lainnya.dan cakupan wilayah pemasarannya bisa lebih luas.

4.2.3.3 Kesimpulan Pengujian Beta

Berdasarkan hasil presentase diatas yang didapatkan dari pengujian beta, maka sistem penerapan *E-Commerce* untuk sistem pemesanan ini sudah sesuai dengan tujuan yaitu dapat membantu mempromosikan produk, mengelola data pemesanan dan memudahkan pembuatan laporan pada toko Mix.