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* di Mag & Shoes.

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 Implementasi Perangkat Keras

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* Standar
7	*Mouse*	*Mouse* Standar
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 Implementasi Perangkat Lunak

No	Perangkat Keras	Spesifikasi
1	Sistem operasi	Windows XP
2	Web Browser	Mozilla firefox, Opera, Internet Explorer, dan Google Chrome

4.1.4 Implementasi Basis Data

Implementasi data dalam aplikasi *E-Commerce* pada Mag & Shoes 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 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;
```

2. 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;
```

3. Table 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;
```

4. Table 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;
```

5. Table detail_produk

Tabeldetail_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,
  `berat` int(10) 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;
```

6. Table 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;
```

7. Table 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';
```

8. Table 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;
```

9. Table 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;
```

10. Table 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;
```

11. Table 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;
```

12. Table 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;
```

13. Table 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,
  `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_kelurahan_member` (`kelurahan_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

14. Table 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,
  `total` int(11) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `FK_pembayaran_pembelian` (`pembelian_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

15. Table pembelian

Tabel pembelian digunakan untuk menyimpan data pembelian.

```
CREATE TABLE IF NOT EXISTS `pembelian` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `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`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

16. Table 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;
```

17. Table 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;
```

18. Table provinsi

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;
```

19. Table 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;
```

20. Table 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;
```

21. Table 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;
```

22. Table 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;
```

23. Table 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;
```

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 operator pada aplikasi E-Commerce di Mag & Shoes dapat dilihat pada tabel 4.3

Tabel 4. 3 Implementasi Antarmuka Operator

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
Web_config	Halaman *web_config* digunakan untuk melihat, mencari, mengubah dan menghapus data *web_config*.	pages/web_config.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	discounts/diskon.php

	melihat, menambah, mencari, dan mengubah data diskon.	
Daftar Harga	Halaman daftar harga digunakan untuk melihat, dan mencari data daftar harga.	price_lists/daftar_harga.php
Data Master: - Master Propinsi - Master Kota/Kab - Master Kecamatan - Master Kelurahan	Halaman yang digunakan untuk mengolah data master 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.	 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 Kasir dapat dilihat seperti pada tabel 4.4 berikut ini:

Tabel 4. 4 Implementasi Antarmuka Kasir

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
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 Retur	Halaman laporan retur digunakan untuk melihat, mencari, dan mencetak data laporan retur.	laporan/laporan_retur.php

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

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

Tabel 4. 5 Implementasi Antarmuka 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 *users* digunakan untuk melihat, mencari, dan mengubah data *member*.	member/member.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	laporan/laporan_pengiriman.php

	pengiriman.	
Laporan Retur	Halaman laporan retur digunakan untuk melihat, mencari, dan mencetak data laporan retur.	laporan/laporan_retur.php
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

Tabel implementasi antarmuka member dapat dilihat seperti pada tabel 4.6 berikut ini:

Tabel 4. 6 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 lengkap perusahaan.	frontend/contactus.php
FAQ	Halaman yang berisi mengenai penjelasan aturan-aturan yang terkait dengan proses transaksi,	frontend/static_pages/10.php

	seperti cara pembelian, pembayaran, pengiriman, dll.	
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 4.7 berikut ini:

Tabel 4. 7 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 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
Login	Halaman login untuk *member* yang sudah terdaftar dan aktif.	frontend/login.php
Register	Halaman untuk pengunjung yang	frontend/register.php

	ingin mendaftar menjadi *member* baru.	
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 Mag & Shoes menggunakan data uji berdasarkan data dari Mag & shoes. Rencana pengujian selengkapnya terlihat pada tabel 4.8.

Tabel 4. 8 Rencana Pengujian Sistem

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Login	Validasi *email* dan *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	Cari Member	*Black Box*
	Edit Member	*Black Box*
Web_config	Edit *Web_config*	*Black Box*
	Hapus *Web_config*	*Black Box*
	Cari *Web_config*	*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	Pembuatan Laporan Retur	*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*
Registrasi *Member*	Pendaftaran sebagai *member*	*Black Box*
Aktivasi SSO	Mengaktivasi 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 di Mag & Shoes 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 user

Pengujian login dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian login dapat dilihat pada tabel 4.9.

Tabel 4. 9 Pengujian login user

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Login berupa *Email* : admin@localhost.com *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)			
Mengosongkan data *Email* dan *Password*	Tidak dapat login/masuk dan menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*email* atau *password* yang Anda masukkan salah"	[√] Diterima [ ] Ditolak
Memasukan data *Email* dan *Password* yang tidak sesuai	Tidak dapat login/masuk dan menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*email* atau *password* yang Anda masukkan salah"	[√] Diterima [ ] Ditolak
Memasukan data *Email* atau *Password* yang tidak terdaftar	Tidak dapat login/masuk dan menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*email* atau *password* yang Anda masukkan salah"	[√] Diterima [ ] Ditolak

Memasukkan *Email* atau *Password* yang belum aktif	Tidak dapat login/masuk dan menampilkan pesan kesalahan	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 4.10.

Tabel 4. 10 Pengujian tambah groups

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah *Groups*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
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 dan deskripsi Contoh : Nama : - Deskripsi : - Status : Aktif	Data *groups* tidak dapat disimpan dalam *database* karena data nama dan deskripsi belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data *groups*dengan mengosongkan nama Contoh : Nama : - Deskripsi : operator Status : Aktif	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

Mengisi data *groups*dengan mengosongkan deskripsi Contoh : Nama : operator Deskripsi : - Status : Aktif	Data *groups* tidak dapat disimpan dalam *database* karena data deskripsi 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 4.11.

Tabel 4. 11 Pengujian edit groups

Kasus dan Hasil Uji Benar (Data Benar)			
Edit*Groups*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
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
Mengisi data *groups*dengan mengosongkan nama dan deskripsi Contoh : Nama : - Deskripsi : - Status : Aktif	Data *groups* tidak dapat disimpan dalam *database* karena data nama dan deskripsi belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Mengisi data *groups*dengan mengosongkan nama Contoh : Nama : - Deskripsi : operator Status : Aktif	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
Mengisi data *groups*dengan mengosongkan deskripsi Contoh : Nama : operator Deskripsi : - Status : Aktif	Data *groups* tidak dapat disimpan dalam *database* karena data deskripsi belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan 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 4.12.

Tabel 4. 12 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 : operator Deskripsi : operator Status : aktif	Tercantum data *groups* yang akan dihapus,setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data *groups* yang akan dihapus,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 4.13.

Tabel 4. 13 Pengujian cari groups

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

berdasarkan nama *groups* contoh : admin	nama *groups* yang dicari	*groups* berdasarkan nama *groups* yang dicari	
Kasus dan Hasil Uji Salah (Data Salah)			
Data yang dicari tidak terdapat dalam database Contoh: petugas	Data groups tidak dapat ditampilkan	Muncul pesan "data tidak tersedia"	[√] Diterima [ ] Ditolak

6. Pengujian Tambah *Users*

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

Tabel 4. 14 Pengujian tambah users

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah *Users*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data *users* Contoh : Username : operator Email : admin@gmail.com Password : 12345 Confirm Password : 12345 Group : operator Status : Aktif	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 username Contoh : Username : - Email : admin@gmail.com Password : 12345 Confirm Password : 12345 Group : operator	Data *users* tidak dapat disimpan dalam *database* karena data username belum terisi	Sistem tidak dapat menyimpan data *users* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Status : Aktif			
Mengisi data *users* dengan mengosongkan *email* Contoh : Username : doni Email : - Password : 12345 Confirm Password : 12345 Group : operator Status : Aktif	Data *users* tidak dapat disimpan dalam *database* karena data email belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data *users* dengan mengosongkan password Contoh : Username : doni Email : admin@gmail.com Password : - Confirm Password : 12345 Group : operator Status : Aktif	Data *users* tidak dapat disimpan dalam *database*karena data password belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data *users* dengan mengosongkan *confirm password* Contoh : Username : doni Email : admin@gmail.com Password : 12345 Confirm Password : - Group : operator Status : Aktif	Data *users* tidak dapat disimpan dalam *database* karena data confirm password belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data *users* dengan mengisi *confirm password* yang tidak sesuai dengan *password* Contoh : Username : doni Email : admin@gmail.com Password : 12345 Confirm Password : 1234 Group : operator Status : Aktif	Data *users* tidak dapat disimpan dalam *database* karena data confirm password belum terisi	Sistem tidak dapat menyimpan data *groups* 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 4.15.

Tabel 4. 15 Pengujian edit users

Kasus dan Hasil Uji Benar (Data Benar)			
Edit*Users*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *users* lama yang akan diubah contoh : Nama : Budi	Tercantumdalam *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
Mengisi data *users* dengan mengosongkan username Contoh : Username : - Email : admin@gmail.com Password : 12345 Confirm Password : 12345 Group : operator Status : Aktif	Data *users* tidak dapat disimpan dalam *database* karena data username belum terisi	Sistem tidak dapat menyimpan data *users* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data *users* dengan mengosongkan email Contoh : Username : doni Email : - Password : 12345 Confirm Password : 12345 Group : operator Status : Aktif	Data *users* tidak dapat disimpan dalam *database* karena data email belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Mengisi data *users* dengan mengosongkan password Contoh : Username : doni Email : admin@gmail.com Password : - Confirm Password : 12345 Group : operator Status : Aktif	Data *users* tidak dapat disimpan dalam *database* karena data password belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data *users* dengan mengosongkan confirm password Contoh : Username : doni Email : admin@gmail.com Password : 12345 Confirm Password : - Group : operator Status : Aktif	Data *users* tidak dapat disimpan dalam *database* karena data confirm password belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data *users* dengan mengisi confirm password yang tidak sesuai dengan password Contoh : Username : doni Email : admin@gmail.com Password : 12345 Confirm Password : 1234 Group : operator Status : Aktif	Data *users* tidak dapat disimpan dalam *database* karena data confirm password belum terisi	Sistem tidak dapat menyimpan data *groups* ke dalam *database* dan 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 4.16.

Tabel 4. 16 Pengujian cari users

Kasus dan Hasil Uji Benar (Data Benar)			
Cari *Users*			
Data Masukan	Yang	Pengamatan	Kesimpulan

	diharapkan		
Data yang akan dicari berdasarkan nama *users* contoh : admin	Menampilkan data *users*berdasarkan nama *users* yang dicari	Berhasil Menampilkan data *users*berdasarkan nama *users* yang dicari	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data yang dicari tidak terdapat dalam database Contoh: djay	Data users tidak dapat ditampilkan	Muncul pesan "data tidak tersedia"	[√] Diterima [ ] Ditolak

9. Pengujian Edit *Member*

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

Tabel 4. 17 Pengujian edit member

Kasus dan Hasil Uji Benar (Data Benar)			
Edit*Member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *member* lama yang akan diedit contoh : Nama : Santi	Tercantum dalam *field member* berupa nama.	Dapat menampilkan *field* data *member* yang akan diedit.	[√] Diterima [ ] Ditolak
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 : - Alamat : Jl. Raya Kelurahan : Coblong Telp : 0856220176 Email : ade@gmail.com Password : 12345 Confirm Password : 12345 Status : aktif	Data *member* tidak berhasil diedit karena nama tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Data *member* yang akan diedit Nama : Ade Alamat : - Kelurahan : Coblong Telp : 0856220176 Email : ade@gmail.com Password : 12345 Confirm Password : 12345 Status : aktif	Data *member* tidak berhasil diedit karena alamat tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Data *member* yang akan diedit Nama : Ade Alamat : Jl. Raya Kelurahan : - Telp : 0856220176 Email : ade@gmail.com Password : 12345 Confirm Password : 12345 Status : aktif	Data *member* tidak berhasil diedit karena kelurahan tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Data *member* yang akan diedit Nama : Ade Alamat : Jl. Raya Kelurahan : Coblong Telp : - Email : ade@gmail.com Password : 12345 Confirm Password : 12345 Status : aktif	Data *member* tidak berhasil diedit karena telp tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Data *member* yang akan diedit Nama : Ade Alamat : Jl. Raya Kelurahan : Coblong Telp : 0856220176 Email : - Password : 12345 Confirm Password : 12345 Status : aktif	Data *member* tidak berhasil diedit karena email tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Data *member* yang akan diedit Nama : Ade Alamat : Jl. Raya Kelurahan : Coblong Telp : 0856220176 Email : ade@gmail.com Password : - Confirm Password : 12345 Status : aktif	Data *member* tidak berhasil diedit karena password tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Data *member* yang akan diedit Nama : Ade Alamat : Jl. Raya Kelurahan : Coblong Telp : 0856220176 Email : ade@gmail.com Password : 12345 Confirm Password : - Status : aktif	Data *member* tidak berhasil diedit karena confirm password tidak di isi.	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

10. Pengujian Cari *Member*

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

Tabel 4. 18 Pengujian cari member

Kasus dan Hasil Uji Benar (Data Benar)			
Cari *Member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
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	[√] Diterima [ ] Ditolak

		dicari	
Kasus dan Hasil Uji Salah (Data Salah)			
Data yang dicari tidak terdapat dalam database Contoh: harry	Data member tidak dapat ditampilkan	Muncul pesan "data tidak tersedia"	[√] Diterima [ ] Ditolak

11. Pengujian Tambah Kategori

Pengujian tambah kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah kategori dilihat pada tabel 4.19.

Tabel 4. 19 Pengujian tambah kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori contoh : Nama : Shoes Deskripsi : Sepatu	Tercantum dalam masing-masing *field* kemudian data kategoritersimpan dalam *database*.	Data kategoriberhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori dengan mengosongkan nama dan deskripsi	Data *kategori* tidak dapat disimpan dalam *database* karena data nama dan deskripsi belum terisi	Sistem tidak dapat menyimpan data kategori ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data kategori Contoh : Nama : *Shirt* Deskripsi : -	Data *kategori* tidak dapat disimpan dalam *database* karena data deskripsi belum terisi	Sistem tidak dapat menyimpan data kategori ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Mengisi data kategori Contoh : Nama : - Deskripsi : Sepatu pria	Data *kategori* tidak dapat disimpan dalam *database* karena data deskripsi belum terisi	Sistem tidak dapat menyimpan data kategori ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

12. Pengujian Edit Kategori

Pengujian edit kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit kategori dilihat pada tabel 4.20.

Tabel 4. 20 Pengujian edit kategori

Kasus dan Hasil Uji Benar (Data Benar)			
EditKategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategori lama yang akan diedit contoh : Nama : Shoes Deskripsi : Sepatu	Tercantum dalam *field* kategori berupa nama, dan deskripsi.	Dapat menampilkan *field* data kategori yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data kategori contoh : Nama : Snapback Deskripsi : Topi	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 kategoriberhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
EditKategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori dengan mengosongkan nama dan deskripsi	Data *kategori* tidak dapat disimpan dalam *database* karena data nama dan deskripsi belum terisi	Sistem tidak dapat menyimpan data kategori ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Mengisi data kategori Contoh : Nama : *Shirt* Deskripsi : -	Data *kategori* tidak dapat disimpan dalam *database* karena data deskripsi belum terisi	Sistem tidak dapat menyimpan data *kategori* ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data kategori Contoh : Nama : - Deskripsi : Sepatu pria	Data *kategori* tidak dapat disimpan dalam *database* karena data deskripsi belum terisi	Sistem tidak dapat menyimpan data kategori ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

13. Pengujian Hapus Kategori

Pengujian hapuskategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus kategori dilihat pada tabel 4.21.

Tabel 4. 21 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 : Shoes Deskripsi : Sepatu	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

14. Pengujian Cari Kategori

Pengujian cari kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari kategori dilihat pada tabel 4.22.

Tabel 4. 22 Pengujian cari kategori

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

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
Kasus dan Hasil Uji Salah (Data Salah)			
Data yang dicari tidak terdapat dalam database Contoh: Gelas	Data kategori tidak dapat ditampilkan	Muncul pesan "data tidak tersedia"	[√] Diterima [ ] Ditolak

15. Pengujian Tambah Produk

Pengujian tambah produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah produk dilihat pada tabel 4.23.

Tabel 4. 23 Pengujian tambah produk

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data produk contoh : Kode : 001 Nama : Vans Era Deskripsi : #cool #shoes Kategori : Shoes Status Produk : Stok	Tercantum dalam masing-masing *field* kemudian data produk tersimpan dalam *database*.	Data produkberhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data produk Contoh : Kode : - Nama : - Deskripsi : - Gambar : - Kategori : *Shoes* Status Produk : Stok	Data produk tidak dapat disimpan dalam *database* karena data belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Mengisi data produk Contoh : Kode : - Nama : *Vans Shoes* Deskripsi : Sepatu pria Gambar : vans.png Kategori : *Shoes* Status Produk : Stok	Data produk tidak dapat disimpan dalam *database* karena data kode belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data produk Contoh : Kode : SHVANS01 Nama : - Deskripsi : Sepatu pria Gambar : vans.png Kategori : *Shoes* Status Produk : Stok	Data produk tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data produk Contoh : Kode : SHVANS01 Nama : *Vans Shoes* Deskripsi : - Gambar : vans.png Kategori : *Shoes* Status Produk : Stok	Data produk tidak dapat disimpan dalam *database* karena data deskripsi belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data produk Contoh : Kode : SHVANS01 Nama : *Vans Shoes* Deskripsi : Sepatu pria Gambar : - Kategori : *Shoes* Status Produk : Stok	Data produk tidak dapat disimpan dalam *database* karena data gambar belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

16. Pengujian Edit Produk

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

Tabel 4. 24 Pengujian edit produk

Kasus dan Hasil Uji Benar (Data Benar)			
EditProduk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data produklama yang akan diedit contoh : Kode : 001	Tercantum dalam *field* produk berupa kode,	Dapat menampilkan field data produk yang	[√] Diterima [ ] Ditolak

Nama : Vans Era Deskripsi : #cool #shoes Kategori : Shoes Status Produk : Stok	nama, deskripsi, dankategori.	akan diedit.	
Mengisi data produk contoh : Kode : 001 Nama : Vans Era Deskripsi : #cool #shoes Kategori : Shoes Status Produk : Stok	Dapat mengisi *field* data yang akan diedit dengan benar dan data produk berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data produkberhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
EditProduk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data produk Contoh : Kode : - Nama : - Deskripsi : - Gambar : - Kategori : *Shoes* Status Produk : Stok	Data produk tidak dapat disimpan dalam *database* karena data belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data produk Contoh : Kode : - Nama : *Vans Shoes* Deskripsi : Sepatu pria Gambar : vans.png Kategori : *Shoes* Status Produk : Stok	Data produk tidak dapat disimpan dalam *database* karena data kode belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data produk Contoh : Kode : SHVANS01 Nama : - Deskripsi : Sepatu pria Gambar : vans.png Kategori : *Shoes* Status Produk : Stok	Data produk tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data produk Contoh : Kode : SHVANS01 Nama : *Vans Shoes* Deskripsi : - Gambar : vans.png Kategori : *Shoes*	Data produk tidak dapat disimpan dalam *database* karena data deskripsi belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Status Produk : Stok			
Mengisi data produk Contoh : Kode : SHVANS01 Nama : *Vans Shoes* Deskripsi : Sepatu pria Gambar : - Kategori : *Shoes* Status Produk : Stok	Data produk tidak dapat disimpan dalam *database* karena data gambar belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

17. Pengujian Tambah Detail Produk

Pengujian tambah detail produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah detail produk dilihat pada tabel 4.25.

Tabel 4. 25 Pengujian tambah detail produk

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Detail Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data detail produk contoh : Nama : Vans Era Warna: Hitam Stok : 12 Status Produk : Stok Ukuran : 42 Harga : 23000	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 Contoh : Nama : - Warna : - Gambar : - Stok : - Ukuran : - Harga : -	Data produk tidak dapat disimpan dalam *database* karena data belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data detailproduk	Data produk tidak dapat	Sistem tidak dapat	[√] Diterima [ ] Ditolak

Contoh : Nama : *Vans Volt* Warna : - Gambar : volt.png Stok : 12 Ukuran : 42 Harga : 420000	disimpan dalam *database* karena data warna belum terisi	menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	
Mengisi data detailproduk Contoh : Nama : - Warna : Hitam Gambar : volt.png Stok : 12 Ukuran : 42 Harga : 420000	Data produk tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data detailproduk Contoh : Nama : *Vans Volt* Warna : Hitam Gambar : - Stok : 12 Ukuran : 42 Harga : 420000	Data produk tidak dapat disimpan dalam *database* karena data gambar belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data detailproduk Contoh : Nama : *Vans Volt* Warna : Hitam Gambar : volt.png Stok : - Ukuran : 42 Harga : 420000	Data produk tidak dapat disimpan dalam *database* karena data stok belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data detailproduk Contoh : Nama : *Vans Volt* Warna : Hitam Gambar : volt.png Stok : 12 Ukuran : - Harga : 420000	Data produk tidak dapat disimpan dalam *database* karena data ukuran belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data detailproduk Contoh : Nama : *Vans Volt* Warna : Hitam Gambar : volt.png Stok : 12	Data produk tidak dapat disimpan dalam *database* karena data harga belum terisi	Sistem tidak dapat menyimpan data produk ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Ukuran : 42 Harga : -			

18. Pengujian Tambah Diskon

Pengujian tambah diskon dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah diskon dilihat pada tabel 4.26.

Tabel 4. 4 Pengujian tambah diskon

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Diskon			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
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 diskontersimpan dalam *database*.	Data diskonberhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Diskon			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data diskon Produk : - Tgl Awal : - Tgl Akhir : - Diskon : - Min. Qty : -	Data diskon tidak dapat disimpan dalam *database* karena databelum terisi	Sistem tidak dapat menyimpan data diskon ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data diskon Produk : BLENJOI Tgl Awal : - Tgl Akhir : 12 Maret 2012 Diskon : 15 Min. Qty : 2	Data diskon tidak dapat disimpan dalam *database* karena data tgl awal belum terisi	Sistem tidak dapat menyimpan data diskon ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data diskon Produk : BLENJOI Tgl Awal : 12 Januari 2012 Tgl Akhir : - Diskon : 15 Min. Qty : 2	Data diskon tidak dapat disimpan dalam *database* karena data tgl akhir belum terisi	Sistem tidak dapat menyimpan data diskon ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Mengisi data diskon Produk : BLENJOI Tgl Awal : 12 Januari 2012 Tgl Akhir : 12 Maret 2012 Diskon : - Min. Qty : 2	Data diskon tidak dapat disimpan dalam *database* karena data diskon belum terisi	Sistem tidak dapat menyimpan data diskon ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data diskon Produk : BLENJOI Tgl Awal : 12 Januari 2012 Tgl Akhir : 12 Maret 2012 Diskon : 15 Min. Qty : -	Data diskon tidak dapat disimpan dalam *database* karena data min. qty belum terisi	Sistem tidak dapat menyimpan data diskon ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

19. Pengujian Edit Diskon

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

Tabel 4. 5 Pengujian edit diskon

Kasus dan Hasil Uji Benar (Data Benar)			
EditDiskon			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategorilama yang akan diedit contoh : Kode Produk: BLENJOI Diskon : 10 Tanggal Awal : 12-07-2012 Tanggal Akhir : 19-07-2012 Min-Quantity : 1	Tercantum dalam *field* diskon berupa kode produk, diskon, tanggal awal, tanggal akhir, danmin quantity.	Dapat menampilkan *field* data diskon yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data diskon contoh : Kode Produk: BLENJOI 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 diskonberhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
EditDiskon			

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data diskon Produk : - Tgl Awal : - Tgl Akhir : - Diskon : - Min. Qty : -	Data diskon tidak dapat disimpan dalam *database* karena databelum terisi	Sistem tidak dapat menyimpan data diskon ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data diskon Produk : BLENJOI Tgl Awal : - Tgl Akhir : 12 Maret 2012 Diskon : 15 Min. Qty : 2	Data diskon tidak dapat disimpan dalam *database* karena data tgl awal belum terisi	Sistem tidak dapat menyimpan data diskon ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data diskon Produk : BLENJOI Tgl Awal : 12 Januari 2012 Tgl Akhir : - Diskon : 15 Min. Qty : 2	Data diskon tidak dapat disimpan dalam *database* karena data tgl akhir belum terisi	Sistem tidak dapat menyimpan data diskon ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data diskon Produk : BLENJOI Tgl Awal : 12 Januari 2012 Tgl Akhir : 12 Maret 2012 Diskon : - Min. Qty : 2	Data diskon tidak dapat disimpan dalam *database* karena data diskon belum terisi	Sistem tidak dapat menyimpan data diskon ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data diskon Produk : BLENJOI Tgl Awal : 12 Januari 2012 Tgl Akhir : 12 Maret 2012 Diskon : 15 Min. Qty : -	Data diskon tidak dapat disimpan dalam *database* karena data min. qty belum terisi	Sistem tidak dapat menyimpan data diskon ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

20. Pengujian Cari Diskon

Pengujian cari diskon dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari diskon dilihat pada tabel 4.28.

Tabel 4. 6 Pengujian cari diskon

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

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
Kasus dan Hasil Uji Salah (Data Salah)			
Data yang dicari tidak terdapat dalam database Contoh: 75	Data diskon tidak dapat ditampilkan	Muncul pesan "data tidak tersedia"	[√] Diterima [ ] Ditolak

21. Pengujian Cari Daftar Harga

Pengujian cari daftar harga dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari daftar harga dilihat pada tabel 4.29.

Tabel 4. 7 Pengujian cari daftar harga

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Daftar Harga			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
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
Kasus dan Hasil Uji Salah (Data Salah)			
Data yang dicari tidak terdapat dalam database Contoh: 75	Data daftar harga tidak dapat ditampilkan	Muncul pesan "data tidak tersedia"	[√] Diterima [ ] Ditolak

22. Pengujian Detail Penjualan

Pengujian detail penjualan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujiandetailpenjualandilihat pada tabel 4.30.

Tabel 4. 8 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

23. Pengujian Cari Penjualan

Pengujian cari penjualan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari penjualan dilihat pada tabel 4.31.

Tabel 4. 9 Pengujian cari penjualan

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Penjualan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
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
Kasus dan Hasil Uji Salah (Data Salah)			
Data yang dicari tidak terdapat dalam database Contoh: 75	Data users tidak dapat ditampilkan	Muncul pesan “data tidak tersedia”	[√] Diterima [ ] Ditolak

24. Pengujian Cari Pembayaran

Pengujian cari pembayaran dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari pembayaran dilihat pada tabel 4.32.

Tabel 4. 32 Pengujian cari pembayaran

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

Data yang dicari tidak terdapat dalam database Contoh: 75	Data pembayaran tidak dapat ditampilkan	Muncul pesan "data tidak tersedia"	[√] Diterima [ ] Ditolak

25. Pengujian Tambah Pengiriman

Pengujian tambah pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah pengiriman dilihat pada tabel 4.33.

Tabel 4. 33 Pengujian tambah pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data pengiriman contoh : Kode Pengirman : 001 Tanggal Pengirman: 17-07-2012 Status Pengiriman : mengirim	Tercantum dalam masing-masing *field* kemudian data pengirimantersimpan dalam *database*.	Data pengirimanberhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data pengiriman Kode pengiriman : - Tgl kirim : -	Data pengiriman tidak dapat disimpan dalam *database* karena data belum terisi	Sistem tidak dapat menyimpan data pengiriman ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data pengiriman Kode pengiriman : 123456 Tgl kirim : -	Data pengiriman tidak dapat disimpan dalam *database* karena data tgl kirim belum terisi	Sistem tidak dapat menyimpan data pengiriman ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Mengisi data pengiriman Kode pengiriman : - Tgl kirim : 12 Januari 2012	Data pengirimantidak dapat disimpan dalam *database* karena data kode pengiriman belum terisi	Sistem tidak dapat menyimpan data pengirimanke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

26. Pengujian Cari Pengiriman

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

Tabel 4. 34 Pengujian cari pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
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
Kasus dan Hasil Uji Salah (Data Salah)			
Data yang dicari tidak terdapat dalam database Contoh: 75	Data pengiriman tidak dapat ditampilkan	Muncul pesan “data tidak tersedia”	[√] Diterima [ ] Ditolak

27. Pengujian Cari Retur

Pengujian cari retur dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari retur dilihat pada tabel 4.35.

Tabel 4. 35 Pengujian cari retur

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

Data yang dicari tidak terdapat dalam database Contoh: 75	Data retur tidak dapat ditampilkan	Muncul pesan "data tidak tersedia"	[√] Diterima [ ] Ditolak

28. Pengujian Pembuatan Laporan Penjualan

Pengujian pembuatan laporan penjualan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian pembuatan laporan penjualan dilihat pada tabel 4.36.

Tabel 4. 36 Pengujian pembuatan laporan

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 penjualan	[√] 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 penjualan pada periode tertentu dalam bentuk *file*.	Dapat menampilkan laporan penjualan pada periode tertentu dalam benuk *file.*	[√] Diterima [ ] Ditolak

29. Pengujian Pembuatan Laporan Pengiriman

Pengujian pembuatan laporan pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian pembuatan laporan pengiriman dilihat pada tabel 4.37.

Tabel 4. 37 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 pengiriman	Dapat menampilkan *form* laporan	[√] Diterima [ ] Ditolak

		pengiriman	
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

30. Pengujian Pembuatan Laporan Stok

Pengujian pembuatan laporanstok dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian pembuatan laporan stok dilihat pada tabel 4.38.

Tabel 4. 38 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 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 stok pada periode tertentu dalam bentuk *file*.	Dapat menampilkan laporan stok pada periode tertentu dalam benuk *file.*	[√] Diterima [ ] Ditolak

31. Pengujian Tambah Propinsi

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

Tabel 4. 39 Pengujian tambah propinsi

Kasus dan Hasil Uji Benar (Data Benar)
Tambah Propinsi

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data propinsi contoh : Nama : Jawa Barat	Tercantum dalam masing-masing *field* kemudian data propinsitersimpan 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	Dat apropinsi 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

32. Pengujian Edit Propinsi

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

Tabel 4. 40 Pengujian edit propinsi

Kasus dan Hasil Uji Benar (Data Benar)			
EditPropinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data propinsilama 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 propinsiberhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
EditPropinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

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

33. Pengujian Hapus Propinsi

Pengujian hapus propinsi dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus propinsi dilihat pada tabel 4.41.

Tabel 4. 41 Pengujian hapus propinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus Propinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data propinsi 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

34. Pengujian Cari Propinsi

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

Tabel 4. 42 Pengujian cari propinsi

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

35. Pengujian Tambah Kota/Kab

Pengujian tambah kota/kab dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah kota/kab dilihat pada tabel 4.43.

Tabel 4. 43 Pengujian tambah kota/kab

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

36. Pengujian Edit Kota/Kab

Pengujian edit kota/kab dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit kota/kab dilihat pada tabel 4.44.

Tabel 4. 44 Pengujian edit kota/kab

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Kota/Kab			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kota/kablama 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 disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data kota/kabberhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
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/kabtidak berhasil diedit karena nama, belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

37. Pengujian Hapus Kota/Kab

Pengujian hapuskota/kab dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus kota/kab dilihat pada tabel 4.45.

Tabel 4. 45 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

38. Pengujian Cari Kota/Kab

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

Tabel 4. 46 Pengujian cari kota/kab

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

39. Pengujian Tambah Kecamatan

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

Tabel 4. 47 Pengujian tambah kecamatan

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

40. Pengujian Edit Kecamatan

Pengujian editkecamatan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit kecamatan dilihat pada tabel 4.48.

Tabel 4. 48 Pengujian edit kecamatan

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Kecamatan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kecamatan lama yang akan diedit contoh : Nama : Sukaraya Propinsi : Jawa Barat Kota : Bandung	Tercantun dalam *field* kecamatan berupa nama, propinsi, dan kota.	Dapat menampilkan *field* data kecamatan yang akan diedit.	[√] Diterima [ ] 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)			
EditKecamatan			
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

41. Pengujian Hapus Kecamatan

Pengujian hapuskecamatan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus kecamatan dilihat pada tabel 4.49.

Tabel 4. 49 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

42. Pengujian Cari Kecamatan

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

Tabel 4. 50 Pengujian cari kecamatan

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

43. Pengujian Tambah Kelurahan

Pengujian tambah kelurahan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah kelurahan dilihat pada tabel 4.51.

Tabel 4. 51 Pengujian tambah kelurahan

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

44. Pengujian Edit Kelurahan

Pengujian editkelurahan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit kelurahan dilihat pada tabel 4.52.

Tabel 4. 52 Pengujian edit kelurahan

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Kelurahan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
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 contoh : Nama : Ciwaruga Propisni : Jawa Barat Kota : Bandung Kecamatan : Cimenyan	Dapat mengisi *field* data yang akan diedit dengan benar dan data kelurahan berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data kelurahan berhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
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

45. Pengujian Hapus Kelurahan

Pengujian hapus kelurahan dilihat pada tabel 4.53.

Tabel 4. 53 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

46. Pengujian Cari Kelurahan

Pengujian cari kelurahan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari kelurahan dilihat pada tabel 4.54.

Tabel 4. 54 Pengujian cari kelurahan

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Kelurahan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dicari berdasarkan nama kelurahan	Menampilkan data kelurahan berdasarkan nama kelurahanyang dicari	Berhasil Menampilkan datakelurahanberdasarkan nama kelurahan yang dicari	[√] Diterima [ ] Ditolak

47. Pengujian Tambah Vendor Pengiriman

Pengujian tambah vendor pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah vendor pengiriman dilihat pada tabel 4.55.

Tabel 4. 55 Pengujian tambah vendor pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Kelurahan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data vendor pengiriman contoh : nama : JNE kode : JE01 deskripsi : Jasa Pengiriman JNE	Tercantum dalam masing-masing *field* kemudian data vendor pengiriman tersimpan dalam *database*.	Data vendor pengiriman berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Vendor Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data vendor pengiriman dengan mengosongkan nama	Datavendor 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

48. Pengujian Edit Vendor Pengiriman

Pengujian editvendor pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit vendor pengiriman dilihat pada tabel 4.56.

Tabel 4. 56 Pengujian edit vendor pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Edit Vendor Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data vendor pengiriman lama yang akan diedit contoh : Nama : JNE	Tercantum dalam *field* vendor pengiriman berupa nama,	Dapat menampilkan *field* datavendor pengirimanyang akan diedit.	[√] Diterima [ ] Ditolak

Kode : JE01 Deskripsi : Jasa Pengirman JNE	kode, dan deskripsi.		
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 dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data vendor pengiriman berhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
EditVendor 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

49. Pengujian Hapus Vendor Pengiriman

Pengujian hapusvendor pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus vendor pengiriman dilihat pada tabel 4.57.

Tabel 4. 57 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

50. Pengujian Cari Vendor Pengiriman

Pengujian cari vendor pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari vendor pengiriman dilihat pada tabel 4.58.

Tabel 4. 58 Pengujian cari vendor pengiriman

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

51. Pengujian Tambah Tarif Pengiriman

Pengujian tambah tarif pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah tarif pengiriman dilihat pada tabel 4.59.

Tabel 4. 59 Pengujian tambah tarif pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Tarif Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data tarif pengiriman pengiriman contoh : Vendor : Tiki Nama : REG Kode : TK01 Tujuan : Bandung - Surabaya Tarif : 11000 Pola Perhitungan : Flat	Tercantum dalam masing-masing *field* kemudian data tarif pengirimantersimpan dalam *database*.	Data tarif pengiriman berhasil disimpan dalam *database.*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah Tarif Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data tarif pengiriman dengan	Datatarif pengirimantidak	Sistem tidak dapat	[√] Diterima [ ] Ditolak

mengosongkan vendor	dapat disimpan dalam *database* karena data vendor belum terisi	menyimpan data tarif pengiriman ke dalam *database* dan muncul pesan kesalahan	

52. Pengujian Edit Tarif Pengiriman

Pengujian edittarif pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian edit tarif pengiriman dilihat pada tabel 4.60.

Tabel 4. 60 Pengujian edit tarif pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
EditTarifPengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
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 vendor, nama, kode, tujuan, tarif, dan pola perhitungan.	Dapat menampilkan *field* data tarif pengiriman yang akan diedit.	[√] Diterima [ ] Ditolak
Mengisi data tarif pengiriman contoh : Vendor : JNE Nama : YES Kode : TK01 Tujuan : Bandung - Surabaya Tarif : 11000 Pola Perhitungan : Flat	Dapat mengisi *field* data yang akan diedit dengan benar dan data tarif pengiriman berhasil diedit dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diedit dengan benar dan data tarif pengiriman berhasil diedit dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
EditVendor Pengiriman			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data tarif pengiriman yang akan diedit Vendor : JNE Nama : - Kode : TK01 Tujuan : Bandung -	Data tarif pengirimantidak berhasil diedit karena nama, belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan	[√] Diterima [ ] Ditolak

Surabaya Tarif : 11000 Pola Perhitungan : Flat		kesalahan	

53. Pengujian Hapus Tarif Pengiriman

Pengujian hapustarif pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus tarif pengiriman dilihat pada tabel 4.61.

Tabel 4. 61 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 tarif pengiriman yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

54. Pengujian Cari Tarif Pengiriman

Pengujian cari tarif pengiriman dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari tarif pengiriman dilihat pada tabel 4.62.

Tabel 4. 62 Pengujian cari tarif pengiriman

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

55. Pengujian Register *Member*

Pengujian register*member* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian register *member* dilihat pada tabel 4.63.

Tabel 4. 63 Pengujian register member

Kasus dan Hasil Uji Benar (Data Benar)			
Pendaftaran Register *Member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Sistem menyimpan data register kedalam *database*	Berhasil menyimpan data register member ke dalam database	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Register*member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Masukan data pendaftaran, contoh : No. KTP :- Nama : - Telp : - Alamat : - Provinsi : - Kota : - Kecamatan : - Kelurahan : - Email : - Password : - Confirm Password : - Captcha : -	Data member tidak dapat disimpan dalam *database* karena databelum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Masukan data pendaftaran, contoh : No. KTP :- Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data member tidak dapat disimpan dalam *database* karena data No. KTP belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : - Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data member tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : - Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com	Data member tidak dapat disimpan dalam *database* karena data Telp belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Password : qwerty Confirm Password : qwerty Captcha : K5PHQ			
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : - Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data *member tidak* dapat disimpan dalam *database* karena data Alamat belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : - Kota : - Kecamatan : - Kelurahan : - Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data member tidak dapat disimpan dalam *database* karena data Provinsi belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : - Kecamatan : - Kelurahan : -	Data member tidak dapat disimpan dalam *database* karena data Kota belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ			
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : - Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data member tidak dapat disimpan dalam *database* karena data kecamatan belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : - Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data member tidak dapat disimpan dalam *database* karena data kelurahan belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung	Data member tidak dapat disimpan dalam *database* karena data email belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : - Password : qwerty Confirm Password : qwerty Captcha : K5PHQ			
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : - Confirm Password : qwerty Captcha : K5PHQ	Data *member* tidak dapat disimpan dalam *database* karena data password belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : - Captcha : K5PHQ	Data *member* tidak dapat disimpan dalam *database* karena data confirm password belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : -	Data *member* tidak dapat disimpan dalam *database* karena data captcha belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

56. Pengujian Ubah Akun

Pengujian ubah akun dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah akun dilihat pada tabel 4.64.

Tabel 4. 64 Pengujian ubah akun

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah akun			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data akun lama yang akan diubah No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Tercantun dalam *field* akun berupa *username*, *password*, nama, *email*, jenis-kelamin, kota, telepon	Dapat menampilkan *field* data akun yang akan diubah.	[√] Diterima [ ] Ditolak
Mengisi data akun contoh : No. KTP :123456	Dapat mengisi *field* data yang akan diubah	Dapat mengisi *field* data yang akan diubah	[√] Diterima [ ] Ditolak

Nama : Deni Telp : 0856712124 Alamat : Jl Buntu Provinsi : DKI Jakarta Kota : Jakarta Utara Kecamatan : Taman Sari Kelurahan : Grogol Email : deni@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	dengan benar dan data akun berhasil diubah dan disimpan dalam *database*.	dengan benar dan data akun berhasil diubah dan disimpan dalam *databsase*	
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah Akun			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Masukan data pendaftaran, contoh : No. KTP :- Nama : - Telp : - Alamat : - Provinsi : - Kota : - Kecamatan : - Kelurahan : - Email : - Password : - Confirm Password : - Captcha : -	Data *member* tidak dapat disimpan dalam *database* karena databelum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :- Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty	Data *member* tidak dapat disimpan dalam *database* karena data No. KTP belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Captcha : K5PHQ			
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : - Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data member tidak dapat disimpan dalam *database* karena data nama belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : - Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data *member* tidak dapat disimpan dalam *database* karena data Telp belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : - Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email :	Data *member* tidak dapat disimpan dalam *database* karena data Alamat belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ			
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : - Kota : - Kecamatan : - Kelurahan : - Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data *member* tidak dapat disimpan dalam *database* karena data Provinsi belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : - Kecamatan : - Kelurahan : - Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data *member* tidak dapat disimpan dalam *database* karena data Kota belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : - Kelurahan : Sukahaji	Data *member* tidak dapat disimpan dalam *database* karena data kecamatan belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ			
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : - Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data *member* tidak dapat disimpan dalam *database* karena data kelurahan belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : - Password : qwerty Confirm Password : qwerty Captcha : K5PHQ	Data *member* tidak dapat disimpan dalam *database* karena data *email* belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung	Data *member* tidak dapat disimpan dalam *database* karena data *password* belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : - Confirm Password : qwerty Captcha : K5PHQ			
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : - Captcha : K5PHQ	Data *member* tidak dapat disimpan dalam *database* karena data *confirm password* belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh : No. KTP :123456 Nama : Budi Telp : 085671234 Alamat : Jl Raya Provinsi : Jawa Barat Kota : Bandung Kecamatan : Bbk Ciparay Kelurahan : Sukahaji Email : budi@yahoo.com Password : qwerty Confirm Password : qwerty Captcha : -	Data *member* tidak dapat disimpan dalam *database* karena data *captcha* belum terisi	Sistem tidak dapat menyimpan data pendaftaran ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

57. 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 4.65.

Tabel 4. 65 Pengujian tambah produk ke shoping cart

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah Produk ke *Shoping Cart*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
membeli produk dengan memasukan data produk contoh : produk : Vans Era	produk masuk kedalam keranjang belanja.	Sistem berhasil memasukan data produk kedalam *shoping cart*.	[√] Diterima [ ] Ditolak

58. 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 4.66.

Tabel 4. 66 Pengujian hapus data produk dari shopping 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 : Vans Era	Tercantum data produk dalam *shoping cart* yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data produk dalam *shoping cart* yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

59. Pengujian Cari Produk

Pengujian cari produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari produk dilihat pada tabel 4.67.

Tabel 4. 67 Pengujian cari produk

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

Kasus dan Hasil Uji Benar (Data Benar)			
Data yang dicari tidak terdapat dalam database Contoh: Asics Onitsuka	Data produk tidak dapat ditampilkan	Muncul pesan "data tidak tersedia"	[√] 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 Mag & Shoes 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 Mag & Shoes. 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 Mag & Shoes

Penelitian di lingkungan Mag & Shoes dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang ada. Penelitian ini ditujukan kepada admin, kasir, dan operator dengan menggunakan teknik wawancara.

Berikut ini adalah pertanyaan dari kuesioner yang diujikan kepada responden sebanyak 30 orang dengan skala *likert* untuk memperoleh kesimpulan, perhitungan ini menggunakan rumus (2.4).

4.2.3.1 Kuesioner Member

Data yang telah terkumpul melalui kuesioner, kemudian penulis olah ke dalam bentuk kuantitatif, yaitu dengan cara menetapkan skor jawaban dari pertanyaan yang telah dijawab oleh responden, dimana pemberian skor tersebut didasarkan pada ketentuan Sugiyono [21].

Tabel 4.68 Kriteria Penilaian

Jawaban	Bobot
SS = Sangat Setuju	5
ST = Setuju	4
RR = Ragu-ragu	3
TS = Tidak Setuju	2
STS = Sangat Tidak Setuju	1

Kuesioner untuk member terdiri dari 7 pertanyaan (contoh kuesioner dapat dilihat pada lampiran), dengan pertanyaan sebagai berikut :

1. Kuesioner untuk responden dengan tujuan mempermudah proses transaksi pembelian produk.
   a. Apakah anda setuju bahwa *website* ini lebih memudahkan dalam melakukan transaksi jual-beli barang khususnya dari tempat yang berjauhan dengan toko?
   b. Apakah anda setuju bahwa sistem pembayaran *online* yang digunakan lebih memudahkan dalam proses transaksi?
   c. Apakah anda setuju bahwa prosedur untuk melakukan proses pembelian di *website* Mag & Shoes dapat dimengerti dengan mudah?

Tabel 4.69 Perhitungan Skor Kuesioner Dengan Tujuan Mempermudah Proses Transaksi

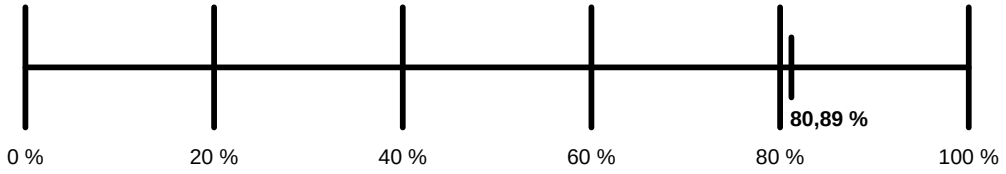
Jawaban	Skor	Responden	Jumlah Pertanyaan	(Skor x Pertanyaan)	Responden x (Skor x Pertanyaan)	Persentase (%)
SS	5	30	3	15	450	100 %
ST	4	30	3	12	360	80 %
RR	3	30	3	9	270	60 %
TS	2	30	3	6	180	40 %
STS	1	30	3	3	90	20 %

Berdasarkan perhitungan dari jumlah skor untuk seluruh responden dari rumus Responden x (skor x pertanyaan) maka didapat jumlah skor ideal (maksimum) adalah 450 dengan persentase 100%. Berikut ini adalah hasil skor perhitungan masing-masing jawaban yang sudah dihitung nilainya. Kuesioner ini diujikan kepada 30 responden sebagai *sample* yang diambil dari tujuan mempermudah proses transaksi bagi konsumen.

Tabel 4.70 Hasil Pengujian Kuesioner Dengan Tujuan Mempermudah Proses Transaksi

Responden	**Pertanyaan**			**Total**
	1	**2**	**3**	
1	5	4	4	13
2	5	4	5	14
3	5	4	5	14
4	5	4	5	14
5	3	3	3	9
6	5	4	5	14
7	3	3	5	11
8	4	5	4	13
9	5	5	5	15
10	5	4	5	14
11	3	3	3	9
12	5	4	4	13
13	5	4	5	14
14	5	4	5	14
15	4	4	4	12
16	4	4	2	10
17	4	3	4	11
18	4	3	4	11
19	5	4	3	12
20	4	5	4	13
21	3	3	3	9
22	3	2	3	8
23	4	5	3	12
24	5	4	3	12
25	5	4	5	14

26	5	5	4	14
27	4	4	4	12
28	2	3	2	7
29	5	5	4	14
30	5	4	3	12
Total				**364**
Skala Persentase (%)				**80,89**



Gambar 4.15 Skala Hasil Kuesioner Dengan Tujuan Mempermudah Proses Transaksi

Berdasarkan perhitungan data yang diperoleh, skor yang didapat berada pada posisi melebihi jawaban "Ragu-Ragu", dimana sudah melewati skor rata-rata dan mendekati skor ideal yaitu 80,89% maka dapat disimpulkan bahwa responden sangat setuju *e-commerce* dapat mempermudah dalam proses transaksi pembelian produk kepada konsumen.

2. Kuesioner untuk responden dengan tujuan mempermudah memperoleh informasi produk secara lengkap dan *up-to-date.*
    a. Apakah anda setuju bahwa anda mudah untuk mendapatkan informasi barang beserta detailnya melalui *website* Mag & Shoes?
    b. Apakah anda setuju bahwa informasi mengenai Mag & Shoes tersedia dengan lengkap di *website* ini?
    c. Apakah anda setuju bahwa informasi produk yang diberikan oleh *website* Mag & Shoes ini sudah sesuai dengan informasi yang anda butuhkan?
    d. Apakah anda setuju bahwa *website* Mag & Shoes ini telah layak untuk dipublikasikan?

Tabel 4.71 Perhitungan Skor Kuesioner Dengan Tujuan Penyampaian Informasi Produk

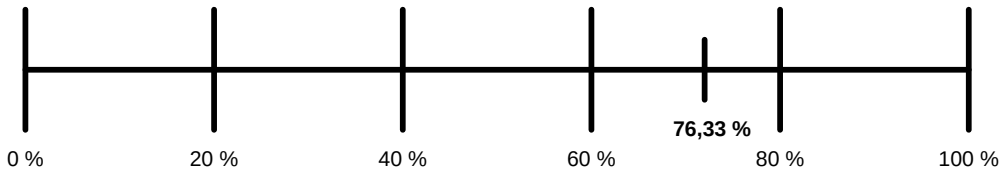
Jawaban	Skor	Responden	Jumlah Pertanyaan	(Skor x Pertanyaan)	Responden x (Skor x Pertanyaan)	Persentase (%)
SS	5	30	4	20	600	100 %
ST	4	30	4	16	480	80 %
RR	3	30	4	12	360	60 %
TS	2	30	4	8	240	40 %
STS	1	30	4	4	120	20 %

Berdasarkan perhitungan dari jumlah skor untuk seluruh responden dari rumus Responden x (skor x pertanyaan) maka didapat jumlah skor ideal (maksimum) adalah 600 dengan persentase 100%. Berikut ini adalah hasil skor perhitungan masing-masing jawaban yang sudah dihitung nilainya. Kuesioner ini diujikan kepada 30 responden sebagai *sample* yang diambil dari tujuan penyampaian informasi produk.

Tabel 4.72 Hasil Pengujian Kuesioner Dengan Tujuan Penyampaian Informasi Produk

Responden	Pertanyaan				Total
	1	2	3	4	
1	5	4	4	4	17
2	5	4	5	3	17
3	5	4	5	4	18
4	5	4	5	2	16
5	3	3	3	5	14
6	5	4	5	3	17
7	3	3	5	2	13
8	4	5	4	4	17
9	5	5	5	3	18
10	2	4	5	4	15
11	3	3	3	5	14
12	5	4	4	2	15
13	5	4	5	3	17
14	5	4	5	5	19
15	4	4	4	2	14
16	4	4	2	4	14
17	4	3	4	2	13

18	4	3	4	3	14
19	5	4	3	4	16
20	4	5	4	2	15
21	3	3	3	3	12
22	3	2	3	5	13
23	1	5	3	4	13
24	5	4	3	2	14
25	5	4	5	3	17
26	5	5	4	5	19
27	4	4	4	3	15
28	2	3	2	2	9
29	5	5	4	4	18
30	5	4	3	3	15
Total					**458**
Skala Persentase (%)					**76,33**



Gambar 4.16 Skala Hasil Kuesioner Dengan Tujuan Mempermudah Penyampaian Informasi Produk

Berdasarkan perhitungan data yang diperoleh, skor yang didapat berada pada posisi melebihi jawaban "Ragu-Ragu", dimana sudah melewati skor rata-rata dan mendekati skor ideal yaitu 76,33% maka dapat disimpulkan bahwa responden setuju *e-commerce* dapat mempermudah dalam proses penyampaian informasi produk kepada konsumen.

4.2.3.2 Wawancara Operator

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

1. Apakah dengan adanya *website e-commerce* ini dapat memudahkan dalam melakukan promosi produk di Mag & Shoes?

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 master.

3. Apakah *website* ini dapat memudahkan dalam mengelola data produk Mag & Shoes?

Jawaban:

Data produk lebih mudah untuk dikelola dan pengolahan stok produk lebih mudah untuk dikontrol.

4. Apakah pengolahan konfigurasi situs ini mudah dipahami dan digunakan?

Jawaban:

Informasi mengenai toko atau biasa disebut *company profile* akan sangat mudah untuk di-*update* setiap ada promo tertentu atau *event* khusus. Fasilitas ini sangat membantu untuk terus meng-*update* setiap info terbaru.

4.2.3.3 Wawancara 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 admin, hasil dari wawancara dapat dilihat sebagai berikut :

1. 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.

2. Apakah anda setuju bahwa dengan adanya website E-Commerce ini akan memudahkan dalam mengelola data *users*?

Jawaban:

Iya, karena menurut saya sistem ini dapat mengelola semua proses tersebut.

3. Apakah aplikasi ini membantu anda dalam proses pembuatan laporan?

   Jawaban:

   Menurut pendapat saya pribadi aplikasi ini sangat membantu dalam proses pembuatan laporan, karena itu sangat penting untuk memantau apakah transaksi yang dilakukan mengalami perkembangan atau tidak.

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

   Jawaban:

   Sudah layak, karena sejauh ini aplikasi *e-commerce* ini sudah memenuhi kebutuhan perusahaan dalam proses meningkatkan nilai penjualan.

4.2.3.4 Wawancara Kasir

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

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

   Jawaban:

   Halaman administrator ini cukup *user friendly*, semua panel mudah dimengerti dan digunakan sehingga dapat membantu proses pengolahan transaksi dengan cepat.

2. Apakah anda setuju bahwa dengan adanya website *E-Commerce* ini akan memudahkan dalam mengelola data transaksi?

   Jawaban:

   Iya, karena setiap transaksi yang masuk langsung dicatat dalam sistem ini. Pihak perusahaan dapat mengetahui seluruh proses penjualan, pembayaran, pengiriman, hingga retur.

3. Apakah aplikasi ini membantu anda dalam proses pembuatan laporan?

   Jawaban:

Sangat membantu karena fasilitas pengolahan laporan yang tersedia disini cukup lengkap mulai dari pembuatan laporan transaksi, laporan produk, laporan pengiriman, hingga laporan retur.

4. Apakah aplikasi yang dibangun sudah layak untuk di *online*-kan?
   Jawaban:
   Sudah layak, karena sejauh ini aplikasi *e-commerce* ini sudah memenuhi kebutuhan perusahaan dalam proses meningkatkan nilai penjualan.

4.2.3.5 Kesimpulan Pengujian Beta

Berdasarkan hasil persentase dan wawancara diatas yang didapatkan dari pengujian beta, maka sistem penerapan *E-Commerce* untuk sistem penjualan *online* ini sudah sesuai dengan tujuan yaitu dapat membantu pihak Mag & Shoes dalam menyampaikan informasi produk secara lengkap dan *up-to-date*, memudahkan konsumen untuk melakukan pembelian tanpa harus datang ke toko, mengelola data transaksi dan produk, serta memudahkan proses pembuatan laporan bagi pihak Mag & Shoes.