BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

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

4.1 Implemantasi Sistem

Tahap implementasi merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan tahap dimana sistem siap untuk dioperasikan, yang terdiri dari penjelesan mengenai lingkungan implementasi, dan implementasi program.

4.1.1 Lingkungan Sistem

Untuk mendukung aplikasi yang akan diterapkan pada lingkungan implementasi, maka dalam hal ini menggunakan perangkat keras dan perangkat lunak yang menunjang dalam pembangunan aplikasi *e-commerce* pada Toko Istana Helmet Bandung

4.1.1.1 Perangkat Keras Yang Digunakan

Spesifikasi perangkat keras yang digunakan dalam pembangunan perangkat keras adalah tercantum pada tabel 4.1 sebagai berikut :

Tabel 4.1 Perangkat Keras Yang Digunakan

Nama Perangkat	Keterangan
Processor	Intel Core i3
RAM	DDR3 2 Gb
Space Harddisk	250 Gb
VGA Card	512 Mb
Monitor	ASUS 17’
DVD	Asus DVD-RW
Keyboard dan Mouse	Logitech
modem	Koneksi speedy telkom

4.1.1.2 Perangkat Lunak Yang Digunakan

Spesifikasi perangkat lunak yang digunakan dalam membangun aplikasi *e-commerce* pada Toko Istana helmet bandung adalah sebagai berikut :

Tabel 4.2 Perangkat Lunak Yang Digunakan

Nama Aplikasi	Keterangan
PHP	Sebagai bahasa pemrograman
Windows 7	Sebagai sistem Operasi
Adobe Dreamweaver CS 4	Sebagai Software tools
MySQL	sebagai Database Management System (DBMS).
Mozilla Firefox.	Sebagai Web browser

4.2 Implementasi Database

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

Tabel 4.3 Implementasi Database

No	Nama Tabel	Source Code
1	Admin	CREATE TABLE IF NOT EXISTS `admin` (`id_admin` int(11) NOT NULL DEFAULT '0', `id_superadmin` int(11) DEFAULT NULL, `nama_admin` varchar(50) COLLATE latin1_general_ci DEFAULT NULL, `alamat_admin` text COLLATE latin1_general_ci, `email_admin` varchar(50) COLLATE latin1_general_ci DEFAULT NULL, `telp_admin` varchar(20) COLLATE latin1_general_ci DEFAULT NULL, `status_admin` enum('0','1') COLLATE latin1_general_ci DEFAULT NULL, `password_admin` varchar(32) COLLATE latin1_general_ci DEFAULT NULL, `vpassword_admin` varchar(32) COLLATE latin1_general_ci DEFAULT NULL, PRIMARY KEY (`id_admin`), KEY `id_superadmin` (`id_superadmin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci; ALTER TABLE `admin` ADD CONSTRAINT `admin_ibfk_1` FOREIGN KEY (`id_superadmin`) REFERENCES `superadmin` (`id_superadmin`);
2	Barang	CREATE TABLE IF NOT EXISTS `barang` (`id_barang` int(4) unsigned zerofill NOT NULL AUTO_INCREMENT, `id_kategori` int(11) NOT NULL, `id_merk` int(11) NOT NULL, `id_ukuran` int(11) NOT NULL,

No	Nama Tabel	Source Code
		`id_warna` int(11) NOT NULL, `nama_barang` varchar(50) NOT NULL, `harga_barang` int(11) NOT NULL, `stok_barang` int(11) NOT NULL, `berat_barang` float NOT NULL, `deskripsi_barang` text NOT NULL, `diskon_barang` float NOT NULL, `rating_barang` float NOT NULL, `voterrating_barang` int(11) NOT NULL, `viewcounter_barang` int(11) NOT NULL, `link_barang` varchar(200) NOT NULL, PRIMARY KEY (`id_barang`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Barang' AUTO_INCREMENT=17 ;
3.	Detail Pemesanan	CREATE TABLE IF NOT EXISTS `detailpemesanan` (`id_detailpemesanan` int(11) NOT NULL AUTO_INCREMENT, `id_member` int(11) DEFAULT NULL, `tanggal_detailpemesanan` datetime NOT NULL, `status_pengiriman` enum('dikeranjang','dipesan','dikonfirmasi','dibayar','dikirim','diterima','dibatalkan') NOT NULL, `keterangan_detailpemesanan` text NOT NULL, `nama_pemesan` varchar(50) NOT NULL, `alamat_pemesan` text NOT NULL, `email_pemesan` varchar(50) NOT NULL, `no_telp_pemesan` varchar(20) NOT NULL, `kodepos_pemesan` varchar(6) NOT NULL, `kota_pemesan` varchar(100) NOT NULL, `provinsi_pemesan` varchar(100) NOT NULL, `no_resi_pemesan` varchar(50) NOT NULL, `totalpembayaran_detailpemesanan` varchar(80) NOT NULL, `id_ongkir` int(11) NOT NULL, `hargaongkir_detailpemesanan` float NOT NULL, `status_pemesanan` enum('ok','cancel') NOT NULL, `session_id` varchar(32) NOT NULL, PRIMARY KEY (`id_detailpemesanan`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel detailpemesan' AUTO_INCREMENT=87 ;

No	Nama Tabel	Source Code
4	Gambar	CREATE TABLE IF NOT EXISTS `gambar` (`id_gambar` int(11) NOT NULL AUTO_INCREMENT, `id_barang` int(4) unsigned zerofill NOT NULL, `path_gambar` varchar(200) NOT NULL, `profile_gambar` enum('0','1') NOT NULL, PRIMARY KEY (`id_gambar`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Gambar' AUTO_INCREMENT=33 ;
5	History	CREATE TABLE IF NOT EXISTS `history` (`id_history` int(11) NOT NULL AUTO_INCREMENT, `id_admin` int(11) NOT NULL, `history_name` varchar(200) NOT NULL, `history_meta` text NOT NULL, `history_create` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP, PRIMARY KEY (`id_history`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel History' AUTO_INCREMENT=229
6	JasaPengiriman	CREATE TABLE IF NOT EXISTS `jasapengiriman` (`id_jasapengiriman` int(11) NOT NULL AUTO_INCREMENT, `nama_jasapengiriman` varchar(100) NOT NULL, `deskripsi_jasapengiriman` text NOT NULL, PRIMARY KEY (`id_jasapengiriman`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Jasa Pengiriman' AUTO_INCREMENT=4 ;
7	Jenis Pengiriman	CREATE TABLE IF NOT EXISTS `jenispengiriman` (`id_jenispengiriman` int(11) NOT NULL AUTO_INCREMENT, `id_jasapengiriman` int(11) NOT NULL, `nama_jenispengiriman` varchar(50) NOT NULL, `deskripsi_jenispengiriman` text NOT NULL, PRIMARY KEY (`id_jenispengiriman`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Jenis Pengiriman' AUTO_INCREMENT=8 ;
8	Kategori	CREATE TABLE IF NOT EXISTS `kategori` (

No	Nama Tabel	Source Code
		`id_kategori` int(11) NOT NULL AUTO_INCREMENT, `kode_kategori` varchar(3) NOT NULL, `nama_kategori` varchar(50) NOT NULL, `deskripsi_kategori` text NOT NULL, PRIMARY KEY (`id_kategori`), UNIQUE KEY `nama_kategori` (`nama_kategori`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Kategori' AUTO_INCREMENT=16 NULL AUTO_INCREMENT, `kode_kategori` varchar(3) NOT NULL, `nama_kategori` varchar(50) NOT NULL, `deskripsi_kategori` text NOT NULL, PRIMARY KEY (`id_kategori`), UNIQUE KEY `nama_kategori` (`nama_kategori`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Kategori' AUTO_INCREMENT=16
9	Jasa pengiriman	CREATE TABLE IF NOT EXISTS `jasapengiriman` (`id_jasapengiriman` int(11) NOT NULL AUTO_INCREMENT, `nama_jasapengiriman` varchar(20) COLLATE latin1_general_ci DEFAULT NULL, PRIMARY KEY (`id_jasapengiriman`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci AUTO_INCREMENT=1 ;
10	Kota	CREATE TABLE IF NOT EXISTS `kota` (`id_kota` int(10) NOT NULL AUTO_INCREMENT, `id_provinsi` int(10) DEFAULT NULL, `nama_kota` varchar(50) DEFAULT NULL, `kabkota` varchar(20) DEFAULT NULL, UNIQUE KEY `kota#PX` (`id_kota`), KEY `id_provinsi` (`id_provinsi`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=427 ;
11	Member	CREATE TABLE IF NOT EXISTS `member` (`id_member` int(11) NOT NULL AUTO_INCREMENT, `id_kota` int(11) NOT NULL,

No	Nama Tabel	Source Code
		`nama_member` varchar(50) NOT NULL, `alamat_member` text NOT NULL, `ttl_member` date NOT NULL, `telp_member` varchar(20) NOT NULL, `kodepos_member` varchar(6) NOT NULL, `email_member` varchar(50) NOT NULL, `password_member` varchar(32) NOT NULL, `status_member` enum('0','1') NOT NULL, PRIMARY KEY (`id_member`), UNIQUE KEY `email_member` (`email_member`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Member' AUTO_INCREMENT=29 ;
12	Merk	CREATE TABLE IF NOT EXISTS `merk` (`id_merk` int(11) NOT NULL AUTO_INCREMENT, `nama_merk` varchar(100) NOT NULL, `keterangan_merk` text NOT NULL, PRIMARY KEY (`id_merk`), UNIQUE KEY `nama_merk` (`nama_merk`), UNIQUE KEY `nama_merk_2` (`nama_merk`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Table Merk' AUTO_INCREMENT=33 ;
13	Ongkos Kirim	CREATE TABLE IF NOT EXISTS `ongkir` (`id_ongkir` int(11) NOT NULL AUTO_INCREMENT, `id_kota` int(11) NOT NULL, `id_jenispengiriman` int(11) NOT NULL, `harga_ongkir` int(11) NOT NULL, PRIMARY KEY (`id_ongkir`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Ongkos Kirim' AUTO_INCREMENT=15
14	Pembayaran	CREATE TABLE IF NOT EXISTS `pembayaran` (`id_pembayaran` int(11) NOT NULL AUTO_INCREMENT, `id_detailpemesanan` int(11) NOT NULL, `payment_gateway` enum('TRANSFER','PAYPAL') NOT NULL, `bank_rekening` varchar(50) DEFAULT NULL, `no_rekening` varchar(50) DEFAULT NULL, `instansi_pembayaran` varchar(25) NOT NULL, `no_transaksi` varchar(25) NOT NULL,

No	Nama Tabel	Source Code
		`totalbayar_pembayaran` float NOT NULL, `kurs_pembayaran` float DEFAULT NULL, `tanggal_pembayaran` datetime NOT NULL, PRIMARY KEY (`id_pembayaran`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Konfirmasi' AUTO_INCREMENT=5 ;
15	Pemesanan	CREATE TABLE IF NOT EXISTS `pemesanan` (`id_pemesanan` int(11) NOT NULL AUTO_INCREMENT, `id_detailpemesanan` int(11) DEFAULT NULL, `id_barang` int(11) NOT NULL, `harga_temp` int(11) NOT NULL, `stok_temp` int(11) NOT NULL, `berat_temp` float NOT NULL, `diskon_temp` float NOT NULL, PRIMARY KEY (`id_pemesanan`), KEY `id_detailpembelian` (`id_detailpemesanan`), KEY `id_detailbarang` (`id_barang`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Pemesanan' AUTO_INCREMENT=4 ;
16	Rekening	CREATE TABLE IF NOT EXISTS `rekening` (`id_rekening` int(11) NOT NULL AUTO_INCREMENT, `nama_rekening` varchar(50) NOT NULL, `bank_rekening` varchar(50) NOT NULL, `cabang_rekening` varchar(100) NOT NULL, `no_rekening` varchar(32) NOT NULL, `gambar_rekening` varchar(200) NOT NULL, PRIMARY KEY (`id_rekening`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Rekening' AUTO_INCREMENT=6 ;
17	Report	CREATE TABLE IF NOT EXISTS `report` (`id_report` int(11) NOT NULL AUTO_INCREMENT, `id_barang` int(11) NOT NULL, `stok_report` int(11) NOT NULL, `tanggal_report` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP, `status_report` enum('in','out') NOT NULL, PRIMARY KEY (`id_report`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Report' AUTO_INCREMENT=3 ;

No	Nama Tabel	Source Code
18	Ukuran	CREATE TABLE IF NOT EXISTS `ukuran` (`id_ukuran` int(11) NOT NULL AUTO_INCREMENT, `nama_ukuran` varchar(10) NOT NULL, `deskripsi_ukuran` text NOT NULL, PRIMARY KEY (`id_ukuran`), UNIQUE KEY `nama_ukuran` (`nama_ukuran`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Ukuran' AUTO_INCREMENT=5 ;
19	Retur	CREATE TABLE IF NOT EXISTS `retur` (`id_retur` int(11) NOT NULL AUTO_INCREMENT, `id_invoice` int(11) DEFAULT NULL, `id_admin` int(11) DEFAULT NULL, `status_retur` enum('dikonfirmasi','diterima','ditolak') COLLATE latin1_general_ci DEFAULT NULL, `pesan_retur` text COLLATE latin1_general_ci, `batas_retur` enum('ok','cancel') COLLATE latin1_general_ci DEFAULT NULL, `tanggal_retur` datetime DEFAULT NULL, PRIMARY KEY (`id_retur`), KEY `id_invoice` (`id_invoice`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci AUTO_INCREMENT=1 ; ALTER TABLE `retur` ADD CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`id_invoice`) REFERENCES `invoice` (`id_invoice`), ADD CONSTRAINT `retur_ibfk_2` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
20	Super*admin*	CREATE TABLE IF NOT EXISTS `superadmin` (`id_superadmin` int(11) NOT NULL AUTO_INCREMENT, `nama_super` varchar(20) COLLATE latin1_general_ci DEFAULT NULL, `alamat_super` text COLLATE latin1_general_ci, `kata sandi_super` varchar(100) COLLATE latin1_general_ci DEFAULT NULL, `vkata sandi_super` varchar(100) COLLATE latin1_general_ci DEFAULT NULL,

No	Nama Tabel	Source Code
		`email_superadmin` varchar(30) COLLATE latin1_general_ci DEFAULT NULL, PRIMARY KEY (`id_superadmin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci AUTO_INCREMENT=1 ;
21	Warna	CREATE TABLE IF NOT EXISTS `warna` (`id_warna` int(11) NOT NULL AUTO_INCREMENT, `nama_warna` varchar(50) NOT NULL, `format_warna` varchar(10) DEFAULT NULL, PRIMARY KEY (`id_warna`), UNIQUE KEY `nama_warna` (`nama_warna`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Tabel Warna' AUTO_INCREMENT=11 ;

4.3 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengunjung dan member, admin, superadmin. Adapun untuk tampilan implementasi pembangunan aplikasi *E-commerce* di Toko Istana Helmet Bandung akan dilampirkan pada halaman lampiran.

Tabel 4.4 Implementasi Antarmuka Pengunjung dan member

Menu	Deskripsi	Nama File
Registrasi member	Digunakan untuk melakukan pelanggan baru.	register.php
Login member	Digunakan untuk sebagai halaman login pelanggan	login.php
Lupa password	Digunakan jika pelanggan lupa dengan *password*.	forgotpassword.php
Ganti Profil	Digunakan untuk mengganti kontak member.	changeprofile.php
Ganti password	Digunakan untuk mengganti *password* member	changepassword.php
Informasi toko	Digunakan untuk melihat informasi toko istanahelmet	about.php

Peta denah toko	Digunakan untuk melihat peta lokasi toko istana helmet bandung	maps.php
History pesanan	Digunakan untuk melihat *history* pemesanan yang pernah dilakukan.	order.php
Pembayaran	Digunakan untuk pembayaran barang yang di pesan.	Payment.php
Keranjang Belanja	Digunakan untuk melihat, merubah, menghapus data pemesanan yang akan dilakukan.	chart.php
Ongkos kirim	Digunakan untuk melihat, data ongkos kirim.	shipping.php
Detail barang	Digunakan untuk melihat Detail barang perbarang	view.php
Barang	Digunakan untuk melihat, data seluruh barang.	menu.php
Logout	Digunakan member untuk keluar dari akunya.	logout.php

Tabel 4.5 Implementasi Antarmuka admin

Menu	Deskripsi	Nama File
Login member	Digunakan sebagai halaman *login* member.	login.php
Lupa password	Digunakan jika operator lupa dengan *password*.	login.php
Profil	Digunakan untuk melimengubah data profil dan data *password*	setting-profile.php
Profil Admin	Digunakan untuk melihat data Admin	admin.php
Profil Member	Digunakan untuk melihat dan menghapus data member.	member.php
Kategori	Digunakan untuk melihat data kategori.	category.php
	Digunakan untuk menambah dan merubah data kategori	category_form.php
Barang	Digunakan untuk melihat data barang	good.php
	menambah, dan menghapus data barang.	good_form.php
Merk	Digunakan untuk melihat data merk.	brand.php
	menambah, dan menghapus data	brand_form.php

	merk.	
Ukuran	Digunakan untuk melihat data ukuran.	size.php
	menambah, dan menghapus data ukuran.	size_form.php
Warna	Digunakan untuk melihat data warna.	color.php
	menambah, dan menghapus data warna.	color_form.php
Provinsi	Digunakan untuk melihat data provinsi.	province.php
	menambah, dan menghapus data provinsi.	province_form.php
Kota	Digunakan untuk melihat data kota.	city.php
	menambah, dan menghapus data kota.	city_form.php
Jasa pengiriman	Digunakan untuk melihat data jasa pengiriman.	service.php
	menambah, dan menghapus data jenis pengiriman.	service_form.php
Jenis Pengiriman	Digunakan untuk melihat data jenis pengiriman.	typedelivery.php
	menambah, dan menghapus data jenis pengiriman.	typedelivery_form.php
Rekening	Digunakan untuk melihat data jenis pengiriman.	bank.php
	Digunakan untuk menambah data Rekening	bank_form.php
Pesanan	Digunakan untuk melihat, dan mengolah data pesanan.	order.php
Ongkos kirim	Digunakan untuk melihat data jenis pengiriman.	shippingcost.php
	Digunakan untuk menambah data Rekening.	shippingcost_from.php
Transaksi	Digunakan untuk melihat data transaksi mengubah status.	transaction.php
Pembayaran	Digunakan untuk melihat data status pembayaran.	order.php
Retur	Digunakan sebagai halaman mengelola retur	retur.php
Laporan Harian	Digunakan untuk melihat data laporan berdasarkan penjualan perhari dan cetak laporan.	report-daily.php
Laporan Bulanan	Digunakan untuk melihat data	report-month.php

	laporan berdasarkan penjualan perbulan dan cetak laporan.	
Laporan Tahunan	Digunakan untuk melihat data laporan berdasarkan penjualan pertahun dan cetak laporan.	report-yearly.php
Back up	Digunakan untuk memback-up database.	backup.php
Restore	Digunakan untuk menambah database.	Restore.php
Keluar	Digunakan admin untuk dapat keluar dari aplikasi.	login.php

Tabel 4.6 Implementasi Antarmuka Superadmin

Menu	Deskripsi	Nama File
Login Superadmin	Digunakan sebagai halaman *login* superadmin.	login.php
Lupa password	Digunakan jika superadmin lupa dengan *password.*	forgotpassword.php
Profil	Digunakan untuk mengubah data profil dan data *password*	setting-profile.php
Admin	Digunakan untuk melihat, menambah, dan merubah data operator.	employee.php
member	Digunakan untuk melihat data pelanggan.	member.php
Laporan	Digunakan untuk melihat laporan.	report.php
Keluar	Digunakan untuk keluar dari menu Superadmin	login.php

4.4 Pengujian Perangkat Lunak

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

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

4.5 Pengujian Alpa

Pengujian alpa dilakukan menggunakan metode *black box*. Untuk menentukan pengujian alpa ini dibagi menjadi tiga, yaitu skenario pengujian aplikasi pelanggan dan skenario pengujian aplikasi *administrator* dan petugas.

4.5.1 Skenario Pengujian Alpa

Tabel 4.7 Skenario Pengujian Aplikasi Pelanggan

Kelas Uji	Butir Uji	Jenis Pengujian
Registrasi member	Menampilkan *form* user baru	*Black Box*
	Isi *form* pendaftaran	*Black Box*
	Simpan data baru	*Black Box*
Login	Isi data login	*Black Box*
	Verifikasi Email	*Black Box*
	Verifikasi kata sandi	*Black Box*
Lupa Password	Menampilkan *form* lupa password	*Black Box*
	Isi Data lupa password	*Black Box*
	Kirim permintaan	*Black Box*
Ubah Profil	Menampilkan *form* ubah profil	*Black Box*
	Ubah data profil	*Black Box*
	Simpan perubahan	*Black Box*
Ganti *Password*	Menampilkan *form* ganti *password*	*Black Box*

	Isi *form* ganti *password*	*Black Box*
	Simpan data baru	*Black Box*
History	Menampilkan history transaksi pembayaran	*Black box*
Informasi	Menampilkan Informasi Ivannashop	*Black box*
Keranjang Belanja	Menampilkan data pemesanan	*Black Box*
	Mengupdate data pemesanan	*Black Box*
	Menghapus data pemesanan	*Black Box*
	Menampilkan konfirmasi transaksi	*Black Box*
Isi Form Alamat Tujuan	Menampilkan form alamat tujuan	*Black Box*
Isi From Pembayaran	Menampilkan form pembayaran	*Black box*
Retur	Menampilkan *form* retur	*Black Box*
Logout	Keluar dari menu pelanggan	*Black Box*

Tabel 4.8 Skenario Pengujian Aplikasi admin

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Login	Isi data *login*	*Black Box*
	Verifikasi *Email*	*Black Box*
	Verifikasi *Password*	*Black Box*
Lupa *password*	Menampilkan *form* lupa *password*	*Black Box*
Pemesanan	Tampilkan data pemesanan	*Black Box*
	Ubah status pemesanan	*Black Box*
Kategori	Tampilkan data kategori	*Black Box*
	Tambah data kategori	*Black Box*
	Ubah data kategori	*Black Box*
Provinsi	Tampilkan data provinsi	*Black Box*
	Tambah data provinsi	*Black Box*
	Ubah data provinsi	*Black Box*
Kota	Tampilkan data kota	*Black Box*
	Tambah data kota	*Black Box*
	Ubah data kota	*Black Box*
Barang	Tampilkan data barang	*Black box*
	Tambah data barang	*Black box*

	Ubah data barang	*Black box*
Detail barang	Tampilkan detail barang	*Black box*
	Tambah detail barang	*Black box*
	Ubah detail barang	*Black box*
Data laporan	Tampilkan data laporan	*Black box*
	Cari data laporan	*Black box*
	Cetak data laporan	*Black box*
Harga	Tampilakn data harga	*Black box*
	Tambah data harga	*Black box*
	Hapus data harga	*Black box*
Diskon	Tampilkan data diskon	*Black box*
	Tambah data diskon	*Black box*
	Hapus data diskon	*Black box*
Ukuran	Tampilkan data ukuran	*Black box*
	Tambah data ukuran	*Black box*
	Ubah data ukuran	*Black box*
Retur	Detail dataretur	*Black Box*
	Terima data retur	*Black Box*
	Tolak data retur	*Black Box*
Logout	Menghapus *session user*	*Black box*

Tabel 4.9 Skenario Pengujian Aplikasi Superadmin

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Login	Isi data *login*	*Black Box*
	Verifikasi *Email*	*Black Box*
	Verifikasi *Password*	*Black Box*
Lupa *password*	Menampilkan *form* lupa *password*	*Black Box*
admin	Tampilkan data admin	*Black box*
	Tambah data admin	*Black box*
	Ubah data admin	*Black box*
Member	Tampilkan data member	*Black box*
Profil	Tampilkan data profil	*Black box*
	Ubah data profil	*Black box*
Logout	Keluar dari *session user*	*Black box*

4.5.2 Kasus Dan Hasil Pengujian

4.5.2.1 Pengujian Daftar *Member*

Berikut ini merupakan pengujian aplikasi daftar *member* dan dapat dilihat pada tabel 4.10 dibawah ini:

Tabel 4.10 Pengujian Daftar *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan menu sign up	Tampil *form* buat pendaftaran akun baru	Dapat menampilkan *form* buat akun baru	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh masukan : Username : matmat Email : mamat@yahoo.com Province : Jawa Barat city : Bandung Addres : Lembang Kodepos : 40766 Phone : 08666561864 password : lelemamat confrim password : lelemamat	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Tekan tombol Sign Up	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi. Jika data masukan benar muncul pesan “Terima Kasih Telah Melakukan Pendaftaran”	Tampil pesan “Terima Kasih Telah Melakukan Pendaftaran” jika data masukan benar dan dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Memasukan data kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data Belum Diisi”	[√] Diterima [ ] Ditolak

4.5.2.2 Pengujian Masuk *Member*

Berikut ini merupakan pengujian masuk *member* dan dapat dilihat pada tabel 4.11 dibawah ini:

Tabel 4.11 Pengujian Masuk *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
email : mamat@yahoo.com	Mengisikan data alamat *email* pada *field* alamat *email*. Jika salah maka akan muncul pesan "Username dan kata sandi tidak ditemukan"	Dapat mengisi *field* alamat *email* dan muncul pesan "email dan kata sandi tidak ditemukan" jika salah dalam pengisian data	[√] Diterima [ ] Ditolak
password : lelemamat	Mengisikan data kata sandi pada *field* kata sandi. Jika salah maka akan muncul pesan "diantara 6 sampai 11 karakter diperbolehkan"	Dapat mengisi *field*kata sandi dan muncul pesan "diantara 6 sampai 11 karakter diperbolehkan" jika salah dalam pengisian data	[√] Diterima [ ] Ditolak
Tekan tombol sign in	Data *email* dan data kata sandi dicari	Tombol masuk dapat berfungsi	[√] Diterima [ ] Ditolak
Data *email* atau kata sandi tidak terdaftar	Tidak dapat login	Muncul pesan "*email* atau kata sandi salah"	[√] Diterima [ ] Ditolak
Data *email* atau kata sandi kosong	Tidak dapat login	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

4.5.2.3 Pengujian Keluar

Berikut ini merupakan pengujian keluar dan dapat dilihat pada tabel 4.12 dibawah ini:

Tabel 4.12 Pengujian Keluar

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan logout	Dapat keluar dan menghapus session pengguna	*User* keluar dari aplikasi	[√] Diterima [ ] Ditolak

4.5.2.4 Pengujian Lupa Kata Sandi

Berikut ini merupakan pengujian lupa kata sandi *member* dan dapat dilihat pada tabel 4.13 dibawah ini:

Tabel 4.13 Pengujian Lupa Kata Sandi

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan forgot your password	Dapat menampilkan *form* lupa kata sandi	Muncul *form* lupa kata sandi	[√] Diterima [ ] Ditolak
Tekan kirim *email*	Dapat menampilkan *form* lupa kata sandi	Mengirimkan pesan ke email untuk verifikasi kata sandi baru	[√] Diterima [ ] Ditolak

4.5.2.5 Pengujian Ubah Profil *Member*

Berikut ini merupakan pengujian ubah akun*member*dan dapat dilihat pada tabel 4.14 dibawah ini:

Tabel 4.14 Pengujian Ubah Profil *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan change profil	Menampilkan *form* ubah akun	Dapat menampilkan *form* ubah akun	[√] Diterima [ ] Ditolak
Data profil yang akan diubah	Tercantum pada text box data yang akan	Dapat mengisi *field* data yang	[√] Diterima [ ] Ditolak

	diubah	akan diubah dengan benar	
Tekan change profile	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[√] Diterima [ ] Ditolak
Data masukan ada yang tidak diisi	Menampilkan pesan kesalahan bahwa field tersebut harus terisi.	Dapat menampilkan pesan kesalahan bahwa field tersebut harus terisi.	[√] Diterima [ ] Ditolak
Data masukan semuanya tidak diisi	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

4.5.2.6 Pengujian Ganti Kata Sandi *Member*

Berikut ini merupakan pengujian ubah kata sandi *member* dan dapat dilihat pada tabel 4.15 dibawah ini:

Tabel 4.15 Pengujian Ganti Kata Sandi *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan change password	Menampilkan *form* ubah kata sandi	Dapat menampilkan *form* ubah kata sandi	[√] Diterima [ ] Ditolak
Data masukan semuanya benar	Kata sandi baru berhasil disimpan di *database*	Dapat menyimpan data kata sandi yang baru dengan muncul pesan " Data berhasil diubah"	[√] Diterima [ ] Ditolak
Salah satu *field* belum terisi	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan "*field* ini harus diisi" bila ada field yang belum diisi	[√] Diterima [ ] Ditolak

Data old Password	muncul pesan "kata sandi salah"	Dapat menampilkan pesan kesalahan ”kata sandi salah”	[√] Diterima [ ] Ditolak
Data kata sandi baru salah atau konfirmasi kata sandi baru tidak sama	Muncul pesan "kata sandi baru dan konfirmasi kata sandi harus sama"	Dapat menampilkan pesan "kata sandi baru dan konfirmasi kata sandi harus sama"	[√] Diterima [ ] Ditolak
Data kosong	Tidak terjadi perubahan kata sandi	Tidak menampilkan pesan apapun.	√] Diterima [ ] Ditolak

4.5.2.7 Pengujian Menu Daftar Barang

Berikut ini merupakan pengujian menu daftar barang dan dapat dilihat pada tabel 4.16 dibawah ini:

Tabel 4.16 Pengujian Menu Daftar Barang

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan menu daftar barang	Menampilkan data barang yang ada di *database*	Dapat menampilkan data barang yang ada di *data base*	[√] Diterima [ ] Ditolak
Tekan detail pada data barang	Menampilkan detail barang	Dapat menampilkan detail barang	[√] Diterima [ ] Ditolak
Tekan add to basket	Menuju ke view chart	Dapat menampilkan keranjang belanja	[√] Diterima [ ] Ditolak

4.5.2.8 Pengujian *Form* view chart

Berikut ini merupakan pengujian *form* keranjang belanja dan dapat dilihat pada tabel 4.17 dibawah ini:

Tabel 4.17 Pengujian *Form* view chart

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan menu view chart	Menampilkan *form* keranjang belanja	Dapat menampilkan *form* keranjang belanja	[√] Diterima [ ] Ditolak
Data jumlah	Tercantum pada text box data yang akan diubah	Dapat mengisi data jumlah dengan benar	[√] Diterima [ ] Ditolak
Tekan tombol ubah	Data jumlah yang diubah tersimpan *didatabase*	Dapat menampilkan perubahan data jumlah	[√] Diterima [ ] Ditolak
Tekan tombol hapus	Data yang dihapus terhapus dari keranjang belanja	Dapat menghapus data pada keranjang belanja	[√] Diterima [ ] Ditolak
Tekan lanjutkan	Menampilkan *form* pengiriman	Dapat menampilkan *form* pengiriman	[√] Diterima [ ] Ditolak

4.5.2.9 Pengujian *Form* Payment confirmation

Berikut ini merupakan pengujian *form* konfirmasi pembayaran dan dapat dilihat pada tabel 4.18 dibawah ini:

Tabel 4.18 Pengujian *Form* Payment confirmation

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Masukan data konfirmasi pembayaran, contoh No. invoice : 13 Nama Bank : Bca Atas Nama :Rachmat Saeful Edi No.Transaksi : 32 Jumlah Transfer : 250000	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak

Memasukan data kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi”	[√] Diterima [ ] Ditolak

4.5.2.10 Pengujian Menu *Order*

Berikut ini merupakan pengujian *form* history belanja dan dapat dilihat pada tabel 4.19 dibawah ini:

Tabel 4.19 Pengujian Menu *Order*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan menu *order*	Menampilkan *order*	Dapat menampilkan *form order*	[√] Diterima [ ] Ditolak
Tekan detail	Menampilkan detail transaksi	Dapat menampilkan detail transaksi	[√] Diterima [ ] Ditolak

4.5.2.11 Pengujian *Form Retur*

Berikut ini merupakan pengujian dari data retur dan dapat dilihat pada tabel 4.20 dibawah ini:

Tabel 4.20 Pengujian *Form Retur*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan *link retur*	Menampilkan *form* pengolahan data *retur*	Dapat menampilkan *form retur*	[√] Diterima [ ] Ditolak
Tekan *link* lihat detail	Menampilkan *form* detail retur	Dapat menampilkan *form* detail *retur*	[√] Diterima [ ] Ditolak
Tekan Tombol kirim *retur*	Merubah status *retur*	Dapat merubah status *retur*	[√] Diterima [ ] Ditolak

4.5.2.12 Pengujian Masuk *Super Administrator*

Berikut ini merupakan pengujian *form* masuk *super administrator* dan dapat dilihat pada tabel 4.21 dibawah ini:

Tabel 4.21 Pengujian Masuk *Super Administrator*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
*Email:*super*admin*@yahoo.com	Tercantum pada text box *email*	Dapat mengisi field *email*dengan benar	[√] Diterima [ ] Ditolak
password : super*admin*	Tercantum pada text box kata sandi	Dapat mengisi field kata sandi dengan benar	[√] Diterima [ ] Ditolak
Tekan tombol masuk	Data *email* dan data kata sandi dicari	Tombol login dapat berfungsi	[√] Diterima [ ] Ditolak
Data *email* atau password tidak terdaftar	Tidak dapat login	Muncul pesan "Username or Kata sandi Incorrect."	[√] Diterima [ ] Ditolak
Data *email* atau password kosong	Tidak dapat login	Dapat menampilkan pesan kesalahan "Username or Kata sandi Incorrect."	[√] Diterima [ ] Ditolak

4.5.2.13 Pengujian Data *Super Adminstrator*

Berikut ini merupakan pengujian data *super administrator* pembayaran dan dapat dilihat pada tabel 4.22 dibawah ini:

Tabel 4.22 Pengujian Data *Super Administrator*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan manage *admin*	Menampilkan data super *admin* maupun *admin*	Dapat menampilkan data super *admin* dan *admin*	[√] Diterima [ ] Ditolak
Tekan ubah profil	Merubah data akun super *admin*	Dapat merubah data super*admin* muncul pesan "data berhasil disimpan"	[√] Diterima [ ] Ditolak
Tekan ubah password	Merubah kata sandi super *admin*istrator	Tampil pesan "kata sandi berhasil disimpan"	[√] Diterima [ ] Ditolak

4.5.2.14 Pengujian Keluar *Super Administrator*

Berikut ini merupakan pengujian keluar*super administrator* dan dapat dilihat pada tabel 4.23 dibawah ini:

Tabel 4.23 Pengujian Keluar *Super Admini*strator

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Kik logout	Dapat keluar dan menghapus session pengguna	*User* keluar dari aplikasi	[√] Diterima [ ] Ditolak

4.5.2.15 Pengujian Data *Admin*istrator

Berikut ini merupakan pengujian data *administrator* dan dapat dilihat pada tabel 4.24 dibawah ini:

Tabel 4.24 Pengujian Data *Admin*istrator

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan manage *administrator*	Menampilkan data *administrator*	Dapat menampilkan data *administrator*	[√] Diterima [ ] Ditolak
Tekan delete	Menghapus data *administrator*	Dapat menghapus data *administrator* dan muncul pesan"*admin* berhasil dihapus"	[√] Diterima [ ] Ditolak

4.5.2.16 Pengujian Tambah Data *Administrator*

Berikut ini merupakan pengujian *form* tambah data *administrator* dan dapat dilihat pada tabel 4.25 dibawah ini:

Tabel 4.25 Pengujian Tambah Data *Administrator*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan tombol tambah *admin*	Menampilkan *form* tambah *administrator*	Dapat menampilkan *form administrator*	[√] Diterima [ ] Ditolak
Mengisi tambah *admin*istrator Contoh: username : *admin*1 email : *admin*1@yahoo.com password : *admin*1	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data *admin*istrator	Data	[√] Diterima

	akan tersimpan didalam *database*	*admin*istrator akan tersimpan didalam *database* dan muncul pesan ”*admin* berhasil disimpan”	[ ] Ditolak
Data *admin* masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”data belum diisi”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.17 Pengujian Masuk *Administrator*

Berikut ini merupakan pengujian *form* masuk *administrator* dan dapat dilihat pada tabel 4.26 dibawah ini:

Tabel 4.26 Pengujian Masuk *Administrator*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
email : *admin*1@yahoo.com	Tercantum pada text box *email*	Dapat mengisi field *email* dengan benar	[√] Diterima [ ] Ditolak
password : *admin*1	Tercantum pada text box kata sandi	Dapat mengisi field kata sandi dengan benar	[√] Diterima [ ] Ditolak
Tekan tombol masuk	Data *email* dan data kata sandi dicari	Tombol masuk dapat berfungsi	[√] Diterima [ ] Ditolak
Data *email* atau kata sandi tidak terdaftar	Tidak dapat login	Muncul pesan “Username or Kata sandi Incorrect.”	[√] Diterima [ ] Ditolak
Data *email* atau kata sandi kosong	Tidak dapat login	Dapat menampilkan pesan kesalahan “Username or Kata sandi Incorrect.”	[√] Diterima [ ] Ditolak

4.5.2.18 Pengujian Data Barang

Berikut ini merupakan pengujian *form* data barang dan dapat dilihat pada tabel 4.27 dibawah ini:

Tabel 4.27 Pengujian Data Barang

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan manage barang	Menampilkan data barang	Dapat menampilkan data barang	[√] Diterima [ ] Ditolak
Tekan ubah barang	Menampilkan *form*ubah data barang	Dapat menampilkan *form*ubah data barang	[√] Diterima [ ] Ditolak
Tekan hapus barang	Menghapus data barang	Dapat menghapus data barang dari *database*	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data barang atau kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”data masih kosong”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil disimpan”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.19 Pengujian Tambah Barang

Berikut ini merupakan pengujian *form* tambah data barang dan dapat dilihat pada tabel 4.28 dibawah ini:

Tabel 4.28 Pengujian Tambah Barang

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan tombol tambah	Menampilkan *form* tambah barang	Dapat menampilkan *form* tambah barang	[√] Diterima [ ] Ditolak
Mengisi data barang Contoh: kategori: Hallface Merk : INK Size : M Color : Blue Name : Simonnceli Harga : 400000 Diskon : 0 % Stok : 2 Gambar :	Jika salah satu *field* tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa *field* tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa *field* tersebut harus terisi jika salah satu *field* tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data barang atau masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data barang atau detail barangakan tersimpan didalam *database*	Data barang atau detail barangakan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.20 Pengujian Data Kategori

Berikut ini merupakan pengujian *form* data kategori dan dapat dilihat pada tabel 4.29 dibawah ini:

Tabel 4.29 Pengujian Data Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan manage kategori	Menampilkan *form* lihat kategori	Dapat menampilkan *form* lihat kategori	[√] Diterima [ ] Ditolak
Tekan ubah	Menampilkan *form* ubah kategori	Dapat menampilkan *form* ubah kategori	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data kategori kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.21 Pengujian Tambah Kategori

Berikut ini merupakan pengujian *form* tambah kategori dan dapat dilihat pada tabel 4.30 dibawah ini:

Tabel 4.30 Pengujian Tambah Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan tambah	Menampilkan *form* tambah kategori	Dapat menampilkan *form* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data kategori Contoh: Nama kategori: INK	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data kategori masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data pengirimanakan tersimpan didalam *database*	Data pengirimanakan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.22 Pengujian Data Ongkos Kirim

Berikut ini merupakan pengujian *form* data ongkos kirim dan dapat dilihat pada tabel 4.31 dibawah ini:

Tabel 4.31 Pengujian Data Ongkos Kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan manage Ongkos kirim	Menampilkan data ongkos kirim	Dapat menampilkan data ongkos kirim	[√] Diterima [ ] Ditolak
Tekan ubah	Menampilkan *form* ubah ongkos kirim	Dapat menampilkan *form*ongkos kirim	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data pengiriman kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.23 Pengujian Tambah Ongkos Kirim

Berikut ini merupakan pengujian *form* tambah ongkos kirim dan dapat dilihat pada tabel 4.32 dibawah ini:

Tabel 4.32 Pengujian Tambah Ongkos Kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan tambah	Menampilkan *form*tambah ongkos kirim	Dapat menampilkan *form* tambah ongkos kirim	[√] Diterima [ ] Ditolak
Mengisi data ongkos kirim Contoh: Provinsi : Jawa Barat Kota : Bandung Jasa pengiriman : JNE Jenis pengiriman : Regular Harga: 5000	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data pengiriman masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data pengirimanakan tersimpan didalam *database*	Data pengirimanakan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.24 Pengujian Data Provinsi

Berikut ini merupakan pengujian *form* data provinsidan dapat dilihat pada tabel 4.33 dibawah ini:

Tabel 4.33 Pengujian Data Provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan provinsi	Menampilkan data provinsi	Dapat menampilkan data provinsi	[√] Diterima [ ] Ditolak
Tekan ubah	Menampilkan *form* ubah provinsi	Dapat menampilkan *form* ubah provinsi	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data provinsi kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.25 Pengujian Tambah Provinsi

Berikut ini merupakan pengujian *form* tambah provinsidan dapat dilihat pada tabel 4.34 dibawah ini:

Tabel 4.34 Pengujian Tambah Provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan tambah provinsi	Menampilkan *form* tambah provinsi	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data provinsi Contoh: Nama Provinsi : Jawa Barat	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data provinsi masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data pengirimanakan tersimpan didalam *database*	Data pengirimanakan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.26 Pengujian Data Kota

Berikut ini merupakan pengujian *form* data kota dan dapat dilihat pada tabel 4.35 dibawah ini:

Tabel 4.35 Pengujian Data Kota

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan kota	Menampilkan data kota	Dapat menampilkan data kota	[√] Diterima [ ] Ditolak
Tekan ubah	Menampilkan *form* ubah kota	Dapat menampilkan *form* ubah kota	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data kota kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.27 Pengujian Tambah Kota

Berikut ini merupakan pengujian *form* tambah kota dan dapat dilihat pada tabel 4.36 dibawah ini:

Tabel 4.36 Pengujian Tambah Kota

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan baru	Menampilkan *form* tambah kota	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data kota Contoh: Provinsi : Jawa Barat Nama Kota : Bandung	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data kota masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data pengirimanakan tersimpan didalam *database*	Data pengirimanakan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.28 Pengujian Data *Member*

Berikut ini merupakan pengujian *form* data *member* dan dapat dilihat pada tabel 4.37 dibawah ini:

Tabel 4.37 Pengujian Data *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan *member*	Menampilkan data member	Dapat menampilkan data member	[√] Diterima [ ] Ditolak
Tekan *check box* pada data yang akan dihapus hapus	Memilih data yang akan dihapus	Data yang akan dihapus telah dipilih	[√] Diterima [ ] Ditolak
Tekan hapus	Data akan dihapus dari *database*. Jika tidak ada data yang dipilih akan muncul pesan kesalahan	Data telah dihapus dari database. Muncul pesan "pilih data yang akan dihapus" bila data belum dipilih	[√] Diterima [ ] Ditolak

4.5.2.29 Pengujian Data Jasa Pengiriman

Berikut ini merupakan pengujian *form* jasa pengiriman dan dapat dilihat pada tabel 4.38 dibawah ini:

Tabel 4.38 Pengujian Data Jasa Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan jasa pengiriman	Menampilkan *form* jasa pengiriman	Dapat menampilkan *form* jasa pengiriman	[√] Diterima [ ] Ditolak
Tekan ubah	Menampilkan *form* ubah jasa pengiriman	Dapat menampilkan *form* ubah jasa pengiriman	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak

Data jasa pengiriman kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.30 Pengujian Tambah Jasa Pengiriman

Berikut ini merupakan pengujian *form* tambah jasa pengiriman dan dapat dilihat pada tabel 4.39 dibawah ini:

Tabel 4.39 Pengujian Tambah Jasa Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan tambah	Menampilkan *form* tambah jasa pengiriman	Dapat menampilkan *form* tambah jasa pengiriman	[√] Diterima [ ] Ditolak
Mengisi data jasa pengiriman Contoh: Nama Jasa Pengiriman : JNE Deskripsi Jasa Pengiriman : JNE - Express Across Nations	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data jasa pengiriman masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak

Tekan tombol simpan	Data jasa pengirimanakan tersimpan didalam *database*	Data jasa pengirimanakan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.31 Pengujian Data Jenis Pengiriman

Berikut ini merupakan pengujian *form*jenis pengiriman dan dapat dilihat pada tabel 4.40 dibawah ini:

Tabel 4.40 Pengujian Data Jenis Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan jenis pengiriman	Menampilkan *form* jenis pengiriman	Dapat menampilkan *form* jenis pengiriman	[√] Diterima [ ] Ditolak
Tekan ubah	Menampilkan *form* ubah jenis pengiriman	Dapat menampilkan *form* ubah jenis pengiriman	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data jenis pengiriman kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.32 Pengujian Tambah Jenis Pengiriman

Berikut ini merupakan pengujian *form* tambah jenis pengirimandan dapat dilihat pada tabel 4.41 dibawah ini:

Tabel 4.41 Pengujian Tambah Jenis Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan tambah	Menampilkan *form* tambah jenis pengiriman	Dapat menampilkan *form* tambah jenis pengiriman	[√] Diterima [ ] Ditolak
Mengisi data jenis pengiriman Contoh: Nama Jenis Pengiriman : ONS Deskripsi Jasa Pengiriman : One Night Service	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data jenis pengiriman masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data jenispengirimanakan tersimpan didalam *database*	Data jenispengirimanakan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.33 Pengujian Data Rekening

Berikut ini merupakan pengujian *form* data rekeningdan dapat dilihat pada tabel 4.42 dibawah ini:

Tabel 4.42 Pengujian Data Rekening

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan rekening	Menampilkan *form* rekening	Dapat menampilkan *form* rekening	[√] Diterima [ ] Ditolak
Tekan ubah	Menampilkan *form* ubah rekening	Dapat menampilkan *form* ubah rekening	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data rekening kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.34 Pengujian Tambah Rekening

Berikut ini merupakan pengujian *form* tambah rekeningdan dapat dilihat pada tabel 4.43 dibawah ini:

Tabel 4.43 Pengujian Tambah Rekening

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan tambah	Menampilkan *form* tambah rekening	Dapat menampilkan *form* tambah rekening	[√] Diterima [ ] Ditolak
Mengisi data rekening Contoh: Nama Bank : BRI Nama Pemilik Bank : Hari No Rek Bank : 000887675656 Cabang : Sukajadi	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data rekening masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data rekeningakan tersimpan didalam *database*	Data rekeningakan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.35 Pengujian Data Warna

Berikut ini merupakan pengujian *form* konfirmasi pembayarandan dapat dilihat pada tabel 4.44 dibawah ini:

Tabel 4.44 Pengujian Data Warna

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan warna	Menampilkan *form* warna	Dapat menampilkan *form* warna	[√] Diterima [ ] Ditolak
Tekan ubah	Menampilkan *form* ubah warna	Dapat menampilkan *form* ubah warna	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data warna kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.36 Pengujian Tambah Warna

Berikut ini merupakan pengujian *form* konfirmasi pembayarandan dapat dilihat pada tabel 4.45 dibawah ini:

Tabel 4.45 Pengujian Tambah Warna

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan tambah	Menampilkan *form* tambah warna	Dapat menampilkan *form* tambah warna	[√] Diterima [ ] Ditolak
Mengisi data warna Contoh: Nama warna : Biru *Form*at Warna : Biru	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data warna masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Tekan tombol simpan	Data rekeningakan tersimpan didalam *database*	Data rekeningakan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Tekan batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.37 Pengujian Data Transaksi

Berikut ini merupakan pengujian dari data transaksidan dapat dilihat pada tabel 4.46 dibawah ini:

Tabel 4.46 Pengujian Data Transaksi

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan transaksi	Menampilkan *form* transaksi	Dapat menampilkan *form* lihat transaksi	[√] Diterima [ ] Ditolak
Tekan ubah status transaksi	Menampilkan *form*ubah status transaksi	Dapat menampilkan *form*ubah status transaksi	[√] Diterima [ ] Ditolak
Tekan konfirmasi	Dapat melakukan proses perubahan status transaksi	Status transaksi berubah dari dipesan menjadi dibayar	[√] Diterima [ ] Ditolak
Masukan no resi dan tekan tombol simpan	Dapat melakukan proses perubahan status transaksi	Status transaksi berubah dari dibayar menjadi dikirim	[√] Diterima [ ] Ditolak
Tekan konfirmasi barang diterima	Dapat melakukan proses perubahan status transaksi	Status transaksi berubah dari dikirim menjadi diterima	[√] Diterima [ ] Ditolak

4.5.2.38 Pengujian Data Laporan Harian

Berikut ini merupakan pengujian dari data laporan hariandan dapat dilihat pada tabel 4.47 dibawah ini:

Tabel 4.47 Pengujian Data Laporan Harian

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan laporan harian	Menampilkan *form* laporan transaksi	Dapat menampilkan *form*transaksi	[√] Diterima [ ] Ditolak
Masukan tanggal	Menyimpan data	Dapat	[√] Diterima

	tanggal	menyimpan data tanggal	[ ] Ditolak
Tekan cari	Menampilkan data transaksi	Dapat menampilkan data tansaksi yang dicari	[√] Diterima [ ] Ditolak
Tekan print PDF	Dapat melakukan proses *download*	*Download* laporan berhasil	[√] Diterima [ ] Ditolak

4.5.2.39 Pengujian Data Laporan Bulanan

Berikut ini merupakan pengujian dari data laporan bulanandan dapat dilihat pada tabel 4.48 dibawah ini:

Tabel 4.48 Pengujian Data Laporan Bulanan

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan laporan bulanan	Menampilkan *form* laporan transaksi	Dapat menampilkan *form*transaksi	[√] Diterima [ ] Ditolak
Masukan bulan	Menyimpan data bulan	Dapat menyimpan data bulan	[√] Diterima [ ] Ditolak
Tekan cari	Menampilkan data transaksi	Dapat menampilkan data tansaksi yang dicari	[√] Diterima [ ] Ditolak
Tekan print PDF	Dapat melakukan proses *download*	*Download* laporan berhasil	[√] Diterima [ ] Ditolak

4.5.2.40 Pengujian Data Laporan Tahunan

Berikut ini merupakan pengujian dari data laporan tahunandan dapat dilihat pada tabel 4.49 dibawah ini:

Tabel 4.49 Pengujian Data Laporan Tahunan

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan laporan tahunan	Menampilkan *form* laporan transaksi	Dapat menampilkan *form*transaksi	[√] Diterima [ ] Ditolak
Masukan tahun	Menyimpan data tahun	Dapat menyimpan data tahun	[√] Diterima [ ] Ditolak
Tekan cari	Menampilkan data transaksi	Dapat menampilkan data tansaksi yang dicari	[√] Diterima [ ] Ditolak
Tekan print PDF	Dapat melakukan proses *download*	*Download* laporan berhasil	[√] Diterima [ ] Ditolak

4.5.2.41 Pengujian Data Retur

Berikut ini merupakan pengujian dari data retur dan dapat dilihat pada tabel 4.48 dibawah ini:

Tabel 4.48 Pengujian Data Retur

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tekan *link* kelola *retur*	Menampilkan *form* pengolahan data *retur*	Dapat menampilkan *form retur*	[√] Diterima [ ] Ditolak
Tekan *link* konfirmasi	Menampilkan *form* detail retur	Dapat menampilkan *form* detail *retur*	[√] Diterima [ ] Ditolak
Tekan Tombol Terima	Merubah status *retur*	Dapat merubah status *retur*	[√] Diterima [ ] Ditolak
Tekan Tombol Tolak	Merubah status *retur*	Dapat merubah status *retur*	[√] Diterima [ ] Ditolak

4.5.3 Kesimpulan Pengujian Alpha

Berdasarkan hasi pengujian alpha (fungsional) dengan kasus uji diatas dapat ditarik kesimpulan bahwa pembangunan aplikasi e-commerce berbasis web di toko Istana Helmet bebas kesalahan proses dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

4.6 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan, dengan menggunakan kuesioner mengenai kepuasan user terhadap aplikasi yang telah dibangun, dengan kandungan point syarat *user friendly* serta apakah sesuai dengan yang diharapkan. Adapun metode penilaian pengujian yang digunakan yaitu metode kuantitatif berdasarkan data sampel dari *user*.

4.6.1 Skenario Pengujian Beta

Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung. Penelitian ini dilakukan di lingkungan masyarakat umum khususnya untuk member yang sudah terbiasa dengan teknologi *internet*, dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat memberikan kemudahan masyarakat umum untuk melakukan transaksi pembelian barang secara online.

Adapun beberapa poin yang diteliti adalah :

1. Tampilan antar muka yang mudah dipahami.
2. Membantu mencari barang yang diinginkan oleh pembeli.
3. Membantu proses pemesanan.

4. Membantu proses pembayaran.
5. Transparansi setiap perubahan transaksi kepada pembeli.

Website menjadi sarana memperluas pangsa pasar Seluruh kota di Indonesia. Berdasarkan data hasil kusioner tersebut, dapat dicari prosentase masing-masing jawaban dengan menggunakan rumus :

Y = P/Q * 100%

Keterangan :

P= Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai persentase

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

1. Apakah dengan adanya web e-commerce ini dapat membantu pelanggan melakukan transaksi pemesanan?

No Soal	Jawaban	Jumlah Jawaban
1	Sangat Membantu	7
	Membantu	2
	Cukup Membantu	1
	Biasa Saja	
	Kurang Membantu	
	Tidak membantu	
	Sngat Tidak Membantu	
		10

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 7 atau 70% menyatakan sangat membantu, 2 atau 20% menyatakan membantu, dan 1 atau 1% menyatakan cukup membantu bahwa proses transaksi pemesanan.

2. Apakah web e-commerce yang dibangun mudah dipergunakan?

No Soal	Jawaban	Jumlah Jawaban
2	Sangat Mudah Digunakan	3
	Mudah Digunakan	3
	Cukup Mudah Digunakan	4
	Biasa Saja	
	Kurang Mudah Digunakan	
	Tidak Mudah Digunakan	
	Sngat Tidak Mudah Digunakan	
		10

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 atau 30% menyatakan sangat mudah digunakan, 3 atau 30% menyatakan mudah digunakan. Dan 4 atau 4% menyatakan cukup mudah digunakan bahwa web ini mudah digunakan.

3. Apakah instruksi yang disediakan mudah dipelajari?

No Soal	Jawaban	Jumlah Jawaban
3	Sangat Mudah Dipelajari	4
	Mudah Dipelajari	4
	Cukup Mudah Dipelajari	2
	Biasa Saja	
	Kurang Mudah Dipelajari	
	Tidak Mudah Dipelajari	
	Sngat Tidak Mudah Dipelajari	
		10

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 4 atau 40% menyatakan sangat mudah dipelajari, 4 atau 40% menyatakan mudah dipelajari, dan 2 atau 2% menyatakan cukup mudah dipelajari bahwa intruksi yang di sediakan mudah dipelajari..

4. Apakah Informasi yang dihasilkan akurat sesuai dengan yang diinginkan?

No Soal	Jawaban	Jumlah Jawaban
4	Sangat Akurat	2
	Akurat	2
	Cukup Akurat	6
	Biasa Saja	
	Kurang Akurat	
	Tidak Akurat	
	Sangat Tidak Akurat	
		10

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 atau 20% menyatakan sangat akurat, 2 atau 20% menyatakan akurat, dan 6 atau 6% menyatakan cukup akurat bahwa informasi yang di hasilkan sesuai yang diinginkan.

5. Apakah tampilan aplikasi e-commerce ini menarik bagi pengguna dan mudah dipahami?

No Soal	Jawaban	Jumlah Jawaban
5	Sangat Dipahami	1
	Mudah Dipahami	6
	Cukup mudah Dipahami	3
	Biasa Saja	
	Kurang Dipahami	
	Tidak Mudah Dipahami	
	Sangat Tidak Dipahami	
		10

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 1 atau 10% menyatakan sangat dipahami, 6 atau 60% menyatakan mudah dipahami, dan 3 atau 3% menyatakan cukup mudah dipahami bahwa tampilan *website* cukup menarik dan mudah untuk dipahami.

4.6.2 Kesimpulan Pengujian Beta

Berdasarkan pengujian beta diatas, bahwa dapat diambil kesimpulan bahwa:

1. Pelanggan dapat melakukan transaksi barang di Toko Iatana Helmet dengan lebih mudah.
2. Aplikasi *E-Commerce* berbasis *web* di Toko Istana Helmet memudahkan pelanggan dalam mendapatkan in*form*asi barang yang tersedia.
3. Proses ecomerrce ini mudah dipelajari
4. Aplikasi ini juga memudahkan pihak toko dalam mendokumentasikan data master dan data transaksi penjualan barangnya.
5. Aplikasi ini juga memudahkan ini sangat menarik dan mudah dipahami.