BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

IV.1 Implementasi Sistem

Tahap implementasi sistem merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan menerjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dapat dimengerti oleh mesin serta penerapan perangka lunak pada keadaan yang sesungguhnya.

IV.1.1 Perangkat keras yang dignakan

Spesifikasi perangkat keras (*hardware*) yang digunakan dalam pembangunan aplikasi *E-Commerce* di toko Yella Perdana adalah tercantum pada tabel IV.1 sebagi berikut:

Tabel IV.1 Perangkat Keras Yang Digunakan

No	Perangkat Keras	Spesifikasi
1	Processor	Intel dual core 2.2 GHz
2	RAM	2048 MB
3	Harddisk	320 Gb
4	Mouse	-
5	Keyboard	-
6	monitor	14 inch

IV.2.1 Perangkat lunak yang digunakan

Spesifikasi perangkat lunak (*Software*) yang dipasang pada sistem komputer yang digunakan untuk mengembangkan sistem penerapan *E-Commerce* untuk sistem penjualan ini adalah tercantum pada tabel IV.2 sebagai berikut:

Tabel IV.2 Perangkat Lunak Yang Digunakan

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows 7
2	PHP	Bahasa Pemrograman
3	*MySQL*	Penyimpanan *database*
4	*PhpMyAdmin*	*interface*
5	*WampServer2.1 e-x32*	*Web server*
6	Macromedia Dreamweaver CS5	Kode dan perancangan tampilan
7	*Mozilla Firefor*	*Web browser*

IV.2 Impelementasi Basis Data

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

Tabel IV.3 Tabel Implementasi Data Base

1.	**Tabel Admin**	CREATE TABLE IF NOT EXISTS `admin` (`id_admin` int(11) NOT NULL AUTO_INCREMENT, `nama_admin` varchar(50) NOT NULL, `username` varchar(50) NOT NULL, `password` varchar(50) NOT NULL, `email` varchar(50) NOT NULL, `level` enum('admin','operator') NOT NULL DEFAULT 'operator', PRIMARY KEY (`id_admin`), UNIQUE KEY `username` (`username`,`email`)) ENGINE=InnoDB DEFAULT CHARSET=latin1

2.	**Tabel Bank**	CREATE TABLE IF NOT EXISTS `bank` (`id_bank` int(11) NOT NULL AUTO_INCREMENT, `nama_bank` varchar(50) NOT NULL, `id_admin` int(11) NOT NULL, PRIMARY KEY (`id_bank`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 ; ALTER TABLE `bank` ADD CONSTRAINT `bank_ibfk_1` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
3.	**Tabel Hubungi**	CREATE TABLE IF NOT EXISTS `hubungi` (`id_hubungi` int(11) NOT NULL AUTO_INCREMENT, `nama` varchar(50) NOT NULL, `email` varchar(50) NOT NULL, `subjek` varchar(50) NOT NULL, `pesan` text NOT NULL, `tgl` datetime NOT NULL, `id_member` int(11) DEFAULT NULL, PRIMARY KEY (`id_hubungi`), KEY `id_member` (`id_member`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 ; ALTER TABLE `hubungi` ADD CONSTRAINT `hubungi_ibfk_1` FOREIGN KEY (`id_member`) REFERENCES `member` (`id_member`);
4.	**Tabel** **Jasa_pengiriman**	CREATE TABLE IF NOT EXISTS `jasa_pengiriman` (`id_jasa_pengiriman` int(11) NOT NULL AUTO_INCREMENT, `nama_jasa_pengiriman` varchar(50) NOT NULL, `id_admin` int(11) NOT NULL, PRIMARY KEY (`id_jasa_pengiriman`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `jasa_pengiriman` ADD CONSTRAINT `jasa_pengiriman_ibfk_1` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
5.	**Tabel** **Kategori_pengiriman**	CREATE TABLE IF NOT EXISTS `kategori_pengiriman` (`id_kategori_pengiriman` int(11) NOT NULL AUTO_INCREMENT, `nama_kategori_pengiriman` varchar(50) NOT

		NULL, `id_jasa_pengiriman` int(11) NOT NULL, `id_admin` int(11) NOT NULL, PRIMARY KEY (`id_kategori_pengiriman`), KEY `id_jasa_pengiriman` (`id_jasa_pengiriman`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `kategori_pengiriman` ADD CONSTRAINT `kategori_pengiriman_ibfk_1` FOREIGN KEY (`id_jasa_pengiriman`) REFERENCES `jasa_pengiriman` (`id_jasa_pengiriman`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `kategori_pengiriman_ibfk_2` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
6.	**Table Provinsi**	CREATE TABLE IF NOT EXISTS `provinsi` (`id_provinsi` int(11) NOT NULL AUTO_INCREMENT, `nama_provinsi` varchar(50) NOT NULL, `id_admin` int(11) NOT NULL, PRIMARY KEY (`id_provinsi`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `provinsi` ADD CONSTRAINT `provinsi_ibfk_1` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
7.	**Tabel Kota**	CREATE TABLE IF NOT EXISTS `kota` (`id_kota` int(11) NOT NULL AUTO_INCREMENT, `nama_kota` varchar(50) NOT NULL, `id_provinsi` int(11) NOT NULL, `id_admin` int(11) NOT NULL, PRIMARY KEY (`id_kota`), KEY `id_provinsi` (`id_provinsi`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `kota` ADD CONSTRAINT `kota_ibfk_1` FOREIGN KEY (`id_provinsi`) REFERENCES `provinsi` (`id_provinsi`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `kota_ibfk_2` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);

8.	**Tabel Member**	CREATE TABLE IF NOT EXISTS `member` (`id_member` int(11) NOT NULL AUTO_INCREMENT, `nama_member` varchar(50) NOT NULL, `username` varchar(50) NOT NULL, `password` varchar(50) NOT NULL, `alamat` text NOT NULL, `email` varchar(50) NOT NULL, `telepon` varchar(20) NOT NULL, `id_kota` int(11) NOT NULL, `kode_pos` varchar(6) NOT NULL, `block` enum('ya','tidak') NOT NULL DEFAULT 'ya', `tgl_reg` datetime NOT NULL, `kode_verifikasi` varchar(50) NOT NULL, `id_admin` int(11) DEFAULT NULL, PRIMARY KEY (`id_member`), UNIQUE KEY `username` (`username`,`email`), KEY `id_kota` (`id_kota`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `member` ADD CONSTRAINT `member_ibfk_1` FOREIGN KEY (`id_kota`) REFERENCES `kota` (`id_kota`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `member_ibfk_2` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
9.	**Tabel Merk**	CREATE TABLE IF NOT EXISTS `merk` (`id_merk` int(11) NOT NULL AUTO_INCREMENT, `nama_merk` varchar(50) NOT NULL, `id_admin` int(11) NOT NULL, PRIMARY KEY (`id_merk`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `merk` ADD CONSTRAINT `merk_ibfk_1` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
10.	**Tabel Ongkir**	CREATE TABLE IF NOT EXISTS `ongkir` (`id_ongkir` int(11) NOT NULL AUTO_INCREMENT, `id_kota` int(11) NOT NULL, `id_kategori_pengiriman` int(11) NOT NULL, `tarif` double NOT NULL, `id_jasa_pengiriman` int(11) NOT NULL, `id_admin` int(11) NOT NULL,

		UNIQUE KEY (`id_ongkir`), KEY `id_kota` (`id_kota`,`id_kategori_pengiriman`), KEY `id_kategori_pengiriman` (`id_kategori_pengiriman`), KEY `id_jasa_pengiriman` (`id_jasa_pengiriman`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `ongkir` ADD CONSTRAINT `ongkir_ibfk_1` FOREIGN KEY (`id_kota`) REFERENCES `kota` (`id_kota`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `ongkir_ibfk_2` FOREIGN KEY (`id_kategori_pengiriman`) REFERENCES `kategori_pengiriman` (`id_kategori_pengiriman`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `ongkir_ibfk_3` FOREIGN KEY (`id_jasa_pengiriman`) REFERENCES `jasa_pengiriman` (`id_jasa_pengiriman`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `ongkir_ibfk_4` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
11.	**Tabel Order**	CREATE TABLE IF NOT EXISTS `orders` (`id_orders` int(11) NOT NULL AUTO_INCREMENT, `status_order` varchar(50) NOT NULL, `id_member` int(11) NOT NULL, `tgl_order` date NOT NULL, `nama_t` varchar(45) NOT NULL, `id_kota_t` int(11) NOT NULL, `alamat_t` text NOT NULL, `kode_pos_t` int(10) NOT NULL, `telepon` varchar(45) NOT NULL, `id_ongkir` int(11) DEFAULT NULL, `tgl_kirim` date NOT NULL, `tgl_diterima` date NOT NULL, `tgl_batas_bayar` date NOT NULL, `tgl_batas_retur` date NOT NULL, `total_bayar` double NOT NULL, `id_bank` int(11) NOT NULL, `dibayar` double NOT NULL, `atas_nama` varchar(50) NOT NULL, `no_rekening` varchar(50) NOT NULL, `no_resi` varchar(100) NOT NULL DEFAULT '-',

		`waktu_bayar` datetime NOT NULL, `paypal_email` varchar(100) NOT NULL, `bayar_paypal` double NOT NULL, `kode` varchar(5) DEFAULT NULL, `jp` varchar(100) NOT NULL, `id_admin` int(11) DEFAULT NULL, PRIMARY KEY (`id_orders`), KEY `id_member` (`id_member`), KEY `orders_ibfk_2` (`id_ongkir`), KEY `orders_ibfk_3` (`id_kota_t`), KEY `id_bank` (`id_bank`), KEY `kode` (`kode`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `orders` ADD CONSTRAINT `orders_ibfk_1` FOREIGN KEY (`id_member`) REFERENCES `member` (`id_member`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `orders_ibfk_4` FOREIGN KEY (`id_ongkir`) REFERENCES `ongkir` (`id_ongkir`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `orders_ibfk_5` FOREIGN KEY (`id_kota_t`) REFERENCES `kota` (`id_kota`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `orders_ibfk_6` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
12.	**Tabel Order_detail**	CREATE TABLE IF NOT EXISTS `order_detail` (`id_order_detail` int(11) NOT NULL AUTO_INCREMENT, `id_orders` int(11) NOT NULL, `id_produk` int(11) NOT NULL, `jumlah` int(11) NOT NULL, UNIQUE KEY (`id_order_detail`), KEY `id_produk` (`id_produk`), KEY `id_orders` (`id_orders`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `retur_detail` ADD CONSTRAINT `retur_detail_ibfk_1` FOREIGN KEY (`id_retur`) REFERENCES `retur` (`id_retur`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `retur_detail_ibfk_2` FOREIGN KEY (`id_order_detail`) REFERENCES `order_detail`

		(`id_order_detail`) ON DELETE NO ACTION ON UPDATE CASCADE;
13	**Tabel Produk**	CREATE TABLE IF NOT EXISTS `produk` (`id_produk` int(11) NOT NULL AUTO_INCREMENT, `nama_produk` varchar(30) NOT NULL, `id_merk` int(11) NOT NULL, `deskripsi` text NOT NULL, `harga` double NOT NULL, `stok` int(11) NOT NULL, `berat` float NOT NULL, `hits` int(11) NOT NULL, `tgl_masuk` date NOT NULL, `last_update` datetime NOT NULL, `dibeli` int(11) NOT NULL, `id_admin` int(11) NOT NULL, PRIMARY KEY (`id_produk`), KEY `id_kategori` (`id_merk`), KEY `nama_produk` (`nama_produk`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `produk` ADD CONSTRAINT `produk_ibfk_4` FOREIGN KEY (`id_merk`) REFERENCES `merk` (`id_merk`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `produk_ibfk_5` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);
14	**Tabel Retur**	CREATE TABLE IF NOT EXISTS `retur` (`id_retur` int(11) NOT NULL AUTO_INCREMENT, `id_orders` int(11) NOT NULL, `status_retur` varchar(50) NOT NULL, `tgl_retur` date NOT NULL, `no_resi` varchar(10) NOT NULL, `id_admin` int(11) DEFAULT NULL, UNIQUE KEY (`id_retur`), KEY `id_orders` (`id_orders`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `retur` ADD CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`id_orders`) REFERENCES `orders` (`id_orders`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `retur_ibfk_2` FOREIGN KEY

		(`id_admin`) REFERENCES `admin` (`id_admin`);
15	**Tabel Retur_detail**	CREATE TABLE IF NOT EXISTS `retur_detail` (`id_retur_detail` int(11) NOT NULL AUTO_INCREMENT, `id_retur` int(11) NOT NULL, `id_order_detail` int(11) NOT NULL, `id_produk` int(11) NOT NULL, `jumlah` int(11) NOT NULL, `keluhan` text NOT NULL, UNIQUE KEY (`id_retur_detail`), KEY `id_retur` (`id_retur`,`id_order_detail`), KEY `id_order_detail` (`id_order_detail`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `retur_detail` ADD CONSTRAINT `retur_detail_ibfk_1` FOREIGN KEY (`id_retur`) REFERENCES `retur` (`id_retur`) ON DELETE NO ACTION ON UPDATE CASCADE, ADD CONSTRAINT `retur_detail_ibfk_2` FOREIGN KEY (`id_order_detail`) REFERENCES `order_detail` (`id_order_detail`) ON DELETE NO ACTION ON UPDATE CASCADE;
16	**Tabel matauang**	CREATE TABLE IF NOT EXISTS `matauang` (`kode` varchar(3) NOT NULL, `kurs` double NOT NULL, `id_admin` int(11) NOT NULL, PRIMARY KEY (`kode`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
17	**Tabel Kategori**	CREATE TABLE IF NOT EXISTS `kategori` (`id_kategori` int(11) NOT NULL AUTO_INCREMENT, `nama_kategori` varchar(50) NOT NULL, `id_admin` int(11) NOT NULL, PRIMARY KEY (`id_kategori`), KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1; ALTER TABLE `kategori` ADD CONSTRAINT `kategori_ibfk_1` FOREIGN KEY (`id_admin`) REFERENCES `admin` (`id_admin`);

IV.3 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna. Adapun untuk tampilan implementasi pembangunan aplikasi *E-commerce* di toko Yella Perdana berbasis *Web* akan dilampirkan pada halaman lampiran.

Tabel IV.4 Implementasi antarmuka Operator

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman *login* Admin	*login*.php
Home Operator	Digunakan sebagai halaman beranda Admin dan terdapat beberapa menu diantaranya menu pengolahan data master, data transaksi, dan laporan.	index.php
Manajemen Merk	Halaman yang berisi informasi merk	merk_list.php
Tambah Merk	Halaman yang brguna untuk menambah data merk	merk _add.php
Edit Merk	Halaman yang digunakan untuk mengubah data merk.	merk_edit.php
Manajemen Produk	Halaman yang berisi informasi produk	produk_list.php
Tambah produk	Halaman yang brguna untuk menambah data produk	produk_add.php
Edit produk	Halaman yang digunakan untuk mengubah data produk	produk_edit.php
Manajemen bank	Halaman yang berisi informasi bank	bank _list.php
Tambah bank	Halaman yang brguna untuk menambah data bank	bank _add.php
Edit bank	Halaman yang digunakan untuk	bank _edit.php

	mengubah data bank	
Manajemen jasa pengiriman	Halaman yang berisi informasi jasa pengiriman	Jasa_pengiriman _list.php
Tambah jasa pengiriman	Halaman yang brguna untuk menambah data jasa pengiriman	Jasa_pengiriman _add.php
Edit jasa pengiriman	Halaman yang digunakan untuk mengubah data jasa pengiriman	Jasa_pengiriman _edit.php
Manajemen kategori pengiriman	Halaman yang berisi informasi kategori pengiriman	kategori _pengiriman _list.php
Tambah kategori pengiriman	Halaman yang brguna untuk menambah data kategori pengiriman	kategori _pengiriman _add.php
Edit kategori pengiriman	Halaman yang digunakan untuk mengubah data kategori pengiriman	kategori _pengiriman _edit.php
Manajemen member	Halaman yang berisi informasi member	member _list.php
Manajemen Kota	Halaman yang berisi informasi kota	kota_list.php
Tambah kota	Halaman yang brguna untuk menambah data kota	kota_add.php
Edit kota	Halaman yang digunakan untuk mengubah data kota	kota_edit.php
Manajemen Provinsi	Halaman yang berisi informasi provinsi	provinsi_list.php
Tambah provinsi	Halaman yang brguna untuk menambah data provinsi	provinsi_add.php
Edit provinsi	Halaman yang digunakan untuk mengubah data provinsi	provinsi_edit.php
Manajemen Ongkir	Halaman yang berisi informasi Ongkir	Ongkir _list.php
Tambah Ongkir	Halaman yang brguna untuk menambah data Ongkir	Ongkir _add.php
Edit Ongkir	Halaman yang digunakan untuk	Ongkir _edit.php

	mengubah data Ongkir	
Manajemen Hubungi	Halaman yang berisi informasi komentar	hubungi _list.php
backup	Halaman yang brguna untuk mem-backup database	Backup.php
restore	Halaman yang brguna untuk me-restore database	Recovery_data.php
Manajemen Pesanan	Halaman yang berisi informasi pesanan	pesanan _list.php
Edit status	Halaman digunakan untuk mengubah status pesanan	pesanan_edit.php
Manajemen Retur	Halaman yang berisi informasi retur	retur _list.php
Edit status	Halaman digunakan untuk mengubah status retur	retur_edit.php
Logout	Digunakan operator untuk keluar dari aplikasi.	Logout.php

Tabel IV.5 Implementasi Antarmuka Admin

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman *login* Admin	*login*.php
Home Admin	Digunakan sebagai halaman beranda Admin dan terdapat beberapa menu diantaranya menu pengolahan data master dan laporan.	index.php
User	Halaman yang berisi informasi *user*	user _list.php
Tambah *user*	Halaman yang brguna untuk menambah data *user*	user _add.php
Edit *user*	Halaman yang digunakan untuk mengubah data *user*	user _edit.php

Laporan penjualan	Halaman yang berisi informasi mengenai laporan pemesanan	laporan_penjualan .php
Laporan stok	Halaman yang berisi informasi mengenai laporan stok yang tersedia	laporan_stok.php
Logout	Digunakan Admin untuk keluar dari aplikasi.	Logout.php

Tabel IV.6 Implementasi Antarmuka Pengunjung

Menu	Deskripsi	Nama File
Registrasi	Digunakan sebagai halaman pendaftaran sebagai member	Registrasi_member.php
Home	Digunakan sebagai halaman beranda	home.php
Semua Produk	Halaman yang berisi informasi mengenai data produk	semua_produk.php
Cara pemesanan dan pembayaran	Halaman yang berisi informasi mengenai tutorial pemesanan,pembayaran dan retur	Cara_pemesanan.php
Profil	Halaman yang berisi informasi tentang perusahaan	about.php
Hubungi Kami	Halaman ini berguna untuk mengirim pesan ke pada Admin disaat Admin tidak *online*	Contact_us.php

Tabel IV.7 Implementasi Antarmuka Member

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman *login* untuk masuk kedalam aplikasi	Admin_*login*.php
Home	Digunakan sebagai halaman beranda	index.php

Semua Produk	Halaman yang berisi informasi mengenai data produk	semua_produk.php
Cara pemesanan dan pembayaran	Halaman yang berisi informasi mengenai tutorial pemesanan,pembayaran dan retur	Cara_pemesanan.php
Profil	Halaman yang berisi informasi tentang perusahaan	about.php
Hubungi Kami	Halaman ini berguna untuk mengirim pesan ke pada Admin disaat Admin tidak *online*	Contact_us.php
ganti *password*	Halaman ini berguna untuk member mengganti *password* nya	gantipass.php
Edit account	Halaman ini berguna untuk member mengganti data pribadinya	Edit_akun.php
History Pemesanan	Halaman ini berguna untuk melihat history pemesanan dan status pemesanan	History_pemesanan.php
Keranjang belanja	Digunakan untuk melihat, mengubah, menghapus data pembelian yang akan dilakukan	-

IV.4 Pengujian Perangkat Lunak

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

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

IV.5 Pengujian Alpa

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

IV.5.1 Skenario Pengujian Alpa

Tabel IV.8 Skenario Pengujian Aplikasi Operator

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data *Login*	*Black Box*
	Validasi *username and password*	*Black Box*
Pengolahan Data Master	Pengolahan Data Produk	*Black Box*
	Pengolahan Data merk	*Black Box*
	Pengolahan Data bank	*Black Box*
	Pengolahan Data jasa pengiriman	*Black Box*
	Pengolahan Data kategori pengiriman	*Black Box*
	Pengolahan Data *member*	*Black Box*
	Pengolahan Data Kota	*Black Box*
	Pengolahan Data Provinsi	*Black Box*
	Pengolahan Data ongkir	*Black Box*
	Pengolahan Data hubungi	*Black Box*
	backup	*Black Box*
	restore	*Black Box*
Pengolahan Data produk	Tampil Data Produk	*Black Box*
	Tambah Data Produk	*Black Box*
	Ubah Data Produk	*Black Box*
Pengolahan Data merk	Tampil Data merk	*Black Box*
	Tambah Data merk	*Black Box*
	Ubah Data merk	*Black Box*
Pengolahan Data bank	Tampil Data bank	*Black Box*
	Tambah Data bank	*Black Box*
	Ubah Data bank	*Black Box*
Pengolahan Data jasa pengiriman	Tampil Data jasa pengiriman	*Black Box*
	Tambah Data jasa pengiriman	*Black Box*
	Ubah Data jasa pengiriman	*Black Box*
Pengolahan Data kategori pengiriman	Tampil Data kategori pengiriman	*Black Box*
	Tambah Data kategori pengiriman	*Black Box*
	Ubah Data kategori pengiriman	*Black Box*

Pengolahan Data *Member*	Tampil Data*Member*	*Black Box*
	Ubah Status	*Black Box*
Pengolahan Data Kota	Tampil Data Kota	*Black Box*
	Tambah Data Kota	*Black Box*
	Ubah Data Kota	*Black Box*
Pengolahan Data Provinsi	Tampil Data Provinsi	*Black Box*
	Tambah Data Provinsi	*Black Box*
	Ubah Data Provinsi	*Black Box*
Pengolahan Data ongkir	Tampil Data ongkir	*Black Box*
	Tambah Data ongkir	*Black Box*
	Ubah Data ongkir	*Black Box*
	Hapus Data ongkir	*Black Box*
Pengolahan Data hubungi	Tampil Data hubungi	*Black Box*
Pengolahan Data Transaksi	Pengolahan Data Pesanan	*Black box*
Logout	Menghapus *session* Admin	*Black Box*

Tabel IV.9 Skenario Pengujian Aplikasi Pengunjug.

Kelas Uji	Butir Uji	Jenis Pengujian
Home	Menampilkan halaman utama.	*Black Box*
Daftar member	Menampilkan form member baru	*Black Box*
	Isi form pendaftaran	*Black Box*
	Simpan data baru	*Black Box*
Lupa *password*	Verifikasi *Email*	*Black Box*
Produk	Menampilkan barang	*Black Box*
Cara pembelian	Menampilkan cara pembelian	*Black Box*
Tentang kami	Menampilkan tentang kami	*Black Box*
Hubungi	Menampilkan form	*Black Box*
	Isi form hubungi	*Black Box*
Keranjang belanja	Menampilkan data pembelian	*Black Box*
	Mengupdate data pembelian	*Black Box*
	Menghapus data pembelian	*Black Box*

Tabel IV.10 Skenario Pengujian Aplikasi Member.

Kelas Uji	Butir Uji	Jenis Pengujian
Home	Menampilkan halaman utama.	*Black Box*
Daftar member	Menampilkan form *user* baru	*Black Box*
	Isi form pendaftaran	*Black Box*
Login	Isi data *login*	*Black Box*
	Verifikasi *Username*	*Black Box*
	Verifikasi *Password*	*Black Box*
Lupa *password*	Verifikasi *Email*	*Black Box*
Produk	Menampilkan barang	*Black Box*
Cara pembelian	Menampilkan cara pembelian	*Black Box*
Tentang kami	Menampilkan tentang kami	*Black Box*
Hubungi	Menampilkan form	*Black Box*
	Isi form hubungi	*Black Box*
Ubah account	Menampilkan form ubah account	*Black Box*
	Ubah data account	*Black Box*
	Simpan perubahan	*Black Box*
Ubah *password*	Menampilkan form ubah *password*	*Black Box*
Keranjang belanja	Menampilkan data pembelian	*Black Box*
	Mengupdate data pembelian	*Black Box*
	Menghapus data pembelian	*Black Box*
Form Konfirmasi	Menampilkan konfirmasi transaksi	*Black Box*
Form Pengiriman	Menampilkan form pengiriman	*Black Box*
	Pilih jenis pengiriman	*Black Box*
Lihat history	Menampilkan history pemesanan	*Black Box*
	Melakukan konfirmasi pembayaran	*Black Box*
Retur	Menampilkan retur pembelian	*Black Box*
	Memilih barang yang di retur	*Black Box*
Logout	Menghapus session *user*	*Black Box*

Tabel IV.11 Skenario pengujian aplikasi Admin.

Kelas Uji	Butir Uji	Jenis Pengujian

Login Admin	Isi data *login*	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Manajemen *User*	Tampil Data *User*	*Black Box*
	Tambah Data *User*	*Black Box*
	Ubah Data *User*	*Black Box*
	Hapus Data *User*	*Black Box*
Laporan stok	Tampilkan data stok barang	*Black Box*
Laporan penjualan	Tampilkan data laporan penjualan	*Black Box*
Logout	Menghapus session *user*	*Black Box*

IV.5.2 Kasus dan Hasil Pengujian

IV.5.2.1 Pengujian Daftar Member

Berikut dapat dilihat pengujian dari daftar *member* yang terdapat pada tabel IV.12 dibawah ini:

Tabel IV.12 Pengujian Daftar *member*.

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Menu Registrasi Member	Tampil form registrasi	Dapat menampilkan form registrasi	[✓] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh masukan : *Username* : saili *Password*: 123 Nama : Ali Sambas Alamat :Binakarya 2 *Email* : budakkaseptea@gmail.com Alamat : Jl. Sekeloa Utara No. 76 Telepon : 08562212544 Provinsi : Jawa Barat	Jika semua *field* terisi atau benar dalam pengisiann, maka akan dapat melajutkan registrasi.	Form registrasi terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses registrasi.	[✓] Diterima [ ] Ditolak

Kota : Bandung Kodepos : 43213			
Klik tombol Daftar	Jika data masukan benar muncul pesan "Aktivasi account telah dikirim ke *email* Anda, Silahkan cek *email* Anda."	Data masukan benar muncu pesan "Aktivasi account telah dikirim ke *email* Anda, Silahkan cek *email* Anda"	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
memasukan data kosong	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan bahwa *field* tersebut harus terisi.	Tampilkan pesan kesalahan dalam pengisian, maka muncul pesan kesalahan bahwa *field* tersebut harus terisi	[✓] Diterima [ ] Ditolak
Email tidak diisi	Tampilkan pesan kesalahan "*email* harus diisi"	Dapat menampilkan pesan kesalahan "*email* harus diisi"	[✓] Diterima [ ] Ditolak
Nama tidak diisi	Tampilkan pesan kesalahan "nama harus diisi"	Dapat menampilkan pesan kesalahan "nama harus diisi"	[✓] Diterima [ ] Ditolak
Alamat tidak diisi	Tampilkan pesan kesalahan "alamat harus diisi"	Dapat menampilkan pesan kesalahan "alamat harus diisi"	[✓] Diterima [ ] Ditolak
Provinsi tidak diisi	Tampilkan pesan kesalahan "Provinsi harus dipilih"	Dapat menampilkan pesan kesalahan "Provinsi harus dipilih"	[✓] Diterima [ ] Ditolak

Kota tidak diisi	Tampilkan pesan kesalahan "Kota harus diisi"	Dapat menampilkan pesan kesalahan "kota harus diisi"	[✓] Diterima [ ] Ditolak
Kodepos tidak diisi	Tampilkan pesan kesalahan "Kodepos harus diisi"	Dapat menampilkan pesan kesalahan "kode harus diisi"	[✓] Diterima [ ] Ditolak
Kodepos diisi dengan huruf	Tampilkan pesan kesalahan "Format kode pos salah"	Dapat menampilkan pesan kesalahan "Format kode pos salah"	[✓] Diterima [ ] Ditolak
Password	Tampilkan pesan kesalahan "*password* harus diisi"	Dapat menampilkan pesan kesalahan "*password* harus diisi"	[✓] Diterima [ ] Ditolak

IV.5.2.2 Pengujian *Login* Member

Berikut dapat dilihat pengujian login member pada tabel no IV.13 dibawah ini :

Tabel IV.13 Pengujian Login *member.*

Kasus dan Hasil Uji (DataBenar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan *username* : saili	Mengisikan data *username* pada *field username*	Dapat mengisi *field username* dan mengisi data *login* dengan benar	[✓] Diterima [ ] Ditolak
Memasukan *password* : 123	Mengisikan data *password* pada *field password*	Mengisikan data *password* pada *field password*	[✓] Diterima [ ] Ditolak
Klik tombol *login*	Data *username* dan data *password* benar	Data *username* dan data *password* benar	[✓] Diterima [ ] Ditolak

	maka akan masuk ke menu utama	maka akan masuk ke menu utama	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan *username* atau *password* yang tidak terdaftar	Tampilkan pesan kesalahan "Maaf account tidak ditemukan"	Muncul pesan "Maaf account tidak ditemukan"	[✓] Diterima [ ] Ditolak
Mengosongkan *username* dan *password*	Tampilkan pesan kesalahan "Maaf *username* dan *password* harus diisi"	Muncul pesan kesalahan "Maaf *username* dan *password* harus diisi"	[✓] Diterima [ ] Ditolak

IV.5.2.3 Pengujian Ubah *Account*

Berikut dapat dilihat pengujian dari logout pada tabel IV.14 dibawah ini :

Tabel IV.14 Pengujian Ubah *Account.*

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu edit account	Menampilkan form edit akun	menampilkan form edit akun	[✓] Diterima [ ] Ditolak
data profil yang akan diubah	dapat mengisi *field* data yang akan diubah	dapat mengisi *field* data yang akan diubah	[✓] Diterima [ ] Ditolak
klik tombol simpan	data berhasil di update di database	data berhasil di update di database	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
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
email tidak diisi	muncul pesan kesalahan "*email* harus diisi	dapat menampilkan pesan kesalahan *email* harus diisi	[✓] Diterima [ ] Ditolak

nama tidak diisi	muncul pesan kesalahan "nama harus diisi"	dapat menampilkan pesan kesalahan nama harus diisi	[✓] Diterima [ ] Ditolak
alamat tidak diisi	muncul pesan kesalahan "alamat harus diisi"	dapat menampilkan pesan kesalahan alamat harus diisi	[ ] Diterima [✓] Ditolak
provinsi tidak diisi	muncul pesan kesalahan "provinsi harus diisi"	dapat menampilkan pesan kesalahan provinsi harus diisi	[✓] Diterima [ ] Ditolak
kota tidak diisi	muncul pesan kesalahan "kota harus diisi"	dapat menampilkan pesan kesalahan kota harus diisi	[✓] Diterima [ ] Ditolak
kode pos tidak diisi	muncul pesan kesalahan "kode pos harus diisi"	dapat menampilkan pesan kesalahan kode pos harus diisi	[✓] Diterima [ ] Ditolak
Kode pos menggunakan huruf	muncul pesan kesalahan "format kode pos salah"	dapat menampilkan pesan kesalahan format kode pos salah	[✓] Diterima [ ] Ditolak

IV.5.2.4 Pengujian Ganti *Password*

Berikut dapat dilihat pengujian dari ubah profil member pada tabel IV.15. dibawah ini:

Tabel IV.15 Pengujian ganti *password*

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
klik menu ubah *password*	menampilkan form ubah *password*	dapat menampilkan form ubah *password*	[✓] Diterima [ ] Ditolak
data yang	*Password* baru	Dapat menyimpan	[✓] Diterima

dimasukkan semuanya benar	berhasil disimpan di *database*	data *password* yang baru di *database*	[ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *password* lama salah	muncul pesan "*password* salah"	Dapat menampilkan pesan kesalahan ”*Password* salah”	[✓] Diterima [ ] Ditolak
Password lama tidak diisi	muncul pesan kesalahan "*password* lama harus diisi”	Dapat menampilkan pesan kesalahan "*password* lama harus diisi”	[✓] Diterima [ ] Ditolak
Confirm *password* tidak diisi	muncul pesan kesalahan "confirm *password* harus diisi”	Dapat menampilkan pesan kesalahan "confirm *password* harus diisi”	[✓] Diterima [ ] Ditolak
Data kosong	Muncul pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan *field* harus diisi	[✓] Diterima [ ] Ditolak

IV.5.2.5 Pengujian Keranjang Belanja Member

Berikut dapat dilihat pengujian dari keranjang belanja pada tabel IV.16 dibawah ini:

Tabel IV.16 Pengujian Keranjang belanja

Kasus dan Hasil Uji (Data Benar)			
data masukan	data yang diharapkan	pengamatan	kesimpulan
Klik menu keranjang belanja	Menampilkan *form* keranjang belanja	Dapat menampilkan *form* keranjang belanja	[✓] Diterima [ ] Ditolak
Jumlah	Tercantum pada text box data yang	Dapat mengisi data jumlah dengan	[✓] Diterima [ ] Ditolak

	akan diubah	benar	
Klik tombol Update	Dapat menampilkan perubahan data jumlah (qty)	Dapat menampilkan perubahan data jumlah (qty)	[✓] Diterima [ ] Ditolak
Klik tombol hapus	Data yang dihapus terhapus dari keranjang belanja	Dapat menghapus data pada keranjang belanja	[✓] Diterima [ ] Ditolak
Klik *checkout*	Menampilkan form pengiriman	Dapat menampilkan form pengiriman	[✓] Diterima [ ] Ditolak

IV.5.2.6 Pengujian Menu Lihat History Pemesanan

Berikut dapat dilihat pengujian dari menu Lihat History pemesanan pada tabel IV.17 dibawah ini:

Tabel IV.17 Pengujian Histori Pemesanan

Kasus dan Hasil Uji (Data Benar)			
data masukan	data yang diharapkan	pengamatan	kesimpulan
Klik menu *history*	Menampilkan *history*	Dapat menampilkan form *history*	[✓] Diterima [ ] Ditolak
Klik lihat	Menampilkan detail pesanan	Dapat menampilkan detail pesanan	[✓] Diterima [ ] Ditolak

IV.5.2.7 Pengujian Menu Produk

Berikut dapat dilihat pengujian dari menu produk pada tabel IV.18 dibawah ini:

Tabel IV.18 Pengujian Menu Produk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu produk	Menampilkan data produk yang ada di *database*	menampilkan data produk yang ada di *database*	[✓] Diterima [ ] Ditolak
Klik view atau selengkapnya	Menampilkan detail produk	Dapat menampilkan detail produk	[✓] Diterima [ ] Ditolak
Klik beli	Menuju ke keranjang belanja	Dapat menampilkan keranjang belanja	[✓] Diterima [ ] Ditolak

IV.5.2.8 Pengujian Profil

Berikut dapat dilihat pengujian dari menu Profil pada tabel IV.19 dibawah ini:

Tabel IV.19 Pengujian Menu Profil

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu Profil	menampilkan profil perusahaan	dapat menampilkan profil perusahaan	[✓] Diterima [ ] Ditolak

IV.5.2.9 Pengujian Hubungi Kami

Berikut dapat dilihat pengujian dari Hubungi Kami pada tabel IV.20 dibawah ini:

Tabel IV.20 Pengujian Hubungi Kami

Kasus dan Hasil Uji (Data Benar)

Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu Hubungi Kami	Menampilkan form hubungi kami	menampilkan form Hubungi kami	[✓] Diterima [ ] Ditolak
data yang dimasukkan semuanya benar	Menampilkan pesan "Terima Kasih Telah Menghubungi Kami"	Dapat menampilkan pesan "Terima Kasih Telah Menghubungi Kami"	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data masukan ada yang tidak diisi	Muncul pesan kesalahan bahwa "*field* tersebut harus terisi"	Menampilkan pesan kesalahan "*field* tersebut harus terisi"	[✓] Diterima [ ] Ditolak
Nama tidak diisi	Menampilkan pesan kesalahan "nama harus diisi"	Menampilkan pesan kesalahan "nama harus diisi"	[✓] Diterima [ ] Ditolak
email tidak diisi	Menampilkan pesan kesalahan "*email* harus diisi"	Menampilkan pesan kesalahan "*email* harus diisi"	[✓] Diterima [ ] Ditolak
Isi pesan tidak diisi	Menampilkan pesan kesalahan "pesan harus diisi"	Menampilkan pesan kesalahan "pesan harus diisi"	[✓] Diterima [ ] Ditolak

IV.5.2.10 Pengujian Merk

Berikut dapat dilihat pengujian dari menu merk pada tabel IV.21 dibawah ini:

Tabel IV.21 Pengujian Menu Merk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu merk	Menampilkan data merk yang ada di *database*	Dapat menampilkan data merk yang ada di *database*	[✓] Diterima [ ] Ditolak
Klik detail pada	Menampilkan	Dapat	[✓] Diterima

data produk	detail produk	menampilkan detail produk	[ ] Ditolak
Klik beli	Menuju ke keranjang belanja	Dapat menampilkan keranjang belanja	[✓] Diterima [ ] Ditolak

IV.5.2.11 Pengujian Checkout

Berikut dapat dilihat pengujian dari checkout pada tabel IV.21 dibawah ini:

Tabel IV.22 Pengujian Form Checkout

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tombol checkout	Menampilkan halaman pengisian data pengiriman	dapat menampilkan halaman pengisian data pengiriman	[✓] Diterima [ ] Ditolak

IV.5.2.12 Pengujian Pengiriman

Berikut dapat dilihat pengujian dari menu hubungi pada tabel IV.23 dibawah ini:

Tabel IV.23 Pengujian Form Pengiriman

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pilih jasa pengiriman	Menampilkan jasa pengiriman	Dapat menampilkan jasa pengiriman	[✓] Diterima [ ] Ditolak
Pilih kategori pengiriman	Menampilkan kategori pengiriman	Dapat menampilkan kategori pengiriman	[✓] Diterima [ ] Ditolak

Pengisian data pengiriman benar	Lanjut ke transaksi selesai	Lanjut ke transaksi selesai	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama penerima	Menampilkan pesan kesalahan "Nama harus diisi"	Dapat menampilkan pesan kesalahan "Nama harus diisi"	[✓] Diterima [ ] Ditolak
Alamat penerima	Menampilkan pesan kesalahan "Alamat harus diisi dengan jelas"	Dapat menampilkan pesan kesalahan "Alamat harus diisi dengan jelas"	[✓] Diterima [ ] Ditolak
Provinsi tujuan	Menampilkan pesan kesalahan "Silahkan pilih"	Dapat menampilkan pesan kesalahan "Silahkan pilih"	[✓] Diterima [ ] Ditolak
Kota tujuan	Menampilkan pesan kesalahan "Silahkan pilih"	Dapat menampilkan pesan kesalahan "Silahkan pilih"	[✓] Diterima [ ] Ditolak
Kode pos	Menampilkan pesan kesalahan "Kode pos harus diisi"	Dapat menampilkan pesan kesalahan "Kode pos harus diisi"	[✓] Diterima [ ] Ditolak
Jenis pengiriman tidak dipilih	Menampilkan pesan kesalahan "Jenis pengiriman harus diisi"	Dapat menampilkan pesan kesalahan "Jenis pengiriman harus diisi"	[✓] Diterima [ ] Ditolak

IV.5.2.13 Pengujian Konfirmasi Pembayaran

Berikut dapat dilihat pengujian dari Konfirmasi pembayaran pada tabel IV.24 dibawah ini:

Tabel IV.24 Pengujian Konfirmasi Pembayaran

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan

Masukan data konfirmasi pembayaran, contoh Id order : 2 Atas Nama : Ferry Dari Bank : Mandiri Tanggal transfer : 2011/07/23 Jumlah Transfer : 125000	Menampilkan *form* konfirmasi	Dapat memunculkan pesan konfirmasi sudah tersimpan.	[✓] Diterima [ ] Ditolak
Memasukan data kosong	Tampilkan pesan kesalahan ”*field* ini harus diisi”	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi”	[✓] Diterima [ ] Ditolak

IV.5.2.14 Pengujian Logout

Berikut dapat dilihat pengujian dari Logout pada tabel IV.25 dibawah ini:

Tabel IV.25 Pengujian Logout

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu logout	Dapat keluar dari menu member dan kembali ke home	Dapat keluar dari menu member dan Kembali ke home	[✓] Diterima [ ] Ditolak

IV.5.2.15 Pengujian Login Operator

Berikut dapat dilihat pengujian dari Login Operator pada tabel IV.26 dibawah ini:

Tabel IV.26 Pengujian Login Operator

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username : Admin	Tercantum pada text box *username*	Dapat mengisi *field username* dengan benar	[✓] Diterima [ ] Ditolak
Password : rahasia	Tercantum pada text box *password*	Dapat mengisi *field password* dengan benar	[✓] Diterima [ ] Ditolak
Klik tombol *login*	jika sama dengan data *login* maka Admin masuk ke menu utama Admin	Tombol *login* dapat berfungsi	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data *username* atau *password* tidak terdaftar	Tidak dapat *login*	Muncul pesan "*username* atau *password* salah, silahkan coba lagi."	[✓] Diterima [ ] Ditolak
Data *username* atau *password* kosong	Tidak dapat *login*	Muncul pesan kesalahan "*username* atau *password* salah, silahkan coba lagi."	[✓] Diterima [ ] Ditolak

IV.5.2.16 Pengujian Pengolahan Data Merk

Berikut dapat dilihat pengujian dari data merk pada tabel IV.27 dibawah ini :

Tabel IV.27 Pengujian Pengolahan Data Merk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan

Klik menu manajemen Merk	Menampilkan *form* Merk	Dapat menampilkan *form* Merk	[✓] Diterima [ ] Ditolak
Klik tombol edit	Menampilkan *form* edit Merk	Dapat menampilkan *form* edit Merk	[✓] Diterima [ ] Ditolak
Klik tombol hapus	Menampilkan pesan apakah data "Anda benar-benar akan menghapus merk?"	Dapat menampilkan pesan apakah data "Anda benar-benar akan menghapus merk?"	[✓] Diterima [ ] Ditolak
Klik tombol tambah	Menampilkan *form* tambah Merk	Menampilkan *form* tambah Merk	[✓] Diterima [ ] Ditolak
Klik tombol kembali	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak

IV.5.2.17 Pengujian Pengolahan Tambah Data Merk

Berikut dapat dilihat pengujian dari tambah data Merk pada tabel IV.28 dibawah ini :

Tabel IV.28 Pengujian Pengolahan Tambah Data Merk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data merk Contoh: Nama merk: Thoshiba	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah Merk	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah Merk	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data merk akan tersimpan didalam *database* dan muncul pesan	Data merk akan tersimpan didalam *database* dan muncul pesan	[✓] Diterima [ ] Ditolak

	”data berhasil di tambah”	”data berhasil di tambah”	
Klik tombol batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama merk masih kosong	Tampilkan pesan kesalahan “merk harus isi”	Dapat menampilkan pesan kesalahan “merk harus diisi”	[✓] Diterima [ ] Ditolak

IV.5.2.18 Pengujian Pengolahan Ubah Data Merk

Berikut dapat dilihat pengujian dari tambah provinsi pada tabel IV.29 dibawah ini :

Tabel IV.29 Pengujian Ubah Data Merk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data merk Contoh: Nama merk: Thoshiba	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah data merk	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah data merk	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data merk akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	Data merk akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama merk masih kosong	Tampilkan pesan kesalahan “merk harus isi”	Dapat menampilkan pesan kesalahan “merk harus diisi”	[✓] Diterima [ ] Ditolak

IV.5.2.19 Pengujian Pengolahan Cari Data Merk

Berikut dapat dilihat pengujian dari cari data merk pada tabel IV.30 dibawah ini **:**

Tabel IV.30 Pengujian Cari Data Merk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi cari merk Contoh: cari merk: Thoshiba	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses cari Merk	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses cari Merk	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama merk masih kosong	Tampilkan pesan kesalahan “merk tidak ada”	Dapat menampilkan pesan kesalahan “merk tidak ada”	[✓] Diterima [ ] Ditolak

IV.5.2.20 Pengujian Pengolahan Tampil Data Produk

Berikut dapat dilihat pengujian dari pengolahan tampil data produk pada tabel IV.31 dibawah ini :

Tabel IV.31 Pengujian Tampil Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu manajemen produk	Menampilkan *form* produk	Dapat menampilkan *form* produk	[✓] Diterima [ ] Ditolak
Klik tombol edit	Menampilkan *form* edit produk	Dapat menampilkan *form* edit produk	[✓] Diterima [ ] Ditolak

Klik tombol hapus	Menampilkan pesan apakah data "Anda benar-benar akan menghapus produk?"	Dapat menampilkan pesan apakah data "Anda benar-benar akan menghapus produk?"	[✓] Diterima [ ] Ditolak
Klik tombol tambah	Menampilkan *form* tambah produk	Menampilkan *form* tambah produk	[✓] Diterima [ ] Ditolak
Klik tombol kembali	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak

IV.5.2.21 Pengujian Pengolahan Tambah Data Produk

Berikut dapat dilihat pengujian dari tambah kota pada tabel IV.32 dibawah ini :

Tabel IV.32 Pengujian Pengolahan Tambah Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu Tambah	Menampilkan *form* tambah produk	Dapat menampilkan *form* tambah produk	[✓] Diterima [ ] Ditolak
Mengisi data produk Contoh: Merk: WD Nama : WD (Harddisk) Harga : 400000 Berat : 0.08 kg Stok : 1 Deskripsi Gambar:	Jika semua *field* terisi semua maka akan muncul pesan " data berhasil ditambah"	Jika semua *field* terisi semua maka akan muncul pesan " data berhasil ditambah"	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Menampilkan pesan "data	Dapat menampilkan	[✓] Diterima [ ] Ditolak

	berhasil ditambah".	pesan "data berhasil ditambah".	
Klik tombol batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
nama produk kosong	Muncul pesan kesalahan"nama produk harus diisi"	Dapat menampilkan pesan kesalahan "nama produk harus diisi"	[✓] Diterima [ ] Ditolak
Harga tidak diisi	Muncul pesan kesalahan"harga harus diisi"	Dapat menampilkan pesan kesalahan "harga harus diisi"	[✓] Diterima [ ] Ditolak
berat tidak diisi	Muncul pesan kesalahan"berat harus diisi"	Dapat menampilkan pesan kesalahan "berat harus diisi"	[✓] Diterima [ ] Ditolak
Stok tidak diisi	Muncul pesan kesalahan"stok harus diisi"	Dapat menampilkan pesan kesalahan "stok harus diisi"	[✓] Diterima [ ] Ditolak

IV.5.2.22 Pengujian Pengolahan Ubah Data Produk

Berikut dapat dilihat pengujian dari ubah data produk pada tabel IV.33 dibawah ini :

Tabel IV.33 Pengujian Ubah Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu Tambah	Menampilkan *form* tambah produk	Dapat menampilkan *form* tambah produk	[✓] Diterima [ ] Ditolak
Mengisi data produk	Jika semua *field* terisi semua maka	Jika semua *field* terisi semua maka	[✓] Diterima [ ] Ditolak

Contoh: Merk: WD Nama : WD (Harddisk) Harga : 400000 Berat : 0.08 kg Stok : 1 Deskripsi Gambar:	akan muncul pesan " data berhasil ditambah"	akan muncul pesan " data berhasil ditambah"	
Klik tombol simpan	Menampilkan pesan "data berhasil ditambah".	Dapat menampilkan pesan "data berhasil ditambah".	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
nama produk kosong	Muncul pesan kesalahan"nama produk harus diisi"	Dapat menampilkan pesan kesalahan "nama produk harus diisi"	[✓] Diterima [ ] Ditolak
Harga tidak diisi	Muncul pesan kesalahan"harga harus diisi"	Dapat menampilkan pesan kesalahan "harga harus diisi"	[✓] Diterima [ ] Ditolak
berat tidak diisi	Muncul pesan kesalahan"berat harus diisi"	Dapat menampilkan pesan kesalahan "berat harus diisi"	[✓] Diterima [ ] Ditolak
Stok tidak diisi	Muncul pesan kesalahan"stok harus diisi"	Dapat menampilkan pesan kesalahan "stok harus diisi"	[✓] Diterima [ ] Ditolak

IV.5.2.23 Pengujian Pengolahan Cari Data Produk

Berikut dapat dilihat pengujian dari cari data produk pada tabel IV.34 dibawah ini :

Tabel IV.34 Pengujian Cari Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi cari barang Contoh: cari barang: Harddisk	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses cari barang	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses cari barang	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama merk masih kosong	Menampilkan semua barang	Dapat menampilkan semua barang	[✓] Diterima [ ] Ditolak

IV.5.2.24 Pengujian Pengolahan Tampil Data provinsi

Berikut dapat dilihat pengujian dari tampil data provinsi pada tabel IV.35 dibawah ini :

Tabel IV.35 Pengujian Tampil Data provinsi

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu manajemen provinsi	Menampilkan *form* provinsi	Dapat menampilkan *form* provinsi	[✓] Diterima [ ] Ditolak
Klik tombol edit	Menampilkan *form* edit provinsi	Dapat menampilkan *form* edit provinsi	[✓] Diterima [ ] Ditolak
Klik tombol hapus	Menampilkan pesan apakah data "Anda benar-benar akan menghapus provinsi?"	Dapat menampilkan pesan apakah data "Anda benar-benar akan menghapus provinsi?"	[✓] Diterima [ ] Ditolak

Klik tombol tambah	Menampilkan *form* tambah provinsi	Menampilkan *form* tambah provinsi	[✓] Diterima [ ] Ditolak
Klik tombol kembali	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak

IV.5.2.25 Pengujian Pengolahan Tambah Data Provinsi

Berikut dapat dilihat pengujian dari tambah data provinsi pada tabel IV.36 dibawah ini :

Tabel IV.36 Pengujian Tambah Data provinsi

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tambah provinsi	Menampilkan *form* tambah provinsi	Dapat menampilkan *form* tambah provinsi	[✓] Diterima [ ] Ditolak
Mengisi data provinsi Contoh: Nama Provinsi : Jawa Barat	Jika semua *field* terisi maka akan muncul pesan "data berhasil ditambah"	Jika semua *field* terisi maka akan muncul pesan "data berhasil ditambah".	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama provinsi masih kosong	Tampilkan pesan kesalahan "nama provinsi harus diisi"	Dapat menampilkan pesan kesalahan " nama provinsi harus diisi"	[✓] Diterima [ ] Ditolak

IV.5.2.26 Pengujian Pengolahan Ubah Data Provinsi

Berikut dapat dilihat pengujian dari tambah data provinsi pada tabel IV.37 dibawah ini :

Tabel IV.37 Pengujian Ubah Data provinsi

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik edit provinsi	Menampilkan *form* edit provinsi	Dapat menampilkan *form* edit provinsi	[✓] Diterima [ ] Ditolak
Mengisi data provinsi Contoh: Nama Provinsi : Jawa Barat	Jika semua *field* terisi maka akan muncul pesan "data berhasil disimpan"	Jika semua *field* terisi maka akan muncul pesan "data berhasil disimpan".	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama provinsi masih kosong	Tampilkan pesan kesalahan "nama provinsi harus diisi"	Dapat menampilkan pesan kesalahan " nama provinsi harus diisi"	[✓] Diterima [ ] Ditolak

IV.5.2.27 Pengujian Cari Data Provinsi

Berikut dapat dilihat pengujian dari cari data provinsi pada tabel IV.38 dibawah ini :

Tabel IV.38 Pengujian Cari Data provinsi

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi cari provinsi Contoh: cari barang: jawa barat	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses cari provinsi	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses cari provinsi	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama provinsi masih	Menampilkan semua provinsi	Dapat menampilkan	[✓] Diterima [ ] Ditolak

kosong		semua provinsi	

IV.5.2.28 Pengujian Pengolahan Data Kota

Berikut dapat dilihat pengujian dari pengolahan data barang pada tabel IV.39 dibawah ini :

Tabel IV.39 Pengujian Liat Kota

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu kota	Menampilkan *form* lihat kota	Dapat menampilkan data kota	[✓] Diterima [ ] Ditolak
Klik tombol ubah	Menampilkan *form* ubah kota	Dapat menampilkan *form* ubah kota	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[✓] Diterima [ ] Ditolak
nama kota masih kosong	Tampilkan pesan kesalahan nama kota harus disi	Dapat menampilkan pesan kesalahan nama kota harus disi	[✓] Diterima [ ] Ditolak

IV.5.2.29 Pengujian Pengolahan Tambah Kota

Berikut dapat dilihat pengujian dari pengolahan tambah kota pada tabel IV.40 dibawah ini :

Tabel IV.40 Pengujian Tambah Kota

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tambah kota	Menampilkan *form* tambah kota	Dapat menampilkan *form* tambah kota	[✓] Diterima [ ] Ditolak
Mengisi data kota Contoh: Provinsi : Jawa Barat Nama Kota : Cianjur	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
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama provinsi masih kosong	Tampilkan pesan kesalahan “kota harus diisi”	Dapat menampilkan pesan kesalahan ” kota harus diisi”	[✓] Diterima [ ] Ditolak

IV.5.2.30 Pengujian Pengolahan Data Ongkos Kirim

Berikut dapat dilihat pengujian dari pengolahan data ongkos kirim pada tabel IV.41 dibawah ini :

Tabel IV.41 Pengujian Lihat Ongkos Kirim

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik pengolahan data ongkos kirim	Menampilkan data ongkos kirirm	Dapat menampilkan data kota ongkos kirim	[✓] Diterima [ ] Ditolak
Klik tombol edit	Menampilkan *form* edit ongkos kirim	Dapat menampilkan *form* edit ongkos kirim	[✓] Diterima [ ] Ditolak
Klik tombol tambah	Menampilkan *form* tambah ongkos kirim	Menampilkan *form* tambah ongkos kirim	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak

IV.5.2.31 Pengujian Pengolahan Tambah Ongkos Kirim

Berikut dapat dilihat pengujian dari pengolahan tambah ongkos kirim pada tabel IV.42 dibawah ini :

Tabel IV.42 Pengujian Tambah Ongkos Kirim

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data ongkos kirim Contoh: Kategori pengiriman : regular Provinsi : Jawa	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah ongkos kirim	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah ongkos kirim	[✓] Diterima [ ] Ditolak

Barat Kota : Cianjur tarif: Rp.5000			
Klik tombol simpan	Data ongkos kirim akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	Data ongkos kirim akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Kategori pengiriman tidak dipilih	Tampilkan pesan kesalahan “silahkan pilih kategori pengiriman”	Dapat menampilkan pesan kesalahan ” silahkan pilih kategori pengiriman”	[✓] Diterima [ ] Ditolak
Provinsi tidak dipilih	Tampilkan pesan kesalahan “silahkan pilih provinsi”	Dapat menampilkan pesan kesalahan ” silahkan pilih provinsi”	[✓] Diterima [ ] Ditolak
Kota tidak dipilih	Tampilkan pesan kesalahan “silahkan pilih kota”	Dapat menampilkan pesan kesalahan ” silahkan pilih kota”	[✓] Diterima [ ] Ditolak
Tarif tidak diisi	Tampilkan pesan kesalahan “tarif harus diisi”	Dapat menampilkan pesan kesalahan ” tarif harus diisi”	[✓] Diterima [ ] Ditolak

IV.5.2.32 Pengujian Pengolahan Data Transaksi

Berikut dapat dilihat pengujian dari pengolahan data Transaksi pada tabel IV.43 dibawah ini :

Tabel IV.43 Pengujian Lihat Transaksi

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik pengeolahan data transaksi	Menampilkan *form* transaksi	Dapat menampilkan *form* lihat transaksi	[✓] Diterima [ ] Ditolak
Klik tabel transaksi pemesanan baru	Menampilkan *form* transaksi pemesanan baru	Dapat menampilkan *form* transaksi pemesanan baru	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik tabel transaksi gagal	Menampilkan *form* transaksi gagal	Dapat menampilkan *form* transaksi gagal	[✓] Diterima [ ] Ditolak

IV.5.2.33 Pengujian Pengolahan Data Kategori

Berikut dapat dilihat pengujian dari data kategori pada tabel IV.44 dibawah ini :

Tabel IV.44 Pengujian Pengolahan Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Klik menu manajemen Merk	Menampilkan *form* Merk	Dapat menampilkan *form* Merk	[✓] Diterima [ ] Ditolak
Klik tombol edit	Menampilkan *form* edit Merk	Dapat menampilkan *form* edit Merk	[✓] Diterima [ ] Ditolak
Klik tombol hapus	Menampilkan pesan apakah data "Anda benar-benar	Dapat menampilkan pesan apakah data	[✓] Diterima [ ] Ditolak

	akan menghapus merk?"	"Anda benar-benar akan menghapus merk?"	
Klik tombol tambah	Menampilkan *form* tambah Merk	Menampilkan *form* tambah Merk	[✓] Diterima [ ] Ditolak
Klik tombol kembali	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak

IV.5.2.34 Pengujian Pengolahan Tambah Data Kategori

Berikut dapat dilihat pengujian dari tambah data kategori pada tabel IV.45 dibawah ini :

Tabel IV.45 Pengujian Pengolahan Tambah Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori Contoh: Nama merk: Laptop	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah Kategori	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah Kategori	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data merk akan tersimpan didalam *database* dan muncul pesan "data berhasil di tambah"	Data merk akan tersimpan didalam *database* dan muncul pesan "data berhasil di tambah"	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama kategori masih kosong	Tampilkan pesan kesalahan "kategori harus isi"	Dapat menampilkan pesan kesalahan "kategori harus diisi"	[✓] Diterima [ ] Ditolak

IV.5.2.35 Pengujian Pengolahan Ubah Data Kategori

Berikut dapat dilihat pengujian dari tambah provinsi pada tabel IV.46 dibawah ini :

Tabel IV.46 Pengujian Ubah Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data merk Contoh: Nama kategori: Harddisk	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah data merk	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah data merk	[✓] Diterima [ ] Ditolak
Klik tombol simpan	Data kategori akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	Data kategori akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[✓] Diterima [ ] Ditolak
Klik tombol batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama kategori masih kosong	Tampilkan pesan kesalahan “kategori harus isi”	Dapat menampilkan pesan kesalahan “kategori harus diisi”	[✓] Diterima [ ] Ditolak

IV.5.2.36 Pengujian Pengolahan Cari Data Kategori

Berikut dapat dilihat pengujian dari cari data kategori pada tabel IV.47 dibawah ini **:**

Tabel IV.47 Pengujian Cari Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi cari merk Contoh: cari kategori: Harddisk	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses cari Kategori	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses cari Kategori	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama kategori masih kosong	Tampilkan pesan kesalahan "kategori tidak ada"	Dapat menampilkan pesan kesalahan "kategori tidak ada"	[✓] Diterima [ ] Ditolak

IV.5.2.37 Pengujian Login admin

Berikut dapat dilihat pengujian dari pengolahan data Transaksi pada tabel IV.48 dibawah ini :

Tabel IV.48 Pengujian *Login* admin

Kasus dan Hasil Uji (DataBenar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan *username* : surabiattack	Mengisikan data *username* pada *field username*	Dapat mengisi *field username* dan mengisi data *login* dengan benar	[✓] Diterima [ ] Ditolak
Memasukan *password* : 123	Mengisikan data *password* pada *field password*	Mengisikan data *password* pada *field password*	[✓] Diterima [ ] Ditolak
Klik tombol *login*	Data *username* dan data *password* benar maka akan masuk ke menu	Data *username* dan data *password* benar maka akan masuk ke menu utama	[✓] Diterima [ ] Ditolak

	utama		
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan *username* atau *password* yang tidak terdaftar	Tampilkan pesan kesalahan "*Username* atau *password* salah, silahkan coba lagi "	Muncul pesan "*Username* atau *password* salah, silahkan coba lagi "	[✓] Diterima [ ] Ditolak

IV.5.2.38 Pengujian Manajemen User

Berikut dapat dilihat pengujian dari pengolahan manajemen user pada tabel IV.49 dibawah ini :

Tabel IV.49 Pengujian Manajemen User

Kasus dan Hasil Uji (DataBenar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan Nama Admin : udin	Mengisikan data nama admin pada *field* nama admin	Dapat mengisikan data nama admin pada *field* nama admin	[✓] Diterima [ ] Ditolak
Memasukan *Username* : master	Mengisikan data *Username* pada *field Username*	Dapat mengisikan data *Username* pada *field Username*	[✓] Diterima [ ] Ditolak
Memasukan *password* : 123	Mengisikan data *password* pada *field password*	Mengisikan data *password* pada *field password*	[✓] Diterima [ ] Ditolak
Klik tombol tambah	Data *username* dan data *password* benar maka akan masuk ke menu utama	Data *username* dan data *password* benar maka akan masuk ke menu utama	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan *username* atau	Tampilkan pesan kesalahan	Muncul pesan "*Username* atau	[✓] Diterima [ ] Ditolak

password yang tidak terdaftar	"*Username* atau *password* salah, silahkan coba lagi "	*password* salah, silahkan coba lagi "	

IV.5.3 Kesimpulan Hasli Pengujian Alpa

Berdasarkan hasil pengujian sistem yang telah dilakukan secara keseluruhan memberikan kesimpulan bahwa pada proses pada situs *e-commerce* telah melalui tahap perbaikan dan sudah dimaksimalkan terhadap proses-proses tersebut dan secara fungsional sistem sudah dapat digunakan dan menghasilkan *output* yang diharapkan.

IV.5.4 Pengujian Betha

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan membuat kuesioner mengenai kepuasan *user* dengan kandungan poin syarat *user friendly* untuk selanjutnya dibagikan kepada sebagian *user* dengan mengambil sample sebanyak 5 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

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

1. Lingkungan Penjual di toko Yella Perdana

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

2. Lingkungan Masyarakat Umum

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

IV.5.5 Kuesioner Pengguna

Kuesioner disebarkan menggunakan teknik sampling yaitu *Simple Random Sampling* yang disebarkan kepada 5 pengguna. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner ini terdiri dari 5 pertanyaan (contoh kuesioner dapat diliihat pada lampiran) dengan menggunakan skala likert dengan fskala 1 sampai 5, dengan ketentuan skala untuk tiap pertanyaan sebagai berikut :

IV.5.5.1 Kuesionner Pengujian Betha Untuk Admin

1. Menurut anda bagaiman tampilan dari situs ini?

A. tidak membantu

B. kurang membantu

D. cukup membantu

E. membantu

C. Cukup Bagus

2. Apakah Anda mengerti dengan instruksi yang terdapat dalam website tersebut mudah dimengerti?

A. Sangat Mudah Digunakan
B. Mudah Digunakan
C. Cukup Mudah Digunakan
D. Cukup Sulit Digunakan
E. Sulit Digunakan

3. Apakah website ini dapat membantu dalam pengolahan data barang?

A. Sangat Membantu
B. Membantu
C. Cukup Membantu
D. Kurang Membantu
E. Tidak Membantu

4. Apakah *website* ini membantu dalam pengoalahan data penjualan secara *online*?

A. Sangat Membantu
B. Membantu
C. Cukup Membantu
D. Kurang Membantu
E. Tidak Membantu

5. Apakah *website* ini dapat membantu dalam membuat laporan penjualan ?

A. Sangat Membantu
B. Membantu
D. Kurang Membantu
E. Tidak Membantu

C. Cukup Membantu

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 2 orang.

1. Bagaimana tampilan perangkat lunak untuk pengolahan data?

Tabel IV.50 Hasil pengujian kuesioner soal nomor 1

Pertanyaan	**No.**	**Keterangan**	**Responden**	**Presentase (%)**
1	1	Tidak membantu	0	0
	2	Kurang membantu	0	0
	3	Cukup bagus	1	50%
	4	Cukup membantu	1	50%
	5	Membantu	0	0
Jumlah			2	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 1 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat cukup bagus

dan 1 orang atau 50% mengatakan perangkat lunak tersebut cukup membantu sehingga secara keseluruhan tampilan perangkat lunak tersebut cukup bagus.

2. Apakah anda mengerti dengan intruksi yang terdapat dalam *website* tersebut mudah dimengerti?

Tabel IV.51 Hasil pengujian kuesioner soal nomor 2

Pertanyaan	No.	Keterangan	Responden	Presentase (%)
2	1	Sangat mudah digunakan	0	0
	2	Mudah digunakan	1	50%
	3	Cukup mudah digunakan	1	50%
	4	Cukup sulit digunakan	0	0
	5	Sulit digunakan	0	0
Jumlah			2	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 1 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat mudah digunakan, 1 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat cukup mudah digunakan. Sehinggan keseluruhan struktur perangkat lunak tersebut cukup mudah digunakan.

3. Apakah *website* ini dapat membantu dalam pengolahan data barang?

Tabel IV.52 Hasil pengujian kuesioner soal nomor 3

Pertanyaan	No.	Keterangan	Responden	Presentase (%)
3	1	Sangat membantu	0	0
	2	Membantu	2	100%
	3	Cukup membantu	0	0
	4	Kurang membantu	0	0
	5	Tidak membantu	0	0
Jumlah			2	100

Berdasarkan hasil presentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 100% mengatakan bahwa perangkat lunak yang dibuat membantu. Sehingga secara keseluruhan perangkan lunak tersebut membantu.

4. Apakah *website* ini membantu dalam pengolahan data penjualan secara *online*?

Tabel IV.53 Hasil pengujian kuesioner soal nomor 4

Pertanyaan	No.	Keterangan	Responden	Presentase (%)
4	1	Sangat membantu	0	0
	2	Membantu	0	0
	3	Cukup membantu	2	100%
	4	Kurang membantu	0	0
	5	Tidak membantu	0	0
Jumlah			2	100

Berdasarkan hasil presentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 100% mengatakan bahwa perangkat lunak yang dibuat cukup membantu. Sehingga secara keseluruhan perangkan lunak tersebut cukup membantu.

5. Apakah *website* ini dapat membantu dalam membuat laporan penjualan?

Tabel IV.54 Hasil pengujian kuesioner soal nomor 5

Pertanyaan	No.	Keterangan	Responden	Presentase (%)
5	1	Sangat membantu	0	0
	2	Membantu	2	100%
	3	Cukup membantu	0	0
	4	Kurang membantu	0	0
	5	Tidak membantu	0	0
Jumlah			2	100

Berdasarkan hasil presentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 100% mengatakan bahwa perangkat lunak yang dibuat membantu. Sehingga secara keseluruhan perangkan lunak tersebut membantu.

Berdasarkan data hasil kusioner tersebut, dapat dicari presentase masing-masing jawaban dengan menggunakan rumus : Y = P/Q * 100%.

Keterangan :

P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden.

Y = Nilai persentase.

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 5 orang.

1. Menurut Anda bagaimana tampilan dari situs ini?

Tabel IV.55 Hasil pengujian kuesioner soal nomor 1

Pertanyaan	**No.**	**Keterangan**	**Responden**	**Presentase (%)**
1	1	Tidak membantu	0	30%
	2	Kurang membantu	0	70%
	3	Cukup membantu	0	0
	4	Membantu	4	0
	5	Sangat membantu	1	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 4 atau 80% menyatakan sangat membantu, dan 1 atau 20% menyatakan sangat membantu bahwa tampilan website cukup menarik dan mudah untuk dipahami/ *user friendly*.

2. Apakah menurut Anda proses pembayaran di situs ini cukup mudah?

Tabel IV.56 Hasil pengujian kuesioner soal nomor 2

Pertanyaan	No.	Keterangan	Responden	Presentase (%)
2	1	Tidak bagus	0	30%
	2	Kurang bagus	0	60%
	3	Cukup bagus	1	10%
	4	Bagus	4	0
	5	Sangat bagus	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 4 orang atau 80% menyatakan bagus, 1 orang atau 20% menyatakan cukup bagus bahwa informasi yang diberikan tentang produk terbaru yang ada di toko Yella Perdana.

3. Apakah menurut Anda, proses *Retur* di toko kami cukup mudah?

Tabel IV.57 Hasil pengujian kuesioner soal nomor 3

Pertanyaan	No.	Keterangan	Responden	Presentase (%)
3	1	Sangat membantu	1	20%
	2	Membantu	2	40%
	3	Cukup membantu	2	40%
	4	Kurang membantu	0	0
	5	Tidak membantu	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 1 atau 20% menyatakan sangat membantu, 2 orang atau 40 % menyatakan membantu dan, 2 atau 20 % menyatakan cukup membantu bahwa *website* ini akan lebih memudahkan pelanggan dalam melakukan retur barang.

4. Apakah menurut Anda situs ini cukup *user friendly*?

Tabel IV.58 Hasil pengujian kuesioner soal nomor 4

Pertanyaan	No.	Keterangan	Responden	Presentase (%)
4	1	Sangat membantu	3	60%
	2	Membantu	1	20%
	3	Cukup membantu	1	20%
	4	Kurang membantu	0	0
	5	Tidak membantu	0	0
Jumlah			10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 orang atau 60% menyatakan sangat membantu,1 orang atau 20% menyatakan membantu dan 1 orang atau 20% menyatakan cukup setuju bahwa *website* ini akan memperluas pangsa pasar pemilik toko Yella Perdana.

5. Apakah anda setuju bahwa transaksi jarak jauh pada website ini memudahkan pelanggan baik yang berada dalam kota maupun luar kota?

Tabel IV.60 Hasil pengujian kuesioner soal nomor 5

Pertanyaan	No.	Keterangan	Responden	Presentase (%)
5	1	Sangat membantu	1	20%
	2	Membantu	3	60%
	3	Cukup membantu	1	20%
	4	Kurang membantu	0	0
	5	Tidak membantu	0	0

Jumlah	10	100

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 1 atau 20% menyatakan sangat membantu, 3 orang atau 60 % menyatakan membantu dan, 2 atau 20 % menyatakan cukup membantu bahwa *website* ini akan lebih memudahkan pelanggan dalam melakukan transaksi jarak jauh.

IV.5.6 Kesimpulan Pengujian Betha

Berdasarkan hasil prosentase diatas yang didapatkan dari pengujian betha pengguna admin dan user, maka sistem penerapan *E-commerce* untuk sistem penjualan ini sudah sesuai dengan tujuan yaitu Yella Perdana dapat mepermudah dalam proses promosi dan penjualan produk yang ditawarkan, pelanggan dapat melakukan transaksi pembelian secara *online*, tampilannya *user friendly*, memudahkan dalam pengelolaan data barang dan memudahkan dalam membuat laporan penjualan.