BAB 4
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

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

4.1.1 Kebutuhan Perangkat Keras

Perangkat keras yang diperlukan untuk membangun Aplikasi *e-commerce* di Toko Low and Lame distro adalah sebagai berikut:

Tabel 4. 1 Spesifikasi Perangkat Keras

Perangkat	Spesifikasi
Processor	Kecepatan 2.8 GHz
RAM	1 GB
Harddisk	160 GB
Monitor	17 Inchi
VGA	On board
Lan Card	10/100 Mbps
Koneksi Internet	384 kbps

4.1.2 Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan untuk mengembangkan Aplikasi *e-commerce* di Toko Low and Lame distro adalah sebagai berikut

Tabel 4. 2 Spesifikasi Perangkat Lunak

Perangkat Lunak	Keterangan
Sistem Operasi	Windows XP SP2
PHP	Bahasa Pemograman
XAMPP	*Web Server*
MySQL	*Database Server*
Mozilla firefox	*Web Browser*
Adobe dreamweaver	*Code Editor*

4.1.3 Implementasi Basis Data

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

Tabel 4. 3 Implementasi Basai Data

No	Nama Tabel	Query
1.	Admins	CREATE TABEL IF NOT EXISTS `admins` (`id_admin` int(3) NOT NULL auto_increment, `useradmin` varchar(50) NOT NULL, `*password*` varchar(50) NOT NULL, `nama_lengkap` varchar(100) NOT NULL, `email` varchar(100) NOT NULL, `no_telp` varchar(20) NOT NULL, `level` varchar(20) NOT NULL, `blokir` enum('Y','N') NOT NULL, `id_session` varchar(100) NOT NULL,

No	Nama Tabel	Query
		PRIMARY KEY (`useradmin`), UNIQUE KEY `id_admin` (`id_admin`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
2.	Kategori	CREATE TABEL IF NOT EXISTS `kategori` (`id_kategori` int(5) NOT NULL auto_increment, `nama_kategori` varchar(100) NOT NULL, `kategori_seo` varchar(100) NOT NULL, PRIMARY KEY (`id_kategori`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;
3.	Produk	CREATE TABLE IF NOT EXISTS `produk` (`id_produk` int(5) NOT NULL AUTO_INCREMENT, `id_kategori` int(5) NOT NULL, `nama_produk` varchar(100) NOT NULL, `produk_seo` varchar(100) NOT NULL, `deskripsi` text NOT NULL, `bahan` varchar(35) DEFAULT NULL, `harga` int(20) NOT NULL, `diskon` int(11) NOT NULL, `berat` decimal(5,2) NOT NULL, `tgl_masuk` date NOT NULL, `gambar` varchar(100) NOT NULL, `dibeli` int(5) NOT NULL, `dilihat` int(5) NOT NULL,

No	Nama Tabel	Query
		PRIMARY KEY (`id_produk`), KEY `id_kategori` (`id_kategori`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=9
4.	Gambar	CREATE TABEL IF NOT EXISTS `gambar` (`id_gambar` int(5) NOT NULL auto_increment, `id_produk` int(5) NOT NULL, `nama_gambar` varchar(100) NOT NULL, `photo` varchar(100) NOT NULL, PRIMARY KEY (`id_gambar`), KEY `id_produk` (`id_produk`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=23 ;
5.	produk_opsi	CREATE TABEL IF NOT EXISTS `produk_opsi` (`id_produk_opsi` int(5) NOT NULL auto_increment, `id_produk` int(5) NOT NULL, `stok` int(5) NOT NULL, PRIMARY KEY (`id_produk_opsi`), KEY `id_produk` (`id_produk`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=15 ;
6.	Members	CREATE TABLE IF NOT EXISTS `members` (`id_member` int(5) NOT NULL AUTO_INCREMENT, `username` varchar(50) NOT NULL, `password` varchar(50) NOT NULL, `nama_lengkap` varchar(100) NOT NULL,

No	Nama Tabel	Query
		`alamat` text NOT NULL, `jk` enum('P','W') NOT NULL, `email` varchar(30) NOT NULL, `no_telp` varchar(13) NOT NULL, `kota` int(10) NOT NULL, `provinsi` int(10) NOT NULL, `kode_pos` varchar(5) NOT NULL, `level` enum('member') NOT NULL, `blokir` enum('Y','N') NOT NULL, `id_session` varchar(100) NOT NULL, PRIMARY KEY (`username`), UNIQUE KEY `id_member` (`id_member`), KEY `kota` (`kota`), KEY `provinsi` (`provinsi`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=32 ;
7.	Provinsi	CREATE TABEL IF NOT EXISTS `provinsi` (`id_prov` int(10) NOT NULL auto_increment, `nama_prov` varchar(100) default NULL, KEY `id_prov` (`id_prov`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=36 ;
8.	Kota	CREATE TABEL IF NOT EXISTS `kota` (`id_kota` int(10) NOT NULL auto_increment, `id_prov` int(10) default NULL,

No	Nama Tabel	Query
		`nama_kota` varchar(50) default NULL, `kabkota` varchar(20) default NULL, UNIQUE KEY `id_kota` (`id_kota`), KEY `id_prov` (`id_prov`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=430 ;
9.	Ongkoskirim	CREATE TABEL IF NOT EXISTS `ongkoskirim` (`id_ongkos` int(5) NOT NULL auto_increment, `id_kota` int(10) NOT NULL, `ongkos_kirim` int(10) NOT NULL, `perusahaan_jasa` varchar(40) default NULL, `jasa_kirim` varchar(25) NOT NULL, PRIMARY KEY (`id_ongkos`), KEY `id_kota` (`id_kota`), KEY `jasa_kirim` (`jasa_kirim`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=859 ; ADD CONSTRAINT `ongkoskirim_ibfk_1` FOREIGN KEY (`jasa_kirim`) REFERENCES `jasa_pengiriman` (`jasa_kirim`); ON DELETE CASCADE ON UPDATE CASCADE;
10.	Orders	CREATE TABEL IF NOT EXISTS `orders` (`id_orders` int(5) NOT NULL auto_increment, `jasa_kirim` varchar(20) NOT NULL, `no_transaksi` varchar(20) NOT NULL, `username` varchar(50) NOT NULL,

No	Nama Tabel	Query
		`nama_kustomer` varchar(50) NOT NULL, `alamat` text NOT NULL, `telpon` varchar(13) NOT NULL, `email` varchar(30) NOT NULL, `status_order` varchar(30) NOT NULL, `status_bayar` varchar(30) NOT NULL, `status_pengiriman` varchar(30) NOT NULL, `kode_pos` int(5) NOT NULL, `tgl_order` date NOT NULL, `tgl_bayar` date NOT NULL, `tgl_batal` date NOT NULL, `tgl_sampai` date NOT NULL, `tot_order` int(10) NOT NULL, `dibayar_dlr` double(10,2) NOT NULL, `dibayar_rp` int(10) NOT NULL, `jam_order` time NOT NULL, `id_prov` int(10) NOT NULL, `id_kota` int(10) NOT NULL, `no_resi` varchar(20) NOT NULL, `jenis_bayar` varchar(20) NOT NULL, `ongkos` double NOT NULL, `no_rek` int(20) NOT NULL, `track_tabel` text NOT NULL, `track_tujuan` varchar(256) default NULL, `track_petugas` varchar(256) default NULL,

No	Nama Tabel	Query
		`track_tgl_terima` varchar(75) default NULL, `track_penerima` varchar(256) default NULL, `track_status` varchar(25) default NULL, `track_tgl_kirim` date NOT NULL, PRIMARY KEY (`id_orders`), KEY `username` (`username`), KEY `id_prov` (`id_prov`), KEY `id_kota` (`id_kota`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=42 ;
11.	orders_detail	CREATE TABEL IF NOT EXISTS `orders_detail` (`id_orders` int(11) NOT NULL, `id_produk` int(11) NOT NULL, `jumlah` int(5) NOT NULL, `total` int(10) NOT NULL, `diskon` int(3) NOT NULL, `harga` int(10) NOT NULL, `id_produk_opsi` int(5) NOT NULL, KEY `id_produk` (`id_produk`), KEY `id_orders` (`id_orders`), KEY `id_produk_opsi` (`id_produk_opsi`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
12.	orders_temp	CREATE TABEL IF NOT EXISTS `orders_temp` (`id_orders_temp` int(5) NOT NULL auto_increment, `id_produk` int(5) NOT NULL,

No	Nama Tabel	Query
		`username` varchar(50) NOT NULL, `jumlah` int(5) NOT NULL, `harga` int(10) NOT NULL, `diskon` int(3) NOT NULL, `id_produk_opsi` int(5) NOT NULL, `tgl_order_temp` date NOT NULL, `jam_order_temp` time NOT NULL, `stok_temp` int(5) NOT NULL, `id_session` varchar(50) NOT NULL, PRIMARY KEY (`id_orders_temp`), KEY `id_produk` (`id_produk`), KEY `username` (`username`), KEY `id_produk_opsi` (`id_produk_opsi`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=98 ; ADD CONSTRAINT `orders_temp_ibfk_1` FOREIGN KEY (`id_produk`) REFERENCES `produk` (`id_produk`), ADD CONSTRAINT `orders_temp_ibfk_2` FOREIGN KEY (`id_produk_opsi`) REFERENCES `produk_opsi` (`id_produk_opsi`);
13.	Kurs	CREATE TABLE IF NOT EXISTS `kurs` (`kurs_id` int(11) NOT NULL AUTO_INCREMENT, `kurs_rupiah` int(7) NOT NULL DEFAULT '0', `kurs_admin` int(6) NOT NULL DEFAULT '0', `kurs_update` varchar(25) NOT NULL, `kurs_flag` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE

No	Nama Tabel	Query
		CURRENT_TIMESTAMP, PRIMARY KEY (`kurs_id`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=248 ;
14.	retur	CREATE TABEL IF NOT EXISTS `retur` (`id_retur` int(5) NOT NULL auto_increment, `username` varchar(50) NOT NULL, `id_orders` int(5) NOT NULL, `tgl_retur` date NOT NULL, `status_retur` varchar(20) NOT NULL, `no_resi_retur` varchar(20) NOT NULL, PRIMARY KEY (`id_retur`), KEY `username` (`username`), KEY `id_orders` (`id_orders`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
15.	detail_retur	CREATE TABEL IF NOT EXISTS `detail_retur` (`id_detail_retur` int(5) NOT NULL auto_increment, `id_retur` int(5) NOT NULL, `id_produk` int(5) NOT NULL, `jumlah` int(10) NOT NULL, `alasan` text NOT NULL, `status_produk_retur` varchar(20) NOT NULL, PRIMARY KEY (`id_detail_retur`), KEY `id_produk` (`id_produk`),

No	Nama Tabel	Query
		KEY `opsi` (`opsi`), KEY `id_retur` (`id_retur`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
16.	Vote	CREATE TABEL IF NOT EXISTS `vote` (`id` int(5) NOT NULL, `username` varchar(50) NOT NULL, `date` datetime NOT NULL, `vote` int(11) NOT NULL, PRIMARY KEY (`id`), KEY `username` (`username`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
17.	Hubungi	CREATE TABEL IF NOT EXISTS `hubungi` (`id_hubungi` int(5) NOT NULL auto_increment, `nama` varchar(50) NOT NULL, `email` varchar(30) NOT NULL, `subjek` varchar(100) NOT NULL, `pesan` text NOT NULL, `tanggal` date NOT NULL, PRIMARY KEY (`id_hubungi`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
18.	Komentar	CREATE TABEL IF NOT EXISTS `komentar` (`id_komentar` int(5) NOT NULL auto_increment, `id_produk` int(5) NOT NULL,

No	Nama Tabel	Query
		`nama` varchar(50) NOT NULL, `komentar` text NOT NULL, `tgl_komentar` date NOT NULL, `jam_komentar` time NOT NULL, `aktif` enum('Y','N') NOT NULL, PRIMARY KEY (`id_komentar`), KEY `id_produk` (`id_produk`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
19.	jasa_pengiriman	CREATE TABLE IF NOT EXISTS `jasa_pengiriman` (`jasa_kirim` varchar(25) NOT NULL, `perusahaan` varchar(100) DEFAULT NULL, PRIMARY KEY (`jasa_kirim`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
20.	ukuran	CREATE TABLE IF NOT EXISTS `ukuran` (`id_ukuran` int(11) NOT NULL AUTO_INCREMENT, `nm_ukuran` varchar(30) NOT NULL, PRIMARY KEY (`id_ukuran`)) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=6
21.	warna	CREATE TABLE IF NOT EXISTS `warna` (`id_warna` int(11) NOT NULL AUTO_INCREMENT, `nama_warna` varchar(50) NOT NULL, PRIMARY KEY (`id_warna`), UNIQUE KEY `nama_warna` (`nama_warna`)) ENGINE=MyISAM DEFAULT CHARSET=latin1

No	Nama Tabel	Query
		COMMENT='Tabel Warna' AUTO_INCREMENT=12

4.1.4 Implementasi Antarmuka

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

4.1.4.1 Implementasi Antarmuka Admin

Tabel 4. 4 Antarmuka Admin

Fungsi	Deskripsi	Proses	File
Login	Untuk menangani proses *login* admin	Login	login.php
Ganti *Password*	Untuk mengubah *password* admin	Ganti *Password* Admin	*password*.php
Kategori	Untuk mengolah data kategori seperti tambah kategori,edit kategori,hapus kategori	Pengolahan Data Kategori	kategori.php
Produk	Untuk mengolah data produk seperti tambah produk,edit produk,hapus produk	Pengolahan Data Produk	produk.php
ukuran	Untuk mengolah data ukuran seperti tambah ukuran, edit ukuran, hapus ukuran	Pengolahan Data ukuran	getukuran.php
Gambar	Untuk mengolah data gambar seperti tambah gambar dan hapus gambar	Pengolahan Data Produk	produk.php
Provinsi	Untuk mengolah data provinsi seperti tambah provinsi,edit provinsi,hapus provinsi dan pencarian provinsi	Pengolahan Provinsi	provinsi.php

Fungsi	Deskripsi	Proses	File
Kota	Untuk mengolah data kota seperti tambah kota,edit kota,hapus kota dan pencarian kota	Pengolahan Data Kota	kota.php
Ongkos Kirim	Untuk mengolah data ongkos kirim seperti tambah ongkos kirim,edit ongkos kirim dan hapus ongkos kirim	Pengolahan Data Ongkos Kirim	ongkoskirim.php
Jasa pengiriman	Untuk mengolah data jasa pengiriman seperti tambah data jasa pengirima,edit data jasa pengirimadan hapus data jasa pengirima	Pengolahan Data Ongkos Kirim	ongkoskirim2.php
Orders Dibayar	Untuk mengolah data orders dibayar yaitu untuk update status bayar	Pengolahan Data Orders	orders-dibayar.php
Orders Lunas	Untuk mengolah data orders lunas yaitu untuk update status bayar dan input no resi	Pengolahan Data Orders	orders-lunas.php
Orders Dikirim	Untuk mengolah data orders dikirim yaitu untuk update status pngiriman	Pengolahan Data Orders	orders-dikirim.php
Orders diterima	Untuk mengolah data orders diterima yaitu untuk update status pengiriman	Pengolahan Data Orders	orders-diterima.php
Retur Produk	Untuk mengolah data retur seperti update status retur dan input no resi retur	Pengolahan Data Retur	retur.php
Hubungi Kami	Untuk mengolah data hubungi seperti membalas pesan dan hapus hubungi	Pengolahan Data Hubungi	hubungi.php
Backup Database	Untuk mengolah data backup yaitu backup database	Pengolahan Data Backup	backup.php
Restore Database	Untuk mengolah data restore yaitu restore database	Pengolahan Data Restore	restore.php

Fungsi	Deskripsi	Proses	File
Cara Pembelian	Untuk mengisi tatacara pembelian	-	carabeli.php
Profil Toko	Untuk mengisi profile toko	-	profil.php
Syarat & Kondisi	Untuk mengisi syarat dan kondisi pembelian	-	keterangan.php
Laporan Penjualan	Untuk mencari dan mencetak laporan penjualan	Pembuatan Laporan Penjualan	laporan-penjualan.php

4.1.4.2 Implementasi Antarmuka Members

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk members dapat dilihat pada tabel 4.5 berikut ini:

Tabel 4. 5 Antarmuka Members

Fungsi	Deskripsi	Proses	File
Masuk	Untuk menangani proses *login* members	Login	login.php
Daftar	Untuk mendaftar sebagai members	Registrasi	register-form.php
Beranda	Tampilan awal antarmuka pengujung dan members	-	index.php
Produk	Untuk menampilkan semua produk	Info Produk	content-center.php
Keranjang Belanja	Untuk menampilkan pemesanan produk	Transaksi	aksi.php
Cara Pembelian	Untuk menampilkan cara pembelian produk	-	content-center.php
Kontak	Untuk menampilkan profile toko	-	content-center.php
Hubungi Kami	Untuk mengirimkan	Hubungi	hubungi.php

Fungsi	Deskripsi	Proses	File
	pertanyaan		
Pencarian Produk	Untuk pencarian produk	Pencarian Produk	index.php
Kategori	Untuk Menampilkan produk berdasarkan kategori	-	content-left.php
Customer Service	Untuk *chating* dengan admin toko	-	content-left.php
Produk Terlaris	Untuk menampilkan produk yang paling laris	-	content-left.php
Facebook	Untuk menampilkan *content* facebook	-	content-right.php
Produk favorit	Untuk menampilkan produk yang paling banyak dilihat	-	content-right.php
Profil	Untuk menampilkan dan mengedit data members	Update Profile	content-center.php
Data Pesanan	Untuk menampilkan history pemesanan	History Orders	content-center.php
Retur	Untuk menampilan retur produk yang dipesan	Retur	content-center.php

4.1.4.3 Implementasi Antarmuka Pengunjung

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk pengunjung dapat dilihat pada tabel 4.6 berikut ini:

Tabel 4. 6 Antarmuka Pengunjung

Fungsi	Deskripsi	Proses	File
Daftar	Untuk mendaftar sebagai members	Registrasi	register-form.php
Beranda	Tampilan awal antarmuka pengujung dan members	-	index.php

Fungsi	Deskripsi	Proses	File
Produk	Untuk menampilkan semua produk	Info Produk	content-center.php
Keranjang Belanja	Untuk menampilkan pemesanan produk	Transaksi	aksi.php
Cara Pembelian	Untuk menampilkan cara pembelian produk	-	content-center.php
Kontak	Untuk menampilkan profile toko	-	content-center.php
Hubungi Kami	Untuk mengirimkan pertanyaan	Hubungi	hubungi.php
Pencarian Produk	Untuk pencarian produk	Pencarian Produk	index.php
Kategori	Untuk Menampilkan produk berdasarkan kategori	-	content-left.php
Customer Service	Untuk *chating* dengan admin toko	-	content-left.php
Produk Terlaris	Untuk menampilkan produk yang paling laris	-	content-left.php
Facebook	Untuk menampilkan *content* facebook	-	content-right.php
Produk favorit	Untuk menampilkan produk yang paling banyak dilihat	-	content-right.php

4.2 Pengujian Alpha

Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan merepresentasikan kajian pokok dari spesifikasi, perancangan, dan pengkodean. Pengujian yang digunakan untuk menguji system ini adalah metode pengujian *black-box*. Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak.

4.2.1 Rencana Pengujian

Rencana pengujian akan dilakukan dengan menguji sistem secara alpha dan beta. Pengujian perangkat lunak ini menggunakan data uji berdasarkan data yang terdapat pada Toko Low and Lame distro.

1. Aplikasi *E-commerce* toko Low and Lame distro bagian Admin.

Tabel 4. 7 Rencana Pengujian Bagian Admin

Item Uji	Detail Pengujian	Jenis Pengujian
Login	Isi data *login*	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Lupa *Password*	Isi *form* lupa *password*	*Black Box*
	Kirim *password* baru ke email	*Black Box*
Kategori	Tampil Data Kategori	*Black Box*
	Tambah Data Kategori	*Black Box*
	Edit Data Kategori	*Black Box*
	Hapus Data Kategori	*Black Box*
Produk	Tambah Data Produk	*Black Box*
	Edit Data Produk	*Black Box*
	Hapus Data Produk	*Black Box*
	Edit Data Produk	*Black Box*
	Pengolahan Data Gambar	*Black Box*
	Pengolahan Data Update Stok	*Black Box*
	Hapus Data Produk	*Black Box*

Item Uji	Detail Pengujian	Jenis Pengujian
Gambar	Tampil Data Gambar	*Black Box*
	Tambah Data Gambar	*Black Box*
	Hapus Data Gambar	*Black Box*
Provinsi	Tampil Data Provinsi	*Black Box*
	Cari Data Provinsi	*Black Box*
	Tambah Data Provinsi	*Black Box*
	Edit Data Provinsi	*Black Box*
	Hapus Data Provinsi	*Black Box*
Kota	Tampil Data Kota	*Black Box*
	Cari Data Kota	*Black Box*
	Tambah Data Kota	*Black Box*
	Edit Data Kota	*Black Box*
	Hapus Data Kota	*Black Box*
Ongkos Kirim	Tampil Data Ongkos Kirim	*Black Box*
	Tambah Data Ongkos Kirim	*Black Box*
	Edit Data Ongkos Kirim	*Black Box*
	Hapus Data Ongkos Kirim	*Black Box*
Jasa pengiriman	Tampil Data Jasa Pengiriman	*Black Box*
	Tambah Data Jasa Pengiriman	*Black Box*
	Edit Data Jasa Pengiriman	*Black Box*
	Hapus Data Jasa Pengiriman	*Black Box*
Ukuran	Tampil Data ukuran	*Black Box*
	Tambah Data ukuran	*Black Box*

Item Uji	Detail Pengujian	Jenis Pengujian
Warna	Edit Data ukuran	*Black Box*
	Hapus data ukuran	*Black Box*
	Tampil data warna	*Black Box*
	Tambah data warna	*Black Box*
	Edit data warna	*Black Box*
	Hapus data warna	*Black Box*
Pemesanan dan Laporan Penjualan	Tampil Data Laporan Penjualan	*Black Box*
	Cari Data Laporan Penjualan	*Black Box*
	Cetak Data Laporan Penjualan	*Black Box*
Retur Penjualan	Tampilkan Data Retur	*Black Box*
	Konfirmasi Data Retur	*Black box*
Logout	Menghapus *session* Admin	*Black Box*

2. Aplikasi *E-commerce* toko Low and Lame distro bagian Members.

Tabel 4. 8 Rencana Pengujian Bagian Member

Item Uji	**Detail Pengujian**	**Jenis Pengujian**
Register	Menampilkan *form* Pendaftaran	*Black Box*
	Isi *form* Pendaftaran	*Black Box*
	Simpan Data Pendaftaran	*Black Box*
Login	Isi data *login*	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Lupa *Password*	Isi *form* lupa *password*	*Black Box*

Item Uji	Detail Pengujian	Jenis Pengujian
	Kirim *password* baru ke email	*Black Box*
Profil Member	Menampilkan *Form Edit* Members	*Black Box*
Keranjang Belanja	Menampilkan Data pembelian	*Black Box*
	Mengupdate data pembelian	*Black Box*
	Menghapus data pembelian	*Black Box*
Alamat pengiriman	Menampilkan form pengiriman	*Black Box*
	Pilih alamat pengiriman	*Black Box*
Jenis pengiriman	Menampilkan form jenis pengiriman	*Black Box*
	Pilih jenis pengiriman	*Black Box*
Konfirmasi Pembayaran	Menampilkan Data Konfirmasi pembayaran	*Black Box*
Retur	Menampilkan Data produk Retur	*Black Box*
Logout	Menghapus *session user*	*Black Box*

4.2.2 Kasus dan Hasil Pengujian

Berdasarkan rencana pengujian, maka dapat dilakukan pengujian *Alpha* pada aplikasi e-commerce di Toko Low and Lame distro sebagai berikut.

4.2.2.1 Fasilitas Admin

Pengujian Login Admin

Untuk penjelasan lebih lanjut tentang skenario pengujian login admin dapat dilihat pada tabel sebagai berikut.

Tabel 4. 9 Pengujian Data Login

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Username :rahmat	Tercantum pada text box *username*	Dapat mengisi field *username* dengan benar	[√] Diterima [ ] Ditolak
Password : admin	Tercantum pada text box *password*	Dapat mengisi field *password* dengan benar	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Username : admin *Password*: 12345	Tidak dapat login	Muncul pesan "*username* atau *password* salah."	[√] Diterima [ ] Ditolak
Username : *Password*:	Tidak dapat login	Muncul pesan kesalahan "masukan *username* dan *password* anda."	[√] Diterima [ ] Ditolak

Pengujian ubah password administrator

Berikut dapat dilihat pengujian dari ubah *password* administrator:

Tabel 4. 10 Pengujian ubah password administrator

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

Klik ubah password	Menampilkan *form* ganti *password* adminstrator	Dapat menampilkan *form* ganti *password* administrator	[√] Diterima [ ] Ditolak
Password: 110786 Konfirmasi *Password*: 110786	*Password* baru berhasil disimpan di *database*	Dapat menyimpan data *password* yang baru dengan muncul pesan " Data berhasil diubah"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Password: Konfirmasi *Password*: 110786	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan "*field* ini harus diisi" bila ada field yang belum diisi	[√] Diterima [ ] Ditolak
Password: 110786 Konfirmasi *Password*: 110786123	Muncul pesan "password baru dan konfirmasi password harus sama"	Dapat menampilkan pesan "password baru dan konfirmasi password harus sama"	[√] Diterima [ ] Ditolak
Password: Konfirmasi *Password*:	Muncul pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan *field* harus diisi	[√] Diterima [ ] Ditolak

Pengujian Pengolahan Data Kategori

a. Tambah Data Kategori

Untuk penjelasan lebih lanjut tentang skenario tambah data kategori dapat dilihat pada tabel sebagai berikut.

Tabel 4. 11 Pengujian Tambah Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori Contoh: Nama : sepatu	Jika semua field terisi maka akan muncul pesan “data berhasil disimpan”	Dapat memunculkan pesan data berhasil di simpan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori Contoh: Nama :	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”nama kategori harus diisi”	[√] Diterima [ ] Ditolak

b. Edit Data Kategori

Untuk penjelasan lebih lanjut tentang skenario edit data kategori dapat dilihat pada tabel sebagai berikut.

Tabel 4. 12 Pengujian Edit Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengedit data kategori Contoh: Nama : Sepatu	Jika semua field terisi maka akan muncul pesan "data berhasil disimpan"	Dapat memunculkan pesan data berhasil di simpan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengedit data kategori Contoh: Nama :	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "nama kategori harus diisi"	[√] Diterima [ ] Ditolak

c. Pengujian Hapus Data Kategori

Untuk penjelasan lebih lanjut tentang skenario hapus data kategori dapat dilihat pada tabel sebagai berikut.

Tabel 4. 13 Pengujian Hapus Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
hapus data kategori Contoh: Nama :	Data kategori dapat dihapus	Dapat memunculkan pesan data berhasil di hapus	[√] Diterima [ ] Ditolak

Sepatu			
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
hapus data kategori Contoh: Nama :	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”data kategori tidak dapat dihapus”	[√] Diterima [ ] Ditolak

3. Pengujian Pengolahan Data Ukuran

   a. Pengujian Tambah Data ukuran

   Untuk penjelasan lebih lanjut tentang skenario pengujian tambah data ukuran dapat dilihat pada tabel sebagai berikut.

Tabel 4. 14 Pengujian Tambah Data ukuran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data ukuran Contoh: Nama : L	Jika semua field terisi maka akan muncul pesan “data berhasil disimpan”	Dapat memunculkan pesan data berhasil di simpan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data ukuran Contoh:	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak

Nama :		”nama ukuran harus diisi”	

b. Pengujian Edit Data ukuran

Untuk penjelasan lebih lanjut tentang skenario pengujian edit data ukuran dapat dilihat pada tabel sebagai berikut.

Tabel 4. 15 Pengujian Edit Data ukuran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengedit data ukuran Contoh: Nama : XL	Jika semua field terisi maka akan muncul pesan “data berhasil disimpan”	Dapat memunculkan pesan data berhasil di simpan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengedit data kategori Contoh: Nama :	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”nama ukuran harus diisi”	[√] Diterima [ ] Ditolak

c. Pengujian Hapus Data Ukuran

Untuk penjelasan lebih lanjut tentang skenario hapus data ukuran dapat dilihat pada tabel sebagai berikut.

Tabel 4. 16 Pengujian Hapus Data Ukuran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
hapus data ukuran Contoh: Nama : L	Data ukuran dapat dihapus	Dapat memunculkan pesan data berhasil di hapus	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
hapus data ukuran Contoh: Nama :	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”data ukuran tidak dapat dihapus”	[√] Diterima [ ] Ditolak

4. Pengujian data warna

   a. Pengujian tambah data warna

      Untuk penjelasan lebih lanjut tentang skenario pengujian tambah data warna dapat dilihat pada tabel sebagai berikut.

Tabel 4. 17 Pengujian tambah data warna

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data warna Contoh: Nama : Hitam	Jika semua field terisi maka akan muncul pesan "data berhasil disimpan"	Dapat memunculkan pesan data berhasil di simpan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data warna Contoh: Nama :	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "nama warna harus diisi"	[√] Diterima [ ] Ditolak

b. Pengujian edit data warna

Untuk penjelasan lebih lanjut tentang skenario pengujian data warna dapat dilihat pada tabel sebagai berikut.

Tabel 4. 18 Pengujian edit data warna

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengedit data warna Contoh:	Jika semua field terisi maka akan muncul pesan	Dapat memunculkan pesan data berhasil di simpan	[√] Diterima [ ] Ditolak

Nama : Biru	“data berhasil disimpan”		
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data warna Contoh: Nama :	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”nama warna harus diisi”	[√] Diterima [ ] Ditolak

c. Pengujian hapus data warna

Untuk penjelasan lebih lanjut tentang skenario pengujian data produk dapat dilihat pada tabel sebagai berikut.

Tabel 4. 19 Pengujian hapus data warna

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
hapus data ukuran Contoh: Nama : L	Data ukuran dapat dihapus	Dapat memunculkan pesan data berhasil di hapus	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
hapus data ukuran Contoh: Nama :	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”data warna tidak	[√] Diterima [ ] Ditolak

		dapat dihapus”	

5. Pengujian Data Produk

a. Pengujian Tambah Data Produk

Untuk penjelasan lebih lanjut tentang skenario pengujian data produk dapat dilihat pada tabel sebagai berikut.

Tabel 4. 20 Pengujian Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data produk Contoh: kategori: Sepatu nama produk : Red Lame shoes deskripsi : berat : 1 harga : 60000 Stok : 10 Gambar : 	Jika semua field terisi semua maka akan muncul pesan “ data berhasil ditambah”	Dapat memunculkan pesan data berhasil ditambah	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukkan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data produk Contoh:	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ada	[√] Diterima [ ] Ditolak

kategori: nama produk : deskripsi : berat : harga : Stok : Gambar :		kesalahan didalam pengisian data, silahkan cek kembali.	

d. Pengujian Edit Data Produk

Untuk penjelasan lebih lanjut tentang skenario pengujian data produk dapat dilihat pada tabel sebagai berikut.

Tabel 4. 21 Pengujian Edit Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengedit data produk Contoh: kategori: sepatu nama produk : Red Lame shoes deskripsi : berat : 1 harga : 60000 Stok : 15 Gambar : 	Jika semua field terisi semua maka akan muncul pesan “ data berhasil ditambah”	Dapat memunculkan pesan data berhasil ditambah	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			

Data Masukkan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data barang Contoh: kategori: nama barang : deskripsi : berat : harga : Stok : Gambar :	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ada kesalahan didalam pengisian data, silahkan cek kembali.	[√] Diterima [ ] Ditolak

e. Pengujian Hapus Data Produk

Untuk penjelasan lebih lanjut tentang skenario hapus data produk dapat dilihat pada tabel sebagai berikut.

Tabel 4. 22 Pengujian Hapus Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
hapus data produk	Data produk dapat dihapus	Dapat memunculkan pesan data berhasil di hapus	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
hapus data produk	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”data produk tidak dapat dihapus”	[√] Diterima [ ] Ditolak

6. Pengujian Data Provinsi

Pengujian Tambah Data Provinsi

Untuk penjelasan lebih lanjut tentang skenario pengujian tambah data provinsi dapat dilihat pada tabel sebagai berikut.

Tabel 4. 23 Pengujian Tambah Data Provinsi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data provinsi Contoh: Nama Provinsi :	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
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama Provinsi : Jawa Barat	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Nama provinsi sudah ada”	[√] Diterima [ ] Ditolak

7. Pengujian Data Kota

Pengujian Tambah Data Kota

Untuk penjelasan lebih lanjut tentang skenario pengujian tambah data kota dapat dilihat pada tabel sebagai berikut.

Tabel 4. 24 Pengujian Tambah Data Kota

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
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
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Provinsi : Jawa Barat Nama Kota :	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Nama Kota dibutuhkan,ongkos kirim dibutuhkan”	[√] Diterima [ ] Ditolak

8. Pengujian Ongkos Kirim

   a. Pengujian Tambah Ongkos Kirim

      Untuk penjelasan lebih lanjut tentang skenario pengujian tambah ongkos kirim dapat dilihat pada tabel sebagai berikut.

Tabel 4. 25 Pengujian Ongkos Kirim

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

Nama Kota : Bandung	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah kota kirim	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah kota kirim	[√] Diterima [ ] Ditolak
Ongkos Kirim : 5000	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah kota kirim	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah kota kirim.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama kota (kosong)	Tampilkan pesan kesalahan “nama kota harus diisi”	Dapat menampilkan pesan kesalahan “nama kota harus disi”	[√] Diterima [ ] Ditolak
Ongkos kirim (kosong)	Tampilkan pesan kesalahan “ongkos kirim harus diisi”	Dapat menampilkan pesan kesalahan “ongkos kirim harus disi”	[√] Diterima [ ] Ditolak

b. Pengujian Edit Ongkos Kirim

Untuk penjelasan lebih lanjut tentang skenario pengujian edit ongkos kirim dapat dilihat pada tabel sebagai berikut.

Tabel 4. 26 Pengujian Edit Ongkos Kirim

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama Kota : Kabupaten Bandung	Jika semua field terisi atau benar dalam pengisian, maka akan dapat	Jika semua field terisi atau benar dalam pengisian, maka akan dapat	[√] Diterima [ ] Ditolak

	melanjutkan proses tambah kota kirim	melanjutkan proses tambah kota kirim	
Ongkos Kirim : 6000	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah kota kirim	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses tambah kota kirim.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama kota (kosong)	Tampilkan pesan kesalahan “nama kota harus diisi”	Dapat menampilkan pesan kesalahan “nama kota harus disi”	[√] Diterima [ ] Ditolak
Ongkos kirim (kosong)	Tampilkan pesan kesalahan “ongkos kirim harus diisi”	Dapat menampilkan pesan kesalahan “ongkos kirim harus disi”	[√] Diterima [ ] Ditolak

c. Pengujian Hapus Ongkos Kirim

Untuk penjelasan lebih lanjut tentang skenario pengujian hapus ongkos kirim dapat dilihat pada tabel sebagai berikut.

Tabel 4. 27 Pengujian Hapus Ongkos Kirim

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
-	Muncul pesan “ Apakah Anda Yakin ?”	Dapat Muncul pesan “ Apakah Anda Yakin ?”	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
-	Batal menghapus data	Tidak dapat menghapus data	[√] Diterima [ ] Ditolak

9. Pengujian pemesanan dan Laporan

Untuk penjelasan lebih lanjut tentang skenario pengujian pengolahan laporan dapat dilihat pada tabel sebagai berikut.

Tabel 4. 28 Pengujian Pengolahan Laporan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Status Order : lunas	Merubah status barang menjadi lunas	Dapat merubah status barang menjadi lunas	[√] Diterima [ ] Ditolak
No.Resi : 10004536	Merubah status barang menjadi dikirim	Dapat merubah status barang menjadi dikirim	[√] Diterima [ ] Ditolak
No.Resi : 10004536	Menuju *tracking* JNE dan merubah status barang menjadi delivered sesuai dengan tracking JNE	Dapat menuju *tracking* JNE dan merubah status barang menjadi delivered sesuai dengan tracking JNE	[√] Diterima [ ] Ditolak
Tanggal Awal : 10-08-2011	Tercantum pada text box tanggal awal	Dapat mencantumkan tanggal pada text box tanggal awal	[√] Diterima [ ] Ditolak
Tanggal Akhir : 20-08-2011	Tercantum pada text box tanggal akhir	Dapat mencantumkan tanggal pada text box tanggal akhir	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

-	Ubah status order dibatalkan	Ubah status order dibatalkan	[√] Diterima [ ] Ditolak
No.resi (kosong)	Tampilkan pesan kesalahan “No.Resi belum diisi”	Dapat menampilkan pesan kesalahan “No.Resi belum diisi”	[√] Diterima [ ] Ditolak
Tanggal Awal (kosong)	Muncul pesan kesalahan “ tanggal awal harus diisi”	Dapat menampilkan pesan kesalahan tanggal awal harus diisi”	[√] Diterima [ ] Ditolak
Tanggal Akhir (kosong)	Muncul pesan kesalahan “ tanggal akhir harus diisi”	Dapat menampilkan pesan kesalahan tanggal akhir harus diisi”	[√] Diterima [ ] Ditolak

10. Pengujian Retur

Untuk penjelasan lebih lanjut tentang skenario pengujian retur dapat dilihat pada tabel sebagai berikut.

Tabel 4. 29 Pengujian Retur

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
No.pemesanan : 5	Menampilkan barang yang akan diretur	Dapat menampilkan barang yang akan diretur	[√] Diterima [ ] Ditolak
Jumlah : 4	Menampilkan jumlah barang yang akan diretur	Dapat menampilkan jumlah barang yang akan diretur	[√] Diterima [ ] Ditolak
Alasan : Barang tidak sesuai pesanan	Menampilkan alasan barang yang akan diretur	Dapat menampilkan alasan barang yang akan diretur	[√] Diterima [ ] Ditolak
Keterangan : ukuran tidak sesuai dengan pesanan	Tercantum pada text box keterangan yang dicantumkan	Dapat tercantum pada text box keterangan yang dicantumkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
No.pemesanan (kosong)	Tampilkan pesan kesalahan ”No.pemesanan harus diisi.	Dapat menampilkan pesan kesalahan ”No.pemesanan harus diisi.	[√] Diterima [ ] Ditolak
Jumlah (kosong)	Menampilkan pesan kesalahan “ jumlah harus diisi”	Dapat menampilkan pesan kesalahan “jumlah harus diisi”	[√] Diterima [ ] Ditolak
Alasan (kosong)	Menampilkan pesan kesalahan “ alasan harus diisi”	Dapat menampilkan pesan kesalahan “alasan harus diisi”	[√] Diterima [ ] Ditolak
Keterangan (kosong)	Menampilkan pesan kesalahan “ keterangan harus diisi”	Dapat menampilkan pesan kesalahan “keterangan harus diisi”	[√] Diterima [ ] Ditolak

4.2.2.2 Fasilitas Member

1. Pengujian Menu Registrasi

   Untuk penjelasan lebih lanjut tentang skenario pengujian menu registrasi dapat dilihat pada tabel sebagai berikut.

Tabel 4. 30 Pengujian Menu Registrasi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Masukan data pendaftaran, contoh masukan : *Username* : rizal Nama : dachrur rizal Alamat Email : dachrur.rizal@gmail.com Alamat : Komplek Padasuka Indah B-55	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses registrasi.	*Form* registrasi terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses registrasi.	[√] Diterima [ ] Ditolak

Telepon : 087878112176 Provinsi : Jawa Barat Kota : Cimahi Kodepos : 40526 *Password* : 123456 Konfirmasi *password* : 123456			
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 akan muncul pesan kesalahan bahwa *field* tersebut harus terisi.	[√] Diterima [ ] Ditolak
Email (kosong)	Tampilkan pesan kesalahan “Email harus diisi”	Dapat menampilkan pesan kesalahan “Email harus diisi”	[√] Diterima [ ] Ditolak
Nama lengkap (kosong)	Tampilkan pesan kesalahan”Nama lengkap harus diisi”	Dapat menampilkan Pesan kesalahan “Nama lengkap harus diisi”	[√] Diterima [ ] Ditolak
Alamat (kosong)	Tampilkan pesan kesalahan”Alamat harus diisi”	Dapat menampilkan pesan kesalahan ”Alamat harus diisi”	[√] Diterima [ ] Ditolak
Provinsi (kosong)	Tampilkan pesan kesalahan ”Provinsi harus diisi”	Dapat menampilkan pesan kesalahan ”Provinsi harus diisi”	[√] Diterima [ ] Ditolak
Kota (kosong)	Tampilkan pesan kesalahan ”kota harus diisi”	Dapat menampilkan Pesan kesalahan ”kota harus diisi”	[√] Diterima [ ] Ditolak
Kode pos (kosong)	Tampilkan pesan kesalahan ”kode pos harus diisi”	Dapat menampilkan pesan kesalahan ”kode pos harus diisi”	[√] Diterima [ ] Ditolak
Password (kosong)	Tampilkan pesan	Dapat menampilkan	[√] Diterima

	kesalahan ”*password* harus diisi”	pesan kesalahan ”*password* harus diisi”	[ ] Ditolak
Konfirmasi *password* (kosong)	Tampilkan pesan kesalahan ”ulang *password* harus diisi”	Dapat menampilkan pesan kesalahan ”ulang *password* harus diisi”	[√] Diterima [ ] Ditolak

2. Pengujian Login Members

Untuk penjelasan lebih lanjut tentang skenario pengujian login members dapat dilihat pada tabel sebagai berikut.

Tabel 4. 31 Pengujian Login Member

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Username : rizal	Mengisikan username pada field *username*	Dapat mengisi field *username* dan dapat mengisi data login dengan benar	[√] Diterima [ ] Ditolak
Password : 123456	Mengisikan data *password* pada *field password*.	Mengisikan data *password* pada *field password*.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
U*sername* dan *Password* yang tidak terdaftar	Tampilkan pesan kesalahan “Akun yang anda masukan salah atau belum	Muncul pesan “Akun yang anda masukan salah atau belum terdaftar ”	[√] Diterima [ ] Ditolak

	terdaftar ”		
Username dan *Password* (kosong)	Tampilkan pesan kesalahan “Maaf *username* dan *password* harus diisi”	Dapat menampilkan pesan kesalahan ” Maaf *username* dan *password* harus diisi”	[√] Diterima [ ] Ditolak
Username (kosong)	Tampilkan pesan kesalahan ”*username* harus diisi”	Dapat menampilkan pesan kesalahan ” *username* harus diisi”	[√] Diterima [ ] Ditolak
Password (kosong)	Tampilkan pesan kesalahan ”*password* harus diisi”	Dapat menampilkan pesan kesalahan”*password* harus diisi”	[√] Diterima [ ] Ditolak

3. Pengujian Lupa *Password*

Untuk penjelasan lebih lanjut tentang skenario pengujian lupa *password* dapat dilihat pada tabel sebagai berikut.

Tabel 4. 32 Pengujian Lupa Password

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Email	Mengisi field email yang tersedia	Dapat mengisi field *email* dan dapat mengisi data email dengan benar	[√] Diterima [ ] Ditolak

Username	Mengisi username di field yang tersedia	Dapat mengisi field *username* dan dapat mengisi data username dengan benar	[√] Diterima [ ] Ditolak
Password Baru	Mengisi kolom password baru difield yang tersedia	Dapat mengisi field password baru dengan benar	[√] Diterima [ ] Ditolak
Konfirmasi Password	Mengisi kolom password baru difield yang tersedia	Dapat mengisi field password baru dengan benar	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Email(kosong)	Tampilkan pesan kesalahan "*Maaf email harus diisi*"	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 50 karakter.	[√] Diterima [ ] Ditolak
Username (kosong)	Tampilkan pesan kesalahan "Maaf *username*"	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 50 karakter.	[√] Diterima [ ] Ditolak
Password Baru (kosong)	Tampilkan pesan kesalahan "maaf *password* harus diisi"	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 50 karakter.	[√] Diterima [ ] Ditolak

Konfirmasi Password (kosong)	Tampilkan pesan kesalahan "maaf *password* harus diisi"	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 50 karakter.	[√] Diterima [ ] Ditolak

4. Pengujian Pesan barang

   a. Pengujian keranjang belanja

   Untuk penjelasan lebih lanjut tentang skenario pengujian keranjang belanja dapat dilihat pada tabel sebagai berikut.

Tabel 4. 33 Pengujian Keranjang Belanja

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Qty : 5	Tercantum pada *text box* data yang akan diubah	Dapat mengisi data jumlah dengan benar	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Qty : 1000	Menampilkan pesan "stok tidak mencukupi"	Dapat menampilkan pesan "stok tidak mencukupi"	[√] Diterima [ ] Ditolak

   b. Pengujian alamat pengiriman

   Untuk penjelasan lebih lanjut tentang skenario pengujian alamat pengiriman dapat dilihat pada tabel sebagai berikut.

Tabel 4. 34 Pengujian Alamat Pengiriman

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data masukkan : Nama :ijal Alamat : Antapani Kota : Bandung Provinsi: JAWA BARAT No telp : 0227277891	Data pengiriman baru akan disimpan di dalam *database*	Dapat data pengirimah baru akan disimpan di dalam *database*	[√] Diterima [ ] Ditolak

c. Pengujian Jenis Pengiriman

Untuk penjelasan lebih lanjut tentang skenario pengujian jenis pengiriman dapat dilihat pada tabel sebagai berikut.

Tabel 4. 35 Pengujian Jenis Pengiriman

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Jenis Pengiriman contoh masukan : JNE	Pilih Jenis Pengiriman [1 |2|3...]	User memilih jenis pengiriman dengan combo box yang tersedia	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Jenis Pengiriman contoh masukan : JNE	data kosong	User tidak memilih jenis pengiriman dengan combo box yang tersedia	[√] Diterima [ ] Ditolak

d. Pengujian Konfirmasi Pembayaran

Untuk penjelasan lebih lanjut tentang skenario pengujian konfirmasi pembayaran dapat dilihat pada tabel sebagai berikut.

Tabel 4. 36 Pengujian Konfirmasi Pembayaran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Jumlah : 500000 Tujuan Bank : BCA No. Transaksi : 95	Menampilkan *pesan " Terimakasih data pembayaran anda sudah kami terima"*	Dapat memunculkan *pesan " Terimakasih data pembayaran anda sudah kami terima"*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Jumlah (kosong) Tujuan Bank (kosong) No. Transaksi (kosong)	Tampilkan pesan kesalahan "Bank harus dipilih, No transaksi harus diisi, Jumlah Harus berupa angka"	Dapat menampilkan pesan kesalahan " Tampilkan pesan kesalahan "Bank harus dipilih, No transaksi harus diisi, Jumlah Harus berupa angka"	[√] Diterima [ ] Ditolak

5. Pengujian Retur

Untuk penjelasan lebih lanjut tentang skenario pengujian retur dapat dilihat pada tabel sebagai berikut:

Tabel 4. 37 Pengujian Retur

Kasus dan Hasil Uji (Data Benar)

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
No.pemesanan : 5	Menampilkan barang yang akan diretur	Dapat menampilkan barang yang akan diretur	[√] Diterima [ ] Ditolak
Jumlah : 4	Menampilkan jumlah barang yang akan diretur	Dapat menampilkan jumlah barang yang akan diretur	[√] Diterima [ ] Ditolak
Alasan : Barang tidak sesuai pesanan	Menampilkan alasan barang yang akan diretur	Dapat menampilkan alasan barang yang akan diretur	[√] Diterima [ ] Ditolak
Keterangan : ukuran tidak sesuai dengan pesanan	Tercantum pada text box keterangan yang dicantumkan	Dapat tercantum pada text box keterangan yang dicantumkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
No.pemesanan (kosong)	Tampilkan pesan kesalahan ”No.pemesanan harus diisi.	Dapat menampilkan pesan kesalahan ”No.pemesanan harus diisi.	[√] Diterima [ ] Ditolak
Jumlah (kosong)	Menampilkan pesan kesalahan “ jumlah harus diisi”	Dapat menampilkan pesan kesalahan “jumlah harus diisi”	[√] Diterima [ ] Ditolak
Alasan (kosong)	Menampilkan pesan kesalahan “ alasan harus diisi”	Dapat menampilkan pesan kesalahan “alasan harus diisi”	[√] Diterima [ ] Ditolak
Keterangan (kosong)	Menampilkan pesan kesalahan “ keterangan harus diisi”	Dapat menampilkan pesan kesalahan “keterangan harus diisi”	[√] Diterima [ ] Ditolak

4.2.3 Kesimpulan Pengujian Aplha

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

1. Jika data yang dimasukan benar maka sistem secara fungsional akan mengeluarkan keluaran sesuai dengan harapan.

2. Jika data yang dimasukan salah maka sistem akan menunjukkan data yang salah dan tombol simpan tidak dapat berfungsi.

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

4.3 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan melakukan wawancara dan membuat kuesioner mengenai kepuasan user dengan kandungan poin syarat *user friendly* untuk selanjutnya dibagikan kepada sebagian user dengan mengambil sample sebanyak 10 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

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

1. Lingkungan Penjual

Penelitian di lingkungan penjual dalam hal ini admin dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan di awal. Pengujian *betha* yang dilakukan pada lingkungan penjual adalah dengan menggunakan wawancara.

Wawancara

Metode wawancara adalah metode pengumpulan data dengan cara tanya jawab secara langsung kepada responden untuk menilai keadaan seseorang atau dengan dasar tujuan penelitian. Pertanyaan diajukan langsung ke manager toko selaku admin dari web low and lame dengan menggunakan teknik wawancara terstruktur yaitu dengan memberikan pertanyaan yang sudah disusun oleh narasumber secara sistematis dan lebih mendalam. Dari hasil wawancara tersebut maka akan terlihat penilaian penerapan sistem. Berikut adalah pertanyaan yang diajukan saat wawancara.

1. **Bagaimana dari segi tampilan perangkat lunak untuk pengolahan data apakah sudah bisa memudahkan pengguna?**

   Jawaban : Lumayan bisa dimengerti dari segi tampilan admin.

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

   Jawaban : Bahasa atau instruksi yang dipakai dapat memahami pengguna admin dengan baik

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

   Jawaban : produk yang diolah bisa tersusun dengan baik, dan bisa dikelola dengan mudah.

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

   Jawaban : Laporan penjualan sudah bisa tersusun dengan baik dan terkelola dengan baik juga.

Dari hasil wawancara tersebut dapat disimpulkan bahwa aplikasi ini sudah bisa membantu admin dalam hal mengelola data produk, laporan penjualan, dan sudah memenuhi tujuan dari pembuatan aplikasi *E-Commerce* Low and Lame ini.

2. Lingkungan Masyarakat Umum

Penelitian di lingkungan pada masyarakat umum khususnya yang sudah terbiasa dengan teknologi *internet*,dalam hal ini pengunjung dan member dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat memberikan kemudahan masyarakat umum untuk melakukan transaksi pembelian barang secara *online*. Pengujian *betha* yang dilakukan pada lingkungan masyarakat umum adalah dengan menggunakan kuisoner.

Kuesioner

Kuesioner disebarkan menggunakan teknik sampling yaitu *Simple Random Sampling* yang disebarkan kepada 12 orang (2 orang admin dan 10 orang pengguna). Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem. Kuesioner ini terdiri dari 5 pertanyaan (contoh kuesioner dapat diliihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 5, dengan ketentuan skala untuk tiap pertanyaan sebagai berikut :

Kuesioner Pengujian Betha Untuk Pengguna

1. **Apakah Anda setuju bahwa tampilan *website* Low and Lame cukup menarik dan mudah dipahami / *user friendly*?**

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

2. **Apakah Anda setuju bahwa *website* E-commerce ini dapat membantu Anda untuk mendapatkan informasi terbaru tentang produk yang ada di Low and Lame?**

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

3. **Apakah Anda setuju bahwa *website* ini akan lebih mempermudah Anda dalam melakukan transaksi pembelian di bandingkan transaksi secara manual?**

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

4. **Apakah Anda setuju dengan adanya *website* ini toko dapat memperluas pangsa pasanya?**

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

5. Apakah anda setuju bahwa sistem pembayaran yang digunakan akan lebih memudahkan anda?

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

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 Anda setuju bahwa tampilan *website* Low and Lame cukup menarik dan mudah dipahami / *user friendly*?

Tabel 4 1 Hasil Pengujian Kuesioner No.1

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	0	0%

	2	Setuju	6	60%
	3	Cukup Setuju	4	40%
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			**10**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak, 6 atau 60% menyatakan setuju, 4 atau 40% menyatakan cukup setuju, bahwa tampilan *website* cukup menarik dan mudah untuk dipahami/ *user friendly*.

2. Apakah anda setuju *website* E-commerce ini dapat membantu Anda untuk mendapatkan informasi terbaru tentang produk yang ada di Low and Lame?

Tabel 4 2 Hasil Pengujian Kuesioner No.2

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	0	0
	2	Setuju	7	70%
	3	Cukup Setuju	1	10%
	4	Kurang Setuju	2	20%
	5	Tidak Setuju	0	0
Jumlah			**10**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 7 orang atau 70% menyatakan setuju, 1 orang atau 10% menyatakan cukup setuju dan 2 orang

atau 20% menyatakan kurang setuju, bahwa informasi yang diberikan tentang produk terbaru yang ada di Low and Lame.

3. Apakah anda setuju bahwa *website* ini akan lebih mempermudah Anda dalam melakukan transaksi pembelian di bandingkan transaksi secara manual?

Tabel 4 3 Hasil Pengujian Kuesioner No.3

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	5	50%
	2	Setuju	2	20%
	3	Cukup Setuju	0	0%
	4	Kurang Setuju	3	30%
	5	Tidak Setuju	0	0
Jumlah			**10**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 5 orang atau 50 % menyatakan sangat setuju, 2 atau 20 % menyatakan setuju, 3 atau 30% menyatakan kurang setuju bahwa *website* ini akan lebih memudahkan pelanggan dalam melakukan transaksi pembelian barang.

4. Apakah dengan adanya *website* ini toko dapat memperluas pangsa pasanya?

Tabel 4 4 Hasil Pengujian Kuesioner No.4

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
4	1	Sangat Setuju	3	30%
	2	Setuju	4	40%
	3	Cukup Setuju	3	30%

	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			**10**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 orang atau 30% menyatakan sangat setuju, 4 orang atau 40% menyatakan setuju, 3 orang atau 30% menyatakan cukup setuju bahwa *website* ini akan memperluas pangsa pasar.

5. Apakah anda setuju bahwa sistem pembayaran yang digunakan akan lebih memudahkan anda.

Tabel 4 5 Hasil Pengujian Kuesioner No.5

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
5	1	Sangat Setuju	2	20%
	2	Setuju	2	20%
	3	Cukup Setuju	5	50%
	4	Kurang Setuju	1	10%
	5	Tidak Setuju	0	0
Jumlah			**10**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 orang atau 20% menyatakan sangat setuju, 2 orang atau 20% menyatakan setuju, 5 orang atau 50% menyatakan cukup setuju, 1 atau 10% menyatakan kurang setuju bahwa sistem pembayaran yang digunakan akan lebih memudahkan.

4.4 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 Low and Lame yaitu dapat mepermudah dalam proses promosi dan penjualan produk yang ditawarkan tidak lagi secara konvensional, pelanggan dapat melakukan transaksi pembelian secara online, tampilannya *user friendly,* memudahkan dalam pengelolaan data barang dan memudahkan dalam membuat laporan penjualan.