BAB IV

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan implementasi dan pengujian tarhadap sistem yang baru. Tahap ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemrograman yang akan digunakan. Setelah implementasi maka dilakukan pengujian sistem yang baru dimana akan dilihat kekurangan-kekurangan pada aplikasi yang baru untuk pengembangan sistem selanjutnya.

4.1 Implementasi

Setelah sistem dianalisis dan didesain secara rinci, maka selanjutnya akan menuju pada tahap implementasi.

Tujuan implementasi adalah untuk mengkonfirmasikan modul program perancangan pada para pelaku sistem sehingga pengguna dapat memberikan masukan kepada pembangun sistem.

Adapun spesifikasi perangkat keras yang digunakan untuk membangun sistem penjualan barang ini adalah sebagai berikut :

4.1.1 Kebutuhan Perangkat Keras

Adapun spesifikasi kebutuhan perangkat keras yang digunakan untuk membangun Sistem Transaksi *On-line* ikhrom dan handuk *letter* di CV. Sentoso adalah sebagai berikut :

Tabel IV.1 Perangkat Keras Yang Digunakan

Perangkat Keras	Spesifikasi (Min)
Processor	Kecepatan 1GHz
RAM	128Mb
Harddisk	20Gb
Monitor	Resolution 1024 x 768
VGA	32Mb
Koneksi Internet	64 Kbps

4.1.2 KebutuhanPerangkat Lunak

Perangkat lunak yang digunakan untuk mengimplementasikan Sistem Sistem Transaksi *On-line* ikhrom dan handuk *letter* di CV. Sentoso adalah sebagai berikut :

Tabel IV.2 Perangkat Lunak yang digunakan

Perangkat Lunak	Spesifikasi (Min)
Sistem Operasi	Windows XP
Bahasa Pemrograman	PHP 5.3.1
Web Server	XAMPP version 1.7.3
Database Server	MySQL
Web Browser	Mozilla firefox 3
Code Ubahor	Macromedia Dreamweaver 8
Image Ubahor	Adobe Photoshop CS3
DFD Modeler	Power Designer 6

4.1.3 Implementasi Basis Data

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

Tabel IV.3 Implementasi Basis Data

Tabel Barang

```
CREATE TABLE IF NOT EXISTS `barang` (
  `barang_id` int(11) NOT NULL auto_increment,
  `barang_kode` varchar(8) NOT NULL,
  `kategori_id` int(5) NOT NULL,
  `jenis_id` int(11) NOT NULL,
  `barang_warna` varchar(30) NOT NULL,
  `barang_nama` varchar(50) NOT NULL,
  `barang_harga` varchar(60) NOT NULL,
```

```
  `barang_diskon` double NOT NULL,
  `barang_berat` varchar(60) NOT NULL,
  `barang_keterangan` text NOT NULL,
  `barang_stok` varchar(20) NOT NULL,
  `barang_sisa_stok` varchar(20) NOT NULL,
  `barang_gambar` varchar(50) NOT NULL,
  `status` varchar(100) NOT NULL,
  `ukuran` varchar(20) NOT NULL,
  `pelanggan_id` int(10) NOT NULL,
  `petugas_id` int(10) NOT NULL,
  `jasa_id` int(10) NOT NULL,
  PRIMARY KEY  (`barang_id`),
  CONSTRAINT `barang_ibfk_1` FOREIGN KEY (`kategori_id`) REFERENCES
`barang` (`kategori_id`),
  CONSTRAINT `barang_ibfk_2` FOREIGN KEY (`jenis_id`) REFERENCES
`barang` (`jenis_id`),
  CONSTRAINT `barang_ibfk_3` FOREIGN KEY (`pelanggan_id`) REFERENCES
`barang` (`pelanggan_id`),
  CONSTRAINT `barang_ibfk_4` FOREIGN KEY (`jasa_id`) REFERENCES
`barang` (`jasa_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=15 ;
```

Tabel Kategori

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `kategori_id` int(5) NOT NULL auto_increment,
  `kategori_nama` varchar(50) NOT NULL,
  PRIMARY KEY  (`kategori_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;
```

Tabel Jenis

```
CREATE TABLE IF NOT EXISTS `jenis` (
  `jenis_id` int(11) NOT NULL auto_increment,
  `kategori_id` int(11) NOT NULL,
  `jenis_nama` varchar(30) NOT NULL,
  PRIMARY KEY  (`jenis_id`),
  CONSTRAINT `jenis_ibfk_1` FOREIGN KEY (`kategori_id`) REFERENCES
`jenis` (`kategori_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=14 ;
```

Tabel Pelanggan

```
CREATE TABLE IF NOT EXISTS `pelanggan` (
  `pelanggan_id` int(5) NOT NULL auto_increment,
  `provinsi_id` int(5) NOT NULL,
  `kota_id` int(5) NOT NULL,
  `pelanggan_nama_lengkap` varchar(30) NOT NULL,
  `pelanggan_email` varchar(50) NOT NULL,
  `pelanggan_username` varchar(50) NOT NULL,
  `pelanggan_password` varchar(150) NOT NULL,
```

```
  `pelanggan_alamat` varchar(100) NOT NULL,
  `pelanggan_kode_pos` varchar(60) NOT NULL,
  `pelanggan_telepon` varchar(60) NOT NULL,
  `pelanggan_hp` varchar(50) NOT NULL,
  `pelanggan_tanggal_aktif` date NOT NULL,
  `pelanggan_status` varchar(50) NOT NULL,
  PRIMARY KEY  (`pelanggan_id`),
  CONSTRAINT `pelanggan_ibfk_2` FOREIGN KEY (`kota_id`) REFERENCES
`pelanggan` (`kota_id`),
  CONSTRAINT `pelanggan_ibfk_1` FOREIGN KEY (`provinsi_id`)
REFERENCES `pelanggan` (`provinsi_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=13 ;
```

Tabel Petugas

```
CREATE TABLE IF NOT EXISTS `petugas` (
  `petugas_id` int(5) NOT NULL auto_increment,
  `petugas_username` varchar(30) NOT NULL,
  `petugas_password` varchar(150) NOT NULL,
  `petugas_email` varchar(50) NOT NULL,
  `petugas_telepon` varchar(50) NOT NULL,
  `petugas_hp` varchar(50) NOT NULL,
  `petugas_alamat` varchar(100) NOT NULL,
  `petugas_status` varchar(50) NOT NULL,
  `status` varchar(30) NOT NULL,
  PRIMARY KEY  (`petugas_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

Tabel Pemesanan

```
CREATE TABLE IF NOT EXISTS `pemesanan` (
  `pemesanan_id` int(5) NOT NULL auto_increment,
  `username` varchar(30) NOT NULL,
  `kota_pengiriman` varchar(50) NOT NULL,
  `jasa_id` int(5) NOT NULL,
  `pemesanan_id_transaksi` varchar(15) NOT NULL,
  `pemesanan_bayar` int(11) NOT NULL,
  `pemesanan_ongkos_kirim` int(11) NOT NULL,
  `pemesanan_status_pesan` varchar(50) NOT NULL,
  `pemesanan_status_bayar` varchar(50) NOT NULL,
  `pemesanan_tgl_pesan` date NOT NULL,
  `pemesanan_tgl_konfirmasi` date NOT NULL,
  `pemesanan_tgl_bayar` date NOT NULL,
  `pemesanan_tgl_kirim` date NOT NULL,
  `pemesanan_penerima` varchar(50) NOT NULL,
  `pemesanan_alamat_penerima` varchar(60) NOT NULL,
  `pemesanan_telepon` varchar(50) NOT NULL,
```

```
  `pemesanan_hp` varchar(50) NOT NULL,
  `pemesanan_email_penerima` varchar(50) NOT NULL,
  `pemesanan_tgl_terima` date NOT NULL,
  `pemesanan_bank` varchar(40) NOT NULL,
  `pemesanan_no_account` varchar(50) NOT NULL,
  `pemesanan_nilai_transfer` varchar(60) NOT NULL,
  `pemesanan_keterangan` varchar(60) NOT NULL,
  `pelanggan_id` int(10) NOT NULL,
  `petugas_id` int(10) NOT NULL,
  `barang_id` int(10) NOT NULL,
  PRIMARY KEY  (`pemesanan_id`),
  CONSTRAINT `pemesanan_ibfk_1` FOREIGN KEY (`jasa_id`) REFERENCES
`pemesanan` (`jasa_id`),
  CONSTRAINT `pemesanan_ibfk_2` FOREIGN KEY (`pelanggan_id`)
REFERENCES `pemesanan` (`pelanggan_id`),
  CONSTRAINT `pemesanan ibfk 3` FOREIGN KEY (`petugas id`) REFERENCES
`pemesanan` (`petugas_id`),
  CONSTRAINT `pemesanan_ibfk_4` FOREIGN KEY (`barang_id`) REFERENCES
`pemesanan` (`barang_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=52 ;
```

Tabel Detail Pemesanan

```
CREATE TABLE IF NOT EXISTS `detail_pemesanan` (
  `pemesanan_id` int(5) NOT NULL,
  `detail_barang_id` int(5) NOT NULL,
  `detail_barang_harga` varchar(60) NOT NULL,
  `detail_pemesanan_jumlah` varchar(60) NOT NULL,
  `detail_pemesanan_session` varchar(60) NOT NULL,
  `ukuran` varchar(30) NOT NULL,
  `letter` varchar(30) NOT NULL,
  `warna` varchar(30) NOT NULL,
  KEY `pemesanan_id` (`pemesanan_id`),
  CONSTRAINT `detail_pemesanan_ibfk_1` FOREIGN KEY (`pemesanan_id`)
REFERENCES `detail_pemesanan` (`pemesanan_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Tabel Retur

```
CREATE TABLE IF NOT EXISTS `retur` (
  `retur_id` int(5) NOT NULL auto_increment,
  `pelanggan_nama_lengkap` varchar(20) NOT NULL,
  `pemesanan_id` varchar(20) NOT NULL,
  `barang_nama` varchar(20) NOT NULL,
  `retur_tanggal` date NOT NULL,
  `retur_aksi` varchar(150) NOT NULL,
  `petugas_id` int(10) NOT NULL,
  PRIMARY KEY  (`retur_id`),
  CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`pemesanan_id`) REFERENCES
```

```
`retur` (`pemesanan_id`),
  CONSTRAINT `retur_ibfk_2` FOREIGN KEY (`petugas_id`) REFERENCES
`retur` (`petugas_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

Tabel Detail_retur

```
CREATE TABLE IF NOT EXISTS `detail_retur` (
  `retur_id` int(5) NOT NULL,
  `retur_keterangan` varchar(50) NOT NULL,
  `retur_jumlah` int(4) NOT NULL,
  KEY `retur_id` (`retur_id`),
  CONSTRAINT `detail_retur_ibfk_1` FOREIGN KEY (`retur_id`)
REFERENCES `detail_retur` (`retur_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

Tabel Provinsi

```
CREATE TABLE IF NOT EXISTS `provinsi` (
  `provinsi_id` int(5) NOT NULL auto_increment,
  `provinsi_nama` varchar(50) NOT NULL,
  PRIMARY KEY  (`provinsi_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=35 ;
```

Tabel kota

```
CREATE TABLE IF NOT EXISTS `kota` (
  `kota_id` int(5) NOT NULL auto_increment,
  `provinsi_id` int(5) NOT NULL,
  `kota_nama` varchar(50) NOT NULL,
  PRIMARY KEY  (`kota_id`),
  CONSTRAINT `kota_ibfk_1` FOREIGN KEY (`provinsi_id`) REFERENCES
`kota` (`provinsi_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=427 ;
```

Tabel Jasa Pengiriman

```
CREATE TABLE IF NOT EXISTS `jasa_pengiriman` (
  `jasa_id` int(5) NOT NULL auto_increment,
  `petugas_id` int(10) NOT NULL,
  `jasa_pengiriman_id` int(5) NOT NULL,
  `kota_id` int(5) NOT NULL,
  `detail_jasa_biaya1` int(11) NOT NULL,
  PRIMARY KEY  (`jasa_id`),
  CONSTRAINT `jasa_pengiriman_ibfk_2` FOREIGN KEY (`kota_id`)
REFERENCES `jasa_pengiriman` (`kota_id`),
  CONSTRAINT `jasa_pengiriman_ibfk_1` FOREIGN KEY (`petugas_id`)
REFERENCES `jasa_pengiriman` (`petugas_id`);
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=121 ;
```

Tabel komentar

```
CREATE TABLE IF NOT EXISTS `komentar` (
  `komentar_id` int(5) NOT NULL auto_increment,
  `pemesan_username` varchar(30) NOT NULL,
  `petugas_id` int(5) NOT NULL,
  `komentar_keterangan` longtext NOT NULL,
```

```
  `komentar_tanggal` date NOT NULL,
  `komentar_status` varchar(50) NOT NULL,
  `pelanggan_id` int(10) NOT NULL,
  PRIMARY KEY  (`komentar_id`),
  CONSTRAINT `komentar_ibfk_1` FOREIGN KEY (`pelanggan_id`)
REFERENCES `komentar` (`pelanggan_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

4.1.4 Implementasi Antarmuka

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

A. Implementasi Antarmuka Admin

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk admin dapat dilihat pada tabel berikut :

Tabel IV.4 Implementasi Antarmuka Admin

Menu	Deskripsi	Nama File
Login Admin	Digunakan untuk halaman login admin	index.php
Beranda	Tampilan antarmuka admin setalah melakukan proses *login*	menu.php
Porfil	Digunakan untuk mengubah data profil dan data *password*	my_profile.php
Kategori	Digunakan untuk melihat, menambah, dan merubah data kategori.	res_kategori.php
Jenis	Digunakan untuk melihat, menambah, dan merubah data jenis.	res_jenis.php
Barang	Digunakan untuk melihat, menambah, dan merubah data barang.	res_barang.php
Pesanan	Digunakan untuk melihat, dan mengolah data pesanan.	res_belanja.php

Laporan	Digunakan untuk melihat data laporan berdasarkan jenis pembayaran berdasarkan waktu yang telah ditentukan dan cetak laporan.	lap_pesanan.php
Komentar	Digunakan untuk melihat, dan mengelolah data komentar.	komentar.php
Keluar	Digunakan operator untuk dapat keluar dari aplikasi.	logout.php

B. Implementasi Antarmuka Pelanggan

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk pelanggan dapat dilihat pada tabel berikut :

Tabel IV.5 Implementasi Antarmuka Pelanggan

Menu	Deskripsi	Nama File
Registrasi pelanggan	Digunakan untuk melakukan pelanggan baru.	cpendaftaran.php
Login *member*	Digunakan untuk sebagai halaman login pelanggan	index.php
Lupa *password*	Digunakan jika pelanggan lupa dengan *password*.	lupa_password.php
Halaman Utama	Digunakan untuk melihat halaman beranda	op.php
Cara Belanja	Digunakan untuk melihat langkah-langkah untuk melakukan pembelian barang	carabelanja.php
Cara Bayar	Digunakan untuk melihat langkah-langkah untuk melakukan pembayaran	carabayar.php
Konfirmasi Pembayaran	Digunakan jika pelanggan ingin melakukan konfirmasi pembayarn.	confirm.php
Konfrimasi Retur	Digunakan jika pelanggan ingin melakukan retur barang	retur.php
Profil	Digunakan untuk melihat dan mengubah profil dan *password* pelanggan.	edit_profile.php
Komentar	Digunakan untuk melihat barang yang telah dikomentari	komentar.php

History pesanan	Digunakan untuk melihat *history* pembelian yang pernah dilakukan.	history.php
Keranjang Belanja	Digunakan untuk melihat, merubah, menghapus data pembelian yang akan dilakukan.	scart.php

4.2 Pengujian Perangkat Lunak

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 sistem ini adalah metode pengujian *black-box.* Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak.

4.2.1 Pengujian Alpa

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

4.2.1.1.1 Skenario Pengujian Aplikasi Pelanggan

Skenario Pengujian Aplikasi Pelanggan dapat dilihat pada tabel 4.6 berikut :

Tabel IV.6 Skenario Pengujian Aplikasi Pelanggan

Item Uji	Detail Pengujian	Jenis Pengujian
Registrasi Pelanggan	Menampilkan *form* pelanggan baru	*Black Box*
	Isi *form* pendaftaran	*Black Box*
	Simpan data baru	*Black Box*
Login	Isi data login	*Black Box*
	Verifikasi *Email*	*Black Box*
	Verifikasi *password*	*Black Box*
Ganti alamat utama	Menampilkan *form* ganti alamat utama	*Black Box*

	Ubah data alamat pelanggan	*Black Box*
	Simpan perubahan	*Black Box*
Ganti *Password*	Menampilkan *form* ganti *password*	*Black Box*
	Isi *form* ganti *password*	*Black Box*
	Simpan data baru	*Black Box*
Komentar Anda	Menampilakan komentar barang	*Black Box*
History	Menampilkan history transaksi pembayaran	*Black box*
Keranjang Belanja	Menampilkan data pembelian	*Black Box*
	Mengupdate data pembelian	*Black Box*
	Menghapus data pembelian	*Black Box*
	Menampilkan konfirmasi transaksi	*Black Box*

4.2.1.1.2 Skenario Pengujian Aplikasi Pelanggan

Skenario Pengujian Aplikasi Admin dapat dilihat pada tabel 4.7 berikut :

Tabel IV.7 Skenario Pengujian Aplikasi Admin

Item Uji	Detail Pengujian	Jenis Pengujian
Login	Verifikasi Login	*Black-box*
Pengolahan Data kategori	Tambah kategori	*Black-box*
	Ubah kategori	*Black-box*
	Hapus Kategori	*Black-box*
Pengolahan Data Jenis	Tambah jenis	*Black-box*
	Ubah Jenis	*Black-box*
	Hapus Jenis	*Black-box*
Pengolahan Data Barang	Tambah Barang	*Black-box*
	Ubah barang	*Black-box*
	Hapus Barang	*Black-box*
Pengolahan data provinsi	Tambah provinsi	*Black-box*
	Ubah provinsi	*Black-box*
	Hapus provinsi	*Black-box*
Pengolahan data kota	Tambah kota	*Black-box*
	Ubah kota	*Black-box*
	Hapus kota	*Black-box*
Pengolahan data detail jasa pengiriman	Tambah kota pengiriman	*Black-box*
	Ubah kota pengiriman	*Black-box*
	Hapus kota pengiriman	*Black-box*
Pengolahan data Jasa pengiriman	Tambah jasa pengiriman	*Black-box*
	Ubah jasa pengiriman	*Black-box*
	Hapus jasa pengiriman	*Black-box*

Pengohalan Petugas	Tambah Petugas	*Black-box*
	Ubah petugas	*Black-box*
	Hapus petugas	*Black-box*
Pengolahan data pesanan	Ubah data pesanan	*Black-box*
	Cetak data pesanan	*Black-box*
Pengolahan data retur	Ubah data retur,	*Black-box*
	hapus data retur	*Black-box*
Pengolahan komentar	Tambah komentar	*Black-box*
	Ubah status komentar	*Black-box*
	Hapus data komentar	*Black-box*

4.2.2 Kasus dan Hasil Pengujian

A. Pengujian Registrasi Pelanggan

Berikut dapat dilihat pengujian dari registrasi pelanggan yang terdapat pada tabel 4.8 dibawah ini :

Tabel IV.8 Pengujian Registrasi Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data pendaftaran, contoh masukan : Nama : YuliYanti Provinsi : Jawa Barat Kota : Bandung Alamat : Jl. Marga Asri VI F No. C-180 Kodepos : 40215 Telepon : 0226040148 No HP : 083115347025 Email : yuliyanti1207@gmail.com *Username* : yuli *Password* : 123456 Konfirmasi *password* : 123456	Jika semua field terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses registrasi.	Form registrasi di disi semua maka sesuai yang di harapkan	[√] Diterima [ ] Ditolak
Klik tombol Daftar	Jika data masukan	Tampil pesan	[√] Diterima

	benar muncul pesan "Terima Kasih Anda Telah Berhasil Registrasi. Silahkan cek email untuk melakukan Aktifasi"	"Terima Kasih Anda Telah Berhasil Registrasi. Silahkan Login	[ ] 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.	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

B. Pengujian *Login* Pelanggan

Berikut dapat dilihat pengujian login pelanggan pada tabel 4.9 berikut ini :

Tabel IV.9 Pengujian Login Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan *username* : yuli	Mengisikan data *username* pada *field username*.	Dapat mengisi *field username* dan dapat mengisi data login dengan benar	[√] Diterima [ ] Ditolak
Memasukan *Password* : 123456	Mengisikan data *password* pada *field password*. Jika benar akan masuk ke menu utama	Mengisikan data *password* pada *field password*. Jika benar akan masuk ke menu utama	[√] Diterima [ ] Ditolak
Login	Data *username* dan data *password* dicari	login dapat berfungsi sesuai dengan yang diharapkan	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data	Tidak dapat login	Muncul pesan	[√] Diterima

username atau *password* yang tidak terdaftar		"*Username* atau *Password* salah	[ ] Ditolak
mengosongkan data *username* atau *password*	Tidak dapat login	Dapat menampilkan pesan kesalahan " *Username* atau *Password* salah"	[√] Diterima [ ] Ditolak

C. Pengunjian *Logout*

Berikut dapat dilihat dari pengujian *logout* pada tabel 4.10dibawah ini :

Tabel IV.10 Pengujian Logout

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

D. Pengujian Ubah Profil

Berikut dapat dilihat pengujian dari ubah profil pada tabel 4.11 dibawah ini :

Tabel IV.11 Pengujian Ubah Profil

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Ubah profil	Menampilkan *form* ubah profil	Dapat menampilkan *form* ubah profil	[√] Diterima [ ] Ditolak
Data profil yang akan diubah	Tercantum pada text box data yang akan diubah	Dapat mengisi *field* data yang akan diubah dengan benar	[√] Diterima [ ] Ditolak
Simpan	Data berhasil di simpan di*database*	Tampil pesan "Data berhasil diubah"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data masukan ada	Menampilkan pesan	Dapat	[√] Diterima

yang tidak diisi	kesalahan bahwa field tersebut harus terisi.	menampilkan pesan kesalahan bahwa field tersebut harus terisi.	[ ] Ditolak

E. Pengujian Ganti *Password* Pelanggan

Berikut dapat dilihat pengujian dari ganti *password* pelanggan pada tabel 4.12 berikut ini :

Tabel IV.12 Pengujian Ganti Password Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Ganti *password*	Menampilkan *form* ganti *password*	Dapat menampilkan *form* ganti *password*	[√] Diterima [ ] Ditolak
data yang dimasukkan semuanya benar	*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**
Salah satu field belum terisi	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi” bila ada field yang belum diisi	[√] Diterima [ ] Ditolak
Data *password* lama salah	muncul pesan "password salah"	Dapat menampilkan pesan kesalahan ”*Password* salah”	[√] Diterima [ ] Ditolak
Data *password* baru dan konfirmasi *password* baru tidak sama	Muncul pesan	Dapat menampilkan pesan "*Password* salah"	[√] Diterima [ ] Ditolak
Data kosong	Muncul pesan kesalahan field harus diisi	Dapat menampilkan pesan kesalahan field harus diisi	[√] Diterima [ ] Ditolak

F. Pengujian Komentar Anda

Berikut dapat dilihat pengujian dari komentar anda pada tabel 4.13 berikut ini :

Tabel IV.13 Pengujian Komentar Anda

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Komentar Anda	Menampilkan menu komentar anda	Tampil menampilkan menu komentar anda	[√] Diterima [ ] Ditolak

G. Pengujian Keranjang Belanja

Berikut dapat dilihat pengujian dari keranjang belanja pada tabel 4.14 berikut ini :

Tabel IV.14 Pengujian Keranjang Belanja

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
keranjang belanja	Menampilkan *form* keranjang belanja	Dapat menampilkan *form* keranjang belanja	[√] Diterima [ ] Ditolak
Data jumlah	Tercantum pada text box data yang akan diubah	Dapat mengisi data jumlah dengan benar	[√] Diterima [ ] Ditolak
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

H. Pengujian Menu Kategori

Berikut dapat dilihat pengujian dari menu produk pada tabel 4.15 dibawah ini:

Tabel IV.15 Pengujian Menu Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
kategori	Menampilkan data produk yang ada di *database*	Dapat menampilkan data produk yang ada di *data base*	[√] Diterima [ ] Ditolak
Detail pada data	Menampilkan detail	Dapat menampilkan detail	[√] Diterima

produk	produk	produk	[ ] Ditolak
Beli	Menuju ke keranjang belanja	Dapat menampilkan keranjang belanja	[√] Diterima [ ] Ditolak

I. Pengujian *Form* Pembayaran

Berikut dapat dilihat pengujian dari *form* pembayaran pada tabel 4.16 berikut ini :

Tabel IV.16 Pengujian Form Pembayaran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data konfirmasi pembayaran, contoh No. Transakdi : TRX236-01 Nama Bank : BNI Atas Nama : YuliYanti Jumlah Transfer : 125000	Menampilkan *form* konfirmasi	Dapat memunculkan pesan form konfirmasi.	[√] Diterima [ ] Ditolak
Memasukan data kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi”	[√] Diterima [ ] Ditolak

A. Pengujian *Form History*

Berikut dapat dilihat pengujian dari menu history pembayaran pada tabel 4.17 berikut ini :

Tabel IV.17 Pengujian Menu History

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
history	Menampilkan *history*	Dapat menampilkan form *history*	[√] Diterima [ ] Ditolak
No. Transaksi	Menampilkan detail transaksi	Dapat menampilkan detail transaksi	[√] Diterima [ ] Ditolak

B. Pengujian *Login* Admin

Berikut dapat dilihat pengujian dari *login* admin pada tabel 4.18 berikut ini :

Tabel IV.18 Pengujian Login Admin

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 : admin	Tercantum pada text box *password*	Dapat mengisi field *password* dengan benar	[√] Diterima [ ] Ditolak
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 "*Login* tidak valid"	[√] Diterima [ ] Ditolak
Data *username* kosong	Tidak dapat login	Muncul pesan kesalahan "*username* kosong."	[√] Diterima [ ] Ditolak
Data *Password* kosong	Tidak dapat login	Muncul pesan kesalahan "*Password* kosong."	[√] Diterima [ ] Ditolak

C. Pengujian Ganti *Password* Admin

Berikut dapat dilihat pengujian dari ganti *password* admin pada tabel 4.19 berikut ini :

Tabel IV.19 Pengujian Ganti Password Admin

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
edit *password*	Menampilkan *form* ganti *password*	Dapat menampilkan *form* ganti *password*	[√] Diterima [ ] Ditolak
Memasukan data yang benar	*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**
Memasukan	muncul pesan	Dapat menampilkan	[√] Diterima

password lama yang salah	"*password* lama yang anda masukkan salah"	pesan kesalahan ”*password* lama yang anda masukkan salah”	[ ] Ditolak
Memasukan data *password* baru dan konfirmasi *password* baru yang tidak sama	muncul pesan "silahkan periksa *password* baru! tidak sama"	Dapat menampilkan pesan "silahkan periksa *password* baru! tidak sama"	[√] Diterima [ ] Ditolak
Mengosongkan semua *field*	muncul pesan "*Password* lama Anda belum diisi"	dapat menampilkan pesan "*Password* lama Anda belum diisi"	[√] Diterima [ ] Ditolak

D. Pengujian Pengolahan Data Kategori

Berikut dapat dilihat pengujian dari pengolahan data kategori pada tabel 4.20 berikut ini :

Tabel IV.20 Pengujian Pengolahan Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kategori	Menampilkan *form* lihat kategori	Dapat menampilkan *form* lihat kategori	[√] Diterima [ ] Ditolak
Ubah	Menampilkan *form* perbaharui kategori	Dapat menampilkan *form* perbaharui kategori	[√] Diterima [ ] Ditolak
Simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak

E. Pengujian Tambah Kategori

Berikut dapat dilihat pengujian dari tambah kategori pada tabel 4.21 berikut ini :

Tabel IV.21 Pengujian Tambah Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tambah kategori	Menampilkan *form* tambah kategori	Dapat menampilkan *form* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data kategori Contoh: Kategori: Handuk	Jika semua field terisi maka akan muncul pesan “data berhasil disimpan”	Dapat memunculkan pesan data berhasil di simpan	[√] Diterima [ ] Ditolak
Simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di simpan”	[√] Diterima [ ] Ditolak
Batal	Kembali kemenu sebelumnya	Tampil menu	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data kategori masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”nama dibutuhkan”	[√] Diterima [ ] Ditolak

F. Pengujian Pengolahan Data Jenis

Berikut dapat dilihat pengujian dari pengolahan data jenis pada tabel 4.22 berikut ini :

Tabel IV.22 Pengujian Pengolahan Data Jenis

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Jenis	Menampilkan *form* lihat jenis	Dapat menampilkan *form* lihat jenis	[√] Diterima [ ] Ditolak
Ubah	Menampilkan *form* perbaharui jenis	Dapat menampilkan *form* perbaharui jenis	[√] Diterima [ ] Ditolak

	dan kategori jenis	dan kategori jenis	
Simpan	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[√] Diterima [ ] Ditolak
Batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "ada kesalahan didalam pengisian data, silahkan cek kembali"	[√] Diterima [ ] Ditolak

G. Pengujian Tambah Jenis

Berikut dapat dilihat pengujian dari tambah jenis pada tabel 4.23 berikut ini :

Tabel IV.23 Pengujian Tambah Jenis

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tambah	Menampilkan *form* tambah jenis	Dapat menampilkan *form* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data jenis Contoh: Kategori jenis: Handuk Nama Jenis : Letter	Jika semua field terisi maka akan muncul pesan "data berhasil disimpan"	Dapat memunculkan pesan data berhasil di simpan	[√] Diterima [ ] Ditolak
Simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan "data berhasil di simpan"	[√] Diterima [ ] Ditolak
Batal	Kembali kemenu sebelumnya	Tampil menu	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data kategori masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "nama dibutuhkan"	[√] Diterima [ ] Ditolak

H. Pengujian Pengolahan Data Barang

Berikut dapat dilihat pengujian dari pengolahan data barang pada tabel 4.24 berikut ini :

Tabel IV.24 Pengujian Pengolahan Data Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Barang	Menampilkan data Barang	Dapat menampilkan data Barang	[√] Diterima [ ] Ditolak
Ubah	Menampilkan *form* perbaharui Barang	Dapat menampilkan *form* perbaharui Barang	[√] Diterima [ ] Ditolak
Simpan	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[√] Diterima [ ] Ditolak
Batal	Kembali kemenu 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
Data produk kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "kategori barang dibutuhkan,harga dibutuhkan"	[√] Diterima [ ] Ditolak

I. Pengujian Tambah Barang

Berikut dapat dilihat pengujian dari pengujian tambah Barang pada tabel 4.25 berikut ini :

Tabel IV.25 Pengujian Tambah Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tambah barang	Menampilkan *form* tambah barang	Dapat menampilkan *form* tambah barang	[√] Diterima [ ] Ditolak
Mengisi data barang	Jika semua field	Dapat memunculkan	[√] Diterima

Contoh: kategori: Handuk Jenis : Letter Nama Barang : Handuk Letter Warna : Hijau Ukuran Letter : 140x70 Harga : 70000 Diskon : 0 % Berat : 0.3 kg Keterangan : Menyerap Stok : 1 Upload Gambar : 	terisi semua maka akan muncul pesan " data berhasil ditambah"	pesan data berhasil ditambah	[ ] Ditolak
Simpan	Data barang akan tersimpan didalam *database*	Data barang akan tersimpan didalam *database* dan muncul pesan "data berhasil di simpan"	[√] Diterima [ ] Ditolak
Batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data salah)			
Data Masukkan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data barang masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ada kesalahan didalam pengisian data, silahkan cek kembali.	[√] Diterima [ ] Ditolak

J. Pengujian Pengolahan Data Provinsi

Berikut dapat dilihat pengujian dari pengolahan data provinsi pada tabel 4.26 dibawah ini :

Tabel IV.26 Pengujian Pengolahan Data Provinsi

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

Provinsi	Menampilkan data provinsi	Dapat menampilkan data provinsi	[√] Diterima [ ] Ditolak
Perbaharui	Menampilkan *form* perbaharui provinsi	Dapat menampilkan *form* perbaharui provinsi	[√] Diterima [ ] Ditolak
Simpan	Data berhasil diubah di*database*	Tampil pesan "Data berhasil disimpan"	[√] Diterima [ ] Ditolak
Batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

K. Pengujian Tambah Provinsi

Berikut dapat dilihat dari pengujian tambah provinsi pada tabel 4.27 dibawah ini :

Tabel IV.27 Pengujian Tambah Provinsi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tambah provinsi	Menampilkan *form* tambah provinsi	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data provinsi Contoh: Nama Provinsi : Jawa Barat	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data provinsi masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "Nama provinsi dibutuhkan"	[√] Diterima [ ] Ditolak
Simpan	Data pengiriman akan	Data pengiriman	[√] Diterima

	tersimpan didalam *database*	akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[ ] Ditolak
Batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

L. Pengujian Pengolahan Data Kota

Berikut dapat dilihat pengujian dari pengolahan data kota pada tabel 4.28 dibawah ini :

Tabel IV.28 Pengujian Pengolahan Data Kota

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kota	Menampilkan data kota	Dapat menampilkan data kota	[√] Diterima [ ] Ditolak
Ubah	Menampilkan *form* Mangubah kota	Dapat menampilkan *form* perbaharui kota	[√] Diterima [ ] Ditolak
Simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Batal	Kembali kemenu 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
Data kota kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak

M. Pengujian Tambah Kota

Berikut dapat dilihat pengujian dari tambah kota pada tabel 4.29 dibawah ini :

Tabel IV.29 Pengujian Tambah Kota

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

N. Pengujian Pengolahan Data Pesanan

Berikut dapat dilihat pengujian dari pengolahan data pesanan pada tabel 4.30 dibawah ini :

Tabel IV.30 Pengujian Pengolahan Data Pesanan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Pengeolahan data pemesanan	Menampilkan *form* transaksi	Dapat menampilkan *form* lihat transaksi	[√] Diterima [ ] Ditolak

	pemesanan	pemesanan	
Proses pada tabel transaksi pemesanan belum dibayar	Menampilkan *form* transaksi pemesanan belum dibayar	Dapat menampilkan *form* transaksi pemesanan belum dibayar	[√] Diterima [ ] Ditolak
Proses pada tabel transaksi pemesanan butuh konfirmasi	Menampilkan *form* transaksi pemesanan butuh konfirmasi	Dapat menampilkan *form* transaksi pemesanan butuh konfirmasi	[√] Diterima [ ] Ditolak
Proses pada tabel transaksi pemesanan sudah dibayar	Menampilkan *form* transaksi pemesanan sudah dibayar	Dapat menampilkan *form* transaksi pemesanan sudah dibayar	[√] Diterima [ ] Ditolak
Proses pada tab pengiriman	Menampilkan *form* proses pengiriman	Dapat menampilkan *form* proses pengiriman	[√] Diterima [ ] Ditolak
Tab pesanan terkirim	Menampilkan *form* pesanan terkirim	Dapat menampilkan *form* pesanan terkirim	[√] Diterima [ ] Ditolak
Proses pada detail pada tabel butuh konfirmasi	Ubah status transaksi pada tabel butuh konfirmasi di*database* menjadi sudah dibayar.	Data status transaksi telah diubah di*database* menjadi sudah dibayar	[√] Diterima [ ] Ditolak
Proses pada tab sudah dibayar	Ubah status transaksi pada tabel sudah dibayar di*database* menjadi proses pengiriman	Data status transaksi telah diubah di*database* menjadi proses pengiriman	[√] Diterima [ ] Ditolak
Proses pada tab proses pengiriman	Ubah status transaksi proses pengiriman di*database* menjadi pesanan terkirim	Data status transaksi telah diubah di*database* menjadi pesanan terkirim	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharpakan**	**Pengamatan**	**Kesimpulan**
Proses pada tabel butuh konfirmasi	Menampilkan *form* transaksi pembayaran tidak valid	Dapat menampilkan *form* transaksi pembayaran tidak valid	[√] Diterima [ ] Ditolak

O. Pengujian Pengolahan Komentar

Berikut dapat dilihat pengujian dari pengolahan komentar pada tabel 4.31 dibahwah ini :

Tabel IV.31 Pengujian Pengolahan Komentar

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Komentar	Menampilkan data komentar	Dapat menampilkan data komentar	[√] Diterima [ ] Ditolak
Hapus	Data berhasil dihapus di*database*	Tampil pesan "Yakin akan menghapus daftar komentar berikut."	[√] Diterima [ ] Ditolak
Batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.2.3 Kesimpulan Hasil Pengujian Alpa

Berdasarkan hasil pengujian sistem yang telah dilakuan secara keseluruhan memberikan kesimpulan bahwa pada proses 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.

4.2.4 Pengujian Betha

Pengujian betha merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan membuat kuisoner mengenai kepuasan pelanggan dengan kandungan poin syarat *user friendly* untuk selanjutnya dibagikan kepada pelanggan dengan mengambil *sample* sebanyak 10 orang. Dari hasil kuesioner tersebut akan dilakukan

perhitungan untuk dapat diambil kesimpulan terhadap penelitian 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 CV. SENTOSO

   Penelitian dilingkungan penjual di CV. SENTOSO dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan diawal.

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*.

A. Wawancara Petugas

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

1. Apakah dengan adanya *website e-commerce* ini dapat membantu dalam pemasaran dan memperkenalkan CV. SENTOSO kepada masyarakat luas ?

   Jawaban : website membantu pemasaran ke masyarakat.

Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website yang dibangun dapat membantu dalam hal pemasaran produk kepada masyarakat.

2. Apakah website dapat membantu untuk memudahkan penentuan harga dan tanggal penyelesaian pesanan pada CV. SENTOSO ?

   Jawaban : Harga lebih mudah ditentukan dan tanggal penyelesaian dapat lebih mudah untuk dikelola.

   Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website dapat membantu dalam proses estimasi harga dan tanggal penyelesaian pesanan.

3. Apakah website dapat memudahkan dalam mengelola data pesanan CV. SENTOSO?

   Jawaban : Data pesanan lebih mudah untuk dikelola dan proses produksi yang dilakukan lebih mudah untuk dikontrol.

   Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website dapat mempermudah dalam mengelola data pesanan.

4. Apakah website dapat memudahkan dalam membuat laporan penjualan produk?

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

   Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website memudahkan dalam pembuatan laporan.

B. Kuesioner Pengguna

Kuesioner disebarkan menggunakan teknik sampling yaitu *Sample Random Sampling* yang disebarkan kepada 10 orang pengguna. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner ini terdiri dari 7 pertanyaan (contoh kuesioner dapat dilihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 10, dengan ketentuan skala untuk tiap pertanyaan berikut :

Adapun kuisioner pengujian betha untuk pengguna yaitu :

1. Sistem transaksi *online* yang dirancang ini dapat memberikan kemudahan dalam proses pembelian produk.
   a. Sangat Setuju
   b. Setuju
   c. Biasa Saja
   d. Tidak Setuju
   e. Sangat Tidak Setuju
2. Sistem transaksi *online* yang dirancang ini dapat menambah media transaksi penjualan bagi CV. SENTOSO serta pelanggan yang ada di CV. SENTOSO.
   a. Sangat Setuju
   b. Setuju
   c. Biasa Saja
   d. Tidak Setuju
   e. Sangat Tidak Setuju
3. Apakah keamanan transaksi dalam Sistem transaksi *online* ini dapat memberikan kepercayaan bagi pelanggan di CV. SENTOSO.
   a. Sangat Setuju
   b. Setuju
   c. Biasa Saja
   d. Tidak Setuju
   e. Sangat Tidak Setuju
4. Dengan adanya Sistem transaksi *online* ini, dapat dijadikan sebuah sarana penghubung yang efektif kepada CV. SENTOSO
   a. Sangat Setuju
   b. Setuju
   d. Tidak Setuju
   e. Sangat Tidak Setuju

c. Biasa Saja

5. Apakah tampilan dari Sistem transaksi *online* ini menarik
   a. Sangat Setuju
   b. Setuju
   c. Biasa Saja
   d. Tidak Setuju
   e. Sangat Tidak Setuju
6. Sistem transaksi *online* yang dirancang ini dapat memberikan keefisienan waktu, dan tempat bagi pelanggan, dalam hal penerimaan info promosi produk baru, serta penerimaan info penjualan
   a. Sangat Setuju
   b. Setuju
   c. Biasa Saja
   d. Tidak Setuju
   e. Sangat Tidak Setuju
7. Sistem transaksi *online* yang dirancang ini sudah *user friendly*
   a. Sangat Setuju
   b. Setuju
   c. Biasa Saja
   d. Tidak Setuju
   e. Sangat Tidak Setuju

Bedasarkan data hasil kuesioner 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. Kuesioner ini diujikan kepada 10 orang.

1. Sistem transaksi *online* yang dirancang ini dapat memberikan kemudahan dalam proses pembelian produk.

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	4	40
	2	Setuju	4	40
	3	Biasa Saja	1	10
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	1	10
Jumlah			**10**	**100**

Bedasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 4 orang atau 40% mengatakan sangat setuju, 4 orang atau 40% mengatakan setuju, 1 orang atau 10% mengatakan biasa saja dan 1 orang atau 10% mengatakan sangat tidak setuju bahwa perangkat lunak yang dibuat memberikan kemudahan dalam proses pembelian produk.

2. Sistem transaksi *online* yang dirancang ini dapat menambah media transaksi penjualan bagi CV. SENTOSO serta pelanggan yang ada di CV. SENTOSO.

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	4	40
	2	Setuju	4	40
	3	Biasa Saja	1	10
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	1	10
Jumlah			**10**	**100**

Bedasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 4 orang atau 40% mengatakan sangat setuju, 4 orang atau 40% mengatakan setuju, 1 orang atau 10% mengatakan biasa saja dan 1 orang atau 10% mengatakan sangat tidak setuju bahwa perangkat lunak yang dibuat menambah media transaksi penjualan dan pelanggan.

3. Apakah keamanan transaksi dalam Sistem transaksi *online* ini dapat memberikan kepercayaan bagi pelanggan di CV. SENTOSO.

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	4	40
	2	Setuju	4	40
	3	Biasa Saja	2	20
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Bedasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 4 orang atau 40% mengatakan sangat setuju, 4 orang atau 40% mengatakan setuju dan 2 orang atau 20% mengatakan sangat tidak setuju bahwa perangkat lunak yang dibuat memiliki keamanan transaksi.

4. Dengan adanya Sistem transaksi *online* ini, dapat dijadikan sebuah sarana penghubung yang efektif kepada CV. SENTOSO

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
4	1	Sangat Setuju	4	40
	2	Setuju	4	40
	3	Biasa Saja	1	10
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	1	10
Jumlah			**10**	**100**

Bedasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 4 orang atau 40% mengatakan sangat setuju, 4 orang atau 40% mengatakan setuju, 1 orang atau 10% mengatakan biasa saja dan 1 orang atau 10% mengatakan sangat tidak setuju bahwa perangkat lunak yang dibuat menjadi sarana penghubung yang efektif.

5. Apakah tampilan dari Sistem transaksi *online* ini menarik

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
5	1	Sangat Setuju	0	0
	2	Setuju	6	60
	3	Biasa Saja	3	30
	4	Tidak Setuju	1	10
	5	Sangat Tidak Setuju	0	10
Jumlah			**10**	**100**

Bedasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 6 orang atau 60% mengatakan setuju, 3 orang atau 30% mengatakan biasa saja dan 1 orang atau 10% mengatakan tidak setuju bahwa tampilan perangkat lunak yang dibuat menarik.

6. Sistem transaksi *online* yang dirancang ini dapat memberikan keefisienan waktu, dan tempat bagi pelanggan, dalam hal penerimaan info promosi produk baru, serta penerimaan info penjualan.

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
6	1	Sangat Setuju	4	40
	2	Setuju	4	40
	3	Biasa Saja	2	20
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Bedasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 4 orang atau 40% mengatakan sangat setuju, 4 orang atau 40% mengatakan setuju dan 2 orang atau 20% mengatakan biasa saja bahwa perangkat lunak yang dibuat memberikan keefisienan waktu, dan tempat bagi pelanggan, dalam hal penerimaan info promosi produk baru, serta penerimaan info penjualan.

7. Sistem transaksi *online* yang dirancang ini sudah *user friendly*

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)

7	1	Sangat Setuju	2	20
	2	Setuju	7	70
	3	Biasa Saja	1	10
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Bedasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 20% mengatakan sangat setuju, 7 orang atau 70% mengatakan setuju dan 1 orang atau 10% mengatakan biasa saja bahwa perangkat lunak yang dibuat sudah *user friendly*.

4.2.5 kesimpulan Hasil Pengujian Betha

Dari pengujian beta yang telah dilakukan yaitu dengan pengujian perhitungan pilihan kategori jawaban dari hasil wawancara petugas dan kuesioner pengguna yang telah dibagikan di lapangan didapat kesimpulan bahwa aplikasi yang dirancang dinilai dapat membantu dalam pengolahan data sesuai kebutuhan, menjadi suatu sarana yang bermanfaat, keefisienan waktu di dapat, info yang di sampaikan cepat, mudah dan akurat.