BAB IV

IMPLEMENTASI DAN PENGUJIAN

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

4.1 Implementasi

Tahap implementasi sistem merupakan tahap menterjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dapat dimengerti oleh mesin serta penerapan perangkat lunak pada keadaan yang sesungguhnya.

4.1.1 Hardware

Perangkat keras (*Hardware*) yang digunakan untuk mengembangkan Sistem Penerapan *E-commerce* untuk Sistem Penjualan adalah sebagai berikut :

1. *Personal Computer* dengan *Prosessor* minimal Pentium 4.
2. *RAM* 512 Mb
3. *Harddisk* 80 Gb

4.1.2 Software

Perangkat lunak (*Software*) yang dipasang pada sistem komputer yang digunakan untuk mengembangkan Sistem Penerapan *E-commerce* untuk Sistem Penjualan ini adalah sebagai berikut :

1. Sistem Operasi Windows XP/ Vista.
2. *MySQL* untuk penyimpanan *database.*
3. *PhpMyAdmin* untuk *interface database*
4. *Macromedia dreamweaver 8* untuk penulisan kode dan perancangan tampilan program.

4.2 Implementasi Basis Data

Pada tahap awal pembangunan sebuah aplikasi, dibutuhkan media penyimpanan data (*database*), implementasi database untuk Sistem Penerapan *E-commerce* untuk Sistem Penjualan adalah sebagai berikut :

1. Tabel Admin

```
CREATE TABLE IF NOT EXISTS `admin` (
 `id_admin` int(3) NOT NULL AUTO_INCREMENT,
 `nama` varchar(30) NOT NULL,
 `alamat` varchar(30) NOT NULL,
 `email` varchar(50) NOT NULL,
 `password` varchar(45) NOT NULL,
 PRIMARY KEY (`id_admin`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin;
```

2. Tabel Member

```
CREATE TABLE IF NOT EXISTS `member` (
```

```
`id_member` int(3) NOT NULL AUTO_INCREMENT,
`nama` varchar(30) NOT NULL,
`alamat` varchar(100) NOT NULL,
`rt` int(3) NOT NULL,
`rw` int(3) NOT NULL,
`kecamatan` varchar(50) NOT NULL,
`kelurahan` varchar(50) NOT NULL,
`email` varchar(30) NOT NULL,
`password` varchar(45) NOT NULL,
`no_telp` varchar(14) NOT NULL,
`kodepos` varchar(5) NOT NULL,
`id_kota` int(11) NOT NULL,
PRIMARY KEY (`id_member`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

3. Tabel Provinsi

```
CREATE TABLE IF NOT EXISTS `provinsi` (
  `id_provinsi` int(3) NOT NULL AUTO_INCREMENT,
  `nama_provinsi` varchar(30) DEFAULT NULL,
  KEY `id_prov` (`id_provinsi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin
  AUTO_INCREMENT=34 ;
```

4. Tabel Warna

```
CREATE TABLE IF NOT EXISTS `warna` (
  `id_warna` int(2) NOT NULL AUTO_INCREMENT,
```

```
  `nama_warna` varchar(15) NOT NULL,
  PRIMARY KEY (`id_warna`)
) ENGINE=InnoDB;
```

5. Tabel Ukuran

```
CREATE TABLE IF NOT EXISTS `ukuran` (
  `id_ukuran` int(2) NOT NULL AUTO_INCREMENT,
  `nama_ukuran` varchar(2) NOT NULL,
  `ket` varchar(6) NOT NULL,
  PRIMARY KEY (`id_ukuran`)
) ENGINE=InnoDB;
```

6. Tabel Kategori

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` int(2) NOT NULL AUTO_INCREMENT,
  `nama_kategori` varchar(30) NOT NULL,
  PRIMARY KEY (`id_kategori`)
) ENGINE=InnoDB;
```

7. Tabel Ongkos_Kirim

```
CREATE TABLE IF NOT EXISTS `ongkos_kirim` (
  `id_ongkos_kirim` int(3) NOT NULL AUTO_INCREMENT,
  `jenis_pengiriman` enum('reguler','ons','oke') NOT NULL,
  `biaya_kirim` double NOT NULL,
  `id_kota` int(3) NOT NULL,
  PRIMARY KEY (`id_ongkos_kirim`),
  FOREIGN KEY (id_kota) REFERENCES kota (id_kota)
) ENGINE=InnoDB ;
```

8. Tabel Kota

```
CREATE TABLE IF NOT EXISTS `kota` (
  `id_kota` int(3) NOT NULL AUTO_INCREMENT,
  `id_provinsi` int(3) NOT NULL,
  `nama_kota` varchar(50) NOT NULL,
  `ongkos_kirim` double NOT NULL,
  UNIQUE KEY `kota#PX` (`id_kota`),
  FOREIGN KEY `id_provinsi` REFERENCES provinsi (`id_provinsi`)
 ) ENGINE=InnoDB ;
```

9. Tabel Kelurahan

```
CREATE TABLE IF NOT EXISTS `kelurahan` (
  `id_kelurahan` int(3) NOT NULL AUTO_INCREMENT,
  `nama_kelurahan` varchar(50) DEFAULT NULL,
  `id_kota` int(3) NOT NULL,
  PRIMARY KEY (`id_kelurahan`),
  KEY `fk_1` (`id_kota`)
) ENGINE= InnoDB ;
```

10. Tabel Kecamatan

```
CREATE TABLE IF NOT EXISTS `kecamatan` (
  `id_kecamatan` int(3) NOT NULL AUTO_INCREMENT,
  `nama_kecamatan` varchar(50) DEFAULT NULL,
  `id_kelurahan` int(3) DEFAULT NULL,
  PRIMARY KEY (`id_kecamatan`),
  KEY `fk_1` (`id_kelurahan`)
```

```
) ENGINE=InnoDB;
```

11. Tabel Barang

```
CREATE TABLE IF NOT EXISTS `barang` (
 `id_barang` int(4) NOT NULL AUTO_INCREMENT,
 `id_kategori` int(2) NOT NULL,
 `nama_barang` varchar(30) NOT NULL,
 `harga_jual` double NOT NULL,
 `harga_beli` double NOT NULL,
 `stok` int(3) NOT NULL,
 `tanggal_beli` date NOT NULL,
 `gambar` varchar(100) NOT NULL,
 `deskripsi` text NOT NULL,
 PRIMARY KEY (`id_barang`),
FOREIGN KEY (id_kategori) REFERENCES kategori (id_kategori)
) ENGINE=InnoDB ;
```

12. Barang Detail

```
CREATE TABLE IF NOT EXISTS `barang_detail` (
 `id_barang_detail` int(4) NOT NULL AUTO_INCREMENT,
 `id_barang` int(4) NOT NULL,
 `id_ukuran` int(2) NOT NULL,
 `id_warna` int(2) NOT NULL,
 `qty` int(3) NOT NULL,
  PRIMARY KEY (`id_barang_detail`),
  foreign key (id_barang) references barang (id_barang),
  foreign key (id_warna) references warna (id_warna),
```

```
foreign key (id_ukuran) references ukuran (id_ukuran)
) ENGINE=InnoDB ;
```

13. Tabel Pembelian_Online

```
CREATE TABLE IF NOT EXISTS `pembelian_online` (
 `id_pembelian_online` int(3) NOT NULL AUTO_INCREMENT,
 `id_barang_detail` int(3) NOT NULL,
 `id_member` int(3) NOT NULL,
 `jumlah` int(11) NOT NULL,
 `session_id` varchar(32) NOT NULL,
 `tanggal_beli` date NOT NULL,
`no_pengiriman` varchar(32),
 `jenis_pembayaran` enum('transfer','paypal') NOT NULL,
`status_pembelian` enum('belum bayar','bayar', 'kirim', 'batal') NOT NULL,
 PRIMARY KEY (`id_pembelian_online`),
foreign key (id_barang_detail) references barang_detail (id_barang_detail),
foreign key (id_member) references member (id_member),
) ENGINE=InnoDB ;
```

14. Tabel Konfirmasi Pembayaran

```
CREATE TABLE IF NOT EXISTS `konfirmasi_pembayaran` (
 `id_konfirmasi` int(11) NOT NULL AUTO_INCREMENT,
 `nama_pengirim` varchar(30) NOT NULL,
 `nama_bank` varchar(20) NOT NULL,
 `no_rek` varchar(15) NOT NULL,
 `tanggal_transfer` date NOT NULL,
 `id_pembelian_online` int(11) NOT NULL,
```

```
 PRIMARY KEY (`id_konfirmasi`),
foreign key (id_pembelian_online references pembelian_online
  (id_pembelian_online),
) ENGINE=InnoDB;
```

15. Tabel Pembelian_Offline

```
CREATE TABLE IF NOT EXISTS `pembelian_offline` (
  `id_pembelian_offline` int(11) NOT NULL AUTO_INCREMENT,
  `id_admin` int(3) NOT NULL,
  `id_barang_detail` int(3) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `tanggal_beli` date NOT NULL,
  PRIMARY KEY (`id_pembelian_offline`),
foreign key (id_barang_detail) references barang_detail (id_barang_detail),
foreign key (id_admin) references admin (id_admin),
foreign key (id_ukuran) references ukuran (id_ukuran)
) ENGINE=InnoDB;
```

16. Tabel Detail_Pembelian_Online

```
CREATE TABLE IF NOT EXISTS `detail_pembelian_online` (
  `id_detail_pembelian_online` int(3) NOT NULL  AUTO_INCREMENT,
  `id_pembelian_online` int(3) NOT NULL,
  `tanggal_transaksi` date NOT NULL,
  `jumlah` int(11) NOT NULL,
  `total` double NOT NULL,
  PRIMARY KEY (`id_detail_pembelian_online`),
```

```
foreign key (id_pembelian_online) references pembelian_online
(id_pembelian_online)
) ENGINE=InnoDB;
```

17. Tabel Detail Pembelian Offline

```
CREATE TABLE IF NOT EXISTS `detail_pembelian_offline` (
 `id_detail_pembelian_offline` int(11) NOT NULL AUTO_INCREMENT,
  `tanggal_transaksi_offline` datetime NOT NULL,
 `jumlah` int(11) NOT NULL,
  `total_harga` double NOT NULL,
  PRIMARY KEY (`id_detail_pembelian_offline`)
) ENGINE=InnoDB;
```

4.3 Imlementasi Antarmuka

Implementasi antar muka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna. Adapun untuk tampilan implementasi akan dilampirkan pada halaman lampiran.

Tabel 4. 1 Implementasi antarmuka admin

Menu	Deskripsi	Nama File
Login Admin	Login sebagai admin	login.php
Lupa Password	Digunakan jika admin lupa terhadap passwordnya.	account.php
Account Setting: - Ubah *Profile* - Ganti *Password*	 Digunakan jika administrator ingin merubah *profile.* Digunakan jika administrator ingin merubah *password*	account.php account.php
Pengolahan Data Master:		

- Pengolahan Data Kategori	Digunakan untuk melihat, menambah, merubah dan menghapus data kategori.	kategori.php
- Pengolahan Data Barang	Digunakan untuk melihat, menambah, merubah dan menghapus data barang.	barang.php
- Pengolahan Data Ukuran	Digunakan untuk melihat, menambah, merubah dan menghapus data ukuran.	ukuran.php
- Pengolahan Data Warna	Digunakan untuk melihat, menambah, merubah dan menghapus data warna.	warna.php
- Pengolahan Data Barang Detail	Digunakan untuk melihat, menambah, merubah dan menghapus data barang detail.	detailbarang.php
- Pengolahan Data Provinsi	Digunakan untuk melihat, menambah, merubah dan menghapus data provinsi.	provinsi.php
- Pengolahan Data Kota	Digunakan untuk melihat, menambah, merubah dan menghapus data kota.	kota.php
- Pengolahan Data Kelurahan	Digunakan untuk melihat, menambah, merubah dan menghapus data kelurahan.	kelurahan.php
- Pengolahan Data Kecamatan	Digunakan untuk melihat, menambah, merubah dan menghapus data kecamatan.	kecamatan.php
- Pengolahan Data Ongkos Kirim	Digunakan untuk melihat, menambah, merubah dan menghapus data ongkos kirim.	ongkir.php
- Pengolahan Data Member	Digunakan untuk melihat, dan menghapus data member.	member.php
Pengolahan Data Transaksi	Digunakan untuk melihat data transaksi online yang masuk dan untuk merubah status	online.php
Pengolahan Laporan:		
- Laporan penjualan online harian	Digunakan untuk melihat laporan penjualan online perhari.	laporanonlineharian.php
- Laporan penjualan online bulanan	Digunakan untuk melihat laporan penjualan online perbulan.	laporanonlinebulanan.php

- Laporan penjualan online tahunan	Digunakan untuk melihat laporan penjualan online pertahun.	laporanonlinetahunan.php
Logout	Digunakan admin untuk keluar dari menu admin	logout.php

Tabel 4. 2 Implementasi antarmuka pengunjung dan *member*

Menu	Deskripsi	Nama File
Registrasi *member*	Digunakan untuk melakukan pendaftaran *member* baru.	signup.php
Login *member*	Digunakan sebagai halaman login *member*	signin.php
Lupa *password*	Digunakan jika *member* lupa dengan *password*	setting.php
Account Setting : - Edit Profile - Ganti *password*	 Digunakan untuk mengubah profil *member.* Digunakan jika *member* ingin merubah *password*	 account.php account.php
Kategori	Digunakan untuk melihat barang perkategori	kategori.php
Barang: Barang terbaru Katalog	 Digunakan untuk melihat seluruh barang terbaru tiap kategori Digunakan untuk melihat seluruh barang tiap kategori	 barang.php barang.php
Help : Cara pembelian Cara pembayaran	 Digunakan untuk melihat Cara melakukan pembelian Digunakan untuk melihat cara pembayaran	 carapembelian.php carapembayaran.php
Konfirmasi Pembayaran	Digunakan untuk melakukan konfirmasi pembayaran	cart.php
Keranjang belanja	Digunakan untuk melihat, merubah, menghapus detail transaksi yang akan dilakukan.	cart.php
Logout	Digunakan *member* untuk keluar dari akunnya.	logout.php

4.4 Pengujian

Pengujian yang digunakan untuk menguji sistem yang baru adalah metode pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

4.4.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara *alpha* dan *beta*. Pengujian situs e-commerce pada butik Yello berikut menggunakan data uji berdasarkan data dari butik Yello. Rencana pengujian selengkapnya terlihat pada tabel dibawah ini.

Tabel 4. 3 Rencana Pengujian Aplikasi Admin

Item Uji	Detail Pengujian	Jenis Pengujian
Login	Input Login Admin	*Black Box*
	Verifikasi Login Admin	*Black Box*
Pengolahan data kategori	Tambah Kategori	*Black Box*
	Ubah Kategori	*Black Box*
	Hapus Kategori	*Black Box*
	Pencarian	*Black Box*
Pengolahan data barang	Tambah Barang	*Black Box*
	Ubah Barang	*Black Box*
	Hapus Barang	*Black Box*
	Pencarian	*Black Box*
Pengolahan data ukuran	Tambah ukuran	*Black Box*
	Ubah ukuran	*Black Box*
	Hapus ukuran	*Black Box*
	Pencarian	*Black Box*
Pengolahan data warna	Tambah warna	*Black Box*
	Ubah warna	*Black Box*
	Hapus warna	*Black Box*
	Pencarian	*Black Box*
Pengolahan data barang detail	Tambah barang detail	*Black Box*
	Ubah barang detail	*Black Box*
	Hapus barang detail	*Black Box*
	Pencarian	*Black Box*
Pengolahan data provinsi	Tambah barang provinsi	*Black Box*
	Ubah barang provinsi	*Black Box*

	Hapus barang provinsi	*Black Box*
	Pencarian	*Black Box*
Pengolahan data kota	Tambah barang kota	*Black Box*
	Ubah barang kota	*Black Box*
	Hapus barang kota	*Black Box*
	Pencarian	*Black Box*

Tabel 4. 4 Skenario pengujian aplikasi *member*

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Registarsi	Input Registrasi	*Black Box*
	Simpan Data	*Black Box*
	Verifikasi Email	*Black Box*
Login	Input Login	*Black Box*
	Verifikasi Login	*Black Box*
Logout	Menghapus *session user*	*Black Box*
Edit Profile	Ubah profile	*Black Box*
Ganti *password*	Menampilkan *form* ubah *password*	*Black Box*
	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verifikasi konfirmasi *password*	*Black Box*
Kategori	Menampilkan data kategori	*Black Box*
	Menampilkan data detail kategori	*Black Box*
Produk	Menampilkan data barang	*Black Box*
	Menampilkan data detail barang	*Black Box*
Menu keranjang belanja	Menampilkan data detail transaksi	*Black Box*
	Mengubah data detail transaksi	*Black Box*
	Menghapus data detail transaksi	*Black Box*
Menu Help	Menampilkan cara pemesanan dan cara pembelian	*Black Box*
Form Konfirmasi Pembayaran	Menampilkan form Konfirmasi Pembayaran	*Black Box*
	Isi form pengiriman	*Black Box*

4.4.2 Pengujian Alpha

Hasil implementasi perangkat lunak sistem informasi ini diuji pada pemilik toko maupun berbasis *online* melalui spesifikasi perangkat sistem agar sistem dapat berjalan dengan baik.

Pengujian perangkat lunak sistem penerapan *e-commerce* menggunakan data uji berdasarkan form yang terdapat didalam aplikasi.

1. Pengujian Registrasi member

Berikut dapat dilihat pengujian dari registrasi *member* pada tabel 4.7 dibawah ini:

Tabel 4. 5 Pengujian Registrasi *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu registrasi *member*	Tampil form registrasi *member*	Dapat menampilkan form registrasi *member*	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh masukan : Nama : Ela Alamat Email : haruhi.neesan@gmail.com Provinsi : Jawa Barat Kota : Bandung Kelurahan: Citaman Kecamatan:Nagreg Kodepos : 40514 Telepon : 081321669663 Password : 123456 Konfirmasi password : 123456	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak

Klik tombol simpan	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi. Jika data masukan benar muncul pesan "Anda telah berhasil melakukan registrasi"	Tampil pesan "Anda telah berhasil melakukan registrasi" jika data masukan benar dan dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Memasukan data kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

2. Pengujian Login *Member*

Berikut dapat dilihat pengujian login *member* pada tabel 4.8 dibawah ini:

Tabel 4. 6 Pengujian login *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Alamat *email* : haruhi.neesan@gmail.com	Mengisikan data alamat *email* pada *field* alamat *email*. Jika salah maka akan muncul pesan "invalid alamat email"	Dapat mengisi *field* alamat *email* dan muncul pesan "invalid alamat email" jika salah dalam pengisian data	[√] Diterima [ ] Ditolak
Password : 123456	Mengisikan	Dapat mengisi	[√]

	data *password* pada *field password*. Jika salah maka akan muncul pesan "diantara 6 sampai 11 karakter diperbolehkan"	*field password* dan muncul pesan "diantara 6 sampai 11 karakter diperbolehkan" jika salah dalam pengisian data	Diterima [ ] Ditolak
Klik tombol login	Data *email* dan data *password* dicari	Tombol login dapat berfungsi	[√] Diterima [ ] Ditolak
Data *email* atau *password* tidak terdaftar	Tidak dapat login	Muncul pesan "*email* atau *password* salah"	[√] Diterima [ ] Ditolak
Data *email* atau *password* kosong	Tidak dapat login	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

3. Pengujian Logout

Berikut dapat dilihat pengujian dari logout pada tabel 4.9. dibawah ini:

Tabel 4. 7 Pengujian logout

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

4. Pengujian Edit Profile Member

Berikut dapat dilihat pengujian dari ubah profil member pada tabel 4.10. dibawah ini:

Tabel 4. 8 Pengujian Akun profil member

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Edit Profile	Menampilkan *form* Edit Profile	Dapat menampilkan *form* edit 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
Klik ubah	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[√] Diterima [ ] Ditolak
Data masukan ada yang tidak diisi	Menampilkan pesan kesalahan bahwa field tersebut harus terisi.	Dapat menampilkan pesan kesalahan bahwa field tersebut harus terisi.	[√] Diterima [ ] Ditolak
Data masukan semuanya tidak diisi	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

5. Pengujian ganti *password member*

Berikut dapat dilihat pengujian dari ganti *password member* pada tabel 4.11. dibawah ini:

Tabel 4. 9 Pengujian ganti *password member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Ganti *password*	Menampilkan *form* ganti *password*	Dapat menampilkan *form* ganti	[√] Diterima [ ] Ditolak

		password	
Data masukan semuanya benar	*Password* baru berhasil disimpan di *database*	Dapat menyimpan data *password* yang baru dengan muncul pesan " Data berhasil diubah"	[√] Diterima [ ] Ditolak
Salah satu field belum terisi	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan "*field* ini harus diisi" bila ada field yang belum diisi	[√] Diterima [ ] Ditolak
Data *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 "password baru dan konfirmasi password harus sama"	Dapat menampilkan pesan "password baru dan konfirmasi password harus sama"	[√] Diterima [ ] Ditolak
Data kosong	Muncul pesan kesalahan field harus diisi	Dapat menampilkan pesan kesalahan field harus diisi	√] Diterima [ ] Ditolak

6. Pengujian menu produk

Berikut dapat dilihat pengujian dari menu produk pada tabel 4.12. dibawah ini:

Tabel 4. 10 Pengujian menu produk

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu produk	Menampilkan data produk yang ada di	Dapat menampilkan data produk	[√] Diterima [ ] Ditolak

	database	yang ada di *data base*	
Klik detail pada data produk	Menampilkan detail produk	Dapat menampilkan detail produk	[√] Diterima [ ] Ditolak
Klik Tambah Ke Keranjang	Menuju ke keranjang belanja	Dapat menampilkan keranjang belanja	[√] Diterima [ ] Ditolak

4.4.2.1 Pengujian Login Admin

Pengujian login terbagi diantaranya verifikasi nama user dan verifikasi password sebagai berikut :

1. **Input Login**

Berikut ini adalah tabel pengujian tampilan form login untuk input login :

Tabel 4. 11 Pengujian Login

Kasus dan Hasil Uji (data normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
email. password	Dapat terisi pada textbox(email). Dapat terisi pada textbox(password).	Dapat mengisi nama email sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Tombol Login.	Pengecekan apakah data valid atau invalid	Tombol login berfungsi sesuai dengan yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (data salah)			
Nama email dan password salah	Tidak dapat login dan menampilkan menu utama admin.	admin tidak dapat login dan mendapatkan pesan admin belum benar.	[√] Diterima [ ] Ditolak

2. **Verifikasi Login**

Berikut ini adalah tabel pengujian tampilan form login untuk verifikasi Login

Tabel 4. 12 Pengujian Verifikasi Login

Kasus dan Hasil Uji (data normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
email password	Mengecek email dan password kosong atau tidak.	Dapat diterima sesuai harapan	[√] Diterima [ ] Ditolak
Klik Tombol Login.	Pengecekan dan jika sama dengan data login maka admin masuk ke menu utama admin	Tombol login berfungsi sesuai dengan yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (data salah)			
Data email dan password salah	Tidak dapat login dan menampilkan menu utama admin.	admin tidak dapat login dan mendapatkan pesan kesalahan	[√] Diterima [ ] Ditolak

4.4.2.2 Pengujian Menu Utama di Halaman Admin

1. **Pengujian ganti *password* admin**

Berikut dapat dilihat pengujian dari ganti *password* admin pada tabel 4.13 dibawah ini:

Tabel 4. 13 Pengujian ganti *password* admin

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

		diisi" bila ada field yang belum diisi	
Data *password* lama salah	muncul pesan "password salah"	Dapat menampilkan pesan kesalahan "*Password* salah"	[√] Diterima [ ] Ditolak
Data *password* baru dah konfirmasi *password* baru tidak sama	Muncul pesan "password baru dan konfirmasi password harus sama"	Dapat menampilkan pesan "password baru dan konfirmasi password harus sama"	[√] Diterima [ ] Ditolak
Data kosong	Muncul pesan kesalahan field harus diisi	Dapat menampilkan pesan kesalahan field harus diisi	√] Diterima [ ] Ditolak

2. Pengujian Pengolahan Data provinsi

Berikut dapat dilihat pengujian dari data provinsi pada tabel 4.14 dibawah ini:

Tabel 4. 14 Pengujian Pengolahan Data Provinsi

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

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

3. Pengujian tambah provinsi

Berikut dapat dilihat pengujian dari tambah provinsi pada tabel 4.15 dibawah ini:

Tabel 4. 15 Pengujian tambah provinsi

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

		”ada kesalahan didalam pengisian data, silahkan cek kembali”	
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4. Pengujian Pengolahan Data Kota

Berikut dapat dilihat pengujian dari data kota pada tabel 4.16 dibawah ini:

Tabel 4. 16 Pengujian Data kota

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik kota	Menampilkan data kota	Dapat menampilkan data kota	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah kota	Dapat menampilkan *form* ubah kota	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data kota kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam	[√] Diterima [ ] Ditolak

		pengisian data, silahkan cek kembali"	
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan "Data berhasil diubah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

5. Pengujian tambah kota

Berikut dapat dilihat pengujian dari tambah kota pada tabel 4.17 dibawah ini:

Tabel 4. 17 Pengujian tambah kota

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

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

6. Pengujian Pengolahan Data ukuran

Berikut dapat dilihat pengujian dari data ukuran pada tabel 4.18 dibawah ini:

Tabel 4. 18 Pengujian Pengolahan Data ukuran

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik ukuran	Menampilkan data ukuran	Dapat menampilkan data ukuran	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah ukuran	Dapat menampilkan *form* ubah ukuran	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data ukuran kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil	Tampil pesan	[√] Diterima

	diubah di*database*	"Data berhasil diubah"	[ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

7. Pengujian tambah ukuran

Berikut dapat dilihat pengujian dari tambah ukuran pada tabel 4.19 dibawah ini:

Tabel 4. 19 Pengujian tambah ukuran

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah ukuran	Menampilkan *form* tambah ukuran	Dapat menampilkan *form* tambah ukuran	[√] Diterima [ ] Ditolak
Mengisi data ukuran Contoh: Kode : S Ukuran : Small	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data ukuran masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "ada kesalahan didalam pengisian data, silahkan cek kembali"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam	[√] Diterima [ ] Ditolak

		database dan muncul pesan ”data berhasil di tambah”	
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

8. Pengujian Pengolahan Data kategori

Berikut dapat dilihat pengujian dari lihat kategori pada tabel 4.20 dibawah ini:

Tabel 4. 20 Pengujian lihat kategori

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

9. Pengujian tambah kategori

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

Tabel 4. 21 Pengujian tambah kategori

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

		muncul pesan ”data berhasil di tambah”	
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

10. Pengujian Pengolahan Data barang

Berikut dapat dilihat pengujian dari data barang pada tabel 4.22 dibawah ini:

Tabel 4. 22 Pengujian lihat barang

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

11. Pengujian tambah barang

Berikut dapat dilihat pengujian dari tambah barang pada tabel 4.23dibawah ini:

Tabel 4. 23 Pengujian tambah barang

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Tambah Barang	Menampilkan *form* tambah barang	Dapat menampilkan *form* tambah barang	[√] Diterima [ ] Ditolak
Mengisi data barang Contoh: kategori: Dress Nama : Dress 1 Deskripsi : Dress yang cocok untuk berbagai acara Stok : 20 Harga Beli : 80000 Harga Jual : 100000 Berat : 0.3 kg Gambar :	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data barang masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak

Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

12. Pengujian Pengolahan Data member

Berikut dapat dilihat pengujian dari data member pada tabel 4.24 dibawah ini:

Tabel 4. 24 Pengujian lihat member

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

13. Pengujian Pengolahan Data warna

Berikut dapat dilihat pengujian dari lihat warna pada tabel 4.25 dibawah ini:

Tabel 4. 25 Pengujian lihat warna

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik kategori	Menampilkan *form* lihat kategori	Dapat menampilkan *form* lihat kategori	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah kategori	Dapat menampilkan *form* ubah kategori	[√] Diterima [ ] Ditolak

Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data kategori kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

14. Pengujian tambah warna

Berikut dapat dilihat pengujian dari tambah warna pada tabel 4.26 dibawah ini:

Tabel 4. 26 Pengujian tambah kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Tambah Warna	Menampilkan *form* tambah warna	Dapat menampilkan *form* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data kategori Contoh: Nama Warna : Merah	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	[√] Diterima [ ] Ditolak

		pengisian.	
Data warna masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.4.2.3 Kesimpulan Hasil Pengujian Alpha

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

4.4.3 Kasus Dan Hasil Pengujian (Pengujian Beta)

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan membuat kuesioner mengenai kepuasan user dengan kandungan poin syarat user friendly untuk selanjutnya dibagikan kepada sebagian user dengan

mengambil sample sebanyak 5 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

4.4.3.1 Hasil Kuesioner Pengguna

4.4.3.1.1 Kuesioner Pengguna Admin

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

Berdasarkan data hasil kuesioner tersebut , dicari prosentase masing-masing jawaban dengan menggunakan rumus : Y = P/Q * 100%

Keterangan : P = Banyaknya jawaban responden tiap soal

Q = Jumlah Responden

Y = Nilai Prosentase

1. Bagaimana tampilan perangkat lunak untuk pengelolaan data ?

No	Keterangan	Responden	Prosentase
1	Sangat Bagus		
2	Bagus	1	50%
3	Cukup Bagus	1	50%
4	Kurang Bagus		
No	Keterangan	Responden	Prosentase
5	Tidak Bagus		

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

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

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

No	Keterangan	Responden	Prosentase
1	Sangat Mudah Digunakan		
2	Mudah Digunakan	1	50%
3	Cukup Mudah Digunakan	1	50%
4	Cukup Sulit Digunakan		
5	Sulit Digunakan		

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

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

No	Keterangan	Responden	Prosentase
1	Sangat Membantu		
2	Membantu	2	100%
3	Cukup Membantu		
4	Kurang Membantu		
5	Tidak Membantu		

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

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

No	Keterangan	Responden	Prosentase
1	Sangat Membantu		
2	Membantu	1	50%

3	Cukup Membantu	1	50%
4	Kurang Membantu		
5	Tidak Membantu		

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 1 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat membantu dan sebanyak 1 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat cukup membantu. Sehingga secara keseluruhan perangkan lunak tersebut membantu dan cukup membantu.

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

No	Keterangan	Responden	Prosentase
1	Sangat Membantu		
2	Membantu	2	100%
3	Cukup Membantu		
4	Kurang Membantu		
5	Tidak Membantu		

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

4.4.3.1.2 Kuesioner Pengguna

Kuesioner disebarkan menggunakan teknik sampling yaitu *Simple Random Sampling* yang disebarkan kepada 30 pengguna. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner ini terdiri dari 5 pertanyaan (contoh kuesioner dapat diliihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 5, dengan ketentuan skala untuk tiap pertanyaan sebagai berikut :

1. Apakah tampilan *website* cukup menarik dan mudah dipahami *user friendly*?

No	Keterangan	Responden	Prosentase

1	Sangat Setuju		
2	Setuju	24	80%
3	Biasa Saja	6	20%
4	Kurang Setuju		
5	Tidak Setuju		

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 24 orang atau 80% mengatakan bahwa setuju bahwa tampilan website *user friendly,* sementara 6 orang atau 20% mengatakan biasa saja . Sehingga secara keseluruhan website tersebut *user friendly* .

2. Apakah bahasa pada situs ini mudah untuk dimengerti?

No	Keterangan	Responden	Prosentase
1	Sangat mudah	10	33,3%
2	Mudah	15	50%
3	Biasa saja	5	16,7%
4	Cukup Sulit		
5	Sulit		

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 10 orang atau 33,3% mengatakan bahasanya sangat mudah dimengerti, sementara 15 orang atau 50% mengatakan mudah dimengerti, sementara 5 orang atau 16,7% mengatakan biasa saja . Sehingga secara keseluruhan bahasa dalam website tersebut mudah dimengerti.

3. Apakah informasi barang yang diberikan oleh *website* ini sesuai dengan informasi yang anda butuhkan?

No	Keterangan	Responden	Prosentase
1	Sangat Sesuai	18	60%
2	Sesuai	9	30%
3	Biasa Saja	3	10%
4	Kurang Sesuai		
5	Tidak Sesuai		

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 18 orang atau 60% mengatakan sangat setuju bahwa website ini

memberikan informasi barang yang dibutuhkan,sementara 9 orang atau 30% setuju bahwa website ini memberikan informasi barang yang dibutuhkan, sementara 3 orang atau 10% mengatakan biasa saja . Sehingga secara keseluruhan website ini memberikan informasi barang yang dibutuhkan.

4. Apakah dengan *website* ini mempermudah anda dalam membeli produk di butik Yello?

No	Keterangan	Responden	Prosentase
1	Sangat Mempermudah	12	40%
2	Mempermudah	18	60%
3	Biasa Saja		
4	Kurang Mempermudah		
5	Tidak Mempermudah		

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 12 orang atau 40% mengatakan bahwa website ini sangat memudahkan dalam membeli produk, sementara 18 orang atau 60% mengatakan bahwa website ini memudahkan dalam melakukan pembelian produk. Sehingga secara keseluruhan website ini memudahkan dalam melakukan transaksi pembelian barang.

5. Apakah informasi proses pembelian pada *website* sudah sesuai dengan yang anda pesan?

No	Keterangan	Responden	Prosentase
1	Sangat Sesuai	24	80%
2	Sesuai	6	20%
3	Biasa Saja		
4	Kurang Sesuai		
5	Tidak Sesuai		

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 24 orang atau 80% mengatakan bahwa website ini memberikan informasi proses pemesanan sangat sesuai dengan yang dipesan, sementara 6 orang atau 20% mengatakan bahwa website ini memberikan informasi proses

pemesanan sesuai dengan yang dipesan. Sehingga secara keseluruhan website ini memberikan informasi proses pemesanan yang sesuai dengan pesanan yang dilakukan.

6. Apakah sistem pembayaran yang digunakan memudahkan anda ?

No	Keterangan	Responden	Prosentase
1	Sangat Mudah	6	20%
2	Mudah	21	70%
3	Biasa Saja	3	10%
4	Kurang Mudah		
5	Tidak Mudah		

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 6 orang atau 20% mengatakan bahwa sistem pembayaran yang digunakan sangat memudahkan, sementara 21 orang atau 70% mengatakan bahwa sistem pembayaran yang digunakan mudah dan 3 orang atau 10% mengatakan sistem pembayaran yang digunakan biasa saja. Sehingga secara keseluruhan sistem pembayaran dalam website ini memudahkan pelanggan.

4.4.3.2 Kesimpulan Hasil Pengujian Betha

Berdasarkan hasil prosentase diatas yang didapatkan dari pengujian betha, maka sistem penerapan *E-commerce* untuk sistem penjualan ini sudah sesuai dengan tujuan yaitu pelanggan dalam melakukan transaksi pembelian secara online, tampilannya *user friendly,* memudahkan dalam pengelolaan data barang dan memudahkan dalam membuat laporan rekapitulasi penjualan.