BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Tahap implementasi merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan tahap dimana sistem siap untuk dioperasikan serta merupakan tahap untuk menerjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dapat dimengerti oleh mesin, yang terdiri dari penjelasan mengenai lingkungan implementasi, dan implementasi program. Wujud dari hasil tahap implementasi ini nantinya adalah sebuah sistem yang siap untuk diuji dan digunakan.

4.1.1 Perangkat Keras yang Digunakan

Spesifikasi perangkat keras (*hardware*) yang digunakan dalam pembangunan aplikasi *e-commerce* di CV. Elvana dapat dilihat pada tabel 4.1.

Tabel 4.1 Perangkat Keras yang Digunakan

Perangkat Keras	Keterangan
Processor	AMD Phenom II X3 720 Processor – 2,8 GHz
Memory	DDR2 2GB
Harddisk	SATA 500 GB
Video Card	Nvidia 9600 GT 512MB
Monitor	Resolusi 1360x768
Mouse	-
Keyboard	-

4.1.2 Perangkat Lunak yang Digunakan

Spesifikasi perangkat lunak (*software*) yang digunakan dalam pembangunan aplikasi *e-commerce* di CV. Elvana dapat dilihat pada tabel 4.2.

Tabel 4.2 Perangkat Lunak yang Digunakan

Perangkat Lunak	Keterangan
Windows 7 Professional	Sebagai Sistem Operasi
PHP	Sebagai Bahasa Pemrograman
MySQL	Sebagai *Database*
PhpMyAdmin	Sebagai *Interface Database*
Wamp Server 2.0	Sebagai *Web Server Local*
Adobe Dreamweaver CS4	Sebagai *Code Editor*
Mozilla Firefox	Sebagai *Web Browser*
Microsoft Visio 2007	Memodelkan analisis perangkat lunak

4.2 Implementasi Basis Data

Pada tahap awal pembangunan sebuah aplikasi, dibutuhkan media penyimpanan data (*database*). Implementasi *database* untuk aplikasi *e-commerce* di CV. Elvana ini adalah sebagai berikut.

1. Tabel Admin

```
CREATE TABLE `tbl_admin` (
  `id_admin` char(5) NOT NULL,
  `nama_lengkap` varchar(50) NOT NULL,
  `username` varchar(20) NOT NULL,
  `email` varchar(50) NOT NULL,
  `password` varchar(70) NOT NULL,
  `level_admin` enum('Administrator','Petugas') NOT NULL,
  PRIMARY KEY (`id_admin`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. Tabel Pelanggan

```
CREATE TABLE `tbl_pelanggan` (
  `id_pelanggan` char(6) NOT NULL,
  `nama_lengkap` varchar(50) NOT NULL,
  `username` varchar(30) NOT NULL,
  `email` varchar(50) NOT NULL,
  `password` varchar(70) NOT NULL,
  `ulang_password` varchar(70) NOT NULL,
  `pass_key` varchar(70) NOT NULL,
```

```
  `alamat_pelanggan` text NOT NULL,
  `id_propinsi` int(11) NOT NULL,
  `id_kabkota` int(11) NOT NULL,
  `kodepos` int(5) NOT NULL,
  `no_hp` varchar(15) NOT NULL,
  `status_aktifasi` enum('Sudah','Belum') NOT NULL,
  PRIMARY KEY (`id_pelanggan`),
  UNIQUE KEY `email` (`email`),
  KEY `id_propinsi` (`id_propinsi`),
  KEY `id_kabkota` (`id_kabkota`),
  CONSTRAINT `tbl_pelanggan_ibfk_6` FOREIGN KEY (`id_kabkota`) REFERENCES
`tbl_kabkota` (`id_kabkota`),
  CONSTRAINT `tbl_pelanggan_ibfk_5` FOREIGN KEY (`id_propinsi`) REFERENCES
`tbl_propinsi` (`id_propinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Tabel Kategori

```
CREATE TABLE `tbl_kategori` (
  `id_kategori` char(5) NOT NULL,
  `nama_kategori` varchar(50) NOT NULL,
  PRIMARY KEY (`id_kategori`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4. Tabel Subkategori

```
CREATE TABLE `tbl_subkategori` (
  `id_subkategori` char(5) NOT NULL,
  `id_kategori` char(5) NOT NULL,
  `nama_subkategori` varchar(50) NOT NULL,
  PRIMARY KEY (`id_subkategori`),
  KEY `id_kategori` (`id_kategori`),
  CONSTRAINT `tbl_subkategori_ibfk_1` FOREIGN KEY (`id_kategori`) REFERENCES
`tbl_kategori` (`id_kategori`) ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5. Tabel Ukuran

```
CREATE TABLE `tbl_ukuran` (
  `id_ukuran` char(5) NOT NULL,
  `id_subkategori` char(5) NOT NULL,
  `besar_ukuran` varchar(20) NOT NULL,
  PRIMARY KEY (`id_ukuran`),
  KEY `id_subkategori` (`id_subkategori`),
  CONSTRAINT `tbl_ukuran_ibfk_1` FOREIGN KEY (`id_subkategori`) REFERENCES
`tbl_subkategori` (`id_subkategori`) ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6. Tabel Warna

```
CREATE TABLE `tbl_warna` (
  `id_warna` char(5) NOT NULL,
  `nama_warna` varchar(30) NOT NULL,
  PRIMARY KEY (`id_warna`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

7. Tabel Produk

```
CREATE TABLE `tbl_produk` (
  `id_produk` char(6) NOT NULL,
  `id_kategori` char(5) NOT NULL,
  `id_subkategori` char(5) NOT NULL,
  `nama_produk` varchar(50) NOT NULL,
  `harga` int(20) NOT NULL,
  `file_gambar` varchar(200) NOT NULL,
  `deskripsi` text NOT NULL,
  `tgl_diinputkan` date NOT NULL,
  `counter` int(11) NOT NULL,
  `id_admin` char(5) NOT NULL,
  PRIMARY KEY (`id_produk`),
  KEY `id_kategori` (`id_kategori`),
  KEY `id_subkategori` (`id_subkategori`),
  KEY `id_admin` (`id_admin`),
  CONSTRAINT `tbl_produk_ibfk_5` FOREIGN KEY (`id_subkategori`) REFERENCES `tbl_subkategori` (`id_subkategori`) ON UPDATE CASCADE,
  CONSTRAINT `tbl_produk_ibfk_3` FOREIGN KEY (`id_admin`) REFERENCES `tbl_admin` (`id_admin`),
  CONSTRAINT `tbl_produk_ibfk_4` FOREIGN KEY (`id_kategori`) REFERENCES `tbl_kategori` (`id_kategori`) ON UPDATE CASCADE
```

8. Tabel Detail Produk

```
CREATE TABLE `tbl_detail_produk` (
  `id_detail_produk` char(6) NOT NULL,
  `id_produk` char(6) NOT NULL,
  `id_ukuran` char(5) NOT NULL,
  `id_warna` char(5) NOT NULL,
  `stok` int(20) NOT NULL,
  `berat` float NOT NULL,
  `diskon` int(5) NOT NULL,
  PRIMARY KEY (`id_detail_produk`),
  KEY `id_produk` (`id_produk`),
  KEY `id_ukuran` (`id_ukuran`),
  KEY `id_warna` (`id_warna`),
  CONSTRAINT `tbl_detail_produk_ibfk_5` FOREIGN KEY (`id_produk`) REFERENCES `tbl_produk` (`id_produk`) ON UPDATE CASCADE,
```

```
 CONSTRAINT `tbl_detail_produk_ibfk_3` FOREIGN KEY (`id_warna`) REFERENCES `tbl_warna` (`id_warna`),
 CONSTRAINT `tbl_detail_produk_ibfk_4` FOREIGN KEY (`id_ukuran`) REFERENCES `tbl_ukuran` (`id_ukuran`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

9. Tabel Propinsi

```
CREATE TABLE `tbl_propinsi` (
  `id_propinsi` int(10) NOT NULL,
  `nama` varchar(100) NOT NULL,
  PRIMARY KEY (`id_propinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

10. Tabel Kab/kota

```
CREATE TABLE `tbl_kabkota` (
  `id_propinsi` int(10) NOT NULL,
  `id_kabkota` int(10) NOT NULL,
  `nama_kabkota` varchar(100) NOT NULL,
  PRIMARY KEY (`id_kabkota`),
  KEY `id_propinsi` (`id_propinsi`),
  CONSTRAINT `tbl_kabkota_ibfk_1` FOREIGN KEY (`id_propinsi`) REFERENCES `tbl_propinsi` (`id_propinsi`) ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

11. Tabel Jenis Pengiriman

```
CREATE TABLE `tbl_jenis_pengiriman` (
  `id_jenis_pengiriman` char(5) NOT NULL,
  `nama_jenis_pengiriman` varchar(50) NOT NULL,
  UNIQUE KEY `id_jenis_pengiriman` (`id_jenis_pengiriman`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

12. Tabel Ongkos Kirim

```
CREATE TABLE `tbl_ongkos` (
  `id_ongkos` char(5) NOT NULL,
  `id_jenis_pengiriman` char(5) NOT NULL,
  `id_kabkota` int(10) NOT NULL,
  `biaya` int(20) NOT NULL,
  PRIMARY KEY (`id_ongkos`),
  KEY `id_jenis_pengiriman` (`id_jenis_pengiriman`),
  KEY `id_kabkota` (`id_kabkota`),
  CONSTRAINT `tbl_ongkos_ibfk_4` FOREIGN KEY (`id_kabkota`) REFERENCES `tbl_kabkota` (`id_kabkota`) ON UPDATE CASCADE,
  CONSTRAINT `tbl_ongkos_ibfk_3` FOREIGN KEY (`id_jenis_pengiriman`) REFERENCES `tbl_jenis_pengiriman` (`id_jenis_pengiriman`) ON UPDATE CASCADE
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

13. Tabel Mata Uang

```
CREATE TABLE `tbl_mata_uang` (
  `kode` varchar(3) NOT NULL,
  `kurs` double NOT NULL,
  PRIMARY KEY (`kode`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

14. Tabel Retur Barang

```
CREATE TABLE `tbl_returbarang` (
  `id_retur` char(6) NOT NULL,
  `no_pesanan` char(8) NOT NULL,
  `id_pelanggan` char(6) NOT NULL,
  `status_retur_barang` enum('Belum Disetujui','Tidak Disetujui','Disetujui') NOT NULL,
  `status_penerimaan_barang_retur` enum('Belum Diterima','Sudah Diterima') NOT NULL,
  `status_pengiriman_retur_barang` enum('Belum Dikirim','Sudah Dikirim','Diterima') NOT NULL,
  `tgl_retur` datetime NOT NULL,
  `no_resi_retur` varchar(20) NOT NULL,
  PRIMARY KEY (`id_retur`),
  KEY `no_pesanan` (`no_pesanan`),
  KEY `id_pelanggan` (`id_pelanggan`),
  CONSTRAINT `tbl_returbarang_ibfk_1` FOREIGN KEY (`no_pesanan`) REFERENCES `tbl_pesanan` (`no_pesanan`),
  CONSTRAINT `tbl_returbarang_ibfk_2` FOREIGN KEY (`id_pelanggan`) REFERENCES `tbl_pelanggan` (`id_pelanggan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

15. Tabel Detail Retur Barang

```
CREATE TABLE `tbl_detail_retur` (
  `id_detail_retur` char(6) NOT NULL,
  `id_retur` char(6) NOT NULL,
  `id_detail_produk` char(6) NOT NULL,
  `id_ukuran` char(5) NOT NULL,
  `id_warna` char(5) NOT NULL,
  `qty_retur` int(20) NOT NULL,
  `keterangan` enum('Kerusakan Fisik','Kerusakan Fungsional','Jumlah Tidak Sesuai','Warna Tidak Sesuai','Ukuran Tidak Sesuai','Tipe Tidak Sesuai','Lain-lain') NOT NULL,
  PRIMARY KEY (`id_detail_retur`),
  KEY `id_retur` (`id_retur`),
  KEY `id_detail_produk` (`id_detail_produk`),
  KEY `id_ukuran` (`id_ukuran`),
  KEY `id_warna` (`id_warna`),
  CONSTRAINT `tbl_detail_retur_ibfk_1` FOREIGN KEY (`id_retur`) REFERENCES
```

```
`tbl_returbarang` (`id_retur`),
 CONSTRAINT `tbl_detail_retur_ibfk_2` FOREIGN KEY (`id_detail_produk`) REFERENCES `tbl_detail_pesanan` (`id_detail_produk`),
 CONSTRAINT `tbl_detail_retur_ibfk_3` FOREIGN KEY (`id_ukuran`) REFERENCES `tbl_detail_pesanan` (`id_ukuran`),
 CONSTRAINT `tbl_detail_retur_ibfk_4` FOREIGN KEY (`id_warna`) REFERENCES `tbl_detail_pesanan` (`id_warna`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

16. Tabel Keranjang

```
CREATE TABLE `tbl_keranjang` (
  `id_pelanggan` char(6) NOT NULL,
  `id_detail_produk` char(6) NOT NULL,
  `id_ukuran` char(5) NOT NULL,
  `id_warna` char(5) NOT NULL,
  `nama_produk` varchar(30) NOT NULL,
  `harga` int(20) NOT NULL,
  `qty` int(20) NOT NULL,
  `total_berat` float NOT NULL,
  `subtotal` int(20) NOT NULL,
  KEY `id_pelanggan` (`id_pelanggan`),
  KEY `id_detail_produk` (`id_detail_produk`),
  CONSTRAINT `tbl_keranjang_ibfk_1` FOREIGN KEY (`id_pelanggan`) REFERENCES `tbl_pelanggan` (`id_pelanggan`),
  CONSTRAINT `tbl_keranjang_ibfk_2` FOREIGN KEY (`id_detail_produk`) REFERENCES `tbl_detail_produk` (`id_detail_produk`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

17. Tabel Pesanan

```
CREATE TABLE `tbl_pesanan` (
  `no_pesanan` char(8) NOT NULL,
  `id_pelanggan` char(6) NOT NULL,
  `tgl_dipesan` datetime NOT NULL,
  `tgl_expired` datetime NOT NULL,
  `tgl_terima` datetime NOT NULL,
  `id_ongkos` char(5) DEFAULT NULL,
  `tarif_ongkos_kirim` int(20) NOT NULL,
  `nama_penerima` varchar(50) NOT NULL,
  `alamat_penerima` text NOT NULL,
  `no_telp_penerima` varchar(15) NOT NULL,
  `status_pengiriman` enum('Belum Dikirim','Sudah Dikirim','Diterima') NOT NULL,
  `status_pesanan` enum('Dipesan','Dibatalkan','Dibeli','Direturkan') NOT NULL,
  `total_berat_produk` float NOT NULL,
  `total_pembayaran` int(15) NOT NULL,
  `no_resi` varchar(20) NOT NULL,
  `id_admin` char(5) NOT NULL,
```

```
  PRIMARY KEY (`no_pesanan`),
  KEY `id_pelanggan` (`id_pelanggan`),
  KEY `id_ongkos` (`id_ongkos`),
  KEY `id_admin` (`id_admin`),
  CONSTRAINT `tbl_pesanan_ibfk_1` FOREIGN KEY (`id_pelanggan`) REFERENCES
`tbl_pelanggan` (`id_pelanggan`),
  CONSTRAINT `tbl_pesanan_ibfk_2` FOREIGN KEY (`id_ongkos`) REFERENCES
`tbl_ongkos` (`id_ongkos`)
  CONSTRAINT `tbl_pesanan_ibfk_3` FOREIGN KEY (`id_admin`) REFERENCES
`tbl_ongkos` (`id_admin`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

18. Tabel Detail Pesanan

```
CREATE TABLE `tbl_detail_pesanan` (
  `no_pesanan` char(8) NOT NULL,
  `id_detail_produk` char(6) NOT NULL,
  `id_ukuran` char(5) NOT NULL,
  `id_warna` char(5) NOT NULL,
  `qty_pesanan` int(20) NOT NULL,
  `subharga` int(20) NOT NULL,
  `diskon` int(5) NOT NULL,
  PRIMARY KEY (`no_pesanan`,`id_detail_produk`),
  KEY `no_pesanan` (`no_pesanan`),
  KEY `id_detail_produk` (`id_detail_produk`),
  KEY `id_ukuran` (`id_ukuran`),
  KEY `id_warna` (`id_warna`),
  CONSTRAINT `tbl_detail_pesanan_ibfk_1` FOREIGN KEY (`no_pesanan`) REFERENCES
`tbl_pesanan` (`no_pesanan`),
  CONSTRAINT `tbl_detail_pesanan_ibfk_2` FOREIGN KEY (`id_detail_produk`)
REFERENCES `tbl_detail_produk` (`id_detail_produk`),
  CONSTRAINT `tbl_detail_pesanan_ibfk_3` FOREIGN KEY (`id_ukuran`) REFERENCES
`tbl_ukuran` (`id_ukuran`),
  CONSTRAINT `tbl_detail_pesanan_ibfk_4` FOREIGN KEY (`id_warna`) REFERENCES
`tbl_warna` (`id_warna`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

19. Tabel Pembayaran

```
CREATE TABLE `tbl_pembayaran` (
  `id_pembayaran` char(8) NOT NULL,
  `no_pesanan` char(8) NOT NULL,
  `kode` varchar(3) DEFAULT NULL,
  `tgl_bayar` datetime NOT NULL,
  `transfer_dari_bank` varchar(30) NOT NULL,
  `no_rekening_pengirim` varchar(20) NOT NULL,
  `tujuan_no_rekening` varchar(50) NOT NULL,
  `atas_nama` varchar(50) NOT NULL,
```

```
  `status_pembayaran` enum('Belum Dibayar','Sudah Dibayar','Lunas') NOT NULL,
  `total_tbl_pembayaran` int(15) NOT NULL,
  `jumlah_yang_dibayar` int(15) NOT NULL,
  `id_paypal` varchar(50) NOT NULL,
  `kode_transaksi` varchar(30) NOT NULL,
  PRIMARY KEY (`id_pembayaran`),
  KEY `no_pesanan` (`no_pesanan`),
  KEY `kode` (`kode`),
  CONSTRAINT `tbl_pembayaran_ibfk_1` FOREIGN KEY (`no_pesanan`) REFERENCES
`tbl_pesanan` (`no_pesanan`),
  CONSTRAINT `tbl_pembayaran_ibfk_2` FOREIGN KEY (`kode`) REFERENCES
`tbl_mata_uang` (`kode`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4.3 Implementasi Antarmuka

Implementasi antarmuka dilakukan pada setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna. Adapun untuk tampilan implementasi pembangunan aplikasi *e-commerce* di CV. Elvana berbasis *web* akan dilampirkan pada halaman lampiran.

Tabel 4.3 Implementasi Antarmuka Administrator dan Petugas

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login administrator atau petugas	index.php
Beranda	Merupakan tampilan beranda administrator atau petugas	halaman_admin.php
Edit data profil	Digunakan untuk merubah profil administrator atau petugas	admin_proses_edit1.php
Ubah password	Digunakan untuk merubah password administrator atau petugas	ganti_password_admin.php
Data admin	Digunakan administrator untuk melihat, menambah, menghapus dan mencari data petugas	admin_list.php
Data pelanggan	Digunakan administrator untuk melihat, menambah, menghapus dan mencari data pelanggan	pelanggan_list.php
Data kategori	Digunakan untuk melihat, menambah, merubah, menghapus dan mencari data kategori	kategori_list.php
Data subkategori	Digunakan untuk melihat, menambah, merubah, menghapus	subkategori_list.php

	dan mencari data subkategori	
Data ukuran	Digunakan untuk melihat, menambah, merubah, menghapus dan mencari data ukuran	ukuran_list.php
Data warna	Digunakan untuk melihat, menambah, merubah, menghapus dan mencari data warna	warna_list.php
Data produk	Digunakan untuk melihat, menambah, merubah, menghapus dan mencari data produk serta melihat data detail produk	produk_list.php
Data detail produk	Digunakan untuk melihat, menambah, merubah, menghapus dan mencari data detail produk	detail_produk.php
Data Propinsi	Digunakan untuk melihat, menambah, merubah, menghapus dan mencari data propinsi	propinsi_list.php
Data Kab/kota	Digunakan untuk melihat, menambah, merubah, menghapus dan mencari data kab/kota	kabkota_list.php
Data jenis pengiriman	Digunakan untuk melihat, menambah, merubah, menghapus dan mencari data jenis pengiriman	jenis_pengiriman_list.php
Data ongkos kirim	Digunakan untuk melihat, menambah, merubah, menghapus dan mencari data ongkos kirim	ongkoskirim_list.php
Data mata uang	Digunakan untuk melihat, menambah, merubah, menghapus dan mencari data mata uang	matauang_list.php
Data retur barang	Digunakan untuk mengolah data retur barang	returbarang_list.php
Data transaksi	Digunakan untuk mengolah data transaksi yaitu data pemesanan dan data pembayaran	pesanan_list.php
Laporan produk	Digunakan untuk melihat laporan produk per-tanggal, per-bulan dan mencetak laporan	laporan_semua_produk.php
Laporan pesanan	Digunakan untuk melihat laporan pesanan per-tanggal, per-bulan dan mencetak laporan	laporan_semua_pesanan.php
Laporan penjualan	Digunakan untuk melihat laporan penjualan per-tanggal, per-bulan dan mencetak laporan	laporan_semua_penjualan.php
Logout	Digunakan administrator atau petugas untuk keluar dari aplikasi	logout.php

Tabel 4.4 Implementasi Antarmuka Pelanggan dan Pengunjung

Menu	Deskripsi	Nama File
Login	Digunakan pelanggan sebagai halaman login	login.php
Lupa password	Digunakan jika pelanggan lupa password untuk login	lupa_pass_login.php
Beranda	Merupakan tampilan beranda pelanggan dan pengunjung	index.php
Edit data profil	Digunakan untuk merubah profil pelanggan	edit_data_pelanggan.php
Ubah password	Digunakan untuk merubah password pelanggan	ubah_password_pelanggan.php
Tentang kami	Digunakan pelanggan dan pengunjung untuk melihat profil perusahaan	tentangkami.php
Cara pesan dan bayar	Digunakan pelanggan dan pengunjung untuk melihat cara pemesanan dan pembayaran	pesanbayar.php
History pemesanan dan pembayaran	Digunakan pelanggan untuk melihat history transaksi dan digunakan untuk melakukan pembayaran	history_pesanan.php
Klaim retur barang	Digunakan pelanggan untuk mereturkan pesanan	retur_barang.php
Daftar	Digunakan pengunjung yang ingin mendaftar menjadi *member*	register.php
Detail Produk	Digunakan pelanggan dan pengunjung untuk melihat data detail produk	detail_produk.php
Beli Produk	Digunakan pelanggan untuk memilih ukuran dan warna produk yang dibeli	beli_produk.php
Keranjang Belanja	Digunakan pelanggan untuk melihat, merubah, menghapus data produk yang dipesan	list_keranjang_belanja.php
Isi data pengiriman	Digunakan pelanggan untuk mengisi data penerima pesanan	isidatatransaksi.php
Selesai Transaksi	Digunakan pelanggan untuk menampilkan data pesanan, mencetak nota pesanan dan pembayaran *PayPal*	selesai_transaksi.php
Logout	Digunakan pelanggan untuk keluar dari halaman pelanggan	logout.php

4.4 Pengujian Perangkat Lunak

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

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

4.4.1 Pengujian Alpha

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

4.4.1.1 Kasus dan Hasil Pengujian

1. Pengujian Pendaftaran Pelanggan

Berikut ini merupakan pengujian pendaftaran pelanggan dapat dilihat pada tabel 4.5 dibawah ini.

Tabel 4.5 Pengujian Pendaftaran Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data pendaftaran, contoh : nama lengkap : yordie akbar *username* : yordie *email* : original_tastee@yahoo.com password : 12345	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses pendaftaran	Form pendaftaran terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses pendaftaran	[√] Diterima [ ] Ditolak

ulang password : 12345 alamat : jl. Cikampek 4 no.1 propinsi : jawa barat kab/kota : bandung kodepos : 40291 nomor *handphone* : 085724072325			
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data nama lengkap kosong	Muncul pesan kesalahan "Nama lengkap harus diisi"	Dapat menampilkan pesan kesalahan "Nama lengkap harus diisi"	[√] Diterima [ ] Ditolak
Data *username* kosong	Muncul pesan kesalahan "*Username* harus diisi"	Dapat menampilkan pesan kesalahan "*Username* harus diisi"	[√] Diterima [ ] Ditolak
Data *email* kosong	Muncul pesan kesalahan "*Email* harus diisi"	Dapat menampilkan pesan kesalahan "*Email* harus diisi"	[√] Diterima [ ] Ditolak
Data password kosong	Muncul pesan kesalahan "Password harus diisi"	Dapat menampilkan pesan kesalahan "Password harus diisi"	[√] Diterima [ ] Ditolak
Data ulang password	Muncul pesan kesalahan "Ulang password harus diisi"	Dapat menampilkan pesan kesalahan "Ulang password harus diisi"	[√] Diterima [ ] Ditolak

Data alamat kosong	Muncul pesan kesalahan "Alamat harus diisi"	Dapat menampilkan pesan kesalahan "Alamat harus diisi"	[√] Diterima [ ] Ditolak
Data propinsi kosong	Muncul pesan kesalahan "Propinsi harus diisi"	Dapat menampilkan pesan kesalahan "Propinsi harus diisi"	[√] Diterima [ ] Ditolak
Data kab/kota kosong	Muncul pesan kesalahan "Kab/kota harus diisi"	Dapat menampilkan pesan kesalahan "Kab/kota harus diisi"	[√] Diterima [ ] Ditolak
Data kodepos kosong	Muncul pesan kesalahan "Kodepos harus diisi"	Dapat menampilkan pesan kesalahan "Kodepos harus diisi"	[√] Diterima [ ] Ditolak
Data nomor *handphone* kosong	Muncul pesan kesalahan "Nomor *handphone* harus diisi"	Dapat menampilkan pesan kesalahan "Nomor *handphone* harus diisi"	[√] Diterima [ ] Ditolak

2. Pengujian Login Pelanggan

Berikut ini merupakan pengujian login pelanggan dapat dilihat pada tabel 4.6 dibawah ini.

Tabel 4.6 Pengujian Login Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan alamat email: original_tastee@yahoo.com	Mengisikan data alamat email pada *field* alamat email	Dapat mengisi *field* alamat email pada *field*	[√] Diterima [ ] Ditolak

		alamat email	
Memasukkan password : 123456	Mengisikan data password pada *field* password	Dapat mengisi *field* password pada *field* password	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengosongkan data email atau data password	Menampilkan pesan kesalahan "Email dan Password Harus Diisi"	Dapat menampilkan pesan kesalahan "Email dan Password Harus Diisi"	[√] Diterima [ ] Ditolak
Data *email* tidak terdaftar	Menampilkan pesan kesalahan "Email Salah"	Dapat menampilkan pesan kesalahan "Email Salah"	[√] Diterima [ ] Ditolak
Data password tidak terdaftar	Menampilkan pesan kesalahan "Password Salah"	Dapat menampilkan pesan kesalahan "Password Salah"	[√] Diterima [ ] Ditolak

3. Pengujian Lupa Password Pelanggan

Berikut ini merupakan pengujian lupa password pelanggan dapat dilihat pada tabel 4.7 dibawah ini.

Tabel 4.7 Pengujian Lupa Password Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data email dengan benar original_tastee@yahoo.com	Data email dapat diproses	Data email dapat diproses	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data email tidak diisi	Menampilkan pesan kesalahan "Email Harus Diisi"	Dapat menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak

		"Email Harus Diisi"	
Data email tidak terdaftar "abcde@yahoo.com"	Menampilkan pesan kesalahan "Email yang Anda Masukkan Tidak Terdaftar"	Dapat menampilkan pesan kesalahan "Email yang Anda Masukkan Tidak Terdaftar"	[√] Diterima [ ] Ditolak

4. Pengujian Ubah Profil Pelanggan

Berikut ini merupakan pengujian uba profil pelanggan dapat dilihat pada tabel 4.8 dibawah ini.

Tabel 4.8 Pengujian Ubah Profil Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan nama lengkap : yordie akbar haryoko	Mengisi data nama lengkap pada *field* nama lengkap	Dapat mengisi *field* nama lengkap	[√] Diterima [ ] Ditolak
Memasukkan *username* : yordie	Mengisi data *username* pada *field username*	Dapat mengisi *field username*	[√] Diterima [ ] Ditolak
Memasukkan email : yordie.akbar@gmail.com	Mengisi data email pada *field* email	Dapat mengisi *field* email	[√] Diterima [ ] Ditolak
Memasukkan alamat : jl. dago no.20	Mengisi data alamat pada *field* alamat	Dapat mengisi *field* alamat	[√] Diterima [ ] Ditolak
Memasukkan propinsi : jawa barat	Mengisi data propinsi pada *field* propinsi	Dapat mengisi *field* propinsi	[√] Diterima [ ] Ditolak
Memasukkan kab/kota : bandung	Mengisi data kab/kota pada *field* kab/kota	Dapat mengisi *field* kab/kota	[√] Diterima [ ] Ditolak
Memasukkan kodepos : 40291	Mengisi data kodepos pada *field* kodepos	Dapat mengisi *field* kodepos	[√] Diterima [ ] Ditolak
Memasukkan nomor	Mengisi data	Dapat mengisi	[√] Diterima

handphone : 0813446699331	nomor *handphone* pada *field* nomor *handphone*	*field* nomor *handphone*	[ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama lengkap kosong	Tampil pesan "Nama Lengkap Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak
Username kosong	Tampil pesan "*Username* Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak
Email kosong	Tampil pesan "Email Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak
Alamat kosong	Tampil pesan "Alamat Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak
Propinsi kosong	Tampil pesan "Propinsi Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak
Kab/kota kosong	Tampil pesan "Kab/kota Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak
Kodepos kosong	Tampil pesan "Kodepos Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak
Nomor *handphone* kosong	Tampil pesan "Nomor *handphone* Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

5. Pengujian Ubah Password Pelanggan

Berikut ini merupakan pengujian ubah password pelanggan dapat dilihat pada tabel 4.9 dibawah ini.

Tabel 4.9 Pengujian Ubah Password Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan data yang benar	Password baru berhasil disimpan ke dalam data	Dapat menyimpan data password pelanggan yang	[√] Diterima [ ] Ditolak

	pelanggan	baru	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan data *username* yang salah	Muncul pesan kesalahan "Data *username* anda salah"	Dapat menampilkan pesan kesalahan "Data *username* anda salah"	[√] Diterima [ ] Ditolak
Memasukkan data password lama yang salah	Muncul pesan kesalahan "Data password lama anda salah"	Dapat menampilkan pesan kesalahan "Data password lama anda salah"	[√] Diterima [ ] Ditolak
Memasukkan data kosong	Muncul pesan kesalahan "*Field* harus diisi"	Dapat menampilkan pesan kesalahan "*Field* harus diisi"	[√] Diterima [ ] Ditolak

6. Pengujian Keranjang Belanja

Berikut ini merupakan pengujian keranjang belanja dapat dilihat pada tabel 4.10 dibawah ini.

Tabel 4.10 Pengujian Keranjang Belanja

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memilih data warna dan data ukuran	Data produk yang dipesan tersimpan di keranjang	Dapat menyimpan data produk yang dipesan ke kreanjang belanja	[√] Diterima [ ] Ditolak
Menampilkan data jumlah pesanan	Tercantum pada *textbox* data yang akan diubah	Dapat mengisi jumlah data dengan benar	[√] Diterima [ ] Ditolak
Mengubah jumlah pemesanan	Data jumlah yang diubah tersimpan dalam data keranjang	Dapat menampilkan perubahan data jumlah	[√] Diterima [ ] Ditolak
Menghapus data pemesanan	Data yang dihapus terhapus dari keranjang belanja	Dapat menghapus data pada keranjang belanja	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan jumlah pemesanan	Melebihi stok yang ada akan	Dapat menampilkan	[√] Diterima [ ] Ditolak

produk	muncul pesan kesalahan "Maaf Jumlah yang Dipesan Melebihi Stok yang Ada"	pesan kesalahan "Maaf Jumlah yang Dipesan Melebihi Stok yang Ada"	

7. Pengujian Isi Data Pengiriman

Berikut ini merupakan pengujian isi data pengiriman dapat dilihat pada tabel 4.11 dibawah ini.

Tabel 4.11 Pengujian Isi Data Pengiriman

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data pengiriman	Data pengiriman baru akan tersimpan di dalam *database*	Dapat menyimpan data pengiriman baru di dalam *database*	[√] Diterima [ ] Ditolak
Proses data pengiriman	Data alamat pengiriman sama dengan data alamat pelanggan yang ada dalam *database*	Dapat menyimpan data pengiriman dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengosongkan salah satu *field*	Menampilkan pesan kesalahan "Field Harus Diisi"	Dapat menampilkan pesan kesalahan "Field Harus Diisi"	[√] Diterima [ ] Ditolak

8. Pengujian Konfirmasi Pembayaran

Berikut ini merupakan pengujian konfirmasi pembayaran dapat dilihat pada tabel 4.12 dibawah ini.

Tabel 4.12 Pengujian Konfirmasi Pembayaran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Masukkan data konfirmasi pembayaran	Jika salah satu *field* tidak terisi atau salah dalam	Dapat memunculkan jika salah satu *field*	[√] Diterima [ ] Ditolak

contoh : transfer dari bank : Bank BCA jenis transfer : rekening nomor rekening : 0131442281 tujuan nomor rekening : BCA No. Rekening 280-354-1302 atas nama : Yordie uang yang di transfer : 100.000	pengisian, maka akan muncul pesan kesalahan bahwa *field* tersebut harus terisi.	tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa *field* tersebut harus terisi.	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengosongkan salah satu *field*	Muncul pesan kesalahan "*Field* Harus Diisi"	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

9. Pengujian Klaim Retur Barang

Berikut ini merupakan pengujian klaim retur barang dapat dilihat pada tabel 4.13 dibawah ini.

Tabel 4.13 Pengujian Klaim Retur Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memilih nomor pesanan yang akan direturkan : OR-000046	Menampilkan data detail pesanan yang akan direturkan	Dapat menampilkan data detail pesanan	[√] Diterima [ ] Ditolak
Mengubah jumlah produk yang akan diretur	Data jumlah yang diubah tersimpan dalam data detail retur barang	Dapat menampilkan perubahan data jumlah	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan data	Menampilkan	Dapat	[√] Diterima

jumlah retur melebihi jumlah pesanan	pesan kesalahan "Maaf produk yang diretur melebihi jumlah pesanan"	menampilkan pesan kesalahan "Maaf produk yang diretur melebihi jumlah pesanan"	[ ] Ditolak

10. Pengujian Logout Pelanggan

Berikut ini merupakan pengujian logout pelanggan dapat dilihat pada tabel 4.14 dibawah ini.

Tabel 4.14 Pengujian Logout Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Session email	Dapat keluar dan menghapus session pengguna	*User* keluar dari aplikasi	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Session email	Tidak dapat keluar dan menghapus session pengguna	*User* tidak dapat keluar	[√] Diterima [ ] Ditolak

11. Pengujian Login Administrator

Berikut ini merupakan pengujian aplikasi *login* administrator dapat dilihat pada tabel 4.15 dibawah ini.

Tabel 4.15 Pengujian Login Administrator

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan *username* : administrator	Mengisikan data *username* pada *field username*	Dapat mengisi *field username* pada field *username*	[√] Diterima [ ] Ditolak
Memasukkan password : administrator	Mengisikan data password pada *field* password	Dapat mengisi *field* password pada *field* password	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan data	Tampilkan pesan	Dapat	[√] Diterima

username dan password yang tidak terdaftar	kesalahan "Login *Invalid*"	menampilkan pesan kesalahan "Login *Invalid*"	[ ] Ditolak
Mengosongkan data *username* dan password	Tampilkan pesan kesalahan "Login *Invalid*"	Dapat menampilkan pesan kesalahan "Login *Invalid*"	[√] Diterima [ ] Ditolak
Mengosongkan *username*	Muncul pesan kesalahan "*Username* atau Password yang Anda Masukkan Salah"	Muncul pesan kesalahan "*Username* atau Password yang Anda Masukkan Salah"	[√] Diterima [ ] Ditolak
Mengosongkan password	Muncul pesan kesalahan "*Username* atau Password yang Anda Masukkan Salah"	Muncul pesan kesalahan "*Username* atau Password yang Anda Masukkan Salah"	[√] Diterima [ ] Ditolak

12. Pengujian Ubah Profil Administrator

Berikut ini merupakan pengujian aplikasi ubah profil administrator dapat dilihat pada tabel 4.16 dibawah ini.

Tabel 4.16 Pengujian Ubah Profil Administrator

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan nama lengkap : Administrator Elvana	Mengisi nama lengkap pada *field* nama lengkap	Dapat mengisi *field* nama lengkap	[√] Diterima [ ] Ditolak
Memasukkan *username* : Administrator	Mengisi username pada *field username*	Dapat mengisi *field username*	[√] Diterima [ ] Ditolak
Memasukkan email : administrator@gmail.com	Mengisi email pada *field* email	Dapat mengisi *field* email	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama lengkap kosong	Tampil pesan "Nama Lengkap Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

Username kosong	Tampil pesan "*Username* Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak
Email kosong	Tampil pesan "Email Harus Diisi"	Tampil pesan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

13. Pengujian Ubah Password Administrator

Berikut ini merupakan pengujian aplikasi ubah password administrator dapat dilihat pada tabel 4.17 dibawah ini.

Tabel 4.17 Pengujian Ubah Password Administrator

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan data yang benar	Password baru berhasil disimpan ke dalam data administrator	Dapat menyimpan data password administrator yang baru	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Memasukkan data *username* yang salah	Muncul pesan kesalahan "Data *username* anda salah"	Dapat menampilkan pesan kesalahan "Data *username* anda salah"	[√] Diterima [ ] Ditolak
Memasukkan data password lama yang salah	Muncul pesan kesalahan "Data password lama anda salah"	Dapat menampilkan pesan kesalahan "Data password lama anda salah"	[√] Diterima [ ] Ditolak
Memasukkan data kosong	Muncul pesan kesalahan "*Field* harus diisi"	Dapat menampilkan pesan kesalahan "*Field* harus diisi"	[√] Diterima [ ] Ditolak

14. Pengujian Tambah Petugas

Berikut ini merupakan pengujian tambah data petugas dapat dilihat pada tabel 4.18 dibawah ini.

Tabel 4.18 Pengujian Tambah Petugas

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data petugas, Contoh : nama lengkap : yordie akbar *username* : yordie *email* : original_tastee@yahoo.com password : 12345 *level* admin : petugas	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field data petugas semua terisi	Data petugas akan tersimpan dalam data petugas	Data petugas akan tersimpan dalam data petugas	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

15. Pengujian tambah kategori

Berikut ini merupakan pengujian tambah data kategori dapat dilihat pada tabel 4.19 dibawah ini.

Tabel 4.19 Pengujian Tambah Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori, Contoh : nama kategori : tas	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field data kategori semua terisi	Data kategori akan tersimpan dalam data kategori	Data kategori akan tersimpan dalam data kategori	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

16. Pengujian ubah kategori

Berikut ini merupakan pengujian ubah data kategori dapat dilihat pada tabel 4.20 dibawah ini.

Tabel 4.20 Pengujian Ubah Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori, Contoh : nama kategori : tas	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field di form ubah kategori semua terisi	Data kategori akan tersimpan dalam data kategori	Data kategori akan tersimpan dalam data kategori	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

17. Pengujian tambah subkategori

Berikut ini merupakan pengujian tambah data subkategori dapat dilihat pada tabel 4.21 dibawah ini.

Tabel 4.21 Pengujian Tambah Subkategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data subkategori, Contoh : nama kategori : tas nama subkategori : tas 2R	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field data subkategori semua terisi	Data subkategori akan tersimpan dalam data subkategori	Data subkategori akan tersimpan dalam data subkategori	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

18. Pengujian ubah subkategori

Berikut ini merupakan pengujian ubah data subkategori dapat dilihat pada tabel 4.22 dibawah ini.

Tabel 4.22 Pengujian Ubah Subkategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data	Jika *field* tidak	Dapat	[√] Diterima

subkategori, Contoh : nama kategori : tas nama subkategori : tas 2R	terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[ ] Ditolak
Field di form ubah subkategori semua terisi	Data subkategori akan tersimpan dalam data subkategori	Data subkategori akan tersimpan dalam data subkategori	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

19. Pengujian tambah ukuran

Berikut ini merupakan pengujian tambah data ukuran dapat dilihat pada tabel 4.23 dibawah ini.

Tabel 4.23 Pengujian Tambah Ukuran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data ukuran, Contoh : besar ukuran : XL	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field data ukuran semua terisi	Data ukuran akan tersimpan dalam data ukuran	Data ukuran akan tersimpan dalam data ukuran	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

20. Pengujian ubah ukuran

Berikut ini merupakan pengujian ubah data ukuran dapat dilihat pada tabel 4.24 dibawah ini.

Tabel 4.24 Pengujian Ubah Ukuran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data ukuran, Contoh : besar ukuran : XL	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field di form ubah ukuran semua terisi	Data ukuran akan tersimpan dalam data ukuran	Data ukuran akan tersimpan dalam data ukuran	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

21. Pengujian tambah warna

Berikut ini merupakan pengujian tambah data warna dapat dilihat pada tabel 4.25 dibawah ini.

Tabel 4.25 Pengujian Tambah Warna

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data warna, Contoh : nama warna : merah	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field data warna semua terisi	Data warna akan tersimpan dalam data warna	Data warna akan tersimpan dalam data warna	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

22. Pengujian ubah warna

Berikut ini merupakan pengujian ubah data warna dapat dilihat pada tabel 4.26 dibawah ini.

Tabel 4.26 Pengujian Ubah Warna

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data warna, Contoh : nama warna : XL	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field di form ubah warna semua terisi	Data warna akan tersimpan dalam data warna	Data warna akan tersimpan dalam data warna	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

23. Pengujian tambah produk

Berikut ini merupakan pengujian tambah data produk dapat dilihat pada tabel 4.27 dibawah ini.

Tabel 4.27 Pengujian Tambah Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data produk, Contoh :	Jika *field* tidak terisi atau salah	Dapat menampilkan	[√] Diterima [ ] Ditolak

nama kategori : tas nama subkategori : tas 2R nama produk : tas 2R sablon model 140 harga : 70.000 gambar : tas.jpg deskripsi : tas 2R merupakan tas yang cocok untuk kegiatan sehari-hari	maka muncul pesan kesalahan bahwa *field* harus diisi	pesan kesalahan bahwa *field* tersebut harus diisi	
Field data produk semua terisi	Data produk akan tersimpan dalam data produk	Data produk akan tersimpan dalam data produk	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

24. Pengujian ubah produk

Berikut ini merupakan pengujian ubah data produk dapat dilihat pada tabel 4.28 dibawah ini.

Tabel 4.28 Pengujian Ubah Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data produk, Contoh : nama kategori : tas nama subkategori : tas 2R nama produk : tas 2R sablon model 140 harga : 70.000 gambar : tas.jpg deskripsi : tas 2R merupakan tas	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak

yang cocok untuk kegiatan sehari-hari			
Field di form ubah produk semua terisi	Data produk akan tersimpan dalam data produk	Data produk akan tersimpan dalam data produk	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

25. Pengujian tambah detail produk

Berikut ini merupakan pengujian tambah data detail produk dapat dilihat pada tabel 4.29 dibawah ini.

Tabel 4.29 Pengujian Tambah Detail Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data detail produk, Contoh : nama produk : tas 2R model 147 ukuran : tidak ada ukuran warna : merah stok : 4 berat : 0.5kg diskon : 0%	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus	[√] Diterima [ ] Ditolak

		Diisi”	

26. Pengujian ubah detail produk

Berikut ini merupakan pengujian ubah data detail produk dapat dilihat pada tabel 4.30 dibawah ini.

Tabel 4.30 Pengujian Ubah Detail Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data detail produk, Contoh : nama produk : tas 2R model 147 ukuran : tidak ada ukuran warna : merah stok : 4 berat : 0.5kg diskon : 0%	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field di form ubah detail produk semua terisi	Data detail produk akan tersimpan dalam data detail produk	Data detail produk akan tersimpan dalam data detail produk	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan “*Field* Harus Diisi”	[√] Diterima [ ] Ditolak

27. Pengujian tambah propinsi

Berikut ini merupakan pengujian tambah data propinsi dapat dilihat pada tabel 4.31 dibawah ini.

Tabel 4.31 Pengujian Tambah Propinsi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data propinsi, Contoh :	Jika *field* tidak terisi atau salah	Dapat menampilkan	[√] Diterima [ ] Ditolak

nama propinsi : kalimantan barat	maka muncul pesan kesalahan bahwa *field* harus diisi	pesan kesalahan bahwa *field* tersebut harus diisi	
Field data warna semua terisi	Data warna akan tersimpan dalam data warna	Data warna akan tersimpan dalam data warna	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

28. Pengujian ubah propinsi

Berikut ini merupakan pengujian ubah data propinsi dapat dilihat pada tabel 4.32 dibawah ini.

Tabel 4.32 Pengujian Ubah Propinsi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data propinsi, Contoh : nama propinsi : kalimantan barat	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field di form ubah propinsi semua terisi	Data propinsi akan tersimpan dalam data propinsi	Data propinsi akan tersimpan dalam data propinsi	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

29. Pengujian tambah kab/kota

Berikut ini merupakan pengujian tambah data kab/kota dapat dilihat pada tabel 4.33 dibawah ini.

Tabel 4.33 Pengujian Tambah Kab/kota

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data kab/kota, Contoh : nama propinsi : kalimantan barat nama kab/kota : pontianak	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field data kab/kota semua terisi	Data kab/kota akan tersimpan dalam data kab/kota	Data kab/kota akan tersimpan dalam data kab/kota	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

30. Pengujian ubah kab/kota

Berikut ini merupakan pengujian ubah data kab/kota dapat dilihat pada tabel 4.34 dibawah ini.

Tabel 4.34 Pengujian Ubah Kab/kota

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data kab/kota, Contoh : nama propinsi : kalimantan barat nama kab/kota : pontianak	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field di form ubah kab/kota semua	Data kab/kota akan tersimpan	Data kab/kota akan tersimpan	[√] Diterima [ ] Ditolak

terisi	dalam data kab/kota	dalam data kab/kota	
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

31. Pengujian tambah jenis pengiriman

Berikut ini merupakan pengujian tambah data jenis pengiriman dapat dilihat pada tabel 4.35 dibawah ini.

Tabel 4.35 Pengujian Tambah Jenis Pengiriman

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data jenis pengiriman, Contoh : nama jenis pengiriman : jne reguler	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field data jenis pengiriman semua terisi	Data jenis pengiriman akan tersimpan dalam data jenis pengiriman	Data jenis pengiriman akan tersimpan dalam data jenis pengiriman	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

32. Pengujian ubah jenis pengiriman

Berikut ini merupakan pengujian ubah data jenis pengiriman dapat dilihat pada tabel 4.36 dibawah ini.

Tabel 4.36 Pengujian Ubah Jenis Pengiriman

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data jenis pengiriman, Contoh : nama jenis pengiriman : jne reguler	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field di form ubah jenis pengiriman semua terisi	Data jenis pengiriman akan tersimpan dalam data jenis pengiriman	Data jenis pengiriman akan tersimpan dalam data jenis pengiriman	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

33. Pengujian tambah ongkos kirim

Berikut ini merupakan pengujian tambah data ongkos kirim dapat dilihat pada tabel 4.37 dibawah ini.

Tabel 4.37 Pengujian Tambah Ongkos Kirim

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data jenis pengiriman, Contoh : nama jenis pengiriman : jne regular nama kab/kota : sumedang biaya : 5.000	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field data ongkos kirim semua terisi	Data ongkos kirim akan tersimpan dalam data ongkos kirim	Data ongkos kirim akan tersimpan dalam data ongkos kirim	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

34. Pengujian ubah ongkos kirim

Berikut ini merupakan pengujian ubah data ongkos kirim dapat dilihat pada tabel 4.38 dibawah ini.

Tabel 4.38 Pengujian Ubah Ongkos Kirim

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data jenis pengiriman, Contoh : nama jenis pengiriman : jne regular nama kab/kota : sumedang biaya : 5.000	Jika *field* tidak terisi atau salah maka muncul pesan kesalahan bahwa *field* harus diisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Field di form ubah ongkos kirim semua terisi	Data ongkos kirim akan tersimpan dalam data ongkos kirim	Data ongkos kirim akan tersimpan dalam data ongkos kirim	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi salah satu *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*Field* Harus Diisi"	[√] Diterima [ ] Ditolak

4.4.1.2 Kesimpulan Hasil Pengujian Alpha

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

4.4.2 Pengujian Beta

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

4.4.2.1 Skenario Pengujian Beta

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

1. Lingkungan Perusahaan

   Penelitian di lingkungan perusahaan dilakukan agar dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan di awal.

2. Lingkungan Masyarakat Umum

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

   Adapun beberapa poin yang diteliti di lingkungan perusahaan adalah :

   1. Sistem promosi produk CV. Elvana.
   2. Fungsional aplikasi *e-commerce* di CV. Elvana.
   3. Pengolahan data produk.

4. Pengolahan data transaksi penjualan
5. Pengolahan laporan produk, pemesanan dan penjualan.
6. Tampilan *website* CV. Elvana.

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

Keterangan :

P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai persentase

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 6 orang karyawan di CV. Elvana.

1. Apakah menurut anda, dengan adanya aplikasi *e-commerce* ini memudahkan perusahaan dalam mempromosikan dan mengenalkan produk-produknya ke pelanggan ?

Tabel 4.39 Hasil Pengujian Kuisioner Perusahaan Soal No.1

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Membantu	3	50%
	2	Membantu	1	17%
	3	Cukup Membantu	2	33%
	4	Kurang Membantu	0	0%
	5	Tidak Membantu	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 3 orang atau 50% mengatakan bahwa aplikasi yang dibuat sangat membantu, 1 orang atau 17% mengatakan bahwa aplikasi yang dibuat membantu,

2 orang atau 33% mengatakan bahwa aplikasi yang dibuat cukup membantu. Sehingga secara keseluruhan aplikasi tersebut sangat membantu dalam mempromosikan produk-produknya ke pelanggan.

2. Apakah anda dapat memahami berbagai fungsional (menu,tombol) yang ada dalam aplikasi *e-commerce* ini ?

Tabel 4.40 Hasil Pengujian Kuisioner Perusahaan Soal No.2

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Memahami	0	0%
	2	Dapat Memahami	4	67%
	3	Cukup Memahami	0	0%
	4	Kurang Memahami	2	33%
	5	Tidak Memahami	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 4 orang atau 67% mengatakan bahwa aplikasi yang dibuat dapat dipahami, 2 orang atau 33% mengatakan bahwa aplikasi yang dibuat kurang untuk dipahami. Sehingga secara keseluruhan struktur aplikasi tersebut dapat dipahami.

3. Apakah menurut anda, aplikasi *e-commerce* ini dapat membantu perusahaan dalam melakukan pengolahan data produk dan data-data master lainnya ?

Tabel 4.41 Hasil Pengujian Kuisioner Perusahaan Soal No.3

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat Membantu	3	50%
	2	Membantu	2	33%
	3	Cukup Membantu	1	17%
	4	Kurang Membantu	0	0%
	5	Tidak Membantu	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 3 orang atau 50% mengatakan bahwa aplikasi yang dibuat sangat membantu, 2 orang atau 33% mengatakan bahwa aplikasi yang dibuat membantu, 1 orang atau 17% mengatakan bahwa aplikasi yang dibuat cukup membantu. Sehingga secara keseluruhan aplikasi tersebut sangat membantu dalam melakukan pengolahan data produk.

4. Apakah menurut anda, aplikasi *e-commerce* ini dapat membantu perusahaan dalam melakukan pengolahan data transaksi penjualan secara *online* ?

Tabel 4.42 Hasil Pengujian Kuisioner Perusahaan Soal No.4

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
4	1	Sangat Membantu	1	17%
	2	Membantu	3	50%
	3	Cukup Membantu	2	33%
	4	Kurang Membantu	0	0%
	5	Tidak Membantu	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 1 orang atau 17% mengatakan bahwa aplikasi yang dibuat cukup membantu, 3 orang atau 50% mengatakan bahwa aplikasi yang dibuat membantu, 2 orang atau 33% mengatakan bahwa aplikasi yang dibuat cukup membantu. Sehingga secara keseluruhan aplikasi tersebut membantu dalam melakukan pengolahan transaksi penjualan.

5. Apakah menurut anda, aplikasi *e-commerce* ini dapat membantu anda dalam menghasilkan sebuah laporan, baik itu laporan produk, laporan pemesanan dan laporan penjualan ?

Tabel 4.43 Hasil Pengujian Kuisioner Perusahaan Soal No.5

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
5	1	Sangat Membantu	2	33%
	2	Membantu	4	67%
	3	Cukup Membantu	0	0%
	4	Kurang Membantu	0	0%
	5	Tidak Membantu	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 33% mengatakan bahwa aplikasi yang dibuat cukup membantu, 4 orang atau 67% mengatakan bahwa aplikasi yang dibuat membantu. Sehingga secara keseluruhan aplikasi tersebut membantu dalam menghasilkan laporan produk, laporan pemesanan dan laporan penjualan.

6. Apakah menurut anda tampilan *backend* aplikasi *e-commerce* ini cukup menarik dan mudah untuk dipahami ?

Tabel 4.44 Hasil Pengujian Kuisioner Perusahaan Soal No.6

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
6	1	Sangat menarik dan sangat mudah dipahami	0	0%
	2	Menarik dan dapat dipahami	3	50%
	3	Cukup menarik dan cukup mudah untuk dipahami	3	50%
	4	Kurang menarik dan agak sulit dipahami	0	0%
	5	Tidak menarik dan sulit untuk dipahami	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 3 orang atau 50% mengatakan bahwa aplikasi yang dibuat menarik dan dapat dipahami, 3 orang atau 50% mengatakan bahwa aplikasi yang dibuat cukup menarik dan cukup mudah untuk dipahami. Sehingga secara keseluruhan struktur aplikasi tersebut cukup menarik dan cukup mudah untuk digunakan.

Sedangkan beberapa poin yang diteliti di lingkungan masyarakat umum adalah :

1. Mudah dalam melakukan pemesanan dan pembelian.
2. Dapat memenuhi kebutuhan informasi kepada pelanggan.
3. Memudahkan pelanggan yang berada di kota Bandung maupun di luar kota Bandung dalam melakukan transaksi secara *online*.
4. Pelanggan mengetahui produk-produk dari CV. Elvana.
5. Proses pemesanan dan proses pembayaran.
6. Tampilan *website* CV. Elvana.

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

Keterangan :

P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai persentase

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 6 orang di lingkungan masyarakat umum.

1. Apakah menurut anda, dengan adanya aplikasi *e-commerce* ini memudahkan anda dalam melakukan pemesanan dan pembelian produk-produk CV. Elvana ?

Tabel 4.45 Hasil Pengujian Kuisioner Masyarakat Umum Soal No.1

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Membantu	5	83%
	2	Membantu	1	17%
	3	Cukup Membantu	0	0%
	4	Kurang Membantu	0	0%
	5	Tidak Membantu	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 5 orang atau 83% mengatakan bahwa aplikasi yang dibuat sangat membantu, 1 orang atau 17% mengatakan bahwa aplikasi yang dibuat membantu. Sehingga secara keseluruhan aplikasi tersebut sangat membantu dalam melakukan pemesanan dan pembelian produk.

2. Bagaimana menurut anda, mengenai kelengkapan informasi yang diberikan melalui e-mail ataupun informasi yang ada di aplikasi *e-commerce* ini ?

Tabel 4.46 Hasil Pengujian Kuisioner Masyarakat Umum Soal No.2

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Lengkap	0	0%
	2	Lengkap	4	67%
	3	Cukup Lengkap	2	33%
	4	Kurang Lengkap	0	0%
	5	Tidak Lengkap	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 4 orang atau 67% mengatakan bahwa aplikasi yang dibuat lengkap, 2 orang atau 33% mengatakan bahwa aplikasi yang dibuat cukup lengkap. Sehingga

secara keseluruhan aplikasi tersebut lengkap dalam memberikan informasi ke pelanggan.

3. Apakah anda setuju, dengan adanya aplikasi *e-commerce* ini dapat membantu para calon pelanggan lainnya yang berada di kota Bandung maupun yang berada di luar kota Bandung ?

Tabel 4.47 Hasil Pengujian Kuisioner Masyarakat Umum Soal No.3

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	2	33%
	2	Setuju	4	67%
	3	Cukup Setuju	0	0%
	4	Kurang Setuju	0	0%
	5	Tidak Setuju	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 33% mengatakan sangat setuju, 4 orang atau 67% mengatakan setuju. Sehingga secara keseluruhan mengatakan setuju bahwa *website* ini membantu pelanggan yang berada di kota Bandung atau yang berada di luar kota Bandung dalam melakukan transaksi secara *online*.

4. Apakah menurut anda, promosi dan pengenalan produk melalui aplikasi *e-commerce* ini membantu anda dalam mengetahui produk yang promosi atau dalam mengenal produk-produk CV. Elvana ?

Tabel 4.48 Hasil Pengujian Kuisioner Masyarakat Umum Soal No.4

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
4	1	Sangat Membantu	2	33%
	2	Membantu	3	50%
	3	Cukup Membantu	1	17%
	4	Kurang Membantu	0	0%
	5	Tidak Membantu	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 33% mengatakan sangat membantu, 3 orang atau 50% mengatakan membantu, 1 orang atau 17% mengatakan cukup membantu. Sehingga secara keseluruhan aplikasi tersebut membantu pelanggan dalam mengetahui produk-produk CV. Elvana secara detail.

5. Bagaimana pendapat anda mengenai proses transaksi yang ada di aplikasi *e-commerce* ini, dari mulai proses pemesanan sampai dengan proses pembayaran ?

Tabel 4.49 Hasil Pengujian Kuisioner Masyarakat Umum Soal No.5

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
5	1	Sangat Mudah	2	33%
	2	Mudah	3	50%
	3	Cukup Mudah	1	17%
	4	Kurang Mudah	0	0%
	5	Tidak Mudah (sulit)	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 33% mengatakan sangat mudah, 3 orang atau 50% mengatakan mudah, 1 orang atau 17% mengatakan cukup mudah. Sehingga secara keseluruhan aplikasi tersebut mudah dalam melakukan pemesanan sampai dengan proses pembayaran.

6. Apakah menurut anda tampilan aplikasi *e-commerce* ini cukup menarik dan mudah untuk dipahami ?

Tabel 4.50 Hasil Pengujian Kuisioner Masyarakat Umum Soal No.6

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
6	1	Sangat menarik dan sangat mudah dipahami	3	50%
	2	Menarik dan dapat dipahami	1	17%
	3	Cukup menarik dan cukup mudah untuk dipahami	2	33%
	4	Kurang menarik dan agak sulit dipahami	0	0%
	5	Tidak menarik dan sulit untuk dipahami	0	0%
			6	100%

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 3 orang atau 50% mengatakan bahwa aplikasi yang dibuat sangat menarik dan sangat mudah dipahami, 1 orang atau 17% mengatakan bahwa aplikasi yang dibuat menarik dan dapat dipahami, 2 orang atau 33% mengatakan bahwa aplikasi yang dibuat cukup menarik dan cukup mudah dipahami. Sehingga secara keseluruhan struktur aplikasi tersebut sangat menarik dan sangat mudah untuk digunakan.

4.4.2.2 Kesimpulan Hasil Pengujian Beta

Berdasarkan pengujian beta yang dilakukan di lingkungan perusahaan dan di lingkungan masyarakat umum, maka dapat diambil kesimpulan bahwa :

1. Aplikasi ini dapat memberi kemudahan pada pengolahan data master dan pengolahan data transaksi penjualan produk serta pembuatan laporan produk, laporan pesanan maupun laporan penjualan.
2. Tampilan website CV. Elvana ini menarik dan mudah untuk dipahami baik dari situs *backend* maupun *frontend*.
3. *Website* CV. Elvana ini sudah cukup lengkap dalam memberikan informasi-informasi ke pelanggan.
4. CV. Elvana dapat dengan mudah mempromosikan produk-produk terbarunya ke pelanggan melalui *website* ini.
5. Website CV. Elvana ini membantu para pelanggan yang berada di kota Bandung maupun yang berada di luar kota bandung dalam melakukan transaksi secara *online*.