BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Tahapan implementasi merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan tahap dimana sistem siap untuk dioperasikan yang terdiri dari penjelasan mengenai lingkungan implementasi dan implementasi program.

4.1.1 Lingkungan Implementasi

Untuk mendukung aplikasi yang akan diterapkan pada lingkungan implementasi, maka dalam hal ini harus memperhatikan perangkat keras dan perangkat lunak yang menunjang dalam pemangunan aplikasi *E-Commerce*.

4.1.2 Implementasi Perangkat Keras

Spesifikasi perangkat keras minimal yang dibutuhkan untuk mengimplementasikan perangkat keras antara lain terdapat pada tabel 4.1 sebagai berikut.

Tabel 4.1 Perangkat Keras Pembangun

No	Perangkat Keras	Spesifikasi
1	Processor	Intel Pentium 4 2.0 GHz
2	Monitor	Monitor 15 inch
3	Memory	DDR2 512Mb
4	Koneksi internet	384 Kbps
5	Keyboard	Standar
6	Mouse	Standar
7	Printer	-

4.1.3 Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang dibutuhkan untuk mengimplementasikan perangkat lunak antara lain terdapat pada tabel 4.2 sebagai berikut

Tabel 4.2 Perangkat Lunak Pembangun

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows7
2	Web Browser	Mozilla Firefox

4.1.4 Implementasi Basis Data

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

Tabel barang

```
CREATE TABLE `barang`
(
      `id_barang` varchar(10) NOT NULL,
      `nama_barang` varchar(30) NOT NULL,
      `detail` varchar(160) NOT NULL,
      `nama_gbr` varchar(30) NOT NULL,
      `dilihat` int(11) NOT NULL DEFAULT '0',
      `tgl_upload` datetime NOT NULL,
      `id_kategori` varchar(10) NOT NULL,
      `username` varchar(30) DEFAULT NULL,
      PRIMARY KEY (`id_barang`),
      CONSTRAINT  `barang_ibfk_1`  FOREIGN  KEY  (`id_kategori`)
      REFERENCES `kategori` (`id_kategori`),
      CONSTRAINT   `barang_ibfk_2`   FOREIGN   KEY   (`username`)
      REFERENCES `petugas` (`username`)
)
ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel **detail_retur**

```
CREATE TABLE `detail_retur`
      (
      `id_detail_retur` int(11) NOT NULL AUTO_INCREMENT,
      `id_retur` int(11) NOT NULL,
      `id_detailpembelian` int(11) NOT NULL,
      `jumlah` int(11) NOT NULL,
      `alasan` varchar(160) DEFAULT NULL,
      PRIMARY KEY (`id_detail_retur`),
      CONSTRAINT  `detail_retur_ibfk_1`  FOREIGN  KEY  (`id_retur`)
REFERENCES `retur` (`id_retur`),
      CONSTRAINT      `detail_retur_ibfk_2`      FOREIGN      KEY
(`id_detailpembelian`)        REFERENCES        `detailpembelian`
(`id_detailpembelian`)
      )
ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel kategori

```
CREATE TABLE IF NOT EXISTS `kategori`
(
      `id_kategori` varchar(10) NOT NULL,
      `nama_kategori` varchar(30) NOT NULL,
      PRIMARY KEY  (`id_kategori`)
)
```

```
ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel detailpembelian

```
CREATE TABLE IF NOT EXISTS `detailpembelian` (
      `id_detailpembelian` int(11) NOT NULL AUTO_INCREMENT,
      `qty` int(11) NOT NULL,
      `id_pembelian` int(11) default NULL,
      `id_detailbarang` int(10) NOT NULL,
      PRIMARY KEY  (`id_detailpembelian`),
      CONSTRAINT     `detailpembalian_ibfk_1`     FOREIGN     KEY
(`id_pembelian`) REFERENCES pembelian(`id_pembelian`),
      CONSTRAINT     `detailpembelian_ibfk_2`     FOREIGN     KEY
(`id_detailbarang`) REFERENCES detailbarang (`id_detailbarang`)
      )
ENGINE=InnoDB  DEFAULT CHARSET=utf8;
```

Tabel komentar

```
CREATE TABLE IF NOT EXISTS `komentar` (
      `nomor` int(11) NOT NULL AUTO_INCREMENT,
      `komentar` varchar(160) NOT NULL,
      `status` varchar(30) NOT NULL,
      `id_member` varchar(30) NOT NULL,
      `id_pembelian` varchar(10) NOT NULL,
      `username` varchar(30) default NULL,
      PRIMARY KEY  (`nomor`),
      CONSTRAINT   `komentar_ibfk_1`   FOREIGN   KEY   (`id_member`)
REFERENCES member (`id_member`),
      CONSTRAINT  `komentar_ibfk_2`  FOREIGN  KEY  (`id_pembelian`)
REFERENCES pembelian (`id_pembelian`),
      CONSTRAINT   `komentar_ibfk_3`   FOREIGN   KEY   (`username`)
REFERENCES petugas(`username`)
      )
ENGINE=InnoDB  DEFAULT CHARSET=utf8;
```

Tabel kota

```
CREATE TABLE IF NOT EXISTS `kota` (
      `id_kota` int(11) NOT NULL AUTO_INCREMENT,
      `nama_kota` varchar(30) NOT NULL,
      `id_provinsi` int(11) NOT NULL,
      `ket` varchar(10) NOT NULL,
       PRIMARY KEY  (`id_kota`),
       CONSTRAINT   `kota_ibfk_1`   FOREIGN   KEY   (`id_provinsi`)
REFERENCES `provinsi` (`id_provinsi`)
)ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel member

```
      CREATE TABLE IF NOT EXISTS `member` (
        `id_member` varchar(30) NOT NULL,
        `password` varchar(50) NOT NULL,
        `nama` varchar(30) NOT NULL,
        `alamat` varchar(160) NOT NULL,
        `kodepos` varchar(5) NOT NULL,
        `no_tlp` varchar(15) default NULL,
        `aktivasi` varchar(100) NOT NULL,
        `status` varchar(15) NOT NULL default 'Tidak Aktif',
        `tgl_daftar` varchar(30) NOT NULL,
        `id_kota` int(11) NOT NULL,
        `username` varchar(30) NOT NULL,
       PRIMARY KEY  (`id_member`),
       CONSTRAINT   `member_ibfk_1`   FOREIGN   KEY   (`id_kota`)
```

```
REFERENCES kota(`id_kota`),
        CONSTRAINT   `member_ibfk_2`   FOREIGN   KEY   (`username`)
REFERENCES petugas(`username`)
      ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel pembayaran

```
CREATE TABLE IF NOT EXISTS `pembayaran` (
        `no_pembayaran` int(11) NOT NULL AUTO_INCREMENT,
        `id_pembelian` int(11) NOT NULL,
        `no_rekening` varchar(20) NOT NULL,
        `tgl_bayar` datetime NOT NULL,
        PRIMARY KEY  (`no_pembayaran`),
        CONSTRAINT      `pembayaran_ibfk_1`      FOREIGN      KEY
(`id_pembelian`) REFERENCES pembelian (`id_pembelian`)
      ) ENGINE=InnoDB  DEFAULT CHARSET=utf8;
```

Tabel pembelian

```
CREATE TABLE IF NOT EXISTS `pembelian` (
        `id_pembelian` int(11) NOT NULL auto_increment,
        `tgl_beli` datetime NOT NULL,
        `total_bayar` int(11) NOT NULL,
        `penerima` varchar(30) NOT NULL,
        `alamat_kirim` varchar(160) NOT NULL,
        `kodepos` varchar(10) NOT NULL,
        `status_bayar`   varchar(20)   NOT   NULL   default   'Belum
Dibayar',
        `status_kirim`   varchar(20)   NOT   NULL   default   'Belum
Dikirim',
        `tgl_kirim` datetime default NULL,
        `penerima_barang` varchar(100) default NULL,
        `tgl_terima` datetime default NULL,
        `no_resi` varchar(25) default NULL,
        `id_paket` int(11) NOT NULL,
        `id_member` varchar(30) default NULL,
        `id_ongkos` int(11) NOT NULL,
        PRIMARY KEY  (`id_pembelian`),
        CONSTRAINT  `pembelian_ibfk_1`  FOREIGN  KEY  (`id_paket`)
REFERENCES paket (`id_paket`),
        CONSTRAINT  `pembelian_ibfk_2`  FOREIGN  KEY  (`id_member`)
REFERENCES member (`id_member`),
        CONSTRAINT  `pembelian_ibfk_3`  FOREIGN  KEY  (`id_ongkos`)
REFERENCES ongkir (`id_ongkos`)
) ENGINE=InnoDB  DEFAULT CHARSET=utf8;
```

Tabel detailbarang

```
      CREATE TABLE IF NOT EXISTS `detailbarang` (
      `id_detailbarang` int(11) NOT NULL AUTO_INCREMENT,
      `id_barang` varchar(10) NOT NULL,
      `harga` int(11) NOT NULL,
      `diskon` float NOT NULL,
      `berat` float NOT NULL,
      `stok` int(11) NOT NULL,
      `id_ukuran` int(11) NOT NULL,
       PRIMARY KEY (`id_detailbarang`),
      CONSTRAINT  `detailbarang_ibfk_1`  FOREIGN  KEY  (`id_barang`)
REFERENCES barang (`id_barang`),
      CONSTRAINT  `detailbarang_ibfk_2`  FOREIGN  KEY  (`id_ukuran`)
```

```
REFERENCES ukuran(`id_ukuran`)
      ) ENGINE=InnoDB AUTO_INCREMENT=12 DEFAULT CHARSET=utf8;
```

Tabel jasa

```
      CREATE TABLE IF NOT EXISTS `jasa` (
       `id_jasa` int(11) NOT NULL AUTO_INCREMENT,
       `jasa` varchar(10) CHARACTER SET utf8 NOT NULL,
      PRIMARY KEY (`id_jasa`)
)ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Tabel rating

```
CREATE TABLE IF NOT EXISTS `rating` (
      `id_rating` varchar(10) NOT NULL,
      `rating` varchar(10) NOT NULL,
      `id_pembelian` varchar(10) NOT NULL,
      `id_member` varchar(10) NOT NULL,
        PRIMARY KEY (`id_rating`),
      CONSTRAINT  `rating_ibfk_1`  FOREIGN  KEY  (`id_pembelian`)
REFERENCES pembelian(`id_pembelian`),
      CONSTRAINT   `rating_ibfk_2`   FOREIGN   KEY   (`id_member`)
REFERENCES member (`id_member`)
      ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel ongkir

```
CREATE TABLE IF NOT EXISTS `ongkir` (
        `id_ongkos` int(11) NOT NULL AUTO_INCREMENT,
        `ongkos_kirim` int(11) NOT NULL,
        `id_kota` int(11) NOT NULL,
        `id_paket` int(11) NOT NULL,
        PRIMARY KEY (`id_ongkos`),
       CONSTRAINT   `ongkir_ibfk_1`   FOREIGN   KEY   (`id_paket`)
REFERENCES `paket` (`id_paket`),
      CONSTRAINT   `ongkir_ibfk_2`   FOREIGN   KEY   (`id_kota`)
REFERENCES `kota` (`id_kota`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel paket

```
CREATE TABLE IF NOT EXISTS `paket` (
       `id_paket` int(11) NOT NULL AUTO_INCREMENT,
       `nama_paket` varchar(20) NOT NULL,
       `id_jasa` int(11) NOT NULL,
       PRIMARY KEY (`id_paket`),
       CONSTRAINT `paket_ibfk_1` FOREIGN KEY (`id_jasa`)
     REFERENCES `jasa` (`id_jasa`)
     ) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel provinsi

```
CREATE TABLE IF NOT EXISTS `provinsi` (
        `id_provinsi` int(11) NOT NULL AUTO_INCREMENT,
        `nama_prov` varchar(30) NOT NULL,
        PRIMARY KEY (`id_provinsi`)
     )
ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel petugas

```
CREATE TABLE IF NOT EXISTS `petugas` (
        `username` varchar(30) NOT NULL,
        `password` varchar(50) NOT NULL,
        `nama_petugas` varchar(30) NOT NULL,
        `email` varchar(30) NOT NULL,
```

```
        `alamat` varchar(160) NOT NULL,
        `telepon` varchar(15) NOT NULL,
        `bagian` varchar(10) NOT NULL,
        PRIMARY KEY (`username`)
      )
ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel retur

```
CREATE TABLE IF NOT EXISTS `retur` (
        `id_retur` int(11) NOT NULL AUTO_INCREMENT,
        `tgl_retur` datetime NOT NULL,
        `tgl_kirim` datetime DEFAULT NULL,
        `no_resi` varchar(15) DEFAULT NULL,
        `status` varchar(20) NOT NULL DEFAULT 'Retur',
        `alasan_invalid` varchar(160) NOT NULL DEFAULT 'Tidak
Ada',
        `id_pembelian` int(11) NOT NULL,
        PRIMARY KEY (`id_retur`),
        CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`id_pembelian`)
REFERENCES `pembelian` (`id_pembelian`)
)ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Tabel ukuran

```
CREATE TABLE IF NOT EXISTS `ukuran` (
        `id_ukuran` int(11) NOT NULL AUTO_INCREMENT,
        `ukuran` varchar(10) NOT NULL,
        PRIMARY KEY (`id_ukuran`)
)
ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

4.1.5 Implementasi Antarmuka

Implementasi antarmuka merupakan tampilan dari aplikasi yang dibangun. Aplikasi pengolahan data proses memilki beberapa bagian tampilan menu dapat dilihat pada tabel berikut :

Tabel 4.4 Implementasi Antarmuka Pengunjung

Menu	Deskripsi	Nama File
Beranda	Digunakan sebagai halaman beranda dan terdapat beberapa menu diantaranya menu produk, keranjang, cari dan daftar	index.php
Informasi	Berisi mengenai semua informasi yaitu informasi seputar perusahaan, aturan bisnis, aturan retur, aturan pengiriman, pembayaran, pesanan.	tentang.php
Semua produk	Halaman yang digunakan untuk menampilkan semua data produk yang tersedia berdasarkan kategori.	produk.php

Detail Produk	Halaman yang digunakan untuk menampilkan data detail produk yang dipilih.	detailproduk.php
Keranjang Anda	Halaman yang digunakan untuk menampilkan dan mengubah data produk yang akan dibeli.	keranjang_belanja.php
Cari	Halaman yang digunakan untuk mencari data produk yang tersedia.	cariproduk.php
Lupa *Password*	Halaman yang digunakan untuk mengirimkan *password* baru ke member	lupa*password*.php
Daftar	Halaman yang digunakan untuk mendaftarkan diri menjadi member	register.php

Tabel 4.5 Implementasi Antarmuka Member

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login member	login.php
Informasi	Berisi mengenai semua informasi yaitu informasi seputar perusahaan, aturan bisnis, aturan retur, aturan pengiriman, pembayaran, pesanan.	tentang.php
Beranda	Digunakan sebagai halaman beranda pelanggan dan menampilkan beberapa produk terbaru, produk banyak dilihat	index.php
Ganti *Password*	Halaman yang digunakan untuk mengganti *password* member.	ganti*password*.php
History Pembelian	Halaman yang digunakan untuk menampilkan data pembelian yang telah dilakukan	history.php
Semua Produk	Halaman yang digunakan untuk menampilkan data produk yang tersedia	produk.php
Detail Produk	Halaman yang digunakan untuk menampilkan data detail produk yang dipilih	detailproduk.php
Keranjang	Halaman yang digunakan untuk menampilkan dan mengubah data produk yang akan dibeli	keranjang_belanja.php
Data Pengiriman	Halaman yang digunakan untuk memasukan data alamat kirim barang	alamat_kirim.php
Pencarian	Halaman yang digunakan untuk mencari data produk yang tersedia	cariproduk.php
Lupa *Password*	Halaman yang digunakan untuk mengirimkan *password* baru ke member.	lupa*password*.php
Logout	Digunakan jika member akan keluar dari aplikasi.	logout.php

Tabel 4.6 Implementasi Antarmuka Admin

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login petugas	login.php

Beranda	Digunakan sebagai halaman beranda admin dan terdapat beberapa menu diantaranya menu data master, menu data user, menu backup.	index.php
Data Master	Halaman yang digunakan untuk mengolah data master diantaranya data kategori, data kota, data ongkos kirim, data provinsi, data kota, data paket, dan data jasa.	master.php
Data User	Digunakan untuk mengolah data member dan data petugas.	datauser.php
Data Petugas	Digunakan untuk mengolah data petugas seperti tambah petugas dan edit data petugas.	petugas.php
Data Member	Digunakan untuk menampilkan data member.	member.php
Backup & Restore	Digunakan untuk membuat cadangan database.	databackup.php
Logout	Digunakan admin untuk keluar dari aplikasi halaman admin.	logout.php

Tabel 4.6 Implementasi Antarmuka Pemilik

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login pemilik	login.php
Laporan Data Pesanan	Digunakan untuk mencetak data laporan data pesanan.	laporan_pesanan.php
Laporan Data Pembayaran	Digunakan untuk mencetak data laporan data pembayaran.	laporan_pembayaran.php
Laporan Data Retur	Digunakan untuk mencetak data laporan data retur.	laporan_retur.php
Laporan Data Barang	Digunakan untuk mencetak data laporan data barang.	laporan_produk.php
Laporan Data Penjualan	Digunakan untuk mencetak data laporan data penjualan.	laporan_penjualan.php
Logout	Digunakan jika member akan keluar dari aplikasi.	logout.php

Tabel 4.7 Implementasi Antarmuka Bagian Produksi

Menu	Deskripsi	Nama File
Login	Digunakan sebagai halaman login bagian produksi	login.php
Beranda	Digunakan sebagai halaman beranda petugas bagian produksi dan terdapat beberapa menu diantaranya menu data produk dan data laporan.	index.php
Data Produk	Digunakan untuk melihat data barang, mengolah data produk dimana diantaranya terdapat menu data barang dan tambah data barang.	dataproduk.php

Data Laporan	Digunakan untuk mengakses data laporan yaitu laporan data barang dan laporan penjualan.	datalaporan.php
Data barang	Digunakan untuk mengolah data barang seperti tambah, edit, hapus.	produk.php
Tambah barang	Digunakan untuk menambah data barang.	produk_form_tambah.php
Ubah Barang	Halaman yang digunakan untuk mengubah data barang	produk _form_edit.php
Hapus Barang	Halaman yang digunakan untuk menghapus data barang	produk_hapus.php
Laporan Data Barang	Digunakan untuk mencetak data laporan data barang.	laporan_produk.php
Laporan Penjualan	Digunakan untuk mencetak data laporan penjualan.	laporan_penjualan.php
Logout	Digunakan jika petugas akan keluar dari sistem.	logout.php

Tabel 4.8 Implementasi Antarmuka Bagian Keuangan

		transaksi.php
Login	Digunakan sebagai halaman login bagian keuangan.	login.php
Beranda	Digunakan sebagai halaman beranda petugas bagian keuangan dan terdapat beberapa menu diantaranya menu data transaksi dan data laporan.	index.php
Data Transaksi	Halaman yang digunakan untuk menampilkan dan mengolaha data transaksi yang diantaranya data pesanan, data pembayaran, data pengiriman dan data retur.	
Data Pesanan	Halaman yang digunakan untuk menampilkan data pesanan.	pesanan.php
Data Pembayaran	Halaman yang digunakan untuk menampilkan data pembayaran.	lihatpembayaran.php
Data Pengiriman	Halaman yang digunakan untuk menampilkan data pengiriman.	lihatkirim.php
Data Retur	Halaman yang digunakan untuk menampilkan data retur.	Retur.php
Data Laporan	Halaman yang digunakan untuk memberikan informasi mengenai laporan pesanan, laporan pembayaran, dan laporan retur.	laporan_kasir.php
Laporan pesanan	Halaman yang digunakan untuk mengakses laporan pesanan.	laporan_pesanan.php

Laporan pembayaran	Halaman yang digunakan untuk mengakses laporan pembayaran.	laporan_pembayaran.php
Logout	Digunakan petugas untuk keluar dari aplikasi.	Logout.php

4.2 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.

4.2.1 Pengujian Alpha

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

4.2.1.1 Skenario Pengujian Alpha

Skenario pengujian alpha di Sifa *Collection* terdiri dari empat pelaku sistem diantaranya skenario pengujian aplikasi pengunjung dan member, skenarion pengujian aplikasi pemilik, skenario pengujian aplikasi admin, skenario pengujian aplikasi petugas bagian produksi, dan skenario pengujian aplikasi petugas bagian keuangan.

Tabel 4.10 Skenario Pengujian Aplikasi Pengunjung dan Member

Kelas Uji	Butir Uji	Jenis Pengujian
Login member	Isi data login	*Black Box*
	Verifikasi *e-mail*	*Black Box*
	Verifikasi *password*	*Black Box*
Daftar	Menampilkan *form* daftar member	*Black Box*
	Isi *form* pendaftaran	*Black Box*
	Simpan data baru member baru	*Black Box*
Lupa *password*	Menampilkan form lupa *password*	*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*
Menu Cari	Menampilkan Halaman pencarian.	*Black Box*
Menu Keranjang	Menampilkan data pembelian	*Black Box*
	Memperbaharui data pembelian	*Black Box*
	Menghapus data pembelian	*Black Box*
Tujuan Pengiriman	Menampilkan form alamat tujuan pengiriman.	*Black Box*
Mengisi form pembayaran	Menampilkan form pembayaran	*Black Box*
Logout	Menghapis session member	*Black Box*

Tabel 4.11 Skenario Pengujian Aplikasi Pemilik

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data login	*Black Box*
	Verifikasi *e-mail*	*Black Box*
	Verifikasi *password*	*Black Box*
Laporan Pesanan	Menampilkan data pesanan	*Black Box*
	Cari data pesanan	*Black Box*
	Cetak laporan data pesanan	*Black Box*
Laporan Pembayaran	Menampilkan data pembayaran	*Black Box*
	Pilih periodik laporan data pembayaran	*Black Box*
	Cetak laporan data pembayaran	*Black Box*
Laporan Retur	Menampilkan data retur	*Black Box*
	Pilih periodik laporan data retur	*Black Box*

	Cetak laporan data retur	*Black Box*
Laporan Data Barang	Menampilkan data barang	*Black Box*
	Pilih periodik laporan data barang	*Black Box*
	Cetak laporan data barang	*Black Box*
Laporan Penjualan	Menampilkan data penjualan	*Black Box*
	Pilih periodik laporan data penjualan	*Black Box*
	Cetak laporan data penjualan	*Black Box*
Logout	Menghapus *session* pemilik	*Black Box*

Tabel 4.12 Skenario Pengujian Aplikasi Admin

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data Login	*Black Box*
	Validasi *usernameand password*	*Black Box*
Data kategori	Menampilkan data kategori	*Black Box*
	Tambah data kategori	*Black Box*
	Ubah data kategori	*Black Box*
Data paket	Menampilkan data paket	*Black Box*
	Tambah data paket	*Black Box*
	Ubah data paket	*Black Box*
	Hapus data paket	*Black Box*
Data Petugas	Tampil Data Petugas	*Black Box*
	Tambah Data Petugas	*Black Box*
	Ubah Data Petugas	*Black Box*
Data Member	Tampil Data Member	*Black Box*
	Aktivasi Status Data Member	*Black Box*
Logout	Menghapus *session* admin	*Black Box*

Tabel 4.11 Skenario Pengujian Aplikasi Bagian Produksi

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data Login	*Black Box*
	Validasi *username and password*	*Black Box*
Data Barang	Menampilkan Data Barang	*Black Box*
	Tambah Data Barang	*Black Box*
	Ubah Data Barang	*Black Box*
Detail Barang	Menampilkan data detail barang	*Black Box*
	Tambah Data Detail Barang	*Black Box*
	Ubah Data Detail Barang	*Black Box*
Laporan Barang	Menampilkan Laporan Barang	*Black Box*
	Pilih Data Periodik Laporan	*Black Box*
	Cetak Laporan Barang	*Black Box*

Laporan Penjualan	Menampilkan Laporan Penjualan	*Black Box*
	Pilih Data Periodik Laporan	*Black Box*
	Cetak Laporan Penjualan	*Black Box*

Tabel 4.12 Skenario Pengujian Aplikasi Bagian Keuangan

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data Login	*Black Box*
	Validasi *username and password*	*Black Box*
Data Pesanan	Tampilkan Data Pesanan	*Black Box*
	Kirim Pesanan	*Black Box*
Data Pembayaran	Tampilkan Data Pembayaran	*Black Box*
	Validasi Pembayaran	*Black Box*
Pengolahan Data Pengiriman	Tampilkan Data Pengiriman	*Black Box*
	Cetak Label	*Black Box*
Pengolahan Data Retur	Tampilkan Data Retur	*Black Box*
	Tampilkan Data Detail Retur	*Black Box*
Data Laporan	Laporan Pesanan	*Black Box*
	Laporan Pembayaran	*Black Box*
	Laporan Retur	*Black Box*

4.2.1.2 Kasus dan Hasil Pengujian Alpha

Berdasarkan rencana pengujian yang disusun, maka dilakukan pengujian sebagai berikut.

4.2.1.3 Pengujian Login Member

Pengujian login dapat dilihat pada tabel 4.14 :

Tabel 4.14 Pengujian Login Member

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Semua *field* diisi dengan data yang benar	Mengisikan data alamat *email* pada *field* alamat email	Dapat mengisi *field* alamat *email* pada *field* alamat *email*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
E-mail dan *password* di kosongkan	Tampilkan pesan kesalahan "*E-mail* dan	Pengguna tidak dapat login dan menampilkan	[√] Diterima [ ] Ditolak

	Password belum diisi"	pesan kesalahan "*E-mail* dan *password* belum diisi"	
E-mail di kosongkan	Tampilkan pesan kesalahan "*E-mail* belum diisi"	Pengguna tidak dapat login dan menampilkan pesan kesalahan "*E-mail* belum diisi"	[√] Diterima [ ] Ditolak
Password di kosongkan	Tampilkan pesan kesalahan "*Password* belum diisi"	Pengguna tidak dapat login dan menampilkan pesan kesalahan "*Password* belum diisi"	[√] Diterima [ ] Ditolak
E-mail dan *password* diisi dengan data yang salah	Tampilkan pesan kesalahan " *Password* yang anda masukkan salah atau *E-mail* belum terdaftar"	Pengguna tidak dapat login dan menampilkan pesan kesalahan "*Password* yang anda masukkan salah atau *E-mail* belum terdaftar"	[√] Diterima [ ] Ditolak
Mengisi data yang sudah terdaftar tetapi belum melakukan aktivasi	Menampilkan pesan kesalahan " *E-mail* belum teraktivasi. Silahkan aktifasi melalui e-mail	Dapat menampilkan pesan kesalahan " *E-mail* belum teraktivasi. Silahkan aktifasi	[√] Diterima [ ] Ditolak

	anda”	melalui e-mail anda”	

4.2.1.4 Pengujian Daftar

Pengujian daftar dapat dilihat pada tabel 4.15 :

Tabel 4.15 Pengujian Daftar

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Semua *field* diisi dengan data yang benar.	Jika semua *field* terisi dengan benar, maka pengunjung telah berhasil melakukan pendaftarn dan dilanjutkan dengan proses aktivasi di *e-mail* yang telah didaftarkan.	Registrasi berhasil dan dapat melanjutkan ke proses aktivasi.Menampilkan pesan “Registrasi berhasil.Silahkan aktivasi melalui e-mail anda”	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Semua *field* di kosongkan	Tampilkan pesan kesalahan “*E-mail* tidak boleh kosong” “*Password* tidak boleh kosong” “Nama tidak boleh kosong” “Alamat tidak	Dapat menampilkan pesan kesalahan “*E-mail* tidak boleh kosong” “*Password* tidak boleh kosong” “Nama tidak boleh kosong”	[√] Diterima [ ] Ditolak

	boleh kosong" "Kodepos tidak boleh kosong" "Kota belum dipilih" "Security image tidak sesuai"	"Alamat tidak boleh kosong" "Kodepos tidak boleh kosong" "Kota belum dipilih" "Security image tidak sesuai"	
Mengosongkan *E-mail*	Tampilkan pesan kesalahan "*E-mail* tidak boleh kosong"	Dapat menampilkan pesan kesalahan "*E-mail* tidak boleh kosong"	[√] Diterima [ ] Ditolak
Mengisi email yang sudah terdaftar Contoh : yuni.sopiawati@yahoo.com	Tampilkan pesan kesalahan "*E-mail* sudah pernah terdaftar"	Dapat menampilkan pesan kesalahan "*E-mail* sudah pernah terdaftar"	[√] Diterima [ ] Ditolak
Mengisi email dengan format tidak valid Contoh : yuni.m.yahoo	Tampilkan pesan kesalahan "*E-mail* tidak valid"	Dapat menampilkan pesan kesalahan "Salah penulisan *E-mail*"	[√] Diterima [ ] Ditolak
Mengosongkan *password*	Tampil Pesan Kesalahan " *Password* tidak boleh kosong"	Dapat menampilan pasan kesalahan "*Password* tidak boleh kosong	[√] Diterima [ ] Ditolak
Mengisi	Tampil pesan	Dapat	[√] Diterima

password kurang dari minimal. Contoh: *Password* : 121	kesalahan "*Password* minimal 6 karakter"	menampilkan pesan kesalahan "*Password* minimal 6 karakter"	[ ] Ditolak
Mengisi *password* yang berbeda dengan konfirmasi-*password* Contoh : *Password*:123456 Re-*Password* : 22222	Tampilkan pesan kesalahan "*Password* dan Konfirmasi *Password* tidak sesuai"	Dapat menampilkan pesan kesalahan "*Password* dan Konfirmasi *Password* tidak sesuai"	[√] Diterima [ ] Ditolak
Mengosongkan nama	Tampilkan pesan kesalahan "Nama tidak boleh kosong"	Dapat menampilkan pesan kesalahan "Nama tidak boleh kosong"	[√] Diterima [ ] Ditolak
Mengosongkan alamat	Tampilkan pesan kesalahan "Alamat tidak boleh kosong"	Dapat menampilkan pesan kesalahan "Alamat tidak boleh kosong"	[√] Diterima [ ] Ditolak
Pilihan kota tidak dipilih	Tampilkan pesan kesalahan "Kota belum dipilih"	Dapat menampilkan pesan kesalahan "Kota belum dipilih"	[√] Diterima [ ] Ditolak

Mengosongkan kodepos	Tampilkan pesan kesalahan "Kodepos tidak boleh kosong"	Dapat menampilkan pesan kesalahan "kodepos tidak boleh kosong	[√] Diterima [ ] Ditolak
Memasukkan kodepos yang salah Contoh : Kodepos : 401ab	Tampilkan pesan kesalahan "kodepos tidak valid"	Dapat menampilkan pesan kesalahan "kodepos tidak valid"	[√] Diterima [ ] Ditolak
Security image dikosongkan	Tampilkan pesan kesalahan "security image tidak sesuai"	Dapat menampilkan pesan kesalahan "security image tidak sesuai"	[√] Diterima [ ] Ditolak

4.2.1.5 Pengujian Lupa *Password*

Pengujian lupa *password* dapat dilihat pada tabel 4.16 :

Tabel 4.16 Pengujian Lupa *Password*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data yang benar pada *field e-mail*.	Mengisi *field* dengan benar dan menampilkan pesan "Ganti *Password* berhasil, data anda telah kami kirim ke *e-mail*	Dapat mengisi *field* dengan benar dan menampilkan pesan "Ganti *Password* berhasil, data anda telah kami kirim ke *e-mail*	[√] Diterima [ ] Ditolak

	anda”	anda”	
Kasus dan Hasil Uji Salah (Data Salah)			
Mengisi data yang salah pada *field e-mail*	Menampilkan pesan kesalahan “ *E-mail* yang anda masukan tidak valid”	Proses lupa *password* gagal dan menampilkan pesan kesalahan “ *E-mail* yang anda masukan tidak valid”	[√] Diterima [ ] Ditolak

4.2.1.6 Pengujian Ganti *Password*

Pengujian ganti *password* dapat dilihat pada tabel 4.17 :

Tabel 4.17 Pengujian Ganti *Password*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Semua *field* diisi dengan data yang benar	Ubah *password* berhasil dan meampilkan pesan “Ganti *password* berhasil”	*Password* berhasil di ubah dan menampilkan pesan “Ganti *password* berhasil”	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Semua *field* di kosongkan	Menampilkan pesan kesalahan “Anda belum memasukkan data apapun.	Dapat menampilkan pesan kesalahan “Anda belum memasukkan	[ ] Diterima [ ] Ditolak

	Silahkan isi data dengan lengkap"	data apapun. Silahkan isi data dengan lengkap"	
Hanya mengisi *field password* lama	Muncul pesan kesalahan "*Password* Lama dan Konfirmasi-*Password* Lama tidak sama"	Menampilkan pesan kesalahan "*Password* Lama dan Konfirmasi-*Password* Lama tidak sama"	[√] Diterima [ ] Ditolak
Hanya mengisi *field password* lama dan re-*password* lama	Menampilkan pesan kesalahan "*Password* Baru dan Konfirmasi *Password* Baru belum diisi"	Dapat menampilkan pesan kesalahan "*Password* Baru dan Konfirmasi *Password* Baru belum diisi"	[√] Diterima [ ] Ditolak
Field Re-*Password* baru di kosongkan	Menampilkan pesan kesalahan "*Password* Baru dan Konfirmasi *Password* Baru tidak sama"	Dapat menampilkan pesan kesalahan "*Password* Baru dan Konfirmasi *Password* Baru tidak sama"	[√] Diterima [ ] Ditolak

4.2.1.7 Pengujian Menu Cari

Pengujian menu cari dapat dilihat pada tabel 4.18 :

Tabel 4.18 Pengujian Menu cari

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi *field* dengan data	Menampilkan data barang yang	Dapat menampilkan	[√] Diterima [ ] Ditolak

yang benar	dicari	data barang yang dicari	
Kasus dan Hasil Uji Salah (Data Salah)			
Field pencarian dikosongkan	Menampilkan pesan kesalahan " Anda belum memasukkan data apapun"	Dapat menampilkan pesan kesalahan " Anda belum memasukkan data apapun"	[√] Diterima [ ] Ditolak
Mengisi *field* dengan data yang salah	Menampilkan psan kesalahan " Barang yang anda cari tidak ditemukan"	Dapat enampilkan pesan kesalahan "Barang yang anda cari tidak ditemukan"	[√] Diterima [ ] Ditolak

4.2.1.8 Pengujian Menu Keranjang

Pengujian menu keranjang dapat dilihat pada tabel 4.19 :

Tabel 4.19 Pengujian Menu Keranjang

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukkan jumlah barang pada record barang yang dipilih	Jumlah barang tercantum pada *field* jumlah	Dapat menampilkan jumlah barang yang tercantum pada *field* jumlah	[√] Diterima [ ] Ditolak
Klik tombol belanja lagi	Menampilkan menu semua produk	Dapat menampilkan menu semua	[√] Diterima [ ] Ditolak

		produk	

Tabel 4.20 Tujuan Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Semua *field* diisi dengan data yang benar	Menampilkan data masukkan pada setiap *field* masing-masing	Dapat menampilkan data masukkan pada setiap *field* masing-masing	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Jasa pengiriman dan paket pengiriman tidak dipilih	Menampilkan pesan kesalahan "Jasa pengiriman belum dipilih", "Paket pengiriman belum dipilih".	Dapat menampilkan pesan kesalahan "Jasa pengiriman belum dipilih", "Paket pengiriman belum dipilih".	[√] Diterima [ ] Ditolak
Pilihan paket tidak dipilih	Menampilkan pesan kesalahan "Paket pengiriman belum dipilih".	Dapat menampilkan pesan kesalahan "Paket pengiriman belum dipilih".	[√] Diterima [ ] Ditolak

4.2.1.9 Pengujian Pembayaran

Pengujian form pembayaran dapat dilihat pada tabel 4.24 :

Tabel 4.21 Pengujian Pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Semua *field* diisi dengan data yang benar	Menampilkan data masukkan pada setiap *field* masing-masing	Dapat menampilkan data masukkan pada setiap *field* masing-masing	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Kosongkan *field* yang harus diisi	Menampilkan pesan kesalahan "Anda belum memasukkan data apapun. Silahkan isi data dengan lengkap"	Dapat menampilkan pesan kesalahan "Anda belum memasukkan data apapun. Silahkan isi data dengan lengkap"	[√] Diterima [ ] Ditolak
Tanggal bayar belum dipilih	Menampilkan pesan kesalahan "Tanggal bayar belum diisi".	Dapat menampilkan pesan kesalahan "Tanggal bayar belum diisi"	
No Rekening di kosongkan	Menampilkan pesan kesalahan "No rekening belum diisi".	Dapat menampilkan pesan kesalahan "No rekening belum diisi".	[√] Diterima [ ] Ditolak

4.2.1.10 Pengujian Login Petugas

Pengujian login dapat dilihat pada tabel 4.31 :

Tabel 4.23 Pengujian Login Petugas

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukkan data login, contoh : *Username* : yuni *Password* : yuni	Memasukkan data login dengan benar	Dapat memasukkan data login dengan benar	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Username dan *password* dikosongkan	Menampilkan pesan kesalahan " Login Gagal , *username* dan *password* belum diisi"	petugas tidak dapat login dan dapat menampilkan pesan kesalahan " Login Gagal , *username* dan *password* belum diisi"	[√] Diterima [ ] Ditolak
Username di kosongkan	Menampilkan pesan kesalahan " Login Gagal , *username* belum diisi"	Petugas tidak dapat login dan menampilkan pesan kesalahan " Login Gagal , *username* belum diisi"	[√] Diterima [ ] Ditolak
Password di kosongkan	Menampilkan pesan kesalahan " Login Gagal ,	Petugas tidak dapat login dan menampilkan	[√] Diterima [ ] Ditolak

	Password belum diisi"	pesan kesalahan " Login Gagal , *Password* belum diisi"	
Username dan *password* diisi dengan data yang salah	Menampilkan pesan kesalahan " *Username* atau *password* yang anda masukkan salah"	Petugas tidak dapat login dan menampilkan pesan kesalahan " *Username atau password* yang anda masukkan salah"	[√] Diterima [ ] Ditolak

4.2.1.11 Pengujian Tambah Data Kategori

Pengujian tambah kategori dapat dilihat pada tabel 4.28 :

Tabel 4.24 Pengujian Tambah kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi field kategori dengan data yang benar	Mengisi data pada *field* dengan benar	Dapat mengisi data pada *field* dengan benar	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Field kategori dikosongkan	Menampilkan pesan kesalahan "Data kategori belum diisi"	Dapat menampilkan pesan kesalahan "Data kategori belum diisi"	[√] Diterima [ ] Ditolak
Mengisi *field*	Menampilkan	Dapat	[√] Diterima

kategori dengan data yang sudah ada	pesan kesalahan "Data kategori sudah terdaftar"	menampilkan pesan kesalahan "Data kategori sudah terdaftar"	[ ] Ditolak

4.2.1.12 Pengujian Tambah Jasa

Pengujian tambah jasa dapat dilihat pada tabel 4.29 :

Tabel 4.25 Pengujian Tambah Data Jasa

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi field jasa dengan data yang benar	Mengisi data pada *field* dengan benar	Dapat mengisi data pada *field* dengan benar	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Field jasa dikosongkan	Menampilkan pesan kesalahan "Data jasa belum diisi"	Dapat menampilkan pesan kesalahan "Data jasa belum diisi"	[√] Diterima [ ] Ditolak
Mengisi *field* jasa dengan data yang sudah ada	Menampilkan pesan kesalahan "Data jasa sudah terdaftar"	Dapat menampilkan pesan kesalahan "Data jasa sudah terdaftar"	[√] Diterima [ ] Ditolak

4.2.1.13 Pengujian Tambah Data Paket

Pengujian tambah kategori dapat dilihat pada tabel 4.28 :

Tabel 4.26 Pengujian Tambah Data Paket

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi field dengan data yang benar	Mengisi data pada *field* dengan benar dan menyimpan data ke dalam database	Data tersimpan ke dalam database dan menampilkan pesan "Data paket berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Jasa pengiriman tidak dipilih dan nama paket dikosongkan	Menampilkan pesan kesalahan "Anda belum mengisi apapun"	Dapat menampilkan pesan kesalahan "Anda belum mengisi apapun"	[√] Diterima [ ] Ditolak
Jasa pengiriman tidak dipilih	Menampilkan pesan kesalahan "Anda belum memilih Jasa Pengiriman"	Dapat menampilkan pesan kesalahan "Anda belum memilih Jasa Pengiriman"	[√] Diterima [ ] Ditolak
Nama paket tidak diisi	Menampilkan pesan kesalahan "Nama paket belum diisi"	Dapat menampilkan pesan kesalahan "Nama paket belum diisi"	

4.2.1.14 Pengujian Edit Data Paket

Pengujian tambah kategori dapat dilihat pada tabel 4.28 :

Tabel 4.28 Pengujian Edit Data Paket

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Semua field diisi dengan data yang benar	Mengisi data pada *field* dengan benar dan menyimpan data ke dalam database	Data tersimpan ke dalam database dan menampilkan pesan "Data paket berhasil diupdate"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Mengisi field dengan data yang sudah ada	Menampilkan pesan kesalahan "Nama Paket ""dengan jasa pengiriman ""sudah ada"	Dapat menampilkan pesan kesalahan "Nama Paket ""dengan jasa pengiriman ""sudah ada"	[√] Diterima [ ] Ditolak
Nama paket dikosongkan	Menampilkan pesan kesalahan "Nama paket belum disii"	Dapat menampilkan pesan kesalahan "Nama paket belum disii"	[√] Diterima [ ] Ditolak

4.2.1.15 Pengujian Hapus Data Paket

Pengujian tambah kategori dapat dilihat pada tabel 4.28 :

Tabel 4.29 Pengujian Hapus Data Paket

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memilih data yang akan dihapus dengan memasukkan data	Mengisi data pada *field* dengan benar dan menyimpan data ke dalam database	Menampilkan pesan "Data paket tidak dapat dihapus, pastikan data yang Anda maksud tidak mempunyai relasi dengan data lain"	[√] Diterima [ ] Ditolak

4.2.1.16 Pengujian Pencarian Provinsi

Pengujian tambah kategori dapat dilihat pada tabel 4.28 :

Tabel 4.30 Pengujian Tambah kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi field dengan data yang dicari	Menampilkan data yang dicari	Dapat mengisi data pada *field* dengan benar	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Field provinsi dikosongkan	Menampilkan pesan kesalahan "Data kategori belum diisi"	Dapat menampilkan pesan kesalahan "Data kategori belum diisi"	[√] Diterima [ ] Ditolak

4.2.1.17 Pengujian Data Member

Pengujian tambah data member dapat dilihat pada tabel 4.30 :

Tabel 4.31Pengujian Data Member

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Member di nonaktifkan	Menampilkan peringatan "Data member berhasil dinonaktifkan"	Dapat menampilkan peringatan "Data member berhasil dinonaktifkan"	[√] Diterima [ ] Ditolak
Member di aktifkan	Menampilkan peringatan "Data member berhasil diaktifkan"	Dapat menampilkan peringatan "Data Member berhasil diaktifkan"	[√] Diterima [ ] Ditolak

4.2.1.18 Pengujian Tambah Data Petugas

Pengujian tambah data petugas dapat dilihat pada tabel 4.31 :

Tabel 4.32 Pengujian Data Petugas

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Semua *field* diisi dengan data yang benar	Data petugas dapat tersimpan dengan benar.	Data petugas berhasil disimpan dan menampilkan pesan "Data petugas berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			

Semua *field* dikosongkan	Menampilkan pesan kesalahan "Anda belum memasukkan data apapun, silahkan isi data dengan lengkap"	Dapat menampilkan pesan kesalahan "Anda belum memasukkan data apapun, silahkan isi data dengan lengkap"	[√] Diterima [ ] Ditolak
Field username dikosongkan	Menampilkan pesan kesalahan "*Username* petugas belum diisi"	Dapat menampilkan pesan kesalahan "*Username* petugas belum diisi"	[√] Diterima [ ] Ditolak
Field username diisi dengan data yang sudah ada	Menampilkan pesan kesalahan "*Username* yang anda masukan sudah digunakan"	Dapat menampilkan pesan kesalahan "*Username* yang anda masukan sudah digunakan"	[√] Diterima [ ] Ditolak
Field password dikosongkan	Menampilkan pesan kesalahan "*Password* belum diisi"	Dapat menampilkan pesan kesalahan "*Password* belum diisi"	[√] Diterima [ ] Ditolak
Field Nama dikosongkan	Menampilkan pesan kesalahan "Nama petugas belum diisi"	Dapat menampilkan pesan kesalahan "Nama petugas	[√] Diterima [ ] Ditolak

		belum diisi”	
Field email dikosongkan	Menampilkan pesan kesalahan “Email petugas belum diisi”	Dapat menampilkan pesan kesalahan “Email belum diisi”	[√] Diterima [ ] Ditolak
Field email diisi dengan data yang sudah ada	Menampilkan pesan kesalahan “Email petugas yang anda masukkan sudah terdaftar”	Dapat menampilkan pesan kesalahan “Email yang anda masukkan sudah terdaftar”	[√] Diterima [ ] Ditolak
Field email diisi dengan data yang tidak valid	Menampilkan pesan kesalahan “Email yang anda masukkan tidak valid”	Dapat menampilkan pesan kesalahan “Email yang anda masukkan tidak valid”	[√] Diterima [ ] Ditolak
Field Alamat dikosongkan	Menampilkan pesan kesalahan “Alamat petugas belum diisi”	Dapat menampilkan pesan kesalahan “Alamat petugas belum diisi”	[√] Diterima [ ] Ditolak

4.2.1.19 Pengujian Tambah Data Barang

Pengujian data tambah barang dapat dilihat pada tabel 4.32 :

Tabel 4.33 Pengujian Tambah Data barang

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

Semua *field* diisi dengan data yang benar	Dapat menyimpan data barang dengan benar.	Dapat menyimpan data barang dengan benar.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Semua field dikosongkan	Menampilkan pesan kesalahan "Anda belum memasukkan data apapun, silahkan masukkan data dengan lengkap*"*	Dapat menampilkan pesan kesalahan "Anda belum memasukkan data apapun, silahkan masukkan data dengan lengkap*"*	[√] Diterima [ ] Ditolak
Field nama barang dikosongkan	Tampilkan pesan kesalahan "Nama barang belum diisi*"*	Dapat menampilkan pesan kesalahan "Nama barang belum diisi*"*	[√] Diterima [ ] Ditolak
Field Harga dikosongkan	Tampilkan pesan kesalahan "Harga barang belum diisi"	Dapat menampilkan pesan kesalahan "Harga belum diisi"	[√] Diterima [ ] Ditolak
Field harga diisi dengan data "0" atau (-)	Menampilkan pesan "Harga barang tidak boleh kurang atau sama dengan 0"	Dapat menampilkan pesan kesalahan ""Harga barang tidak boleh kurang atau	[√] Diterima [ ] Ditolak

		sama dengan 0””	
Field berat dikosongkan	Tampilkan pesan peringatan “Berat barang belum diisi”	Dapat menampilkan pesan kesalahan “Berat barang belum diisi”	[√] Diterima [ ] Ditolak
Field berat diisi dengan selain angka	Tampilkan pesan peringatan “Berat barang harus angka”	Dapat menampilkan pesan kesalahan “Berat barang harus angka”	[√] Diterima [ ] Ditolak
Field stok dikosongkan	Tampilkan pesan kesalahan “stok barang belum diisi”	Dapat menampilkan pesan kesalahan “stok barang belum diisi”	[√] Diterima [ ] Ditolak
Field diskon diisi dengan negatif	Tampilkan pesan kesalahan “Diskon barang tidak boleh negatif”	Dapat menampilkan pesan kesalahan “Diskon barang belum diisi”	[√] Diterima [ ] Ditolak
Field gambar dikosongkan	Menampilkan pesan kesalahan “Gambar barang belum diisi”	Dapat menampilkan pesan kesalahan “Gambar barang belum diisi”	[√] Diterima [ ] Ditolak
Field gambar diisi dengan format selain JPG dan JPEG	Menampilkan pesan kesalahan “Gambar barang harus berekstensi	Dapat menampilkan pesan kesalahan “Gambar barang	[√] Diterima [ ] Ditolak

	.JPG atau .JPEG"	harus berekstensi .JPG atau .JPEG"	

4.2.1.20 Pengujian Tambah Data Detail Barang

Pengujian tambah detail barang dapat dilihat pada tabel 4.33 :

Tabel 4.36 Pengujian Tambah Data Detail Barang

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Semua *field* diisi dengan data yang benar	Mengisi data pada *field* dengan benar	Dapat mengisi data pada *field* dengan benar	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Field stok dikosongkan	Tampilkan pesan kesalahan "Stok belum diisi	Dapat menampilkan pesan kesalahan "Stok belum diisi"	[√] Diterima [ ] Ditolak

4.2.1.21 Kesimpulan Hasil Pengujian Alpa

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

4.2.2 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan, dengan menggunakan kuisioner mengenai kepuasan pengguna terhadap aplikasi yang telah dibangun apakah sesuai dengan yang diharapkan. Pengujian beta dilakukan melalui dua teknik pengambilan data, yaitu melalui wawancara dan kuesioner.

Wawancara dilakukan terhadap pengujian beta untuk halaman masing-masing petugas yang terdiri dari petugas bagian keuangan, petugas bagian produksi dan pemilik.

Pengambilan data melalui kuesioner dilakukan terhadap pengujian beta untuk halaman pelanggan. Kuesioner terssebut kemudian disebarkan sepada sebagian pengguna dengan mengambil *sample* sebanyak 30 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penerapan sistem yang baru.

4.2.2.1 Skenario Pengujian Beta

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

1. Lingkungan Petugas *Sifa Collection*

   Penelitian di lingkungan petugas toko dilakukan agar dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan diawal.

2. Lingkungan Masyarakat Umum

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

4.2.2.1.1 Wawancara Pengguna

Berikut ini adalah hasil wawancara dari pemilik *Sifa Collection*, dengan menanyakan pertanyaan dan menyimpulkan jawaban dengan tiga kemungkinan jawaban (Ya, Cukup, Kurang).

1. Apakah menurut anda, dengan adanya *website e-commerce* ini dapat membantu konsumen dalam melakukan pembelian produk-produk *Sifa Collection* ?

   Jawaban : Ya

2. Apakah menurut anda, transaksi pembayaran yang ada pada aplikasi *e-commerce* ini cukup mudah dilakukan konsumen dalam melakukan pembayaran secara *online* ?

   Jawaban : Cukup

3. Apakah menurut anda, dengan adanya *website e-commerce* ini membantu *Sifa Collection* dalam memberikan informasi *detail* produk kepada konsumen dibandingkan dengan cara sebelumnya ?

   Jawaban : Ya

4. Apakah menurut anda, dengan adanya *website e-commerce* ini dapat mempermudah pengolahan laporan penjualan ?

   Jawaban : Ya

5. Apakah menurut anda, dengan adanya aplikasi *e-commerce* biaya promosi produk-produk *Sifa Collection* lebih ringan ?

   Jawaban : Ya

Kesimpulan yang didapat dari hasil wawancara yang dilakukan kepada Bapak Taufik Wahyudi adalah *website e-commerce* yang dibuat sudah bisa membantu untuk melakukan jual beli *online* dan pengolahan data yang diperlukan.

4.2.2.1.2 Kuesioner Pengguna

Kuesioner disebarkan kepada 30 responden. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner ini dianalisis dengan membuat skor A=5, B=4, C=3, D=2, dan E=1 untuk masing-masing jawaban dari setiap pertanyaan.Kuisioner ini terdiri dari 4 pertanyaan (contoh kuisioner dapat dilihat pada lampiran) dengan menggunakan skala *likert* dengan skala 1 sampai 5.

Kuesioner Pengujian Beta Untuk Responden Masyarakat Umum

1. Menurut anda, apakah anda setuju bahwa *website e-commerce* ini sudah membantu konsumen dalam melakukan pembelian secara *online* ?

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

2. Menurut anda, apakah anda setuju bahwa dengan adanya website *e-commerce* ini memudahkan anda dalam melakukan transaksi pembayaran secara online ?

   A. Sangat Setuju
   D Kurang Setuju

B. Setuju E Tidak Setuju

C. Cukup Setuju

3. Menurut anda, apakah jasa dan paket pengiriman yang disediakan sudah memenuhi kebutuhan konsumen ?

A. Sangat Setuju D Kurang Setuju

B. Setuju E Tidak Setuju

C. Cukup Setuju

4. Apakah anda setuju bahwa tampilan website e-commerce ini sudah menarik.?

A. Sangat Setuju D Kurang Setuju

B. Setuju E Tidak Setuju

C. Cukup Setuju

Tabel 4.77 Perhitungan Jumlah Skor Untuk Seluruh Responden Masyarakat Umum Pertanyaan No.1-3

Jawaban	Skor	Responden	Jumlah Pertanyaan	(Skor x Pertanyaan)	Responden x (Skor x Pertanyaan)
A	5	30	3	15	450
B	4	30	3	12	360
C	3	30	3	9	270
D	2	30	3	6	180
E	1	30	3	3	90

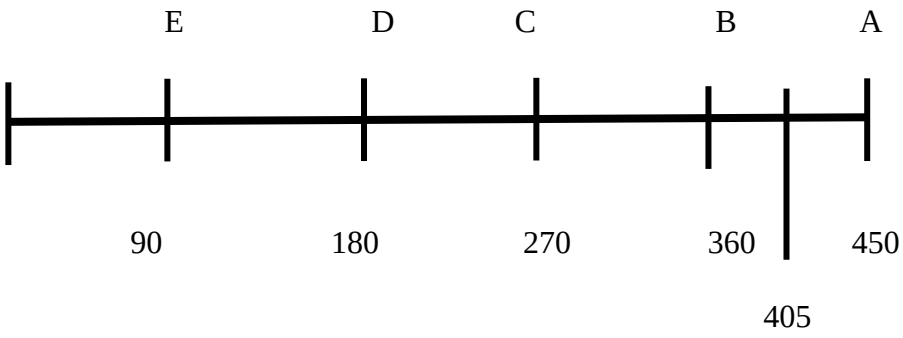
Berdasarkan perhitungan dari jumlah skor untuk seluruh responden dari rumus Responden x (skor x pertanyaan) maka didapat Jumlah skor ideal (maksimum) adalah 450. Berikut ini adalah hasil skor perhitungan masing-masing jawaban yang sudah dihitung nilainya. Kuisioner ini diujikan kepada 30 responden sebagai *sample* yang diambil dari masyarakat umum.

Tabel 4.78 Hasil Pengujian Kuesioner Beta Masyarakat Umum

Responden	Pertanyaan			Total
	No. 1	No.2	No.3	
1	4	4	4	12
2	4	4	4	12
3	4	5	5	14
4	5	4	4	13
5	4	4	4	12
6	4	5	5	14
7	5	5	5	15
8	5	4	4	13
9	4	5	5	14
10	4	4	4	12
1	4	4	4	12
12	4	4	4	12
13	4	5	5	14
14	5	4	4	13
15	4	4	4	12
16	4	5	5	14
17	5	5	5	15
18	5	4	4	13
19	4	5	5	14
20	4	4	4	12
21	4	4	4	12
22	4	4	4	12
23	4	5	5	14
24	5	4	4	13
25	4	4	4	12
26	4	5	5	14
27	5	5	5	15
28	5	4	4	13
29	4	5	5	14
30	4	4	4	12
	Total			131

Setelah menghitung total skor dari masing-masing pertanyaan dari masing-masing responden, maka didapat total skor dari penjumlahan total skor masing-masing skor responden yaitu 131.

Secara kontinum dapat digambarkan sebagai berikut.



Berdasarkan hasil perhitungan data yang diperoleh, skor yang didapat terletak pada posisi melebihi jawaban B, dimana sudah melewati skor rata-rata dan mendekati skor ideal skor ideal 450 maka dapat disimpulkan bahwa responden setuju *website e-commerce* ini seudah membantu mempermudah konsumen dalam melakukan transaksi pembelian secara online.

Tabel 4.3 Tabel 4.77 Perhitungan Jumlah Skor Untuk Seluruh Responden Masyarakat Umum Pertanyaan No.4

Jawaban	Skor	Responden	Jumlah Pertanyaan	(Skor x Pertanyaan)	Responden x (Skor x Pertanyaan)
A	5	30	1	5	150
B	4	30	1	4	120
C	3	30	1	3	90
D	2	30	1	2	60
E	1	30	1	1	30

Berdasarkan perhitungan dari jumlah skor untuk seluruh responden dari rumus Responden x (skor x pertanyaan) maka didapat Jumlah skor ideal (maksimum) adalah 150. Berikut ini adalah hasil skor perhitungan masing-masing

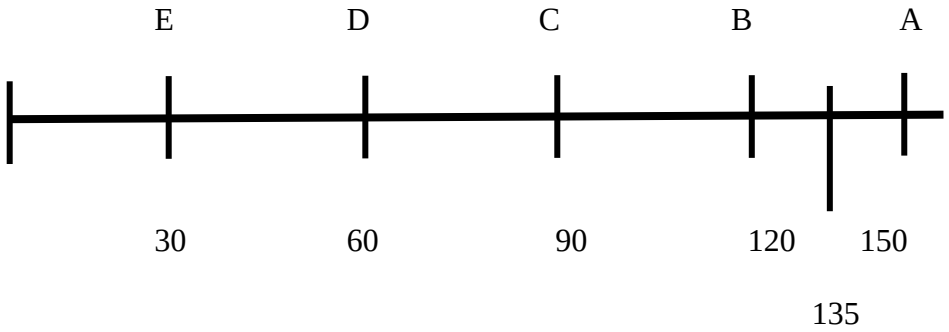
jawaban yang sudah dihitung nilainya. Kuisioner ini diujikan kepada 30 responden sebagai *sample* yang diambil dari masyarakat umum.

Tabel 4.78 Hasil Pengujian Kuesioner Beta Masyarakat Umum

Responden	Pertanyaan	Total
	No.4	
1	4	4
2	4	4
3	5	5
4	4	4
5	4	4
6	5	5
7	5	5
8	4	4
9	5	5
10	4	4
1	4	4
12	4	4
13	5	5
14	4	4
15	4	4
16	5	5
17	5	5
18	4	4
19	5	5
20	4	4
21	4	4
22	4	4
23	5	5
24	4	4
25	4	4
26	5	5
27	5	5
28	4	4
29	5	5
30	4	4

Setelah menghitung total skor dari masing-masing pertanyaan dari masing-masing responden, maka didapat total skor dari penjumlahan total skor masing-masing skor responden yaitu 44.

Secara kontinum dapat digambarkan sebagai berikut.



Berdasarkan hasil perhitungan data yang diperoleh, skor yang didapat terletak pada posisi melebihi jawaban B, dimana sudah melewati skor rata-rata dan mendekati skor ideal skor ideal 150 maka dapat disimpulkan bahwa responden setuju website e-commercer ini sudah memiliki tampilan yang menarik.

4.2.2.2 Kesimpulan Pengujian Beta

Berdasarkan hasil wawancara dan kuesioner yang didapatkan dari pengujian beta pengguna masyarakat, pemilik Sifa Collection dan petugas butik, maka sistem penerapan *e-commerce* untuk sistem penjualan ini sudah sesuai dengan tujuan yaitu membantu memudahkan perusahaan dalam mempromosikan produknya, memudahkan konsumen membeli barang tanpa harus mendatangi tempat penjualan, memperluas cakupan penjualan dan mempermudah dalam merekap hasil laporan.