BAB 4
IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

Implementasi sistem adalah tahapan yang dilakukan setelah perancangan, tahapan ini adalah mengimplementasikan perancangan ke dalam bahasa pemrograman sehingga terciptanya aplikasi. Tujuan dari implementasi sistem adalah untuk menerapkan perancangan yang telah dilakukan terhadap sistem sehingga pengguna dapat memberikan masukan bagi pengembangan sistem yang telah dibangun.

4.1.1 Implementasi Perangkat Keras

Spesifikasi perangkat keras yang akan digunakan dalam penerapan aplikasi *website* penjualan *online* yang dibangun adalah sebagai berikut :

1. *Processor* : 2.20 Ghz.
2. *Harddisk* : 160 GB.
3. *Memory* : 2 GB.
4. *Monitor* : 14” Resolusi 1024 x 768.
5. VGA : 256 MB.

4.1.2 Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam penerapan aplikasi *website* penjualan *online* yang dibangun adalah sebagai berikut :

1. Sistem Operasi : Sistem Operasi *Windows* XP SP2 dan versi diatasnya.
2. Bahasa Pemrograman : PHP.
3. DBMS : MySQL.
4. *Web Browser* : *Mozilla Firefox* 2.5
5. Program Editor : *NetBeans* 7.0
6. Sistem *Modeller* : *Rational Rose* 2003 *Enterprise*

4.1.3 Implementasi *Hosting* dan *Server*

Spesifikasi *hosting*, *domain*, IP *dedicated* dan *SSL (Secure Socket Layer)* yang dibutuhkan untuk membangun aplikasi *e-commerce* untuk bisa *online* adalah sebagai berikut :

1. Domain menggunakan nama *gareushoes.com* dengan biaya Rp. 560.000,-/tahun.
2. Penggunaan IP *dedicated* dengan biaya Rp. 200.000,-/tahun.
3. Penggunaan SSL *(Secure Socket Layer)* untuk keamanan situs dan *hosting* dengan biaya Rp. 350.000,- /tahun.

Rincian diatas merupakan syarat minimal agar suatu aplikasi *e-commerce* dapat berjalan dengan baik di *internet*, untuk *hosting* dengan *space* 1 GB merupakan *hosting* dengan *space* minimal untuk *e-commerce*, kemungkinan untuk setiap *merchant* membutuhkan *space* untuk *hosting* berbeda-beda.

4.1.4 Implementasi Database

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

1. Implementasi database

```
CREATE DATABASE ‘gareushoes_db;
```

2. Implementasi Tabel Produk

```
CREATE TABLE `produk` (
    `id_produk` INT(11) NOT NULL AUTO_INCREMENT,
    `id_kategori` INT(11) NOT NULL,
    `id_edisi_katalog` INT(11) NOT NULL,
    `kd_produk` VARCHAR(12) NOT NULL,
    `image` VARCHAR (255) DEFAULT '0',
    `deskripsi` TEXT NOT NULL,
    `harga` DECIMAL(10,2) NOT NULL DEFAULT '0.00',
    `diskon` INT(4) NOT NULL,
    `dilihat` INT(4) DEFAULT '0',
    `berat` INT(4) DEFAULT '0',
    `status` Tinyint(1),
    `tgl_awal_diskon` DATETIME DEFAULT NULL,
    `tgl_akhir_diskon` DATETIME DEFAULT NULL,
    `tgl_tersedia` DATETIME DEFAULT NULL,
    `tgl_insert` DATETIME DEFAULT NULL,
    `tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_produk`),
KEY `FK_produk_kategori` (`id_kategori`),
KEY `FK_produk_edisi` (`id_edisi_katalog`),
CONSTRAINT `FK_produk_kategori` FOREIGN KEY
(`id_kategori`) REFERENCES `kategori` (`id_kategori`),
CONSTRAINT `FK_produk_edisi` FOREIGN KEY
(`id_edisi_katalog`) REFERENCES `edisi_katalog`
(`id_edisi_katalog`),
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

3. Implementasi Tabel Produk Gambar

```
CREATE TABLE `produk_gambar` (
    `id_produk_gambar` INT(11) NOT NULL
    AUTO_INCREMENT,
    `id_produk` INT(11) NOT NULL,
    `image` VARCHAR(255) NOT NULL
PRIMARY KEY (`id_produk_gambar`),
KEY `FK_produk ` (`id_produk`),
CONSTRAINT `FK_produk_gambar ` FOREIGN KEY
(`id_produk`) REFERENCES `produk` (`id_produk`) ON UPDATE
CASCADE
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

4. Implementasi Tabel Produk Stok

```
CREATE TABLE `produk_stok` (
    `id_produk_stok` INT(11) NOT NULL AUTO_INCREMENT,
    `id_produk` INT(11) NOT NULL,
    `id_warna` INT(11) NOT NULL,
    `id_ukuran` INT(11) NOT NULL,
    `jumlah` INT(4) DEFAULT '0',
PRIMARY KEY (`id_produk_stok`),
KEY `FK_stok_ukuran` (`id_ukuran`),
KEY `FK_stok_warna` (`id_warna`),
CONSTRAINT `FK_stok_warna` FOREIGN KEY (`id_warna`)
REFERENCES `warna` (`id_warna`),
CONSTRAINT `FK_stok_ukuran` FOREIGN KEY (`id_ukuran`)
REFERENCES `ukuran` (`id_ukuran`)
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

5. Implementasi Tabel Warna

```
CREATE TABLE `warna` (
    `id_warna` INT(11) NOT NULL AUTO_INCREMENT,
    `nama` VARCHAR(50) NOT NULL
PRIMARY KEY (`id_warna`,`nama`),
UNIQUE KEY `nama` (`nama`)
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

6. Implementasi Tabel Ukuran

```
CREATE TABLE `ukuran` (
    `id_ukuran` INT(11) NOT NULL AUTO_INCREMENT,
    `nama` VARCHAR(50) NOT NULL,
PRIMARY KEY (`id_ukuran`,`nama`),
UNIQUE KEY `nama` (`nama`)
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

7. Implementasi Tabel Edisi Katalog

```
CREATE TABLE `edisi_katalog` (
    `id_edisi_katalog` INT(11) NOT NULL AUTO_INCREMENT,
    `nama` VARCHAR(50) NOT NULL
PRIMARY KEY (`id_edisi_katalog`),
UNIQUE KEY `nama` (`nama`)
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

8. Implementasi Tabel Kategori

```
CREATE TABLE `kategori` (
    `id_kategori` INT(11) NOT NULL AUTO_INCREMENT,
    `id_parent` INT(11) NOT NULL,
    `nama` VARCHAR(255) NOT NULL,
    `status`TINYINT(1) NOT NULL,
    `tgl_insert` DATETIME DEFAULT NULL,
    `tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_kategori'),
UNIQUE KEY `nama` (`nama`),
KEY `FK_kategori` (`id_parent`),
CONSTRAINT `FK_parent` FOREIGN KEY (`id_kategori`)
REFERENCES `kategori` (`id_kategori`),
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

9. Implementasi Tabel Operator

```
CREATE TABLE `operator` (
    `id_operator` INT(11) NOT NULL AUTO_INCREMENT,
    `username` VARCHAR(128) NOT NULL,
    `password` VARCHAR(50) NOT NULL,
    `nama_lengkap` VARCHAR(50) NOT NULL,
    `email` VARCHAR(96) DEFAULT NULL,
    `nama_grup` ENUM() DEFAULT NULL,
    `hak_akses` TEXT DEFAULT NULL,
    `status`TINYINT(1) NOT NULL,
    `tgl_insert` DATETIME DEFAULT NULL,
    `tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_perator`),
UNIQUE KEY `email` (`email`)
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

10. Implementasi Tabel Provinsi

```
CREATE TABLE `provinsi` (
    `id_provinsi` INT(11) NOT NULL AUTO_INCREMENT,
    `nama` VARCHAR(50) NOT NULL,
    `status` TINYINT(1) NOT NULL',
PRIMARY KEY (`id_provinsi`),
UNIQUE KEY `nama` (`nama`)
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

11. Implementasi Tabel Kota

```
CREATE TABLE `kota` (
    `id_kota` INT(11) NOT NULL AUTO_INCREMENT,
    `id_provinsi` INT(11) NOT NULL,
    `nama` VARCHAR(128) NOT NULL,
    `status` TINYINT(1) NOT NULL,
PRIMARY KEY (`id_kota`),
KEY `FK_kota_provinsi` (`id_provinsi`),
CONSTRAINT `FK_kota_provinsi` FOREIGN KEY (`id_provinsi`)
REFERENCES `provinsi` (`id_provinsi`) ON DELETE CASCADE
ON UPDATE CASCADE
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

12. Implementasi Tabel *Member*

```
CREATE TABLE `operator` (
    `id_member` INT(11) NOT NULL AUTO_INCREMENT,
    `nama_lengkap` VARCHAR(32) NOT NULL,
    `email` VARCHAR(96) NOT NULL,
    `perusahaan` VARCHAR(32) NOT NULL,
    `telepon` VARCHAR(15) DEFAULT NULL,
    `fax` VARCHAR(32) DEFAULT NULL,
    `alamat` VARCHAR(128) DEFAULT NULL,
    `id_kota` INT(11) NOT NULL,
    `kodepos` INT(6) DEFAULT NULL,
    `password` VARCHAR(40) NOT NULL,
    `kode_aktivasi` VARCHAR(128) DEFAULT NULL,
    `minat` TEXT DEFAULT NULL,
    `keranjang` TEXT DEFAULT NULL,
    `token` VARCHAR(128) DEFAULT NULL,
    `status` ENUM('Registrasi,'Aktivasi') DEFAULT 'Registrasi',
    `tgl_insert` DATETIME DEFAULT NULL,
    `tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_member`),
UNIQUE KEY `email` (`email`),
KEY `FK_member_kota` (`id_kota`),
CONSTRAINT `FK_member_kota` FOREIGN KEY (`id_kota`)
REFERENCES `kota` (`id_kota`) ON UPDATE CASCADE
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

13. Implementasi Tabel Jenis Kirim

```
CREATE TABLE `jenis_kirim` (
    `id_jenis` INT(11) NOT NULL AUTO_INCREMENT,
    `pengiriman` VARCHAR(50) NOT NULL,
    `nama` VARCHAR(50) NOT NULL,
    `status`TINYINT(1) NOT NULL,
PRIMARY KEY (`id_jenis`),
UNIQUE KEY `nama` (`nama`),
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

14. Implementasi Tabel Biaya Kirim

```
CREATE TABLE `biaya_kirim` (
    `id_biaya_kirim` INT(11) NOT NULL AUTO_INCREMENT,
    `id_jenis_kirim` INT(11) NOT NULL,
    `id_kota` INT(11) NOT NULL,
    `harga` DECIMAL(10,2) DEFAULT NULL,
    `status` TINYINT NOT NULL,
    `tgl_insert` DATETIME DEFAULT NULL,
    `tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_biaya_kirim`),
UNIQUE KEY `idx_kota_jenis_kirim` (`id_jenis_kirim`,`id_kota`),
KEY `FK_biaya_kirim_kota` (`id_kota`),
CONSTRAINT `FK_biaya_kirim_jenis_kirim` FOREIGN KEY
(`id_jenis_kirim`) REFERENCES `jenis_kirim` (`id_jenis_kirim`)
ON UPDATE CASCADE,
CONSTRAINT `FK_biaya_kirim_kota` FOREIGN KEY (`id_kota`)
REFERENCES `kota` (`id_kota`) ON UPDATE CASCADE
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

15. Implementasi Tabel Komentar

```
CREATE TABLE `komentar` (
    `id_komentar` INT(11) NOT NULL AUTO_INCREMENT,
    `id_produk` INT(11) NOT NULL,
    `id_member` INT(11) NOT NULL,
    `deskripsi` VARCHAR(50) NOT NULL,
    `rating` INT(4) NOT NULL,
    `status` TINYINT(1) DEFAULT NOT NULL,
    `tgl_insert` DATETIME DEFAULT NULL,
    `tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_komentar`),
KEY `FK_komen_produk` (`id_produk`),
KEY `FK_komen_member` (`id_member`),
CONSTRAINT `FK_komen_produk` FOREIGN KEY (`id_warna`)
REFERENCES `produk` (`id_produk`),

CONSTRAINT `FK_komen_member` FOREIGN KEY
(`id_member`) REFERENCES `member` (`id_member`)
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

16. Implementasi Tabel Testimonial

```
CREATE TABLE `testimonial` (
    `id_testimonial` INT(11) NOT NULL AUTO_INCREMENT,
    `id_member` INT(11) NOT NULL,
    `id_operator` INT(11) NOT NULL,
    `judul` Varchar(64) NOT NULL,
    `deskripsi` TEXT NOT NULL,
    `status` TINYINT(1) DEFAULT NOT NULL,
    `tgl_insert` DATETIME DEFAULT NULL,
    `tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_testimonial`),
KEY `FK_testimonial_member` (`id_member`),
KEY `FK_testimonial_operator` (`id_operator`),
CONSTRAINT `FK_testimonial_member` FOREIGN KEY
(`idMember`) REFERENCES `member` (`id_member`) ON
UPDATE CASCADE,
CONSTRAINT `FK_testimonial_operator` FOREIGN KEY
(`id_operator`) REFERENCES `operator` (`id_operator`) ON
UPDATE CASCADE
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

17. Implementasi Tabel Pesanan

```
CREATE TABLE `pesanan` (
    `id_pesanan` INT(11) NOT NULL AUTO_INCREMENT,
    `no_faktur` VARCHAR(26) DEFAULT NULL,
    `faktur_sementara` VARCHAR(26) DEFAULT NULL,
    `no_resi` VARCHAR(26) DEFAULT NULL,
    `id_member` INT(11) NOT NULL,
    `id_biaya_kirim` INT(11) DEFAULT NULL,
    `kirim_nama_lengkap` VARCHAR(32) DEFAULT NULL,
    `kirim_perusahaan` VARCHAR(32) DEFAULT NULL,
    `kirim_alamat` VARCHAR(128) DEFAULT NULL,
    `kirim_provinsi` VARCHAR(128) DEFAULT NULL,
    `kirm_kota` VARCHAR(128) DEFAULT NULL,
    `kirim_kodepos` CHAR(10) DEFAULT NULL,
    `kirim_metode` VARCHAR(128) DEFAULT NULL,
    `bayar_nama_lengkap` VARCHAR(32) DEFAULT NULL,
    `bayar_perusahaan` VARCHAR(32) DEFAULT NULL,
    `bayar_alamat` VARCHAR(128) DEFAULT NULL,
    `bayar_provinsi` VARCHAR(128) DEFAULT NULL,
    `bayar_kota` VARCHAR(128) DEFAULT NULL,
    `bayar_kodepos` VARCHAR(10) DEFAULT NULL,
    `cara_bayar` ENUM('Transfer Bank','Paypal') DEFAULT NULL,
    `rekening_tujuan` ENUM('BCA','Mandiri','Paypal') DEFAULT NULL,
    `sub_total` DECIMAL(10,2) DEFAULT NULL,
    `pengiriman` DECIMAL(10,2) DEFAULT NULL,
    `ambil_simpanan` DECIMAL(10,2) DEFAULT NULL,
    `total` DECIMAL(10,2) DEFAULT NULL,
    `status` ENUM('Pesanan Dikonfirmasi','Pesanan Dibatalkan','Konfirmasi
    Pembayaran','Pembayaran Dikonfirmasi','Pembayaran Gagal','Sedang
    Dikirim','Sudah Diterima','Kadaluarsa') DEFAULT 'Pesanan
    Dikonfirmasi',
    ‘tgl_insert` DATETIME DEFAULT NULL,
    ‘tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_Pesanan`),
KEY `FK_pesanan_member` (`id_member`),
KEY `FK_pesanan_biaya_kirim` (`id_biaya_kirim`),
CONSTRAINT `FK_pesanan_biaya_kirim` FOREIGN KEY
(`idBiayaKirim`) REFERENCES `biayakirim` (`id_biaya_kirim`) ON
UPDATE CASCADE,
CONSTRAINT `FK_pesanan_member` FOREIGN KEY (`id_member`)
REFERENCES `member` (`id_member`) ON UPDATE CASCADE,
UPDATE CASCADE
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

18. Implementasi Tabel Pesanan Produk

```
CREATE TABLE `pesanan_produk` (
    `id_pesanan_produk`INT(11) NOT NULL
    AUTO_INCREMENT,
    `id_pesanan` INT(11) DEFAULT NULL,
    `id_produk_stok` INT(11) DEFAULT NULL,
    `jumlah` INT(4) DEFAULT '1',
    `harga` DECIMAL(10,2) DEFAULT NULL,
    `total` DECIMAL(10,2) DEFAULT NULL,
PRIMARY KEY (`id_pesanan_produk`),
KEY `FK_pesanan_produk_pesanan` (`id_pesanan`),
KEY `FK_pesanan_produk_stok` (`id_produk_stok`),
CONSTRAINT `FK_pesanan_produk_pesanan` FOREIGN KEY
(`id_pesanan`) REFERENCES `pesanan` (`id_pesanan`) ON
DELETE CASCADE ON UPDATE CASCADE
CONSTRAINT `FK_pesanan_stok` FOREIGN KEY
(`id_produk_stok`) REFERENCES `produk_stok` (`id_produk_stok
`) ON UPDATE CASCADE,
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

19. Implementasi Tabel *Retur*

```
CREATE TABLE `retur` (
    `id_retur` INT(11) NOT NULL AUTO_INCREMENT,
    `id_pesanan` INT(11) DEFAULT NULL,
    `tgl_terima_toko` DATE DEFAULT NULL,
    `no_resi_member` VARCHAR(50) DEFAULT NULL,
    `keterangan` TEXT,
    `status` ENUM('Pengajuan Retur','Retur Diproses','Retur
    Ditolak','Retur Dibatalkan','Retur Diterima','Retur
    Dikonfirmasi','Retur Telah Dikonfirmasi','Barang Retur Dikirim
    Member','Barang Retur Diterima Toko',’Barang Retur Diterima
    Member') DEFAULT 'Pengajuan Retur',
    `tgl_insert` DATETIME DEFAULT NULL,
    `tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_retur`),
KEY `FK_retur_pesanan` (`id_pesanan`),
CONSTRAINT `FK_retur_pesanan` FOREIGN KEY (`id_pesanan`)
REFERENCES `pesanan` (`id_pesanan`) ON UPDATE CASCADE
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

20. Implementasi Tabel *Retur* produk

```
CREATE TABLE `retur_produk` (
    `id_produk_retur` INT(11) NOT NULL AUTO_INCREMENT,
    `id_retur` INT(11) NOT NULL,
    `id_pesanan_produk` INT(11) NOT NULL,
    `jumlah` INT(4) NOT NULL DEFAULT '1',
    `alasan_retur` ENUM('Barang Rusak','Barang Tidak
    Sesuai',’Tukar Ukuran’,’Tukar Warna,’Tukar Model’) NOT
    NULL,
    `status` ENUM('Diajukan','Diterima','Ditolak','Dibatalkan')
    DEFAULT 'Diajukan',
    `tgl_insert` DATETIME DEFAULT NULL,
    `tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_retur_produk`),
KEY `FK_retur_produk` (`id_retur`),
CONSTRAINT `FK_retur ` FOREIGN KEY (`id_retur`)
REFERENCES `retur` (`id_retur`) ON DELETE CASCADE ON
UPDATE CASCADE
CONSTRAINT `FK_retur_produk` FOREIGN KEY
(`id_produk_pesanan`) REFERENCES `pesanan_produk`
(`id_pesanan_produk`) ON DELETE CASCADE ON UPDATE
CASCADE
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

21. Implementasi Tabel Simpanan

```
CREATE TABLE `simpanan` (
    `id_simpanan` INT(11) NOT NULL AUTO_INCREMENT,
    `id_retur` INT(11) NOT NULL,
    `deskripsi` VARCHAR(50) DEFAULT NULL,
    `jumlah` INT(11) NOT NULL DEFAULT '1',
    `debet` DECIMAL(10,2) DEFAULT NULL,
    `kredit` DECIMAL(10,2) DEFAULT NULL,
    `tgl_insert` DATETIME DEFAULT NULL,
    `tgl_update` DATETIME DEFAULT NULL,
PRIMARY KEY (`id_simpanan`),
KEY `FK_simpanan_retur` (`id_retur`),
CONSTRAINT `FK_retur ` FOREIGN KEY (`id_retur`)
REFERENCES `retur` (`id_retur`) ON DELETE CASCADE ON
UPDATE CASCADE
) ENGINE=INNODB DEFAULT CHARSET=latin1;
```

4.1.5 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dengan pengkodeannya dalam bentuk berkas program. Berikut ini adalah implementasi antar muka yang ditunjukkan dengan Tabel 4.1.

Tabel 4.1 Implementasi Antarmuka

Menu	Deskripsi	Nama File
Halaman Utama Pengunjung	*File* program untuk menampilkan halaman utama pengunjung.	Utama. Beranda.php
Halaman Penyajian Informasi Produk	*File* program untuk menampilkan halaman informasi tentang produk	Utama. Produk.php
Halaman Profil Toko	*File* program untuk menampilkan informasi mengenai Toko.	Utama Profile.php
Halaman Cara Belanja	*File* program untuk menampilkan informasi mengenai cara belanja.	Utama Belanja.php
Halaman Cara Pembayaran	*File* program untuk menampilkan informasi mengenai cara pembayaran.	Utama. Pembayaran.php
Halaman Cara Pengiriman	*File* program untuk menampilkan informasi mengenai cara pengiriman.	Utama. Pengiriman.php
Halaman Daftar *Member*	*File* program untuk menampilkan *form* pendaftaran member	Akun. Pendaftaran.php
Halaman Utama *Member*	*File* program untuk menampilkan halaman utama *member*	Akun. Akun.php
Halaman Kelola Profil *Member*	*File* program untuk menampilkan halaman kelola profil *member*	Akun. *Edit*.php
Halaman *Edit Password*	*File* program untuk menampilkan halaman ubah data *password member*	Akun. Password.php
Halaman Daftar Minat Produk	*File* program untuk menampilkan halaman dafta minat produk member	Akun. Minat.php
Halaman Riwayat Pesanan	*File* program untuk menampilkan halaman data riwayat pesanan *member*	Akun. Pesanan.php
Halaman Riwayat Retur Produk	*File* program untuk menampilkan data riwayat retur produk.	Akun. Retur_list.php
Halaman detail retur produk	*File* program untuk menampilkan halaman detail retur produk	Akun. Retur_info.php

Tabel 4.1 Implementasi Antarmuka (Lanjutan)

Menu	Deskripsi	Nama File
Halaman Riwayat Transaksi *Member*	*File* program untuk menampilkan halaman transaksi *member*	Akun. Transaksi.php
Halaman Pengajuan Retur	*File* program untuk menampilkan halaman pengajuan retur barang	Akun. Retur_form.php
Halaman Keranjang	*File* program untuk menampilkan halaman produk yang akan dibeli	Checkout. keranjang.php
Halaman Konfirmasi Pengiriman Pesanan	*File* program untuk menampilkan halaman konfirmasi pengiriman pesanan	Informasi. Pembayaran.php
Halaman Data Katalog Produk	*File* program untuk menampilkan halaman data katalog produk	Katalog. Produk.php
Halaman Data Produk Diskon	*File* program untuk menampilkan semua data produk diskon	Utama. beranda.php
Halaman Hasil *Tracking* Pengiriman JNE	*File* program untuk menampilkan halaman hasil *tracking* pengiriman JNE	Pengiriman. *tracking*.php
Halaman *Input* Data Testimonial	*File* program untuk menampilkan *form* penambahan data testimonial	Produk. Isitestimonial.php
Halaman Lupa *Password member*	*File* program untuk menampilkan *form* permintaan lupa *password member*	Akun. Lupa_password.php
Halaman *Login* Operator	*File* program untuk menampilkan *form login* administrasi atau operator	Utama. Login.php
Halaman Lupa *Password* Operator	*File* program untuk menampilkan *form* lupa *password* operator	Utama. Lupa_*password*.php
Halaman Utama Operator	*File* program untuk menampilkan halaman utama operator	Utama. Beranda.php
Halaman Pengolahan Data Produk	*File* program untuk menampilkan halaman pengolahan data produk	Katalog. Produk_list.php
Halaman Ubah dan tambah Produk	*File* program untuk menampilkan halaman ubah produk	Katalog. Produk_form.php

Tabel 4.1 Implementasi Antarmuka (Lanjutan)

Menu	Deskripsi	Nama File
Halaman Pengolahan Data Kategori	*File* program untuk menampilkan halaman pengolahan data kategori	Katalog. Kategori_list.php
Halaman Tambah Kategori	*File* program untuk menampilkan halaman tambah kategori	Katalog. Kategori_form.php
Halaman Pengolahan Data Warna	*File* program untuk menampilkan halaman pengolahan data warna	Katalog. Warna_list.php
Halaman Ubah Warna	*File* program untuk menampilkan halaman ubah warna	Katalog. Warna_form.php
Halaman Pengolahan Data Ukuran	*File* program untuk menampilkan halaman pengolahan data ukuran	Katalog. Ukuran_list.php
Halaman Tambah Ukuran	*File* program untuk menampilkan halaman tambah ukuran	Katalog. Ukuran_form.php
Halaman Pengolahan Komentar	*File* program untuk menampilkan halaman tambah ukuran	Katalog. Komentar_list.php
Halama merubah data komentar	*File* program untuk merubah data komentar produk.	Katalog. Komentar_form.php
Halaman Pengolahan Testimonial	*File* program untuk menampilkan halaman pengolahan data testimonial	Katalog. Testimonial_list.php
Halaman melihat data Testimonial	*File* program untuk melihat data testimonial	Katalog. Testimonial_info.php
Halaman Pengolahan Data Provinsi	*File* program untuk menampilkan halaman pengolahan provinsi	Lokasi. Provinsi_list.php
Halaman Tambah Provinsi	*File* program untuk menambah atau merubah data provinsi	Lokasi. Provinsi_form.php
Halaman Pengolahan Data Kategori	*File* program untuk menampilkan halaman pengolahan data kategori	Katalog. Kategori_list.php
Halaman Pengolahan Data Kota	*File* program untuk menampilkan halaman pengolahan data kota	Lokasi. kota_list.php
Halaman Tambah Kota	*File* program untuk menambah atau merubah data kota	Lokasi. Kota_form.php
Halaman Pengolahan Data pengiriman	*File* program untuk menampilkan halaman pengolahan pengiriman	Lokasi. Pengiriman_list.php
Halaman Tambah atau Edit Pengiriman	*File* program untuk menampilkan halaman tambah atau edit pengiriman	Lokasi. Pengiriman_form.php
Halaman Pengolahan Data Member	*File* program untuk menampilkan halaman pengolahan member	Penjualan. member_list.php

Tabel 4.1 Implementasi Antarmuka (Lanjutan)

Menu	Deskripsi	Nama File
Halaman Pengolahan data pesanan	*File* program untuk menampilkan halaman pengolahan pesanna	Penjualan. Pesanan_list.php
Halaman melihat data pesanan	*File* program untuk melihat detail dari data pesanan.	Penjualan. pesanan_info.php
Halama membuat faktur penjualan	*File* program untuk menampilkan halaman tambah ukuran	Penjualan. Pesanan_faktur.php
Halaman Pengolahan data retur	*File* program untuk menampilkan halaman pengolahan data retur produk.	Penjualan. Retur_list.php
Halaman melihat data retur	*File* program untuk menampilkan halaman detail dari retur produk.	Penjualan. Retur_info.php
Halaman Pengolahan data operator	*File* program untuk menampilkan halaman pengolahan operator.	Operator. Operator_list.php
Halaman Tambah dan edit operator	*File* program untuk menambah atau merubah data operator	Operator. Operator_form.php
Halaman Pengolahan Data Member	*File* program untuk menampilkan halaman pengolahan member	Penjualan. member_list.php
Halaman Pengolahan data pesanan	*File* program untuk menampilkan halaman pengolahan pesanna	Penjualan. Pesanan_list.php
Halaman melihat data pesanan	*File* program untuk melihat detail dari data pesanan.	Penjualan. pesanan_info.php
Halama membuat faktur penjualan	*File* program untuk menampilkan halaman tambah ukuran	Penjualan. Pesanan_faktur.php
Halaman Pengolahan data retur	*File* program untuk menampilkan halaman pengolahan data retur produk.	Penjualan. Retur_list.php
Halaman melihat data retur	*File* program untuk menampilkan halaman detail dari retur produk.	Penjualan. Retur_info.php
Halaman Pengolahan data operator	*File* program untuk menampilkan halaman pengolahan operator.	Operator. Operator_list.php
Halaman Tambah dan edit operator	*File* program untuk menambah atau merubah data operator	Operator. Operator_form.php
Halaman Pengolahan Data Member	*File* program untuk menampilkan halaman pengolahan member	Penjualan. member_list.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 alpa dilakukan dengan menggunakan metode *black box.* Untuk pengujian alpa ini yaitu pada pengujian sebagai pengguna.

Proses pengujian *black box* adalah pengujian yang dilakukan dengan cara mencoba program aplikasi dengan memasukkan data ke dalam form-form yang telah disediakan. Pada tahap ini merupakan kelanjutan dari tahap implementasi. Pengujian yang akan dilakukan dengan cara alpha yaitu dengan metode pengujian *black box* yang berfokus pada persyaratan fungsional perangkat lunak. Pengujian ini memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya semua persyaratan fungsional untuk suatu program.

4.2.1.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara alpha menggunakan metode *black box* dan *betha* dengan kuesioner.

1. Rencana pengujian halaman Administrator.

   Rencana pengujian halaman administrator dilakukan oleh kepala *supervisor* Toko dan dibimbing oleh pihak *developer* dapat dilihat pada Tabel 4.2.

Tabel 4.2 Rencana Pengujian Halaman Administrator

Item Pengujian	Detail Pengujian	Jenis Pengujian
Login administrator	Isi data *login*	*Black Box*
	Verifikasi data *login* Administrator	*Black Box*
Pengolahan Kategori	Tampil data kategori	*Black Box*
	Tambah data kategori	*Black Box*
	Edit data kategori	*Black Box*
	Hapus data kategori	*Black Box*
Pengolahan data produk	Tampil data produk	*Black Box*
	Tambah data produk	*Black Box*
	Edit data produk	*Black Box*
	Hapus data produk	*Black Box*
	Filter data produk	*Black Box*
Pengolahan data provinsi	Tampil data komentar produk	*Black Box*
	Edit data komentar produk	*Black Box*
	Hapus data komentar produk	*Black Box*
Pengolahan data testimonial	Tampil data testimonial	*Black Box*
	Edit data testimonial	*Black Box*
	Hapus data testimonial	*Black Box*
Pengolahan data operator	Tampil data operator	*Black Box*
	Tambah data operator	*Black Box*
	Edit data operator	*Black Box*

2. Rencana pengujian halaman *Supervisor* Penjualan.

   Rencana pengujian halaman Supervisor Penjualan dilakukan oleh kepala supervisor toko dan dibimbing oleh pihak *developer* dapat dilihat pada Tabel 4.3.

Tabel 4.3 Rencana Pengujian Halaman *Supervisor* Penjualan

	Detail Pengujian	Jenis Pengujian
Login Supervisor Penjualan	Isi data *login*	*Black Box*
	Verifikasi data *login*	*Black Box*
	Menampilkan menu *supervisor* penjualan	*Black Box*
Pengolahan Pesanan	Menampilkan data pesanan	*Black Box*
	Membatalkan pesanan	*Black Box*
	Melihat data pesanan	*Black Box*
	Edit data pesanana	*Black Box*
	Hapus data pesanan	*Black Box*
Pengolahan retur	Menampilkan data retur	*Black Box*
	Membatalkan retur	*Black Box*
	Konfirmasi pengajuan retur	*Black Box*
	Melihat detail retur	*Black Box*
Laporan	Membuat laporan produk	*Black Box*
	Membuat laporan pesanan	*Black Box*
logout	Menghapus sesi login operator	*Black Box*

3. Rencana pengujian halaman Operator Penjualan.

   Rencana pengujian halaman Operator Penjualan dilakukan oleh pemilik toko dan dibimbing oleh pihak *devoleper* dapat dilihat pada Tabel 4.4.

Tabel 4.44 Rencana Pengujian Halaman Operator Penjualan

	Detail Pengujian	Jenis Pengujian
Login administrator	Isi data *login*	*Black Box*
	Verifikasi data *login* Administrator	*Black Box*
	Menampilkan menu	*Black Box*
Pengolahan Pesanan	Tampil data pesanan	*Black Box*
	Tambah data pesanna	*Black Box*
	Edit data pesanan	*Black Box*
	Hapus data pesanan	*Black Box*

4. Rencana pengujian halaman *Member*

   Rencana pengujian halaman *member* dapat dilihat pada Tabel 4.5.

Tabel 4.45 Rencana Pengujian Halaman *Member*

	Detail Pengujian	**Jenis Pengujian**
Login Member	Isi data *login*	*Black Box*
	Verifikasi email	*Black Box*
	Verifikasi password	
Lupa *Password*	Menampilkan *form* lupa *password*	*Black Box*
	Masukkan *email* member	*Black Box*
	Kirim *link reset password* ke *email* member	*Black Box*
Ubah *Password*	Menampilkan *form* ubah *password* member	*Black Box*
	Mengubah *password*	*Black Box*
Keranjang Belanja	Menampilkan data detail pemesanan	*Black Box* *Black Box*
	Mengubah data detail pemesanan	*Black Box*
	Menghapus data detail pemesanan	*Black Box*
Form Pengiriman	Menampilkan *form* alamat tujuan pengiriman barang	*Black Box*
	Isi form alamat tujuan pengiriman barang	*Black Box*
Form Pembayaran	Menampilkan total bayar dan cara pembayaran	*Black Box*
	Isi data pembayaran	*Black Box*
	Menyimpan data pembayaran	*Black Box*
Riwayat Pesanan	Menampilkan riwayat pesanan	*Black Box*
	Membatalkan pesanan	*Black Box*
	Mengajukan retur barang	*Black Box*
	Menampilkan detail pesanan	*Black Box*
	Menampilkan detail retur	*Black Box*
Form Testimonial	Menampilkan data testimonial yang dimasukkan oleh *member*	*Black Box*
	Isi data testimonial	*Black Box*
	Menyimpan data testimonial	*Black Box*

5. Rencana pengujian halaman Pengunjung

   Rencana pengujian halaman Pengunjung dapat dilihat pada Tabel 4.6.

Tabel 4.6 Rencana Pengujian Halaman Pengunjung

***Item* Pengujian**	***Detail* Pengujian**	**Jenis Pengujian**
Pendaftaran	Menampilkan *form* pendaftaran	*Black Box*
	Isi *form* pendaftaran	*Black Box*
	Simpan data pendaftaran	*Black Box*
	Buka *email*	*Black Box*
Aktivasi	Aktivasi pendaftaran	*Black Box*

4.2.1.1 Kasus dan Hasil Pengujian Alpha

Proses pengujian *alpha* dilakukan untuk mengetahui apakah data yang dimasukkan (*input*) sudah sesuai dengan apa yang diharapkan (*output*). Berdasarkan skenario pengujian, maka dapat dilakukan pengujian *alpha* pada *website e-commerce* di Toko Gareu Shoes.

1. **Pengujian Pendaftaran *Member***

   Pengujian *alpha* pada proses pendaftaran *member* dapat dilihat pada Tabel 4.7.

Tabel 4.7 Pengujian Pendaftaran *Member*

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data pendaftaran, contoh masukan : Nama : Dudi ahmad Email : dudi_ahmad@yahoo.com Alamat : Jl. Cibaduyut Lama No. 31 Telepon : 082100766142 Provinsi : Jawa Barat Kota : Bandung Kodepos : 40132 *Password :* T-r3xx Konfirmasi *password* : T-r3xx	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

Tabel 4.7 Pengujian Pendaftaran *Member* (Lanjutan)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data masukan pada form pendaftaran tidak lengkap	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan "Peringatan: Periksa kembali form dengan benar untuk mencari kesalahan".	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan "Peringatan: Periksa kembali form dengan benar untuk mencari kesalahan".	[√] Diterima [ ] Ditolak
Nama lengkap tidak diisi	Tampilkan pesan kesalahan "Nama harus terdiri dari 1 sampai 32 karakter"	Dapat menampilkan pesan kesalahan "Nama harus terdiri dari 1 sampai 32 karakter"	[√] Diterima [ ] Ditolak
Password tidak diisi	Tampilkan pesan kesalahan "*password* harus terdiri dari 4 sampai 20 karakter"	Dapat menampilkan pesan kesalahan " *password* harus terdiri dari 4 sampai 20 karakter"	[√] Diterima [ ] Ditolak
Konfirmasi *Password* tidak diisi	Tampilkan pesan kesalahan "*Password* konfirmasi tidak sesuai dengan *password*"	Dapat menampilkan pesan kesalahan " *Password* konfirmasi tidak sesuai dengan *password*"	[√] Diterima [ ] Ditolak
Alamat *email* tidak diisi atau penulisannya tidak sesuai.	Tampilkan pesan kesalahan"Alamat *email* tidak *valid*"	Dapat menampilkan Pesan kesalahan "Alamat e*mail* tidak *valid*"	[√] Diterima [ ] Ditolak
Alamat *email* sudah tersedia	Tampilkan pesan kesalahan " Error: Alamat E-Mail sudah terdaftar!"	Dapat menampilkan Pesan kesalahan "Error: Alamat E-Mail sudah terdaftar!"	[√] Diterima [ ] Ditolak

Tabel 4.7 Pengujian Pendaftaran *Member* (Lanjutan)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Alamat tidak diisi	Tampilkan pesan kesalahan ”Alamat harus terdiri dari 3 sampai 128 karakter”	Dapat menampilkan pesan kesalahan ”Alamat harus terdiri dari 3 sampai 128 karakter”	[√] Diterima [ ] Ditolak
Provinsi tidak diisi	Tampilkan pesan kesalahan ”Silahkan pilih provinsi”	Dapat menampilkan pesan kesalahan ”Silahkan pilih provinsi”	[√] Diterima [ ] Ditolak
Kota tidak diisi	Tampilkan pesan kesalahan ”Silahkan pilih kota”	Dapat menampilkan Pesan kesalahan ”Silahkan pilih kota”	[√] Diterima [ ] Ditolak
Kode pos tidak diisi	Tampilkan pesan kesalahan ”Kode pos harus terdiri dari 2 sampai 10 karakter”	Dapat menampilkan pesan kesalahan ”Kode pos harus terdiri dari 2 sampai 10 karakter”	[√] Diterima [ ] Ditolak
Kode verifikasi tidak sesuai dengan gambar	Tampilkan pesan kesalahan ” Kode verifikasi tidak sesuai dengan gambar!”	Dapat menampilkan pesan kesalahan ”Kode verifikasi tidak sesuai dengan gambar!”	[√] Diterima [ ] Ditolak
Tidak menyetujui ketentuan member	Tampil pesan kesalahan “Error: Anda harus menyetujui ketentuan member.	Dapat menampilkan pesan kesalahan “Error: Anda harus menyetujui ketentuan member.	[√] Diterima [ ] Ditolak

2. Pengujian *Login* Member

Pengujian *alpha* pada proses login *member* dapat dilihat pada Tabel 4.8.

Tabel 4.8 Pengujian *Login Member*

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan alamat *email :* dudi.ahmad@yahoo.com Password: Pa$$w0rd	Mengisikan data alamat *email* pada *field* alamat *email*.	Dapat mengisi *field* alamat *email* dan dapat mengisi data *login* dengan benar	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data *email* atau *password* yang tidak terdaftar	Tampilkan pesan kesalahan "*Error*: Tidak cocok antara alamat *email* dengan *password* "	Tampil pesan "*Error*: Tidak cocok antara alamat *email* dengan *password*"	[√] Diterima [ ] Ditolak

3. Pengujian *Login* Member

Pengujian *alpha* pada proses pengubahan *password member* dapat dilihat pada Tabel 4.9.

Tabel 4.9 Pengujian *Login Member*

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data login contoh: *username:*Indra Password :Pa$$w0rd	Mengisikan data login berupa *username* dan password dengan benar dan menampilkan halaman utama.	Dapat mengisi data login berupa *username* dan dengan benar dan menampilkan halaman utama	[√] Diterima [ ] Ditolak

Tabel 4.9 Pengujian *Login Member* (Lanjutan)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukan data *login* berupa *username* dan *password* yang tidak terdaftar	Tampilkan pesan kesalahan "Tidak ada *Username* atau *Password* yang anda masukan "	Muncul pesan "Tidak ada *Username* atau *Password* yang anda masukan"	[√] Diterima [ ] Ditolak

4. Pengujian Kelola *Akun Member*

Pengujian *alpha* pada proses kelola *akun member* berupa pengubahan data akun member dapat dilihat pada Tabel 4.10.

Tabel 4.10 Pengujian Kelola *Akun Member*

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data akun member yang yang akan diubah	Dapat melakukan pengolahan pada data akun *member*.	Dapat melakukan pengolahan pada data akun *member*.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data pada form masukan akun member tidak lengkap	Menampilkan pesan kesalahan "Peringatan: Periksa kembali form dengan benar untuk mencari kesalahan".	Dapat menampilkan pesan kesalahan "Peringatan: Periksa kembali form dengan benar untuk mencari kesalahan".	[√] Diterima [ ] Ditolak
Nama lengkap tidak diisi	Tampilkan pesan kesalahan "Nama harus terdiri dari 1 sampai 32 karakter"	Dapat menampilkan pesan kesalahan "Nama harus terdiri dari 1 sampai 32 karakter"	[√] Diterima [ ] Ditolak

Tabel 4.11 Pengujian Pengubahan *Akun Member* (Lanjutan)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Alamat tidak diisi	Tampilkan pesan kesalahan "Nama harus terdiri dari 1 sampai 32 karakter"	Tampilkan pesan kesalahan "Nama harus terdiri dari 1 sampai 32 karakter"	[√] Diterima [ ] Ditolak
Provinsi tidak diisi	Tampilkan pesan kesalahan "Silahkan pilih provinsi"	Dapat menampilkan pesan kesalahan "Silahkan pilih provinsi"	[√] Diterima [ ] Ditolak
Kota tidak diisi	Tampilkan pesan kesalahan "Silahkan pilih kota"	Dapat menampilkan Pesan kesalahan "Silahkan pilih kota"	[√] Diterima [ ] Ditolak
Kode pos tidak diisi	Tampilkan pesan kesalahan "Kode pos harus terdiri dari 2 sampai 10 karakter"	Dapat menampilkan Pesan kesalahan "Kode pos harus terdiri dari 2 sampai 10 karakter"	[√] Diterima [ ] Ditolak

5. Pengujian Keranjang Belanja

Pengujian *alpha* pada proses pengelolaan keranjang belanja dapat dilihat pada Tabel 4.12.

Tabel 4.12 Pengujian Keranjang Belanja

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data jumlah : 2	Mengisi data jumlah	Dapat mengisi data jumlah dengan benar	[√] Diterima [ ] Ditolak

Tabel 4.12 Pengujian Keranjang Belanja (Lanjutan)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisikan *field* jumlah dengan melebihi stok: Contoh: Jumlah: 5 Stok: 2	Tampilan pesan kesalahan "Produk tidak tersedia dalam jumlah yang diinginkan"	Dapat menampilkan pesan kesalahan "Produk tidak tersedia dalam jumlah yang diinginkan "	[√] Diterima [ ] Ditolak

6. Pengujian Form Pengiriman

Pengujian *alpha* pada proses konfirmasi pengiriman pesanan dapat dilihat pada Tabel 4.13.

Tabel 4.13 Pengujian *Form* Pengiriman

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi semua Data alamat tujuan	Mengisikan semua data pada form alamat pengiriman	Dapat mengisi semua data pada form alamat pengiriman	[√] Diterima [ ] Ditolak
Memilih metode pengiriman Contoh: Memilih TIKI pada *Checkbox*	Dapat melanjutkan pada tahap berikutnya yaitu menentukan metode pembayaran.	Dapat melanjutkan ke tahap berikutnya pada proses menuju kasir/checkout.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Metode pengiriman tidak dipilih	Menampilkan pesan kesalahan "Peringatan: Metode pengiriman diperlukan"	Dapat menampilkan pesan kesalahan "Peringatan: Metode pengiriman diperlukan"	[√] Diterima [ ] Ditolak

7. Pengujian *Form* Konfirmasi Pembayaran

Pengujian *alpha* pada proses konfirmasi pembayaran dapat dilihat pada Tabel 4.14.

Tabel 4.14 Pengujian Form Konfirmasi Pembayaran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan data konfirmasi pembayaran, contoh No. pesanan : 1 Nama Bank : BCA-ATM Atas Nama : Amtori Rekening Tujuan: Bank BCA No. Rek. 346-1899984	Dapat mengisi data form pembayaran dan menampilkan pesan "Suses: Anda telah melakukan konfirmasi pembayaran"	Dapat menampilkan form konfirmasi pembayaran.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data pada form masukan konfirmasi pembayaran tidak lengkap	Tampilkan pesan kesalahan "Peringatan: periksa kembali form dengan benar untuk mencari kesalahan"	Dapat menampilkan pesan kesalahan " Peringatan: periksa kembali form dengan benar untuk mencari kesalahan"	[√] Diterima [ ] Ditolak
Jumlah pembayaran tidak sesuai dengan jumlah pembelian	Tampilkan pesan kesalahan "Jumlah pembayaran tidak benar"	Dapat menampilkan pesan "Jumlah pembayaran tidak benar".	[√] Diterima [ ] Ditolak
Kode verifikasi tidak sesuai dengan gambar	Tampilkan pesan kesalahan " Kode verifikasi tidak sesuai dengan gambar r"	Tampilkan pesan kesalahan " Kode verifikasi tidak sesuai dengan gambar r"	[√] Diterima [ ] Ditolak

8. Pengujian Form Testimonial

Pengujian *alpha* pada proses memasukkan data *testimonial* dapat dilihat pada Tabel 4.15.

Tabel 4.15 Pengujian *Form Testimonial*

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan data testimonial contoh: Judul: pelayanan Deskripsi: xxxx Rating: 1 Status: 0	Menambah data *testimonial* yang dilakukan oleh *member* dan menampilkan pesan "Sukses: Anda telah menambah data"	Dapat menambahkan data testimonial serta menampilkan pesan "Sukses: Anda telah menambah data".	[√] Diterima [ ] Ditolak
Menyetujui data testimonial baru Contoh: Judul: pelayanan Deskripsi: xxxx Rating: 1 Status: 1	Mengubah status data testimonial baru menjadi *enabled* dan menampilkan data testimonial yang dilakukan oleh *supervisor* penjualan dengan pesan sukses "Sukses: Anda telah mengubah data".	Dapat mengubah dan menampilkan data testimonial baru dengan menampilkan pesan "Sukses: Anda telah mengubah data"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data pada form masukan tidak lengkap	Menampilkan pesan "Peringatan: periksa kembali form dengan benar untuk mencari kesalahan"	Dapat menghasilkan pesan "Peringatan: periksa kembali form dengan benar untuk mencari kesalahan"	[√] Diterima [ ] Ditolak

9. Pengujian *Login* Operator

Pengujian *alpha* pada proses login operator dapat dilihat pada Tabel 4.16.

Tabel 4.16 Pengujian *Login Administrator*

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Username : admin	Mengisikan data *username* pada *field* isian *username*	Dapat mengisi *field username* pada *field* isian *username*	[√] Diterima [ ] Ditolak
Password : admin	Mengisikan data *password* pada *field* isian *password*	Dapat mengisi *field password* pada *field* isian *password*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Username atau *password* tidak terdaftar.	Tampilan kesalahan "*username* atau *password* tidak cocok"	Dapat menampilkan pesan kesalahan "*username* atau *pasword* tidak cocok"	[√] Diterima [ ] Ditolak

10. Pengujian Penambahan Data Provinsi

Pengujian *alpha* pada proses penambahan data provinsi dapat dilihat pada Tabel 4.17.

Tabel 4.17 Pengujian Penambahan Data Provinsi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data provinsi Contoh: Nama Provinsi : Jawa Barat	Jika semua *field* terisi maka akan muncul pesan "Sukses: anda telah menambah data"	Jika semua *field* terisi maka akan muncul pesan "Sukses: anda telah menambah data".	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama provinsi sudah tersedia	Tampilkan pesan kesalahan "Nama provinsi telah tersedia"	Dapat menampilkan pesan kesalahan "Nama provinsi telah tersedia"	[√] Diterima [ ] Ditolak

11. Pengujian Penghapusan Data Provinsi

Pengujian *alpha* pada proses penghapusan data provinsi dapat dilihat pada Tabel 4.18.

Tabel 4.18 Pengujian Penghapusan Data Provinsi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih data provinsi yang akan dihapus	Jika semua *field* terisi maka akan muncul pesan "Sukses: anda telah menghapus data"	Jika semua *field* terisi maka akan muncul pesan "Sukses: anda telah menghapus data".	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih data provinsi yang akan dihapus	Tampilkan pesan kesalahan "Peringatan: Data provinsi ini tidak dapat dihapus karena sedang digunakan pada data kota"	Dapat menampilkan pesan kesalahan " Peringatan: Data provinsi ini tidak dapat dihapus karena sedang digunakan pada data kota"	[√] Diterima [ ] Ditolak

12. Pengujian Pengubahan Data Provinsi

Pengujian *alpha* pada proses pengubahan data provinsi dapat dilihat pada Tabel 4.19.

Tabel 4.19 Pengujian Pengubahan Data Provinsi

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih data provinsi yang akan diubah	Menampilkan data produk pada form produk	Dapat menampilkan data produk pada form produk	[√] Diterima [ ] Ditolak
Menyimpan data provinsi yang telah diubah	Tampilkan pesan "Sukses: anda telah mengubah data"	Dapat menampilkan pesan "Sukses: anda telah mengubah data"	

Tabel 4.19 Pengujian Pengubahan Data Provinsi (Lanjutan)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data masukan masih kosong	Tampilkan pesan kesalahan "Peringatan: Periksa form dengan benar untuk mencari kesalahan"	Dapat menampilkan pesan kesalahan " Peringatan: Periksa form dengan benar untuk mencari kesalahan"	[√] Diterima [ ] Ditolak

13. Pengujian Penambahan Data Kota

Pengujian *alpha* pada proses penambahan data kota dapat dilihat pada Tabel 4.20.

14. Tabel 4.20 Pengujian Penambahan Data Kota

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data provinsi Contoh: Pilih Provinsi: Jawa Barat Nama Kota: Bandung	Jika semua *field* terisi maka akan muncul pesan "Sukses: anda telah menambah data"	Jika semua *field* terisi maka akan muncul pesan "Sukses: anda telah menambah data".	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data masukan masih kosong	Tampilkan pesan kesalahan "Peringatan: Periksa form dengan benar untuk mencari kesalahan"	Dapat menampilkan pesan kesalahan " Peringatan: Periksa form dengan benar untuk mencari kesalahan"	[√] Diterima [ ] Ditolak
Nama kota sudah tersedia	Tampilkan pesan kesalahan "Nama kota telah tersedia"	Dapat menampilkan pesan kesalahan "Nama kota telah tersedia"	

15. Pengujian Penghapusan Data Kota

Pengujian *alpha* pada proses penghapusan data kota dapat dilihat pada Tabel 4.21.

Tabel 4.21 Pengujian Penghapusan Data Kota

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh: ID kota dengan nama: Ciamis akan dihapus	Jika ID kota dengan nama Casual belum digunakan pada transaksi maka tampil pesan "Sukses: Anda telah menghapus data".	Data kota dapat dihapus jika ID kota dengan nama Ciamis belum digunakan pada transaksi maka tampil pesan "Sukses: Anda telah menghapus data".	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh: ID kota dengan nama: Ciamis sudah digunakan pada transaksi	Tampilkan pesan kesalahan "Peringatan: Data kota ini tidak dapat dihapus karena sedang digunakan pada data member maupun data transaksi"	Dapat menampilkan pesan kesalahan " Peringatan: Data kota ini tidak dapat dihapus karena sedang digunakan pada data member maupun data transaksi"	[√] Diterima [ ] Ditolak

16. Pengujian Pengubahan Data Kota

Pengujian *alpha* pada proses pengubahan data kota dapat dilihat pada Tabel 4.22.

Tabel 4.22 Pengujian Pengubahan Data Kota

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh: Nama sebelum dirubah: Camis Nama sesudah dirubah: Ciamis	Dapat mengubah data kota dan muncul pesan "Sukses: anda telah menambah data"	Dapat mengubah data kota	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data masukan masih kosong	Tampilkan pesan kesalahan "Peringatan: Periksa form dengan benar untuk mencari kesalahan"	Dapat menampilkan pesan kesalahan " Peringatan: Periksa form dengan benar untuk mencari kesalahan"	[√] Diterima [ ] Ditolak
Data masukan sudah tersedia	Tampilkan pesan kesalahan "Nama kota sudah tersedia"	Dapat menampilkan pesan kesalahan " Nama kota sudah tersedia"	[√] Diterima [ ] Ditolak

17. Pengujian Penambahan Data Kategori

Pengujian *alpha* pada proses penambahan data kategori dapat dilihat pada Tabel 4.23

Tabel 2.23 Pengujian Penambahan Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data kategori Contoh: Nama kategori: Sepatu Casual	Jika semua *field* terisi maka akan muncul pesan "Sukses: Anda telah menambah data"	Jika semua *field* terisi maka akan muncul pesan "Sukses: Anda telah menampah data"	[√] Diterima [ ] Ditolak

Tabel 2.23 Pengujian Penambahan Data Kategori (Lanjutan)

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data masukan pada form kelola kategori tidak lengkap	Tampilkan pesan kesalahan "Peringatan: Periksa form dengan benar untuk mencari kesalahan"	Dapat menampilkan pesan kesalahan "Periksa form dengan benar untuk mencari"	[√] Diterima [ ] Ditolak
Nama kategori masih kosong	Tampilkan pesan kesalahan "Nama harus terdiri dari 3 sampai 32 karakter"	Dapat menampilkan pesan kesalahan "Nama kategori harus terdiri dari 3 sampai 32 karakter"	[√] Diterima [ ] Ditolak

18. Pengujian Penghapusan Data Kategori

Pengujian *alpha* pada proses penghapusan data kategori dapat dilihat pada Tabel 4.24.

Tabel 4.24 Pengujian Penghapusan Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh: ID kategori dengan nama: Casual akan dihapus	Jika ID kategori dengan nama Casual belum digunakan pada transaksi maka tampil pesan "Sukses: Anda telah menghapus data".	Data kategori dapat dihapus jika ID kategori dengan nama Casual belum digunakan pada transaksi maka tampil pesan "Sukses: Anda telah menghapus data".	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh: ID kategori dengan nama: Casual telah digunakan pada transaksi	Tampilkan pesan kesalahan "Peringatan: Data kategori ini tidak dapat dihapus karena sedang digunakan pada data produk"	Dapat menampilkan pesan kesalahan " Peringatan: Data kategori ini tidak dapat dihapus karena sedang digunakan pada data produk"	[√] Diterima [ ] Ditolak

19. Pengujian Pengubahan Data Kategori

Pengujian *alpha* pada proses pengubahan data kategori dapat dilihat pada Tabel 4.25.

Tabel 4.25 Pengujian Pengubahan Data Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh: Nama sebelum dirubah: Kasual Nama sesudah dirubah: Casual	Dapat mengubah data kategori dan muncul pesan "Sukses: anda telah mengubah data"	Dapat mengubah data kategori	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data masukan masih kosong	Tampilkan pesan kesalahan "Peringatan: Periksa form dengan benar untuk mencari kesalahan"	Dapat menampilkan pesan kesalahan " Peringatan: Periksa form dengan benar untuk mencari kesalahan"	[√] Diterima [ ] Ditolak
Data masukan sudah tersedia	Tampilkan pesan kesalahan "Data kategori telah digunakan"	Dapat menampilkan pesan kesalahan " Data kategori telah digunakan"	[√] Diterima [ ] Ditolak

20. Pengujian Penambahan Data Produk

Pengujian *alpha* pada proses penambahan data produk dapat dilihat pada Tabel 4.26

Tabel 4.26 Pengujian Penambahan Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data produk Contoh: Kode Produk : G 5678 Kategori:Sepatu Pria Edisi katalog: GRU 2013	Jika ID produk dengan kode G 5678 belum digunakan pada transaksi maka tampil pesan "Sukses: Anda telah menghapus data".	Data produk dapat dihapus jika ID produk dengan kode G 5678 belum digunakan pada transaksi maka tampil pesan "Sukses: Anda telah menghapus data".	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data masukan masih kosong	Tampilkan pesan kesalahan "Peringatan: Periksa form dengan benar untuk mencari kesalahan"	Dapat menampilkan pesan kesalahan "Peringatan: Periksa form dengan benar untuk mencari kesalahan"	[√] Diterima [ ] Ditolak
Tidak mengisi data produk stok	Tampilkan pesan kesalahan "Peringatan: Kode Produk harus memiliki identitas lengkap yaitu warna dan ukuran"	Dapat menampilkan pesan kesalahan "Peringatan: Kode Produk harus memiliki identitas lengkap yaitu warna dan ukuran"	[√] Diterima [ ] Ditolak
Kategori tidak dipilih	Tampilkan pesan kesalahan "Kategori harus dipilih"	Dapat menampilkan pesan kesalahan "Kategori harus dipilih"	[√] Diterima [ ] Ditolak
Edisi katalog tidak dipilih	Tampilkan pesan kesalahan "Edisi katalog harus dipilih"	Dapat menampilkan pesan kesalahan "Edisi katalog harus dipilih"	[√] Diterima [ ] Ditolak

21. Pengujian Penghapusan Data Produk

Pengujian *alpha* pada proses penghapusan data produk dapat dilihat pada Tabel 4.27.

Tabel 4.27 Pengujian Penghapusan Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh: ID produk dengan kode: G 5678 akan dihapus	Jika ID produk dengan kode G 5678 belum digunakan pada transaksi maka tampil pesan "Sukses: Anda telah menghapus data".	Data produk dapat dihapus jika ID produk dengan kode G 5678 belum digunakan pada transaksi maka tampil pesan "Sukses: Anda telah menghapus data".	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh: ID produk dengan kode: G 5678 telah digunakan pada transaksi	Tampilkan pesan kesalahan "Peringatan: Data produk ini tidak dapat dihapus karena telah digunakan pada data transaksi"	Dapat menampilkan pesan kesalahan " Peringatan: Data produk ini tidak dapat dihapus karena telah digunakan pada data produk"	[√] Diterima [ ] Ditolak

22. Pengujian Pengubahan Data Produk

Pengujian *alpha* pada proses pengubahan data produk dapat dilihat pada Tabel 4.28.

Tabel 4.29 Pengujian Pengubahan Data Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data produk yang akan diubah Contoh: Data awal Kode Produk : G 5678 Kategori:Sepatu Pria Edisi katalog: GRU 2013 Data akhir Kode Produk : G 5678 Kategori:Sepatu Wanita Edisi katalog: GRU 2013	Dapat mengubah data produk dan muncul pesan "Sukses: anda telah mengubah data"	Dapat mengubah data produk	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Diharapkan**	**Pengamatan**	**Kesimpulan**
Data masukan masih kosong	Tampilkan pesan kesalahan "Peringatan: Periksa form dengan benar untuk mencari kesalahan"	Dapat menampilkan pesan kesalahan " Peringatan: Periksa form dengan benar untuk mencari kesalahan"	[√] Diterima [ ] Ditolak
Data masukan sudah tersedia	Tampilkan pesan kesalahan "Data produk telah digunakan"	Dapat menampilkan pesan kesalahan " Data produk telah digunakan"	[√] Diterima [ ] Ditolak

4.2.1.2 Kesimpulan Hasil Pengujian Alpha

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

4.2.2 Pengujian Betha

4.2.2.1 Skenario Pengujian Betha

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan membuat kuesioner mengenai kepuasan pelanggan dengan kandungan poin syarat *user friendly* untuk selanjutnya dibagikan kepada sebagian *user* secara langsung, sedangkan untuk bagian administrasi, pengujian dilakukan dengan cara wawancara tertulis. Dari hasil kuesioner dan wawancara tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem.

A. Kuesioner Pengujian Betha

1. Skala Likert

Skala likert digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang. Dengan skala likert, maka variable yang akan diukur dijabarkan menjadi indikator variabel. Kemudian indikator tersebut dijadikan sebagai titik tolak ukur untuk menyusun item-item instrumen yang dapat berupa pertanyaan.

Jawaban setiap item instrumen yang menggunakan skala likert dapat berupa kata-kata dengan skor jawaban, dapat dilihat pada Tabel 4.27.

Tabel 4.27 Bobot

Keterangan	Bobot
Sangat setuju	5
Setuju	4
Ragu ragu	3
Tidak setuju	2
Sangat tidak setuju	1

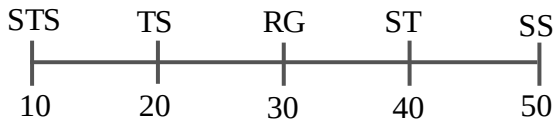
Untuk mencari nilai prosentase dari masing–masing jawaban kuesioner digunakan rumus skala likert sebagai berikut :

$$\mathrm{Y} = \frac{S}{skorideal} \times 100 \qquad (4.1)$$

Keterangan :

Y	=	Nilai persentase yang dicari.
S	=	Total yang diperoleh dari S =$\sum_{i=1}^{5} n_i . r_i$
r_i	=	Banyak responden yang menjawab jawaban ke-i
n_i	=	Nilai jawaban ke-i
Skor ideal		Nilai tertinggi dikalikan dengan jumlah sampel atau responden

Dibawah ini merupakan skala untuk mengambil kesimpulan dari setiap petanyaan



Gambar 2.4 *Skala Likert* [15]

2. Kuesioner Pengujian Betha untuk Pengunjung

Berikut ini diberikan beberapa pertanyaan yang diberikan kepada pengunjung, tentang tanggapan pengunjung terhadap *user friendly* dari aplikasi *e-commerce* penjualan *online* pada Toko Gareu Shoes adalah sebagai berikut :

1. Apakah instruksi-instruksi yang diberikan *website* http://gareushoes.com mudah untuk dimengerti?
2. Apakah anda setuju tampilan *website* penjualan *online* http://gareushoes.com sudah menarik?
3. Apakah bahasa yang disajikan pada aplikasi *website* penjualan online http://gareushoes mudah dimengerti?
4. Apakah anda merasa nyaman dengan tampilan dari aplikasi *website* penjualan *online* http://gareushoes.com?
5. Apakah penggunaan warna, *font* dari aplikasi website penjualan online http://gareushoes.com sudah konsisten?

Hasil Kuesioner yang dibagikan kepada 30 orang pengunjung dengan 5 pertanyaan mengenai *user friendly* dengan menggunakan 5 pilihan jawaban pada setiap pertanyaan dapat dilihat pada Tabel 4.27.

Tabel 4.27 Hasil Pengujian Kuesioner *User Frendly* (Pengunjung)

No.Soal	Sangat Setuju	Setuju	Ragu ragu	Tidak setuju	Sangat tidak setuju
1.	9	12	9	0	0
2.	11	11	8	0	0
3.	15	11	4	0	0
4.	10	11	9	0	0
5.	13	14	3	0	0
Total	58	59	33	0	0

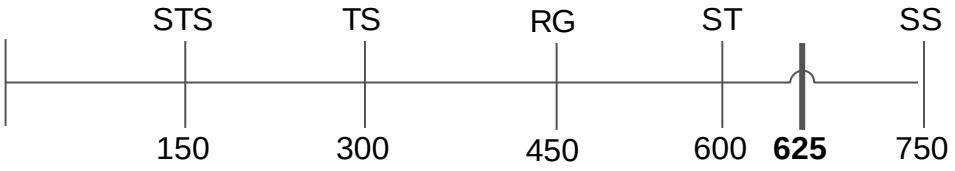
Sedangkan hasil dari perhitungan pada setiap jawaban dapat dilihat pada Tabel 4.27 .

Tabel 4.27 Hasil Pengukuran Pengujian Kuesioner *User Frendly* (Pengunjung)

Keterangan	N	R	N x R
Sangat setuju	5	58	290
Setuju	4	59	124
Ragu ragu	3	33	27
Tidak setuju	2		0
Sangat tidak setuju	1		0
		150	625

Berdasarkan data dari hasil kuesioner tersebut jumlah skor ideal (kriterium) untuk seluruh item = 5 x 30 = 150 (seandainya semua jawaban SS). Jumlah skor yang diperoleh dari pertanyaan mengenai *user frendly* = 625. Jadi berdasarkan data tersebut maka tingkat persetujuan terhadap *user frendly* pada *website* penjualan aplikasi *e-commerce* Toko Gareu Shoes = (625 : (150*5)) x 100% = 83% dari yang diharapkan (100%).

Secara kontinum dapat digambarkan pada Gambar 4.3:



Gambar 4.3 *Skala Likert* Pengujian Kuesioner *User Frendly* (Pengunjung)

Berdasarkan data yang diperoleh dari 30 responden maka rata-rata 625 persen terletak pada daerah setuju.

Berikut ini diberikan beberapa pertanyaan yang diberikan kepada pengunjung, tentang tanggapan pengunjung terhadap *user friendly* dari aplikasi *e-commerce* penjualan *online* pada Toko Gareu Shoes adalah sebagai berikut :

6. Apakah *website* http://gareushoes.com memberikan informasi secara detail mengenai produk yang dijual?
7. Apakah *website* http://gareushoes.com memberikan informasi mengenai produk diskon, produk paling laris?
8. Setelah melihat *website e-commerce* http://gareushoes.com apakah anda berminat untuk mengunjunginya lagi lain waktu?
9. Apakah berminat untuk bergabung dengan *website* http://gareushoes.com?
10. Apakah informasi mengenai Toko Gareu Shoesdapat mudah dicari pada *search engine*?

Hasil Kuesioner yang dibagikan kepada 30 pengunjung dengan 5 pertanyaan dan menggunakan 5 pilihan jawaban pada setiap pertanyaan dapat dilihat pada Tabel 4.27.

Tabel 4.27 Hasil Pengujian Kuesioner Pengunjung (Proses Promosi)

No.Soal	Sangat Setuju	Setuju	Ragu ragu	Tidak setuju	Sangat tidak setuju
1.	14	11	5	0	0
2.	12	14	4	0	0
3.	10	11	9	0	0
4.	9	15	6	0	0
5.	13	14	3	0	0
Total	58	65	27	0	0

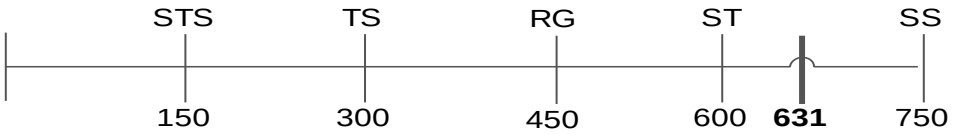
Sedangkan hasil dari perhitungan pada setiap jawaban dapat dilihat pada Tabel 4.27 .

Tabel 4.27 Perhitungan Pengujian Kuesioner Pengunjung (Proses Promosi)

Keterangan	N	R	N x R
Sangat setuju	5	58	290
Setuju	4	65	260
Ragu ragu	3	27	81
Tidak setuju	2	0	0
Sangat tidak setuju	1	0	0
		150	631

Berdasarkan data dari hasil kuesioner tersebut jumlah skor ideal (kriterium) untuk seluruh item = 5 x 30 = 150 (seandainya semua jawaban SS). Jumlah skor yang diperoleh dari pertanyaan mengenai promosi produk = 631. Jadi berdasarkan data tersebut maka tingkat persetujuan terhadap promosi produk pada *website* penjualan aplikasi *e-commerce* Toko Gareu Shoes = (631: (150*5) x 100% = 84% dari yang diharapkan (100%).

Secara kontinum dapat digambarkan seperti berikut:



Gambar 4.3 *Skala Likert* Pengujian Kuesioner Promosi Produk (Pengunjung)

Berdasarkan data yang diperoleh dari 30 responden maka rata-rata 631 persen terletak pada daerah sangat setuju.

3. Kuesioner Pengujian Betha untuk *Member*

Berikut ini diberikan beberapa pertanyaan yang diberikan kepada *member* Toko Gareu Shoes, tentang tanggapan terhadap *user friendly* dari aplikasi *e-commerce* penjualan *online* pada Toko Gareu Shoes adalah sebagai berikut :

1. Apakah anda setuju tampilan *website http://gareushoes.com* dan mudah dipahami?
2. Apakah anda setuju dalam menggunakan aplikasi *website* penjualan *online http://gareushoes.com* dapat dioperasikan dengan mudah?
3. Apakah anda setuju penggunaan aplikasi *website* penjualan *online* http://gareushoes.com dapat dipelajari dengan mudah?

Hasil Kuesioner yang dibagikan kepada 30 orang member dengan 3 pertanyaan tentang *user friendly* menggunakan 5 pilihan jawaban pada setiap pertanyaan dapat dilihat pada Tabel 4.27 dan Tabel 4.28.

Tabel 4.27 Hasil Pengujian Kuesioner *User Frendly* (Member)

No.Soal	Sangat Setuju	Setuju	Ragu ragu	Tidak setuju	Sangat tidak setuju
1.	9	16	5	0	0
2.	15	12	3	0	0
3.	8	16	6	0	0
4.	9	15	6	0	0
5.	15	11	4	0	0
Total	56	70	24	0	0

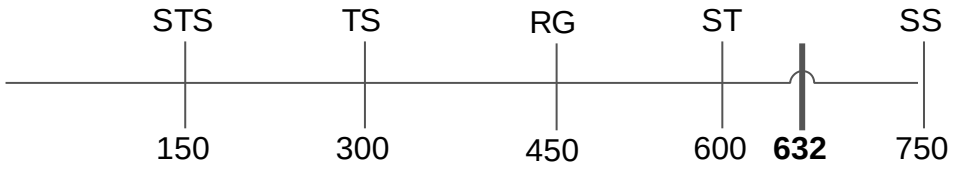
Sedangkan hasil dari perhitungan pada setiap jawaban dapat dilihat pada Tabel 4.27 .

Tabel 4.27 Pengukuran Pengujian Kuesioner *User Frendly* (*Member*)

Keterangan	N	R	N x R
Sangat setuju	5	56	280
Setuju	4	70	280
Ragu ragu	3	24	72
Tidak setuju	2	0	0
Sangat tidak setuju	1	0	0
		150	632

Berdasarkan data dari hasil kuesioner tersebut jumlah skor ideal (kriterium) untuk seluruh item = 5 x 30 = 150 (seandainya semua jawaban SS). Jumlah skor yang diperoleh dari pertanyaan mengenai proses *user frendly* = 632. Jadi berdasarkan data tersebut maka tingkat persetujuan terhadap *user frendly* pada *website* penjualan aplikasi *e-commerce* Toko Gareu Shoes = (632 : (150*5) x 100% = 84% dari yang diharapkan (100%).

Secara kontinum dapat digambarkan pada Gambar 4.3:



Gambar 4.3 *Skala Likert* Pengujian Kuesioner *User Frendly* (Pengunjung)

Jadi berdasarkan data yang diperoleh dari 30 responden mengenai *user frendly* maka rata-rata 632 terletak pada daerah sangat setuju.

Berikut ini diberikan beberapa pertanyaan yang diberikan kepada *member* Toko Gareu Shoes, tentang tanggapan terhadap proses pembelian produk dari aplikasi *e-commerce* penjualan *online* pada Toko Gareu Shoes adalah sebagai berikut:

4. Apakah *website http://gareushoes.com* memberikan anda kemudahan dalam proses pembelian produk?
5. Apakah anda setuju dengan adanya *website* penjualan *online http://gareushoes.com* dapat mengefektifkan waktu dan biaya dalam transaksi pembelian produk?
6. Apakah anda setuju dengan sistem pembayaran melalui paypal dan transfer bank yang diterapkan pada *website http://gareushoes.com* dapat mempermudah dalam melakukan pembayaran?
7. Apakah anda setuju bahwa semua informasi transaksi yang dikirimkan melalui email oleh *website http://gareushoes.com* sudah lengkap?
8. Apakah anda setuju bahwa aplikasi web ini dapat memberikan rekomendasi produk yang akan dibeli?

Hasil Kuesioner yang dibagikan kepada 10 orang pengunjung dengan 5 pertanyaan dan menggunakan 5 pilihan jawaban pada setiap pertanyaan dapat dilihat pada Tabel 4.27 dan Tabel 4.28.

Tabel 4.27 Hasil Pengujian Kuesioner untuk Member

No.Soal	Sangat Setuju	Setuju	Ragu ragu	Tidak setuju	Sangat tidak setuju
1.	8	16	6	0	0
2.	14	13	3	0	0
3.	8	16	6	0	0
4.	9	11	10	0	0
5.	6	16	8	0	0
Total	45	72	33	0	0

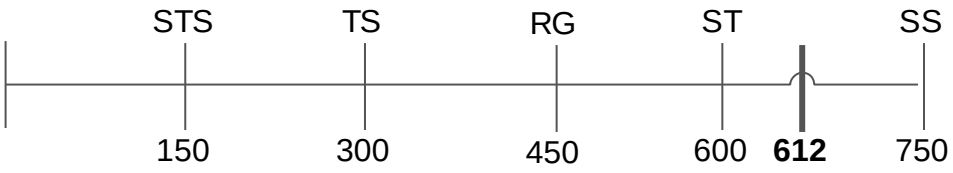
Sedangkan hasil dari perhitungan pada setiap jawaban dapat dilihat pada Tabel 4.27 .

Tabel 4.27 Pengukuran Pengujian Proses Pembelian (*Member*)

Keterangan	N	R	N x R
Sangat setuju	5	45	225
Setuju	4	72	288
Ragu ragu	3	33	99
Tidak setuju	2	0	0
Sangat tidak setuju	1	0	0
		150	612

Berdasarkan data dari hasil kuesioner tersebut jumlah skor ideal (kriterium) untuk seluruh item = 5 x 30 = 150 (seandainya semua jawaban SS). Jumlah skor yang diperoleh dari pertanyaan mengenai proses pembelian produk = 612. Jadi berdasarkan data tersebut maka tingkat persetujuan terhadap proses pembelian produk pada *website* penjualan aplikasi *e-commerce* Toko Gareu Shoes = (612 : (150*5) x 100% = 82% dari yang diharapkan (100%).

Secara kontinum dapat digambarkan seperti berikut:



Gambar 4.3 *Skala Likert* Pengujian Kuesioner Proses Pembelian (*Member*)

Berdasarkan data yang diperoleh dari 30 responden maka rata-rata 612 terletak pada daerah sangat setuju.

B. Wawancara Pengujiaan Betha Untuk Administrator

Daftar pertanyaan yang diberikan kepada administrator yang dipegang oleh pemilik usaha Toko Gareu Shoes adalah sebagai berikut:

1. Menurut pendapat anda, apakah *website* penjualan *online* http://gareushoes.com sudah sesuai dengan keinginan anda sebagai *supervisor* kepala toko ?

   Jawaban: Sesuai, namun saya membutuhkan bimbingan tentang penggunaan dari aplikasi web ini secara keseluruhan.

2. Menurut pendapat anda, apakah anda setuju website penjualan *online* http://gareushoes.com membantu anda dalam manajemen data operator ?

   Jawaban: Setuju, karena dengan adanya aplikasi web ini saya sebagai pengelola dapat memberikan hak akses sesuai dengan *job* deskripsi masing-masing.

3. Menurut pendapat anda, apakah anda setuju dengan adanya website penjualan *online* http://gareushoes.com memudahkan dalam pengolahan data katalog produk ?

   Jawaban: Setuju, dengan adanya aplikasi *Web* ini membantu pekerjaan saya sebagai pengelola dalam pengolahan produk.

4. Menurut pendapat anda, apakah anda setuju dengan adanya website penjualan *online* http://gareushoes.com membantu dalam proses tracking pengiriman barang ?

   Jawaban: Setuju, dengan adanya fasilitas tracking pengiriman dari aplikasi ini membantu saya dalam memeriksa status pengiriman kepada pelanggan tanpa harus membuka situs jasa pengiriman yang digunakan.

5. Menurut pendapat anda, apakah anda setuju dengan pengolahan data pesanan, data produk, dan retur produk pada web penjualan *online http://gareushoes.com* sesuai dengan prosedur yang berlaku di perusahaan anda ?

   Jawaban: Setuju, setelah saya mempelajari aplikasi web ini sudah sesuai dengan prosedur yang sedang berjalan.

C. Wawancara Pengujiaan Betha Untuk Bagian *Supervisor* Penjualan

Daftar pertanyaan yang diberikan kepada supervisor penjualan adalah sebagai berikut:

1. Menurut pendapat anda, apakah anda setuju bahwa website penjualan *online* http://gareushoes.com menghasilkan output yang sesuai dengan keinginan anda sebagai supervisor penjualan?

   Jawaban: Setuju, beberapa hasil data yang ditampilkan sangat membantu pekerjaan saya.

2. Menurut pendapat anda, apakah anda setuju website penjualan *online* http://gareushoes.com mempermudah dalam pengolahan data laporan produk dan laporan pembelian?

   Jawaban : Setuju, dengan adanya aplikasi web ini dapat membantu saya dalam proses pembuatan laporan.

3. Menurut pendapat anda, apakah anda setuju dengan adanya *website* penjualan *online* http://gareushoes.com membantu dalam proses *tracking* pengiriman barang?

   Jawaban: Setuju, dengan adanya fasilitas tracking pengiriman saya dapat mengecek status pengiriman kepada pelanggan kami.

4. Menurut pendapat anda, apakah anda setuju dengan *website* penjualan *online* http://gareushoes.com membantu anda dalam pengolahan laporan?

   Jawaban: Setuju, dengan adanya aplikasi web ini mambantu saya dalam pengolahan laporan kepada atasan saya.

5. Menurut pendapat anda, apakah anda setuju dengan adanya *website* penjualan *online* http://gareushoes.com telah sesuai dengan prosedur pesanan dan retur barang yang berlaku ?

   Jawaban: Setuju, apliasi web ini telah sesuai dengan prosedure yang sedang.

D. Wawancara Pengujiaan Betha untuk Bagian Operator Penjualan

Daftar pertanyaan yang diberikan kepada operator penjualan adalah sebagai berikut:

1. Menurut pendapat anda, apakah anda setuju bahwa website penjualan *online* http://gareushoes.com menghasilkan output yang sesuai dengan keinginan anda sebagai operator penjualan?

   Jawaban: Setuju, pengolahan pesanan pada apliasi web ini telah sesuai dengan keinginan saya.

2. Menurut pendapat anda, apakah anda setuju website penjualan *online* http://gareushoes.com mempermudah dalam pengolahan data pembelian?

Jawaban: Setuju, dengan adanya aplikasi web ini membantu pekerjaan saya dalam pengolahan pembelian pada konfirmasi pembayaran dari pelanggan.

3. Menurut pendapat anda, apakah anda setuju dengan adanya *website* penjualan *online* http://gareushoes.com membantu dalam proses pengolahan konfirmasi pembayaran dari pelanggan?

   Jawaban: Setuju, dengan adanya aplikasi web ini proses konfirmasi pembayaran dari pelanggan dapat saya layani dengan cepat.

4. Menurut pendapat anda, apakah anda setuju dengan *website* penjualan *online* http://gareushoes.com membantu anda dalam pengolahan laporan?

   Jawaban: Setuju, dengan adanya aplikasi web ini membantu saya dan supervisor penjualan dalam pengolahan laporan transaksi.

5. Menurut pendapat anda, apakah anda setuju dengan adanya *website* penjualan *online* http://gareushoes.com telah sesuai dengan prosedur pembelian?

   Jawaban: Setuju, aplikasi web ini telah sesuai dengan prosedur yang sedang berjalan.

4.2.2.2 Kesimpulan Hasil Pengujian Beta

Berdasarkan hasil persentase yang diperoleh dari pengujian beta pengunjung, member dan wawancara, sebagian besar mengatakan sangat setuju terhadap penerapan sistem yang baru yang memiliki kandungan poin syarat *user friendly* dan sesuai dengan tujuan yang ingin dicapai, yaitu memudahkan pihak Toko Gareu Shoes dan pelanggan dalam layanan transaksi pembelian yang tidak mengenal batasan waktu, mempermudah pihak Toko Gareu Shoes dalam melakukan pengolahan transaksi, manajemen stok dan pelaporan transaksi, memudahkan mempromosikan produk yang menyajikan informasi produk secara detail, *up-to-date,* dan dapat diakses oleh siapa saja secara *online* melalui *internet,* memberikan suatu layanan rekomendasi yang memudahkan pelanggan dalam menentukan produk yang dibeli berdasarkan dari *rating* pelanggan lain.