BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Implementasi merupakan tahap menerjemahkan perancangan berdasarkan hasil analisis. Tujuan implementasi adalah untuk mengkonfirmasikan modul program perancangan pada para pelaku sistem sehingga *user* dapat memberikan masukan kepada pengembang sistem.

4.1.1 Perangkat Keras Yang Digunakan

Perangkat keras yang digunakan untuk mengimplementasikan sistem ini dijelaskan pada tabel 4.1 sebagai berikut :

Tabel 4. 1 Perangkat keras yang digunakan

No.	Perangkat Keras	Spesifikasi Minimum	Eksistensi	Keterangan
1	*Processor*	Intel Pentium IV 1,3 GHz	Dual Core T4500 @2.30GHz	Sudah melebihi batas minimum
2	RAM	256 MB	1 GB	Sudah melebihi batas minimum
3	Harddisk	40 GB	250 GB	Sudah melebihi batas minimum

4.1.2 Perangkat Lunak Yang Digunakan

Perangkat lunak yang digunakan dalam mengimplementasikan sistem ini dijelaskan pada tabel 4.2 sebagai berikut :

Tabel 4. 2 Perangkat lunak yang digunakan

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows XP profesional
2	PHP	Bahasa Pemrograman
3	MySQL	Penyimpanan *database*
4	PhpMyAdmin	*interface database*
5	WampServer 2.0	*web server*
6	Macromedia Dreamweaver 8	Kode dan perancangan tampilan
7	Mozilla Firefox, Google Chrome, Opera, Internet Explorer	*Web browser*

4.2 Implementasi Basis Data

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

1. **Pembuatan *database*:**

```
CREATE DATABASE `genicksh_dbgenick`;
```

2. **Pembuatan tabel admin**

```
CREATE TABLE IF NOT EXISTS `admin` (
  `id_admin` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(50) NOT NULL,
  `password` varchar(50) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `email` varchar(100) DEFAULT NULL,
  `jabatan` varchar(50) DEFAULT NULL,
  `status` enum('Y','T') NOT NULL,
  PRIMARY KEY (`id_admin`),
  UNIQUE KEY `username` (`username`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

3. Pembuatan tabel provinsi

```
CREATE TABLE IF NOT EXISTS `provinsi` (
  `id_provinsi` int(11) NOT NULL AUTO_INCREMENT,
  `id_admin` int(11) NOT NULL,
  `nama_provinsi` varchar(60) NOT NULL,
  PRIMARY KEY (`id_provinsi`),
  UNIQUE KEY `u_provinsi` (`nama_provinsi`),
  KEY `id_admin` (`id_admin`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `provinsi`
  ADD    CONSTRAINT    `provinsi_ibfk_1`    FOREIGN    KEY
(`id_admin`) REFERENCES `admin` (`id_admin`);
```

4. Pembuatan tabel kota

```
   CREATE TABLE IF NOT EXISTS `kota` (
     `id_kota` int(11) NOT NULL AUTO_INCREMENT,
     `id_provinsi` int(11) NOT NULL,
     `id_admin` int(11) NOT NULL,
     `nama_kota` varchar(60) NOT NULL,
     PRIMARY KEY (`id_kota`),
     KEY `id_admin` (`id_admin`),
     KEY `id_provinsi` (`id_provinsi`)
   ) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
   ALTER TABLE `kota`
     ADD    CONSTRAINT    `kota_ibfk_4`    FOREIGN    KEY
   (`id_provinsi`) REFERENCES `provinsi` (`id_provinsi`),
     ADD CONSTRAINT `kota_ibfk_5` FOREIGN KEY (`id_admin`)
   REFERENCES `admin` (`id_admin`);
```

5. Pembuatan tabel member

```
CREATE TABLE IF NOT EXISTS `member` (
  `id_member` varchar(100) NOT NULL,
  `password` varchar(50) NOT NULL,
  `id_kota` int(11) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `alamat` text,
  `telp` varchar(15) DEFAULT NULL,
  `kodepos` varchar(10) DEFAULT NULL,
  `kodeaktivasi` varchar(50) DEFAULT NULL,
  `status` enum('T','Y','B') NOT NULL DEFAULT 'T',
  PRIMARY KEY (`id_member`),
  KEY `id_kota` (`id_kota`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `member`
  ADD CONSTRAINT `member_ibfk_1` FOREIGN KEY (`id_kota`)
REFERENCES `kota` (`id_kota`);
```

6. Pembuatan tabel jasa_kirim

```
CREATE TABLE IF NOT EXISTS `jasa_kirim` (
  `id_jasakirim` int(11) NOT NULL AUTO_INCREMENT,
  `id_admin` int(11) NOT NULL,
  `nama_jasa` varchar(20) NOT NULL,
  `lama` int(11) DEFAULT NULL,
  PRIMARY KEY (`id_jasakirim`),
  UNIQUE KEY `u_jasa` (`nama_jasa`),
  KEY `id_admin` (`id_admin`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `jasa_kirim`
  ADD   CONSTRAINT   `jasa_kirim_ibfk_1`   FOREIGN   KEY
(`id_admin`) REFERENCES `admin` (`id_admin`);
```

7. Pembuatan tabel tarif_kirim

```
CREATE TABLE IF NOT EXISTS `tarif_kirim` (
  `id_tarifkirim` int(11) NOT NULL AUTO_INCREMENT,
  `id_admin` int(11) NOT NULL,
  `id_kota` int(11) NOT NULL,
  `id_jasakirim` int(11) NOT NULL,
  `biaya` decimal(11,0) DEFAULT NULL,
  PRIMARY KEY (`id_tarifkirim`),
  KEY `id_admin` (`id_admin`),
  KEY `id_kota` (`id_kota`),
  KEY `id_jasakirim` (`id_jasakirim`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `tarif_kirim`
  ADD   CONSTRAINT   `tarif_kirim_ibfk_7`   FOREIGN   KEY
(`id_admin`) REFERENCES `admin` (`id_admin`),
  ADD   CONSTRAINT   `tarif_kirim_ibfk_8`   FOREIGN   KEY
(`id_kota`) REFERENCES `kota` (`id_kota`),
  ADD   CONSTRAINT   `tarif_kirim_ibfk_9`   FOREIGN   KEY
(`id_jasakirim`)          REFERENCES          `jasa_kirim`
(`id_jasakirim`);
```

8. Pembuatan tabel kategori

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` int(11) NOT NULL AUTO_INCREMENT,
  `id_admin` int(11) NOT NULL,
  `nama_kategori` varchar(50) NOT NULL,
  PRIMARY KEY (`id_kategori`),
  UNIQUE KEY `u_kategori` (`nama_kategori`),
  KEY `id_admin` (`id_admin`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `kategori`
  ADD    CONSTRAINT    `kategori_ibfk_1`    FOREIGN    KEY
(`id_admin`) REFERENCES `admin` (`id_admin`);
```

9. Pembuatan tabel ukuran

```
CREATE TABLE IF NOT EXISTS `ukuran` (
  `kode_ukuran` int(11) NOT NULL AUTO_INCREMENT,
  `id_admin` int(11) NOT NULL,
  `nama_ukuran` varchar(25) NOT NULL,
  PRIMARY KEY (`kode_ukuran`),
  UNIQUE KEY `nama_ukuran` (`nama_ukuran`),
  KEY `id_admin` (`id_admin`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `ukuran`
  ADD CONSTRAINT `ukuran_ibfk_1` FOREIGN KEY (`id_admin`)
REFERENCES `admin` (`id_admin`);
```

10. Pembuatan tabel warna

```
 CREATE TABLE IF NOT EXISTS `warna` (
   `id_warna` int(11) NOT NULL AUTO_INCREMENT,
   `id_admin` int(11) NOT NULL,
   `nama_warna` varchar(30) NOT NULL,
   `id_spec_warna` int(11) NOT NULL,
   PRIMARY KEY (`id_warna`),
   UNIQUE KEY `u_warna` (`nama_warna`),
   KEY `id_admin` (`id_admin`),
   KEY `id_detail_warna` (`id_spec_warna`)
 ) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
 ALTER TABLE `warna`
   ADD CONSTRAINT `warna_ibfk_1` FOREIGN KEY (`id_admin`)
 REFERENCES `admin` (`id_admin`),
 ADD     CONSTRAINT     `warna_ibfk_2`     FOREIGN     KEY
 (`id_spec_warna`)     REFERENCES     `spesifikasi_warna`
 (`id_spec_warna`);
```

11. Pembuatan tabel spesifikasi warna

```
CREATE TABLE IF NOT EXISTS `spesifikasi_warna` (
  `id_spec_warna` int(11) NOT NULL AUTO_INCREMENT,
  `nama_detail_warna` varchar(50) NOT NULL,
  PRIMARY KEY (`id_spec_warna`),
  UNIQUE KEY `nama_detail_warna` (`nama_detail_warna`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

12. Pembuatan tabel spesifikasi ukuran

```
CREATE TABLE IF NOT EXISTS ` spesifikasi _ukuran` (
  `id_spec_ukuran` int(11) NOT NULL AUTO_INCREMENT,
  `kode_ukuran` int(11) NOT NULL,
  `deskripsi_ukuran` varchar(100) NOT NULL,
  PRIMARY KEY (`id_spec_ukuran`),
```

```
  KEY `kode_ukuran` (`kode_ukuran`),
  KEY `id_kategori` (`id_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `spec_ukuran`
  ADD   CONSTRAINT   `spec_ukuran_ibfk_1`   FOREIGN   KEY
   (`kode_ukuran`) REFERENCES `ukuran` (`kode_ukuran`);
```

13. Pembuatan tabel produk

```
CREATE TABLE IF NOT EXISTS `produk` (
  `id_produk` int(11) NOT NULL AUTO_INCREMENT,
  `id_kategori` int(11) NOT NULL,
  `id_admin` int(11) NOT NULL,
  `nama_produk` varchar(50) NOT NULL,
  `harga` decimal(11,0) DEFAULT NULL,
  `hargabeli` decimal(11,0) DEFAULT NULL,
  `deskripsi` text,
  `diskon` float NOT NULL DEFAULT '0',
  `berat` float DEFAULT NULL,
  PRIMARY KEY (`id_produk`),
  UNIQUE KEY `u_produk` (`nama_produk`),
  KEY `id_admin` (`id_admin`),
  KEY `id_kategori` (`id_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `produk`
  ADD    CONSTRAINT    `produk_ibfk_1`    FOREIGN    KEY
(`id_kategori`) REFERENCES `kategori` (`id_kategori`),
  ADD CONSTRAINT `produk_ibfk_2` FOREIGN KEY (`id_admin`)
REFERENCES `admin` (`id_admin`);
```

14. Pembuatan tabel gambar

```
CREATE TABLE IF NOT EXISTS `gambar` (
  `id_gambar` int(11) NOT NULL AUTO_INCREMENT,
  `id_produk` int(11) NOT NULL,
  `file_gambar` varchar(100) NOT NULL,
  `nama_gambar` varchar(30) NOT NULL,
  PRIMARY KEY (`id_gambar`),
  KEY `id_produk` (`id_produk`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `gambar`
  ADD    CONSTRAINT    `gambar_ibfk_1`    FOREIGN    KEY
(`id_produk`) REFERENCES `produk` (`id_produk`);
```

15. Pembuatan tabel detail_produk

```
CREATE TABLE IF NOT EXISTS `detail_produk` (
  `id_detailproduk` int(11) NOT NULL AUTO_INCREMENT,
  `kode_ukuran` int(11) NOT NULL,
  `id_warna` int(11) NOT NULL,
```

```
  `id_produk` int(11) NOT NULL,
  `stok` int(11) NOT NULL DEFAULT '0',
  `tanggal_msk` datetime DEFAULT NULL,
  `tanggal_update` datetime DEFAULT NULL,
  PRIMARY KEY (`id_detailproduk`),
  KEY `id_produk` (`id_produk`),
  KEY `id_warna` (`id_warna`),
  KEY `kode_ukuran` (`kode_ukuran`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `detail_produk`
  ADD   CONSTRAINT   `detail_produk_ibfk_1`   FOREIGN   KEY
(`kode_ukuran`) REFERENCES `ukuran` (`kode_ukuran`),
  ADD   CONSTRAINT   `detail_produk_ibfk_2`   FOREIGN   KEY
(`id_warna`) REFERENCES `warna` (`id_warna`),
  ADD   CONSTRAINT   `detail_produk_ibfk_3`   FOREIGN   KEY
(`id_produk`) REFERENCES `produk` (`id_produk`);
```

16. Pembuatan tabel pemesanan

```
CREATE TABLE IF NOT EXISTS `pemesanan` (
  `id_pemesanan` varchar(8) NOT NULL,
  `id_member` varchar(50) NOT NULL,
  `totalbayar` decimal(11,0) DEFAULT NULL,
  `tglpesan` datetime DEFAULT NULL,
  `status` varchar(20) DEFAULT NULL,
  `total_item` int(11) DEFAULT NULL,
  `tgl_expired` datetime DEFAULT NULL,
  PRIMARY KEY (`id_pemesanan`),
  KEY `id_member` (`id_member`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `pemesanan`
  ADD    CONSTRAINT    `pemesanan_ibfk_1`    FOREIGN    KEY
(`id_member`) REFERENCES `member` (`id_member`);
```

17. Pembuatan tabel detail_pemesanan

```
CREATE TABLE IF NOT EXISTS `detail_pemesanan` (
  `id_detailpesan` int(11) NOT NULL AUTO_INCREMENT,
  `id_detailproduk` int(11) NOT NULL,
  `id_pemesanan` varchar(8) NOT NULL,
  `harga` decimal(11,0) DEFAULT NULL,
  `qty` int(11) DEFAULT NULL,
  `diskon` float DEFAULT NULL,
  PRIMARY KEY (`id_detailpesan`),
  KEY `id_pemesanan` (`id_pemesanan`),
  KEY `id_detailproduk` (`id_detailproduk`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `detail_pemesanan`
  ADD  CONSTRAINT  `detail_pemesanan_ibfk_4`  FOREIGN  KEY
  (`id_pemesanan`)          REFERENCES          `pemesanan`
  (`id_pemesanan`),
  ADD  CONSTRAINT  `detail_pemesanan_ibfk_5`  FOREIGN  KEY
  (`id_detailproduk`)      REFERENCES      `detail_produk`
  (`id_detailproduk`);
```

18. Pembuatan tabel shipping

```
CREATE TABLE IF NOT EXISTS `shipping` (
  `id_shipping` int(11) NOT NULL AUTO_INCREMENT,
  `id_kota` int(11) NOT NULL,
  `id_pemesanan` varchar(15) NOT NULL,
  `id_tarifkirim` int(11) NOT NULL,
  `nama_penerima` varchar(50) NOT NULL,
  `alamat_penerima` text,
  `kodepos` varchar(10) DEFAULT NULL,
  `telepon` varchar(15) DEFAULT NULL,
  PRIMARY KEY (`id_shipping`),
  KEY `id_kota` (`id_kota`),
  KEY `id_pemesanan` (`id_pemesanan`),
  KEY `id_tarifkirim` (`id_tarifkirim`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `shipping`
  ADD    CONSTRAINT    `shipping_ibfk_4`    FOREIGN    KEY
(`id_kota`) REFERENCES `kota` (`id_kota`),
  ADD    CONSTRAINT    `shipping_ibfk_5`    FOREIGN    KEY
(`id_pemesanan`) REFERENCES `pemesanan` (`id_pemesanan`),
  ADD    CONSTRAINT    `shipping_ibfk_6`    FOREIGN    KEY
(`id_tarifkirim`)        REFERENCES        `tarif_kirim`
(`id_tarifkirim`);
```

19. Pembuatan tabel faktur_penjualan

```
CREATE TABLE IF NOT EXISTS `faktur_penjualan` (
  `id_faktur` varchar(8) NOT NULL,
  `id_pemesanan` varchar(8) NOT NULL,
  `tgl_bayar` datetime DEFAULT NULL,
  `jenis_bayar` varchar(20) DEFAULT NULL,
  `bank_pengirim` varchar(50) DEFAULT NULL,
  `no_rek` varchar(25) DEFAULT NULL,
  `jml_bayar` decimal(11,0) DEFAULT NULL,
  `atas_nama` varchar(50) DEFAULT NULL,
  `no_trans_bayar` varchar(30) DEFAULT NULL,
  `no_resi` varchar(20) DEFAULT NULL,
  `status` varchar(20) DEFAULT NULL,
  `tgl_kirim` date DEFAULT NULL,
  `tgl_terima` date DEFAULT NULL,
  `bank_tujuan` enum('BNI','BCA') DEFAULT NULL,
  `jml_usd` decimal(10,0) DEFAULT NULL,
  `penerima` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`id_faktur`),
  KEY `id_pemesanan` (`id_pemesanan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `faktur_penjualan`
  ADD  CONSTRAINT  `faktur_penjualan_ibfk_1`  FOREIGN  KEY
(`id_pemesanan`) REFERENCES `pemesanan` (`id_pemesanan`);
```

20. Pembuatan tabel retur

```
CREATE TABLE IF NOT EXISTS `retur` (
  `id_retur` varchar(8) NOT NULL,
  `id_faktur` varchar(15) NOT NULL,
  `tgl_retur` datetime DEFAULT NULL,
  `tgl_kirim_retur` datetime DEFAULT NULL,
  `status` varchar(20) DEFAULT NULL,
  `resi_retur` varchar(50) DEFAULT NULL,
  `tgl_terima_retur` datetime DEFAULT NULL,
  `opsi_retur` enum('Tukar Barang Lain','Uang Kembali')
   DEFAULT NULL,
  `bank_penerima` varchar(50) DEFAULT NULL,
  `no_rek_penerima` varchar(25) DEFAULT NULL,
  PRIMARY KEY (`id_retur`),
  KEY `id_faktur` (`id_faktur`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `retur`
  ADD CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`id_faktur`)
REFERENCES `faktur_penjualan` (`id_faktur`);
```

21. Pembuatan tabel detail_ retur

```
CREATE TABLE IF NOT EXISTS `detail_retur` (
  `id_detailretur` int(11) NOT NULL AUTO_INCREMENT,
  `id_retur` varchar(15) NOT NULL,
  `id_detailpesan` int(11) NOT NULL,
  `qty` int(11) DEFAULT NULL,
  `deskripsi` text,
  `id_produk_pilihan` int(11) DEFAULT NULL,
  PRIMARY KEY (`id_detailretur`),
  KEY `id_retur` (`id_retur`),
  KEY `id_detailproduk` (`id_detailpesan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `detail_retur`
  ADD   CONSTRAINT   `detail_retur_ibfk_1`   FOREIGN   KEY
   (`id_detailpesan`)    REFERENCES    `detail_pemesanan`
   (`id_detailpesan`),
  ADD   CONSTRAINT   `detail_retur_ibfk_2`   FOREIGN   KEY
   (`id_retur`) REFERENCES `retur` (`id_retur`);
```

4.3 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan menampilkan setiap tampilan sistem yang dibangun dan pengkodeannya dalam bentuk *file* program. Adapun untuk tampilan implementasi Pembangunan Website *E-Commerce* Penjualan

Produk *Fashion* Di Toko Genick akan dilampirkan pada halaman lampiran dan deskripsi implementasi antarmuka untuk pengguna adalah sebagai berikut :

1. Implementasi Antarmuka Pengunjung dijelaskan pada tabel 4.3

Tabel 4. 3 Deskripsi Implementasi Antarmuka Pengunjung

Menu	Deskripsi	Nama File
Home	Digunakan untuk menampilkan menu-menu yang ada pada sistem	Index.php
Daftar	Digunakan untuk melakukan pendaftaran member baru.	daftar.php
Produk	Digunakan untuk menampilkan katalog produk yang dijual	Produk.php
Bantuan	Digunakan untuk menampilkan informasi tentang tata cara pemesanan, pembayaran, serta retur	Bantuan.php
Tarif kirim	Digunakan untuk menampilkan informasi tarif kirim yang tersedia	Tarifkirim.php
Tentang kami	Digunakan untuk menampilkan informasi tentang profil perusahaan	Tentangkami.php
Login member	Digunakan sebagai halaman login member	login.php
Lupa password	Digunakan jika member lupa dengan password	Lupa_password. php
Kategori	Digunakan untuk melihat katalog produk perkategori	kategori. php
Detail Produk	Digunakan untuk melihat, data produk secara detail	Detail_produk.php

2. Implementasi Antarmuka Member dijelaskan pada tabel 4.4

Tabel 4. 4 Deskripsi Implementasi Antarmuka Member

Menu	Deskripsi	Nama File
Home	Digunakan untuk menampilkan menu-menu yang ada pada sistem	Index.php

Menu	Deskripsi	Nama File
Produk	Digunakan untuk menampilkan katalog produk yang dijual	Produk.php
Bantuan	Digunakan untuk menampilkan informasi tentang tata cara pemesanan, pembayaran, serta retur	Bantuan.php
Tarif kirim	Digunakan untuk menampilkan informasi tarif kirim yang tersedia	Tarifkirim.php
Tentang kami	Digunakan untuk menampilkan informasi tentang profil perusahaan	Tentangkami.php
Login member	Digunakan sebagai halaman login member	login.php
Lupa password	Digunakan jika member lupa dengan password	Lupa_password. php
Profil	Digunakan untuk melihat data profil member	Profil_member.php
Riwayat pemesanan	Digunakan untuk melihat *history* pemesanan yang pernah dilakukan.	history.php
Retur	Digunakan untuk melakukan retur barang, jika barang yang dipesan tidak sesuai	Retur.php
Ubah Profil	Digunakan untuk mengubah data profil member.	profil_member_ubah.php
Ganti password	Digunakan jika member ingin mengubah password	member_password_ ubah.php
Kategori	Digunakan untuk melihat katalog produk perkategori	kategori. php
Detail Produk	Digunakan untuk melihat, data produk secara detail	Detail_produk.php
Pembayaran	Digunakan untuk melakukan konfirmasi pembayaran	Pembayaran.php
Keranjang belanja	Digunakan untuk melihat, mengubah, menghapus data pemesanan yang akan dilakukan.	Keranjang.php

3. Implementasi Antarmuka Manajer dijelaskan pada tabel 4.5

Tabel 4. 5 Deskripsi Implementasi Antarmuka Manajer

Menu	Deskripsi	Nama File
Login manajer	Digunakan sebagai halaman login manajer	index.php
Lupa Password	Digunakan jika manajer lupa terhadap passwordnya	Lupa_password.php
Halaman utama	Digunakan untuk menampilkan, menu-menu yang ada pada sistem	Home1.php
Pengolahan admin	Digunakan untuk melihat, menambah, dan mengubah data admin	admin_view.php
Tambah admin	Digunakan untuk menambah data admin	Admin_tambah.php
Laporan	Digunakan untuk melihat, mencetak laporan	Laporan_penjualan.php
Data member	Digunakan untuk melihat data member	member_view.php
Backup	Digunakan untuk membackup data	backup.php
Restore	Digunakan untuk mengembalikan data	restore.php
Logout	Digunakan untuk dapat keluar dari sistem	logout.php

4. Implementasi Antarmuka Karyawan Bagian Kasir dijelaskan pada tabel 4.6

Tabel 4. 6 Deskripsi Implementasi Antarmuka Karyawan Bagian Kasir

Menu	Deskripsi	Nama File
Login karyawan	Digunakan sebagai halaman login karyawan	index.php
Lupa Password	Digunakan jika karyawan lupa terhadap passwordnya	Lupa_password.php
Halaman utama	Digunakan untuk menampilkan, menu-menu yang ada pada sistem	Home.php

Menu	Deskripsi	Nama File
Profil	Digunakan untuk menampilkan data profil karyawan	Profil.php
Ganti password	Digunakan untuk mengubah password karyawan	admin_ubah_pass.php
Pengolahan kategori	Digunakan untuk melihat, menambah, dan mengubah data kategori	Kategori_view.php
Pengolahan ukuran	Digunakan untuk melihat, menambah, dan mengubah data ukuran	Ukuran_view.php
Pengolahan warna	Digunakan untuk melihat, menambah, dan mengubah data warna	Warna_view.php
Pengolahan Kota	Digunakan untuk melihat, menambah, dan mengubah data kota	Kota_view. php
Pengolahan Provinsi	Digunakan untuk melihat, menambah, dan mengubah data provinsi	Provinsi_view.php
Pengolahan Jasa kirim	Digunakan untuk melihat, menambah, dan mengubah data jasa kirim	Jasa_view.php
Pengolahan Tarif kirim	Digunakan untuk melihat, menambah, dan mengubah data tarif kirim	tarif_view.php
Pengolahan Produk	Digunakan untuk melihat, menambah, dan mengubah data produk	Produk_view.php
Pengolahan Gambar	Digunakan untuk melihat, menambah, dan mengubah data gambar	Gambar_view.php
Pengolahan detail produk	Digunakan untuk melihat, menambah, dan mengubah data detail produk	detail_view.php
Pengolahan pemesanan	Digunakan untuk melihat data transaksi pemesanan yang masuk dan meubah statusnya.	Transaksi.php
Pengolahan pengiriman	Digunakan untuk melihat data pengiriman dan mengubah statusnya	Olah_pengiriman.php
Pengolahan retur	Digunakan untuk melihat data retur dan mengubah status retur	Olah_retur.php
Logout	Digunakan karywan untuk dapat keluar dari sistem	logout.php

4.4 Pengujian Perangkat Lunak

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan kesalahan atau kekurangan pada perangkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut. Pengujian perangkat lunak ini menggunakan pengujian black box. Pengujian black box berfokus pada persyaratan fungsional perangkat lunak.

4.4.1 Pengujian Alpha

Pengujian alpha merupakan pengujian yang menitikberatkan pada hasil *output* dari kendali *input* yang dimasukkan pada tampilan kendali *input form* tampilan, dimana pengujian dikatakan berhasil apabila *output* sesuai dengan kendali *input* yang dimasukkan pada setiap tampilan.

4.4.1.1 Skenario Pengujian Alpha

1. Skenario pengujian alpha untuk member ditunjukkan pada tabel 4.7 berikut :

Tabel 4. 7 Skenario Pengujian Untuk Member

Kelas Uji	Butir Uji	Jenis Pengujian
Daftar *member*	Menampilkan *form* user baru	*Black Box*
	Isi *form* pendaftaran	*Black Box*
	Simpan data baru	*Black Box*
Login	Isi data login	*Black Box*
	Verifikasi *Email*	*Black Box*
	Validasi *password*	*Black Box*
Edit profil	Menampilkan *form* ubah profil	*Black Box*
	Ubah data *member*	*Black Box*
	Simpan perubahan	*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*

Kelas Uji	Butir Uji	Jenis Pengujian
Pencarian	Menampilkan data barang	*Black Box*
	Menampilkan data barang	*Black Box*
	Cari barang	*Black Box*
Menu keranjang belanja	Menampilkan data pembelian	*Black Box*
	Mengupdate data pembelian	*Black Box*
	Menghapus data pembelian	*Black Box*
Form konfirmasi	Menampilkan konfirmasi transaksi	*Black Box*
Form pengiriman	Menampilkan form pengiriman	*Black Box*
	Pilih jenis pengiriman	*Black Box*
Logout	Menghapus *session user*	*Black Box*

2. Skenario pengujian alpha untuk manajer ditunjukkan pada tabel 4.8 berikut :

Tabel 4. 8 skenario pengujian untuk manajer

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Lupa *password*	Menampilkan *form* Lupa *password*	*Black Box*
Data Laporan	Tampilkan data laporan	*Black Box*
	Cetak data laporan	*Black Box*
Pengolahan admin	Tampilkan data admin	*Black Box*
	Tambah data admin	*Black Box*
	Ubah data admin	*Black Box*
Data member	Tampilkan data member	*Black Box*
	Ubah data member	*Black Box*
Logout	Menghapus *session user*	*Black Box*

3. Skenario pengujian alpha untuk member ditunjukkan pada tabel 4.9 berikut :

Tabel 4. 9 skenario pengujian untuk karyawan bagian kasir

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Validasi *password*	*Black Box*
Lupa *password*	Menampilkan *form* Lupa *password*	*Black Box*
Pengolahan Transaksi	Tampilkan data transaksi	*Black Box*
	Ubah status transaksi	*Black Box*
Pengolahan Kategori	Tampilkan data kategori	*Black Box*
	Tambah Data katgori	*Black Box*
	Ubah data kategori	*Black Box*

Kelas Uji	Butir Uji	Jenis Pengujian
Pengolahan Provinsi	Tampilkan data Provinsi	*Black Box*
	Tambah data Provinsi	*Black Box*
	Ubah data Provinsi	*Black Box*
Pengolahan Kota	Tampilkan data kota	*Black Box*
	Tambah data kota	*Black Box*
	Ubah data kota	*Black Box*
Pengolahan Ukuran	Tampilkan data Ukuran	*Black Box*
	Tambah data Ukuran	*Black Box*
	Ubah data ukuran	*Black Box*
Pengolahan Detail Produk	Tampilkan detail produk	*Black Box*
	Tambah detail produk	*Black Box*
	Ubah detail produk	*Black Box*
Pengolahan Data Produk	Tampilkan data produk	*Black Box*
	Tambah data produk	*Black Box*
	Ubah data produk	*Black Box*
Pengolahan Data warna	Tampilkan data warna	*Black Box*
	Tambah data warna	*Black Box*
	Ubah data warna	*Black Box*
Pengolahan Data gambar	Tampilkan data gambar	*Black Box*
	Tambah data gambar	*Black Box*
	Ubah data gambar	*Black Box*
Pengolahan Data jasa kirim	Tampilkan data jasa kirim	*Black Box*
	Tambah data jasa kirim	*Black Box*
	Ubah data jasa kirim	*Black Box*
Pengolahan Data tarif kirim	Tampilkan data tarif kirim	*Black Box*
	Tambah data tarif kirim	*Black Box*
	Ubah data tarif kirim	*Black Box*
Logout	Menghapus *session user*	*Black Box*

4.4.2 Kasus dan Hasil Pengujian

1. Contoh Uji Daftar Member

   Berikut dapat dilihat pengujian dari pendaftaran *member* pada tabel 4.10

Tabel 4. 10 Pengujian Pendaftaran Member

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : felu.chey@yahoo.com Nama depan : Suci Nama belakang : Rahayu Alamat : Jalan Kubang Selatan 108 Provinsi : Jawa Barat Kota : Bandung Kodepos : 42314 Telepon : 08563636960 Password : 12345 Ulangi password : 12345	Menampilkan pesan "Terima Kasih Anda Telah Berhasil Registrasi. Silahkan cek email untuk melakukan Aktivasi"	Data tersimpan dan menampilkan pesan "Terima Kasih Anda Telah Berhasil Registrasi. Silahkan cek email untuk melakukan Aktivasi"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Email : felu.chey@yahoo.com Nama depan : Nama belakang : Rahayu Alamat : Jalan Kubang Selatan 108 Provinsi : Jawa Barat Kota : Bandung Kodepos : 42314 Telepon : 08563636960 Password : 12345 Ulangi password : 12345	Menampilkan pesan "Nama tidak boleh kosong"	Data tidak tersimpan dan menampilkan pesan "Nama tidak boleh kosong"	[√] Diterima [ ] Ditolak
Email : Nama depan : Suci Nama belakang : Rahayu Alamat : Jalan Kubang Selatan 108 Provinsi : Jawa Barat Kota : Bandung Kodepos : 42314 Telepon : 08563636960 Password : 12345 Ulangi password : 12345	Menampilkan pesan "Email tidak boleh kosong"	Data tidak tersimpan dan menampilkan pesan "Email tidak boleh kosong"	[√] Diterima [ ] Ditolak
Email : felu.chey@yahoo.com	Menampilkan	Data tidak	[√] Diterima

Nama depan : Suci Nama belakang : Rahayu Alamat : Provinsi : Jawa Barat Kota : Bandung Kodepos : 42314 Telepon : 08563636960 Password : 12345 Ulangi password : 12345	pesan"Alamat tidak boleh kosong"	tersimpan dan menampilkan pesan "Alamat tidak boleh kosong"	[ ] Ditolak
Email : felu.chey@yahoo.com Nama depan : Suci Nama belakang : Rahayu Alamat : Jalan Kubang Selatan Provinsi : Kota : Bandung Kodepos : 42314 Telepon : 08563636960 Password : 12345 Ulangi password : 12345	Menampilkan pesan "Provinsi harus dipilih"	Data tidak tersimpan dan menampilkan pesan "Provinsi harus dipilih"	[√] Diterima [ ] Ditolak
Email : felu.chey@yahoo.com Nama depan : Suci Nama belakang : Rahayu Alamat : Jalan Kubang Selatan 108 Provinsi : Jawa Barat Kota : Kodepos : 42314 Telepon : 08563636960 Password : 12345 Ulangi password : 12345	Menampilkan pesan "kota harus dipilih"	Data tidak tersimpan dan menampilkan pesan "kota harus dipilih"	[√] Diterima [ ] Ditolak
Email : felu.chey@yahoo.com Nama depan : Suci Nama belakang : Rahayu Alamat : Jalan Kubang Selatan 108 Provinsi : Jawa Barat Kota : Bandung Kodepos : Telepon : 08563636960 Password : 12345 Ulangi password : 12345	Menampilkan pesan "kode pos tidak boleh kosong"	Data tidak tersimpan dan menampilkan pesan "kode pos tidak boleh kosong"	[√] Diterima [ ] Ditolak
Email : felu.chey@yahoo.com Nama depan : Suci Nama belakang : Rahayu Alamat : Jalan Kubang Selatan 108	Menampilkan pesan "password harus diisi"	Data tidak tersimpan dan menampilkan pesan "password	[√] Diterima [ ] Ditolak

Provinsi : Jawa Barat Kota : Bandung Kodepos : 42314 Telepon : 08563636960 Password : Ulangi password :		harus diisi"	

2. Contoh Uji Ganti *Password* Member

Berikut adalah pengujian dari ganti *password* pada member pada tabel 4.11:

Tabel 4. 11 Pengujian Ganti Password Member

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Password Lama: 123456 Password Baru : 12345 Ulangi password baru : 12345	Jika semua data diisi dengan benar maka menampilkan pesan "Password baru telah disimpan"	Jika semua data diisi dengan benar maka menampilkan pesan "Password baru telah disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Password Lama: Password Baru : Ulangi password baru :	Menampilkan pesan "Anda harus mengisikan semua data pada form Ganti Password."	Data tidak tersimpan dan menampilkan pesan "Anda harus mengisikan semua data pada form Ganti Password."	[√] Diterima [ ] Ditolak
Password Lama: 1331412 Password Baru : Ulangi password baru :	Menampilkan pesan "Anda harus mengisikan semua data pada form Ganti Password."	Data tidak tersimpan dan menampilkan pesan "Anda harus mengisikan semua data pada form Ganti Password."	[√] Diterima [ ] Ditolak

3. Contoh Uji Keranjang Belanja

Berikut dapat dilihat pengujian dari menu keranjang belanja pada tabel 4.12

Tabel 4. 12 Pengujian Menu Keranjang Belanja

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama produk : sepatu Berat : 0 kg Ukuran : S Warna : hitam Diskon : 10% Stok : 18	Menampilkan *form* keranjang belanja	menampilkan *form* keranjang belanja	[√] Diterima [ ] Ditolak
Data masukkan, Qty : 1	Setelah menekan selesai maka akan menampilkan form data tujuan pengiriman	Setelah menekan selesai maka akan menampilkan form data tujuan pengiriman	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data masukkan, Qty : 19	Setelah menekan update maka akan menampilkan pesan "Jumlah yang dibeli melebihi stok yang ada"	Setelah menekan update maka akan menampilkan pesan "Jumlah yang dibeli melebihi stok yang ada"	[√] Diterima [ ] Ditolak
Data masukkan, Qty : -3	Setelah menekan update maka akan menampilkan pesan "Jumlah yang dibeli harus lebih dari 0"	Setelah menekan update maka akan menampilkan pesan "Jumlah yang dibeli harus lebih dari 0"	[√] Diterima [ ] Ditolak

4. Contoh Uji Form Pengiriman

Berikut dapat dilihat pengujian dari pengiriman pada tabel 4.13.

Tabel 4. 13 Pengujian form Pengiriman

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : sheila Alamat lengkap: jl.sedayu 152 Telpon/Hp: 089920219 Email : felu.chey@gmail.com Kodepos : 43892 Provinsi: jawa barat Kota : Kota Bandug	Data tersimpan dan menampilan detail transaksi pemesanan	Data berhasil disimpan dan menampilkan detail transaksi pemesanan	[√] Diterima [ ] Ditolak

Paket pengiriman : JNE REG Biaya : 4000 Lama pengiriman : 3			
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : Alamat lengkap: jl.sedayu 152 Telpon/Hp: 089920219 Email : felu.chey@gmail.com Kodepos : 43892 Provinsi: jawa barat Kota : Kota Bandug Paket pengiriman : JNE REG Biaya : 4000 Lama pengiriman : 3	Jika data nama belum diisi maka menampilkan pesan "Nama tidak boleh kosong"	Jika data nama belum diisi maka menampilkan pesan "Nama tidak boleh kosong"	[√] Diterima [ ] Ditolak
Nama : sheila Alamat lengkap: jl.sedayu 152 Telpon/Hp: 089920219 Email : felu.chey@gmail.com Kodepos : 43892 Provinsi: Kota : Paket pengiriman : Biaya : Lama pengiriman :	Jika data provinsi belum diisi maka muncul pesan "data provinsi tidak boleh kosong"	Jika data provinsi belum diisi maka muncul pesan "data provinsi tidak boleh kosong"	[√] Diterima [ ] Ditolak

5. Pengujian Form Konfirmasi Pembayaran

Berikut dapat dilihat pengujian dari *Form* konfirmasi pembayaran pada tabel 4.14.

Tabel 4. 14 Pengujian *Form* konfirmasi pembayaran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
No.transaksi : 97 Tanggal bayar : 14-01-2012 01:31:31 Bank asal : BCA Jumlah yg dibayar : 88000	Menampilkan pesan "konfirmasi pembayaran telah berhasil dikirim, tunggu konfirmasi"	menampilkan pesan "konfirmasi pembayaran telah berhasil dikirim, tunggu konfirmasi"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
No.transaksi : 97 Tgl bayar : 14-01-2012 Atas Nama : Jumlah Transfer : 88000	Jika atas nama tidak diisi,maka tampilkan pesan kesalahan ”nama harus diisi”	Jika atas nama tidak diisi,maka tampilkan pesan kesalahan ”nama harus diisi”	[√] Diterima [ ] Ditolak
No.transaksi : 97 Tgl bayar : 14-01-2012 Atas Nama : suci Jumlah Transfer : 790	Jika jumlah transfer kurang dari jumlah yang dibayar, maka tampilan pesan “jumlah kurang dari jumlah yang harus dibayar”	Jika jumlah transfer kurang dari jumlah yang dibayar, maka tampilan pesan “jumlah kurang dari jumlah yang harus dibayar”	[√] Diterima [ ] Ditolak

6. Contoh Uji Login manajer dan karyawan bagian kasir

Berikut dapat dilihat pengujian dari login manajer dan karyawan bagian kasir pada tabel 4.15

Tabel 4. 15 Pengujian login manajer dan karyawan bagian kasir

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Username : karyawan *Password* : karyawan	Data login valid maka akan masuk ke menu utama administrator	Data login valid maka akan masuk ke menu utama administrator	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Username : *Password* :	Menampilkan pesan “gagal login. Username atau password yang anda masukan	Login gagal dan menampilkan pesan “gagal login. Username atau password yang anda	[√] Diterima [ ] Ditolak

	salah,silahkan login kembali"	masukan salah,silahkan login kembali"	
Username : karyawan *Password* :	Jika Username dan password kosong, maka menampilkan pesan "gagal login. Username atau password yang anda masukan salah,silahkan login kembali"	Jika Username dan password kosong, maka menampilkan pesan "gagal login. Username atau password yang anda masukan salah,silahkan login kembali"	[√] Diterima [ ] Ditolak

7. Pengujian Tambah Kategori

Berikut dapat dilihat pengujian dari tambah kategori pada tabel 4.16

Tabel 4. 16 Pengujian Tambah kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama kategori: Baju	Data kategori akan tersimpan didalam *database* dan menampilkan pesan "data berhasil disimpan"	Data kategori akan tersimpan didalam *database* dan menampilkan pesan "data berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Nama kategori:	Jika nama Kategori kosong, maka akan menampilkan pesan"nama kategori tidak boleh kosong"	Jika nama Kategori kosong, maka akan menampilkan pesan " nama kategori tidak boleh kosong"	[√] Diterima [ ] Ditolak

8. Pengujian Tambah Data produk

Berikut adalah pengujian dari pengolahan data produk pada tabel 4.17

Tabel 4. 17 Pengujian Tambah Produk

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data barang Contoh: kategori: Baju Nama : blouse Harga : 25000 Berat : 0.2 Deskripsi : Bagus loh	Jika semua field terisi semua maka akan menampilkan pesan " data berhasil disimpan"	Jika semua field terisi semua maka akan menampilkan pesan " data berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data barang Contoh: kategori: Baju Nama : blouse Harga : 25000 Berat : 0.2 Deskripsi : Bagus loh	Jika Nama Kategori kosong, maka akan menampilkan pesan"kategori tidak boleh kosong"	Jika Nama Kategori kosong, maka akan menampilkan pesan " kategori tidak boleh kosong"	[√] Diterima [ ] Ditolak
Mengisi data barang Contoh: kategori: Baju Nama : blouse Harga : aa Berat : 0.2 Deskripsi : Bagus loh	Jika harga berisi bukan angka, maka akan menampilkan pesan"harga harus diisi oleh angka"	Jika harga berisi bukan angka, maka akan menampilkan pesan "harga harus diisi oleh angka"	[√] Diterima [ ] Ditolak
Mengisi data barang Contoh: kategori: Baju Nama : blouse Harga : Berat : 0.2 Deskripsi : Bagus loh	Jika data harga tidak diisi, maka akan menampilkan pesan"harga tidak boleh kosong"	Jika data harga tidak diisi, maka akan menampilkan pesan "harga tidak boleh kosong"	[√] Diterima [ ] Ditolak

9. Pengujian Tambah Provinsi

Berikut dapat dilihat pengujian dari tambah provinsi pada tabel 4.18

Tabel 4. 18 Pengujian Tambah Provinsi

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

Mengisi data provinsi Contoh: Nama Provinsi :	Jika nama provinsi tidak diisi maka akan menampilkan pesan "nama provinsi tidak boleh kosong"	Jika nama provinsi tidak diisi maka akan menampilkan pesan "nama provinsi tidak boleh kosong"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data provinsi Contoh: Nama Provinsi : Jawa Barat	Jika semua field terisi maka akan menampilkan pesan "data berhasil disimpan"	Jika semua field terisi maka akan menampilkan pesan "data berhasil disimpan".	[√] Diterima [ ] Ditolak

10. Pengujian Tambah Kota

Berikut dapat dilihat pengujian tambah kota pada tabel 4.19

Tabel 4. 19 Pengujian Tambah Kota

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data kota Contoh: Provinsi : Jawa Barat Nama Kota : Bandung	Data kota telah tersimpan didalam *database*	Data kota telah tersimpan didalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data kota Contoh: Provinsi : Nama Kota :	Jika nama provinsi tidak terisi maka akan menampilkan pesan "field tidak boleh kosong."	Jika nama provinsi tidak terisi maka akan menampilkan pesan "field tidak boleh kosong."	[√] Diterima [ ] Ditolak
Mengisi data kota Contoh: Provinsi : Jawa Barat Nama Kota :	Jika nama kota tidak terisi maka akan menampilkan pesan "field tidak boleh kosong."	Jika nama kota tidak terisi maka akan menampilkan pesan "field tidak boleh kosong."	[√] Diterima [ ] Ditolak

11. Pengujian Tambah Ukuran

Berikut dapat dilihat pengujian dari tambah ukuran pada tabel 4.20

Tabel 4. 20 Pengujian Tambah Ukuran

Kasus dan Hasil Uji (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data ukuran Contoh: Nama ukuran : S / Small	Data ukuran akan tersimpan didalam *database*	Data ukuran akan tersimpan didalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data ukuran Contoh: Nama ukuran :	Jika nama ukuran field tidak terisi, maka akan menampilkan pesan "field tidak boleh kosong"	Jika salah satu field tidak terisi, maka akan menampilkan pesan "field tidak boleh kosong"	[√] Diterima [ ] Ditolak

12. Pengujian Tambah Tarif Kirim

Berikut dapat dilihat pengujian dari lihat transaksi pada tabel 4.21.

Tabel 4. 21 Pengujian Tambah Ongkos Kirim

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data ongkos kirim Contoh: Provinsi : Jawa Barat Kota : Bandung Jenis pengiriman : JNE harga: Rp.2500	Data ongkos kirim akan tersimpan didalam *database* dan menampilkan pesan "data berhasil di tambah"	Data ongkos kirim akan tersimpan didalam *database* dan menampilkan pesan "data berhasil di tambah"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**

Mengisi data ongkos kirim Contoh: Provinsi : Kota : Jenis pengiriman : harga:	Jika nama provinsi tidak terisi, maka akan menampilkan pesan "field tidak boleh kosong"	Jika nama provinsi tidak terisi, maka akan menampilkan pesan "field tidak boleh kosong"	[√] Diterima [ ] Ditolak
Mengisi data ongkos kirim Contoh: Provinsi : Jawa Barat Kota : Jenis pengiriman : harga:	Jika nama kota tidak terisi, maka akan menampilkan pesan "field tidak boleh kosong"	Jika nama kota tidak terisi, maka akan menampilkan pesan "field tidak boleh kosong"	[√] Diterima [ ] Ditolak

4.4.3 Kesimpulan Hasil Pengujian Alpha

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

4.5 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan membuat kuesioner mengenai kepuasan user dengan kandungan poin syarat *user friendly* untuk selanjutnya dibagikan kepada sebagian *user* dengan mengambil *sample* sebanyak 10 orang dan 1 orang dari pihak Toko Genick.

4.5.1 Skenario Pengujian Beta

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

1. Lingkungan Toko Genick

   Penelitian di lingkungan pegawai Toko Genick dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan di awal.

2. Lingkungan Masyarakat Umum

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

4.5.1.1 Kuisioner Pengujian Beta Untuk Pengguna

1. Apakah anda setuju bahwa informasi produk yang diberikan atau ditawarkan oleh website ini sesuai dengan informasi yang anda butuhkan?

   A. Sangat Setuju
   B. Setuju
   C. Cukup setuju
   D. Biasa saja
   E. Tidak setuju

2. Apakah anda setuju bahwa tampilan website cukup menarik dan mudah dipahami?

   A. Sangat Setuju
   B. Setuju
   C. Cukup setuju
   D. Biasa saja
   E. Tidak setuju

3. Apakah anda setuju bahwa website ini akan lebih memudahkan pelanggan dalam melakukan transaksi pembelian barang?

   A. Sangat Setuju
   B. Setuju
   D. Biasa saja
   E. Tidak setuju

C. Cukup setuju

4. Apakah anda setuju bahwa sistem pembayaran yang digunakan akan lebih memudahkan anda dan terjamin?

A. Sangat Setuju
B. Setuju
C. Cukup setuju
D. Biasa saja
E. Tidak setuju

5. Apakah anda setuju bahwa website ini telah layak untuk dionlinekan?

A. Sangat Setuju
B. Setuju
C. Cukup setuju
D. Biasa saja
E. Tidak setuju

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

Keterangan :

P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai persentase

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus . Kuisioner ini diujikan kepada 10 orang.

1. Apakah anda setuju bahwa informasi produk yang diberikan atau ditawarkan oleh *website* ini sesuai dengan informasi yang anda butuhkan?

Tabel 4. 22 Hasil pengujian kuesioner soal nomor 1

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	0	0
	2	Setuju	8	80
	3	Cukup Setuju	2	20

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

Berdasarkan hasil prosentase maka dapat disimpulkan sebanyak 8 atau 80% menyatakan setuju, dan 2 atau 20% menyatakan setuju bahwa informasi yang diberikan sesuai dengan informasi yang dibutuhkan *costumer*.

2. Apakah anda setuju bahwa tampilan *website* cukup menarik dan mudah dipahami?

Tabel 4. 23 Hasil pengujian kuesioner soal nomor 2

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	4	40
	2	Setuju	6	60
	3	Cukup Setuju	0	0
	4	Biasa Saja	0	0
	5	Tidak Setuju	0	0
jumlah			**10**	**100**

Berdasarkan hasil prosentase maka dapat disimpulkan sebanyak 4 orang atau 40 % menyatakan sangat setuju, 6 orang atau 60% menyatakan cukup setuju bahwa tampilan *website* cukup menarik dan mudah untuk dipahami.

3. Apakah anda setuju bahwa *website* ini akan lebih memudahkan pelanggan dalam melakukan transaksi pembelian barang?

Tabel 4. 24 Hasil pengujian kuesioner soal nomor 3

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	0	0
	2	Setuju	2	20
	3	Cukup Setuju	7	70
	4	Biasa Saja	1	10
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase maka dapat disimpulkan sebanyak 2 orang atau 20 % menyatakan setuju dan, 7 atau 70 % menyatakan cukup setuju dan 1 atau 10% menyatakan biasa saja, bahwa *website* ini akan lebih memudahkan pelanggan dalam melakukan transaksi pembelian barang.

4. Apakah anda setuju bahwa sistem pembayaran yang digunakan akan lebih memudahkan anda dan terjamin?

Tabel 4. 25 Hasil pengujian kuesioner soal nomor 4

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
4	1	Sangat Setuju	0	0
	2	Setuju	6	60
	3	Cukup Setuju	2	20
	4	Biasa Saja	2	20
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase maka dapat disimpulkan sebanyak 6 orang atau 60 % menyatakan setuju, 2 orang atau 20% menyatakan cukup setuju bahwa sistem pembayaran yang digunakan akan lebih memudahkan, dan 2 orang atau 20% menyatakan biasa saja bahwa sistem pembayaran yang digunakan lebih mudah.

5. Apakah anda setuju dengan adanya website ini Toko Genick telah layak untuk di-*online*-kan?

Tabel 4. 26 Hasil pengujian kuesioner soal nomor 5

Pertanyaan	No.	Keterangan	Responden	Prosentase(%)
5	1	Sangat Setuju	0	0
	2	Setuju	8	80
	3	Cukup Setuju	2	20
	4	Biasa Saja	0	0
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase maka dapat disimpulkan sebanyak 8 orang atau 80% menyatakan setuju dan 2 orang atau 20% menyatakan setuju bahwa *website* ini akan memperluas jangkauan konsumen Toko Genick.

4.5.1.2 Kuisioner Pengujian Beta untuk karyawan

1. Menurut anda, aplikasi *E-commerce* ini cukup membantu dalam penyebaran informasi produk yang dijual?

   Menurut saya, cukup membantu karena informasi akan cepat diterima oleh konsumen dimana saja dan kapan saja.

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

   Instruksi-instruksi yang terdapat di dalam website ini mudah dimengerti, mungkin untuk kedepannya dapat ditambahkan lagi instruksi pada bagian karyawan.

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

   Untuk saat ini cukup membantu, karena selama ini pengolahan data barangnya masih menggunakan cara yang manual.

4. Apakah anda setuju bahwa dengan adanya *website* ini akan lebih mudah mengelola data transaksi penjualan?

   Iya, karena sebelumnya transaksi penjualan belum secara *online*, maka dengan adanya aplikasi ini sangat membatu.

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

   Ya, sangat membantu karena selama ini laporan dikerjakan secara manual sehingga membutuhkan waktu lebih lama.

4.5.2 Kesimpulan Pengujian Beta

Berdasarkan pengujian beta, bahwa dapat diambil kesimpulan bahwa aplikasi ini memudahkan pelanggan dalam mendapatkan informasi barang yang tersedia pada Toko Genick selain itu pelanggan juga dapat melakukan transaksi pembelian barang dengan lebih mudah.

Dalam Proses pembayaran yang digunakan cukup mudah digunakan bagi pelanggan dan dapat dipercaya. Aplikasi ini juga memudahkan pihak toko dalam mengolah data master, data barang dan data transaksi dari keluar masuknya barang. Aplikasi ini juga dapat memperluas usaha pada Toko Genick tidak terbatas pemasarannya secara geografis karena dapat digunakan dimanapun.