BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini merupakan tahap penerjemahan kebutuhan pembuatan aplikasi ke dalam representasi perangkat lunak sebelum penulisan kode program di mulai sesuai dengan hasil analisis yang telah dilakukan atau implementasi. Implementasi yang dilakukan mencakup data-data yang digambarkan dengan tampilan, sedangkan untuk pengujian meliputi pengujian terhadap perangkat lunak hasil implementasi.

4.1 Implementasi

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

4.1.1 Implementasi Perangkat Keras (*Hardware*)

Analisis perangkat keras (*hardware*) merupakan proses analisis yang lebih menekankan kepada aspek pemanfaatan perangkat keras yang selama ini telah dimiliki toko Badak Jaya.

Adapun spesifikasi perangkat keras komputer yang ada di toko Badak Jaya adalah sebagai berikut :

1. *Processor* Pentium 4 2.8 GHz
2. Kapasitas RAM 512 Mb
3. VGA *card* 256 MB
4. Kapasitas *Hardisk* 80 Gb
5. *Monitor*
6. *Keyboard*
7. *Mouse.*

Kebutuhan minimal perangkat keras yang akan dibangun memiliki spesifikasi sebagai berikut :

1. *Processor* dengan kecepatan 1.8 GHz
2. RAM sebesar 256 MB
3. VGA *card* 256 MB
4. Kapasitas *free Hardisk* 80 GB
5. *Monitor* dengan resolusi 1024 x 768
6. *Keyboard*
7. *Mouse.*

Setelah dilakukan analisis terhadap perangkat keras yang dimiliki oleh toko Badak Jaya, maka dapat diambil kesimpulan bahwa perangkat keras yang ada sudah layak dan dapat mendukung sistem *e-commerce* yang akan dibangun, namun demikian perlu akses *internet* yang memadai agar transaksi dapat berjalan maksimal karena dilakukan secara *online* serta memudahkan dalam pengolahan *database* yang disimpan di *server* dan *printer* standar untuk mencetak laporan barang dan laporan penjualan.

4.1.2 Implementasi Perangkat Lunak (*Software)*

Adapun perangkat lunak yang digunakan untuk mengimplementasikan perangkat lunak ini adalah sebagai berikut:

a. *Microsoft Windows XP Profesional Service Pack 2* sebagai sistem operasi

b. Macromedia Dreamweaver sebagai media *coder*

c. Bahasa pemrograman yang digunakan yaitu *PHP*

d. Software aplikasi *Web Server Xampp*

e. Software aplikasi *database MySQL*

f. Internet Explorer dan *Mozilla Firefox* sebagai media *browser*.

4.1.3 Implementasi Database

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

a. Tabel admin

Tabel ini berfungsi sebagai tempat untuk menyimpan username dan passworsd admin. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `admin` (
  `username` varchar(15) NOT NULL,
  `password` varchar(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `verification` varbinary(40) NOT NULL,
```

```
  PRIMARY KEY (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

b. Tabel detail_pembelian

Tabel ini berfungsi sebagai tempat untuk menyimpan data detail pembelian yang dilakukan oleh member. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `detail_pembelian` (
  `id_detailbeli` int(11) NOT NULL AUTO_INCREMENT,
  `idpembelian` int(11) NOT NULL,
  `hargabeli` int(20) NOT NULL,
  `idproduk` int(8) NOT NULL,
  `jenis` varchar(20) NOT NULL,
  `qty` int(11) NOT NULL,
  `diskon` int(11) NOT NULL,
  PRIMARY KEY (`id_detailbeli`),
  UNIQUE KEY `id_detailbeli` (`id_detailbeli`),
  KEY `idproduk` (`idproduk`),
  KEY `idpembelian` (`idpembelian`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=131 ;
```

```
ALTER TABLE `detail_pembelian`
  ADD CONSTRAINT `detail_pembelian_ibfk_1` FOREIGN KEY
(`idpembelian`) REFERENCES `pembelian` (`id_pembelian`),
  ADD CONSTRAINT `detail_pembelian_ibfk_2` FOREIGN KEY (`idproduk`)
REFERENCES `temp_pemesanan` (`idproduk`);
```

c. Tabel detail_produk

Tabel ini berfungsi sebagai tempat untuk menyimpan data detail produk. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `detail_produk` (
  `id_detailp` int(11) NOT NULL AUTO_INCREMENT,
  `id_produk` int(8) NOT NULL,
  `jenis` varchar(12) NOT NULL,
  `stok` int(11) NOT NULL,
  PRIMARY KEY (`id_detailp`),
  KEY `id_produk` (`id_produk`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=57 ;
ALTER TABLE `detail_produk`
ADD CONSTRAINT `detail_produk_ibfk_1` FOREIGN KEY (`id_produk`)
REFERENCES `produk` (`id_produk`);
```

d. Tabel detail_retur

Tabel ini berfungsi untuk menyimpan detail dari retur. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `detail_retur` (
  `id_retur` int(11) NOT NULL,
  `id_pembelian` int(11) NOT NULL,
  `id_produk` int(11) NOT NULL,
  `jenis` varchar(11) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `alasan` text NOT NULL,
  `status` varchar(100) NOT NULL,
  `status_kadaluarsa` varchar(20) DEFAULT NULL,
  `status_kemasan` varchar(20) DEFAULT NULL,
  UNIQUE KEY `id_retur` (`id_retur`),
  KEY `id_retur` (`id_retur`),
  KEY `id_pembelian` (`id_pembelian`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `detail_retur`
```

```
 ADD CONSTRAINT `detail_retur_ibfk_1` FOREIGN KEY (`id_retur`)
REFERENCES `retur` (`id_retur`),

 ADD CONSTRAINT `detail_retur_ibfk_2` FOREIGN KEY (`id_pembelian`)
REFERENCES `detail_pembelian` (`idpembelian`);
```

e. Tabel jasa_pengiriman

Tabel ini berfungsi sebagai tempat menyimpan data-data jenis pengiriman. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `jasa_pengiriman` (

  `id_jasakirim` int(11) NOT NULL AUTO_INCREMENT,

  `nama_jasakirim` varchar(100) NOT NULL,

  `jenis_pengiriman` varchar(20) NOT NULL,

  PRIMARY KEY (`id_jasakirim`)

) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

f. Tabel table kabkota

Tabel ini berfungsi sebagai tempat untuk menyimpan data kabupaten dan kota. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `kabkota` (

  `id_kabkota` int(10) NOT NULL AUTO_INCREMENT,

  `id_prov` int(10) DEFAULT NULL,
```

```
  `nama` varchar(50) DEFAULT NULL,
  `kabkota` varchar(20) DEFAULT NULL,
  PRIMARY KEY (`id_kabkota`),
  UNIQUE KEY `nama` (`nama`),
  UNIQUE KEY `kabkota` (`kabkota`),
  KEY `id_prov` (`id_prov`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=425 ;
ALTER TABLE `kabkota`
 ADD CONSTRAINT `kabkota_ibfk_1` FOREIGN KEY (`id_prov`)
REFERENCES `provinsi` (`id_prov`);
```

g. Tabel kategori

Tabel ini berfungsi sebagai tempat untuk menyimpan data kategori dari barang. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` int(11) NOT NULL AUTO_INCREMENT,
  `kategori` varchar(40) NOT NULL,
  PRIMARY KEY (`id_kategori`),
  UNIQUE KEY `kategori` (`kategori`)
```

```
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;
```

h. Tabel member

Tabel ini berfungsi sebagai tempat untuk menyimpan data-data dari member. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `member` (
  `id_member` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(100) NOT NULL,
  `alamat` text NOT NULL,
  `id_kabkota` int(10) NOT NULL,
  `kodepos` varchar(10) NOT NULL,
  `telepon` varchar(20) NOT NULL,
  `email` varchar(100) NOT NULL,
  `password` varchar(50) NOT NULL,
  `tgl_reg` datetime NOT NULL,
  `status` int(11) NOT NULL,
  PRIMARY KEY (`id_member`),
  UNIQUE KEY `email` (`email`),
  KEY `id_kabkota` (`id_kabkota`)
```

```
) ENGINE=InnoDB  DEFAULT CHARSET=latin1
ROW_FORMAT=COMPACT AUTO_INCREMENT=51 ;

ALTER TABLE `member`
 ADD CONSTRAINT `member_ibfk_1` FOREIGN KEY (`id_kabkota`)
REFERENCES `kabkota` (`id_kabkota`);
```

i. Tabel ongkos_kirim

Tabel ini berfungsi sebagai tempat untuk menyimpan data ongkos kirim. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `ongkos_kirim` (
  `id_ongkir` int(11) NOT NULL AUTO_INCREMENT,
  `harga_ongkir` int(11) NOT NULL,
  `id_kabkota` int(11) NOT NULL,
  `id_jasakirim` int(11) NOT NULL,
  PRIMARY KEY (`id_ongkir`),
  KEY `id_jasakirim` (`id_jasakirim`),
  KEY `id_kabkota` (`id_kabkota`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=52 ;

ALTER TABLE `ongkos_kirim`
```

```
 ADD CONSTRAINT `ongkos_kirim_ibfk_1` FOREIGN KEY (`id_kabkota`)
REFERENCES `kabkota` (`id_kabkota`),
ADD CONSTRAINT `ongkos_kirim_ibfk_2` FOREIGN KEY (`id_jasakirim`)
REFERENCES `jasa_pengiriman` (`id_jasakirim`);
```

j. Tabel pembelian

Tabel ini berfungsi sebagai tempat untuk menyimpan data-data pembelian yang dilakukan oleh member. Adapun query yang digunakan

```
CREATE TABLE IF NOT EXISTS `pembelian` (
  `id_pembelian` int(11) NOT NULL AUTO_INCREMENT,
  `id_produk` int(11) NOT NULL,
  `session_id` varchar(32) NOT NULL,
  `tgl_beli` datetime NOT NULL,
  `tgl_bayar` datetime NOT NULL,
  `status` enum('pesan','bayar','konfirmasi','kirim','selesai','retur','batal') NOT
NULL,
  `id_member` int(11) NOT NULL,
  `pembayaran` enum('rekening','paypal','cod') NOT NULL,
  `kirim_nama` varchar(50) NOT NULL,
  `kirim_alamat` text NOT NULL,
```

```
`kirim_telp` varchar(12) NOT NULL,
`kirim_kota` int(11) NOT NULL,
`kirim_kdpos` varchar(6) NOT NULL,
`kirim_ongkos` int(32) NOT NULL,
`id_ongkir` int(11) NOT NULL,
`kirim_resi` varchar(15) NOT NULL,
`transfer_bank` varchar(20) NOT NULL,
`transfer_rekening` varchar(30) NOT NULL,
`no_transaksi` int(11) NOT NULL,
`transfer_jumlah` int(32) NOT NULL,
`transfer_tgl` varchar(30) NOT NULL,
`penerima_bank` varchar(15) NOT NULL,
`penerima_rekening` varchar(30) NOT NULL,
`totalbayar` decimal(32,2) NOT NULL,
`tgl_kirim` datetime NOT NULL,
`tgl_terima` datetime NOT NULL,
`tgl_terima_barang` varchar(20) DEFAULT NULL,
```

```
  `jam_terima_barang` varchar(10) DEFAULT NULL,
  `penerima_barang` varchar(30) DEFAULT NULL,
  `status_kiriman` varchar(20) DEFAULT NULL,
  PRIMARY KEY (`id_pembelian`),
  KEY `id_member` (`id_member`),
  KEY `id_produk` (`id_produk`),
  KEY `id_ongkir` (`id_ongkir`),
  KEY `id_pembelian` (`id_pembelian`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=156 ;
ALTER TABLE `pembelian`
ADD CONSTRAINT `pembelian_ibfk_1` FOREIGN KEY (`id_member`)
REFERENCES `member` (`id_member`),
 ADD CONSTRAINT `pembelian_ibfk_2` FOREIGN KEY (`id_produk`)
REFERENCES `detail_produk` (`id_produk`),
ADD CONSTRAINT `pembelian_ibfk_3` FOREIGN KEY (`id_ongkir`)
REFERENCES `ongkos_kirim` (`id_ongkir`);
```

k. Table produk

Tabel ini berfungsi sebagai tempat untuk menyimpan data-data produk. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `produk` (
  `id_produk` int(8) NOT NULL,
  `id_kategori` int(11) NOT NULL,
  `produk` varchar(30) NOT NULL,
  `detail` text NOT NULL,
  `diskon` int(3) DEFAULT NULL,
  `harga` int(20) NOT NULL,
  `berat` int(11) NOT NULL,
  `tgl_input` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_produk`),
  UNIQUE KEY `produk` (`produk`),
  KEY `id_kategori` (`id_kategori`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `produk`
```

```
ADD CONSTRAINT `produk_ibfk_1` FOREIGN KEY (`id_produk`)
REFERENCES `detail_produk` (`id_produk`),
 ADD CONSTRAINT `produk_ibfk_2` FOREIGN KEY (`id_kategori`)
REFERENCES `kategori` (`id_kategori`);
```

l. Table provinsi

Tabel ini berfungsi sebagai tempat untuk menyimpan data provinsi. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `provinsi` (
  `id_prov` int(10) NOT NULL AUTO_INCREMENT,
  `nama` varchar(100) DEFAULT NULL,
  PRIMARY KEY (`id_prov`),
  UNIQUE KEY `nama` (`nama`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=34 ;
```

m. Table retur

Tabel ini berfungsi sebagai tempat untuk menyimpan data-data retur barang. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `retur` (
  `id_retur` int(11) NOT NULL AUTO_INCREMENT,
  `id_member` int(11) NOT NULL,
```

```
  `id_pembelian` int(11) NOT NULL,
  `tgl_retur` datetime DEFAULT NULL,
  `keterangan` text,
  `resi_retur` varchar(50) NOT NULL,
  `id_ongkir` int(11) DEFAULT NULL,
  `status` varchar(20) NOT NULL,
  PRIMARY KEY (`id_retur`),
  KEY `id_pembelian` (`id_pembelian`),
  KEY `id_member` (`id_member`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=54 ;

ALTER TABLE `retur`
 ADD CONSTRAINT `retur_ibfk_2` FOREIGN KEY (`id_member`)
REFERENCES `pembelian` (`id_member`),
 ADD CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`id_pembelian`)
REFERENCES `pembelian` (`id_pembelian`);
```

n. Table temp_pemesanan

Tabel ini berfungsi sebagai tempat sementara penyimpanan pesanan. Adapun query yang digunakan adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `temp_pemesanan` (
  `id_temp` int(11) NOT NULL AUTO_INCREMENT,
  `idproduk` int(8) NOT NULL,
  `session_id` varchar(32) NOT NULL,
  `qty` int(11) NOT NULL,
  `temp_jenis` varchar(9) NOT NULL,
  `temp_hargadiskon` int(20) NOT NULL,
  `tanggal_pesan` datetime NOT NULL,
  `id_pembelian` int(11) NOT NULL,
  PRIMARY KEY (`id_temp`),
  KEY `id_pembelian` (`id_pembelian`),
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=140 ;
ALTER TABLE `temp_pemesanan`
  ADD CONSTRAINT `temp_pemesanan_ibfk_3` FOREIGN KEY
(`id_pembelian`) REFERENCES `pembelian` (`id_pembelian`),
  ADD CONSTRAINT `temp_pemesanan_ibfk_2` FOREIGN KEY (`idproduk`)
REFERENCES `produk` (`id_produk`);
```

4.1.4 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap halaman web yang dibuat dan pengkodeannya dalam bentuk file program berekstensi .php dan .html. Berikut ini adalah implementasi antarmuka yang dibuat dan dibedakan antara antarmuka untuk user pengunjung, member dan admin.

Tabel 4. 1 Implementasi Antarmuka Pengunjung

Menu	Deskripsi	Nama File
Halaman Utama	Menampilkan produk terbaru, barang diskon.	home.php
Registrasi member	Halaman registrasi digunakan untuk mendaftar sebagai member baru	register.php
Produk	Halaman produk berisi tentang semua produk yang di jual	produk.php
Kontak	Halaman kontak berisi tentang komentar yang ingin disampaikan	kontak.php
Detail Produk	Halaman detail produk digunakan untuk melihat detail produk dari produk yang di pilih	Detail_produk.php
Keranjang Belanja / pemesanan	Halaman keranjang belanja digunakan untuk melihat, merubah dan membatalkan pemesanan	pemesanan.php
Login Member	Login member digunakan sebagai halaman login member	login.php

Tabel 4. 2 Implementasi Antarmuka Member

Menu	Deskripsi	Name File
Halaman Utama	Menampilkan produk terbaru, barang diskon.	home.php
Produk	Halaman produk berisi tentang semua produk yang di jual	produk.php
Retur Produk	Halaman retur produk digunakan untuk melakukan kegiatan untuk retur produk yang telah di beli	retur.php
Profil Member	Halaman profil member digunakan untuk melihat data member dan edit member	member.php

Kontak	Halaman kontak berisi tentang komentar yang ingin disampaikan	kontak.php
Detail Produk	Halaman detail produk digunakan untuk melihat detail produk dari produk yang di pilih	Detail_produk.php
Keranjang Belanja / pemesanan	Halaman keranjang belanja digunakan untuk melihat, merubah dan membatalkan pemesanan	pemesanan.php
Form pengiriman	Digunakan untuk mengisi alamat data penerima pengiriman produk yang dibeli	pengiriman.php
Form Jasa Pengiriman	Digunakan untuk memilih jasa pengiriman yang akan digunakan	Jasa kirim.php
History transaksi	Digunakan untuk melihat history pembelian, memilih metode pembayaran dan melihat status transaksi	History.php
History retur	Digunakan untuk melihat history retur	Retur_history.php
Logout	Digunakan member untuk keluar dari aplikasi	logout.php

Tabel 4. 3 Implementasi Antarmuka Admin

Menu	Deskripsi	Name File
Login admin	Digunakan sebagai halaman login admin	login.php
Halaman Utama	Menampilkan informasi tentang halaman administrator	index.php
Data admin	Digunakan jika admin ingin mengubah datanya	admin.php
Data kategori	Digunakan untuk melihat, menambah dan merubah data kategori	kategori.php
Data produk	Digunakan untuk melihat, menambah dan merubah data produk	produk.php
Detail Produk	Halaman detail produk digunakan untuk melihat detail produk dari produk yang di pilih	Detail_produk.php
Data member	Digunakan untuk melihat data member	member.php

Data provinsi	Digunakan untuk melihat, menambah dan merubah data provinsi	provinsi.php
Data kota	Digunakan untuk melihat, menambah dan merubah data kota	kota.php
Jasa Pengiriman	Digunakan untuk melihat, menambah dan merubah jasa pengiriman	jasakirim.php
Ongkos kirim	Digunakan untuk melihat, menambah dan merubah ongkos kirim	ongkir.php
Transaksi	Digunakan untuk memproses transaksi dari mulai konfirmasi pembayaran, pengiriman sampai selesai	transaksi.php
Retur	Digunakan untuk memproses transaksi retur produk	Retur_history.php
Buku Tamu	Digunakan Untuk mengelola komentar saran, dan kritik serta feedback baik dari pengunjung maupun member.	Kontak.php
Laporan	Digunakan untuk melihat data laporan harian,bulanan maupun tahunan setra laporan data produk dalam bentuk PDF.	laporan.php
Logout	Digunakan admin untuk keluar dari aplikasi	logout.php

4.2 Pengujian

Pengujian yang digunakan untuk menguji sistem yang baru adalah metode pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak dan pengujian ini berusaha menemukan kesalahan dalam kategori sebagai berikut :

1. Kesalahan *interfaces*

2. Fungsi-fungsi yang tidak sesuai dengan prosedur

3. Kesalahan dalam struktur data

4. Kesalahan kinerja sistem

4.2.1 Metode Pengujian

Pengujian program ini menggunakan metode *Black box* . Pengujian *Black-Box* merupakan pengujian program berdasarkan fungsi dari program. Tujuan dari metode *Black Box* ini adalah untuk menemukan kesalahan fungsi pada program. Pengujian dengan metode *blackbox* dilakukan dengan cara memberikan sejumlah input pada program aplikasi yang kemudian diproses sesuai dengan kebutuhan fungsionalnya untuk melihat apakah program aplikasi menghasilkan output yang diinginkan dan sesuai dengan fungsi dari program tersebut. Apabila dari input yang diberikan proses menghasilkan output yang sesuai dengan kebutuhan fungsionalnya, maka program aplikasi yang bersangkutan telah benar, tetapi jika output yang dihasilkan tidak sesuai dengan kebutuhan fungsionalnya, maka masih terdapat kesalahan pada program aplikasi.

Pengujian dilakukan dengan mencoba semua kemungkinan yang terjadi dan pengujian dilakukan berulang-ulang. Jika dalam pengujian ditemukan kesalahan, maka akan dilakukan penelusuran dan perbaikan untuk memperbaiki kesalahan yang terjadi. Jika telah selesai melakukan perbaikan, maka akan dilakukan pengujian kembali. Pengujian dan perbaikan dilakukan secara terus menerus hingga diperoleh hasil yang terbaik.

4.2.2 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara *Alpha* dan *Betha* dengan metode pengujian yang sama. Pengujian *aplikasi e-commerce* berikut menggunakan data uji berdasarkan input data yang dimasukkan. Rencana pengujian selengkapnya terlihat pada tabel di bawah ini :

Tabel 4.4 Rencana Pengujian Halaman Admin

Pengujian Halaman Admin		
Item Uji	**Detail Pengujian**	**Jenis Uji**
Login Admin	Isi data login admin	*Black Box*
	Verifikasi data login admin	*Black Box*
Produk	Tampilkan data produk	*Black Box*
	Tambah, Edit data produk	*Black Box*
Detail produk	Tampilkan detail produk	*Black Box*
	Tambah, Edit dan hapus detail produk	*Black Box*
Kategori	Tampilkan data kategori	*Black Box*
	Tambah, edit data kategori	*Black Box*
Member	Tampilkan data member	*Black Box*
Pengolahan data transaksi	Tampilkan data transaksi	*Black Box*
	Konfirmasi pembayaran	*Black Box*
	Konfirmasi pengiriman	*Black Box*
Admin	Tampilkan data admin	*Black Box*
	Edit data admin	*Black Box*
	Tambah data admin	*Black Box*
	Hapus data admin	*Black Box*
Provinsi	Tampilkan data Provinsi	*Black Box*
	Tambah data Provinsi	*Black Box*
	Ubah data Provinsi	*Black Box*
	Hapus data Provinsi	*Black Box*
Kota	Tampilkan data Kota	*Black Box*
	Tambah data Kota	*Black Box*
	Ubah data Kota	*Black Box*
	Hapus data Kota	*Black Box*
Laporan	Tampilan laporan data produk Harian, Bulanan dan tahunan	*Black Box*
	Tampilkan laporan data produk	*Black Box*
	Tampilkan laporan data retur	*Black Box*

Tabel 4.5 Rencana Pengujian Halaman Member

Pengujian Halaman User		
Item Uji	**Detail Pengujian**	**Jenis Uji**
Registrasi *member*	Menampilkan *form* registrasi member	*Black Box*
	Isi *form* registrasi	*Black Box*
	Simpan data member baru	*Black Box*
Login	Menampilkan halaman login user	*Black Box*
Data Profil Member	Menampilkan data member	*Black Box*
	Menampilkan *form* ubah akun member	*Black Box*
	Simpan perubahan data member	*Black Box*
Menu History Transaksi	Menampilkan Daftar history	*Black Box*
	Menampilkan detail Pembelian	*Black Box*
	Menampilkan Status transaksi dan informasinya	*Black Box*
	Pilih metode pembayaran	*Black Box*
Menu Produk	Menampikan data Produk	*Black Box*
	Menampilkan produk berdasarkan kategori	*Black Box*
Menu keranjang belanja	Menampilkan, mengupdate dan menghapus data pemesanan	*Black Box*
Form Pengiriman	Menampilkan form pengiriman	*Black Box*
	Menyimpan data pengiriman	*Black Box*
Form Pembayaran	Menampilkan konfirmasi Pemesanan	*Black Box*
Pencarian Produk	Menampilkan Hasil pencarian produk	*Black Box*

4.2.3 Kasus dan Hasil Pengujian

Berikut adalah rencana pengujian yang akan dilakukan :

4.2.3.1 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 dimasukan pada tiap tampilan.

Tabel 4.6 Pengujian Cari Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan kategori dengan benar Kategori : alergi	Menampilkan kategori alergi	Dapat Menampilkan kategori alergi	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data kategori salah atau tidak ada Kategori : bronchitis	Kategori tidak ditemukan dan menampilkan pesan kesalahan “Data tidak ditemukan”	Kategori tidak ditemukan dan menampilkan pesan kesalahan “Data tidak ditemukan	[√] Diterima [ ] Ditolak

Tabel 4.7 Pengujian Tambah Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan kategori yang belum ada Kategori : batuk	Kategori baru dapat disimpan dan menampilkan pesan"kategori berhasil disimpan"	Kategori baru dapat disimpan dan menampilkan pesan"kategori berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kategori yang dimasukkan sudah ada Kategori : alergi	Kategori tidak dapat disimpan dan menampilkan pesan kesalahan " kategori sudah terdaftar di *database*"	Kategori tidak dapat disimpan dan menampilkan pesan kesalahan " kategori sudah terdaftar di *database*"	[√] Diterima [ ] Ditolak

Tabel 4.8 Pengujian Edit Kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Edit dengan kategori baru Kategori Baru asma	Kategori baru dapat diedit	Kategori baru dapat diedit	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Edit dengan kategori yang sudah ada Kategori : alergi	Kategori tidak dapat diedit dan menampilkan pesan"kategori sudah terdaftar di *database*"	Kategori tidak dapat diedit dan menampilkan pesan"kategori sudah terdaftar di *database*"	[√] Diterima [ ] Ditolak

Tabel 4.9 Pengujian Cari Produk

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan produk dengan benar Nama Produk : alupent	Menampilkan data produk alupent	Menampilkan data produk alupent	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukkan data produk yang belum ada Nama Produk : ciprox	Produk tidak ditemukan menampilkan pesan kesalahan "Data tidak ditemukan"	Produk tidak ditemukan menampilkan pesan kesalahan "Data tidak ditemukan"	[√] Diterima [ ] Ditolak

Tabel 4.10 Pengujian Tambah Data Produk

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan data produk dengan benar Kategori : asma Gambar : alupent.jpg Nama Produk : alupent Berat Produk : 0.8 Harga : 3100	Data tersimpan ke *database* Dan menampilkan pesan "data berhasil disimpan"	Data tersimpan ke *database* Dan menampilkan pesan "data berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukan nama produk yang sama dengan kategori yang sama Kategori : asma Nama Poduk : alupent	Data produk tidak tersimpan dan menampilkan pesan kesalahan "Nama Produk sudah terdaftar di database"	Data produk tidak tersimpan dan menampilkan pesan kesalahan "Nama Produk sudah terdaftar di database"	[√] Diterima [ ] Ditolak
Harga produk dan berat produk diisi dengan bukan angka Berat Produk : aa	Data produk tidak tersimpan dan menampilkan pesan kesalahan	Data produk tidak tersimpan dan menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak

Harga Produk : bb	"Berat dan harga produk harus angka"	"Berat dan harga produk harus angka"	

Tabel 4.11 Pengujian Edit Data Produk

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan data edit produk dengan benar Kategori : asma Gambar : alupent.jpg Nama Produk : alupent Berat Produk : 0.8 Harga : 80000	Data baru berhasil disimpan dan menampilkan pesan"data berhasil di *update"*	Data baru berhasil disimpan dan menampilkan pesan"data berhasil di *update"*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukkan data gambar yang *extensi* selain jpeg dan jpg alupent.png	Data produk tidak dapat diedit dan menampilkan pesan kesalahan "Extensi gambar hanya .jpg dan .jpeg"	Data produk tidak dapat diedit dan menampilkan pesan kesalahan "Extensi gambar hanya .jpg dan .jpeg"	[√] Diterima [ ] Ditolak

Tabel 4.12 Pengujian *Update* Stok Produk

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data stok dengan benar Stok : 3	Stok produk dapat diedit dan menampilkan pesan " stok berhasil di *update*""	Stok produk dapat diedit dan menampilkan pesan " stok berhasil di *update*""	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Stok diisi angka kurang dari 0	Stok produk tidak dapat di*update*	Stok produk tidak dapat	[√] Diterima

	dan menampilkan pesan " Stok harus lebih dari 0"	di*update* dan menampilkan pesan " Stok harus lebih dari 0"	[ ] Ditolak
Stok diisi dengan selain angka Stok : c	Stok produk tidak dapat di*update* dan menampilkan pesan "Stok harus di isi dengan angka"	Stok produk tidak dapat di*update* dan menampilkan pesan "Stok harus di isi dengan angka"	[√] Diterima [ ] Ditolak

Tabel 4.13 Pengujian Tambah Harga Kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan harga kirim yang benar Provinsi : jawa barat Kota : bandung Paket JNE : YES Harga : 18000	Harga kirim berhasil tambah Dan menampilkan pesan" data berhasil disimpan"	Harga kirim berhasil tambah Dan menampilkan pesan" data berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Harga kirim diisi selain dengan angka Harga kirim : k	Harga kirim tidak diedit dan menampilkan pesan kesalahan" harga harus angka"	Harga kirim tidak diedit dan menampilkan pesan kesalahan" harga harus angka"	[√] Diterima [ ] Ditolak
Harga kirim diisi kurang dari 0	Harga kirim tidak diedit dan menampilkan pesan kesalahan" harga harus angka"	Harga kirim tidak diedit dan menampilkan pesan kesalahan" harga harus angka"	[√] Diterima [ ] Ditolak

Tabel 4.14 Pengujian Edit Harga Kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan data edit harga kirim yang benar Provinsi : jawa barat Kota : bandung Paket JNE : YES Harga : 18000	Harga kirim berhasil diedit Dan menampilkan pesan"data berhasil di *update*"	Harga kirim berhasil diedit Dan menampilja pesan"data berhasil di *update*"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Harga kirim diisi selain dengan angka Harga : J	Harga kirim tidak diedit dan menampilkan pesan kesalahan" harga harus angka"	Harga kirim tidak diedit dan menampilkan pesan kesalahan" harga harus angka"	[√] Diterima [ ] Ditolak
Harga kirim diisi kurang dari 0	Harga kirim tidak diedit dan menampilkan pesan kesalahan" harga harus angka"	Harga kirim tidak diedit dan menampilkan pesan kesalahan" harga harus angka"	[√] Diterima [ ] Ditolak

Tabel 4.15 Pengujian Laporan Pembelian harian

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih tanggal, bulan dan tahun laporan Tanggal : 25 Bulan : februari Tahun : 2012	Menampilkan data pembelian pada tanggal 25 februari 2012	Menampilkan data pembelian pada tanggal 25 februari 2012	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Tanggal atau Bulan atau tahun tidak diisi	Tidak dapat menampilkan data dan menampilkan pesan	Tidak dapat menampilkan data dan menampilkan pesan	[√] Diterima [ ] Ditolak

	kesalahan"Silahkan pilih tanggal ,bulan dan tahun"	kesalahan"Silahkan pilih tanggal, bulan dan tahun"	

Tabel 4.16 Pengujian Laporan Pembelian Bulanan

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih bulan dan tahun laporan dan klik tombol filter Bulan : februari Tahun : 2012	Menampilkan data pembelian pada bulan februari 2012	Menampilkan data pembelian pada bulan februari 2012	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Bulan atau tahun tidak diisi	Tidak dapat menampilkan data dan menampilkan pesan kesalahan"Silahkan pilih bulan dan tahun"	Tidak dapat menampilkan data dan menampilkan pesan kesalahan"Silahkan pilih bulan dan tahun"	[√] Diterima [ ] Ditolak

Tabel 4.17 Pengujian Tambah Provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan provinsi yang belum ada Provinsi : Irian Jaya	provinsi baru dapat disimpan dan menampilkan pesan" Data provinsi berhasil disimpan"	provinsi baru dapat disimpan dan menampilkan pesan Data provinsi berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Provinsi yang dimasukkan sudah ada Provinsi : Irian Jaya	Provinsi tidak dapat disimpan dan menampilkan pesan kesalahan "Nama Provinsi	Provinsi tidak dapat disimpan dan menampilkan pesan kesalahan "Nama Provinsi	[√] Diterima [ ] Ditolak

	sudah terdaftar di *database*"	sudah terdaftar di *database*"	

Tabel 4.18 Pengujian Cari Provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan data provinsi dengan benar Provinsi : Lampung	Menampilkan data provinsi Lampung	Menampilkan data provinsi Lampung	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data provinsi salah Provinsi : Lampung 10	Data provinsi tidak ditemukan dan menampilkan pesan kesalahan "Data tidak ditemukan"	Data provinsi tidak ditemukan dan menampilkan pesan kesalahan "Data tidak ditemukan"	[√] Diterima [ ] Ditolak

Tabel 4.19 Pengujian Edit Provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Edit dengan Provinsi baru Provinsi Lama : Jakarta Provinsi Baru : Lampung	Provinsi dapat diedit dan menampilkan pesan kesalahan "Provinsi berhasil di*update*"	Provinsi dapat diedit dan menampilkan pesan kesalahan "Provinsi berhasil di*update*"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Edit dengan provinsi yang sudah ada Provinsi Lama : Lampung Provinsi Baru : Bali	provinsi tidak dapat diedit dan menampilkan pesan" provinsi sudah terdaftar di *database*"	provinsi tidak dapat diedit dan menampilkan pesan" provinsi sudah terdaftar di *database*"	[√] Diterima [ ] Ditolak

Tabel 4.20 Pengujian Tambah Kota

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan kota yang dengan benar Nama Provinsi : Lampung Kota : Suwoh	kota baru dapat disimpan dan menampilkan pesan" Data kota berhasil disimpan"	kota baru dapat disimpan dan menampilkan pesan" Data kota berhasil disimpan"	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kota yang dimasukkan sudah ada Kota : Pringsewu	Kota tidak dapat disimpan dan menampilkan pesan kesalahan "Kota sudah terdaftar di *database*"	Kota tidak dapat disimpan dan menampilkan pesan kesalahan "Kota sudah terdaftar di *database*"	[√] Diterima [ ] Ditolak

Tabel 4.21 Pengujian Cari Data Kota

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan data kota dengan benar Kota : Bangun Rejo	Menampilkan data kota	Menampilkan data kota	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Masukkan data kota yang tidak ada Kota : Majalaalla	Data kota tidak ditemukan dan menampilkan pesan kesalahan "Data tidak ditemukan"	Data kota tidak ditemukan dan menampilkan pesan kesalahan "Data tidak ditemukan"	[√] Diterima [ ] Ditolak

Tabel 4.22 Pengujian Edit Data Kota

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Edit dengan kota baru Provinsi : Lampung Kota : Bangun Rejo	kota baru dapat diedit dan menampilkan pesan "data kota berhasil di*update"*	kota baru dapat diedit dan menampilkan pesan "data kota berhasil di*update"*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Edit dengan kota yang sudah ada Provinsi : Lampung Kota : Pringsewu	Kota tidak dapat disimpan dan menampilkan pesan" Nama Kota sudah ada,silahkan ulangi"	Kota tidak dapat disimpan dan menampilkan pesan" Nama Kota sudah ada,silahkan ulangi"	[√] Diterima [ ] Ditolak

Tabel 4.23 Pengujian Daftar *Member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi dengan data yang benar Nama Lengkap : Johan Mamoru Alamat : Jl Makasar No 125 Provinsi : Bali Kota : Buleleng Kodepos : 14523 Email : johan@yahoo.com Kata Sandi : 1234569 Ketik ulang sandi : 1234569	Data member berhasil tersimpan, mengirimkan email aktivasi data member dan menampilkan informasi pendaftaran	Data member berhasil tersimpan, mengirimkan email aktivasi data member dan menampilkan informasi pendaftaran	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Kode pos diisi dengan selain angka Kodepos : klmk	Menampilkan pesan kesalahan"harus angka"	Menampilkan pesan kesalahan"harus angka"	[√] Diterima [ ] Ditolak
Email sudah terdaftar Email : ar3_di@yahoo.co.id	Menampilkan pesan kesalahan"email sudah digunakan"	Menampilkan pesan kesalahan"email sudah digunakan"	[√] Diterima [ ] Ditolak

Tabel 4.24 Pengujian *Login*

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi dengan data login yang benar Email : johan@yahoo.com Sandi :1234569	Menampilkan menu utama *member*	Dapat menampilkan menu utama *member*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi dengan data yang salah Email : a@yahoo.com Password : 1236547	Menampilkan pesan kesalahan"*Email* atau *password* anda salah, silahkan ulangi"	Dapat menampilkan pesan kesalahan"*Email* atau *password* anda salah, silahkan ulangi"	[√] Diterima [ ] Ditolak

Tabel 4.25 Pengujian Jumlah Pembelian

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi jumlah pembelian dengan benar Jumlah pembelian : 1	Data pembelian masuk ke keranjang belanja	Data pembelian masuk ke keranjang belanja	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**

Mengisi jumlah pembelian dengan selain angka Jumlah : xvt	Menampilkan pesan kesalahan"jumlah pesanan tidak valid"	Menampilkan pesan kesalahan"jumlah pesanan tidak valid"login anda"	[√] Diterima [ ] Ditolak
Mengisi jumlah pembelian melebihi stok Jumlah Pembelian : 1000	Menampilkan pesan kesalahan"maaf stok yang tersedia tidak mencukupi pesanan anda stok yang tersedia adalah.."	Menampilkan pesan kesalahan"maaf stok yang tersedia tidak mencukupi pesanan anda stok yang tersedia adalah.."	[√] Diterima [ ] Ditolak

Tabel 4.26 Pengujian Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data pengiriman dengan benar Nama Penerima : Ahmad Jaelani Alamat : Jl Merdeka no 1423 Provinsi : Lampung Kota : Bangunrejo Paket JNE : YES Kodepos : 35389	Menampilkan form alamat pengiriman dan detail pembelian	Menampilkan form alamat pengiriman dan detail pembelian	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
kodepos diisi selain angka kodepos : abc	Menampilkan pesan kesalahan"harus angka"	Menampilkan pesan kesalahan"harus angka"	[√] Diterima [ ] Ditolak

Tabel 4.27 Pengujian Retur

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data retur dengan benar Qty : 1 Alasan : kemasan rusak	data retur disimpan ke *database*	data retur disimpan ke *database*	[√] Diterima [ ] Ditolak

tidak sesuai			
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Qty melebihi jumlah pembelian QTY : 1000	Menampilkan pesan "Jumlah retur barang melebihi jumlah pembelian"	Menampilkan pesan "Jumlah retur barang melebihi jumlah pembelian"	[√] Diterima [ ] Ditolak

Tabel 4.28 Pengujian Data Pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data pembayaran dengan benar Jumlah pembayaran : 225000 No rek : 811255996332	data pembayaran disimpan ke *database*	data pembayaran disimpan ke *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Jumlah pembayaran diisi selain angka Jumlah pembayaran : klmo	Menampilkan pesan kesalahan"jumlah pembayaran tidak valid"	Menampilkan pesan kesalahan"jumlah pembayaran tidak valid"	[√] Diterima [ ] Ditolak
Jumlah pembayaran kurang dari jumlah tagihan Jumlah pembayaran : 500	Menampilkan pesan kesalahan"jumlah pembayaran tidak mencukupi"	Menampilkan pesan kesalahan"jumlah pembayaran tidak mencukupi"	[√] Diterima [ ] Ditolak

4.2.3.2 Kesimpulan Hasil Pengujian Alpha

Berdasarkan hasil pengujian dengan kasus uji contoh diatas dapat ditarik kesimpulan bahwa pada setiap proses masih memungkinkan untuk terjadinya kesalahan sintaks, tetapi secara fungsional sistem dapat mengeluarkan hasil yang diharapkan.

4.2.3.3 Pengujian Betha

Pengujian *betha* adalah pengujian yang dilakukan secara objektif dengan kata lain pengujian ini adalah pengujian secara langsung di lapangan. Metode penelitian yang digunakan adalah metode kuantitatif dimana calon *responden* diberikan pertanyaan atau kuesioner (terlampir) dengan ketentuan skala untuk setiap pertanyaan.

Kuisioner dibagikan kepada masing-masing calon responden, diantaranya pelanggan, pengunjung dan admin. Kuesioner ini terdiri dari 7 pertanyaan (contoh kuesioner dapat dilihat pada lampiran). Dari hasil kuesioner tersebut, lalu dihitunglah prosentase masing-masing jawaban dengan menggunakan rumus:

$$\mathbf{Y = (P/Q) * 100\ \%}$$

Keterangan :

P = banyaknya jawaban responden setiap pertanyaan

Q = jumlah responden

Y = nilai prosentase

Kuesioner ini terdiri dari 7 pertanyaan dengan menggunakan empat pilihan jawaban yang mewakili dari tujuan akhir yang ingin dicapai dalam pembangunan *e-commerce* dalam penjualan obat pada Toko badak jaya, dimana daftar pertanyaan yang diberikan untuk masing-masing responden adalah sebagai berikut:

A. Pengujian pada admin

Pengujian betha untuk admin digunakan metode wawancara, dimana kuesioner terdiri dari 7 pertanyaan dengan daftar pertanyaan yang diberikan adalah sebagai berikut:

1. Apakah tampilan website untuk backend sudah dan mudah digunakan (*user friendly*)?
   Jawab :

2. Bagaimana menurut pendapat anda, apakah website ini dapat membantu dalam mengolah data produk dan pelanggan?
   Jawab :

3. Menurut anda apakah dengan adanya website ini dapat membantu dalam memasarkan produk di Toko Badak Jaya?
   Jawab :

4. Apakah proses transaksi pemesanan dan pembayaran menjadi lebih mudah?
   Jawab :

5. Apakah *website* ini bisa dengan mudah diakses kapan saja selama masih ada jaringan internet?(jelaskan)
   Jawab :

   _____

   _____

6. Apakah website ini dapat membantu dalam proses pembuatan laporan?
   Jawab :

   _____

   _____

7. Apakah dengan adanya website ini pemberian informasi antara toko dan pelanggan menjadi lebih mudah?
   Jawab :

   _____

   _____

Dari hasil wawancara yang telah dilakukan pada admin, dapat disimpulkan bahwa tampilan dari website yang dibangun untuk bagian admin mudah dipahami serta cukup membantu admin dalam proses pengolahan data produk, pelanggan dan transaksi serta pembuatan laporan.

B. Pengujian pada member

Pengujian betha untuk member digunakan metode kuesioner, dimana kuesioner terdiri dari 7 pertanyaan dengan menggunakan 4 pilihan jawaban, dimana daftar pertanyaan yang diberikan adalah sebagai berikut:

Tabel 4.29 Tabel Skala Pertanyaan

Pertanyaan	Tingkat Skala Pertanyaan
1. Apakah dengan adanya *website* ini dapat mempermudah member untuk mengetahui produk terbaru pada Toko Badak jaya ?	A. Sangat setuju
	B. Setuju
	C. Biasa saja
	D. Kurang setuju
2. Apakah anda setuju dengan adanya *website* ini dapat memberikan solusi alternatif terhadap pemesanan produk pada Toko Badak jaya (terutama yang jauh dari lokasi) ?	A. Sangat setuju
	B. Setuju
	C. Biasa saja
	D. Kurang setuju
3. Apakah anda setuju bahwa website ini memudahkan pelanggan dalam melakukan transaksi pembelian produk?	A. Sangat setuju
	B. Setuju
	C. Biasa saja
	D. Kurang setuju
4. Apakah anda setuju bahwa *website* ini memberikan keamanan bagi Anda pada saat melakukan transaksi pembayaran?	A. Sangat setuju
	B. Setuju
	C. Biasa saja
	D. Kurang setuju
5. Apakah anda setuju *website* ini dapat dengan mudah diakses kapan saja selama masih ada jaringan internet?	A. Sangat setuju
	B. Setuju
	C. Biasa saja
	D. Kurang setuju
6. Apakah anda setuju bahwa informasi produk	A. Sangat setuju
	B. Setuju

yang diberikan oleh website ini sesuai dengan informasi yang anda butuhkan?	C. Biasa saja D. Kurang setuju
7. Apakah anda setuju *website* ini dapat dengan mudah digunakan?	A. Sangat setuju B. Setuju C. Biasa saja D. Kurang setuju

Berikut adalah hasil perhitungan masing-masing jawaban dari setiap pertanyaan yang di ajukan:

Tabel 4.30 Pertanyaan no 1

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.1	Apakah dengan adanya *website* ini dapat mempermudah member untuk mengetahui produk terbaru pada Toko Badak jaya ?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	2	8	0	0
sampel	10	10	10	10
Jumlah persentase	20%	80%	0%	0%

Tabel 4.31 Pertanyaan no 2

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.2	Apakah anda setuju dengan adanya *website* ini dapat memberikan solusi alternatif terhadap pemesanan produk pada Toko Badak jaya (terutama yang jauh dari lokasi) ?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	3	7	0	0
sampel	10	10	10	10
Jumlah persentase	30%	70%	0%	0%

Tabel 4.32 Pertanyaan no 3

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.3	Apakah anda setuju bahwa website ini memudahkan pelanggan dalam melakukan transaksi pembelian produk?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	2	5	3	0
sampel	10	10	10	10
Jumlah persentase	20%	50%	30%	0%

Tabel 4.33 Pertanyaan no 4

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.4	Apakah anda setuju bahwa *website* ini memberikan keamanan bagi Anda pada saat melakukan transaksi pembayaran?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	0	7	3	0
sampel	10	10	10	10
Jumlah persentase	0%	70%	30%	0%

Tabel 4.34 Pertanyaan no 5

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.5	Apakah anda setuju *website* ini dapat dengan mudah diakses kapan saja selama masih ada jaringan internet?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	3	7	0	0
sampel	10	10	10	10
Jumlah persentase	30%	70%	0%	0%

Tabel 4. 35 Pertanyaan no 6

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.6	Apakah anda setuju bahwa informasi produk yang diberikan oleh website ini sesuai dengan informasi yang anda butuhkan?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	1	6	3	0

sampel	10	10	10	10
Jumlah persentase	10%	60%	30%	0%

Tabel 4. 36 Pertanyaan no 7

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.7	Apakah anda setuju *website* ini dapat dengan mudah digunakan?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	0	10	0	0
sampel	10	10	10	10
Jumlah persentase	0%	100%	0%	0%

C. Pengujian pada pengunjung

Pengujian betha untuk pengunjung digunakan metode kuesioner, dimana kuesioner terdiri dari 7 pertanyaan dengan menggunakan 4 pilihan jawaban, dimana daftar pertanyaan yang diberikan adalah sebagai berikut:

Tabel 4.37 Tabel Skala Pertanyaan Pengunjung

Pertanyaan	**Tingkat Skala Pertanyaan**
1. Apakah dengan adanya *website* ini dapat mempermudah mengetahui produk terbaru pada Toko Badak jaya ?	A. Sangat setuju B. Setuju C. Biasa saja D. Kurang setuju
2. Apakah anda setuju bahwa tampilan website menarik dan mudah dpahami (*user friendly*)	A. Sangat setuju B. Setuju C. Biasa saja D. Kurang setuju

3. Apakah anda setuju bahwa *website* ini dapat memberikan informasi sesuai dengan yang diinginkan?	A. Sangat setuju
	B. Setuju
	C. Biasa saja
	D. Kurang setuju
4. Apakah anda setuju dengan adanya website ini Toko Badak jaya dapat memperluas pangsa pasarnya?	A. Sangat setuju
	B. Setuju
	C. Biasa saja
	D. Kurang setuju
5. Apakah *website* ini bisa dengan mudah diakses kapan saja selama masih ada jaringan internet?	A. Sangat setuju
	B. Setuju
	C. Biasa saja
	D. Kurang setuju
6. Apakah anda setuju bahwa website ini telah layak untuk di-online-kan?	A. Sangat setuju
	B. Setuju
	C. Biasa saja
	D. Kurang setuju
7. Apakah *website* ini dapat dengan mudah digunakan?	A. Sangat setuju
	B. Setuju
	C. Biasa saja
	D. Kurang setuju

Berikut adalah hasil perhitungan masing-masing jawaban dari setiap pertanyaan, yakni sebagai berikut:

Tabel 4. 38 Pertanyaan no 1

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.1	Apakah dengan adanya *website* ini dapat mempermudah mengetahui produk terbaru pada Toko Badak jaya ?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	3	7	0	0
Populasi sampel	10	10	10	10
Jumlah persentase	30 %	70 %	0 %	0%

Tabel 4. 39 Pertanyaan no 2

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.2	Apakah anda setuju bahwa tampilan website menarik dan mudah dpahami (*user friendly*)			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	2	6	2	0
Populasi sampel	10	10	10	10
Jumlah persentase	20 %	60 %	20 %	0

Tabel 4. 40 Pertanyaan no 3

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.3	Apakah anda setuju bahwa *website* ini dapat memberikan informasi sesuai dengan yang diinginkan?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	4	6	0	0
Populasi sampel	10	10	10	10
Jumlah persentase	40 %	60 %	0 %	0%

Tabel 4. 41 Pertanyaan no 4

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.4	Apakah anda setuju dengan adanya website ini Toko Badak jaya dapat memperluas pangsa pasarnya?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	3	3	4	0
Populasi sampel	10	10	10	10

Jumlah persentase	30 %	30 %	40 %	0%

Tabel 4. 42 Pertanyaan no 5

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.5	Apakah *website* ini bisa dengan mudah diakses kapan saja selama masih ada jaringan internet?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	6	3	1	0
Populasi sampel	10	10	10	10
Jumlah persentase	60 %	30 %	10 %	0%

Tabel 4. 43 Pertanyaan no 6

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.6	Apakah anda setuju bahwa website ini telah layak untuk di-online-kan?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	4	6	0	0
Populasi sampel	0	0	0	0
Jumlah persentase	40 %	60 %	0 %	0

Tabel 4. 44 Pertanyaan no 7

PERTANYAAN DAN HASIL PENGUJIAN				
Pertanyaan No.7	Apakah *website* ini dapat dengan mudah digunakan?			
Kategori jawaban	**Sangat setuju**	**Setuju**	**Biasa Saja**	**Kurang setuju**
Frekuensi jawaban	3	5	2	0
Populasi sampel	0	0	0	0
Jumlah persentase	30 %	50 %	20 %	0%

4.2.3.4 Kesimpulan Hasil Pengujian Betha

Kesimpulan yang dapat ditarik dari hasil pengujian *Betha* adalah :

1. Aplikasi yang dibuat dapat mempermudah dalam melihat produk terbaru yang ditawarkan oleh Toko Badak jaya.
2. Aplikasi yang dibuat dapat mempermudah dan mempercepat dalam pemesanan produk.
3. Aplikasi yang dibuat dapat memberikan keamanan pada saat melakukan transaksi pembayaran.
4. *Website* ini dapat dengan mudah digunakan oleh berbagai responden.