BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Pada bab ini akan dilakukan implementasi dan pengujian terhadap web yang telah dirancang. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan dengan menggunakan script PHP. Tujuan Implementasi adalah untuk menerapkan hasil perancangan dengan tujuan agar maksud dan tujuan pembuatan website e-commerce masyarakat seksi pengolahan data dan informasi dapat tercapai. Setelah implementasi dilakukan, maka tahap pengujian terhadap pembuatan website *e-commerce* seksi pengolahan data dan informasi yang telah dibangun dilakukan guna mengetahui apakah maksud dan tujuan yang ingin dicapai telah terpenuhi sehingga dapat ditarik kesimpulan.

4.1.1 Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan oleh member baik itu dari sisi pengunjung dan administrator bersifat relatif, tergantung dari spesifikasi yang dimiliki dan akan digunakan, namun sebagai gambaran berikut sistem minimal yang dapat dipakai :

1. *Processor* AMD athlon X2 240 2.8 GHz
2. Kapasitas RAM 1 GB
3. Kapasitas *Harddisk* 160 GB

4. Monitor
5. *Mouse*
6. *Keyboard*
7. Modem kecepatan akses 384 kbps
8. Printer

4.1.2 Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang dipakai dalam pembangunan aplikasi penjualan jaket *online* pada PD Ridha Lestari dapat dilihat seperti pada dibawah ini:

1. Sistem operasi *Windows XP SP2*
2. *Web browser Mozila firefox* 3.6
3. *Macromedia dreamweaver* 8.0
4. *XAMPP* versi 1.6.3 sebagai *Web servernya*

4.1.3 Implementasi Basis Data

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

1. Tabel Admin

Struktur tabel admin pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `admin` (
  `id_admin` int(10) unsigned NOT NULL auto_increment,
  `username` varchar(45) collate latin1_general_ci NOT NULL,
  `password` varchar(35) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`id_admin`),
  UNIQUE KEY `username` (`username`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
AUTO_INCREMENT=4 ;
```

2. **Tabel Detail Pembelian**

Struktur tabel detail pembelian pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `detail_pembelian` (
  `id_detail_pembelian` int(10) unsigned NOT NULL
auto_increment,
  `id_produk` int(10) unsigned NOT NULL,
  `id_pembelian` int(10) unsigned NOT NULL,
  `qty` double NOT NULL,
  `harga` double NOT NULL,
  `diskon` double NOT NULL,
  `harga_satuan` double NOT NULL,
  `status` varchar(15) collate latin1_general_ci NOT NULL
default '_OK',
  PRIMARY KEY  (`id_detail_pembelian`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=24 ;

ALTER TABLE `detail_pembelian`
  ADD CONSTRAINT `detail_pembelian_ibfk_1` FOREIGN KEY
(`id_produk`) REFERENCES `produk` (`id_produk`),
  ADD CONSTRAINT `detail_pembelian_ibfk_2` FOREIGN KEY
(`id_pembelian`) REFERENCES `pembelian` (`id_pembelian`);
```

3. Tabel Kabkota

Struktur tabel kabkota pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `kabkota` (
  `id_kabkota` int(10) unsigned NOT NULL auto_increment,
  `provinsi` int(11) NOT NULL,
  `nama_kabkota` varchar(75) collate latin1_general_ci NOT
NULL,
  `ongkos_kirim` double NOT NULL,
  PRIMARY KEY  (`id_kabkota`),
  UNIQUE KEY `nama_kabkota` (`nama_kabkota`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=427 ;

ALTER TABLE `kabkota`
  ADD CONSTRAINT `kabkota_ibfk_1` FOREIGN KEY (`provinsi`)
REFERENCES `provinsi` (`id_provinsi`);
```

4. Tabel Kategori

Struktur tabel kategori pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `kategori` (
  `id_kategori` int(10) unsigned NOT NULL auto_increment,
  `nama_kategori` varchar(75) collate latin1_general_ci NOT
NULL,
  PRIMARY KEY  (`id_kategori`),
  UNIQUE KEY `nama_kategori` (`nama_kategori`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=7 ;
```

5. Tabel Member

Struktur tabel member pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `member` (
  `id_member` int(10) unsigned NOT NULL auto_increment,
  `nama_lengkap` varchar(45) collate latin1_general_ci NOT
NULL,
  `alamat` varchar(255) collate latin1_general_ci NOT NULL,
  `kode_pos` varchar(10) collate latin1_general_ci NOT NULL,
  `email` varchar(45) collate latin1_general_ci NOT NULL,
  `tanggal_lahir` date NOT NULL,
  `no_telepon` varchar(25) collate latin1_general_ci NOT NULL,
  `password` varchar(35) collate latin1_general_ci NOT NULL,
  `id_kota` int(11) NOT NULL,
  PRIMARY KEY  (`id_member`),
  UNIQUE KEY `email` (`email`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=9 ;

ALTER TABLE `member`
  ADD CONSTRAINT `member_ibfk_2` FOREIGN KEY (`id_kota`)
REFERENCES `kabkota` (`id_kabkota`);
```

6. Tabel Pembelian

Struktur tabel pembelian pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `pembelian` (
  `id_pembelian` int(10) unsigned NOT NULL auto_increment,
  `jumlah` int(10) unsigned NOT NULL,
  `tgl_pembelian` datetime NOT NULL,
  `id_member` int(10) unsigned NOT NULL,
  `status` varchar(45) collate latin1_general_ci NOT NULL,
  `id_kota` int(11) NOT NULL,
  `pembayaran` varchar(20) collate latin1_general_ci NOT NULL,
  `alamat` varchar(255) collate latin1_general_ci NOT NULL,
  `kodepos` varchar(10) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`id_pembelian`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=14 ;

ALTER TABLE `pembelian`
  ADD CONSTRAINT `pembelian_ibfk_1` FOREIGN KEY (`id_member`)
REFERENCES `member` (`id_member`) ON DELETE CASCADE ON UPDATE
CASCADE;
```

7. Tabel Produk

Struktur tabel produk pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `produk` (
  `id_produk` int(10) unsigned NOT NULL auto_increment,
  `nama_produk` varchar(100) collate latin1_general_ci NOT
NULL,
  `id_kategori` int(10) unsigned NOT NULL,
  `harga` double NOT NULL,
  `ukuran` varchar(45) collate latin1_general_ci NOT NULL,
  `gambar` varchar(255) collate latin1_general_ci NOT NULL,
  `warna` varchar(100) collate latin1_general_ci NOT NULL,
  `berat` double NOT NULL,
  `stok` double NOT NULL,
  `diskon` double NOT NULL default '0',
  `view` int(11) NOT NULL default '0',
  PRIMARY KEY  (`id_produk`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=15 ;

ALTER TABLE `produk`
  ADD CONSTRAINT `produk_ibfk_1` FOREIGN KEY (`id_kategori`)
REFERENCES `kategori` (`id_kategori`) ON DELETE CASCADE ON
UPDATE CASCADE;
```

8. Tabel Provinsi

Struktur tabel provinsi pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `provinsi` (
  `id_provinsi` int(10) unsigned NOT NULL auto_increment,
  `nama_provinsi` varchar(75) collate latin1_general_ci NOT
NULL,
  PRIMARY KEY  (`id_provinsi`),
  UNIQUE KEY `nama_provinsi` (`nama_provinsi`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=38 ;
```

9. Tabel Retur

Struktur tabel retur pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `retur` (
  `id_retur` int(10) unsigned NOT NULL auto_increment,
  `id_member` int(10) unsigned NOT NULL,
  `id_pembelian` int(10) unsigned NOT NULL,
  `id_produk` int(11) NOT NULL,
  `tgl_retur` datetime NOT NULL,
  `ket_retur` varchar(255) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`id_retur`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=4 ;
```

10. Tabel detail_retur

Struktur tabel detail retur pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `temp` (
  `id_produk` int(10) default NULL,
  `nama_produk` varchar(255) collate latin1_general_ci default
NULL,
  `id_kategori` int(4) default NULL,
  `harga` double default NULL,
  `gambar` varchar(255) collate latin1_general_ci default
NULL,
  `jumlah` int(8) default NULL,
  `id_session` varchar(255) collate latin1_general_ci default
NULL,
  `id_user` int(11) default NULL
```

```
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;

ALTER TABLE `detail_retur`
  ADD CONSTRAINT `detail_retur_ibfk_1` FOREIGN KEY
(`id_retur`) REFERENCES `retur` (`id_retur`);
```

11. Tabel Bank

Struktur tabel bank pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `bank` (
  `id_bank` int(10) unsigned NOT NULL auto_increment,
  `no_rek` int(11) NOT NULL,
  `nama_bank` varchar(45) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`id_bank`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=1 ;
```

12. Tabel Pembayaran

Struktur tabel pembayaran pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `pembayaran` (
  `id_pembayaran` int(10) unsigned NOT NULL auto_increment,
  `id_bank` int(11) NOT NULL,
  `id_ongkir` int(11) NOT NULL,
  `accountpaypal` varchar(45) collate latin1_general_ci NOT
NULL,
  `carabayar` varchar(45) collate latin1_general_ci NOT
NULL,
  PRIMARY KEY  (`id_pembayaran`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=1 ;

ALTER TABLE `pembayaran`
  ADD CONSTRAINT `pembayaran_ibfk_2` FOREIGN KEY
(`id_ongkir`) REFERENCES `ongkos_kirim` (`id_ongkir`),
  ADD CONSTRAINT `pembayaran_ibfk_1` FOREIGN KEY
(`id_bank`) REFERENCES `bank` (`id_bank`);
```

13. Tabel Kurs

Struktur tabel kurs pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `kurs` (
  `Id_kurs` int(10) unsigned NOT NULL,
  `kurs` double NOT NULL,
  `update` double NOT NULL,
  `tanggal` date NOT NULL,
  PRIMARY KEY  (`Id_kurs`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;
```

14. Tabel Jasa_pengiriman

Struktur tabel jasa_pengiriman pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `jasa_pengiriman` (
  `id_jaspeng` int(10) unsigned NOT NULL auto_increment,
  `namajaspeng` varchar(45) collate latin1_general_ci NOT
NULL,
  PRIMARY KEY  (`id_jaspeng`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=1 ;
```

15. Tabel Paket

Struktur tabel paket pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `paket` (
  `id_paket` int(10) unsigned NOT NULL auto_increment,
  `id_jaspeng` int(10) unsigned NOT NULL,
  `namapeket` varchar(45) collate latin1_general_ci NOT
NULL,
  PRIMARY KEY  (`id_paket`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci AUTO_INCREMENT=1 ;

ALTER TABLE `paket`
  ADD CONSTRAINT `paket_ibfk_1` FOREIGN KEY (`id_jaspeng`)
REFERENCES `jasa_pengiriman` (`id_jaspeng`);
```

16. Tabel Ongkos_kirim

Struktur tabel detail retur pada pembangunan situs *e-commerce* penjualan jaket PD Ridha Lestari adalah sebagai berikut :

```
CREATE TABLE `ongkos_kirim` (
  `id_ongkir` int(10) unsigned NOT NULL auto_increment,
  `id_kabkota` int(10) unsigned NOT NULL,
  `ongkos` varchar(45) collate latin1_general_ci NOT NULL,
  PRIMARY KEY  (`id_ongkir`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
AUTO_INCREMENT=1 ;

ALTER TABLE `ongkos_kirim`
  ADD CONSTRAINT `ongkos_kirim_ibfk_1` FOREIGN KEY
(`id_kabkota`) REFERENCES `kabkota` (`id_kabkota`);
```

4.1.4 Implementasi Antar Muka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk admin dapat dilihat dilihat pada tabel 4.1

Tabel 4.1 Implementasi Antar Muka Admin

Menu	Deskripsi	Nama file
Login	Digunakan Admin untuk masuk ke halaman Admin.	Login.php
Home	Tampilan Halaman Utama Admin.	Index.php
Admin	Digunakan untuk mengubah admin, mengganti password.	Admin.php
Kategori	Digunakan admin untuk menambah dan mengedit kategori produk.	Kategori.php
Produk	Digunakan admin untuk menambah, dan mengedit produk.	Produk.php
Member	Digunakan Admin untuk mengubah status member, mengubah data member, menghapus data member.	Member.php
Provinsi	Digunakan Admin untuk mengelola data provinsi	Provinsi.php
Kota	Digunakan untuk melihat, menambah, mengedit data kota.	Kota.php

Tips	Digunakan Admin untuk mengelola data Tips	Tips.php
Transaksi	Digunakan Admin untuk mengelola data transaksi(mengganti status), menghapus data transaksi, memberikan konfirmasi pengiriman barang	Transaksi.php
Kontak	Digunakan untuk melihat pengunjung, member yang menghubungi admin.	Hubungi.php
Laporan	Digunakan untuk melihat, mencetak data laporan	Laporan.php
Logout	Digunakan untuk keluar dari menu admin	Logout.php

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengunjung dan member dapat dilihat dilihat pada tabel 4.2

Tabel 4.2 Implementasi Antar Muka Pengunjung

Menu	Deskripsi	Nama file
Halaman Utama	Tampilan halaman utama website.	index.php
Produk	Digunakan untuk menampilkan koleksi produk jaket	Produk.php
Retur	Digunakan untuk mengetahui cara melakukan proses retur.	Retur.php
Cara Belanja	Digunakan untuk melakukan pendaftaran member.	Carabelanja.php
Tips	Digunakan pengunjung untuk melihat tips yang diberikan perusahaan	Tips.php
Kontak	Digunakan Pengunjung untuk mengetahui Kontak.	Kontak.php
Tentang Kami	Digunakan untuk melihat profil perusahaan	Tentangkami.php
Cara Pembayaran	Digunakan untuk melihat cara pembayaran yang berlaku di website ini.	Carabayar.php
Cara Belanja	Digunakan untuk melihat cara	Carabelanja.php

	bagaimana belanja.	
Daftar	Digunakan pengunjung untuk melakukan registrasi	Daftar.php
Login	Digunakan jika agen ingin login	login.php

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk member dapat dilihat dilihat pada tabel 4.3

Tabel 4.3 Implementasi Antar Muka Member

Menu	Deskripsi	Nama file
Halaman Utama	Tampilan halaman utama website.	index.php
Produk	Dugunakan untuk menampilkan koleksi produk jaket	Produk.php
Retur	Digunakan untuk melihat prosedur dan batasan produk yang layak diretur	Retur.php
Cara belanja	Digunakan untuk menetahui cara belnja di website ini.	carabelanja.php
Kontak	Digunakan member untuk menghubungi admin jika ada perihal yang ingin ditanyakan atau memberikan saran dan kritik.	Kontak.php
Kategori	Digunakan untuk melihat produk yang berada dalam kategori	Kategori.php
Edit Profil	Digunakan untuk mengubah data member	Profil.php
History pembelian	Digunakan untuk melihat history pemeblian	History_beli.php
History retur	Digunakan untuk melakukan proses retur	History_retur.php
Keranjang belanja	Digunakan untuk melihat isi keranjang belanja	Cek_belanjaan.php
Tips	Digunakan pengunjung untuk melihat tips yang diberikan perusahaan	Tips.php
Cara Pembayaran	Digunakan untuk melihat cara pembayaran	Carabayar.php
Logout	Digunakan jika member ingin keluar dari halaman member.	Lib_func.php

Member	Digunakan member untuk melihat profil atau merubah data member.	Member.php
History Pembelian	Digunakan member untuk melihat data transaksi yang pernah dilakukan dan melakukan pembayaran.	History.php

4.2 Pengujian Perangkat Lunak

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

Pengujian perangkat lunak ini menggunakan pengujian *black box* yang berfokus pada persyaratan fungsional perangkat lunak dan teknik yang digunakan adalah *equivalence partitioning* dimana pengujian perangkat lunak dilakukan dengan cara membagi data masukan dari unit perangkat lunak menjadi beberapa partisi data dari mana test case dapat diturunkan. Pada prinsipnya, uji kasus dirancang untuk menutupi setiap partisi minimal sekali. Teknik ini mencoba mendefinisikan kasus uji yang mengungkap kelas kesalahan, sehingga mengurangi jumlah kasus uji yang harus di kembangkan.

4.3 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara *alpha* dan *beta*. Skenario pengujian untuk member dapat dilihat pada tabel 4.4.

Tabel 4.4 Menjelaskan Skenario Pengujian Aplikasi Member.

Kelas Uji	Butir Uji	Jenis Pengujian
Registrasi	Menampilkan *form*	*Black Box*

	member baru	
	Isi *form* pendaftaran	*Black Box*
	Simpan data baru	*Black Box*
Login	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Logout	Menghapus *sessionuser*	*Black Box*
Ubah profil	Menampilkan *form* ubah profil	*Black Box*
	Ubah data *member*	*Black Box*
	Simpan perubahan	*Black Box*
Menu produk	Menampilkan data produk	*Black Box*
	Menampilkan data detail produk	*Black Box*
Menu halaman utama	Menampilkan profil perusahaan secara umum	*Black Box*
Menu kontak	Menampilkan form menghubungi perusahaan(admin perusahaan)	*Black Box*
Menu Tips	Menampilkan tips	*Black Box*
Menu keranjang Belanja	Menampilkan data detail Pemesanan(transaksi)	*Black Box*
	Mengubah data detail pemesanan(transaksi)	*Black Box*
	Menghapus data detail pemesanan (transaksi)	*Black Box*
Menu history pembelian	Menampilkan data history pemesanan	*Black Box*

Skenario pengujian untuk admin ada beberapa butir uji adapun jenis pengujianya mengunakan metode *Black Box* untuk lebih jelasnya dapat dilihat pada tabel 4.5

Tabel 4.5 Menjelaskan Skenario Pengujian Aplikasi Admin

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data login	*Black Box*

	Verifikasi *username*	*Black Box*
	Verivikasi *password*	*Black Box*
Logout	Menghapus *session user*	*Black Box*
Menu Admin	Menampilkan *form* ganti data admin	*Black Box*
Menu Kategori	Tampilkan list kategori	*Black Box*
	Tambah data Kategori	*Black Box*
	Ubah data Kategori	*Black Box*
Menu Provinsi	Tampilkan data provinsi	*Black Box*
	Tambah data provinsi	*Black Box*
	Ubah data provinsi	*Black Box*
Menu Kota	Tampilkan data kota	*Black Box*
	Tambah data kota	*Black Box*
	Ubah data kota	*Black Box*
Menu Produk	Tampilkan data produk	*Black Box*
	Ubah data produk	*Black Box*
	Lihat data detail produk	*Black Box*
	Tambah Data Produk	*Black Box*
Menu Transaksi	Tampilkan data Transaksi	*Black Box*
	Tambah data transaksi	*Black Box*
	Ubah data transaksi	*Black Box*
	Hapus data transaksi	*Black box*
Menu Kontak	Tampilakn data kontak	*Black box*
	Hapus data kontak	*Black box*
	Replay data kontak	*Black box*
Menu Tips	Tampilakan data tips	*Black box*
	Ubah data tips	*Black box*
	Hapus data tips	*Black box*
	Tambah data tips	*Black box*
Menu pembayaran	Tampilkan data pembayaran	*Black box*
	Hapus data pembayaran	*Black box*
Menu Member	Tampilkan list data member	*Black box*
	Hapus data member	*Black box*

4.3.1 Pengujian Alpha

Hasil implementasi perangkat lunak sistem informasi ini diuji pada pemilik toko maupun pengunjung yang berbasis *online* melalui spesifikasi perangkat sistem agar sistem dapat berjalan dengan baik.

Pengujian perangkat lunak sistem penerapan *e-commerce* menggunakan data uji berdasarkan form yang terdapat didalam aplikasi dengan menggunakan *equivalence partioning* dan *Cause effect testing* sebgai berikut :

4.3.1.1 Equivalence Partioning

4.3.1.1.1 Equivalence Partioning Untuk Admin

1. Pengujian Login Admin

Berikut dapat dilihat pengujian dari login member dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.6 :

Tabel 4.6 Pengujian login admin (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Email contoh masukan : username : ridhalestari@yahoo.com	[A-Z|a-z|0-9|@|.]	Data Email dapat diterima jika input data ; 1.Email : 5 sampai 30 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4 Harus mengandung simbol @ dan .	[√] Diterima [ ] Ditolak
Password contoh masukan : Password : admin1234	[A-Z|a-z|0-9] 5-32 char	Data Password dapat diterima jika input data ; 1.Nama 5 sampai 32	[√] Diterima [ ] Ditolak

		char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	

Berikut dapat dilihat pengujian dari login admin dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.7 :

Tabel 4.7 Pengujian logon admin (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Email contoh masukan : Username: ridha Username :	[A-Z|a-z|0-9|] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter.	[ ] Diterima [√] Ditolak
Password contoh masukan : Password : yy Password:zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt Password :	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter.	[ ] Diterima [√] Ditolak

2. Pengujian pengolahan data kategori

Berikut dapat dilihat pengujian dari pengolahan data kategori dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.8 :

Tabel 4.8 Pengujian pengolahan data kategori (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Kategori contoh masukan : kategori : jaket anak	[A-Z|a-z|0-9|@|.] 1-32 char	Data Kategori dapat diterima jika input data ; 1. 1 sampai 30 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. klik icon edit untuk mengedit kategori yang sudah ada atau klik tombol tambah untuk menambahkan kategori	[√] Diterima [ ] Ditolak

Berikut dapat dilihat pengujian dari pengolahan data kategori dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.9 :

Tabel 4.9 Pengujian pengolahan data kategori (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Kategori contoh masukan : kategori : blablabla kategori :	kosong > 32 char	menghasilkan pesan kesalahan apabila menekan tombol hapus jika kategori tersebut berrelasi dengan data yang lainnya yang berhubungan dengan transaksi.Menampilkan pesan kesalahan data kosong	[ ] Diterima [√] Ditolak

4. Pengujian Pengolahan Data Produk

Pengujian pengolahan data produk dilakukan ketika sudah menginputkan data kategori sebelumnaya. Berikut dapat dilihat pengujian dari pengolahan data produk dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.10 :

Tabel 4.10 Pengujian pengolahan data produk (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
kategori contoh masukan : jaket dewasa	pilih [kategori..]	Memilih data kategori yang sebelumnya telah di inputkan	[√] Diterima [ ] Ditolak
produk contoh masukan : produk : jaket kulit abu-abu	[A-Z|a-z|0-9|@|.] 1-32 char	Data produk dapat diterima jika input data ; 1. 1 sampai 30 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	[√] Diterima [ ] Ditolak

Berikut dapat dilihat pengujian dari pengolahan data produk dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.11 :

Tabel 4.11 Pengujian pengolahan data produk (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
kategori	kosong	Menampilkan pesan error jika data belum ada atau kosong	[ ] Diterima [√] Ditolak
produk contoh masukan : produk : fsdjfsjdfgjsgfjsgdfjhgd shfgdsjhfghjsdgfjhsdg fsgfhsgfjsgfjsgdfjhsgd	kosong > 32 char	menghasilkan pesan kesalahan apabila menekan tombol hapus jika produk tersebut berrelasi dengan data yang	[ ] Diterima [√] Ditolak

fsdjgffsjfkjsdfkshjfh produk :		lainnya yang berhubungan dengan transaksi. menampilkan pesan kesalahan data kosong	

5. Pengujian Pengolahan Data Provinsi

Berikut dapat dilihat pengujian dari pengolahan data provinsi dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.12 :

Tabel 4.12 Pengujian pengolahan data provinsi (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Provinsi contoh masukan : Provinsi : Jawa timur	[A-Z|a-z|0-9|@|.] 1-32 char	Data Provinsi dapat diterima jika input data ; 1. 1 sampai 30 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. klik icon edit untuk mengedit provinsi yang sudah ada atau klik tombol tambah untuk menambahkan provinsi	[√] Diterima [ ] Ditolak

Berikut dapat dilihat pengujian dari pengolahan data provinsi dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.13 :

Tabel 4.13 Pengujian pengolahan data detail provinsi (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Provinsi contoh masukan : provinsi : fsdjfsjdfgjsgfjsgdfjhgd shfgdsjhfghjsdgfjhsdg fsgfhsgfjsgfjsgdfjhsgd fsdjgffsjfkjsdfkshjfh kategori :	kosong > 32 char	Menghasilkan pesan kesalahan apabila menekan tombol hapus jika provinsi tersebut berelasi dengan data yang lainnya yang berhubungan dengan transaksi. menampilkan pesan kesalahan data kosong	[ ] Diterima [√] Ditolak

6. Pengujian Pengolahan Data Kota

Pengujian pengolahan data kota dilakukan ketika sudah menginputkan data provinsi Berikut dapat dilihat pengujian dari pengolahan data kota dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.14 :

Tabel 4.14 Pengujian pengolahan data kota (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Provinsi contoh masukan : provinsi jawa barat	pilih [provinsi..]	Memilih data provinsi yang sebelumnya telah di inputkan	[√] Diterima [ ] Ditolak
Nama kota contoh masukan : kota : garut	[A-Z|a-z|0-9|@|.] 1-32 char	Data kota dapat diterima jika input data ; 1. 1 sampai 30 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. klik icon edit untuk mengedit nama kota yang sudah ada atau klik tombol tambah	[√] Diterima [ ] Ditolak

		untuk menambahkan produk	

Berikut dapat dilihat pengujian dari pengolahan data kota dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.15 :

Tabel 4.15 Pengujian pengolahan data kota (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Provinsi	kosong	Menampilkan pesan error jika data belum ada atau kosong	[ ] Diterima [√] Ditolak
Nama Kota contoh masukan : nama kota : fsdjfsjdfgjsgfjsgdfjhgd shfgdsjhfghjsdgfjhsdg fsgfhsgfjsgfjsgdfjhsgd fsdjgffsjfkjsdfkshjfh nama kota :	kosong > 32 char	menghasilkan pesan kesalahan apabila menekan tombol hapus jika nama kota tersebut berrelasi dengan data yang lainnya yang berhubungan dengan transaksi. menampilkan pesan kesalahan data kosong	[ ] Diterima [√] Ditolak

7. Pengujian Pengolahan Data Kurs

Berikut dapat dilihat pengujian dari pengolahan data kurs dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.16 :

Tabel 4.16 Pengujian pengolahan data kurs (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Harga Dollar contoh masukan : Harga Dollar : 10000	[0-9]	Data no rekening di terima jika data yang diinputkan adalah data angka	[√] Diterima [ ] Ditolak

Berikut dapat dilihat pengujian dari pengolahan data rekening dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.17 :

Tabel 4.17 Pengujian pengolahan data kurs (data invalid)

Kasus dan Hasil Uji (Data inValid)			
Data Masukan	invalid Class	Pengamatan	Kesimpulan
Harga Dollar contoh masukan : harga dolar : ceban	[A-Z|a-z] symbol	Menampilkan pesan jika data yang di inputkan huruf atau simbol.	[ ] Diterima [√] Ditolak

8. Pengujian Pengolahan Data Laporan

Pengujian laporan dapat dilakukan apabila tekan tombol printpdf untuk melakukan pengolahan data transaksi. Berikut dapat dilihat pengujian dari pengolahan data dan mencetak laporan untuk laporan bulanan maka admin memilih tahun dan bulan, sedangkan untuk laporan tahunan admin memilih tahun. Berikut dapat dilihat pengujian dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.18 :

Tabel 4.18 Pengujian pengolahan data laporan (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
pilih tahun	pilih [tahun..]	Untuk print laporan perharilan dan laporan barang langgsung tekan tombol PrintPdf,untuk laporan per bulan pilih bulan dan pilih tahun, hasil dari pelaporan yang akan diprint disajikan dalama format dalam bentuk pdf	[√] Diterima [ ] Ditolak
Pilih bulan	pilih [bulan..]	Untuk print laporan	

		perharilan dan laporan barang langgsung tekan tombol PrintPdf,untuk laporan per bulan pilih bulan dan pilih tahun, hasil dari pelaporan yang akan diprint disajikan dalama format dalam bentuk pdf	

Berikut dapat dilihat pengujian dari pengolahan data rekening dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.19 :

Tabel 4.19 Pengujian pengolahan data laporan (data invalid)

Kasus dan Hasil Uji (Data inValid)			
Data Masukan	invalid Class	Pengamatan	Kesimpulan
pilih tahun	kosong	Printpdf tidak bisa dilakukan jika data sebelumnya yang berhubungan denagn pelaporan kosong	[ ] Diterima [√] Ditolak
pilih bulan	kosong	Printpdf tidak bisa dilakukan jika data sebelumnya yang berhubungan denagn pelaporan kosong	[ ] Diterima [√] Ditolak

9. Pengujian Pengolahan Retur

Pengujian laporan dapat dilakukan apabila member melakukan retur pembelian.Berikut dapat dilihat pengujian dari pengolahan retur dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.20 :

Tabel 4.20 Pengujian pengolahan data retur data invalid)

Kasus dan Hasil Uji (Data Valid)

Data Masukan	Valid Class	Pengamatan	Kesimpulan
Status retur	pilih [ditolak|terima]	Admin memilih status retur di tolak atau diteriama. dan status returpun berubah	[ ] Diterima [√] Ditolak

10. Pengujian Edit Profile

Berikut dapat dilihat pengujian dari Edit Profile dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.21 :

Tabel 4.21 Pengujian edit profile (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : ridwan	[A-Z|a-z|0-9] 5-32 char	Data nama dapat diterima jika input data 1.nama 5 sampai 32 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Email contoh masukan : Email : ridwan_murinasgar@ yahoo.com	[A-Z|a-z|0-9|@|.]	Data Email dapat diterima jika input data ; 1.Email : 5 sampai 30 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	[√] Diterima [ ] Ditolak

		4 Harus mengandung simbol @ dan .	
Telephone contoh masukan : Telephone : 4415133	[0-9]	Data kode pos dapat diterima jika input data kode pos diisi dengan number	No Telp contoh masukan : No telepon : 7792310
Alamat contoh masukan : Alamat : jl bratayuda	[A-Z|a-z|0-9]	Data Alamat dapat diterima jika input data ; 1.Alamat diisi dengan char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	Alamat contoh masukan : Alamat : jl milan no 18

Berikut dapat dilihat pengujian dari dari edit profile dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.22 :

Tabel 4.22 Pengujian edit profile (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : di Nama : dsfsgdgdgdgfdgdgdgdgdgdgdgdgdfghgfjhhhgjhggjgjjg Nama :	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter.	[ ] Diterima [√] Ditolak
Email contoh masukan : Email : murin Email :	[A-Z|a-z|0-9|] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika	[ ] Diterima [√] Ditolak

		kurang dari 5 karakter dan lebih dari 32 karakter. tidak mengandung karakter @ dan .	
Telephone contoh masukan : Telephone : 23232323	[A-Z|a-z] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika mengisi dengan huruf	[ ] Diterima [√] Ditolak
Alamat contoh masukan : Alamat : Jl raya garut	data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong	[ ] Diterima [√] Ditolak

4.3.1.1.2 Equivalence Partioning untuk Pengunjung

1. Pengujian Registrasi Member

Berikut dapat dilihat pengujian dari registrasi *member* dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.23:

Tabel 4.23 Pengujian Registrasi Member (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : ridwan	[A-Z|a-z|0-9] 45 char	Data nama dapat diterima jika input data 1.nama 1 sampai 45 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. Dapat memunculkan pesan kesalahan bahwa field tersebut	[√] Diterima [ ] Ditolak

		harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	
Password contoh masukan : Password : murin1234	[A-Z|a-z|0-9] 10 char	Data Password dapat diterima jika input data ; 1.Isi password 1- 10 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	[√] Diterima [ ] Ditolak
Comfirm Password contoh masukan Comfirm Password : Murin1234	[A-Z|a-z|0-9] 10 char	Data Comfirm Password dapat diterima jika input data ; 1.Comfrim Password 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4 Harus sama dengan field Password	[√] Diterima [ ] Ditolak
Tanggal_lahir contoh masukan : Password : 2011-10-23	[A-Z|a-z|0-9] date	Data Password dapat diterima jika input data ; 1.Isi date 2.Boleh mengunakan angka	[√] Diterima [ ] Ditolak
Email contoh masukan : Email : ridwan_murinasgar@ yahoo.com	[A-Z|a-z|0-9|@|.]	Data Email dapat diterima jika input data ; 1.Email : 5 sampai 30 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4 Harus mengandung simbol	[√] Diterima [ ] Ditolak

		@ dan .	
Alamat contoh masukan : Alamat : Jln.samarang alfalah garut	[A-Z|a-z|0-9]	Data Alamat dapat diterima jika input data ; 1.Alamat diisi dengan char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	[√] Diterima [ ] Ditolak
Provinsi contoh masukan : Jawa Barat	Pilih Provinsi [1 |2|3...]	User memilih provinsi dengan checkbox yang tersedia	[√] Diterima [ ] Ditolak
Kota contoh masukan : Kota : Garut	Pilih Kota [1 |2|3...]	User memilih provinsi dengan checkbox yang tersedia dan sebelumnya harus memilih provinsi	[√] Diterima [ ] Ditolak
Kode Pos contoh masukan : Kode Pos : 44151	[0-9]	Data kode pos dapat diterima jika input data kode pos diisi dengan number saja	[√] Diterima [ ] Ditolak
No Telepon contoh masukan : No telepon : 085223272777	[0-9]	Data kode pos dapat diterima jika input data kode pos diisi dengan number	[√] Diterima [ ] Ditolak

Berikut dapat dilihat pengujian dari registrasi *member* dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.24 :

Tabel 4.24 Pengujian registrasi member (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : murin	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut	[ ] Diterima [√] Ditolak

Nama : dsfsgdgdgdgfdgdgdgdgdgdgdgdgdfghgfjhhhgjhggjgjjg Nama :		harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter.	
Password contoh masukan : Password : zz Password : zzxzxzxzxzxsdfgdgdgdgdgdgdgdgdfhgfhgfhfghgfhfhfghfgh Password :	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter.	[ ] Diterima [√]Ditolak
Comfirm Password contoh masukan : Comfirm Password : ssa Comfirm Password : asasasadsfvfdfgdhkgvdfhgndhvdgbvdddbvxngvxfhfv98867675 Comfirm Password :	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter.	[ ] Diterima [√] Ditolak
Email contoh masukan : Email : ridmu Email :	[A-Z|a-z|0-9|] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter. tidak mengandung karakter @ dan .	[ ] Diterima [√] Ditolak
Alamat contoh masukan : Alamat : jl samarang	data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong	[√] Diterima [ ] Ditolak
Provinsi contoh masukan :	data kosong	User tidak memilih provinsi dengan	[√] Diterima [ ] Ditolak

Jawa Barat		checkbox yang tersedia	
Kota contoh masukan :	data kosong	User tidak memilih provinsi dengan checkbox yang tersedia dan sebelumnya harus memilih provinsi	[ ] Diterima [√] Ditolak
Kode Pos contoh masukan : Kode Pos : 565656565	[A-Z|a-z] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika mengisi dengan huruf	[√] Diterima [ ] Ditolak
No Telepon contoh masukan : No telepon : 45454545	[A-Z|a-z] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika mengisi dengan huruf	[√] Diterima [ ] Ditolak
Masukan kode contoh masukan : Masukan kode : xxxxww	data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau tidak sesuai dengan generate key	[√] Diterima [ ] Ditolak

2. Pengujian login member

Berikut dapat dilihat pengujian dari login *member* dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.25 :

Tabel 4.25 Pengujian registrasi login (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Email contoh masukan : Email : ridwan_murinasgar@yahoo.com	[A-Z|a-z|0-9|@|.]	Data Email dapat diterima jika input data ; 1.Email : 5 sampai 30 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4 Harus mengandung simbol @ dan .	[√] Diterima [ ] Ditolak
Password contoh masukan : Password : murin12334	[A-Z|a-z|0-9] 5-32 char	Data Password dapat diterima jika input data ; 1.Nama 5 sampai 32 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	[√] Diterima [ ] Ditolak

Berikut dapat dilihat pengujian dari login member dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.26 :

Tabel 4.26 Pengujian login member (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Email contoh masukan : Email : ridwantah Email :	[A-Z|a-z|0-9|] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter. tidak mengandung karakter @ dan .	[ ] Diterima [√] Ditolak
Password contoh masukan : Password : zz Password:zzxzxzxzxz xsdfgdgdgdgdgdgdf 78656775575gjhghfct yytyytrtyryt Password :	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter.	[ ] Diterima [√] Ditolak

3. Pengujian Edit Akun

Berikut dapat dilihat pengujian dari edit akun dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.27 :

Tabel 4.27 Pengujian edit akun (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : murin	[A-Z|a-z|0-9] 5-32 char	Data nama dapat diterima jika input data 1.nama 5 sampai 32 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan	[√] Diterima [ ] Ditolak

		huruf. Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	
Alamat contoh masukan : Alamat : Jln tarogong	[A-Z|a-z|0-9]	Data Alamat dapat diterima jika input data ; 1.Alamat diisi dengan char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	[√] Diterima [ ] Ditolak
Provinsi contoh masukan : Jawa Barat	Pilih Provinsi [1 |2|3...]	User memilih provinsi dengan checkbox yang tersedia	[√] Diterima [ ] Ditolak
Kota contoh masukan : Kota : garut	Pilih Kota [1 |2|3...]	User memilih provinsi dengan checkbox yang tersedia dan sebelumnya harus memilih provinsi	[√] Diterima [ ] Ditolak
Kode Pos contoh masukan : Kode Pos : 44151	[0-9]	Data kode pos dapat diterima jika input data kode pos diisi dengan number	[√] Diterima [ ] Ditolak
No Telepon contoh masukan : No telepon : 238952	[0-9]	Data kode pos dapat diterima jika input data kode pos diisi dengan number	[√] Diterima [ ] Ditolak
Password contoh masukan : Password : murin	[A-Z|a-z|0-9] 5-32 char	Data Masukan kode dapat diterima jika input data ; 1.Nama diisi dengan char	[√] Diterima [ ] Ditolak

		2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4.Harus sama dengan password yang lama.	
Password Baru contoh masukan : Password Baru: murin jaya	[A-Z|a-z|0-9] 5-32 char	Data Masukan kode dapat diterima jika input data ; 1.Nama diisi dengan char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	[√] Diterima [ ] Ditolak
Comfirm Password contoh masukan : Comfirm Password : murinjaya	[A-Z|a-z|0-9] 5-32 char	Data Masukan kode dapat diterima jika input data ; 1.Nama diisi dengan char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4.Harus sama dengan password yang baru.	[√] Diterima [ ] Ditolak

Berikut dapat dilihat pengujian dari edit akun dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.28

Tabel 4.28 Pengujian edit akun (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : ri Nama : dsfsgdgdgdgfdgdgdgdgdgdgdgdgdfghgfjhhhgjhggjgjjg Nama :	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter.	[√] Diterima [ ] Ditolak
Email contoh masukan : Email : ridwan Email :	[A-Z|a-z|0-9|] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter. tidak mengandung karakter @ dan .	[ ] Diterima [√] Ditolak
Alamat contoh masukan : Alamat : Jln samarang	data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong	[√] Diterima [ ] Ditolak
Provinsi contoh masukan : Jawa Barat	data kosong	User tidak memilih provinsi dengan checkbox yang tersedia	[√] Diterima [ ] Ditolak
Kota contoh masukan :	data kosong	User tidak memilih provinsi dengan checkbox yang tersedia dan sebelumnya harus memilih provinsi	[ ] Diterima [√] Ditolak
Kode Pos contoh masukan : Kode Pos : 44151	[A-Z|a-z] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika mengisi dengan huruf	[√] Diterima [ ] Ditolak

No Telepon contoh masukan : No telepon : 085222222	[A-Z|a-z] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika mengisi dengan huruf	[√] Diterima [ ] Ditolak
Password Lama contoh masukan : Password : zz Password:zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt Password :	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter. data harus berbeda dengan password yang lama.	[ ] Diterima [√] Ditolak
Password Baru contoh masukan : Password Baru: zz Password Baru :zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt Password Baru :	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter. data beda dengan password yang lama.	[ ] Diterima [√] Ditolak
Comfirm Password contoh masukan : Comfirm Password : zz Comfirm Password:zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt Comfirm Password :	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter. data beda dengan password yang baru	[ ] Diterima [√] Ditolak

4. Pengujian Keranjang Belanja / *CheckOut*

Berikut dapat dilihat pengujian dari keranjang belanja / *CheckOut* dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.29 :

Tabel 4.29 Pengujian keranjang belanja / checkout (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Data Produk contoh : jaket kulit	Pilih Produk <= Stok	Data Detail dapat dipilih jika stok masih tersedia.	[√] Diterima [ ] Ditolak
Jumlah produk contoh; jumlah produk : 4 ; semetara stok 10	[0-9] <= Stok	Data Jumlah produk dapat diterima jika input data ; 1.Berupa angka 2.Boleh mengunakan angka 3.kllik tombol *Update* Dapat menampilkan perubahan data jumlah produk,sub total, total diskon dan total pembayaran yang masih belum di tambahkan ongkos kirim	[√] Diterima [ ] Ditolak

Berikut dapat dilihat pengujian dari dari keranjang belanja / *CheckOut* dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.30 :

Tabel 4.30 Pengujian keranjang belanja / checkout (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Data Produk contoh : pilih data produk : jaket kulit 0 atau habis	> Stok	Dapat memunculkan pesan kesalahan bahwa detail produk telah habis / stok kosong.	[ ] Diterima [√] Ditolak
Jumlah produk contoh;	[A-Z|a-z] > Stok	Dapat memunculkan pesan kesalahan bahwa stok tidak	[ ] Diterima [√] Ditolak

jumlah produk : 10 ; semetara stok 8		mencukupi dan data yang dimasukan invalid apabila diisi dengan huruf atau symbol	

5. Pengujian Pengiriman

Menentukan data pengiriman dilakukan setelah proses *checkout* selesai Berikut dapat dilihat pengujian dari pengiriman apabila memilih alamat lain dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.31 :

Tabel 4.31 Pengujian pengiriman (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Nama Lengkap contoh masukan : Nama : ridwan	[A-Z|a-z|0-9] 5-32 char	Data nama dapat diterima jika input data 1.nama 5 sampai 32 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Alamat contoh masukan : Alamat : Jl.samarang	[A-Z|a-z|0-9]	Data Alamat dapat diterima jika input data ; 1.Alamat diisi dengan char 2.Boleh mengunakan angka 3. Boleh kombinasi	[√] Diterima [ ] Ditolak

		antara angka dengan huruf.	
Provinsi contoh masukan : Jawa Barat	Pilih Provinsi [1 |2|3...]	User memilih provinsi dengan checkbox yang tersedia	[√] Diterima [ ] Ditolak
Kota contoh masukan : Kota : garut	Pilih Kota [1 |2|3...]	User memilih provinsi dengan checkbox yang tersedia dan sebelumnya harus memilih provinsi	[√] Diterima [ ] Ditolak
Kode Pos contoh masukan : Kode Pos : 44151	[0-9]	Data kode pos dapat diterima jika input data kode pos diisi dengan number	[√] Diterima [ ] Ditolak
Email contoh masukan : Email : ridhalestari@yahoo.com	[A-Z|a-z|0-9|@|.]	Data Email dapat diterima jika input data ; 1.Email : 5 sampai 30 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4 Harus mengandung simbol @ dan .	[√] Diterima [ ] Ditolak
No Telp contoh masukan : No telepon : 08522323232	[0-9]	Data kode pos dapat diterima jika input data kode pos diisi dengan number	[√] Diterima [ ] Ditolak

Berikut dapat dilihat pengujian dari dari pengiriman apabila memilih alamat lain dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.32 :

Tabel 4.32 Pengujian pengiriman (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	Invalid Class	Pengamatan	Kesimpulan
Nama Lengkap contoh masukan : Nama : di Nama : dsfsgdgdgdgfdgdgdgdgdgdgdgdgdfghgfjhhhgjhggjgjjg Nama :	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter.	[ ] Diterima [√] Ditolak
Alamat contoh masukan : Alamat : jl targongf	data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong	[√] Diterima [ ] Ditolak
Provinsi contoh masukan : Jawa Barat	data kosong	User tidak memilih provinsi dengan checkbox yang tersedia	[√] Diterima [ ] Ditolak
Kota contoh masukan :	data kosong	User tidak memilih provinsi dengan checkbox yang tersedia dan sebelumnya harus memilih provinsi	[√] Diterima [ ] Ditolak
Kode Pos contoh masukan : Kode Pos : 44151	[A-Z|a-z] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika mengisi dengan huruf	[√] Diterima [ ] Ditolak
Email contoh masukan : Email : murin Email :	[A-Z|a-z|0-9|] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter. tidak mengandung karakter @ dan .	[ ] Diterima [√] Ditolak

No Telepon contoh masukan : No telepon : 238952	[A-Z|a-z] data kosong	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika kosong atau salah dalam pengisian jika mengisi dengan huruf	[√] Diterima [ ] Ditolak

6. Pengujian Konfirmasi Pembayaran

Konfirmasi pembayaran dapat dilakukan setelah proses *checkout* selesai dan sudah memilih data pengiriman yang terdapat di menu history. Berikut dapat dilihat pengujian dari konfirmasi pembayaran dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.33 :

Tabel 4.33 Pengujian konfirmasi pembayaran (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
klik konfirmasi pembayaran melaui paypal	Akun paypal yang sudah terverivikasi, terisi saldo yang cukup	Dapat Terintegrasi dengan sistem pembayaran paypal	[√] Diterima [ ] Ditolak
Nama Bank contoh masukan : Nama Bank : BRI	[A-Z|a-z|0-9]	Data Nama Bank dapat diterima jika input data ; 1. Nama Bank diisi dengan char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	[√] Diterima [ ] Ditolak
No Rekening / No Pesanan contoh masukan : No Rekening :	[0-9]	User memasukan data No Rekening dan untuk yang membayar secara	[√] Diterima [ ] Ditolak

4475689879 untuk yang membayar secara tunia dan tidak mempunyai rekening bank mengunakan No Pesanan No Pesanan : 27		tunia dan tidak mempunyai rekening bank mengunakan No Pesanan data dapat di terima jika input data : 1. No Rekening atau no .Pesanan merupakan angka	
Jumlah pembayaran contoh masukan : jumlahpembayaran : 50000 sementara jumlah yang harus dibayar 50000	[0-9] >= jumlah yang harus dibayar	Data diterima jika input data jumlah pembayaran : 1.Data berupa angka 2. harus lebih besar atau sama dengan jumlah yang harus dibayarkan sesuai dengan yang tertera Pesanannya.	[√] Diterima [ ] Ditolak

Berikut dapat dilihat pengujian dari dari konfirmasi pembayaran dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.34 :

Tabel 4.34 Pengujian konfirmasi pembayaran (data invalid)

Kasus dan Hasil Uji (Data InValid)			
Data Masukan	InValid Class	Pengamatan	Kesimpulan
klik konfirmasi pembayaran melaui paypal	Akun paypal yang belum terverivikasi, saldo yang ada tidak mencukupi	Dapat Terintegrasi dengan sistem pembayaran paypal , namun tidak bisa melakukan transaksi	[ ] Diterima [√] Ditolak
Nama Bank contoh masukan : Nama Bank :	data kosong	memunculkan bahwa data nama bank kosong	[ ] Diterima [√] Ditolak
No Rekening / No Pesanan contoh masukan : No Rekening : lalallalalal untuk yang membayar secara tunia dan tidak	[A-Z | a-z] data kosong	Menampilkan data salah dan menampilkan pesan data kosong apabila data belum di isi.	[ ] Diterima [√] Ditolak

mempunyai rekening bank mengunakan No Pesanan No Pesanan :			
Jumlah pembayaran contoh masukan : jumlah pembayaran: seribu jumlah pembayaran : 23 sementara jumlah yang harus dibayar	[A-Z | a-z] <= jumlah yang harus dibayar	Data diterima jika input data jumlah pembayaran : 1.Data berupa angka 2. harus lebih besar atau sama dengan jumlah yang harus dibayarkan sesuai dengan yang tertera Pesanannya.	[ ] Diterima [√] Ditolak

7. Pengujian Retur Pemesanan

Retur pemesanan dapat dilakukan setelah barang diterima. Berikut dapat dilihat pengujian dari retur pemesanan dengan kasus dan hasil uji (*data valid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.35 :

Tabel 4.35 Pengujian retur (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	Valid Class	Pengamatan	Kesimpulan
Memilih jumlah yang akan di retur contoh ; jumlah : 2	[0-9]	Memilih jumlah barang yang akan direturkan sesuai dengan data Pesanannya	[√] Diterima [ ] Ditolak
Keterangan contoh : lengan jaket sobek	Pilih [alasan retur]	Data Keterangan dapat diterima jika input data ; memilih alasan retur yang telah di sediakan hasil dari keterangan ini akan manjadi pertimbangan admin untuk menindaklanjuti	[√] Diterima [ ] Ditolak

		proses retur selanjutnya.	

Berikut dapat dilihat pengujian dari dari retur pemesanan dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.36 :

Tabel 4.36 Pengujian retur (data invalid)

Kasus dan Hasil Uji (Data InValid)			
Data Masukan	InValid Class	Pengamatan	Kesimpulan
Memilih jumlah yang akan di retur contoh ; jumlah : we jumlah :	[A-Z|a-z|] data kosong	Memunculkan peasan kesalahan atau barang dianggap tidak di returkan	[ ] Diterima [√] Ditolak
Keterangan contoh ; Keterangan :	kosong	Memunculkan peasan kesalahan data kosong atau barang dianggap tidak di returkan	[ ] Diterima [√] Ditolak

4.3.1.2 Cause Effect Testing

4.3.1.2.1 Cause Effect Testing Untuk Admin

Teknik ini merupakan suplemen dari untuk *equivalence testing* menyediakan cara untuk memilih kombinasi data input, melibatkan kodisi input *(Cause)* dan kodisi output *(effect)* untuk mencegah pendefinisian kasus uji yang terlalu banyak. Berikut ini *cause effect testing* untuk Admin ;

1. *Cause effect testing* Login Admin

***Input states* :**

1. Username memenuhi kondisi yang telah ditentukan

2. Password memenuhi kondisi yang telah ditentukan

3. Type memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Login berhasil

B. Menampilkan pesan kesalahan dari username yang diinputkan

C. Menampilkan pesan kesalahan dari password yang diinputkan

D. Menampilkan pesan kesalahan dari type yang diinputkan atau dipilih

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari login admin. Tabel *cause effect testing* untuk login admin dapat dilihat pada tabel 4.37

Tabel 4.37 *Cause effect testing* untuk login admin

Test case / *Cause effect*	1	2	3	4
1	**1**	**0**	**1**	**1**
2	**1**	**1**	**0**	**1**
3	**1**	**1**	**1**	**0**
A	**1**	**0**	**0**	**0**
B	**0**	**1**	**0**	**0**
C	**0**	**0**	**1**	**0**
D	**0**	**0**	**0**	**1**

***2. Cause effect testing* pengolahan data produk**

***Input states* :**

1. Kategori memenuhi kondisi yang telah ditentukan

2. Produk memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari kategori yang diinputkan atau dipilih

C. Menampilkan pesan kesalahan dari produk yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari pengolahan data produk. Tabel *cause effect testing* untuk pengolahan data produk dapat dilihat pada tabel 4.38 ;

Tabel 4.38 *Cause effect testing* pengolahan data produk

Test case / *Cause effect*	1	2	3	4
1	**1**	**1**	**0**	**0**
2	**1**	**0**	**1**	**0**
A	**1**	**0**	**0**	**0**
B	**0**	**1**	**0**	**1**
C	**0**	**0**	**1**	**1**

3. *Cause effect testing* pengolahan data produk

***Input states* :**

1. Nama produk memenuhi kondisi yang telah ditentukan

2. Nama manufaktur memenuhi kondisi yang telah ditentukan

3.Harga memenuhi kondisi yang telah ditentukan

4.Stok memenuhi kondisi yang telah ditentukan

5.Berat memenuhi kondisi yang telah ditentukan

6.Diskon memenuhi kondisi yang telah ditentukan

7.Deskripsi memenuhi kondisi yang telah ditentukan

8.Gambar Produk memenuhi kondisi yang telah ditentukan

***Output states* :**

a. Data disimpan kedalam database
b. Menampilkan pesan kesalahan dari nama produk yang diinputkan atau dipilih
c. Menampilkan pesan kesalahan dari manufaktur yang diinputkan atau dipilih
d. Menampilkan pesan kesalahan dari harga yang diinputkan
e. Menampilkan pesan kesalahan dari stok yang diinputkan
f. Menampilkan pesan kesalahan dari berat yang diinputkan
g. Menampilkan pesan kesalahan dari diskon yang diinputkan
h. Menampilkan pesan kesalahan dari deskripsi yang diinputkan
i. Menampilkan pesan kesalahan dari gambar produk yang diinputkan atau dipilih

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk

mempermudah membaca hasil dari uji atau testing dari pengolahan data detail produk. Tabel *cause effect testing* untuk pengolahan data detail produk dapat dilihat pada tabel 4.39

Tabel 4.39 *Cause effect testing* pengolahan data detail produk

Test case / *Cause effect*	1	2	3	4	5	6	7	8	9
1	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
2	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**
3	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**
4	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**
5	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**
6	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**
7	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**
8	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**
A	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
B	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
C	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**
D	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**
E	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**
F	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**
G	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**
H	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**
I	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**

4. *Cause effect testing* pengolahan data kota

***Input states* :**

1. Provinsi memenuhi kondisi yang telah ditentukan

2. Nama kota memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari nama provinsi yang diinputkan atau dipilih

C. Menampilkan pesan kesalahan dari nama kota yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari pengolahan data kota. Tabel *cause effect testing* untuk pengolahan data kota dapat dilihat pada tabel 4.40

Tabel 4.40 *Cause effect testing* untuk pengolahan data kota

Test case / *Cause effect*	1	2	3	4
1	**1**	**1**	**0**	**0**
2	**1**	**0**	**1**	**0**
A	**1**	**0**	**0**	**0**
B	**0**	**1**	**0**	**1**
C	**0**	**0**	**1**	**1**

5. *Cause Effect Testing* pengolahan data rekening

***Input states* :**

1. Nama bank memenuhi kondisi yang telah ditentukan

2. Nama pemilik memenuhi kondisi yang telah ditentukan

3.No rekening memenuhi kondisi yang telah ditentukan

4. Cabang memenuhi kondisi yang telah ditentukan

5. Gambar memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B.Menampilkan pesan kesalahan dari nama bank yang diinputkan

C.Menampilkan pesan kesalahan dari nama pemilik yang diinputkan

D.Menampilkan pesan kesalahan dari rekening yang diinputkan

E.Menampilkan pesan kesalahan dari cabang yang diinputkan

F. Menampilkan pesan kesalahan dari gambar yang diinputkan atau dipilih

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari pengolahan data rekening. Tabel *cause effect testing* untuk pengolahan data rekening dapat dilihat pada tabel 4.41

Tabel 4.41 *Cause effect testing* untuk pengolahan data rekening

Test case / *Cause effect*	1	2	3	4	5	6
1	**1**	**0**	**1**	**1**	**1**	**1**
2	**1**	**1**	**0**	**1**	**1**	**1**
3	**1**	**1**	**1**	**0**	**1**	**1**
4	**1**	**1**	**1**	**1**	**0**	**1**
5	**1**	**1**	**1**	**1**	**1**	**0**
A	**1**	**0**	**0**	**0**	**0**	**0**
B	**0**	**1**	**0**	**0**	**0**	**0**
C	**0**	**0**	**1**	**0**	**0**	**0**
D	**0**	**0**	**0**	**1**	**0**	**0**
E	**0**	**0**	**0**	**0**	**1**	**0**

F	**0**	**0**	**0**	**0**	**0**	**1**

7. *Cause Effect Testing* edit profil

***Input states* :**

1. Nama memenuhi kondisi yang telah ditentukan

2. Email memenuhi kondisi yang telah ditentukan

3. Telephone memenuhi kondisi yang telah ditentukan

4. Alamat memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B.Menampilkan pesan kesalahan dari nama yang diinputkan

C.Menampilkan pesan kesalahan dari email yang diinputkan

D.Menampilkan pesan kesalahan dari telephone yang diinputkan

E.Menampilkan pesan kesalahan dari alamat yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari edit profil. Tabel *cause effect testing* untuk edit profil dapat dilihat pada tabel 4.42

Tabel 4.42 *Cause effect testing* untuk edit profil

Test case / *Cause effect*	1	2	3	4	5

1	**1**	**0**	**1**	**1**	**1**
2	**1**	**1**	**0**	**1**	**1**
3	**1**	**1**	**1**	**0**	**1**
4	**1**	**1**	**1**	**1**	**0**
A	**1**	**0**	**0**	**0**	**0**
B	**0**	**1**	**0**	**0**	**0**
C	**0**	**0**	**1**	**0**	**0**
D	**0**	**0**	**0**	**1**	**0**
E	**0**	**0**	**0**	**0**	**1**

8. *Cause Effect Testing* edit password

***Input states* :**

1. Password lama memenuhi kondisi yang telah ditentukan

2. Password baru memenuhi kondisi yang telah ditentukan

3. Confirm password memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B.Menampilkan pesan kesalahan dari password lama yang diinputkan

C.Menampilkan pesan kesalahan dari password baru yang diinputkan

D.Menampilkan pesan kesalahan dari confirm yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari edit password. Tabel *cause effect testing* untuk edi password dapat dilihat pada tabel 4.43 :

Tabel 4.43 *Cause effect testing* untuk edit password

Test case / *Cause effect*	1	2	3	4
1	**1**	**0**	**1**	**1**
2	**1**	**1**	**0**	**1**
3	**1**	**1**	**1**	**0**
A	**1**	**0**	**0**	**0**
B	**0**	**1**	**0**	**0**
C	**0**	**0**	**1**	**0**
D	**0**	**0**	**0**	**1**

4.3.1.2.2 Cause Effect Testing Untuk Pengunjung dan Member

Teknik ini merupakan suplemen dari untuk *equivalence testing* menyediakan cara untuk memilih kombinasi data input, melibatkan kodisi input *(Cause)* dan kodisi output *(effect)* untuk mencegah pendefinisian kasus uji yang terlalu banyak. Adapun langkah-langkahnya membagi spesifikasi kebutuhan menjadi bagian yang memiliki kemungkinan kerja kemudian definisikan *cause effect* berdasarkan spesifikasi kebutuhan dan membuat tabel keputusan, kolom untuk *test case* dan baris untuk *cause/effect*, symbol 1 untuk mewakili ya symbol 0 untuk mewakili tidak. Berikut ini *cause effect testing* untuk pengunjung ;

1. *Cause effect testing* registrasi member

***Input states* :**

1. Nama memenuhi kondisi yang telah ditentukan
2. Password memenuhi kondisi yang telah ditentukan
3. Comfirm Password memenuhi kondisi yang telah ditentukan
4. Email memenuhi kondisi yang telah ditentukan

5.Alamat memenuhi kondisi yang telah ditentukan

6. Tanggal lahir memenuhi kondisi yang telah ditentukan

7. Provinsi memenuhi kondisi yang telah ditentukan

8. Kota memenuhi kondisi yang telah ditentukan

9. Kode pos memenuhi kondisi yang telah ditentukan

10. No.Telepon memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari nama yang diinputkan

C. Menampilkan pesan kesalahan dari password yang diinputkan

D.Menampilkan pesan kesalahan dari comfirm password yang diinputkan

E.Menampilkan pesan kesalahan dari email yang diinputkan

F. Menampilkan pesan kesalahan dari alamat yang diinputkan

G. Menampilkan pesan kesalahan dari tanggal lahir yang diinputkan

H. Menampilkan pesan kesalahan dari provinsi yang diinputkan

I. Menampilkan pesan kesalahan dari kota yang diinputkan

J. Menampilkan pesan kesalahan dari kode pos yang diinputkan

K. Menampilkan pesan kesalahan dari no.telepon yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari registrasi member. Tabel *cause effect testing* untuk registrasi member dapat dilihat pada tabel 4.44

Tabel 4.44 *Cause effect* testing registrasi member

Test case / *Cause effect*	1	2	3	4	5	6	7	8	9	10	11	12
1	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
2	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
3	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
4	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
5	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**
6	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**
7	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**
8	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**
9	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**
10	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**
11	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**
A	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
B	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
C	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
D	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
E	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
F	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**
G	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**
H	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**
I	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**
J	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**
K	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**

2. *Cause effect testing* login member

***Input states* :**

1. Email memenuhi kondisi yang telah ditentukan

2. Password memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Login berhasil

B. Menampilkan pesan kesalahan dari email yang diinputkan

C. Menampilkan pesan kesalahan dari password yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari login member. Tabel *cause effect testing* untuk login member dapat dilihat pada tabel 4.45

Tabel 4.45 Cause effect testing login member

Test case / *Cause effect*	1	2	3	4
1	**1**	**0**	**1**	**0**
2	**1**	**1**	**0**	**0**
A	**1**	**0**	**0**	**0**
B	**0**	**1**	**0**	**1**
C	**0**	**0**	**1**	**1**

Keterangan dari tabel *cause effect testing* login member adalah menghasilkan kasus uji sebagai beikut;

Jika member memasukan email benar dan password benar maka login berhasil.

Jika member memasukan email salah dan password benar maka sistem akan menampilkan pesan kesalahan dari email yang diinputkan

Jika member memasukan email salah dan password benar maka sistem akan menampilkan pesan kesalahan dari password yang diinputkan.

Jika member memasukan email salah dan password salah maka sistem akan menampilkan pesan kesalahan dari email yang diinputkan dan menampilkan pesan kesalahan dari password yang diinputkan.

3. *Cause effect testing* edit akun Member

***Input states* :**

1. Nama memenuhi kondisi yang telah ditentukan
2. Alamat memenuhi kondisi yang telah ditentukan
3. Provinsi memenuhi kondisi yang telah ditentukan
4, Kota memenuhi kondisi yang telah ditentukan
5. Kode Pos memenuhi kondisi yang telah ditentukan
6. No Telepon memenuhi kondisi yang telah ditentukan
7. Password Lama memenuhi kondisi yang telah ditentukan
8. Password Baru memenuhi kondisi yang telah ditentukan
9. Comfirm Password Baru memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari password lama yang diinputkan

C. Menampilkan pesan kesalahan password baru dengan comfirm password baru berbeda

D. Menampilkan pesan kesalahan dari password baru yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari edit akun member. Tabel *cause effect testing* untuk edit akun member dapat dilihat pada tabel 4.39:

Tabel 4.46 *Cause effect* testing edit akun member

Test case / *Cause effect*	1	2	3	4	5	6	7	8	9	10
1	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
2	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
3	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**
4	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**
5	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**
6	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**
7	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**
8	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**
9	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**
A	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**0**	**0**
B	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**
C	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**
D	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**

4. *Cause effect testing checkout*

***Input states* :**

1. Data produk memenuhi kondisi yang telah ditentukan

2. Jumlah produk memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari jumlah produk yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari *checkout*. Tabel *cause effect testing* untuk *checkout* dapat dilihat pada tabel 4.47

Tabel 4.47 *Cause effect testing checkout*

Test case / *Cause effect*	**1**	**2**
1	1	1
2	1	0
A	1	0
B	0	1

5. *Cause effect testing* Pengiriman alamat lain

***Input states* :**

1. Nama lengkap memenuhi kondisi yang telah ditentukan

2. Alamat memenuhi kondisi yang telah ditentukan

3. Provinsi memenuhi kondisi yang telah ditentukan

4. Kota memenuhi kondisi yang telah ditentukan

5. Kode pos memenuhi kondisi yang telah ditentukan

6. Email memenuhi kondisi yang telah ditentukan

7 No.Telepon memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari nama lengkap yang diinputkan

C. Menampilkan pesan kesalahan dari alamat yang diinputkan

D. Menampilkan pesan kesalahan dari provinsi yang diinputkan

E. Menampilkan pesan kesalahan dari kota yang diinputkan

F. Menampilkan pesan kesalahan dari kode pos yang diinputkan

G. Menampilkan pesan kesalahan dari email yang diinputkan

H. Menampilkan pesan kesalahan dari no.telepon yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari pengiriman alamat lain. Tabel *cause effect testing* untuk pengiriman alamat lain dapat dilihat pada tabel 4.48

Tabel 4.48 *Cause effect testing* pengiriman alamat lain

Test case / *Cause effect*	1	2	3	4	5	6	7	8
1	**1**	**0**	**1**	**1**	**1**	**1**	**1**	**1**
2	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**1**
3	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**
4	**1**	**1**	**1**	**1**	**0**	**1**	**1**	**1**
5	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**1**
6	**1**	**1**	**1**	**1**	**1**	**1**	**0**	**1**
7	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**0**
A	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
B	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**
C	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**
D	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**

E	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**
F	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**
G	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**
H	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**

6. *Cause effect testing* konfirmasi pembayaran melalui bank

***Input states* :**

1. Nama bank memenuhi kondisi yang telah ditentukan

2. No Rekening / No Pesanan memenuhi kondisi yang telah ditentukan

3. Jumlah pembayaran memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari nama bank yang diinputkan

C. Menampilkan pesan kesalahan dari No Rekening / No Pesanan yang diinputkan

D. Menampilkan pesan kesalahan dari jumlah pembayaran yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari konfirmasi pembayaran melalui bank. Tabel *cause effect testing* untuk konfirmasi pembayaran melalui bank dapat dilihat pada tabel 4.49

Tabel 4.49 *Cause effect testing* konfirmasi pembayaran melalui bank

Test case / *Cause effect*	1	2	3	4

1	**1**	**0**	**1**	**1**
2	**1**	**`1**	**0**	**1**
3	**1**	**1**	**1**	**0**
A	**1**	**0**	**0**	**0**
B	**0**	**1**	**0**	**0**
C	**0**	**0**	**1**	**0**
D	**0**	**0**	**0**	**1**

7. *Cause effect testing* retur

***Input states* :**

1. Jumlah produk yang akan diretur memenuhi kondisi yang telah ditentukan

2. Pilih Keterangan memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari Jumlah produk yang akan diretur bank yang diinputkan

C. Menampilkan pesan kesalahan dari Pilih Keterangan yang akan diretur bank yang diinputkan

Dari *input states* dan *output states* yang didapatkan dan didefinisikan sebelumnya maka dapat ditampilkan tabel *cause effect testing* untuk mempermudah membaca hasil dari uji atau testing dari retur pemesanan. Tabel *cause effect testing* untuk retur dapat dilihat pada tabel 4.43

Tabel 4.50 *Cause effect testing* untuk retur

Test case / ***Cause effect***

	1	2	3	4
1	**1**	**1**	**0**	**0**
2	**1**	**0**	**1**	**0**
A	**1**	**0**	**0**	**0**
B	**0**	**1**	**0**	**1**
C	**0**	**0**	**1**	**1**

4.3.1.2.3 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.

4.3.2 Pengujian Betha

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan membuat kuesioner mengenai kepuasan user dengan kandungan poin syarat *user friendly* untuk selanjutnya dibagikan kepada sebagian user dengan mengambil sample sebanyak 5 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

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

1. Lingkungan Penjual

Penelitian di lingkungan penjual dalam hali ini super admin dan admin 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*,dalam hal ini pengunjung dan member dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat memberikan kemudahan masyarakat umum untuk melakukan transaksi pembelian barang secara *online*.

4.3.2.1 Kuisioner Pengguna

4.3.2.1.1 Kuesioner untuk admin

Kuesioner disebarkan menggunakan teknik sampling yaitu *Simple Random Sampling* yang disebarkan kepada 4 pengguna. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner ini terdiri dari 5 pertanyaan (contoh kuesioner dapat diliihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 5, berikut adalah hasil kuesioner :

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

Keterangan : P = Banyaknya jawaban responden tiap soal

Q = Jumlah Responden

Y = Nilai Prosentase

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 2 orang.

1. Bagaiman tampilan perangkat lunak untuk pengolahan data?

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Bagus	0	0
	2	Bagus	2	100%
	3	Cukup Bagus	0	0
	4	Kurang Bagus	0	0
	5	Tidak Bagus	0	0
Jumlah			**2**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 100% mengatakan bahwa perangkat lunak yang dibuat . Sehingga secara keseluruhan tampilan perangkat lunak tersebut bagus.

2. Apakah anda mengerti dengan intruksi yang terdapat dalam *website* tersebut mudah dimengerti?

Pertanyaan	No.	Keterangan	Responden	Prosentase
2	1	Sangat Mudah Digunakan	0	0
	2	Mudah Digunakan	1	50%
	3	Cukup Mudah Digunakan	1	50%
	4	Cukup Sulit Digunakan	0	0
	5	Sulit Digunakan	0	0
Jumlah			**2**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 1 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat mudah digunakan, 1 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat cukup mudah digunakan. Sehinggan keseluruhan struktur perangkat lunak tersebut cukup mudah digunakan.

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

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

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 1 orang 50% mengatakan bahwa perangkat lunak yang dibuat membantu dan 1 orang 50% mengatakan cukup membantu. Sehingga secara keseluruhan perangkan lunak tersebut cukup membantu.

4. Apakah *website* ini membantu dalam pengolahan data penjualan secara online?

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

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang 100% mengatakan bahwa perangkat lunak yang dibuat cukup membantu. Sehingga secara keseluruhan perangkan lunak tersebut cukup membantu.

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

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)

5	1	Sangat Membantu	0	0
	2	Membantu	1	50%
	3	Cukup Membantu	1	50%
	4	Kurang Membantu	0	0
	5	Tidak Membantu	0	0
Jumlah			**2**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 1 orang 50% mengatakan bahwa perangkat lunak yang dibuat membantu dan 1 orang 50% mengatakan cukup membantu. Sehingga secara keseluruhan perangkan lunak tersebut cukup membantu.

4.3.2.1.2 Kuesioner untuk Pengguna atau Pengunjung Website

1. **Apakah Anda setuju bahwa tampilan *website* PD Ridha Lestari cukup menarik dan mudah dipahami / *user friendly*?**

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	3	15%
	2	Setuju	10	50%
	3	Cukup Setuju	5	25%
	4	Kurang Setuju	2	10%
	5	Tidak Setuju	0	0
Jumlah			**20**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 10 orang atau 50% mengatakan bahwa setuju bahwa tampilan website *user friendly,* . Sehingga secara keseluruhan website tersebut setuju tampilannya menarik *user friendly* .

2. Apakah Anda setuju bahwa *website* E-commerce ini dapat membantu Anda untuk mendapatkan informasi terbaru tentang produk yang ada di PD Ridha Lestari?

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	2	10%
	2	Setuju	8	40%
	3	Cukup Setuju	10	50%
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			**20**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 10% mengatakan bahasanya sangat setuju, sementara 8 orang atau 40% mengatakan setuju, 10 orang atau 50% mengatakan cukup setuju. Sehingga secara keseluruhan website tersebut cukup setuju membantu *user friendly*.

3. Apakah Anda setuju bahwa *website* ini akan lebih mempermudah Anda dalam melakukan transaksi pembelian di bandingkan transaksi secara manual?

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	3	15%
	2	Setuju	7	35%
	3	Cukup Setuju	10	50%
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			**20**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 3 orang atau 15% mengatakan sangat setuju bahwa website ini mempermudah melakukan pembelanjaan produk ,sementara 7 orang atau 35%

setuju bahwa website ini mempermudah melakukan pembelanjaan produk, 10 orang atau 50% mengatakan cukup setuju. Sehingga secara keseluruhan website ini cukup setuju mempermudah melakukan pembelanjaan produk yang dibutuhkan.

4. Apakah Anda setuju dengan adanya *website* ini toko dapat memperluas pangsa pasarnya?

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
4	1	Sangat Setuju	5	25%
	2	Setuju	4	20%
	3	Cukup Setuju	11	55%
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			**20**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 5 orang atau 25% mengatakan sangat setuju bahwa website ini sesuai informasi mengenai produk yang diberikan, sementara 4 orang atau 20% mengatakan setuju bahwa website ini sesuai informasi mengenai produk yang diberikan dan 11 orang mengatakan cukup setuju. Sehingga secara keseluruhan website ini cukup setuju mempermudah melakukan pembelanjaan produk yang dibutuhkan.

5. Apakah anda setuju bahwa sistem pembayaran yang digunakan akan lebih memudahkan anda?

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
5	1	Sangat Setuju	3	15%
	2	Setuju	14	70%
	3	Cukup Setuju	3	15%
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			**20**	**100%**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 3 orang atau 15% mengatakan Sangat setuju bahwa system pembayaran yang digunakan mempermudah transaksi, 14 orang atau 70% mengatakan setuju bahwa system pembayaran yang digunakan mempermudah transaksi, 3 orang atau 15% mengatakan cukup setuju bahwa system pembayaran yang digunakan mempermudah transaksi. Sehingga secara keseluruhan Setuju website ini memberikan informasi proses pemesanan yang sesuai dengan pesanan yang dilakukan.

4.3.2.1.3 Kesimpulan Hasil Pengujian Betha

Berdasarkan hasil prosentase diatas yang didapatkan dari pengujian betha pengguna admin, maka sistem penerapan *e-commerce* untuk sistem penjualan ini sudah sesuai dengan tujuan yaitu PD Ridha Lestari dapat mepermudah dalam proses promosi dan penjualan produk yang ditawarkan, pelanggan dapat melakukan transaksi pembelian secara online, tampilannya *user friendly,* memudahkan dalam pengelolaan data barang dan memudahkan dalam membuat laporan barang dan laporan penjualan dari sistem *e-commerce* ini.