BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Tahap implementasi sistem merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan menerjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dapat dimengerti oleh mesin serta penerapan perangkat lunak pada keadaan yang sesungguhnya.

4.1.1 Perangkat Keras yang Digunakan

Spesifikasi perangkat keras (*hardware*) yang digunakan dalam pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut:

a. *Processor* AMD athlon X2 240 2.8 GHz
b. Kapasitas RAM 2 GB
c. Kapasitas *Harddisk* 250 GB
d. 15.6” LCD
e. *Mouse*
f. *Keyboard*
g. Modem Speddy kecepatan akses 384 kbps
h. Printer Canon IP 1700

4.1.2 Perangkat Lunak yang Digunakan

Spesifikasi perangkat lunak (*Software*) yang dipasang pada sistem komputer yang digunakan untuk mengembangkan sistem situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut :

a. Sistem operasi *Windows* XP SP3 32-bit

b. Web browser *Mozila firefox* 3.6

c. *Adobe Dreamweaver C4* sebagai pembangun aplikasinya .

d. *XAMPP* sebagai *Web servernya* dan untuk mengaksesnya diperlukan *web browser*.

4.2 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 knalpot sepeda motor pada KP16 adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `admin` (
  `idadm` smallint(5) NOT NULL AUTO_INCREMENT,
  `admuser` varchar(50) NOT NULL,
  `password` varchar(100) NOT NULL,
  `status` enum('3','2','1','0') NOT NULL DEFAULT '0',
  PRIMARY KEY (`idadm`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

2. Tabel Matauang

Struktur tabel kurs pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `matauang` (
  `idkurs` int(11) NOT NULL AUTO_INCREMENT,
  `namakurs` varchar(15) NOT NULL,
  `kurs` varchar(20) NOT NULL,
  `tanggal` date NOT NULL,
```

```
  `idadm` smallint(5) NOT NULL,
  PRIMARY KEY (`idkurs`),
  KEY `idadm` (`idadm`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
ALTER TABLE `matauang`
  ADD   CONSTRAINT   `matauang_ibfk_1`   FOREIGN   KEY   (`idadm`)
REFERENCES `admin` (`idadm`);
```

3. Tabel Kategori Merk

Struktur tabel detail produk pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `kategori_merk` (
  `id_jenis` int(11) NOT NULL AUTO_INCREMENT,
  `catid` int(11) NOT NULL,
  `id_merk` int(11) NOT NULL,
  PRIMARY KEY (`id_jenis`),
  KEY `id_merk` (`id_merk`),
  KEY `catid` (`catid`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
ALTER TABLE `kategori_merk`
  ADD  CONSTRAINT  `kategori_merk_ibfk_3`  FOREIGN  KEY  (`catid`)
REFERENCES `kategori` (`catid`) ON UPDATE CASCADE,
  ADD  CONSTRAINT  `kategori_merk_ibfk_2`  FOREIGN  KEY  (`id_merk`)
REFERENCES `merk` (`idmerk`);
```

4. Tabel Detail Pesanan

Struktur tabel gambar pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `detail_pesanan` (
  `id_detail` int(10) NOT NULL,
  `idpro` int(10) NOT NULL,
  `no_transaksi` int(10) NOT NULL,
  `qty` smallint(5) NOT NULL,
  `harga` double NOT NULL,
  `idkurs` int(11) NOT NULL,
  `idadm` smallint(5) NOT NULL,
  PRIMARY KEY (`id_detail`),
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `detail_pesanan`
  ADD  CONSTRAINT  `detail_pesanan_ibfk_3`  FOREIGN  KEY  (`idkurs`)
REFERENCES `matauang` (`idkurs`),
  ADD  CONSTRAINT  `detail_pesanan_ibfk_4`  FOREIGN  KEY  (`idadm`)
REFERENCES `admin` (`idadm`),
  ADD  CONSTRAINT  `detail_pesanan_ibfk_5`  FOREIGN  KEY  (`idpro`)
REFERENCES  `produk`  (`idpro`)  ON  DELETE  CASCADE  ON  UPDATE
CASCADE,
  ADD    CONSTRAINT    `detail_pesanan_ibfk_6`    FOREIGN    KEY
(`no_transaksi`) REFERENCES `pesanan` (`no_transaksi`) ON DELETE
CASCADE ON UPDATE CASCADE;
```

5. Tabel Retur

Struktur tabel kurs pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `detail_retur` (
  `id_det_ret` int(11) NOT NULL AUTO_INCREMENT,
  `id_retur` int(3) NOT NULL,
  `no_transaksi` varchar(100) NOT NULL,
  PRIMARY KEY (`id_det_ret`)
) ENGINE= InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
ALTER TABLE `retur`
  ADD CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`no_transaksi`)
REFERENCES `pesanan` (`no_transaksi`);
```

6. Tabel Jasa Pengiriman

Struktur tabel kategori pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `jasa_pengiriman` (
  `kdjasa` smallint(5) NOT NULL AUTO_INCREMENT,
  `nama_jasa` varchar(50) NOT NULL,
  PRIMARY KEY (`kdjasa`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

7. Tabel Jenis Pengiriman

Struktur tabel kategori pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `jenis_pengiriman` (
  `kdjenis` smallint(5) NOT NULL AUTO_INCREMENT,
  `kdjasa` smallint(5) NOT NULL,
  `nama_jenis` varchar(50) NOT NULL,
  `ongkos` double NOT NULL,
  PRIMARY KEY (`kdjenis`),
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `jenis_pengiriman`
  ADD CONSTRAINT `jenis_pengiriman_ibfk_1` FOREIGN KEY (`kdjasa`)
REFERENCES `jenis_pengiriman` (`kdjasa`) ON DELETE CASCADE ON
UPDATE CASCADE;
```

8. Tabel Kategori

Struktur tabel kategori pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `catid` int(10) NOT NULL AUTO_INCREMENT,
  `catname` varchar(25) NOT NULL,
  `idpro` int(10) NOT NULL,
```

```
  PRIMARY KEY (`catid`),
  UNIQUE KEY `catname` (`catname`)
) ENGINE= InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

9. Tabel Kota

Struktur tabel member pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `kota` (
  `kdkota` smallint(5) NOT NULL AUTO_INCREMENT,
  `kdpro` smallint(5) NOT NULL,
  `nama_kota` varchar(50) NOT NULL,
  `kabkota` varchar(50) NOT NULL,
  PRIMARY KEY (`kdkota`),
  UNIQUE KEY `nama_kota` (`nama_kota`,`kabkota`),
  KEY `kdpro` (`kdpro`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=425;
ALTER TABLE `kota`
  ADD CONSTRAINT `kota_ibfk_1` FOREIGN KEY (`kdpro`) REFERENCES
`propinsi` (`kdpro`) ON UPDATE CASCADE;
```

10. Tabel Member

Struktur tabel ongkos kirim pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `member` (
  `iduser` int(10) NOT NULL AUTO_INCREMENT,
  `username` varchar(30) NOT NULL,
  `password` varchar(50) NOT NULL,
  `email` varchar(75) NOT NULL,
  `jk` enum('P','W') NOT NULL,
  `kdkota` smallint(5) NOT NULL,
  `alamat` varchar(100) NOT NULL,
  `kodepos` int(7) NOT NULL,
  `telepon` varchar(15) NOT NULL,
  `kodeaktif` varchar(50) NOT NULL,
  `status` enum('1','0') NOT NULL,
  PRIMARY KEY (`iduser`),
  KEY `kdkota` (`kdkota`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
ALTER TABLE `member`
  ADD CONSTRAINT `member_ibfk_1` FOREIGN KEY (`kdkota`) REFERENCES
`kota` (`kdkota`);
```

11. Tabel Merk

Struktur tabel produk pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `merk` (
  `idmerk` int(10) NOT NULL AUTO_INCREMENT,
  `nama_merk` varchar(50) NOT NULL,
  `image` varchar(100) NOT NULL,
  PRIMARY KEY (`idmerk`),
  UNIQUE KEY `nama_merk` (`nama_merk`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=9;
```

12. Tabel Ongkos Kirim

Struktur tabel provinsi pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `ongkos_kirim` (
  `id_ongkir` int(3) NOT NULL AUTO_INCREMENT,
  `ongkir` int(9) NOT NULL,
  `kdjenis` smallint(6) NOT NULL,
  `kdjasa` smallint(6) NOT NULL,
  `kdkota` smallint(5) NOT NULL,
  PRIMARY KEY (`id_ongkir`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 ;
ALTER TABLE `ongkos_kirim`
  ADD CONSTRAINT `ongkos_kirim_ibfk_4` FOREIGN KEY (`kdjasa`)
REFERENCES `jasa_pengiriman` (`kdjasa`) ON DELETE CASCADE,
  ADD CONSTRAINT `ongkos_kirim_ibfk_1` FOREIGN KEY (`kdjenis`)
REFERENCES `jenis_pengiriman` (`kdjenis`) ON DELETE CASCADE ON
UPDATE CASCADE,
  ADD CONSTRAINT `ongkos_kirim_ibfk_3` FOREIGN KEY (`kdkota`)
REFERENCES `kota` (`kdkota`) ON DELETE CASCADE ON UPDATE CASCADE;
```

13. Tabel Pesanan

Struktur tabel rekening pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `pesanan` (
  `no_transaksi` int(10) NOT NULL AUTO_INCREMENT,
  `id_user` int(10) NOT NULL,
  `nama_t` varchar(30) NOT NULL,
  `kdkota` smallint(5) NOT NULL,
  `alamat_t` varchar(100) NOT NULL,
  `kodepos` int(10) NOT NULL,
  `telepon` varchar(15) NOT NULL,
  `total_bayar` double NOT NULL,
  `konfirmasi` enum('1','0') NOT NULL DEFAULT '0',
  `atas_nama` varchar(100) NOT NULL,
  `jml_byr` double NOT NULL,
```

```
  `id_ongkir` int(3) NOT NULL,
  `status` varchar(20) NOT NULL DEFAULT 'belum bayar',
  `tanggal` date NOT NULL,
  `paypal_transaksi` varchar(100) NOT NULL,
  `waktu_bayar` datetime NOT NULL,
  `paypal_email` varchar(100) NOT NULL,
  `bayar_paypal` double NOT NULL,
  `tanggal_kirim` datetime NOT NULL,
  PRIMARY KEY (`no_transaksi`),
  KEY `iduser` (`iduser`),
  KEY `kdkota` (`kdkota`),
  KEY `id_ongkir` (`id_ongkir`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `pesanan`
  ADD CONSTRAINT `pesanan_ibfk_1` FOREIGN KEY (`iduser`)
REFERENCES `member` (`iduser`),
  ADD CONSTRAINT `pesanan_ibfk_2` FOREIGN KEY (`kdkota`)
REFERENCES `kota` (`kdkota`),
  ADD CONSTRAINT `pesanan_ibfk_3` FOREIGN KEY (`id_ongkir`)
REFERENCES `ongkos_kirim` (`id_ongkir`);
```

14. Tabel produk

Struktur tabel retur pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `produk` (
  `idpro` int(10) NOT NULL AUTO_INCREMENT,
  `idmerk` int(10) NOT NULL,
  `catid` int(10) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `harga` int(11) NOT NULL,
  `diskon` int(3) NOT NULL,
  `description` varchar(100) NOT NULL,
  `berat` int(11) NOT NULL,
  `stock` int(5) NOT NULL,
  `image` varchar(100) NOT NULL,
  `image2` varchar(100) NOT NULL,
  `klik` varchar(50) NOT NULL,
  `seller` int(10) NOT NULL,
  PRIMARY KEY (`idpro`),
  KEY `catid` (`catid`),
  KEY `idmerk` (`idmerk`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=15 ;
ALTER TABLE `produk`
   ADD FOREIGN KEY (`idmerk`)
REFERENCES `kp16_web`.`merk`(`idmerk`),
   ADD FOREIGN KEY (`catid`)
REFERENCES `kp16_web`.`kategori`(`catid`);
```

15. Tabel Propinsi

Struktur tabel transaksi pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `propinsi` (
  `kdpro` smallint(5) NOT NULL AUTO_INCREMENT,
  `nama_propinsi` varchar(50) NOT NULL,
  PRIMARY KEY (`kdpro`),
  UNIQUE KEY `nama_pro` (`nama_propinsi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=34 ;
```

16. Tabel Retur

Struktur tabel order temp pada pembangunan situs *e-commerce* penjualan knalpot sepeda motor pada KP16 adalah sebagai berikut:

```
CREATE TABLE IF NOT EXISTS `retur` (
  `id_retur` int(3) NOT NULL,
  `no_transaksi` int(10) NOT NULL,
  `tanggal` date NOT NULL,
  PRIMARY KEY (`id_retur`),
  KEY `no_transaksi` (`no_transaksi`),
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `retur`
  ADD CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`no_transaksi`)
REFERENCES `pesanan` (`no_transaksi`);
```

4.3 Implementasi Antarmuka

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 4.1

Tabel 4. 1 Implementasi antarmuka pengunjung dan member

Proses	Deskripsi	Nama File
Registarsi member	Digunakan untuk melakukan pendaftaran *member* baru.	daftar.php
Login member	Digunakan Digunakan sebagai halaman *login member*	*login*.php

Lupa *password*	Digunakan jika *member* lupa dengan *password*	lupa*password*.php
Edit Akun	Digunakan untuk mengubah profil dan merubah *password* member.	*edit*.account.php
Kategori	Digunakan untuk melihat barang perkategori	left.php
Produk	Digunakan untuk melihat, data produk	home.php
Detail Produk	Digunakan untuk melihat, data produk	detail.php
History pemesanan	Digunakan untuk melihat *history* transaksi yang pernah dilakukan	history.php
Retur	Digunakan member untuk melakukan retur terhadap produk sang sudah diterima	viewretur.php
checkout	Digunakan untuk melihat informasi pembayaran dan memilih alamat pengiriman	checkout.php
logout	Digunakan *member* untuk keluar dari akunnya.	*logout*.php

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program**,** adapun implementasi antarmuka untuk *admin* dan admin situs *e-commerce penjualan knalpot sepeda motor pada KP16* dapat dilihat pada tabel 4.2

Tabel 4. 2 Implementasi antarmuka *admin* dan admin

Proses	Deskripsi	Nama File
Login admin dan *admin*	Digunakan sebagai halaman *login* admin dan *admin*	*login*.php
Lupa *Password*	Digunakan jika admin lupa terhadap *password*nya	lupa_*login*.php
Home	Digunakan sebagai hal awal ketika berhasil *login* yang berisi selamat datang	home.php
Ubah Profil	Digunakan jika admin lupa ingin merubah profil	*editprofile*.php
Ubah *Password* administrator	Digunakan jika ingin merubah *password*	*editpassword*.php
Pengolahan data admin	Digunakan oleh *admin* untuk melihat, menambah, dan merubah status admin	admin.php
Pengolahan data kota	Digunakan oleh admin untuk melihat, menambah, dan merubah data kota	kota.php
Pengolahan data Provinsi	Digunakan oleh admin untuk melihat, menambah, dan merubah data provinsi	provinsi.php
Pengolahan data kategori	Digunakan oleh admin untuk melihat, menambah, dan merubah data kategori	kategori.php
Pengolahan data Produk	Digunakan oleh admin untuk melihat, menambah, dan merubah data produk	produk.php
Pengolahan data Kategori Merk	Digunakan oleh admin untuk melihat, menambah, dan merubah	detail.php

	data KategoriMerk	
Pengolahan data Member	Digunakan oleh admin untuk melihat, mencari, dan merubah status member	member.php
Pengolahan data Kurs	Digunakan oleh admin untuk merubah harga per dollar yang akan digunakan untuk paypal	kurs.php
Pengolahan data Data Transaksi	Digunakan oleh admin untuk melihat data transaksi, data detail transaksi, dan mengubabarang Detail	datatransaksi.php
Retur penjualan	Digunakan oleh admin untuk melakukan konfirmasi pengolahan data retur.	dataretur.php
Ongkos Kirim	Digunakan oleh admin untuk melihat data onkos kirim / harga kirim, merubah, menghapus onkos kirim	ongkir.php
Laporan penjualan harian	Digunakan untuk melihat data laporan penjualan perhari kemudian dicetak ke pdf	laporanharian.php
Laporan penjualan bulanan	Digunakan untuk melihat data laporan penjualan perbulan kemudian dicetak ke pdf	laporanbulan.php
Laporan penjualan tahunan	Digunakan untuk melihat data laporan penjualan pertahun kemudian dicetak ke pdf	laporantahuan.php
Logout	Digunakan administrator Untuk dapat keluar dari aplikasi	*logout*.php

4.4 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.

4.4.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem. Skenario pengujian untuk member dapat dilihat pada tabel 4.3

Tabel 4. 3 Skenario pengujian aplikasi untuk member

Kelas Uji	Butir Uji	Jenis Pengujian
Registrasi *member*	Isi *form* registrasi	*Black Box*
	Verifikasi pendaftaran lewat email	*Black Box*
Login	*Input login* member	*Black Box*
	Verifikasi *login*	*Black Box*
	Lupa *password*	*Black Box*

Edit akun	Menampilkan *form* ubah profil	*Black Box*
	Ubah data *member*	*Black Box*
	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Simpan perubahan	*Black Box*
Produk diskon	Menampilkan data produk diskon	*Black Box*
produk telaku	Menampilkan data produk terlaku	*Black Box*
produk paling banyak dilihat	Menampilkan data produk yang paling banyak dilihat	*Black Box*
Pencarian	Menampilkan hasil dari pencarian berdasarkan produk dan kategori.	*Black Box*
Menu keranjang belanja	Menampilkan data pembelian	*Black Box*
	Mengupdate data pembelian	*Black Box*
	Menghapus data pembelian	*Black Box*
Detail produk	Menampilkan detail produk	*Black Box*
	Share produk melaui *facebook*	*Black Box*
Form konfirmasi	Menampilkan konfirmasi pemesanan	*Black Box*
	Melakukan pembatalan pemesanan	*Black Box*
	melakukan konfirmasi pembayaran	*Black Box*
Form pengiriman	Menampilkan form pengiriman	*Black Box*
	Pilih alamat pengiriman	*Black Box*
Logout	Menghapus *session user*	*Black Box*

Skenario pengujian untuk admin terdiri dari 20 kelas dan beberapa butir uji adapun jenis pengujianya mengunakan metode *Black Box* untuk lebih jelasnya dapat dilihat pada tabel 4.4

Tabel 4. 4 Skenario pengujian untuk admin

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data *login*	*Black Box*
	Verifikasi *login*	*Black Box*
	lupa *password*	*Black Box*
Kategori	Tampilkan data kategori	*Black Box*
	Tambah data kategori	*Black Box*
	Cari data kategori	*Black Box*
	Ubah data ategori	*Black Box*
	Hapus data kategori	*Black Box*
Produk	Tampilkan data produk	*Black Box*
	Tambah data produk	*Black Box*
	Cari data produk	*Black Box*
	Ubah data produk	*Black Box*
	Hapus data produk	*Black Box*
Provinsi	Tampilkan data provinsi	*Black Box*
	Tambah data provinsi	*Black Box*
	Cari data provinsi	*Black Box*
	Ubah data provinsi	*Black Box*
	Hapus data provimsi	*Black Box*
Kota	Tampilkan data kota	*Black Box*
	Tambah data kota	*Black Box*
	Cari data kota	*Black Box*
	Ubah data kota	*Black Box*
	Hapus data kota	*Black Box*
Ongkos kirim	Tampilkan data ongkos kirim	*Black Box*
	Tambah data ongkos kirim	*Black Box*
	Cari data ongkos kirim	*Black Box*
	Ubah data ongkos kirim	*Black Box*
	Hapus data ongkos kirim	*Black Box*

Member	Tampilkan data member	*Black Box*
	Cari data member	*Black Box*
	Ubah data status member	*Black Box*
	Hapus data member	*Black Box*
Kurs	Tampilkan data kurs	*Black Box*
	Ubah data kurs	*Black Box*
Laporan Penjualan Harian	Tampilkan data laporan panjualan harian	*Black Box*
	Cari data laporan panjualan harian	*Black Box*
	Lihat detail data laporan panjualan harian	*Black Box*
	Hapus data laporan panjualan harian	*Black Box*
	Print pdf data laporan panjualan harian	*Black Box*
Laporan Penjualan Bulanan	Tampilkan data laporan panjualan bulanan	*Black Box*
	Cari data laporan panjualan bulanan	*Black Box*
	Lihat detail data laporan panjualan bulanan	*Black Box*
	Hapus data laporan panjualan bulanan	*Black Box*
	Print pdf data laporan panjualan bulanan	*Black Box*
Laporan Penjualan Tahunan	Tampilkan data laporan panjualan tahunan	*Black Box*
	Cari data laporan panjualan tahunan	*Black Box*
	Lihat detail data laporan panjualan tahunan	*Black Box*
	Hapus data laporan panjualan tahunan	*Black Box*

	Print pdf data laporan panjualan tahunan	*Black Box*
Retur Penjualan	Tampilkan data retur	*Black Box*
	Konfirmasi data retur	*Black Box*
Edit Akun	*Edit* profil	*Black Box*
	Edit password	*Black Box*
Logout	Menghapus *session admin*	*Black Box*

Skenario pengujian untuk admin terdiri dari empat kelas dan beberapa butir uji adapun jenis pengujianya mengunakan metode *Black Box* untuk lebih jelasnya dapat dilihat pada tabel 4.5

Tabel 4. 5 Skenario pengujian untuk *admin*

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data *login*	*Black Box*
	Verifikasi *login*	*Black Box*
	lupa *password*	*Black Box*
pengolahan data admin	Tampilkan data admin	*Black Box*
	Tambah data admin	*Black Box*
	Cari data admin	*Black Box*
	Ubah data status admin	*Black Box*
	Hapus data admin	*Black Box*
Edit Akun	*Edit* profil	*Black Box*
	Edit password	*Black Box*
Logout	Menghapus *session Super admin*	*Black Box*

4.4.2 Pengujian

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.yaitu sebagai berikut :

4.4.2.1 *Equivalence Partitioning* 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 4.6:

Tabel 4. 6 Pengujian registrasi member (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : seno	[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	[√] Diterima [ ] Ditolak

		salah dalam pengisian.	
Password contoh masukan : *Password* : seno1234	[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
Confirm *Password* contoh masukan : Comfirm *Password* : seno1234	[A-Z|a-z|0-9] 5-32 char	Data Comfirm *Password* dapat diterima jika *input* data ; 1.Konfirmasi *Password* 5 sampai 32 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4 Harus sama dengan *field Password*	[√] Diterima [ ] Ditolak
Email contoh masukan : Email : seno53@gmail.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 : jl astar gg.jaswar no.9	[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 : Kota. Bandung	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 : 40242	[0-9]	Data kode pos dapat diterima jika *input* data kode pos diisi dengan number saja	[√] Diterima [ ] Ditolak
No Telepon contoh masukan : No telepon : 08562377555	[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 4.7 :

Tabel 4. 7 Pengujian registrasi member (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : se 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
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
Confirm *Password* contoh masukan : Confirm *Password* : ssa Confirm *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 : senowe 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.	[√] Diterima [ ] Ditolak

		tidak mengandung karakter @ dan .	
Alamat contoh masukan : Alamat : jl talun rancaekek no 28	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 : 40394	[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 : 7792310	[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 :	data kosong	Dapat memunculkan pesan kesalahan	[√] Diterima [ ] Ditolak

Masukan kode : XJJWD		bahwa *field* tersebut harus terisi jika kosong atau tidak sesuai dengan generate key	

2. Pengujian *Login* Member

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 4.8 :

Tabel 4. 8 Pengujian *login* member (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Email contoh masukan : Email : senowe@gmail.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*** : tahupedas	[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 4.9 :

Tabel 4. 9 Pengujian *login* member (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
Email contoh masukan : Email : senoaja 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*: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

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 4.10 :

Tabel 4. 10 Pengujian *edit* akun (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : seno	[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 astar gg.jaswar no 9	[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 : Kota. Bandung	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 : 40394	[0-9]	Data kode pos dapat diterima jika *input* data kode pos diisi dengan number	[√] Diterima [ ] Ditolak
No Telepon contoh masukan : No telepon : 7792310	[0-9]	Data kode pos dapat diterima jika *input* data kode pos diisi dengan number	[√] Diterima [ ] Ditolak
Password contoh masukan : ***Password*** : tahupedas	[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 lama.	[√] Diterima [ ] Ditolak
***Password* Baru** contoh masukan : ***Password* Baru:** tahupedasrasaoncom	[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	[√] Diterima [ ] Ditolak

		3. Boleh kombinasi antara angka dengan huruf.	
Confirm *Password* contoh masukan : Confirm *Password* : tahupedas	[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 4.11

Tabel 4. 11 Pengujian *edit* akun (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : se 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 : senoaja 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 astar gg.jaswar no.9	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 : 40394	[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 :	[A-Z|a-z] data kosong	Dapat memunculkan pesan kesalahan bahwa *field* tersebut harus terisi jika	[√] Diterima [ ] Ditolak

08562345		kosong atau salah dalam pengisian jika mengisi dengan huruf	
***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
Confirm *Password* contoh masukan : Confirm *Password* : zz Confirm *Password*:zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfct	<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	[√] Diterima [ ] Ditolak

yytyytrtyryt Confirm *Password* :		*password* yang baru	

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 4.12 :

Tabel 4. 12 Pengujian keranjang belanja / *checkout* (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Data Produk contoh : pilih data produk : knalpot ninja 150 R Kolong; dan persediaanstok'y masih ada.	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	[√] Diterima [ ] Ditolak

		kirim	

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 4.13 :

Tabel 4. 13 Pengujian keranjang belanja / checkout (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
Data Produk contoh : pilih data produk : knalpot ninja 150 R Kolong ; dan persediaan stok'y 0 atau habis	> Stok	Dapat memunculkan pesan kesalahan bahwa detail produk telah habis / stok kosong.	[√] Diterima [ ] Ditolak
Jumlah produk contoh; jumlah produk : 10 ; semetara stok 8	[A-Z|a-z] > Stok	Dapat memunculkan pesan kesalahan bahwa stok tidak mencukupi dan data yang dimasukan invalid apabila diisi dengan huruf atau symbol	[√] Diterima [ ] Ditolak

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 4.14 :

Tabel 4. 14 Pengujian pengiriman (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan

Nama Lengkap contoh masukan : Nama : seno	[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.astar no.9	[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 : Bandung	Pilih Kota [1 |2|3...]	*User* memilih provinsi dengan checkbox yang tersedia dan sebelumnya harus	[√] Diterima [ ] Ditolak

		memilih provinsi	
Kode Pos contoh masukan : Kode Pos : 40394	[0-9]	Data kode pos dapat diterima jika *input* data kode pos diisi dengan number	[√] Diterima [ ] Ditolak
Email contoh masukan : Email : kukurunyk@gmail.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 : 08562377665	[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 4.15 :

Tabel 4. 15 Pengujian pengiriman (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
Nama Lengkap contoh masukan : Nama : di Nama : dsfsgdgdgdgfdgdgdgdgdgdgdgdgdfghgfjhhhgj	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa *field* tersebut harus terisi jika kosong atau salah dalam pengisian jika	[√] Diterima [ ] Ditolak

hggjgjjg Nama :		kurang dari 5 karakter dan lebih dari 32 karakter.	
Alamat contoh masukan : Alamat : jl astar no.9	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 : 40394	[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 : senowe 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	[A-Z|a-z]	Dapat memunculkan	[√] Diterima

contoh masukan : No telepon : 7792310	data kosong	pesan kesalahan bahwa *field* tersebut harus terisi jika kosong atau salah dalam pengisian jika mengisi dengan huruf	[ ] 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 4.16 :

Tabel 4. 16 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 : Mandiri Syariah	[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 : 4475689879** untuk yang membayar secara tunia dan tidak mempunyai rekening bank mengunakan No Pesanan **No Pesanan : 23**	[0-9]	*User* memasukan data No Rekening dan untuk yang membayar secara tunia dan tidak mempunyai rekening bank mengunakan No Pesanan data dapat di terima jika *input* data : 1. No Rekening atau no .Pesanan merupakan angka	[√] Diterima [ ] Ditolak
Jumlah pembayaran contoh masukan : jumlahpembayaran : 2000 sementara jumlah yang harus dibayar 2000	[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 4.17 :

Tabel 4. 17 Pengujian konfirmasi pembayaran (data invalid)

Kasus dan Hasil Uji (Data InValid)			
Data Masukan	In*Valid 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 mempunyai rekening bank mengunakan No Pesanan **No Pesanan :**	[A-Z | a-z] data kosong	Menampilkan data salah dan menampilkan pesan data kosong apabila data belum di isi.	[√] Diterima [ ] Ditolak
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 4.18 :

Tabel 4. 18 Pengujian retur (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Memilih jumlah yang akan di retur contoh ; jumlah : 1	[0-9]	Memilih jumlah barang yang akan direturkan sesuai dengan data Pesanannya	[√] Diterima [ ] Ditolak
Keterangan contoh : barang rusak atau cacad	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 proses retur selanjutnya.	[√] Diterima [ ] Ditolak

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 4.19 :

Tabel 4. 19 Pengujian retur (data invalid)

Kasus dan Hasil Uji (Data InValid)			
Data Masukan	In*Valid 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 pesan kesalahan data kosong atau barang tidak di returkan	[√] Diterima [ ] Ditolak

4.4.2.2 *Equivalence Partitioning* 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 4.20 :

Tabel 4. 20 Pengujian *login* admin (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Email contoh masukan : *user*name : admin	[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 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	[√] Diterima [ ] Ditolak

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 4.21 :

Tabel 4. 21 Pengujian logon admin (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
Email contoh masukan : *User*name: se *User*name :	[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* : 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.	[√] 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 4.22 :

Tabel 4. 22 Pengujian pengolahan data kategori (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Kategori contoh masukan : kategori : Knalpot	[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 meng*edit* 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 4.23 :

Tabel 4. 23 Pengujian pengolahan data kategori (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
Kategori contoh masukan : kategori : fsdjfsjdfgjsgfjsgdfjhgd shfgdsjhfghjsdgfjhsdg fsgfhsgfjsgfjsgdfjhsgd fsdjgffsjfkjsdfkshjfh 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

3. Pengujian Pengolahan Data Merk

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

Tabel 4. 24 Pengujian pengolahan data merk (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Merj contoh masukan : Merk : HONDA	[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 meng*edit* manufktur yang sudah ada atau klik tombol tambah untuk menambahkan manufktur	[√] Diterima [ ] Ditolak

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

Tabel 4. 25 Pengujian pengolahan data merk (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
merk contoh masukan : merk : fsdjfsjdfgjsgfjsgdfjhgd	kosong > 32 char	menghasilkan pesan kesalahan apabila menekan tombol hapus jika kategori	[√] Diterima [ ] Ditolak

shfgdsjhfghjsdgfjhsdg fsgfhsgfjsgfjsgdfjhsgd fsdjgffsjfkjsdfkshjfh manufktur :		tersebut berrelasi dengan data yang lainnya yang berhubungan dengan transaksi. menampilkan pesan kesalahan data kosong	

4. Pengujian Pengolahan Data Produk

Pengujian pengolahan data produk dilakukan ketika sudah meng*input*kan 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 4.26 :

Tabel 4. 26 Pengujian pengolahan data produk (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
kategori contoh masukan : kategori 2TAX	pilih [kategori..]	Memilih data kategori yang sebelumnya telah di *input*kan	[√] Diterima [ ] Ditolak
produk contoh masukan : produk : Knalpot Ninja 150 R Kolong	[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 4.27 :

Tabel 4. 27 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 fsdjgffsjfkjsdfkshjfh produk :	kosong > 32 char	menghasilkan pesan kesalahan apabila menekan tombol hapus jika produk tersebut berrelasi dengan data yang lainnya yang berhubungan dengan transaksi. menampilkan pesan kesalahan data kosong	[√] Diterima [ ] Ditolak

6. 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 4.30 :

Tabel 4. 28 Pengujian pengolahan data provinsi (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Provinsi contoh masukan : Provinsi : Jawa barat	[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	[√] Diterima [ ] Ditolak

		meng*edit* provinsi yang sudah ada atau klik tombol tambah untuk menambahkan provinsi	

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 4.31 :

Tabel 4. 29 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

7. Pengujian Pengolahan Data Kota

Pengujian pengolahan data kota dilakukan ketika sudah meng*input*kan 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 4.32 :

Tabel 4. 30 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 *input*kan	[√] Diterima [ ] Ditolak
Nama kota contoh masukan : kota : Bandung	[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 meng*edit* nama kota yang sudah ada atau klik tombol tambah untuk menambahkan produk	[√] Diterima [ ] Ditolak
Jenis kota contoh masukan : jenis kota : KABUPATEN	pilih kota [KOTA|KABUPATEN]	Memilih data provinsi yang sebelumnya telah di *input*kan	[√] Diterima [ ] Ditolak

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 4.33 :

Tabel 4. 31 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
Jenis kota	kosong	Menampilkan pesan error jika data belum ada atau kosong	[√] Diterima [ ] Ditolak

8. 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 4.38 :

Tabel 4. 32 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 di*input*kan 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 4.39 :

Tabel 4. 33 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 *input*kan huruf atau simbol.	[√] Diterima [ ] Ditolak

9. 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 4.42 :

Tabel 4. 34 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	[√] Diterima [ ] Ditolak

		pelaporan yang akan diprint disajikan dalama format dalam bentuk pdf	
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 4.43 :

Tabel 4. 35 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

13. 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 4.44 :

Tabel 4. 36 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

14. 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 4.45 :

Tabel 4. 37 Pengujian *edit profile* (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : seno	[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	[√] Diterima [ ] Ditolak

		memunculkan pesan kesalahan bahwa *field* tersebut harus terisi jika salah satu *field* tidak terisi atau salah dalam pengisian.	
Email contoh masukan : Email : kukurunyk@gmail.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
Telephone contoh masukan : Telephone : 7792310	[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 astar no.9	[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 4.46 :

Tabel 4. 38 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 : senowe 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
Telephone contoh masukan : Telephone : 7792310	[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 talun rancaekek no 28	data kosong	Dapat memunculkan pesan kesalahan bahwa *field* tersebut harus terisi jika kosong	[√] Diterima [ ] Ditolak

15. Pengujian *Edit Password*

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 4.47 :

Tabel 4. 39 Pengujian *edit password* (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
***Password* lama** contoh masukan : *Password* lama : akuganteng	[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
***Password* Baru** contoh masukan : *Password* baru: kp16.com	[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

Confirm *Password* contoh masukan : Confirm *Password* : kp16aja	[A-Z|a-z|0-9] 5-32 char sama dengan *field password* baru	Data Confirm *Password* dapat diterima jika *input* data ; 1.Konfirmasi *Password* 5 sampai 32 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4 Harus sama dengan *field Password* baru	[√] Diterima [ ] Ditolak

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 4.48 :

Tabel 4. 40 Pengujian *edit password* (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
***Password* Lama** contoh masukan : *Password* : zz *Password*:zzxzxzxzxz xsdfgdgdgdgdgdgdgdf 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. data harus berbeda dengan *password* yang lama.	[√] Diterima [ ] Ditolak
***Password* Baru** contoh masukan : *Password* Baru: zz	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa *field* tersebut	[√] Diterima [ ] Ditolak

Password Baru :zzxzxzxzxzxsdfgdgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt *Password* Baru :		harus terisi jika kosong atau salah dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter. data beda dengan *password* yang lama.	
Confirm *Password* contoh masukan : Confirm *Password* : zz Confirm *Password*:zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt Confirm *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

4.4.2.3 *Equivalence Partioning* Untuk *Admin*

1. Pengujian *Login* Super 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 4.49 :

Tabel 4. 41 Pengujian *login admin* (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Email contoh masukan : *User*name : senowe	[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
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 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf.	[√] Diterima [ ] Ditolak

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 4.42 :

Tabel 4. 42 Pengujian *login admin* (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
Email contoh masukan : *User*name : senowe *User*name :	[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.	
Password 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.	[√] Diterima [ ] Ditolak

2. Pengujian Pengolahan Data Admin

Pengujian pengolahan data admin melingkupi tambah data admin baru dan menonaktifkan status admin. berikut dapat dilihat pengujian pengolahan data admin untuk menambah data admin baru dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.43 :

Tabel 4. 43 Pengujian pengolahan data admin (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Nama Admin contoh masukan : Nama : operator	[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	[√] Diterima [ ] Ditolak

		antara angka dengan huruf. Dapat memunculkan pesan kesalahan bahwa *field* tersebut harus terisi jika salah satu *field* tidak terisi atau salah dalam pengisian.	
Email contoh masukan : Email : kukurunyk@gmail.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
Telephone contoh masukan : Telephone : 7792310	[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.astar no.9	[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	**Alamat** contoh masukan : Alamat : jl milan no 18

		antara angka dengan huruf.	
Password contoh masukan : *Password* baru: kp16aja	[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
Confirm contoh masukan : Confirm *Password* : kp16aja	[A-Z|a-z|0-9] 5-32 char sama dengan *field password* baru	Data Comfirm *Password* dapat diterima jika *input* data ; 1.Konfirmasi *Password* 5 sampai 32 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4 Harus sama dengan *field Password* baru	[√] Diterima [ ] Ditolak

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 4.44 :

Tabel 4. 44 Pengujian pengolahan data admin (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 : senowe 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
Telephone contoh masukan : Telephone : 7792310	[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 astana anyar no.9	data kosong	Dapat memunculkan pesan kesalahan bahwa *field* tersebut harus terisi jika kosong	[√] Diterima [ ] Ditolak

Password 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
Confirm contoh masukan : Confirm *Password* : zz Confirm *Password*:zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt Confirm *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

3. 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 4.45 :

Tabel 4. 45 Pengujian *edit profile* (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
Nama contoh masukan : Nama : seno	[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 : kukurunyk@gmail.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
Telephone contoh masukan : Telephone :	[0-9]	Data kode pos dapat diterima jika *input* data kode pos diisi dengan number	**No Telp** contoh masukan :

08562377666			No telepon : 7792310
Alamat contoh masukan : Alamat : jl astar no.9	[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 4.54 :

Tabel 4. 46 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 : senowe 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	[√] Diterima [ ] Ditolak

		karakter dan lebih dari 32 karakter. tidak mengandung karakter @ dan .	
Telephone contoh masukan : Telephone : 7792310	[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 talun rancaekek no 28	data kosong	Dapat memunculkan pesan kesalahan bahwa *field* tersebut harus terisi jika kosong	[√] Diterima [ ] Ditolak

4. Pengujian *Edit Password*

Berikut dapat dilihat pengujian dari *Edit Password* dengan kasus dan hasil uji (*data Invalid*) kemudian diamati dan menghasilkan kesimpulan dari pengujian tersebut, terdapat pada tabel 4.55 :

Tabel 4. 47 Pengujian *edit password* (data valid)

Kasus dan Hasil Uji (Data Valid)			
Data Masukan	*Valid Class*	Pengamatan	Kesimpulan
***Password* lama** contoh masukan : *Password* lama : akuganteng	[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	[√] Diterima [ ] Ditolak

		antara angka dengan huruf.	
***Password* Baru** contoh masukan : *Password* baru: kp16aja	[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
Confirm *Password* contoh masukan : Confirm *Password* : kp16aja	[A-Z|a-z|0-9] 5-32 char sama dengan *field password* baru	Data Confirm *Password* dapat diterima jika *input* data ; 1.Konfirmasi *Password* 5 sampai 32 char 2.Boleh mengunakan angka 3. Boleh kombinasi antara angka dengan huruf. 4 Harus sama dengan *field Password* baru	[√] Diterima [ ] Ditolak

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 4.56 :

Tabel 4. 48 Pengujian *edit password* (data invalid)

Kasus dan Hasil Uji (Data Invalid)			
Data Masukan	*Invalid Class*	Pengamatan	Kesimpulan
***Password* Lama** contoh masukan : *Password* : zz	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa *field* tersebut	***Password* Lama** contoh

Password:zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt *Password* :		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.	masukan : *Password* : zz *Password*:zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt *Password* :
***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.	***Password* Baru** contoh masukan : *Password* Baru: zz *Password* Baru :zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt *Password* Baru :
Confirm *Password* contoh masukan : Confirm *Password* : zz	<5 char >32 char data kosong	Dapat memunculkan pesan kesalahan bahwa *field* tersebut harus terisi jika kosong atau salah	**Confirm *Password*** contoh masukan :

Confirm *Password*:zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt Confirm *Password* :		dalam pengisian jika kurang dari 5 karakter dan lebih dari 32 karakter. data beda dengan *password* yang baru	Confirm *Password* : zz Confirm *Password*:zzxzxzxzxzxsdfgdgdgdgdgdgdgdf78656775575gjhghfctyytyytrtyryt Confirm *Password* :

4.4.2.3 *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 di*input*kan

C. Menampilkan pesan kesalahan dari *password* yang di*input*kan

D.Menampilkan pesan kesalahan dari comfirm *password* yang di*input*kan

E.Menampilkan pesan kesalahan dari email yang di*input*kan

F. Menampilkan pesan kesalahan dari alamat yang di*input*kan

G. Menampilkan pesan kesalahan dari tanggal lahir yang di*input*kan

H. Menampilkan pesan kesalahan dari provinsi yang di*input*kan

I. Menampilkan pesan kesalahan dari kota yang di*input*kan

J. Menampilkan pesan kesalahan dari kode pos yang di*input*kan

K. Menampilkan pesan kesalahan dari no.telepon yang di*input*kan

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 4.49

Tabel 4. 49 *Cause effect* testing registrasi member

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

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 di*input*kan

C. Menampilkan pesan kesalahan dari *password* yang di*input*kan

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 4.50

Tabel 4. 50 *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;

1. Jika member memasukan email benar dan *password* benar maka *login* berhasil.
2. Jika member memasukan email salah dan *password* benar maka sistem akan menampilkan pesan kesalahan dari email yang di*input*kan

3. Jika member memasukan email salah dan *password* benar maka sistem akan menampilkan pesan kesalahan dari *password* yang di*input*kan.
4. Jika member memasukan email salah dan *password* salah maka sistem akan menampilkan pesan kesalahan dari email yang di*input*kan dan menampilkan pesan kesalahan dari *password* yang di*input*kan.

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 di*input*kan

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

D. Menampilkan pesan kesalahan dari *password* baru yang di*input*kan

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 4.51

Tabel 4. 51 *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 di*input*kan

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 4.52

Tabel 4. 52 *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 di*input*kan

C. Menampilkan pesan kesalahan dari alamat yang di*input*kan

D. Menampilkan pesan kesalahan dari provinsi yang di*input*kan

E. Menampilkan pesan kesalahan dari kota yang di*input*kan

F. Menampilkan pesan kesalahan dari kode pos yang di*input*kan

G. Menampilkan pesan kesalahan dari email yang di*input*kan

H. Menampilkan pesan kesalahan dari no.telepon yang di*input*kan

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 4.53

Tabel 4. 53 *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. Atas nama bank memenuhi kondisi yang telah ditentukan

2. Jumlah pembayaran memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari atas nama bank yang di*input*kan

C. Menampilkan pesan kesalahan dari jumlah pembayaran yang di*input*kan

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 4.54

Tabel 4. 54 *Cause effect testing* konfirmasi pembayaran melalui bank

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

7. *Cause effect testing* retur

***Input states* :**

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

2. Isi Alasan dikembalikan yang memenuhi kondisi yang telah ditentukan

***Output states* :**

A. Data disimpan kedalam database

B. Menampilkan pesan kesalahan dari Jumlah produk yang akan diretur bank yang di*input*kan

C. Menampilkan pesan kesalahan dari Alasan dikembalikan barang yang akan diretur bank yang di*input*kan

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 4.55

Tabel 4. 55 *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

4.4.2.4 *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. *User*name memenuhi kondisi yang telah ditentukan

2. *Password* memenuhi kondisi yang telah ditentukan

***Output states* :**

A. *Login* berhasil

B. Menampilkan pesan kesalahan dari *user*name yang di*input*kan

C. Menampilkan pesan kesalahan dari *password* yang di*input*kan

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 4.56

Tabel 4. 56 *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 di*input*kan atau dipilih

C. Menampilkan pesan kesalahan dari produk yang di*input*kan

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 table 4.57.

Tabel 4. 57 *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 di*input*kan atau dipilih

C. Menampilkan pesan kesalahan dari manufaktur yang di*input*kan atau dipilih

D. Menampilkan pesan kesalahan dari harga yang di*input*kan

E. Menampilkan pesan kesalahan dari stok yang di*input*kan

F.Menampilkan pesan kesalahan dari berat yang di*input*kan

G.Menampilkan pesan kesalahan dari diskon yang di*input*kan

H.Menampilkan pesan kesalahan dari deskripsi yang di*input*kan

I.Menampilkan pesan kesalahan dari gambar produk yang di*input*kan 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 4.58

Tabel 4. 58 *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 di*input*kan atau dipilih

C. Menampilkan pesan kesalahan dari nama kota yang di*input*kan

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 4.59

Tabel 4. 59 *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 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 di*input*kan

C.Menampilkan pesan kesalahan dari email yang di*input*kan

D.Menampilkan pesan kesalahan dari telephone yang di*input*kan

E.Menampilkan pesan kesalahan dari alamat yang di*input*kan

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 4.61

Tabel 4. 60 *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

6. *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 di*input*kan

C.Menampilkan pesan kesalahan dari *password* baru yang di*input*kan

D.Menampilkan pesan kesalahan dari confirm yang di*input*kan

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 4.62

Tabel 4. 61 *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

4.4.2.5 *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. Email 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 email yang di*input*kan

C. Menampilkan pesan kesalahan dari *password* yang di*input*kan

D. Menampilkan pesan kesalahan dari type yang di*input*kan 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 4.63

Tabel 4. 62 *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 edit* profil

***Input states* :**

1. Nama memenuhi kondisi yang telah ditentukan

2. *User*name 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 di*input*kan

C.Menampilkan pesan kesalahan dari *user*name yang di*input*kan

D.Menampilkan pesan kesalahan dari telepon yang di*input*kan

E.Menampilkan pesan kesalahan dari alamat yang di*input*kan

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 4.65.

Tabel 4. 63 *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

4. *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 di*input*kan

C.Menampilkan pesan kesalahan dari *password* baru yang di*input*kan

D.Menampilkan pesan kesalahan dari confirm yang di*input*kan

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 *edit password* dapat dilihat pada tabel 4.66.

Tabel 4. 64 *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

4.4.2.6 Kuisioner Beta

1. Kuesioner untuk Pegawai dan Pemilik

Kuesioner disebarkan menggunakan teknik sampling yaitu *Simple Random Sampling* yang disebarkan kepada 5 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

Daftar pertanyaan yang diberikan kepada pemilik dan karyawan adalah sebagai berikut :

1) Bagaimana tampilan perangkat lunak untuk pengelolaan data?

   a. Sangat Menarik
   b. Menarik
   c. Cukup Menarik
   d. Kurang Menarik
   e. Tidak Menarik

2) Apakah Anda mengerti dengan instruksi yang terdapat dalam website tersebut mudah dimengerti?

   a. Sangat mudah dimengerti
   b. Mudah dimengerti
   c. Cukup mudah dimengerti
   d. Cukup sulit dimengerti
   e. Sulit dimengerti

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

   a. Sangat membantu
   b. Membantu
   c. Cukup membantu
   d. Kurang Membantu
   e. Tidak Membantu

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

   a. Sangat membantu
   b. Membantu
   c. Cukup membantu
   d. Kurang membantu
   e. Tidak membantu

5) Apakah website ini dapat membantu dalam membuat laporan penjualan?

   a. Sangat membantu
   b. Membantu
   c. Cukup membantu
   d. Kurang membantu
   e. Tidak membantu

Berikut ini adalah hasil prosentase masing-masing jawaban yang ditujukan untuk pemilik dan 4 orang pegawai KP16.

1. Bagaimana tampilan perangkat lunak untuk pengelolaan data ?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat Menarik			
2	Menarik	3	60%	12
3	Cukup Menarik	2	40%	6
4	Kurang Menarik			
5	Tidak Menarik			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 3 orang atau 50% mengatakan bahwa perangkat lunak yang dibuat Menarik dan 2 orang atau 50% mengatakan perangkat lunak tersebut cukup menarik sehingga secara keseluruhan tampilan perangkat lunak tersebut cukup menarik.

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

No	Keterangan	Responden	Prosentase	Skala
1	Sangat Dimengerti			
2	Dimengerti	1	50%	4
3	Cukup Dimengerti	4	50%	12
4	Kurang Dimengerti			
5	Tidak Dimengerti			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 4 orang atau 80% 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?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat Membantu			
2	Membantu	4	80%	16
3	Cukup Membantu	1	20%	3
4	Kurang Membantu			
5	Tidak Membantu			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 4 orang atau 80% mengatakan bahwa perangkat lunak yang dibuat membantu dan sebanyak 1 orang atau 20% mengatakan bahwa perangkat lunak yang dibuat cukup membantu. Sehingga secara keseluruhan perangkan lunak tersebut membantu.

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

No	Keterangan	Responden	Prosentase	Skala
1	Sangat Membantu	3	80%	15
2	Membantu	2	20%	8
3	Cukup Membantu			
4	Kurang Membantu			
5	Tidak Membantu			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 3 orang atau 60% mengatakan bahwa perangkat lunak yang dibuat sangat membantu dan sebanyak 2 orang atau 40% mengatakan bahwa perangkat lunak yang dibuat membantu. Sehingga secara keseluruhan perangkan lunak tersebut membantu dan cukup membantu.

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

No	Keterangan	Responden	Prosentase	Skala
1	Sangat Membantu			
2	Membantu	1	50%	4
3	Cukup Membantu	4	50%	12

4	Kurang Membantu			
5	Tidak Membantu			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 1 orang atau 20% mengatakan bahwa perangkat lunak yang dibuat membantu dan sebanyak 4 orang atau 80% mengatakan bahwa perangkat lunak yang dibuat cukup membantu. Sehingga secara keseluruhan perangkan lunak tersebut membantu.

2. Kuesioner untuk pengguna

Kuesioner disebarkan menggunakan teknik sampling yaitu *Simple Random Sampling* yang disebarkan kepada 10 pengguna. Dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner ini terdiri dari 6 pertanyaan (contoh kuesioner dapat diliihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 5.

Daftar pertanyaan yang diberikan kepada 10 pengguna secara *random* adalah sebagai berikut :

1) Apakah anda setuju tampilan website ini menarik dan mudah dipahami bagi pengguna?

   a. Sangat setuju
   b. Setuju
   c. Netral
   d. Kurang setuju
   e. Tidak setuju

2) Apakah website ini anda butuhkan untuk melakukan pembelanjaan di KP16?

   a. Sangat dibutuhkan
   d. Kurang dibutuhkan

b. Dibutuhkan
e. Tidak dibutuhkan

c. Netral

3) Apakah website ini mempermudah anda melakukan pembelanjaan produk di KP16 ?

a. Sangat mempermudah
d. Kurang Mempermudah

b. Mempermudah
e. Tidak Mempermudah

c. Netral

4) Apakah instruksi-instruksi yang diberikan mudah untuk dimengerti ?

a. Sangat mudah
d. Sulit

b. Mudah
e. Sangat sulit

c. Biasa saja

5) Setelah melihat website e-commerce KP16 apakah anda berminat untuk mengunjunginya lagi lain waktu ?

a. Sangat berminat
d. Tidak berminat

b. Berminat
e. Sangat tidak berminat

c. Mungkin

6) Website e-commerce KP16 memberikan informasi secara detail mengenai produk yang ditawarkan ?

a. Sangat setuju
d. Tidak setuju

b. Setuju
e. Sangat tidak setuju

c. Biasa saja

7) Apakah website http://www.kp-16.com menghasilkan *output* yang sesuai dengan keinginan anda sebagai pengguna ?

a. Sangat sesuai
b. Sesuai
c. Biasa saja
d. Tidak sesuai
e. Sangat tidak sesuai

8) Apakah informasi produk yang diberikan sesuai dengan informasi yang dibutuhkan?

a. Sangat sesuai
b. Sesuai
c. Netral
d. Kurang sesuai
e. Tidak sesuai

9) Apakah anda setuju sistem pembayaran yang digunakan mempermudah transaksi?

a. Sangat setuju
b. Setuju
c. Netral
d. Kurang setuju
e. Tidak setuju

10) Apakah anda puas berbelanja dengan mengunakan website kp16 ini ?

a. Sangat puas
b. Puas
c. Netral
d. Kurang puas
e. Tidak puas

Berikut ini adalah hasil prosentase masing-masing jawaban yang ditujukan untuk pemilik dan 4 orang pegawai KP16

1. Apakah anda setuju tampilan website ini menarik dan mudah dipahami bagi pengguna?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat Setuju	3	30 %	15
2	Setuju	5	50%	20
3	Netral	2	20%	6
4	Kurang Setuju			
5	Tidak Setuju			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 3 orang atau 30% mengatakan bahwa sangat setuju bahwa tampilan website *user friendly*, sebanyak 5 orang atau 30% mengatakan bahwa setuju bahwa tampilan website *user friendly*, sementara 2 orang atau 20% mengatakan biasa saja . Sehingga secara keseluruhan website tersebut *user friendly* .

2. Apakah website ini anda butuhkan untuk melakukan pembelanjaan di KP16?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat dibutukan	2	20%	10
2	Dibutuhkan	3	30%	12
3	Netral	5	50%	15
4	Kurang dibutuhkan			
5	Tidak dibutuhkan			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 20% mengatakan bahasanya sangat dibutuhkan, sementara 3 orang atau 30% mengatakan mudah dibutuhkan, sementara 5 orang atau 50% mengatakan netral . Sehingga secara keseluruhan website tersebut dibutuhkan.

3. Apakah website ini mempermudah anda melakukan pembelanjaan produk di KP16 ?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat mempermudah	2	20%	10
2	mempermudah	5	50%	20
3	Netral	3	30%	15
4	Kurang mempermudah			
5	Tidak mempermudah			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 60% mengatakan sangat setuju bahwa website ini

mempermudah melakukan pembelanjaan produk ,sementara 5 orang atau 50% setuju bahwa website ini mempermudah melakukan pembelanjaan produk, sementara 3 orang atau 10% mengatakan netral . Sehingga secara keseluruhan website ini mempermudah melakukan pembelanjaan produk yang dibutuhkan.

4. Apakah instruksi-instruksi yang diberikan mudah untuk dimengerti ?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat mudah			
2	Mudah	7	70%	28
3	Biasa saja	3	30%	9
4	Sulit			
5	Sangat Sulit			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 7 orang atau 70% mengatakan bahwa website ini mudah dalam tahap instrusksi yang diberikan, sementara 3 orang atau 30% mengatakan bahwa website ini biasa saja.

5. Setelah melihat website e-commerce KP16 apakah anda berminat untuk mengunjunginya lagi lain waktu?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat berminat			
2	Berminat	8	80%	32
3	Mungkin	2	20%	6
4	Kurang Berminat			
5	Tidak berminat			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 8 orang atau 80% mengatakan bahwa website ini diminati oleh *user*, sementara 2 orang atau 20% mengatakan bahwa website ini sedikit yang mungkin berminat berkunjung kembali.

6. Website e-commerce KP16 memberikan informasi secara detail mengenai produk yang ditawarkan?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat setuju			
2	Setuju	6	60%	24
3	Biasa saja	4	40%	12
4	Tidak Setuju			
5	Sangat Tidak Setuju			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 6 orang atau 60% mengatakan bahwa website ini memberikan informasi secara detail mengenai produk yang ditawarkan, sementara 4 orang atau 40% mengatakan bahwa website ini biasa saja dalam hal memberikan informasi secara detail mengenai produk yang ditawarkan.

7. Apakah website http://www.kp-16.com menghasilkan *output* yang sesuai dengan keinginan anda sebagai pengguna ?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat sesuai			
2	Sesuai	8	80%	32
3	Biasa saja	2	20%	6
4	Tidak Sesuai			
5	Sangat Tidak Sesuai			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 8 orang atau 20% mengatakan bahwa website ini sesuai keinginan pengguna, sementara 2 orang atau 20% mengatakan bahwa website ini biasa saja.

8. Apakah informasi produk yang diberikan sesuai dengan informasi yang dibutuhkan?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat sesuai			
2	Sesuai	5	50%	20
3	Netral	5	50%	15
4	Kurang Sesuai			
5	Tidak Sesuai			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 5 orang atau 50% mengatakan bahwa website ini sesuai informasi mengenai produk yang diberikan, sementara 5 orang atau 50% mengatakan bahwa website ini sesuai informasi mengenai produk yang diberikan

9. Apakah anda setuju sistem pembayaran yang digunakan mempermudah transaksi ?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat Setuju	2	20%	10
2	Setuju	6	60%	24
3	Netral	1	10%	3
4	Kurang Setuju	1	10%	2
5	Tidak Setuju			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 2 orang atau 20% mengatakan Sangat setuju bahwa sistem pembayaran yang digunakan mempermudah transaksi, 6 orang atau 60% mengatakan setuju bahwa sistem pembayaran yang digunakan mempermudah transaksi, 1 orang atau 10% mengatakan netral bahwa sistem pembayaran yang digunakan mempermudah transaksi dan 1 orang atau 10% mengatakan kurang setuju bahwa sistem pembayaran yang digunakan mempermudah transaksi Sehingga secara

keseluruhan website ini memberikan informasi proses pemesanan yang sesuai dengan pesanan yang dilakukan.

10. Apakah anda puas berbelanja dengan mengunakan website kp16 ini ?

No	Keterangan	Responden	Prosentase	Skala
1	Sangat puas			
2	Puas	5	50%	20
3	Netral	5	50%	15
4	Kurang Puas			
5	Tidak Puas			

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa sebanyak 5 orang atau 50% mengatakan bahwa anda puas berbelanja dengan mengunakan website ini, sementara 5 orang atau 50% mengatakan bahwa anda puas berbelanja dengan mengunakan website ini Sehingga secara keseluruhan sistem pembayaran dalam website ini memudahkan pelanggan.

4.4.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil prosentase diatas yang didapatkan dari pengujian pengguna admin dan *user*, maka sistem penerapan *e-commerce* untuk sistem penjualan ini sudah sesuai dengan tujuan yaitu KP16 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.