BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini akan dilakukan implementasi dan pengujian terhadap sistem informasi kearsipan yang dibangun. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan di implementasikan ke dalam bahasa pemrograman. Setelah implementasi maka dilakukan pengujian terhadap sistem yang baru akan dilihat kekurangan-kekurangan pada aplikasi yang baru untuk pengembangan sistem selanjutnya.

4.1 Implementasi Sistem

Tahapan implementasi merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan tahap dimana system siap untuk dioperasikan, yang terdiri dari penjelasan mengenai lingkungan implementasi, dan implementasi program.

Untuk mendukung aplikasi yang diterapkan pada Toko Yoyo Sport, maka dalam hal ini menggunakan perangkat keras dan perangkat lunak yang menunjang dalam pembangunan aplikasi *e-commerce* pada Toko Yoyo Sport.

4.1.1 Perangkat Keras Yang Digunakan

Spesifikasi perangkat keras yang digunakan dalam pembangunan *website e-commerce* pada toko Yoyo Sport adalah tercantum pada Tabel IV.1 :

Tabel IV.1. Perangkat Keras yang Digunakan

No	Perangkat Keras	Spesifikasi
1	Processor	Core i3 2,4 GHz
2	RAM	1 GB DDR3
3	Harddisk	320 GB
4	Monitor	HD LED LCD 14”
5	VGA	ATI Radeon HD 5470 512 MB
6	Lan Card	10/100 Mbps
7	Koneksi Internet	600 kbps
8	Printer	Epson CX 5500

4.1.2 Perangkat Lunak Yang Digunakan

Spesifikasi perangkat lunak yang digunakan dalam pembangunan *website e-commerce* pada Yoyo Sport adalah tercantum pada table IV.2 sebagai berikut :

Tabel IV.2. Perangkat Lunak yang Digunakan

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows 7 Ultimate 32 Bit
2	Web Browser	Mozilla Firefox 3.0, Google Chrome
3	Bahasa Pemrograman	PHP
4	DBMS	MySQL 5.0.27
5	Code Editor	Adobe Dreamweaver CS5
6	DFD Modeler	Microsoft Visio 2007
7	Web Server	XAMP 2.5.8

4.1.3 Implementasi Database

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

4.1.3.1 Implementasi Tabel Barang

Implementasi dari tabel barang dapat dilihat pada Tabel IV.3 :

Tabel IV.3. Implementasi Tabel Barang

Field	Type	Collation	Attributes	Null	Default	Extra
id_barang	bigint(20)			No	*None*	auto_increment
id_kategori	int(11)			No	*None*	
nama_barang	varchar(50)	latin1_general_ci		No	*None*	
id_merek	int(11)			No	*None*	
ditampilkan	varchar(15)	latin1_general_ci		No	true	

```
CREATE TABLE IF NOT EXISTS `barang` (
  `id_barang` bigint(20) NOT NULL AUTO_INCREMENT,
  `id_kategori` int(11) NOT NULL,
  `nama_barang`     varchar(50)     CHARACTER     SET     latin1     COLLATE
latin1_general_ci NOT NULL,
  `id_merek` int(11) NOT NULL,
  `ditampilkan`     varchar(15)     CHARACTER     SET     latin1     COLLATE
latin1_general_ci NOT NULL DEFAULT 'true',
  PRIMARY KEY (`id_barang`),
  KEY `id_kategori` (`id_kategori`),
  KEY `id_merek` (`id_merek`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=50 ;
ALTER TABLE `barang`
  ADD  CONSTRAINT  `barang_ibfk_2`  FOREIGN  KEY  (`id_merek`)  REFERENCES
`merek` (`id_merek`),
  ADD CONSTRAINT `barang_ibfk_1` FOREIGN KEY (`id_kategori`) REFERENCES
`kategori` (`id_kategori`);
```

4.1.3.2 Implementasi Tabel Merek

Implementasi dari tabel merek dapat dilihat pada Tabel IV.4 :

Tabel IV.4 Implementasi Tabel Merek

Field	Type	Collation	Attributes	Null	Default	Extra
id_merek	int(11)			No	*None*	auto_increment
nama_merek	varchar(21)	latin1_swedish_ci		No	*None*	

```
CREATE TABLE IF NOT EXISTS `merek` (
  `id_merek` int(11) NOT NULL AUTO_INCREMENT,
  `nama_merek` varchar(21) NOT NULL,
  PRIMARY KEY (`id_merek`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;
```

4.1.3.3 Implementasi Tabel Detail Barang

Implementasi dari tabel detail_barang dapat dilihat pada Tabel IV.5 :

Tabel IV.5 Implementasi Tabel detail Barang

Field	Type	Collation	Attributes	Null	Default	Extra
id_detail_barang	bigint(20)			No	None	auto_increment
id_barang	bigint(20)			No	None	
id_warna	int(11)			No	None	
ukuran	varchar(10)	latin1_general_ci		No	None	
harga	double			No	0	
diskon	int(11)			No	0	
berat	double			No	None	
stok	int(11)			No	None	
url_gambar_1	text	latin1_swedish_ci		Yes	NULL	
url_gambar_2	text	latin1_swedish_ci		Yes	NULL	
url_gambar_3	text	latin1_swedish_ci		Yes	NULL	
ditampilkan	varchar(15)	latin1_general_ci		No	true	

```
CREATE TABLE IF NOT EXISTS `detail_barang` (
  `id_detail_barang` bigint(20) NOT NULL AUTO_INCREMENT,
  `id_barang` bigint(20) NOT NULL,
  `id_warna` int(11) NOT NULL,
  `ukuran` varchar(10) CHARACTER SET latin1 COLLATE latin1_general_ci NOT
NULL,
  `harga` double NOT NULL DEFAULT '0',
  `diskon` int(11) NOT NULL DEFAULT '0',
  `berat` double NOT NULL,
  `stok` int(11) NOT NULL,
  `url_gambar_1` text,
  `url_gambar_2` text,
  `url_gambar_3` text,
  `ditampilkan`    varchar(15)    CHARACTER    SET    latin1    COLLATE
latin1_general_ci NOT NULL DEFAULT 'true',
  PRIMARY KEY (`id_detail_barang`),
  KEY `id_barang` (`id_barang`),
  KEY `id_warna` (`id_warna`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=69 ;
ALTER TABLE `detail_barang`
  ADD   CONSTRAINT   `detail_barang_ibfk_2`   FOREIGN   KEY   (`id_warna`)
REFERENCES `warna` (`id_warna`),
```

```
  ADD   CONSTRAINT   `detail_barang_ibfk_1`   FOREIGN   KEY   (`id_barang`)
REFERENCES `barang` (`id_barang`);
```

4.1.3.4 Implementasi Tabel Kategori

Implementasi dari tabel kategori dapat dilihat pada Tabel IV.6 :

Tabel IV.6 Implentasi Tabel Kategori

Field	Type	Collation	Attributes	Null	Default	Extra
id_kategori	int(11)			No	*None*	auto_increment
nama_kategori	varchar(50)	latin1_general_ci		No	*None*	
url_gambar	text	latin1_swedish_ci		Yes	*NULL*	
ditampilkan	varchar(15)	latin1_general_ci		No	true	

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` int(11) NOT NULL AUTO_INCREMENT,
  `nama_kategori`    varchar(50)    CHARACTER    SET    latin1    COLLATE
latin1_general_ci NOT NULL,
  `url_gambar` text,
  `ditampilkan`    varchar(15)    CHARACTER    SET    latin1    COLLATE
latin1_general_ci NOT NULL DEFAULT 'true',
  PRIMARY KEY (`id_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;
```

4.1.3.5 Implementasi Tabel Detail_Pesanan

Implementasi dari tabel detail_pesanan dapat dilihat pada Tabel IV.7 :

Tabel IV.7 Implementasi Tabel Detail Pesanan

Field	Type	Collation	Attributes	Null	Default	Extra
id_detail_pesanan	bigint(20)			No	*None*	auto_increment
id_pesanan	bigint(20)			No	*None*	
id_detail_barang	bigint(20)			No	*None*	
harga	double			No	*None*	
diskon	int(11)			Yes	0	
ukuran	varchar(10)	latin1_general_ci		No	*None*	
berat	double			No	*None*	
jumlah	int(11)			No	*None*	

```
CREATE TABLE IF NOT EXISTS `detail_pesanan` (
  `id_detail_pesanan` bigint(20) NOT NULL AUTO_INCREMENT,
  `id_pesanan` bigint(20) NOT NULL,
  `id_detail_barang` bigint(20) NOT NULL,
  `harga` double NOT NULL,
  `diskon` int(11) DEFAULT '0',
  `ukuran` varchar(10) COLLATE latin1_general_ci NOT NULL,
  `berat` double NOT NULL,
  `jumlah` int(11) NOT NULL,
```

```
  PRIMARY KEY (`id_detail_pesanan`),
  KEY `id_pesanan` (`id_pesanan`),
  KEY `id_detail_barang` (`id_detail_barang`)
)  ENGINE=InnoDB     DEFAULT  CHARSET=latin1  COLLATE=latin1_general_ci
AUTO_INCREMENT=87 ;
ALTER TABLE `detail_pesanan`
  ADD  CONSTRAINT  `detail_pesanan_ibfk_1`  FOREIGN  KEY  (`id_pesanan`)
REFERENCES `pesanan` (`id_pesanan`),
  ADD CONSTRAINT `detail_pesanan_ibfk_2` FOREIGN KEY (`id_detail_barang`)
REFERENCES `detail_barang` (`id_detail_barang`);
```

4.1.3.6 Implementasi Tabel Provinsi

Implementasi dari tabel provinsi dapat dilihat pada Tabel IV.8 :

Tabel IV.8 Implementasi Tabel Provinsi

Field	Type	Collation	Attributes	Null	Default	Extra
id_provinsi	int(11)			No	*None*	auto_increment
nama_provinsi	varchar(30)	latin1_swedish_ci		No	*None*	
ditampilkan	varchar(15)	latin1_swedish_ci		No	true	

```
CREATE TABLE IF NOT EXISTS `provinsi` (
  `id_provinsi` int(11) NOT NULL AUTO_INCREMENT,
  `nama_provinsi` varchar(30) NOT NULL,
  `ditampilkan` varchar(15) NOT NULL DEFAULT 'true',
  PRIMARY KEY (`id_provinsi`),
  UNIQUE KEY `nama_provinsi` (`nama_provinsi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=34 ;
```

4.1.3.7 Implementasi Tabel Komentar

Implementasi dari tabel komentar dapat dilihat pada Tabel IV.9 :

Tabel IV.9 Implementasi dari tabel komentar

Field	Type	Collation	Attributes	Null	Default	Extra
id_komentar	bigint(20)			No	*None*	auto_increment
id_pelanggan	bigint(20)			No	*None*	
id_barang	bigint(20)			No	0	
tanggal	date			No	*None*	
isi	longtext	latin1_general_ci		No	*None*	
ditampilkan	varchar(15)	latin1_general_ci		No	true	
id_petugas	int(11)			No	*None*	

```
CREATE TABLE IF NOT EXISTS `komentar` (
  `id_komentar` bigint(20) NOT NULL AUTO_INCREMENT,
  `id_pelanggan` bigint(20) NOT NULL,
  `id_barang` bigint(20) NOT NULL DEFAULT '0',
  `tanggal` date NOT NULL,
  `isi` longtext COLLATE latin1_general_ci NOT NULL,
```

```
  `ditampilkan` varchar(15) COLLATE latin1_general_ci NOT NULL DEFAULT
'true',
  `id_petugas` int(11) NOT NULL,
  PRIMARY KEY (`id_komentar`),
  KEY `id_pelanggan` (`id_pelanggan`),
  KEY `id_barang` (`id_barang`),
  KEY `id_petugas` (`id_petugas`)
)  ENGINE=InnoDB     DEFAULT  CHARSET=latin1  COLLATE=latin1_general_ci
AUTO_INCREMENT=4 ;
ALTER TABLE `komentar`
  ADD CONSTRAINT `komentar_ibfk_5` FOREIGN KEY (`id_petugas`) REFERENCES
`petugas` (`id_petugas`),
  ADD  CONSTRAINT  `komentar_ibfk_3`  FOREIGN  KEY  (`id_pelanggan`)
REFERENCES `pelanggan` (`id_pelanggan`),
  ADD CONSTRAINT `komentar_ibfk_4` FOREIGN KEY (`id_barang`) REFERENCES
`barang` (`id_barang`);
```

4.1.3.8 Implementasi Tabel Kota

Implementasi dari tabel kota dapat dilihat pada Tabel IV.10 :

Tabel IV.10 Implementasi Dari Tabel Kota

Field	Type	Collation	Attributes	Null	Default	Extra
id_kota	bigint(20)			No	*None*	auto_increment
id_provinsi	int(11)			No	*None*	
nama_kota	varchar(50)	latin1_swedish_ci		No	*None*	
ongkos_kirim	double			No	*None*	
ditampilkan	varchar(15)	latin1_swedish_ci		No	true	

```
CREATE TABLE IF NOT EXISTS `kota` (
  `id_kota` bigint(20) NOT NULL AUTO_INCREMENT,
  `id_provinsi` int(11) NOT NULL,
  `nama_kota` varchar(50) NOT NULL,
  `ongkos_kirim` double NOT NULL,
  `ditampilkan` varchar(15) NOT NULL DEFAULT 'true',
  PRIMARY KEY (`id_kota`),
  KEY `id_provinsi` (`id_provinsi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=513 ;
ALTER TABLE `kota`
  ADD  CONSTRAINT  `kota_ibfk_1`  FOREIGN  KEY  (`id_provinsi`)  REFERENCES
`provinsi` (`id_provinsi`);
```

4.1.3.9 Implementasi Tabel Kunjungan

Implementasi dari tabel kunjungan dapat dilihat pada Tabel IV.11 :

Tabel IV.11. Implementasi Dari Tabel Kunjungan

Field	Type	Collation	Attributes	Null	Default	Extra
id_lihat_barang	bigint(20)			No	None	auto_increment
id_barang	bigint(20)			No	None	
id_pelanggan	bigint(20)			No	0	
waktu	varchar(25)	latin1_general_ci		No	None	
deskripsi	text	latin1_general_ci		No	None	

```
CREATE TABLE IF NOT EXISTS `kunjungan` (
  `id_kunjungan` int(11) NOT NULL AUTO_INCREMENT,
  `id_barang` varchar(9) NOT NULL,
  `id_pelanggan` int(11) NOT NULL,
  `waktu` varchar(61) NOT NULL,
  `deskripsi` text NOT NULL,
  PRIMARY KEY (`id_kunjungan`)
Foreign Key (id_barang) references barang(id_barang),
Foreign Key (id_pelanggan) references pelanggan(id_pelanggan)
) ENGINE= innoDB;
```

4.1.3.10 Implementasi Tabel Logstok

Implementasi dari tabel logstok dapat dilihat pada Tabel IV.12 :

Tabel IV.12. Implementasi Dari Tabel Logstok

Field	Type	Collation	Attributes	Null	Default	Extra
di_log_stok	int(11)			No	None	auto_increment
id_detail_barang	int(11)			No	None	
stok_awal	int(11)			No	None	
stok_akhir	int(11)			No	None	
perubahan_stok	int(11)			No	None	
tanggal	varchar(25)	latin1_swedish_ci		No	None	
catatan	varchar(255)	latin1_swedish_ci		No	None	

```
CREATE TABLE IF NOT EXISTS `logstok` (
  `di_logstok` int(11) NOT NULL AUTO_INCREMENT,
  `id_detail_barang` int(11) NOT NULL,
  `stok_awal` int(11) NOT NULL,
  `stok_akhir` int(11) NOT NULL,
  `perubahan_stok` int(11) NOT NULL,
  `tanggal` date NOT NULL,
  `catatan` varchar(255) NOT NULL,
  PRIMARY KEY (`id_logstok`),
Foreign Key (id_detail_barang) references detail_barang(id_detail_barang)
) ENGINE=InnoDB;
```

4.1.3.11 Implementasi Tabel Pelanggan

Implementasi dari tabel log_stok dapat dilihat pada Tabel IV.13 :

Tabel IV.13. Implementasi Dari Tabel Pelanggan

Field	Type	Collation	Attributes	Null	Default	Extra
id_pelanggan	bigint(20)			No	*None*	auto_increment
tanggal_daftar	date			No	*None*	
email_pelanggan	varchar(100)	latin1_general_ci		No	*None*	
password_pelanggan	varchar(64)	latin1_general_ci		No	*None*	
nama_pelanggan	varchar(50)	latin1_general_ci		No	*None*	
alamat	text	latin1_general_ci		No	*None*	
id_kota	bigint(20)			No	*None*	
kode_pos	varchar(15)	latin1_general_ci		No	*None*	
telepon	varchar(20)	latin1_general_ci		No	*None*	
telepon_genggam	varchar(20)	latin1_general_ci		Yes	*NULL*	
fax	varchar(20)	latin1_general_ci		Yes	*NULL*	
diaktifkan	varchar(15)	latin1_general_ci		No	true	

```
  CREATE TABLE IF NOT EXISTS `pelanggan` (
    `id_pelanggan` bigint(20) NOT NULL AUTO_INCREMENT,
    `tanggal_daftar` date NOT NULL,
    `email_pelanggan` varchar(51) COLLATE latin1_general_ci NOT NULL,
    `password_pelanggan` varchar(64) COLLATE latin1_general_ci NOT NULL,
    `nama_pelanggan` varchar(50) COLLATE latin1_general_ci NOT NULL,
    `alamat` text COLLATE latin1_general_ci NOT NULL,
    `id_kota` bigint(20) NOT NULL,
    `kode_pos` varchar(15) COLLATE latin1_general_ci NOT NULL,
    `telepon` varchar(20) COLLATE latin1_general_ci NOT NULL,
    `telepon_genggam` varchar(20) COLLATE latin1_general_ci DEFAULT
  NULL,
    `fax` varchar(20) COLLATE latin1_general_ci DEFAULT NULL,
    `diaktifkan` varchar(15) COLLATE latin1_general_ci NOT NULL DEFAULT
  'true',
    PRIMARY KEY (`id_pelanggan`),
    KEY `id_kota` (`id_kota`)
  ) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
  AUTO_INCREMENT=7 ;
  ALTER TABLE `pelanggan`
    ADD CONSTRAINT `pelanggan_ibfk_1` FOREIGN KEY (`id_kota`) REFERENCES
`kota` (`id_kota`);
```

4.1.3.12 Implementasi Tabel Pengaturan

Implementasi dari tabel pengaturan dapat dilihat pada Tabel IV.14 :

Tabel IV.14. Implementasi Dari Tabel Pengaturan

Field	Type	Collation	Attributes	Null	Default	Extra
id_pengaturan	bigint(20)			No	*None*	auto_increment
id_petugas	int(11)			No	*None*	
nama_pengaturan	varchar(50)	latin1_general_ci		No	*None*	
isi	longtext	latin1_general_ci		No	*None*	
diaktifkan	varchar(15)	latin1_general_ci		No	true	

```
CREATE TABLE IF NOT EXISTS `pengaturan` (
  `id_pengaturan` bigint(20) NOT NULL AUTO_INCREMENT,
  `id_petugas` int(11) NOT NULL,
  `nama_pengaturan` varchar(50) COLLATE latin1_general_ci NOT NULL,
  `isi` longtext COLLATE latin1_general_ci NOT NULL,
  `diaktifkan` varchar(15) COLLATE latin1_general_ci NOT NULL DEFAULT
'true',
  PRIMARY KEY (`id_pengaturan`),
  KEY `id_petugas` (`id_petugas`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
AUTO_INCREMENT=57 ;
ALTER TABLE `pengaturan`
  ADD CONSTRAINT `pengaturan_ibfk_1` FOREIGN KEY (`id_petugas`)
REFERENCES `petugas` (`id_petugas`);
```

4.1.3.13 Implementasi Tabel Pengiriman

Implementasi dari tabel pengiriman dapat dilihat pada Tabel IV.15 :

Tabel IV.15. Implementasi dari tabel pengiriman

Field	Type	Collation	Attributes	Null	Default	Extra
id_kirim	varchar(21)	latin1_swedish_ci		No	*None*	
id_pesanan	int(11)			No	*None*	
tgl_kirim	varchar(21)	latin1_swedish_ci		No	*None*	
tgl_terima	varchar(21)	latin1_swedish_ci		No	*None*	
status_kirim	varchar(31)	latin1_swedish_ci		No	*None*	

```
CREATE TABLE IF NOT EXISTS `pengiriman` (
  `id_kirim` varchar(21) NOT NULL,
  `id_pesanan` bigint(20) NOT NULL,
  `tgl_kirim` varchar(21) NOT NULL,
  `tgl_terima` varchar(21) NOT NULL,
  `status_kirim` varchar(31) NOT NULL,
  PRIMARY KEY (`id_kirim`),
  KEY `id_pesanan` (`id_pesanan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `pengiriman`
  ADD   CONSTRAINT   `pengiriman_ibfk_1`   FOREIGN   KEY   (`id_pesanan`)
REFERENCES `pesanan` (`id_pesanan`);
```

4.1.3.14 Implementasi Tabel Petugas

Implementasi dari tabel petugas dapat dilihat pada Tabel IV.16 :

Tabel IV.16. Implementasi Dari Tabel Petugas

Field	Type	Collation	Attributes	Null	Default	Extra
id_petugas	int(11)			No	*None*	auto_increment
tanggal_daftar	date			No	*None*	
email_petugas	varchar(51)	latin1_general_ci		No	*None*	
password_petugas	varchar(64)	latin1_general_ci		No	*None*	
nama_petugas	varchar(50)	latin1_general_ci		No	*None*	
level	int(1)			No	0	
diaktifkan	varchar(15)	latin1_general_ci		No	true	

```
CREATE TABLE IF NOT EXISTS `petugas` (
  `id_petugas` int(11) NOT NULL AUTO_INCREMENT,
  `tanggal_daftar` date NOT NULL,
  `email_petugas` varchar(51) COLLATE latin1_general_ci NOT NULL,
  `password_petugas` varchar(64) COLLATE latin1_general_ci NOT NULL,
  `nama_petugas` varchar(50) COLLATE latin1_general_ci NOT NULL,
  `level` int(1) NOT NULL DEFAULT '0',
  `diaktifkan` varchar(15) COLLATE latin1_general_ci NOT NULL DEFAULT
'true',
  PRIMARY KEY (`id_petugas`)
)  ENGINE=InnoDB     DEFAULT   CHARSET=latin1   COLLATE=latin1_general_ci
AUTO_INCREMENT=15 ;
```

4.1.3.15 Implementasi Tabel Pesanan

Implementasi dari tabel pesanan dapat dilihat pada Tabel IV.17 :

Tabel IV.17. Implementasi dari tabel pesanan

Field	Type	Collation	Attributes	Null	Default	Extra
id_pesanan	bigint(20)			No	*None*	auto_increment
id_pelanggan	bigint(20)			No	*None*	
tanggal_pesanan	date			No	*None*	
tanggal_kadaluwarsa	date			No	*None*	
nama_penerima	varchar(50)	latin1_general_ci		No	*None*	
alamat	text	latin1_general_ci		No	*None*	
kota	varchar(50)	latin1_general_ci		No	*None*	
provinsi	varchar(50)	latin1_general_ci		Yes	*NULL*	
kode_pos	varchar(15)	latin1_general_ci		No	*None*	
telepon	varchar(20)	latin1_general_ci		Yes	*NULL*	
telepon_genggam	varchar(20)	latin1_general_ci		Yes	*NULL*	
subtotal_bayar	double			No	*None*	
ongkos	double			No	*None*	
total_bayar	double			No	*None*	
status	varchar(30)	latin1_general_ci		No	order	

```
CREATE TABLE IF NOT EXISTS `pesanan` (
  `id_pesanan` bigint(20) NOT NULL AUTO_INCREMENT,
  `id_pelanggan` bigint(20) NOT NULL,
  `tanggal_pesanan` date NOT NULL,
  `tanggal_kadaluwarsa` date NOT NULL,
  `nama_penerima` varchar(50) COLLATE latin1_general_ci NOT NULL,
  `alamat` text COLLATE latin1_general_ci NOT NULL,
  `kota` varchar(50) COLLATE latin1_general_ci NOT NULL,
  `provinsi` varchar(50) COLLATE latin1_general_ci DEFAULT NULL,
  `kode_pos` varchar(15) COLLATE latin1_general_ci NOT NULL,
  `telepon` varchar(20) COLLATE latin1_general_ci DEFAULT NULL,
  `telepon_genggam` varchar(20) COLLATE latin1_general_ci DEFAULT NULL,
  `subtotal_bayar` double NOT NULL,
  `ongkos` double NOT NULL,
  `total_bayar` double NOT NULL,
  `status` varchar(30) COLLATE latin1_general_ci NOT NULL DEFAULT
'order',
  PRIMARY KEY (`id_pesanan`),
  KEY `id_pelanggan` (`id_pelanggan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci
AUTO_INCREMENT=50 ;
ALTER TABLE `pesanan`
  ADD CONSTRAINT `pesanan_ibfk_1` FOREIGN KEY (`id_pelanggan`) REFERENCES
`pelanggan` (`id_pelanggan`);
```

4.1.3.16 Implementasi Tabel Pembayaran

Implementasi dari tabel pembayaran dapat dilihat pada Tabel IV.18 :

Tabel IV.18 Implementasi dari tabel pembayaran

Field	Type	Collation	Attributes	Null	Default	Extra
id_pembayaran	varchar(21)	latin1_swedish_ci		No	None	
id_pesanan	int(11)			No	None	
jenis_bayar	varchar(14)	latin1_swedish_ci		No	None	
tgl_bayar	date			No	None	
keterangan	text	latin1_swedish_ci		No	None	

```
CREATE TABLE IF NOT EXISTS `pembayaran` (
  `id_pembayaran` varchar(21) NOT NULL,
  `id_pesanan` bigint(20) NOT NULL,
  `jenis_bayar` varchar(14) NOT NULL,
  `tgl_bayar` date NOT NULL,
  `keterangan` text NOT NULL,
  PRIMARY KEY (`id_pembayaran`),
  KEY `id_pesanan` (`id_pesanan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `pembayaran`
  ADD CONSTRAINT `pembayaran_ibfk_1` FOREIGN KEY (`id_pesanan`)
REFERENCES `pesanan` (`id_pesanan`);
```

4.1.3.17 Implementasi Tabel Rating

Implementasi dari tabel rating dapat dilihat pada Tabel IV.19 :

Tabel IV.19. Implementasi Dari Tabel Rating

Field	Type	Collation	Attributes	Null	Default	Extra
id_barang	bigint(11)			No	*None*	
total_nilai	float			No	0	
total_penilai	int(11)			No	0	

```
CREATE TABLE IF NOT EXISTS `rating` (
  `id_barang` bigint(11) NOT NULL,
  `total_nilai` float NOT NULL DEFAULT '0',
  `total_penilai` int(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id_barang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `rating`
  ADD CONSTRAINT `rating_ibfk_1` FOREIGN KEY (`id_barang`) REFERENCES
`barang` (`id_barang`);
```

4.1.3.18 Implementasi Tabel Warna

Implementasi dari tabel warna dapat dilihat pada Tabel IV.20:

Tabel IV.20. Implementasi Dari Tabel Warna

Field	Type	Collation	Attributes	Null	Default	Extra
id_warna	int(11)			No	*None*	auto_increment
nama_warna	varchar(11)	latin1_swedish_ci		No	*None*	

```
CREATE TABLE IF NOT EXISTS `warna` (
  `id_warna` int(11) NOT NULL AUTO_INCREMENT,
  `nama_warna` varchar(11) NOT NULL,
  PRIMARY KEY (`id_warna`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=9 ;
```

4.1.3.19 Implementasi Tabel Retur

Implementasi dari tabel retur dapat dilihat pada Tabel IV.21:

Tabel IV.21 implementasi tabel retur

Field	Type	Collation	Attributes	Null	Default	Extra
id_retur	int(11)			No	None	auto_increment
id_pesanan	bigint(20)			No	None	
tanggal_retur	varchar(21)	latin1_swedish_ci		No	None	
resi_retur	varchar(21)	latin1_swedish_ci		No	None	
resi_kirim	varchar(21)	latin1_swedish_ci		No	None	

```
CREATE TABLE IF NOT EXISTS `retur` (
  `id_retur` int(11) NOT NULL AUTO_INCREMENT,
  `id_pesanan` bigint(20) NOT NULL,
  `tanggal_retur` varchar(21) NOT NULL,
  `resi_retur` varchar(21) NOT NULL,
  `resi_kirim` varchar(21) NOT NULL,
  PRIMARY KEY (`id_retur`),
  KEY `id_pesanan` (`id_pesanan`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=9 ;
ALTER TABLE `retur`
  ADD CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`id_pesanan`) REFERENCES
`pesanan` (`id_pesanan`);
```

4.1.3.20 Implementasi Tabel Detail Retur

Implementasi dari tabel detail retur dapat dilihat pada Tabel IV.22:

Tabel IV.22 implementasi tabel detail retur

Field	Type	Collation	Attributes	Null	Default	Extra
id_detail_retur	int(11)			No	None	auto_increment
id_retur	int(11)			No	None	
id_detail_pesanan	bigint(20)			No	None	
jumlah	int(11)			No	None	
alasan	text	latin1_swedish_ci		No	None	

```
CREATE TABLE IF NOT EXISTS `detail_retur` (
  `id_detail_retur` int(11) NOT NULL AUTO_INCREMENT,
  `id_retur` int(11) NOT NULL,
  `id_detail_pesanan` bigint(20) NOT NULL,
  `jumlah` int(11) NOT NULL,
  `alasan` text NOT NULL,
  PRIMARY KEY (`id_detail_retur`),
  KEY `id_retur` (`id_retur`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=9 ;
ALTER TABLE `detail_retur`
  ADD   CONSTRAINT   `detail_retur_ibfk_1`   FOREIGN   KEY   (`id_retur`)
REFERENCES `retur` (`id_retur`);
```

4.1.4 Implementasi Antar Muka

Implementasi antar muka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengunjung dan member pada Tabel IV.23, operator pada Tabel IV.24, dan administrator pada Tabel IV.25. Adapun untuk tampilan implementasi pembangunan aplikasi *e-commerce* di toko Yoyo Sport akan dilampirkan pada halaman lampiran.

Tabel IV.23 Implementasi Antar Muka Pengunjung Dan Member

No	Menu	Deskripsi	Nama *File*
1	Daftar *member*	Digunakan oleh pengunjung untuk melakukan pendaftaran menjadi member baru	pelanggan.php
2	Login *member*	Digunakan oleh pelanggan untuk melakukan login member	pelanggan.php
3	Lupa *password*	Digunakan jika pelanggan lupa dengan *password*	pelanggan.php
4	Alamat pengiriman	Digunakan untuk mengganti alamat pelanggan	pelanggan.php
5	Kontak	Digunakan untuk mengganti kontak pelanggan	pelanggan.php
6	Ganti *password*	Digunakan untuk mengganti *password* pelanggan	pelanggan.php
7	Komentar	Digunakan untuk melihat barang yang telah dikomentari oleh member	pelanggan.php
8	Syarat & informasi	Digunakan untuk melihat syarat dan informasi penjualan	pelanggan.php
9	Pesanan	Digunakan untuk melihat history pesanan yang pernah dilakukan	pelanggan.php
10	Keranjang belanja	Digunakan untuk melihat, merubah, menghapus daftar pesanan yang akan dilakukan	transaksi.php
11	Biaya pengiriman	Digunakan untuk melihat data ongkos kirim	hal.php
12	Kategori	Digunakan untuk melihat barang perkategori	hal.php

13	Barang	Digunakan untuk melihat data barang dan detail barang	detail-barang.php
14	Logout	Digunakan member untuk keluar dari akunya	pelanggan.php
15	Pencarian barang	Digunakan member atau pengunjung untuk pencarian barang	hal.php

Tabel IV.24 Implementasi Antar Muka Operator

No	Menu	Deskripsi	Nama *File*
1	*Login* operator	Digunakan sebagai halaman *login* operator	login.php
2	Lupa *password*	Digunakan jika operator lupa dengan *password*	login.php
3	Ganti *password*	Digunakan untuk mengubah data *password*	profile.php
4	Data kategori	Digunakan untuk melihat, menambah, dan merubah data kategori	product_category.php
5	Data barang	Digunakan untuk melihat, menambah, dan merubah data barang	product.php
6	Laporan pesanan	Digunakan untuk melihat dan mengolah data pesanan	order.php
7	Provinsi & Kota	Digunakan untuk melihat, menambah dan merubah data provinsi, kota dan ongkos kirim	exspenses.php
8	Laporan penjualan	Digunakan untuk melihat data laporan penjualan berdasarkan jenis pembayaran berdasarkan waktu yang telah ditentukan dan cetak laporan	report.php
9	Backup & restore	Digunakan untuk melakukan backup & restore database	br.php
10	Komentar	Digunakan untuk melihat dan mengelola data komentar pada barang	feedback.php
11	Logout	Digunakan operator untuk dapat keluar dari halaman operator	login.php

Tabel IV.25 Implementasi Antar Muka Administrator

No	Menu	Deskripsi	Nama *File*

1	Login Administrator	Digunakan sebagai halaman *login* administrator	login.php
2	Lupa *password*	Digunakan jika administrator lupa dengan *password*	login.php
3	Ganti *password*	Digunakan untuk mengubah data *password*	profile.php
4	Data Petugas	Digunakan untuk melihat, menambah, dan merubah data petugas	employee.php
5	Data Pelanggan	Digunakan untuk melihat data pelanggan / member	customer.php
6	Pengaturan web	Digunakan untuk melihat,dan mengubah data pengaturan web	setting.php
7	*Logout*	Digunakan administrator untuk keluar dari halaman administrator	login.php

4.2 Pengujian Perangkat Lunak

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

Pengujian perangkat lunak ini menggunakan pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

4.2.1 Pengujian Alpha

Pengujian alpha dilakukan menggunakan metode *black box*. Untuk menentukan pengujian alpha ini dibagi menjadi tiga, yaitu skenario pengujian halaman pelanggan, skenario pengujian halaman administrator dan skenario pengujian aplikasi operator.

4.2.1.1 Skenario Pengujian Alpha

Untuk penjelasan lebih lanjut tentang skenario pengujian aplikasi pelanggan maka dapat dilihat pada Tabel IV.26 sebagai berikut :

Tabel IV.26 Skenario Pengujian Aplikasi Pelanggan

Kelas Uji	Butir Uji	Jenis Pengujian
Lupa *password*	Menampilkan *form* lupa *password*	*Black Box*
Daftar pelanggan	Menampilkan *form* daftar pelanggan	*Black Box*
	Isi *form* daftar pelanggan	*Black Box*
	Simpan data pelanggan baru	*Black Box*
Login	Isi data *login*	*Black Box*
	Verifikasi *e-mail*	*Black Box*
	Verifikasi *password*	*Black Box*
Ganti alamat utama	Menampilkan *form* ganti alamat utama	*Black Box*
	Ubah data alamat utama	*Black Box*
	Simpan perubahan	*Black Box*
Ganti kontak	Menampilkan *form* ganti kontak	*Black Box*
	Ubah data kontak	*Black Box*
	Simpan perubahan	*Black Box*
Ganti *password*	Menampilkan form ganti *password*	*Black Box*
	Isi *form* ganti *password*	*Black Box*
	Simpan data baru	*Black Box*
Komentar	Menampilkan komentar barang	*Black Box*
Syarat & Informasi	Menampilkan syarat retur dan informasi Toko Yoyo Sport	*Black Box*
Keranjang belanja	Menampilkan data pembelian	*Black Box*
	Mengubah data pembelian	*Black Box*
	Menghapus data pembelian	*Black Box*
	Menampilkan konfirmasi transaksi	*Black Box*
Isi *form* alamat	Menampilkan *form* alamat tujuan	*Black Box*

tujuan		
Isi *form* pembayaran	Menampilkan *form* pembayaran	*Black Box*
Pesanan	Menampilkan daftar pemesanan yang telah dilakukan	*Black Box*
Logout	Menghapus *session* pelanggan	*Black Box*

Untuk penjelasan lebih lanjut tentang skenario pengujian aplikasi operator maka dapat dilihat pada Tabel IV.27 sebagai berikut :

Tabel IV.27 Skenario Pengujian Aplikasi Operator

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Lupa *password*	Menampilkan *form* lupa *password*	*Black Box*
Login	Isi data *login*	*Black Box*
	Verifikasi *e-mail*	*Black Box*
	Verifikasi *password*	*Black Box*
Data kategori	Menampilkan data kategori	*Black Box*
	Tambah data kategori	*Black Box*
	Ubah data kategori	*Black Box*
Data barang	Menampilkan data barang	*Black Box*
	Tambah data barang	*Black Box*
	Ubah data barang	*Black Box*
Data detail barang	Menampilkan data detail barang	*Black Box*
	Tambah data detail barang	*Black Box*
	Ubah data detail barang	*Black Box*
Laporan pesanan	Menampilkan data pesanan	*Black Box*
	Ubah status pesanan	*Black Box*
Laporan penjualan	Menampilkan data laporan	*Black Box*

	penjualan	
	Cari data laporan penjualan	*Black Box*
	Cetak data laporan penjualan	*Black Box*
Backup & restore	Backup database	*Balak Box*
	Restore database	*Black Box*
Provinsi dan Kota	Menampilkan data ongkos kirim	*Black Box*
	Menambah data ongkos kirim	*Black Box*
	Mengubah data ongkos kirim	*Black Box*

Untuk penjelasan lebih lanjut tentang skenario pengujian aplikasi administrator maka dapat dilihat pada Tabel IV.28 sebagai berikut :

Tabel IV.28 Skenario Pengujian Aplikasi Admin

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Lupa *password*	Menampilkan *form* lupa *password*	*Black Box*
Login	Isi data *login*	*Black Box*
	Verifikasi e-mail	*Black Box*
	Verifikasi *password*	*Black Box*
Data petugas	Menampilkan data petugas	*Black Box*
	Tambah data petugas	*Black Box*
	Ubah data petugas	*Black Box*
Data pelanggan	Menampilkan data pelanggan	*Black Box*
	Ubah data pelanggan	*Black Box*
Pengaturan web	Menampilkan *form* pengaturan web	*Black Box*
	Ubah *form* pengaturan web	*Black Box*
Ganti *password*	Menampilkan *form* ganti *password*	*Black Box*
	Isi *form* ganti *password*	*Black Box*

	Simpan data baru	*Black Box*
Logout	Menghapus *session* administrator	*Black Box*

4.2.1.2 Kasus Dan Hasil Pengujian

Berdasarkan rencana pengujian yang disusun, maka dilakukan pengujian sebagai berikut.

1. Pengujian Lupa Password Pelanggan

Untuk penjelasan lebih lanjut tentang pengujian data benar lupa password pelanggan maka dapat dilihat pada Tabel IV.29 sebagai berikut :

Tabel IV.29 Pengujian Data Benar Lupa Password Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi e-mail dengan benar	Tercantum pada text box pada *field* e-mail	Dapat tercantum pada text box pada *field* e-mail	[√] Diterima [ ] Ditolak
Mengisi perhitungan dengan benar	Tercantum pada text box pada *field* perhitungan	Dapat tercantum pada text box pada *field* perhitungan	[√] Diterima [ ] Ditolak
Email : rudi.zaman@gmail .com Hitung : misal 5 + 5 = 10	Menampilkan data masukkan pada *field* email dan *field* hitung	Dapat menampilkan data masukkan pada *field* email dan *field* hitung	[√] Diterima [ ] Ditolak

Untuk penjelasan lebih lanjut tentang pengujian data salah lupa password pelanggan maka dapat dilihat pada Tabel IV.30 sebagai berikut :

Tabel IV.30 Pengujian Data Salah Lupa Password Pelanggan

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

Format email tidak valid, contoh : Email : rudi.zaman.gmail.com	Menampilkan pesan kesalahan “e-mail tidak valid”	Dapat menampilkan pesan kesalahan “e-mail tidak valid”	[√] Diterima [ ] Ditolak
Email belum terdaftar di dalam database	Menampilkan pesan kesalahan “ERROR : E-mail belum terdaftar!”	Dapat menampilkan pesan kesalahan “ERROR : E-mail belum terdaftar!”	[√] Diterima [ ] Ditolak

2. Pengujian Daftar Pelanggan

Untuk penjelasan lebih lanjut tentang pengujian data benar daftar pelanggan maka dapat dilihat pada Tabel IV.31 sebagai berikut :

Tabel IV.31 Pengujian Data Benar Daftar Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data daftar pelanggan, contoh : Email baru : rudi.zaman@gmail.com Password baru : 123456 Konfirmasi password : 123456 Nama Lengkap : Rudi Badru Zaman Alamat : Jl. Tubagus Ismail Dalam No. 42A Provinsi : Jawa Barat Kota : Bandung Kode Pos : 40132 Hitung : 8	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses registrasi	Form registrasi terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses registrasi	[√] Diterima [ ] Ditolak

Untuk penjelasan lebih lanjut tentang pengujian data salah daftar pelanggan maka dapat dilihat pada Tabel IV.32 sebagai berikut :

Tabel IV.32 Pengujian Data Salah Daftar Pelanggan

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi email yang sudah terdaftar	Tampilkan pesan kesalahan "Email sudah terdaftar! Apakah anda lupa password?"	Dapat menampilkan pesan kesalahan "Email sudah terdaftar! Apakah anda lupa password?"	[√] Diterima [ ] Ditolak
Mengisi email yang dengan format tidak valid, contoh : Email baru : rudi.zaman.gmail.com	Tampilkan pesan kesalahan "Email tidak valid"	Dapat menampilkan pesan kesalahan "Email tidak valid"	[√] Diterima [ ] Ditolak
Mengisi password baru kurang dari 6 karakter, contoh : Password Baru : 1234	Tampilkan pesan kesalahan "Password Baru" minimal 6 karakter	Dapat menampilkan pesan kesalahan "Password Baru" minimal 6 karakter	[√] Diterima [ ] Ditolak
Mengisi konfirmasi password yang berbeda dengan password baru, contoh : Password Baru : 222222 Konfirmasi Pasword : 111111	Tampilkan pesan kesalahan "Konfirmasi password" harus sama dengan "Password Baru"	Dapat menampilkan pesan kesalahan "Konfirmasi password" harus sama dengan "Password Baru"	[√] Diterima [ ] Ditolak
Format kode pos tidak sesuai, contoh : Kode Pos : abcde	Tampilkan pesan kesalahan "Kode Pos" tidak valid	Dapat menampilkan pesan kesalahan "Kode Pos" tidak valid	[√] Diterima [ ] Ditolak
Memasukkan data kosong	Menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "E-	[√] Diterima

	“E-mail dibutuhkan”, “Password Baru dibutuhkan”, “Konfirmasi Password dibutuhkan”, “Nama dibutuhkan”, “Alamat dibutuhkan”, “Provinsi dibutuhkan”, “Kota dibutuhkan”, “Kode Pos dibutuhkan”, “Perhitungan dibutuhkan”	mail dibutuhkan”, “Password Baru dibutuhkan”, “Konfirmasi Password dibutuhkan”, “Nama dibutuhkan”, “Alamat dibutuhkan”, “Provinsi dibutuhkan”, “Kota dibutuhkan”, “Kode Pos dibutuhkan”, “Perhitungan dibutuhkan”	[ ] Ditolak

3. Pengujian Login Pelanggan

Untuk penjelasan lebih lanjut tentang pengujian data benar login pelanggan maka dapat dilihat pada Tabel IV.33 sebagai berikut :

Tabel IV.33 Pengujian Data Benar Login Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan alamat *email* : rudi.zaman@gmail .com	Mengisikan data alamat *email* pada *field* alamat email	Dapat mengisi *field* alamat *email* pada *field* alamat *email*	[√] Diterima [ ] Ditolak
Memasukan Password : 123456	Mengisikan data password pada field password	Mengisikan data password pada field password	[√] Diterima [ ] Ditolak

Untuk penjelasan lebih lanjut tentang pengujian data salah login pelanggan maka dapat dilihat pada Tabel IV.34 sebagai berikut :

Tabel IV.34 Pengujian Data Salah Login Pelanggan

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

Memasukan data *email* dan *password* yang tidak sesuai	Tampilkan pesan kesalahan “ERROR E-mail atau password salah”	Dapat menampilkan pesan kesalahan “ ERROR E-mail atau password salah”	[√] Diterima [ ] Ditolak
Memasukan data *email* atau *password* yang tidak terdaftar	Tampilkan pesan kesalahan “ERROR E-mail atau password salah”	Dapat menampilkan pesan kesalahan “ ERROR E-mail atau password salah”	[√] Diterima [ ] Ditolak
Memasukan data *email* yang tidak valid, contoh : Email : rudi.zaman.gmail.com	Tampilkan pesan kesalahan “e-mail tidak valid”	Dapat menampilkan pesan kesalahan “e-mail tidak valid”	[√] Diterima [ ] Ditolak
Mengosongkan data *email* atau *password*	Tampilkan pesan kesalahan “E-mail dibutuhkan”, “password dibutuhkan”	Dapat menampilkan pesan kesalahan “E-mail dibutuhkan”, “password dibutuhkan”	[√] Diterima [ ] Ditolak

4. Pengujian Pencarian Barang

Untuk penjelasan lebih lanjut tentang pengujian data benar pencarian barang maka dapat dilihat pada Tabel IV.35 sebagai berikut :

Tabel IV.35 Pengujian Data Benar Pencarian Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data pada form menu pencarian, contoh : Nama barang : Mitre Vicous Diskon : 0 % Kategori : Aksesoris	Menampilkan barang sesuai dengan pencarian barang yang dimasukkan	Dapat menampilkan barang sesuai dengan pencarian barang yang dimasukkan	[√] Diterima [ ] Ditolak

5. Pengujian Keranjang Belanja

Untuk penjelasan lebih lanjut tentang pengujian data benar keranjang belanja maka dapat dilihat pada Tabel IV.36 sebagai berikut :

Tabel IV.36 Pengujian Data Benar Keranjang Belanja

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukkan data jumlah barang pada *record* barang yang dipilih. Misal : 3	Tercantum pada text box jumlah pada *record* barang yang dipilih	Dapat tercantum pada text box pada *record* barang yang dipilih	[√] Diterima [ ] Ditolak

Untuk penjelasan lebih lanjut tentang pengujian data salah keranjang belanja maka dapat dilihat pada Tabel IV.37 sebagai berikut :

Tabel IV.37 Pengujian Data Salah Keranjang Belanja

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Jumlah barang yang dimasukkan ke keranjang belanja melebihi stok barang. Misal : 10 (Stok barang yang tersedia 9 barang)	Mengambil stok maksimal pada stok barang dan menampilkan pesan kesalahan ”kolom jumlah yang berwarna merah merupakan batas maksimal stok”	Dapat mengambil stok maksimal pada stok barang dan dapat menampilkan pesan kesalahan ”kolom jumlah yang berwarna merah merupakan batas maksimal stok”	[√] Diterima [ ] Ditolak
Mengisi jumlah nol pada *record* barang yang dipilih	Menghapus *record* barang yang dipilih	Dapat menghapus *record* barang yang dipilih	[√] Diterima [ ] Ditolak

6. Pengujian Form Alamat Pengiriman

Untuk penjelasan lebih lanjut tentang pengujian data benar form alamat pengiriman maka dapat dilihat pada Tabel IV.38 sebagai berikut :

Tabel IV.38 Pengujian Data Benar Form Alamat Pengiriman

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Nama : Rudi Badru Zaman Alamat : Perumahan Griya Muncang Asri Q7 Provinsi : Jawa Barat Kota : Tasikmalaya Kode pos : 46121	Menampilkan data masukkan pada setiap *field* masing-masing	Dapat menampilkan data masukkan pada setiap *field* masing-masing	[√] Diterima [ ] Ditolak

Untuk penjelasan lebih lanjut tentang pengujian data salah form alamat tujuan pengiriman maka dapat dilihat pada Tabel IV.39 sebagai berikut :

Tabel IV.39 Pengujian Data Salah Form Alamat Tujuan Pengiriman

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Format kode pos tidak sesuai, contoh : Kode Pos : abcde	Menampilkan pesan kesalahan "Kode Pos tidak valid"	Dapat menampilkan pesan kesalahan "Kode Pos tidak valid"	[√] Diterima [ ] Ditolak
Memasukkan data kosong	Menampilkan pesan kesalahan "Nama	Dapat menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak

	dibutuhkan”, “Alamat dibutuhkan”, “Provinsi dibutuhkan”, “Kota dibutuhkan”, “Kode Pos dibutuhkan”.	“Nama dibutuhkan”, “Alamat dibutuhkan”, “Provinsi dibutuhkan”, “Kota dibutuhkan”, “Kode Pos dibutuhkan”.	

7. Pengujian Form Formulir Pembayaran

Untuk penjelasan lebih lanjut tentang pengujian data benar form formulir pembayaran maka dapat dilihat pada Tabel IV.40 sebagai berikut :

Tabel IV.40 Pengujian Data Benar Form Formulir Pembayaran

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi form pembayaran via transfer bank, contoh : Nama bank pengirim : BCA No rekening asal : 0108995947 Nama pemilik rekening asal : Rudi Badru Zaman Nama bank penerima : BCA a/n Rudi	Mengisi form pembayaran via transfer bank dengan benar	Dapat mengisi form pembayaran via transfer bank dengan benar	[√] Diterima [ ] Ditolak

Untuk penjelasan lebih lanjut tentang pengujian data salah form formulir pembayaran maka dapat dilihat pada Tabel IV.41 sebagai berikut :

Tabel IV.41 Pengujian Data Salah Form Formulir Pembayaran

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Format No.Rekening Pengirim tidak sesuai, contoh : No.Rekening Pengirim : abcdefgh	Menampilkan pesan kesalahan "no rekening pengirim tidak valid"	Dapat menampilkan pesan kesalahan "no rekening pengirim tidak valid"	[√] Diterima [ ] Ditolak
Format Nama Pemilik Rekening Pengirim tidak sesuai, contoh : Nama Pemilik Rekening Pengirim : 123456	Menampilkan pesan kesalahan "nama pemilik rekening pengirim tidak valid"	Dapat menampilkan pesan kesalahan "nama pemilik rekening pengirim tidak valid"	[√] Diterima [ ] Ditolak
Tidak memasukkan no rekening pengirim	Menampilkan pesan kesalahan "no rekening asal dibutuhkan"	Dapat menampilkan pesan kesalahan " no rekening asal dibutuhkan"	[√] Diterima [ ] Ditolak
Tidak memasukkan nama pemilik rekening pengirim	Menampilkan pesan kesalahan "nama pemilik rekening asal dibutuhkan"	Dapat menampilkan pesan kesalahan " nama pemilik rekening asal dibutuhkan"	[√] Diterima [ ] Ditolak

8. Pengujian Ganti Alamat Pengiriman

Untuk penjelasan lebih lanjut tentang pengujian data benar ganti alamat pengiriman maka dapat dilihat pada Tabel IV.42 sebagai berikut :

Tabel IV.42 Pengujian Data Benar Ganti Alamat Pengiriman

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah data pada form ganti alamat utama, contoh : Alamat : Asrama Polisi No.36 Provinsi : Jawa Barat Kota : Tasikmalaya Kode Pos : 40132	Mengisi *field* data yang akan diubah dengan benar	Dapat mengisi *field* data yang akan diubah dengan benar	[√] Diterima [ ] Ditolak

Untuk penjelasan lebih lanjut tentang pengujian data salah ganti alamat pengiriman maka dapat dilihat pada Tabel IV.43 sebagai berikut :

Tabel IV.43 Pengujian Data Salah Ganti Alamat Pengiriman

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Format kode pos tidak sesuai, contoh : Kode Pos : abcde	Tampilkan pesan kesalahan "Kode Pos" tidak valid	Dapat menampilkan pesan kesalahan "Kode Pos" tidak valid	[√] Diterima [ ] Ditolak
Belum mengisi data pada semua *field*	Tampilkan pesan kesalahan "Alamat dibutuhkan", "Provinsi dibutuhkan", "Kota dibutuhkan", "Kode Pos dibutuhkan"	Dapat menampilkan pesan kesalahan "Alamat dibutuhkan", "Provinsi dibutuhkan", "Kota dibutuhkan", "Kode Pos dibutuhkan"	[√] Diterima [ ] Ditolak

Belum mengisi alamat	Tampilkan pesan kesalahan "Alamat dibutuhkan"	Dapat menampilkan pesan kesalahan "Alamat dibutuhkan"	[√] Diterima [ ] Ditolak
Belum mengisi provinsi	Tampilkan pesan kesalahan "Provinsi dibutuhkan", "kota dibutuhkan"	Dapat menampilkan pesan kesalahan "Provinsi dibutuhkan", "kota dibutuhkan"	[√] Diterima [ ] Ditolak
Belum mengisi kota	Tampilkan pesan kesalahan "Kota dibutuhkan"	Dapat menampilkan pesan kesalahan "Kota dibutuhkan"	[√] Diterima [ ] Ditolak
Belum mengisi kode pos	Tampilkan pesan kesalahan "Kode pos dibutuhkan"	Dapat menampilkan pesan kesalahan "kode pos dibutuhkan"	[√] Diterima [ ] Ditolak

9. Pengujian Ganti Kontak

Untuk penjelasan lebih lanjut tentang pengujian data benar ganti kontak maka dapat dilihat pada Tabel IV.44 sebagai berikut :

Tabel IV.44 Pengujian Data Benar Ganti Kontak

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah kontak anda, contoh : Handphone : 08997726667 Telepon : 0265339438 Fax : 0265339438	Mengisi *field* data yang akan diubah dengan benar	Dapat mengisi *field* data yang akan diubah dengan benar	[√] Diterima [ ] Ditolak

10. Pengujian Ganti Password Pelanggan

Untuk penjelasan lebih lanjut tentang pengujian data benar ganti password pelanggan maka dapat dilihat pada Tabel IV.45 sebagai berikut :

Tabel IV.45 Pengujian Data Benar Ganti Password Pelanggan

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi *field* dengan benar, contoh : Password sekarang : 123456 Password baru : 654321 Konfirmasi password : 654321	Mengisi *field* dengan benar pada data yang akan dirubah	Dapat mengisi *field* dengan benar pada data yang akan dirubah	[√] Diterima [ ] Ditolak

Untuk penjelasan lebih lanjut tentang pengujian data salah ganti password pelanggan maka dapat dilihat pada Tabel IV.46 sebagai berikut :

Tabel IV.46 Pengujian Data Salah Ganti Password Pelanggan

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data password sekarang tidak sesuai dengan database	Menampilkan pesan kesalahan “password tidak valid”	Dapat menampilkan pesan kesalahan ”password tidak valid”	[√] Diterima [ ] Ditolak
Data password baru dan data konfirmasi password tidak sama	Menampilkan pesan kesalahan “konfirmasi password harus sama dengan password baru”	Dapat menampilkan pesan kesalahan ” konfirmasi password harus sama dengan password baru”	[√] Diterima [ ] Ditolak

Tidak mengisi password sekarang	Menampilkan pesan kesalahan “password dibutuhkan”	Dapat menampilkan pesan kesalahan ”password dibutuhkan”	[√] Diterima [ ] Ditolak
Tidak mengisi password baru	Menampilkan pesan kesalahan “password baru dibutuhkan”	Dapat menampilkan pesan kesalahan ”password baru dibutuhkan”	[√] Diterima [ ] Ditolak
Tidak mengisi konfirmasi password	Menampilkan pesan kesalahan “konfirmasi password dibutuhkan”	menampilkan pesan kesalahan ”konfirmasi password dibutuhkan”	[√] Diterima [ ] Ditolak
Tidak ada data yang dimasukan pada semua field	Menampilkan pesan kesalahan “password dibutuhkan”, “password baru dibutuhkan”, “konfirmasi password dibutuhkan”	Dapat menampilkan pesan kesalahan “password dibutuhkan”, “password baru dibutuhkan”, “konfirmasi password dibutuhkan”	[√] Diterima [ ] Ditolak

11. Pengujian Lupa Password Administrator Dan Operator

Untuk penjelasan lebih lanjut tentang pengujian data benar lupa password administrator dan operator maka dapat dilihat pada Tabel IV.47 sebagai berikut :

Tabel IV.47 Pengujian Data Benar Lupa Password Admin Dan Operator

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi e-mail pada *field*, contoh : rudi.zaman@hotmail.com	Mengisi *field* dengan benar	Dapat mengisi *field* dengan benar	[√] Diterima [ ] Ditolak

Untuk penjelasan lebih lanjut tentang pengujian data salah lupa password administrator dan operator maka dapat dilihat pada Table IV.48 sebagai berikut :

Tabel IV.48 Pengujian Data Salah Lupa Password Admin Dan Operator

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Format email tidak valid, contoh : Email : rudi.zaman.gmail.com	Menampilkan pesan kesalahan "e-mail tidak valid"	Dapat menampilkan pesan kesalahan "e-mail tidak valid"	[√] Diterima [ ] Ditolak
Email belum terdaftar di dalam database	Menampilkan pesan kesalahan "ERROR : E-mail belum terdaftar!"	Dapat menampilkan pesan kesalahan "ERROR : E-mail belum terdaftar!"	[√] Diterima [ ] Ditolak

12. Pengujian Login Admin Dan Operator

Untuk penjelasan lebih lanjut tentang pengujian data benar login administrator dan operator maka dapat dilihat pada Tabel IV.49 sebagai berikut :

Tabel IV.49 Pengujian Data Benar Login Admin Dan Operator

Kasus dan Hasil Uji (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukkan data login, contoh : E-mail : rudi.zaman@hotmail.com Password : 123456	Memasukkan data login dengan benar	Dapat memasukkan data login dengan benar	[√] Diterima [ ] Ditolak

Untuk penjelasan lebih lanjut tentang pengujian data salah login administrator dan operator maka dapat dilihat pada Tabel IV.50 sebagai berikut :

Tabel IV.50 Pengujian Data Salah Login Admin Dan Operator

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data *email* dan *password* yang tidak sesuai	Tampilkan pesan kesalahan "ERROR E-mail atau password salah"	Dapat menampilkan pesan kesalahan " ERROR E-mail atau password salah"	[√] Diterima [ ] Ditolak
Memasukan data *email* atau *password* yang tidak terdaftar	Tampilkan pesan kesalahan "ERROR E-mail atau password salah"	Dapat menampilkan pesan kesalahan " ERROR E-mail atau password salah"	[√] Diterima [ ] Ditolak
Memasukan data *email* yang tidak valid, contoh : Email : rudi.zaman.gmail.com	Tampilkan pesan kesalahan "e-mail tidak valid"	Dapat menampilkan pesan kesalahan "e-mail tidak valid"	[√] Diterima [ ] Ditolak
Mengosongkan data *email* atau *password*	Tampilkan pesan kesalahan "E-mail dibutuhkan", "password dibutuhkan"	Dapat menampilkan pesan kesalahan "E-mail dibutuhkan", "password dibutuhkan"	[√] Diterima [ ] Ditolak

4.2.1.3 Kesimpulan Pengujian Alpha

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

4.2.2 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan mengenai kepuasan user dengan kandungan poin yaitu pemenuhan kebutuhan dari tujuan awal pembangunan *website e-commerce* dan tampilan antarmuka dari *website e-commerce* tersebut. Pengujian beta dilakukan melalui dua teknik pengambilan data, yaitu melalui wawancara dan kuesioner.

Pengujian beta melalui wawancara dilakukan untuk halaman administrator dan halaman operator pada website e-commerce toko Yoyo Sport karena terbatasnya jumlah pengguna untuk administrator dan operator. Wawancara mengenai halaman administrator dilakukan terhadap pemilik toko Yoyo Sport, sedangkan untuk wawancara mengenai halaman operator dilakukan terhadap pegawai toko Yoyo Sport.

Pengujian beta melalui kuesioner dilakukan dengan membuat kuesioner mengenai kepuasan user dengan kandungan poin syarat user friendly untuk selanjutnya dibagikan kepada sebagian user dengan mengambil sample sebanyak 10 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

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

1. Lingkungan penjualan di toko Yoyo Sport

Penelitian di lingkungan penjual di toko Yoyo Sport dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan di awal.

2. Lingkungan masyarakat umum

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

4.2.2.1 Wawancara Pengguna

Wawancara dilakukan dengan menggunakan teknik kualitatif melalui pertanyaan terbuka. Wawancara mengenai halaman administrator dilakukan kepada 1 orang yaitu pemilik toko Yoyo Sport, sedangkan untuk wawancara mengenai halaman operator dilakukan terhadap 1 orang yaitu pegawai toko Yoyo Sport. Dari hasil wawancara tersebut akan dilakukan pengambilan kesimpulan terhadap penilaian penerapan sistem yang baru.

Wawancara terdiri dari 5 pertanyaan terbuka yang akan ditanyakan kepada narasumber.

4.2.2.2 Wawancara Pengujian Beta Untuk Administrator

Wawancara dilakukan untuk pengujian beta terhadap *website e-commerce* toko Yoyo Sport kepada pemilik toko dengan total narasumber sebanyak 1 orang. Berikut daftar pertanyaan yang diajukan kepada narasumber :

1. Menurut Anda, apakah halaman administrator pada *website e-commerce* toko Yoyo Sport ini sudah mampu memenuhi kebutuhan Anda dalam pengolahan data ?

2. Menurut pendapat Anda, apakah cara pemasaran barang yang diterapkan di *website e-commerce* ini dapat memperluas cakupan pemasaran toko Yoyo Sport ?
3. Menurut pendapat Anda, apakah halaman administrator dalam *website* ini dapat membantu Anda dalam mengelola data pelanggan ?
4. Menurut pendapat Anda, apakah halaman administrator dalam *website* ini dapat membantu Anda dalam mengelola data operator ?
5. Menurut pendapat Anda, apakah fungsionalitas dari halaman administrator ini mudah dipahami dan digunakan / *User friendly* ?

Berikut ini adalah masing-masing jawaban dari pertanyaan hasil wawancara tersebut .

1. Menurut Anda, apakah halaman administrator pada *website e-commerce* toko Yoyo Sport ini sudah mampu memenuhi kebutuhan Anda dalam pengolahan data ?

   Jawaban :

Sudah mampu, karena dengan adanya halaman administrator ini saya merasa terbantu untuk mengolah data pegawai toko.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman administrator pada website e-commerce toko Yoyo Sport sudah mampu memenuhi kebutuhan administrator dalam mengolah data.

2. Menurut pendapat Anda, apakah cara pemasaran barang yang diterapkan di *website e-commerce* ini dapat memperluas cakupan pemasaran toko Yoyo Sport ?

   Jawaban :

Dapat, karena konsumen yang jauh dari toko pun dapat mengakses *website* Yoyo Sport sehingga secara otomatis dapat memperluas cakupan pemasaran toko Yoyo Sport.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa cara pemasaran barang yang diterapkan di *website e-commerce* ini dapat memperluas cakupan pemasaran toko Yoyo Sport.

3. Menurut pendapat Anda, apakah halaman administrator dalam *website* ini dapat membantu Anda dalam mengelola data pelanggan ?

   Jawaban :

Sebenarnya selama ini data pelanggan tidak dikelola di sini, karena pembeli setiap harinya berbeda-beda dan kami pihak toko memang tidak mengelola hal tersebut, tapi dengan adanya *website* ini yang menyediakan pengelolaan data pelanggan, kami dapat mengetahui jumlah pelanggan toko Yoyo Sport dan memudahkan dalam pengelolaannya.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman administrator dalam *website* ini dapat membantu administrator dalam mengelola data pelanggan.

4. Menurut pendapat Anda, apakah halaman administrator dalam *website* ini dapat membantu Anda dalam mengelola data operator ?

   Jawaban :

Dapat, dengan adanya *website* ini saya dapat dengan mudah mengelola data petugas yang menangani penjualan melalui *website*.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman administrator dalam *website* ini dapat membantu administrator dalam mengelola data operator.

5. Menurut pendapat Anda, apakah fungsionalitas dari halaman administrator ini mudah dipahami dan digunakan / *User friendly* ?

   Jawaban :

Secara keseluruhan halaman administrator ini mudah dipahami dan digunakan sehingga mempermudah saya dalam melakukan proses pengelolaan data.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa fungsionalitas dari halaman administrator ini mudah dipahami dan digunakan / *User friendly*.

4.2.2.3 Wawancara Pengujian Beta Untuk Operator

Wawancara dilakukan untuk pengujian beta terhadap *website e-commerce* toko Yoyo Sport kepada pegawai toko dengan total narasumber sebanyak 1 orang. Berikut daftar pertanyaan yang diajukan kepada narasumber :

1. Menurut anda apakah halaman operator pada *website e-commerce* toko Yoyo Sport ini sudah mampu memenuhi kebutuhan Anda dalam pengelolaan data ?
2. Menurut pendapat Anda, apakah halaman operator dalam *website* ini dapat membantu Anda dalam mengelola data barang ?
3. Menurut pendapat Anda, apakah halaman aperator dalam *website* ini dapat membantu Anda dalam mengelola data penjualan secara *online* ?
4. Menurut pendapat Anda, apakah halaman aperator dalam *website* ini dapat membantu Anda dalam pembuatan laporan penjualan ?
5. Menurut pendapat Anda, apakah fungsionalitas dari halaman operator ini mudah dipahami dan digunakan / *User friendly* ?

Berikut ini adalah masing-masing jawaban dari pertanyaan hasil wawancara tersebut . Wawancara ini dilakukan kepada 1 orang.

1. Menurut anda apakah halaman operator pada *website e-commerce* toko Yoyo Sport ini sudah mampu memenuhi kebutuhan Anda dalam pengelolaan data ?

   Jawaban :

Sudah, karena dengan adanya halaman operator ini saya merasa terbantu untuk mengelola berbagai macam data yang ada di toko yang sebelumnya belum sepenuhnya menggunakan komputer.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman operator pada *website e-commerce* toko Yoyo Sport ini sudah mampu memenuhi kebutuhan operator dalam pengelolaan data.

2. Menurut pendapat Anda, apakah halaman operator dalam *website* ini dapat membantu Anda dalam mengelola data barang ?

   Jawaban :

Dapat, karena data barang di toko Yoyo Sport jumlahnya cukup banyak maka dengan adanya *website* ini sangat membantu dalam proses pengelolaan datanya.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman operator dalam *website* ini dapat membantu operator dalam mengelola data barang.

3. Menurut pendapat Anda, apakah halaman operator dalam *website* ini dapat membantu Anda dalam mengelola data penjualan secara *online* ?

   Jawaban :

Dapat, halaman operator dalam website ini sangat membantu dalam mengelola data penjualan secara *online*.

Berdasarkan hasil prosentase diatas maka dapat disimpulkan bahwa halaman operator dalam *website* ini dapat membantu operator dalam mengelola data penjualan secara *online*.

4. Menurut pendapat Anda, apakah halaman operator dalam *website* ini dapat membantu Anda dalam pembuatan laporan penjualan ?

   Jawaban :

Ya, dengan adanya *website* ini pembuatan laporan penjualan menjadi lebih mudah.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa halaman aperator dalam *website* ini dapat membantu operator dalam pembuatan laporan penjualan.

5. Menurut pendapat Anda, apakah fungsionalitas dari halaman operator ini mudah dipahami dan digunakan / *User friendly* ?

   Jawaban :

Secara keseluruhan saya tidak bayak menemukan kesulitan dalam mengoprasikan halaman operator ini.

Berdasarkan hasil wawancara diatas maka dapat disimpulkan bahwa fungsionalitas dari halaman operator ini mudah dipahami dan digunakan / *User friendly*.

4.2.2.4 Kuesioner 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. Kuisioner ini terdiri dari 5 pertanyaan (contoh kuisioner dapat dilihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 5, dengan ketentuan skala untuk tiap pertanyaan sebagai berikut :

4.2.2.5 Kuesioner Pengujian Beta Untuk Pengguna

1. Menurut anda, apakah *website e-commerce* toko Yoyo Sport ini sudah mampu memenuhi keinginan pelanggan dalam pelayanan transaksi jual-beli secara *online* ?

A. Sangat Mampu D. Kurang Mampu

B. Mampu E. Tidak Mampu

C. Cukup Mampu

2. Apakah anda setuju dengan adanya *website e-commerce* ini pemilik dapat memperluas cakupan pemasarannya ?

   A. Sangat Setuju D. Kurang Setuju

   B. Setuju E. Tidak Setuju

   C. Cukup Setuju

3. Apakah anda setuju dengan adanya *website e-commerce* ini dapat memudahkan anda untuk melakukan pembelian barang tanpa harus datang langsung ke toko Yoyo Sport ?

   A. Sangat Setuju D. Kurang Setuju

   B. Setuju E. Tidak Setuju

   C. Cukup Setuju

4. Apakah anda setuju bahwa *website e-commerce* ini dapat membantu anda untuk mendapatkan informasi tentang produk terbaru yang ada pada toko Yoyo Sport ?

   A. Sangat Setuju D. Kurang Setuju

   B. Setuju E. Tidak Setuju

   C. Cukup Setuju

5. Apakah anda setuju bahwa tampilan *website e-commerce* toko Yoyo Sport cukup menarik dan mudah dipahami / *User friendly* ?

   A. Sangat Setuju D. Kurang Setuju

B. Setuju E. Tidak Setuju

C. Cukup Setuju

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

Keterangan :

P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai persentase

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

1. Menurut anda, apakah *website e-commerce* toko Yoyo Sport ini sudah mampu memenuhi keinginan pelanggan dalam pelayanan transaksi jual-beli secara *online* ?

Tabel IV.51 Hasil Pengujian Kuisioner Soal Nomor 1

Pertanyaan	**No.**	**Keterangan**	**Responden**	**Prosentase (%)**
1	1	Sangat Mampu	3	30
	2	Mampu	6	60
	3	Cukup Mampu	1	10
	4	Kurang Mampu	0	0
	5	Tidak Mampu	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase pada Tabel IV.51 maka dapat disimpulkan sebanyak 3 atau 30% menyatakan sangat mampu, 6 atau 60% menyatakan mampu dan 1 atau 10% menyatakan cukup mampu bahwa *website e-commerce* toko

Yoyo Sport ini sudah mampu memenuhi keinginan pelanggan dalam pelayanan transaksi jual-beli secara *online*.

2. Apakah anda setuju dengan adanya *website e-commerce* ini pemilik dapat memperluas cakupan pemasarannya ?

Tabel IV.52 Hasil Pengujian Kuisioner Soal Nomor 2

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	3	30
	2	Setuju	6	60
	3	Cukup Setuju	1	10
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase pada Tabel IV.52 maka dapat disimpulkan sebanyak 3 orang atau 30% menyatakan sangat setuju, 6 orang atau 60% menyatakan setuju dan 1 orang atau 10% menyatakan cukup setuju bahwa dengan adanya *website e-commerce* ini pemilik dapat memperluas cakupan pemasarannya.

3. Apakah anda setuju dengan adanya *website e-commerce* ini dapat memudahkan anda untuk melakukan pembelian barang tanpa harus datang langsung ke toko Yoyo Sport ?

Tabel IV.53 Hasil Pengujian Kuisioner Soal Nomor 3

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

Berdasarkan hasil prosentase pada Tabel IV.53 maka dapat disimpulkan sebanyak 2 atau 20% menyatakan sangat setuju dan 6 orang atau 60% menyatakan setuju dan, 2 atau 20% menyatakan cukup setuju bahwa *website e-commerce* ini dapat mempermudah pelanggan dalam melakukan pembelian barang tanpa harus datang langsung ke toko Yoyo Sport.

4. Apakah anda setuju bahwa *website e-commerce* ini dapat membantu anda untuk mendapatkan informasi tentang produk terbaru yang ada pada toko Yoyo Sport ?

Tabel IV.54 Hasil Pengujian Kuisioner Soal Nomor 4

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

Berdasarkan hasil prosentase pada Tabel IV.54 maka dapat disimpulkan sebanyak 6 orang atau 60% menyatakan sangat setuju dan 4 orang atau 40% menyatakan setuju bahwa *website e-commerce* ini dapat membantu untuk mendapatkan informasi tentang produk terbaru yang ada pada toko Yoyo Sport.

5. Apakah anda setuju bahwa tampilan *website e-commerce* toko Yoyo Sport cukup menarik dan mudah dipahami / *User friendly* ?

Tabel IV.55 Hasil Pengujian Kuisioner Soal Nomor 5

Pertanyaan	**No.**	**Keterangan**	**Responden**	**Prosentase (%)**
5	1	Sangat Setuju	3	40
	2	Setuju	7	60

	3	Cukup Setuju	0	0
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase pada Tabel IV.55 maka dapat disimpulkan sebanyak 3 orang atau 30% menyatakan sangat setuju dan 7 orang atau 70% menyatakan bahwa tampilan *website e-commerce* toko Yoyo Sport cukup menarik dan mudah dipahami / *User friendly*.

4.2.2.6 Kesimpulan Pengujian Beta

Berdasarkan hasil wawancara dan kuesioner yang didapatkan dari pengujian beta pengguna administrator, operator dan pengguna, maka sistem penerapan e-commerce untuk sistem penjualan ini sudah sesuai dengan tujuan yaitu untuk memenuhi tuntutan ekonomi digital sehingga dapat meningkatkan pelayanan kepada konsumen, memperluas cakupan pemasaran toko Yoyo Sport, memudahkan konsumen untuk melakukan pembelian barang tanpa harus datang langsung ke toko Yoyo Sport dan mempermudah toko Yoyo Sport dalam mempromosikan produk-produk kepada konsumen tanpa harus membangun cabang-cabang baru yang memakan banyak biaya.