BAB 4

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan implementasi dan pengujian terhadap sistem yang baru. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemrograman yang akan digunakan. Setelah implementasi maka dilakukan pengujian sistem yang baru dimana akan dilihat kekurangan-kekurangan pada aplikasi yang baru untuk pengembangan sistem selanjutnya.

4.1. Implementasi

Setelah sistem dianalisis dan didesain secara rinci, selanjutnya akan menuju tahap implementasi.

Tujuan implementasi adalah untuk mengkonfirmasikan modul program perancangan pada para pelaku sistem sehingga pengguna dapat memberi masukan kepada pembangun sistem.

4.1.1 Kebutuhan Perangkat Keras

Perangkat keras yang diperlukan untuk mengimplementasikan perangkat lunak dari Sistem Informasi Pengolahan Data Pesanan Dan *Inventory Control* Berbasis *Client-Server* Pada CV.Bali67 *Advertising*, antara lain:

Tabel 4.1 Perangkat keras yang digunakan

Perangkat	Server	Client
Processor	Kecepatan 3 GHz	Kecepatan 2GHz
RAM	4 GHz	2 GHz
Harddisk	500 Gb	300 Gb
Monitor	Resolusi 1360 x 768	Resolusi 1360 x 768
VGA	256 Mb	256 Mb
Lan Card	10/100 Mbps	10/100 Mbps

4.1.2 Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan untuk mengimplementasikan Sistem Informasi Pengolahan Data Pesanan Dan *Inventory Control* Berbasis *Client-Server* Pada CV.Bali67 *Advertising*:

Tabel 4.2 Perangkat lunak yang digunakan

Perangkat Lunak	Keterangan
Sistem Operasi	Windows 7 Ultimate
Bahasa Pemrograman	Borland Delphi XE
Database	MySQL Server 5
DFD Modeler	Visio 2007

4.1.3 Impelemtasi Basis Data

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

1. Tabel Jabatan

```
CREATE TABLE `jabatan` (
 `id_jabatan` smallint(3) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `nama_jabatan` varchar(20) NOT NULL,
 `spesifikasi` varchar(80) NOT NULL,
 PRIMARY KEY (`id_jabatan`),
 UNIQUE KEY `spesifikasi` (`spesifikasi`),
 UNIQUE KEY `nama_jabatan` (`nama_jabatan`),
  KEY `idx_idjabatan` (`id_jabatan`),
  KEY `idx_namajabatan` (`nama_jabatan`),
  KEY `idx_spesifikasi` (`spesifikasi`)
```

```
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1;
```

2. TabelPegawai

```
CREATE TABLE `pegawai` (
`id_pegawai` smallint(4) unsigned zerofill NOT NULL AUTO_INCREMENT,
`id_jabatan` smallint(3) unsigned zerofill NOT NULL,
`nama_pegawai` varchar(20) NOT NULL,
`jenis_kelamin` varchar(2) DEFAULT NULL,
 `alamat` varchar(200) NOT NULL,
`no_telp` varchar(20) NOT NULL,
 `email` varchar(30) DEFAULT NULL,
`username` varchar(10) NOT NULL,
`password` varchar(10) NOT NULL,
`status_pegawai` varchar(10) NOT NULL,
PRIMARY KEY (`id_pegawai`),
KEY `idx_idpegawai` (`id_pegawai`),
KEY `idx_namapegawai` (`nama_pegawai`),
 KEY `idx_username` (`username`),
KEY `idx_statuspegawai` (`status_pegawai`),
KEY `fk_idjabatan` (`id_jabatan`),
CONSTRAINT `fk_idjabatan` FOREIGN KEY (`id_jabatan`) REFERENCES `jabatan`
(`id_jabatan`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=latin1;
```

3. Tabel Jenis Pelanggan

```
CREATE TABLE `jenis_pelanggan` (
 `id_jenis_Pelanggan` smallint(3) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `nama_jenis_pelanggan` varchar(50) NOT NULL,
 `diskon` float(4,1) unsigned zerofill NOT NULL,
 `keterangan` varchar(200) DEFAULT NULL,
 PRIMARY KEY (`id_jenis_Pelanggan`),
 KEY `idx_jenis_Pelanggan` (`nama_jenis_pelanggan`)
) ENGINE=InnoDB AUTO_INCREMENT=9 DEFAULT CHARSET=latin1;
```

4. Tabel Pelanggan

```
CREATE TABLE `pelanggan` (
 `id_pelanggan` smallint(5) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `id_jenis_pelanggan` smallint(3) unsigned zerofill NOT NULL,
 `nama_pelanggan` varchar(20) NOT NULL,
```

```
 `jenis_kelamin` varchar(2) DEFAULT NULL,
 `alamat` varchar(200) DEFAULT NULL,
 `no_telp` varchar(15) DEFAULT NULL,
 `email` varchar(30) DEFAULT NULL,
 `status` varchar(20) NOT NULL,
 `keterangan` varchar(100) DEFAULT NULL,
 PRIMARY KEY (`id_pelanggan`),
 KEY `fk_id_jenis_pelanggan` (`id_jenis_pelanggan`),
 KEY `idx_nama_pelanggan` (`nama_pelanggan`),
 KEY `idx_id_pelanggan` (`id_pelanggan`),
 CONSTRAINT `fk_id_jenis_pelanggan` FOREIGN KEY (`id_jenis_pelanggan`)
REFERENCES `jenis_pelanggan` (`id_jenis_Pelanggan`) ON DELETE CASCADE ON
UPDATE CASCADE
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=latin1;
```

5. TabelPemasok

```
CREATE TABLE `pemasok` (
 `id_pemasok` smallint(3) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `nama_pemasok` varchar(20) NOT NULL,
 `alamat` varchar(200) NOT NULL,
 `no_telp` varchar(20) NOT NULL,
 `email` varchar(30) DEFAULT NULL,
 `status` varchar(20) DEFAULT NULL,
 PRIMARY KEY (`id_pemasok`),
 KEY `idx_idpemasok` (`id_pemasok`),
 KEY `idx_namapemasok` (`nama_pemasok`)
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=latin1;
```

6. TabelKategori

```
CREATE TABLE `kategori` (
 `id_kategori` smallint(3) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `nama_kategori` varchar(30) NOT NULL,
 `keterangan` varchar(100) DEFAULT NULL,
 PRIMARY KEY (`id_kategori`),
 KEY `idx_namakategori` (`nama_kategori`) USING BTREE,
 KEY `idx_idkategori` (`id_kategori`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=latin1;
```

7. Tabel Jenis Produk

```
CREATE TABLE `jenis_produk` (
 `id_jenis_produk` smallint(3) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `id_kategori` smallint(3) unsigned zerofill NOT NULL,
 `nama_jenis_produk` varchar(50) NOT NULL,
 `keterangan` varchar(100) DEFAULT NULL,
 PRIMARY KEY (`id_jenis_produk`),
 KEY `idx_id_jenis_produk` (`id_jenis_produk`),
 KEY `idx_nama_produk` (`nama_jenis_produk`),
 KEY `fk_id_kategori` (`id_kategori`),
 CONSTRAINT `fk_id_kategori` FOREIGN KEY (`id_kategori`) REFERENCES
`kategori` (`id_kategori`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1;
```

8. TabelProduk

```
CREATE TABLE `produk` (
 `id_produk` smallint(4) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `id_jenis_produk` smallint(3) unsigned zerofill NOT NULL,
 `nama_produk` varchar(100) NOT NULL,
 `harga_pesan` float(20,0) NOT NULL,
 `gambar` mediumblob,
 `Ukuran_Produk` varchar(20) NOT NULL,
 `status_produk` varchar(20) NOT NULL,
 `keterangan` varchar(100) DEFAULT NULL,
 PRIMARY KEY (`id_produk`),
 KEY `idx_idproduk` (`id_produk`),
 KEY `idx_idjenisproduk` (`id_jenis_produk`),
 KEY `idx_namaproduk` (`nama_produk`),
 KEY `idx_statusproduk` (`status_produk`),
 CONSTRAINT `FFKIdJenisProduk` FOREIGN KEY (`id_jenis_produk`)
REFERENCES `jenis_produk` (`id_jenis_produk`) ON DELETE CASCADE ON
UPDATE CASCADE
) ENGINE=InnoDB AUTO_INCREMENT=9 DEFAULT CHARSET=latin1;
```

9. TabelSatuan

```
CREATE TABLE `satuan` (
 `id_satuan` smallint(3) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `unit_satuan` varchar(5) NOT NULL,
 `nama_satuan` varchar(20) NOT NULL,
 PRIMARY KEY (`id_satuan`),
```

```
 KEY `idx_idsatuan` (`id_satuan`),
 KEY `idx_unitsatuan` (`unit_satuan`),
 KEY `idx_namasatuan` (`nama_satuan`)
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=latin1;
```

10. Tabel Bahan Baku

```
CREATE TABLE `bahan_baku` (
 `id_bahan_baku` smallint(4) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `id_satuan` smallint(3) unsigned zerofill NOT NULL,
 `nama_bahan_baku` varchar(50) NOT NULL,
 `harga_beli` float(20,0) NOT NULL,
 `minimal_order` float(10,0) NOT NULL,
 `stok` decimal(10,2) NOT NULL,
 `keterangan` varchar(100) DEFAULT NULL,
 PRIMARY KEY (`id_bahan_baku`),
 KEY `idx_idbahanbaku` (`id_bahan_baku`),
 KEY `idx_idsatuan` (`id_satuan`),
 KEY `idx_namabahanbaku` (`nama_bahan_baku`),
 KEY `idx_hargabeli` (`harga_beli`),
 KEY `idx_stok` (`stok`),
 CONSTRAINT `fk_idsatuan` FOREIGN KEY (`id_satuan`) REFERENCES `satuan`
(`id_satuan`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB AUTO_INCREMENT=11 DEFAULT CHARSET=latin1;
```

11. Tabel BoM

```
CREATE TABLE `bom` (
 `No_Item_BoM` mediumint(6) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `Id_Produk` smallint(4) unsigned zerofill NOT NULL,
 `Id_Pegawai` smallint(4) unsigned zerofill NOT NULL,
 PRIMARY KEY (`No_Item_BoM`),
 UNIQUE KEY `Id_Produk` (`Id_Produk`),
 KEY `Id_Pegawai` (`Id_Pegawai`),
 KEY `idx_NoitemBoM` (`No_Item_BoM`) USING BTREE,
 KEY `idx_idproduk` (`Id_Produk`) USING BTREE,
 CONSTRAINT `Fk_IdPegawaiBom` FOREIGN KEY (`Id_Pegawai`) REFERENCES
`pegawai` (`id_pegawai`) ON DELETE CASCADE ON UPDATE CASCADE,
 CONSTRAINT `FK_IdProduk` FOREIGN KEY (`Id_Produk`) REFERENCES
`produk` (`id_produk`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=latin1
ROW_FORMAT=COMPACT;
```

12. Tabel Detail BoM

```
CREATE TABLE `detail_bom` (
 `No_Item_BoM` mediumint(6) unsigned zerofill NOT NULL,
 `Id_Bahan_Baku` smallint(4) unsigned zerofill NOT NULL,
 `Level` smallint(4) NOT NULL,
 `Qty_Per_Unit` mediumint(5) NOT NULL,
 `Jumlah_Komposisi` decimal(10,2) unsigned zerofill NOT NULL,
 `Tipe` varchar(10) NOT NULL,
 `Keterangan` varchar(100) DEFAULT NULL,
 KEY `idx_NoItemBoM` (`No_Item_BoM`),
 KEY `idx_idbahanbaku` (`Id_Bahan_Baku`),
 KEY `idx_tipe` (`Tipe`),
 CONSTRAINT `FK_IdBahanBaku` FOREIGN KEY (`Id_Bahan_Baku`)
REFERENCES `bahan_baku` (`id_bahan_baku`) ON DELETE CASCADE ON
UPDATE CASCADE,
 CONSTRAINT `FK_NoItemBom` FOREIGN KEY (`No_Item_BoM`) REFERENCES
`bom` (`No_Item_BoM`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

13. Tabel EOQ

```
CREATE TABLE `eoq` (
 `No_EOQ` bigint(10) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `id_pegawai` smallint(4) unsigned zerofill NOT NULL,
 `Status` varchar(30) DEFAULT NULL,
 `Total_Biaya_EOQ` float(20,0) DEFAULT NULL,
 PRIMARY KEY (`No_EOQ`),
 KEY `FKEOQIdPegawai` (`id_pegawai`),
 CONSTRAINT `FKEOQIdPegawai` FOREIGN KEY (`id_pegawai`) REFERENCES
`pegawai` (`id_pegawai`)
) ENGINE=InnoDB AUTO_INCREMENT=42 DEFAULT CHARSET=latin1;
```

14. Tabel Detail EOQ

```
CREATE TABLE `detail_eoq` (
 `No_EOQ` bigint(10) unsigned zerofill NOT NULL,
 `No_Faktur_Pemesanan` bigint(10) unsigned zerofill NOT NULL,
 `No_Item_BoM` mediumint(6) unsigned zerofill NOT NULL,
 `Id_Bahan_Baku` smallint(4) unsigned zerofill NOT NULL,
 `Jumlah_Kebutuhan` decimal(10,2) DEFAULT NULL,
```

```
 `Biaya_Pesan` float(20,0) DEFAULT NULL,
 `Biaya_Simpan` float(20,0) DEFAULT NULL,
 `Jumlah_eoq` decimal(10,2) DEFAULT NULL,
 `Frekuensi_Pemesanan` decimal(5,2) DEFAULT NULL,
 `Periode_Pemesanan` decimal(5,2) DEFAULT NULL,
 `Persediaan` float(5,0) DEFAULT NULL,
 `Jumlah_Biaya_Simpan` float(20,0) DEFAULT NULL,
 `Jumlah_Biaya_Pesan` float(20,0) DEFAULT NULL,
 `Total_Biaya` float(20,0) DEFAULT NULL,
 KEY `FKNoEOQ` (`No_EOQ`),
 KEY `FKDetailNoItemBOM` (`No_Item_BoM`),
 KEY `FKNoFakPemesanan` (`No_Faktur_Pemesanan`),
 KEY `FKIdBB` (`Id_Bahan_Baku`),
 CONSTRAINT `FKDetailNoItemBOM` FOREIGN KEY (`No_Item_BoM`)
REFERENCES `bom` (`No_Item_BoM`) ON DELETE CASCADE ON UPDATE
CASCADE,
 CONSTRAINT `FKIdBB` FOREIGN KEY (`Id_Bahan_Baku`) REFERENCES
`detail_bom` (`Id_Bahan_Baku`) ON DELETE CASCADE ON UPDATE CASCADE,
 CONSTRAINT `FKNoEOQ` FOREIGN KEY (`No_EOQ`) REFERENCES `eoq`
(`No_EOQ`) ON DELETE CASCADE ON UPDATE CASCADE,
 CONSTRAINT `FKNoFakPemesanan` FOREIGN KEY (`No_Faktur_Pemesanan`)
REFERENCES `pesanan_pelanggan` (`No_Faktur_Pemesanan`) ON DELETE
CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

15. Tabel Pesanan Pelanggan

```
CREATE TABLE `pesanan_pelanggan` (
 `No_Faktur_Pemesanan` bigint(10) unsigned zerofill NOT NULL
AUTO_INCREMENT,
 `Status` varchar(20) NOT NULL,
 `Tanggal_Bukti` date NOT NULL,
 `Id_Pegawai` smallint(4) unsigned zerofill NOT NULL,
 `Id_Pelanggan` smallint(4) unsigned zerofill NOT NULL,
 `DP` float(20,0) DEFAULT NULL,
 `Total` float(20,0) DEFAULT NULL,
 `Keterangan_Pesanan` varchar(300) DEFAULT NULL,
 PRIMARY KEY (`No_Faktur_Pemesanan`),
 KEY `Fk_IdPelanggan` (`Id_Pelanggan`),
 KEY `Fk_IdPegawai` (`Id_Pegawai`),
 CONSTRAINT `Fk_IdPegawai` FOREIGN KEY (`Id_Pegawai`) REFERENCES
`pegawai` (`id_pegawai`) ON DELETE CASCADE ON UPDATE CASCADE,
```

```
 CONSTRAINT `Fk_IdPelanggan` FOREIGN KEY (`Id_Pelanggan`) REFERENCES
`pelanggan` (`id_pelanggan`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=latin1;
```

16. Tabel Detail Pesanan Pelanggan

```
CREATE TABLE `detail_pesanan_pelanggan` (
 `No_Faktur_Pemesanan` bigint(10) unsigned zerofill NOT NULL,
 `id_produk` smallint(4) unsigned zerofill NOT NULL,
 `Jumlah_Pesan` smallint(5) NOT NULL,
 `Diskon` float(3,0) NOT NULL,
 `Total_Harga` float(20,0) NOT NULL,
 `Subtotal` float(20,0) DEFAULT NULL,
 `P` decimal(5,2) DEFAULT NULL,
 `L` decimal(5,2) DEFAULT NULL,
 `Luas_Ukuran` decimal(5,2) DEFAULT NULL,
 `Total_Bayar` float(20,0) DEFAULT NULL,
 `Keterangan` varchar(200) DEFAULT NULL,
 KEY `FK_IdProdukPemesanan` (`id_produk`),
 KEY `FK_NoFakturDetailPemesanan` (`No_Faktur_Pemesanan`),
 CONSTRAINT `FK_IdProdukPesanan` FOREIGN KEY (`id_produk`) REFERENCES
`produk` (`id_produk`) ON DELETE CASCADE ON UPDATE CASCADE,
 CONSTRAINT `FK_NoFakturDetailPemesanan` FOREIGN KEY
(`No_Faktur_Pemesanan`) REFERENCES `pesanan_pelanggan`
(`No_Faktur_Pemesanan`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

17. Tabel Produksi

```
CREATE TABLE `produksi` (
 `No_Produksi` bigint(10) unsigned zerofill NOT NULL AUTO_INCREMENT,
 `No_EOQ` bigint(10) unsigned zerofill NOT NULL,
 `Id_Pegawai` smallint(4) unsigned zerofill NOT NULL,
 `Letak_File_Desain` varchar(500) DEFAULT NULL,
 `Letak_File_Cetak` varchar(500) DEFAULT NULL,
 `Status_Produksi` varchar(50) DEFAULT NULL,
 `Tanggal_Produksi` date DEFAULT NULL,
 `Tanggal_Selesai_Produksi` date DEFAULT NULL,
 `Keterangan_Produksi` varchar(300) DEFAULT NULL,
 PRIMARY KEY (`No_Produksi`),
 KEY `FK1` (`No_EOQ`),
 KEY `FK2` (`Id_Pegawai`),
```

```
 CONSTRAINT `FK1` FOREIGN KEY (`No_EOQ`) REFERENCES `eoq`
(`No_EOQ`) ON DELETE CASCADE ON UPDATE CASCADE,
 CONSTRAINT `FK2` FOREIGN KEY (`Id_Pegawai`) REFERENCES `pegawai`
(`id_pegawai`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=latin1;
```

18. Tabel Pembelian Bahan Baku

```
CREATE TABLE `pembelian_bahan_baku` (
 `No_Faktur_Pembelian` bigint(10) unsigned zerofill NOT NULL,
 `Tanggal_Pembelian` date DEFAULT NULL,
 `Id_Pemasok` smallint(4) unsigned zerofill NOT NULL,
 `Total` float(20,0) DEFAULT NULL,
 `Status` varchar(30) DEFAULT NULL,
 `Id_Pegawai` smallint(4) unsigned zerofill NOT NULL,
 PRIMARY KEY (`No_Faktur_Pembelian`),
 KEY `FK2Pembelian` (`Id_Pegawai`),
 KEY `FK1Pembelian` (`Id_Pemasok`),
 CONSTRAINT `FK1Pembelian` FOREIGN KEY (`Id_Pemasok`) REFERENCES
`pemasok` (`id_pemasok`) ON UPDATE CASCADE,
 CONSTRAINT `FK2Pembelian` FOREIGN KEY (`Id_Pegawai`) REFERENCES
`pegawai` (`id_pegawai`) ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

19. Tabel Detail Pembelian Bahan Baku

```
CREATE TABLE `detail_pembelian_bahan_baku` (
 `No_Faktur_Pembelian` bigint(10) unsigned zerofill NOT NULL,
 `No_EOQ` bigint(10) unsigned zerofill NOT NULL,
 `Jumlah` smallint(6) DEFAULT NULL,
 `Subtotal` float(20,0) DEFAULT NULL,
 `Total_bayar` float(20,0) DEFAULT NULL,
 `Cara_Bayar` varchar(10) DEFAULT NULL,
 `Diskon` float(5,0) DEFAULT NULL,
 `Keterangan` varchar(300) DEFAULT NULL,
 KEY `FK1DetailPem` (`No_Faktur_Pembelian`),
 KEY `FK2DetailNoEOQ` (`No_EOQ`),
 CONSTRAINT `FK1DetailPem` FOREIGN KEY (`No_Faktur_Pembelian`)
REFERENCES `pembelian_bahan_baku` (`No_Faktur_Pembelian`) ON DELETE
CASCADE ON UPDATE CASCADE,
 CONSTRAINT `FK2DetailNoEOQ` FOREIGN KEY (`No_EOQ`) REFERENCES
`eoq` (`No_EOQ`) ON DELETE CASCADE ON UPDATE CASCADE
```

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

4.1.4. Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap halaman program yang dibuat dan pengkodeannya dalam bentuk file program. Berikut ini adalah implementasi antarmuka yang dibuat dan dibedakan antara antarmuka untuk manajer, divisi administrasi dan divisi produksi.

4.1.4.1. Implementasi Antarmuka Manajer

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk divisi produksi dapat dilihat pada tabel berikut ini:

Tabel 4.3. Implementasi Antarmuka Manajer

Menu	**Deskripsi**
Login	Untuk menangani proses *login*admin.
Menu Manajer	Tampilan awal antarmuka manajer masuk ke proses selanjutnya.
Jabatan	Untuk mengolah data jabatan. Seperti tambah, ubah, cari, dan hapus data tahun ajaran.
Pegawai	Untuk mengelolah data pegawai. Seperti tambah,ubah,hapus dan cari
MRP	Untuk menghitung kebutuhan bahan baku dengan metode EOQ
Laporan	Untuk data laporan. Seperti pesanan pelanggan,pengolahan mrp, prodksi,dan pembelian bahan baku
About	Informasi nama aplikasi

4.1.4.2. Implementasi Antarmuka Divisi Administrasi

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk divisi administrasi dapat dilihat pada tabel berikut ini:

Tabel 4.4. Implementasi Antarmuka divisi administrasi

Menu	Deskripsi
Login	Untuk menangani proses *login*divisi administrasi.
Menu divisi administrasi	Tampilan awal antarmuka divisi administrasi masuk ke proses selanjutnya.
Jenis Pelanggan	Untuk mengolah data tahun ajaran. Seperti tambah, dan hapus data tahun ajaran.
Pelanggan	Untuk mengelolah data sponsor.seperti tambah,ubah,hapus
Kategori	Untuk mengolah data tahun ajaran. Seperti tambah, dan hapus data tahun ajaran.
Jenis Produk	Untuk mengelolah data sponsor.seperti tambah,ubah,hapus
Produk	Untuk mengolah data kriteria. Seperti tambah, ubah, dan hapus
Satuan	Untuk mengolah data pemohon. Seperti tambah, ubah, dan hapus
Bahan Baku	Penyeleksian beasiswa. Seperti tambah, ubah dan hapus.
Pesanan Pelanggan	Untuk mengolah data kriteria. Seperti tambah, ubah, dan hapus
Produksi	Untuk mengolah data pemohon. Seperti tambah, ubah, dan hapus
Pembelian Bahan Baku	Penyeleksian beasiswa. Seperti tambah, ubah dan hapus.

4.1.4.3. Implementasi Antarmuka Divisi Produksi

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk divisi produksi dapat dilihat pada tabel berikut ini:

Tabel 4.5. Implementasi Antarmuka divisi produksi

Menu	Deskripsi
Login	Untuk menangani proses *login*divisi administrasi.
Menu divisi produksi	Tampilan awal antarmuka divisi produksimasuk ke proses selanjutnya.
Produksi	Pengolahan proses produksi

4.2. Pengujian Alpha

Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan merepresentasikan kajian pokok dari spesifikasi, perancangan, dan pengkodean. Pengujian yang digunakan untuk menguji sistem ini adalah metode pengujian *black-box*. Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak.

4.2.1 Rencana Pengujian

Rencana pengujian akan dilakukan dengan menguji sistem secara alpha dan beta. Pengujian perangkat lunak Sistem Informasi Pengolahan Data Pesanan dan *Inventory Control* Berbasis *Client-Server* Pada CV.Bali67 *Advertising*. Rencana pengujian selengkapnya dapat dilihat pada tabel di bawah ini :

Tabel 4.6. Skenario Pengujian Aplikasi Untuk Bagian Manajer

Item Uji	Detail Pengujian	Jenis Pengujian
Login	Verifikasi username	*Black-Box*
	Verifikasi password	*Black-Box*

Pengolahan Jabatan	Tambah Data Jabatan	*Black-Box*
	Ubah Data Jabatan	*Black-Box*
	Hapus Data Jabatan	*Black-Box*
Pengolahan Pegawai	Tambah Data pegawai	*Black-Box*
	Ubah Data pegawai	*Black-Box*
	Hapus Data pegawai	*Black-Box*
Pengolahan Pemasok	Tambah Data Produk	*Black-Box*
	Ubah Data Produk	*Black-Box*
	Hapus Data Produk	*Black-Box*
Pengolahan MRP	Tambah EOQ	*Black-Box*

Tabel 4.7. Skenario Pengujian Aplikasi Untuk Bagian Divisi Administrasi

Item Uji	**Detail Pengujian**	**Jenis Pengujian**
Login	Verifikasi username	*Black-Box*
	Verifikasi password	*Black-Box*
Pengolahan Jenis Pelanggan	Tambah Data jenis pelanggan	*Black-Box*
	Ubah Data jenis pelanggan	*Black-Box*
	Hapus Data jenis pelanggan	*Black-Box*
Pengolahan Pelanggan	Tambah Data Pelanggan	*Black-Box*
	Ubah Data Pelanggan	*Black-Box*
Pengolahan Kategori	Tambah Data jenis pelanggan	*Black-Box*
	Ubah Data jenis pelanggan	*Black-Box*
	Hapus Data jenis pelanggan	*Black-Box*
Pengolahan Jenis Produk	Tambah Data Jenis Produk	*Black-Box*
	Ubah Data Jenis Produk	*Black-Box*
	Hapus Data Jenis Produk	*Black-Box*
Pengolahan Produk	Tambah Data Produk	*Black-Box*
	Ubah Data Produk	*Black-Box*
	Hapus Data Produk	*Black-Box*
Pengolahan Bahan Baku	Tambah Data Bahan Baku	*Black-Box*
	Ubah Data Bahan Baku	*Black-Box*
	Hapus Data Bahan Baku	*Black-Box*

Pengolahan BoM	Tambah Data BoM	*Black-Box*
Pengolahan Pesanan Pelanggan	Tambah Transaksi Pesanan Pelanggan	*Black-Box*
Pengolahan Produksi	Tambah Data Produksi	*Black-Box*
	Ubah Data Produksi	*Black-Box*
Pengolahan Pembelian Bahan Baku	Tambah Pembelian Bahan Baku	*Black-Box*

Tabel 4.8. Skenario Pengujian Aplikasi Untuk Bagian Divisi Produksi

Item Uji	Detail Pengujian	Jenis Pengujian
Login	Verifikasi username	*Black-Box*
	Verifikasi password	*Black-Box*
Pengolahan Produksi	Tambah Data Produksi	*Black-Box*
	Ubah Data Produksi	*Black-Box*

4.2.2. Kasus dan Hasil Pengujian Alpha

4.2.2.1 Pengujian Login

1. Pengujian Login Manajer

Tabel 4.9 Pengujian Login Manajer

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Username Contoh : 123 *Password* Contoh : 123	Masuk kehalaman manajer dan status pegawai berubah menjadi "Aktif"	*Login* berhasil dan dapat masuk ke halaman manajer sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Username Contoh : Kosong	Menampilkan pesan "Anda belum mengisi Username, silahakan isi terlebih	Menampilkan pesan "Anda belum mengisi Username,	[✓] Diterima [ ] Ditolak

	dahulu"	silahakan isi terlebih dahulu". Sesuai yang diharapkan	
Password Contoh : kosong	Menampilkan pesan "Anda belum mengisi Password, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi Password, silahakan isi terlebih dahulu". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak

2. Pengujian Login Divisi Administrasi

Tabel 4.10 Pengujian Login Divisi Administrasi

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Username Contoh : qwe *Password* Contoh : qwe	Masuk kehalaman manajer dan status pegawai berubah menjadi "Aktif"	*Login* berhasil dan dapat masuk ke halaman manajer sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Username Contoh : Kosong	Menampilkan pesan "Anda belum mengisi Username, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi Username, silahakan isi terlebih dahulu"	[✓] Diterima [ ] Ditolak
Password Contoh : kosong	Menampilkan pesan "Anda belum mengisi Password, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi Password, silahakan isi terlebih dahulu". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak

3. Pengujian Login Divisi Produksi

Tabel 4.11 Pengujian Login Divisi Produksi

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Username Contoh : Nono *Password* Contoh : nono	Masuk kehalaman manajer dan status pegawai berubah menjadi "Aktif"	*Login* berhasil dan dapat masuk ke halaman manajer sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Username Contoh : Kosong	Menampilkan pesan "Anda belum mengisi Username, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi Username, silahakan isi terlebih dahulu"	[✓] Diterima [ ] Ditolak
Password Contoh : kosong	Menampilkan pesan "Anda belum mengisi Password, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi Password, silahakan isi terlebih dahulu". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak

4.2.2.2. Pengujian Data Jabatan

Berdasarkan Skenario pengujian yang disusun, maka dilakukan pengujian sebagai berikut, ada 3 pengujian pada pengolahan data jabatan yaitu :

1. Pengujian Tambah Data Jabatan

Table 4.12 Pengujian Tambah Data Jabatan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama jabatan Contoh : Divisi Produksi Spesifikasi Contoh : Desainer	Data tersimpan pada database dan Muncul pesan "Data telah berhasil disimpan"	Data tersimpan pada databese dan Muncul pesan "Data telah berhasil disimpan" sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama jabatan Contoh : Kosong	Menampilkan pesan "Anda belum mengisi Nama jabatan, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi Nama jabatan, silahakan isi terlebih dahulu".Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Spesifikasi Contoh : kosong	Menampilkan pesan "Anda belum mengisi spesifikasi, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi Spesifikasi, silahakan isi terlebih dahulu".Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak

2. Pengujian Ubah Data Jabatan

Table 4.13 Pengujian Ubah Data Jabatan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama jabatan Contoh : Divisi Produksi Spesifikasi Contoh : Desainer	Data tersimpan pada database dan Muncul pesan "Data telah berhasil diubah"	Data tersimpan pada databese dan Muncul pesan "Data telah berhasil diubah" sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama jabatan Contoh : Kosong	Menampilkan pesan "Nama jabatan masih kosong"	Menampilkan pesan "Nama jabatan masih kosong". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Spesifikasi Contoh : kosong	Menampilkan pesan "Spesifikasi masih kosong"	Menampilkan pesan "Spesifikasi masih kosong". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak

3. Pengujian Hapus Data Jabatan

Table 4.14 Pengujian Hapus Data Jabatan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Jabatan Contoh : 1	Data di database berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus"	Data di databese berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus" sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak

4.2.2.3. Pengujian Pengolahan Data Pegawai

Berdasarkan Skenario pengujian yang disusun, maka dilakukan pengujian sebagai berikut, ada 3 pengujian pada pengolahan data pegawai yaitu :

1. Pengujian Tambah Data Pegawai

Table 4.15 Pengujian Tambah Data Pegawai

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Jabatan Contoh : 2 Nama Pegawai Contoh : Winda Hamidah Jenis Kelamin Contoh : P Alamat Contoh : Jl. Sukamulya No.90 Cilacap No Telepon Contoh : 0216498766 Email Contoh : winda_Cute@gmail.com *username* Contoh : qwe *password* Contoh : qwe Status Contoh : NON AKTIF	Data tersimpan kedalam database dan muncul pesan “Data berhasil disimpan”	Data tersimpan kedalam database dan muncul pesan “Data berhasil disimpan”. Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Jabatan Contoh : Kosong	Menampilkan pesan “Anda belum mengisi kode jabatan, silahakan isi terlebih dahulu”	Menampilkan pesan “Anda belum mengisi kode jabatan, silahakan isi terlebih dahulu”	[✓] Diterima [ ] Ditolak

Nama Pegawai Contoh : Kosong	Menampilkan pesan "Anda belum mengisi nama pegawai, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi nama pegawai, silahakan isi terlebih dahulu"	[✓] Diterima [ ] Ditolak
Jenis Kelamin Contoh : Kosong	Menampilkan pesan "Anda belum mengisi jenis kelamin, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi jenis kelamin, silahakan isi terlebih dahulu"	[✓] Diterima [ ] Ditolak
Alamat Contoh : Kosong	Menampilkan pesan "Anda belum mengisi alamat, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi alamat, silahakan isi terlebih dahulu"	[✓] Diterima [ ] Ditolak
No Telpon Contoh : Kosong	Menampilkan pesan "Anda belum mengisi no telpon, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi no telpon, silahakan isi terlebih dahulu"	[✓] Diterima [ ] Ditolak
Email Contoh : Kosong	Menampilkan pesan "Anda belum mengisi email, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi email, silahakan isi terlebih dahulu"	[✓] Diterima [ ] Ditolak
username Contoh : Kosong	Menampilkan pesan "Anda belum mengisi *username*, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi *username*, silahakan isi terlebih dahulu"	[✓] Diterima [ ] Ditolak
password Contoh : Kosong	Menampilkan pesan "Anda belum mengisi *password*, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi *password*, silahakan isi terlebih dahulu"	[✓] Diterima [ ] Ditolak
Status Contoh : Kosong	Menampilkan pesan "Anda belum mengisi status, silahakan isi terlebih dahulu"	Menampilkan pesan "Anda belum mengisi status, silahakan isi terlebih dahulu"	[✓] Diterima [ ] Ditolak

2. Pengujian Ubah Data Pegawai

Table 4.16 Pengujian Ubah Data Pegawai

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Jabatan Contoh : 2 Nama Pegawai Contoh : Winda Hamidah Jenis Kelamin Contoh : P Alamat Contoh : Jl. Sukamulya No.90 Cilacap No Telepon Contoh : 0216498766 Email Contoh : winda_Cute@gmail.com *username* Contoh : qwe *password* Contoh : qwe Status Contoh : NON AKTIF	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah"	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Jabatan Contoh : Kosong	Menampilkan pesan "Kode jabatan masih kosong"	Menampilkan pesan "Kode jabatan masih kosong"	[✓] Diterima [ ] Ditolak
Nama Pegawai Contoh : Kosong	Menampilkan pesan "Nama pegawai masih kosong"	Menampilkan pesan "Nama pegawai masih kosong"	[✓] Diterima [ ] Ditolak
Jenis Kelamin Contoh : Kosong	Menampilkan pesan "Jenis Kelamin masih kosong"	Menampilkan pesan "Jenis Kelamin masih kosong"	[✓] Diterima [ ] Ditolak
Alamat Contoh : Kosong	Menampilkan pesan "Alamat masih kosong"	Menampilkan pesan "Alamat masih kosong"	[✓] Diterima [ ] Ditolak
No Telepon Contoh : Kosong	Menampilkan pesan "no Telpon masih	Menampilkan pesan "no Telpon masih	[✓] Diterima [ ] Ditolak

	kosong”	kosong”	
Email Contoh : Kosong	Menampilkan pesan “Email masih kosong”	Menampilkan pesan “Email masih kosong”	[✓] Diterima [ ] Ditolak
username Contoh : Kosong	Menampilkan pesan “*Username* masih kosong”	Menampilkan pesan “*Username* masih kosong”	[✓] Diterima [ ] Ditolak
password Contoh : Kosong	Menampilkan pesan “*Password* masih kosong”	Menampilkan pesan “*Password* masih kosong”	[✓] Diterima [ ] Ditolak
Status Contoh : Kosong	Menampilkan pesan “Status masih kosong”	Menampilkan pesan “Status masih kosong”	[✓] Diterima [ ] Ditolak

3. Pengujian Hapus Data Pegawai

Table 4.17 Pengujian Hapus Data Pegawai

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Pegawai Contoh : 1	Data di database berhasil dihapus dan Muncul pesan “Data telah berhasil dihapus”	Data di databese berhasil dihapus dan Muncul pesan “Data telah berhasil dihapus”. sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak

4.2.2.4. Pengujian Pengolahan Data Pemasok

Berdasarkan Skenario pengujian yang disusun, maka dilakukan pengujian sebagai berikut, ada 3 pengujian pada pengolahan data pegawai yaitu :

1. Pengujian Tambah Data Pemasok

Tabel 4.18 Pengujian Tambah Data Pemasok

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Pemasok Contoh : My Jet Alamat Contoh : Jl. MH.Toha No.123 Jakarta Selatan No Telepon Contoh : 08132546658 Email Contoh : myjet@yahoo.com Status Contoh : Percetakaan Lain	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan"	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Pemasok Contoh : Kosong	Menampilkan pesan "Masukkan Nama Pemasok"	Menampilkan pesan "Masukkan Nama Pemasok"	[✓] Diterima [ ] Ditolak
Alamat Contoh : Kosong	Menampilkan pesan "Masukkan alamat"	Menampilkan pesan "Masukkan alamat"	[✓] Diterima [ ] Ditolak
No Telepon Contoh : Kosong	Menampilkan pesan "Masukkan no telepon"	Menampilkan pesan "Masukkan no telepon"	[✓] Diterima [ ] Ditolak
Email Contoh : Kosong	Menampilkan pesan "Masukkan Email"	Menampilkan pesan "Masukkan Email"	[✓] Diterima [ ] Ditolak
Status Contoh : Kosong	Menampilkan pesan "Masukkan Status"	Menampilkan pesan "Masukkan Status"	[✓] Diterima [ ] Ditolak

2. Pengujian Ubah Data Pemasok

Tabel 4.19 Pengujian Ubah Data Pemasok

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Pemasok Contoh : My Jet Alamat Contoh : Jl. MH.Toha No.123 Jakarta Selatan No Telepon Contoh : 08132546658 Email Contoh : myjet@yahoo.com Status Contoh : Percetakaan Lain	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah"	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Pemasok Contoh : Kosong	Menampilkan pesan "Nama Pemasok masih kosong"	Menampilkan pesan "Nama Pemasok masih kosong"	[✓] Diterima [ ] Ditolak
Alamat Contoh : Kosong	Menampilkan pesan "Alamat masih kosong"	Menampilkan pesan "Alamat masih kosong"	[✓] Diterima [ ] Ditolak
No Telepon Contoh : Kosong	Menampilkan pesan "No Telpon masih kosong"	Menampilkan pesan "No Telpon masih kosong"	[✓] Diterima [ ] Ditolak
Email Contoh : Kosong	Menampilkan pesan "Email masih kosong"	Menampilkan pesan "Email masih kosong"	[✓] Diterima [ ] Ditolak
Status Contoh : Kosong	Menampilkan pesan "Status masih kosong"	Menampilkan pesan "Status masih kosong"	[✓] Diterima [ ] Ditolak

3. Pengujian Hapus Data Pemasok

Tabel 4.20 Pengujian Hapus Data Pemasok

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Pemasok Contoh : 1	Data di database berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus"	Data di databese berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus". sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak

4.2.2.5. Pengujian Pengolahan Data Jenis Pelanggan

Berdasarkan Skenario pengujian yang disusun, maka dilakukan pengujian sebagai berikut, ada 3 pengujian pada pengolahan data jenis Pelanggan yaitu :

1. Pengujian Tambah Data Jenis Pelanggan

Tabel 4.21 Pengujian Tambah Data Jenis Pelanggan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Jenis Pelanggan Contoh : Instansi Pemerintahan Diskon Contoh : 20 % Keterangan Contoh : Rumah Sakit	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan"	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Jenis Pelanggan Contoh : Kosong	Menampilkan pesan "Masukkan Nama Jenis Pelanggan"	Menampilkan pesan "Masukkan Nama Jenis Pelanggan"	[✓] Diterima [ ] Ditolak
Diskon Contoh : Kosong	Menampilkan pesan "Masukkan Diskon"	Menampilkan pesan "Masukkan Diskon"	[✓] Diterima [ ] Ditolak
Keterangan Contoh : Kosong	Menampilkan pesan "Masukkan Keterangan"	Menampilkan pesan "Masukkan Keterangan"	[✓] Diterima [ ] Ditolak

2. Pengujian Ubah Data Jenis Pelanggan

Tabel 4.22 Pengujian Ubah Data Jenis Pelanggan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Jenis Pelanggan Contoh : Instansi Pemerintahan Diskon Contoh : 20 % Keterangan Contoh : Rumah Sakit	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah"	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Jenis Pelanggan Contoh : Kosong	Menampilkan pesan "Nama Jenis Pelanggan masih kosong"	Menampilkan pesan "Nama Jenis Pelanggan masih kosong"	[✓] Diterima [ ] Ditolak
Diskon Contoh : Kosong	Menampilkan pesan "Diskon masih	Menampilkan pesan "Diskon	[✓] Diterima [ ] Ditolak

	kosong”	masih kosong”	
Keterangan Contoh : Kosong	Menampilkan pesan “Keterangan masih kosong”	Menampilkan pesan “Keterangan masih kosong”	[✓] Diterima [ ] Ditolak

3. Pengujian Hapus Data Jenis Pelanggan

Tabel 4.23 Pengujian Hapus Data Jenis Pelanggan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Jenis Pelanggan Contoh : 1	Data di database berhasil dihapus dan Muncul pesan “Data telah berhasil dihapus”	Data di databese berhasil dihapus dan Muncul pesan “Data telah berhasil dihapus”. sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak

4.2.2.6. Pengujian Pengolahan Data Pelanggan

Berdasarkan Skenario pengujian yang disusun, maka dilakukan pengujian sebagai berikut, ada 2 pengujian pada pengolahan data Pelanggan yaitu :

1. Pengujian Tambah Data Pelanggan

Tabel 4.24 Pengujian Tambah Data Pelanggan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Jenis Pelanggan Contoh : 1 Nama Pelanggan Contoh : Mitha Jenis Kelamin Contoh : P Alamat Contoh : JL. Muncul No.80 Cilacap No Telepon Contoh : 0813646233 Email Contoh : mitha_cute@yahoo.com Status Contoh : Pelanggan Baru Keterangan Contoh : RSUD	Data tersimpan kedalam database dan muncul pesan “Data berhasil disimpan”	Data tersimpan kedalam database dan muncul pesan “Data berhasil disimpan”. Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Jenis Pelanggan Contoh : Kosong	Menampilkan pesan “Masukkan Kode Jenis Pelanggan”	Menampilkan pesan “Masukkan Kode Jenis Pelanggan”	[✓] Diterima [ ] Ditolak
Nama Pelanggan Contoh : Kosong	Menampilkan pesan “Masukkan Nama Pelanggan”	Menampilkan pesan “Masukkan Nama Pelanggan”	[✓] Diterima [ ] Ditolak
Jenis Kelamin Contoh : Kosong	Menampilkan pesan “Masukkan Jenis Kelamin”	Menampilkan pesan “Masukkan Jenis Kelamin”	[✓] Diterima [ ] Ditolak

Alamat Contoh : Kosong	Menampilkan pesan "Masukkan Alamat"	Menampilkan pesan "Masukkan Alamat"	[✓] Diterima [ ] Ditolak
No Telepon Contoh : Kosong	Menampilkan pesan "Masukkan No Telepon"	Menampilkan pesan "Masukkan No Telepon"	[✓] Diterima [ ] Ditolak
Email Contoh : Kosong	Menampilkan pesan "Masukkan Email"	Menampilkan pesan "Masukkan Email"	[✓] Diterima [ ] Ditolak
Status Contoh : Kosong	Menampilkan pesan "Masukkan Status"	Menampilkan pesan "Masukkan Status"	[✓] Diterima [ ] Ditolak
Keterangan Contoh : Kosong	Menampilkan pesan "Masukkan Keterangan"	Menampilkan pesan "Masukkan Keterangan"	[✓] Diterima [ ] Ditolak

2. Pengujian Ubah Data Pelanggan

Tabel 4.25 Pengujian Ubah Data Pelanggan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Jenis Pelanggan Contoh : 1 Nama Pelanggan Contoh : Mitha Jenis Kelamin Contoh : P Alamat Contoh : JL. Muncul No.80 Cilacap No Telepon Contoh : 0813646233 Email Contoh : mitha_cute@yahoo.com Status Contoh : Pelanggan Baru Keterangan Contoh : RSUD	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah"	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Jenis Pelanggan Contoh : Kosong	Menampilkan pesan	Menampilkan pesan	[✓] Diterima [ ] Ditolak

	"Kode Jenis Pelanggan masih kosong"	"Kode Jenis Pelanggan masih kosong"	
Nama Pelanggan Contoh : Kosong	Menampilkan pesan "Nama Pelanggan masih kosong"	Menampilkan pesan "Nama Pelanggan masih kosong"	[✓] Diterima [ ] Ditolak
Jenis Kelamin Contoh : Kosong	Menampilkan pesan "Data tidak ditemukan"	Menampilkan pesan "Data tidak ditemukan"	[✓] Diterima [ ] Ditolak
Alamat Contoh : Kosong	Menampilkan pesan "Alamat masih kosong"	Menampilkan pesan "Alamat masih kosong"	[✓] Diterima [ ] Ditolak
No Telepon Contoh : Kosong	Menampilkan pesan "No Telpon masih kosong"	Menampilkan pesan "No Telpon masih kosong"	[✓] Diterima [ ] Ditolak
Email Contoh : Kosong	Menampilkan pesan "Email masih kosong"	Menampilkan pesan "Email masih kosong"	[✓] Diterima [ ] Ditolak
Status Contoh : Kosong	Menampilkan pesan "Status masih kosong"	Menampilkan pesan "Status masih kosong"	[✓] Diterima [ ] Ditolak
Keterangan Contoh : Kosong	Menampilkan pesan "Keterangan masih kosong"	Menampilkan pesan "Keterangan masih kosong"	[✓] Diterima [ ] Ditolak

4.2.2.7. Pengujian Pengolahan Data Kategori

Berdasarkan Skenario pengujian yang disusun, maka dilakukan pengujian sebagai berikut, ada 2 pengujian pada pengolahan data kategori yaitu :

1. Pengujian Tambah Data Kategori

Table 4.26 Pengujian Tambah Data Kategori

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Kategori Contoh : Digital Printing Keterangan Contoh : Digital	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan"	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Kategori Contoh : Kosong	Menampilkan pesan "Masukkan Nama Kategori"	Menampilkan pesan "Masukkan Nama Kategori"	[✓] Diterima [ ] Ditolak
Keterangan Contoh : Kosong	Menampilkan pesan "Masukkan Keterangan"	Menampilkan pesan "Masukkan Keterangan"	[✓] Diterima [ ] Ditolak

2. Pengujian Ubah Data Kategori

Table 4.27 Pengujian Ubah Data Kategori

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Kategori Contoh : Digital Printing Keterangan Contoh : Digital	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah"	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Kategori Contoh : Kosong	Menampilkan pesan "Nama Kategori masih kosong"	Menampilkan pesan "Data tidak ditemukan"	[✓] Diterima [ ] Ditolak
Keterangan Contoh : Kosong	Menampilkan pesan "Keterangan masih kosong"	Menampilkan pesan "Data tidak ditemukan"	[✓] Diterima [ ] Ditolak

3. Pengujian Hapus Data Kategori

Table 4.28 Pengujian Hapus Data Kategori

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Kategori Contoh : 1	Data di database berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus"	Data di databese berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus". sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak

4.2.2.8. Pengujian Data Jenis Produk

Berdasarkan Skenario pengujian yang disusun, maka dilakukan pengujian sebagai berikut, ada 3 pengujian pada pengolahan data Jenis Produk yaitu :

1. Pengujian Tambah Data Jenis Produk

Table 4.29 Pengujian Tambah Data Jenis Produk

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Kategori Contoh : 1 Nama Jenis Produk Contoh : Standing Banner Keterangan Contoh : Flexy	Data tersimpan kedalam database dan muncul pesan “Data berhasil disimpan”	Data tersimpan kedalam database dan muncul pesan “Data berhasil disimpan”. Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Kategori Contoh : Kosong	Menampilkan pesan “Masukkan Kode Kategori”	Menampilkan pesan “Masukkan Kode Kategori”	[✓] Diterima [ ] Ditolak
Nama Jenis Produk Contoh : Kosong	Menampilkan pesan “Masukkan Nama Jenis Produk”	Menampilkan pesan “Masukkan Nama Jenis Produk”	[✓] Diterima [ ] Ditolak
Keterangan Contoh : Kosong	Menampilkan pesan “Masukkan Keterangan”	Menampilkan pesan “Masukkan Keterangan”	[✓] Diterima [ ] Ditolak

2. Pengujian Ubah Data Jenis Produk

Table 4.30 Pengujian Ubah Data Jenis Produk

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Kategori Contoh : 1 Nama Jenis Produk Contoh : Standing Banner Keterangan Contoh : Flexy	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah"	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Kategori Contoh : Kosong	Menampilkan pesan "Kode Kategori masih kosong"	Menampilkan pesan "Kode Kategori masih kosong"	[✓] Diterima [ ] Ditolak
Nama Jenis Produk Contoh : Kosong	Menampilkan pesan "Nama Jenis Produk masih kosong"	Menampilkan pesan "Nama Jenis Produk masih kosong"	[✓] Diterima [ ] Ditolak
Keterangan Contoh : Kosong	Menampilkan pesan "Keterangan masih kosong"	Menampilkan pesan "Keterangan masih kosong"	[✓] Diterima [ ] Ditolak

3. Pengujian Hapus Data Jenis Produk

Table 4.31 Pengujian Hapus Data Jenis Produk

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Kategori Contoh : 1	Data di database berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus"	Data di databese berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus". sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak

4.2.2.9. Pengujian pengolahan data Produk

Berdasarkan Skenario pengujian yang disusun, maka dilakukan pengujian sebagai berikut, ada 3 pengujian pada pengolahan data Produk yaitu :

1. Pengujian Tambah Data Produk

Table 4.32 Pengujian Tambah Data Produk

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Produk Contoh : Standing X Banner uk 160x60 Kode Jenis Produk Contoh : 1 Harga Pesan Contoh : 120000 Ukuran Produk Contoh : Ukuran Tetap Status Produk Contoh : Stok Stabil Keterangan Contoh : Digital Flexy	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan"	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Produk Contoh : Kosong	Menampilkan pesan "Masukkan Nama Produk"	Menampilkan pesan "Masukkan Nama Produk"	[✓] Diterima [ ] Ditolak
Kode Jenis Produk Contoh : Kosong	Menampilkan pesan "Masukkan Kode Jenis Produk"	Menampilkan pesan "Masukkan Kode Jenis Produk"	[✓] Diterima [ ] Ditolak
Harga Pesan Contoh : Kosong	Menampilkan pesan "Masukkan Harga Pesan"	Menampilkan pesan "Masukkan Harga Pesan"	[✓] Diterima [ ] Ditolak
Ukuran Produk Contoh : Kosong	Menampilkan pesan "Masukkan Ukuran Produk"	Menampilkan pesan "Masukkan Ukuran Produk"	[✓] Diterima [ ] Ditolak
Status Produk Contoh : Kosong	Menampilkan pesan "Masukkan Status Produk"	Menampilkan pesan "Masukkan Status Produk"	[✓] Diterima [ ] Ditolak

Keterangan Contoh : Kosong	Menampilkan pesan "Masukkan Keterangan"	Menampilkan pesan "Masukkan Keterangan"	[✓] Diterima [ ] Ditolak

2. Pengujian Ubah Data Produk

Table 4.33 Pengujian Ubah Data Produk

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Produk Contoh : Standing X Banner uk 160x60 Kode Jenis Produk Contoh : 1 Harga Pesan Contoh : 120000 Ukuran Produk Contoh : Ukuran Tetap Status Produk Contoh : Stok Stabil Keterangan Contoh : Digital Flexy	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah"	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Nama Produk Contoh : Kosong	Menampilkan pesan "Nama Produk masih kosong"	Menampilkan pesan "Nama Produk masih kosong"	[✓] Diterima [ ] Ditolak
Kode Jenis Produk Contoh : Kosong	Menampilkan pesan "Kode Jenis Produk masih kosong"	Menampilkan pesan "Kode Jenis Produk masih kosong"	[✓] Diterima [ ] Ditolak
Harga Pesan Contoh : Kosong	Menampilkan pesan "Harga Pesan masih kosong"	Menampilkan pesan "Harga Pesan masih kosong"	[✓] Diterima [ ] Ditolak
Ukuran Produk Contoh : Kosong	Menampilkan pesan "Ukuran Produk masih kosong"	Menampilkan pesan "Ukuran Produk masih kosong"	[✓] Diterima [ ] Ditolak
Status Produk Contoh : Kosong	Menampilkan pesan "Status Produk masih kosong"	Menampilkan pesan "Status Produk masih kosong"	[✓] Diterima [ ] Ditolak
Keterangan Contoh : Kosong	Menampilkan pesan "Keterangan masih kosong"	Menampilkan pesan "Keterangan masih kosong"	[✓] Diterima [ ] Ditolak

3. Pengujian Hapus Data Produk

Table 4.34 Pengujian Hapus Data Produk

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Produk Contoh : 1	Data di database berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus"	Data di databese berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus". sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak

4.2.2.10. Pengujian pengolahan Data Satuan

Berdasarkan Skenario pengujian yang disusun, maka dilakukan pengujian sebagai berikut, ada 3 pengujian pada pengolahan data satuan yaitu :

1. Pengujian Tambah Data Satuan

Table 4.35 Pengujian Tambah Data Satuan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Unit Satuan Contoh : ml Nama Satuan Contoh : Mililiter	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan"	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Unit Satuan Contoh : Kosong	Menampilkan pesan "Masukkan Unit	Menampilkan pesan "Masukkan Unit	[✓] Diterima [ ] Ditolak

	Satuan"	Satuan"	
Nama Satuan Contoh : Kosong	Menampilkan pesan "Masukkan Nama Satuan"	Menampilkan pesan "Masukkan Nama Satuan"	[✓] Diterima [ ] Ditolak

2. Pengujian Ubah Data Satuan

Table 4.36 Pengujian Ubah Data Satuan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Unit Satuan Contoh : ml Nama Satuan Contoh : Mililiter	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah"	Data tersimpan kedalam database dan muncul pesan "Data berhasil diubah". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Unit Satuan Contoh : Kosong	Menampilkan pesan "Unit Satuan masih kosong"	Menampilkan pesan "Unit Satuan masih kosong"	[✓] Diterima [ ] Ditolak
Nama Satuan Contoh : Kosong	Menampilkan pesan "Nama Satuan masih kosong"	Menampilkan pesan "Nama Satuan masih kosong"	[✓] Diterima [ ] Ditolak

3. Pengujian Hapus Data Satuan

Table 4.37 Pengujian Hapus Data Satuan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Satuan Contoh : 1	Data di database berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus"	Data di databese berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus". sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak

4.2.2.11. Pengujian Pengolahan Data Bahan Baku

Berdasarkan Skenario pengujian yang disusun, maka dilakukan pengujian sebagai berikut, ada 3 pengujian pada pengolahan data satuan yaitu :

1. Pengujian Tambah Data Bahan Baku

Table 4.38 Pengujian Tambah Data Bahan Baku

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Satuan Contoh : 1 Nama Bahan Baku Contoh : Flexy Banner Harga Contoh : 1536000 Stok Contoh : 44.8 Minimal Order Contoh : 160	Data tersimpan kedalam database dan muncul pesan “Data berhasil disimpan”	Data tersimpan kedalam database dan muncul pesan “Data berhasil disimpan”. Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Satuan Contoh : Kosong	Menampilkan pesan “Masukkan Kode Satuan”	Menampilkan pesan “Masukkan Kode Satuan”	[✓] Diterima [ ] Ditolak
Nama Bahan Baku Contoh : Kosong	Menampilkan pesan “Masukkan Nama Bahan Baku”	Menampilkan pesan “Masukkan Nama Bahan Baku”	[✓] Diterima [ ] Ditolak
Harga Contoh : Kosong	Menampilkan pesan “Masukkan Harga”	Menampilkan pesan “Masukkan Harga”	[✓] Diterima [ ] Ditolak
Stok Contoh : Kosong	Menampilkan pesan “Masukkan Stok”	Menampilkan pesan “Masukkan Stok”	[✓] Diterima [ ] Ditolak
Minimal Order Contoh : Kosong	Menampilkan pesan “Masukkan Minimal order”	Menampilkan pesan “Masukkan Minimal order”	[✓] Diterima [ ] Ditolak

2. Pengujian Ubah Data Bahan Baku

Table 4.39 Pengujian Ubah Data Bahan Baku

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Satuan Contoh : 1 Nama Bahan Baku Contoh : Flexy Banner Harga Contoh : 1536000 Stok Contoh : 44.8 Minimal Order Contoh : 160	Data tersimpan kedalam database dan muncul pesan “Data berhasil diubah”	Data tersimpan kedalam database dan muncul pesan “Data berhasil diubah”. Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Satuan Contoh : Kosong	Menampilkan pesan “Satuan masih kosong”	Menampilkan pesan “Satuan masih kosong”	[✓] Diterima [ ] Ditolak
Nama Bahan Baku Contoh : Kosong	Menampilkan pesan “Nama Bahan Baku masih kosong”	Menampilkan pesan “Nama Bahan Baku masih kosong”	[✓] Diterima [ ] Ditolak
Harga Contoh : Kosong	Menampilkan pesan “Harga masih kosong”	Menampilkan pesan “Harga masih kosong”	[✓] Diterima [ ] Ditolak
Stok Contoh : Kosong	Menampilkan pesan “Stok masih kosong”	Menampilkan pesan “Stok masih kosong”	[✓] Diterima [ ] Ditolak
Minimal Order Contoh : Kosong	Menampilkan pesan “Minimal order masih kosong”	Menampilkan pesan “Minimal order masih kosong”	[✓] Diterima [ ] Ditolak

3. Pengujian Hapus Data Bahan Baku

Table 4.40 Pengujian Hapus Data Bahan Baku

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
Kode Bahan Baku Contoh : 1	Data di database berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus"	Data di databese berhasil dihapus dan Muncul pesan "Data telah berhasil dihapus". sesuai dengan yang diharapkan	[✓] Diterima [ ] Ditolak

4.2.2.12 Pengujian Data BoM

Table 4.41 Pengujian Tambah Data BoM

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No Item Bom Contoh : 1 Kode Produk Contoh : 1 Kode Pegawai Contoh : 2 Kode Bahan Baku Contoh : 3 Level Contoh : 1 Qty Per Unit Contoh : 1 JumlahKomposisi Contoh : 12 Tipe Contoh : Beli Keterangan Contoh : Produksi	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan"	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No Item Bom Contoh : Kosong	Menampilkan pesan "Masukkan No Item Bom"	Menampilkan pesan "Masukkan No Item Bom"	[✓] Diterima [ ] Ditolak

Kode Produk Contoh : Kosong	Menampilkan pesan "Masukkan Kode Produk"	Menampilkan pesan "Masukkan Kode Produk"	[✓] Diterima [ ] Ditolak
Kode Pegawai Contoh : Kosong	Menampilkan pesan "Masukkan Kode Pegawai"	Menampilkan pesan "Masukkan Kode Pegawai"	[✓] Diterima [ ] Ditolak
Kode Bahan Baku Contoh : Kosong	Menampilkan pesan "Masukkan Kode Bahan Baku"	Menampilkan pesan "Masukkan Kode Bahan Baku"	[✓] Diterima [ ] Ditolak
Level Contoh : Kosong	Menampilkan pesan "Masukkan Level"	Menampilkan pesan "Masukkan Level"	[✓] Diterima [ ] Ditolak
Qty Per Unit Contoh : Kosong	Menampilkan pesan "Masukkan Qty Per Unit"	Menampilkan pesan "Masukkan Qty Per Unit"	[✓] Diterima [ ] Ditolak
Jumlah Komposisi Contoh : Kosong	Menampilkan pesan "Masukkan Jumlah Komposisi"	Menampilkan pesan "Masukkan Jumlah Komposisi"	[✓] Diterima [ ] Ditolak
Tipe Contoh : Kosong	Menampilkan pesan "Masukkan Tipe"	Menampilkan pesan "Masukkan Tipe"	[✓] Diterima [ ] Ditolak
Keterangan Contoh : Kosong	Menampilkan pesan "Masukkan Keterangan"	Menampilkan pesan "Masukkan Keterangan"	[✓] Diterima [ ] Ditolak

4.2.2.13 Pengujian Tambah Data Pesanan Pelanggan

Table 4.42 Pengujian Tambah Data Pesanan Pelanggan

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No_Faktur_Pemesanan, Tanggal_Bukti, Id_Pegawai, Id_Pelanggan,Jumlah, Total,Id_Produk, Jumlah_Pesan,P,L, Luas_Ukuran,Diskon, DP,Total_Harga, Subtotal,Total_Bayar, Status,Keterangan	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan"	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No_Faktur_Pemesanan, Tanggal_Bukti, Id_Pegawai,	Data tersimpan kedalam database dan muncul pesan "Data	Data tersimpan kedalam database dan muncul pesan "Data	[✓] Diterima [ ] Ditolak

Id_Pelanggan,Jumlah, Total,Id_Produk, Jumlah_Pesan,P,L, Luas_Ukuran,Diskon, DP,Total_Harga, Subtotal,Total_Bayar, Status,Keterangan	berhasil disimpan"	berhasil disimpan". Sesuai yang diharapkan	

4.2.2.14 Pengujian Pengolahan Data MRP

Pada pengolahan MRP ada 2 tahap yaitu :

1. Pemilihan Faktur Pesanan Pelanggan yang akan diolah

Table 4.43 Pengujian Pengolahan Data MRP (Memilih faktur pesanan pelanggan**)**

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No_EOQ Contoh : 1 Kode Pegawai Contoh : 1 Total Bayar Contoh : 0	Muncul Detail pesanan pada grid. Data Tersimpan pada tabel EOQ	Muncul Detail pesanan pada grid. Data Tersimpan pada tabel EOQ sesuai yang diharapkan	[✓] Diterima [ ] Ditolak [✓] Diterima [ ] Ditolak [✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No_EOQ Contoh : Kosong	Menampilkan pesan "No EOQ masih kosong"	Menampilkan pesan "No_EOQ masih kosong"	[✓] Diterima [ ] Ditolak
Kode Pegawai Contoh : Kosong	Menampilkan pesan "Kode Pegawai masih kosong"	Menampilkan pesan "Kode Pegawai masih kosong"	[✓] Diterima [ ] Ditolak
Total Bayar Contoh : Kosong	Tidak ada pesan	Tidak ada pesan	[✓] Diterima [ ] Ditolak

2. Pemilihan BoM yang sesuai dengan produk yang dipesan

Table 4.44 Pengujian Pengolahan Data MRP (Memilih BoM**)**

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No EOQ Contoh : 1 No Faktur Pemesanan Contoh : 2 No Item BoM Contoh : 1	Muncul Detail BOM pada grid. Data Tersimpan pada Detail_EOQ, dan muncul pesan "Data Berhasil Dihitung"	Muncul Detail BOM pada grid. Data Tersimpan pada Detail_EOQ, dan muncul pesan "Data Berhasil Dihitung"sesuai yang diharapkan	[✓] Diterima [ ] Ditolak [✓] Diterima [ ] Ditolak [✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No EOQ Contoh : Kosong	Menampilkan pesan "No EOQ masih kosong"	Menampilkan pesan "No EOQ masih kosong"	[✓] Diterima [ ] Ditolak
No Faktur Pemesanan Contoh : Kosong	Menampilkan pesan "No Faktur Pemesanan masih kosong"	Menampilkan pesan "No Faktur Pemesanan masih kosong"	[✓] Diterima [ ] Ditolak
No Item BoM Contoh : Kosong	Menampilkan pesan "No Item BoM masih kosong"	Menampilkan pesan "No Item BoM masih kosong"	[✓] Diterima [ ] Ditolak

4.2.2.14 Pengujian Tambah Data Produksi

Table 4.45 Pengujian Tambah Data Produksi

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No Produksi Contoh : 1 No EOQ Contoh : 1 Kode Pegawai Contoh : 3 Letak File Desain Contoh : D:\Skripsi\contoh\Coba2 Letak File Cetak Contoh : D:\Skripsi\contoh\Coba2 Status Produksi Contoh : Proses Desain Tanggal Produksi Contoh : 12/05/2012 Tanggal Selesai Produksi Contoh : 12/05/2012 Keterangan Produksi Contoh : Proses	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan"	Data tersimpan kedalam database dan muncul pesan "Data berhasil disimpan". Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No Produksi Contoh : Kosong	Menampilkan pesan "No Produksi masih kosong"	Menampilkan pesan "No Produksi masih kosong"	[✓] Diterima [ ] Ditolak
No EOQ Contoh : Kosong	Menampilkan pesan "No EOQ masih kosong"	Menampilkan pesan "No EOQ masih kosong"	[✓] Diterima [ ] Ditolak
Kode Pegawai Contoh : Kosong	Menampilkan pesan "Kode Pegawai masih kosong"	Menampilkan pesan "Kode Pegawai masih kosong"	[✓] Diterima [ ] Ditolak
Letak File Desain Contoh : Kosong	Menampilkan pesan "Letak File Desain masih kosong"	Menampilkan pesan "Letak File Desain masih kosong"	[✓] Diterima [ ] Ditolak
Letak File Cetak Contoh : Kosong	Menampilkan pesan "Letak File Cetak masih kosong"	Menampilkan pesan "Letak File Cetak masih kosong"	[✓] Diterima [ ] Ditolak
Status Produksi	Menampilkan pesan	Menampilkan pesan	[✓] Diterima

Contoh : Kosong	"Status Produksi masih kosong"	"Status Produksi masih kosong"	[ ] Ditolak
Tanggal Produksi Contoh : Kosong	Menampilkan pesan "Tanggal Produksi masih kosong"	Menampilkan pesan "Tanggal Produksi masih kosong"	[✓] Diterima [ ] Ditolak
Tanggal Selesai Produksi Contoh : Kosong	Menampilkan pesan "Tanggal Selesai Produksi masih kosong"	Menampilkan pesan "Tanggal Selesai Produksi masih kosong"	[✓] Diterima [ ] Ditolak
Keterangan Produksi Contoh : Kosong	Menampilkan pesan "Keterangan Produksi masih kosong"	Menampilkan pesan "Keterangan Produksi masih kosong"	[✓] Diterima [ ] Ditolak

4.2.2.14 Pengujian Tambah Data Pembelian Bahan Baku

Table 4.46 Tambah Pengolahan Data Pembelian Bahan Baku

Kasus dan Hasil Uji (Data Normal)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No Faktur Pembelian Contoh : 1 Tanggal Pembelian Contoh : 12/12/2012 Kode Pemasok Contoh : 1 Total Contoh : 1536000 Status Contoh : Belum Lunas Kode Pegawai Contoh : 2 No EOQ Contoh : 1 Kode Bahan Baku Contoh : 1 Jumlah Contoh : 160 Cara Bayar Contoh : Hutang Diskon Contoh : 20% Subtotal Contoh : 3870000 Total Bayar Contoh : 3560000 Keterangan Contoh : Nope	Data tersimpan kedalam database	Data tersimpan kedalam database. Sesuai yang diharapkan	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	Yang Diharapkan	Pengamatan	Kesimpulan
No Faktur Pembelian Contoh : Kosong	Menampilkan pesan “No Faktur Pembelian masih kosong”	Menampilkan pesan “No Faktur Pembelian masih kosong”	[✓] Diterima [ ] Ditolak
Tanggal Pembelian Contoh : Kosong	Menampilkan pesan “Tanggal Pembelian masih kosong”	Menampilkan pesan “Tanggal Pembelian masih kosong”	[✓] Diterima [ ] Ditolak
Kode Pemasok Contoh : Kosong	Menampilkan pesan “Kode Pemasok masih kosong”	Menampilkan pesan “Kode Pemasok masih kosong”	[✓] Diterima [ ] Ditolak

Total Contoh : Kosong	Tidak Menampilkan pesan	Tidak Menampilkan pesan	[✓] Diterima [ ] Ditolak
Status Contoh : Kosong	Menampilkan pesan "Status masih kosong"	Menampilkan pesan "Status masih kosong"	[✓] Diterima [ ] Ditolak
Kode Pegawai Contoh : Kosong	Menampilkan pesan "Kode Pegawai masih kosong"	Menampilkan pesan "Kode Pegawai masih kosong"	[✓] Diterima [ ] Ditolak
No EOQ Contoh : Kosong	Menampilkan pesan "No EOQ masih kosong"	Menampilkan pesan "No EOQ masih kosong"	[✓] Diterima [ ] Ditolak
Kode Bahan Baku Contoh : Kosong	Menampilkan pesan "Kode Bahan Baku masih kosong"	Menampilkan pesan "Kode Bahan Baku masih kosong"	[✓] Diterima [ ] Ditolak
Jumlah Contoh : Kosong	Menampilkan pesan "Jumlah masih kosong"	Menampilkan pesan "Jumlah masih kosong"	[✓] Diterima [ ] Ditolak
Cara Bayar Contoh : Kosong	Menampilkan pesan "Cara Bayar masih kosong"	Menampilkan pesan "Cara Bayar masih kosong"	[✓] Diterima [ ] Ditolak
Diskon Contoh : Kosong	Tidak Menampilkan pesan	Tidak Menampilkan pesan	[✓] Diterima [ ] Ditolak
Subtotal Contoh : Kosong	Tidak Menampilkan pesan	Tidak Menampilkan pesan	[✓] Diterima [ ] Ditolak
Total Bayar Contoh : Kosong	Tidak Menampilkan pesan	Tidak Menampilkan pesan	[✓] Diterima [ ] Ditolak
Keterangan Contoh : Kosong	Menampilkan pesan "Keterangan masih kosong"	Menampilkan pesan "Keterangan masih kosong"	[✓] Diterima [ ] Ditolak

4.2.3. Kesimpulan Hasil Pengujian Alpha

Berdasarkan hasil pengujian alpha (fungsional) dengan kasus uji diatas dapat ditarik kesimpulan sebagai berikut:

1. Jika data yang dimasukan benar maka sistem secara fungsional akan mengeluarkan keluaran sesuai dengan harapan.
2. Jika data yang dimasukan salah maka sistem akan mengeluarkan pesan kesalahan dan data keluaran tidak akan sesuai dengan harapan.

Dengan kata lain bahwa pembangunan Sistem ini bebas kesalahan dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

4.3. Pengujian Beta

Pengujian beta dilakukan dengan tujuan untuk mengetahui sejauh mana kualitas dari perangkat lunak yang dibangun, apakah sudah sesuai dengan harapan atau belum. Untuk itu dalam pengujian beta dilakukan penelitian dengan cara memberikan wawancara pada calon pengguna perangkat lunak yang dibangun. Adapun metode penelitian yang digunakan adalah metode kuantitatif.

Wawancara diberikan kepada beberapa *sample* calon pengguna perangkat lunak yang dibangun.

Wawancara ini terdiri dari 5 pertanyaan untuk setiap bagian (contoh wawancara dilampirkan). Wawancara ini diberikan kepada 1 orang manajer, 1 orang divisi administrasi, 1 orang divisi produksi.

a. Wawancara untuk Manajer

1. Apakah anda setuju dibangunnya perangkat lunak (*software*) ini dapat membantu mengatasi kesalahan-kesalahan dalam pencatatan transaksi ?

Saya setuju sekali, aplikasi ini membantu sekali untuk mengatasi kesalahan-kesalahan dalam pencatatan transaksi yang tadinya masih mengandalkan nota dan *form order*.

2. Dengan dibangunnya perangkat lunak (*software*) ini apakah dapat meningkatkan kinerja dan efisiensi waktu?

Aplikasi ini sangat membantu sekali dalam urusan mengelola pesanan pelanggan dari pemesanan, proses produksi sampai pengambilan pesanan dilakukan dengan cepat dan prosesnya dapat langsung dirasakan oleh pelanggan.

3. Apakah perangkat lunak (*software*) ini memudahkan dalam membantu menyelesaikan masalah dalam mencari kebutuhan bahan baku yang akan digunakan ?

Aplikasi ini sangat membantu sekali dalam mencari kebutuhan bahan baku yang akan digunakan untuk proses produksi sesuai dengan pesanan, yang sebelumnya disini hanya menggunakan bahan baku tanpa melihat berapa stok yang tersisa untuk pesanan berikutnya.

4. Apakah perangkat lunak (*software*) berbasis client-server ini memudahkan dalam mengelola laporan dengan akurat?

Perangkat lunak ini sangat memudahkan sekali, karena laporan yang sebelumnya dibuat hanya mengandalkan pencatatan laporan pada arsip yang ditulis tangan. itu membuat kerjaan semakin berat karena terkadang nota pesanan dan form order sering hilang.

5. Apakah penggunaan metode *MRP* dapat membantu mengatasi masalah dalam mencari kebutuhan bahan baku untuk memproses sebuah produk sesuai pesanan?

Saya sangat setuju, penggunaan metode ini memberikan hasil yang baik karena penggunaan bahan baku sangatlah penting untuk kelangsungan proses produksi agar proses tersebut dapat berjalan dengan lancar.

b. Wawancara untuk Divisi Administrasi

1. Apakah dibangunnya perangkat lunak (*software*) ini dapat membantu mengatasi masalah pencatatan pesanan pelanggan tanpa adanya *form order* ?

Aplikasi ini sangat membantu sekali walaupun tidak *form order* sudah tidak dibutuhkan lagi, karena pencatatan transaksi pada nota haruslah benar dan tidak boleh ada kesalahan karena akan berpengaruh pada pencatatan laporan dan proses yang akan dikerjakan nanti.

2. Apakah dengan dibangunnya perangkat lunak (*software*) ini dapat meningkatkan kinerja dan efisiensi waktu?

Saya sangat setuju sekali, karena pencatata pesanan pada nota dan *form order* yang ditulis tangan membuat pengerjaan pesanan lama dan berdampak pada pelanggan berikutnya yang akan memesanan karena harus meununggu.

3. Apakah perangkat lunak (*software*) ini memudahkan dalam membantu mengkoordinasikan data pesanan kepada bagian yang terkait?

Aplikasi ini memudahkan dalam mengkoordinasikan pesanan yang telah masuk untuk segera diproses, yang tadinya badian divisi administrasi selalu kesanan-kemari untuk memberikan file atau info pesanan kepada bagian desain dan operator printer.

4. Apakah perangkat lunak (*software*) berbasis client-server ini memudahkan dalam mengelola pesanan tanpa menggunakan *form order*?

Saya sangat setuju penggunaan client-server ini memudahkan pengelolaan pesanan karena info pesanan mudah dicari dan pendistribusiannya cepat.

5. Apakah penggunaan metode *MRP* ini merupakan metode yang terbaik untuk meningkatkan proses produksi dengan pengelolaan bahan baku yang terencana ?

Metode ini sangat baik sekali digunakan karena perusahaan sangat membutuhkan perhitungan untuk menghitung semua biaya produksi yang telah dikeluarkan.

a. Wawancara untuk Divisi Produksi

1. Apakah dibangunnya perangkat lunak (*software*) ini dapat membantu dalam meningkatkan proses produksi tanpa harus adanya *form order* ?

   Saya sangat setuju sekali, aplikasi ini sangat banyak membantu karena pesanan yang telah masuk sesuai dengan hasil cetakan dan tidak perlu lagi mengecek *form order* dengan nota pesanan untuk memprosesnya.

2. Apakah dibangunnya perangkat lunak (*software*) ini dapat meningkatkan kinerja dan efisiensi waktu pelayanan pesanan?

   Aplikasi ini benar-benar bagus. Karena waktu pelayanan pelanggan sangatlah penting agar pelanggan tidak kecewa dengan hasil dan prosesnya cepat.

3. Apakah perangkat lunak (*software*) ini memudahkan dalam membantu menyelesaikan kekurangan atau kelebihan bahan baku agar proses produksi bisa berjalan dengan lancar ?

   Saya setuju bahwa aplikasi ini membantu memudahkan proses produksi yang tadinya hanya memanfaatkan sisa stok dan tidak pernah memprediksi berapa banyak bahan baku yang dibutuhkan untuk proses produksi.

4. Apakah perangkat lunak (*software*) ini yang berbasiskan client-server memudahkan dalam mengelola pesanan tanpa penggunaan *form order*?

   Saya setuju penggunaan *client-server* sangat membantu memudahkan pendistribusian file yang telah didesain dan siap untuk dicetak karena semua file yang telah dibuat disimpan pada satu tempat tidak perlu mengecek kembali *form order*.

5. Apakah hasil dari penggunaan metode *MRP* ini berdampak besar pada kelancaran proses produksi ?

Setuju, karena penggunaan perhitungan kebutuhan bahan baku ini sangat berpengaruh besar pada kepuasan pelanggan. Karena perusahaan tidak perlu khawatir lagi kehabisan stok pada saat pelanggan sangat membutuhkan pesanannya cepat selesai.

4.3.1 Kesimpulan Hasil Pengujian Beta

Dari pengujian yang telah dilakukan dapat disimpulkan bahwa perangkat lunak yang dibangun ini :

1. Membantu mengatasi kesalahan-kesalahan dalam pencatatan transaksi.
2. Meningkatkan kinerja dan efisiensi waktu.
3. Membantu menyelesaikan masalah dalam mencari kebutuhan bahan baku yang akan digunakan.
4. Memudahkan dalam mengelola pesanan tanpa menggunakan *form order*.
5. Membantu mengatasi masalah dalam mencari kebutuhan bahan baku untuk memproses sebuah produk sesuai pesanan

Hasil pengujian ini baru secara persentase statistik biasa, dan mungkin saja ini kurang akurat. Untuk itu, apabila ingin mengetahui jawaban dengan kedalaman akurasi, maka perlu adanya suatu pengolahan data statistik dengan acuan dari data hasil wawancara yang sama.