BAB 4
IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1. Implementasi Sistem

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

4.1.1 Perangkat Keras Yang Digunakan

Spesifikasi perangkat keras yang digunakan dalam pembangunan aplikasi Sistem Informasi Monitoring Bengkel Kurnia Motor dapat dilihat pada tabel 4.1.

Tabel 4.1 Perangkat Keras yang Digunakan

No.	Perangkat Keras	Spesifikasi
1	Processor	*Speed* 2,2 GHz
2	RAM	2GB
3	Harddisk	500GB
4	Monitor	1366 x 768
5	VGA	1GB
6	LAN Card	10/100 Mbps
7	Printer	HP D1050
8	Koneksi Internet	384 kbps

4.1.2 Perangkat Lunak Yang Digunakan

Spesifikasi perangkat lunak yang digunakan dalam pembangunan sistem informasi monitoring bengkel Kurnia Motor dapat dilihat pada tabel 4.2 :

Tabel 4.2 Perangkat Lunak yang Digunakan

No.	**Perangkat Lunak**	**Spesifikasi**
1	Sistem Operasi	Windows 7 Professional 32bit
2	Bahasa Pemrograman	PHP 5.3.0
3	*DBMS*	MySQL 5.1.36
4	*Web Browser*	Mozilla Firefox 8.0.1, Google Chrome,

		Internet Explorer 9
5	*Code Editor*	Macromedia Dreamweaver CS5
6	*DFD*	Microsoft Visio 2007
7	*Web Server*	WAMP Server 2.0

4.2. Implementasi *Database*

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

1. Tabel user

```
CREATE TABLE IF NOT EXISTS `user` (
  `id` varchar(20) NOT NULL,
  `username` varchar(30) NOT NULL,
  `status` enum('admin','pegawai') NOT NULL,
  `password` varchar(100) NOT NULL,
  `nama` varchar(25) NOT NULL,
  `alamat` varchar(50) NOT NULL,
  `no_telpon` varchar(13) NOT NULL,
  `no_ktp` varchar(20) NOT NULL,
  `pertanyaan` varchar(100) NOT NULL,
  `jawaban` varchar(100) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. Tabel pelanggan

```
CREATE TABLE IF NOT EXISTS `pelanggan` (
  `id` int(20) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) NOT NULL,
  `alamat` varchar(100) NOT NULL,
  `no_telpon` varchar(14) NOT NULL,
  `utang` double NOT NULL,
  `tgl_daftar` date NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

3. Tabel supplier

```
CREATE TABLE IF NOT EXISTS `supplier` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) NOT NULL,
  `perusahaan` varchar(30) NOT NULL,
  `alamat` varchar(50) NOT NULL,
  `no_telpon` varchar(20) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

4. Tabel barang

```
CREATE TABLE IF NOT EXISTS `barang` (
  `kode_barang` varchar(12) NOT NULL,
  `KG` varchar(10) NOT NULL,
  `kodejenis` int(3) NOT NULL,
  `nama_barang` varchar(100) NOT NULL,
  `stok` int(11) NOT NULL,
  `merek` varchar(10) NOT NULL,
  `kategori` varchar(20) NOT NULL,
  `harga_beli` double NOT NULL,
  `harga_jual` double NOT NULL,
  `tanggal_entri` date NOT NULL,
  PRIMARY KEY (`kode_barang`),
  KEY `fk_barang_gudang1_idx` (`KG`),
  KEY `fk_barang_jenisbarang1_idx` (`kodejenis`),
  KEY `fk_barang_merek1_idx` (`merek`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5. Tabel transaksi_pembelian

```
CREATE TABLE IF NOT EXISTS `transaksi_pembelian` (
  `idtransaksi` int(11) NOT NULL AUTO_INCREMENT,
  `tanggal` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `id_pegawai` varchar(20) NOT NULL,
  `id_supplier` varchar(20) NOT NULL,
  `jumlah_total` double NOT NULL,
  PRIMARY KEY (`idtransaksi`),
  KEY `fk_id_pegawai` (`id_pegawai`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6. Tabel detail_transaksi_pembelian

```
CREATE TABLE IF NOT EXISTS
`detail_transaksi_pembelian` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `idtransaksi` int(11) NOT NULL,
  `kodebarang` varchar(12) NOT NULL,
  `jumlahbarang` int(11) NOT NULL,
  `hargasatuan` double NOT NULL,
  PRIMARY KEY (`id`),
  KEY
`fk_detail_transaksi_pembelian_transaksi_pembelian1_id
x` (`idtransaksi`),
  KEY `fk_detail_transaksi_pembelian_barang1_idx`
(`kodebarang`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

7. Tabel transaksi_penjualan

```
CREATE TABLE IF NOT EXISTS `transaksi_penjualan` (
  `idtransaksi` int(11) NOT NULL AUTO_INCREMENT,
  `tanggal` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `id_pegawai` varchar(20) NOT NULL,
  `id_pelanggan` varchar(20) NOT NULL,
  `jumlah_total` double NOT NULL,
  PRIMARY KEY (`idtransaksi`),
  KEY `fk_transaksi_penjualan_user_idx` (`id_pegawai`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
```

8. Tabel detail_transaksi_penjualan

```
CREATE TABLE IF NOT EXISTS
`detail_transaksi_penjualan` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `idtransaksi` int(11) NOT NULL,
  `kodebarang` varchar(9) NOT NULL,
  `jumlahbarang` int(11) NOT NULL,
  `hargasatuan` double NOT NULL,
  PRIMARY KEY (`id`),
  KEY
`fk_detail_transaksi_penjualan_transaksi_penjualan1_id
x` (`idtransaksi`),
  KEY `fk_detail_transaksi_penjualan_barang1_idx`
(`kodebarang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

9. Tabel log

```
CREATE TABLE IF NOT EXISTS `log` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `id_pegawai` varchar(20) NOT NULL,
  `tabel` varchar(25) NOT NULL,
  `aksi` varchar(100) NOT NULL,
  `tanggal` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY `fk_log_user1_idx` (`id_pegawai`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

10. Tabel peramalan

```
CREATE TABLE IF NOT EXISTS `peramalan` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `bulan_transaksi` date NOT NULL,
  `data_aktual` int(11) NOT NULL,
  `data_peramalan` int(11) NOT NULL,
```

```
  `idbarang` varchar(12) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `fk_peramalan_detail_transaksi_penjualan1_idx`
(`idbarang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

11. Tabel gudang

```
CREATE TABLE IF NOT EXISTS `gudang` (
  `kodegudang` varchar(3) NOT NULL,
  `nama_gudang` varchar(50) NOT NULL,
  `keterangan` varchar(100) NOT NULL,
  PRIMARY KEY (`kodegudang`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

12. Tabel merek

```
CREATE TABLE IF NOT EXISTS `merek` (
  `kodemerek` int(11) NOT NULL AUTO_INCREMENT,
  `merek` varchar(100) NOT NULL,
  `keterangan` varchar(100) NOT NULL,
  PRIMARY KEY (`kodemerek`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

13. Tabel jenisbarang

```
CREATE TABLE IF NOT EXISTS `jenisbarang` (
  `kodejenis` int(3) NOT NULL AUTO_INCREMENT,
  `namajenis` varchar(100) NOT NULL,
  `keterangan` varchar(100) NOT NULL,
  `nomorterakhir` int(3) NOT NULL DEFAULT '1',
  PRIMARY KEY (`kodejenis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

14. Implementasi *Foreign Key*

```
    Constraints for table `barang`
    --
    ALTER TABLE `barang`
      ADD CONSTRAINT `barang_ibfk_1` FOREIGN KEY
    (`KG`) REFERENCES `gudang` (`kodegudang`),
      ADD CONSTRAINT `barang_ibfk_2` FOREIGN KEY
    (`kodejenis`) REFERENCES `jenisbarang`
    (`kodejenis`),
      ADD CONSTRAINT `barang_ibfk_3` FOREIGN KEY
    (`merek`) REFERENCES `merek` (`kodemerek`);
    --
    -- Constraints for table
    `detail_transaksi_pembelian`
```

```
--
ALTER TABLE `detail_transaksi_pembelian`
  ADD CONSTRAINT
`detail_transaksi_pembelian_ibfk_1` FOREIGN KEY
(`idtransaksi`) REFERENCES `transaksi_pembelian`
(`idtransaksi`),
  ADD CONSTRAINT
`detail_transaksi_pembelian_ibfk_2` FOREIGN KEY
(`kodebarang`) REFERENCES `barang`
(`kode_barang`);
--
-- Constraints for table
`detail_transaksi_penjualan`
--
ALTER TABLE `detail_transaksi_penjualan`
  ADD CONSTRAINT
`detail_transaksi_penjualan_ibfk_1` FOREIGN KEY
(`kodebarang`) REFERENCES `barang`
(`kode_barang`),
  ADD CONSTRAINT
`detail_transaksi_penjualan_ibfk_2` FOREIGN KEY
(`idtransaksi`) REFERENCES `transaksi_penjualan`
(`idtransaksi`);
--
-- Constraints for table `log`
--
ALTER TABLE `log`
  ADD CONSTRAINT `log_ibfk_1` FOREIGN KEY
(`id_pegawai`) REFERENCES `user` (`id`);
--
-- Constraints for table `peramalan`
--
ALTER TABLE `peramalan`
  ADD CONSTRAINT `peramalan_ibfk_1` FOREIGN KEY
(`idbarang`) REFERENCES `barang` (`kode_barang`);
--
-- Constraints for table `transaksi_pembelian`
--
ALTER TABLE `transaksi_pembelian`
  ADD CONSTRAINT `transaksi_pembelian_ibfk_1`
FOREIGN KEY (`id_pegawai`) REFERENCES `user`
(`id`),
  ADD CONSTRAINT `transaksi_pembelian_ibfk_2`
FOREIGN KEY (`id_supplier`) REFERENCES `supplier`
(`id`);
--
-- Constraints for table `transaksi_penjualan`
--
ALTER TABLE `transaksi_penjualan`
  ADD CONSTRAINT `transaksi_penjualan_ibfk_1`
FOREIGN KEY (`id_pegawai`) REFERENCES `user`
```

```
      (`id`),
  ADD CONSTRAINT `transaksi_penjualan_ibfk_2` FOREIGN
KEY (`id_pelanggan`) REFERENCES `pelanggan` (`id`);
```

4.3. Implementasi Antarmuka

Implementasi antar muka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Terdapat dua macam tampilan antarmuka, yaitu antarmuka admin dan antarmuka pegawai.

Implementasi antarmuka pegawai dijelaskan pada tabel 4.3 dan implementasi antarmuka pegawai dijelaskan pada tabel 4.4.

Tabel 4.3 Implementasi Antarmuka Untuk Admin

No.	Menu	Deskripsi	Nama File
1.	Login	Halaman untuk melakukan login agar bisa masuk beranda sistem.	index.php
2.	Beranda	Halaman awal yang digunakan untuk menyajikan pilihan yang ada pada sistem.	menu.php
3.	Data Barang	Halaman yang menampilkan informasi data barang.	barang.php
4.	Data Pegawai	Digunakan untuk menampilkan, menambahkan, mengubah dan menghapus data pegawai.	pegawai.php
5.	Data Supplier	Digunakan untuk menampilkan informasi data supplier.	supplier.php
6.	Data Pelanggan	Digunakan untuk menampilkan informasi data pelanggan.	pelanggan.php
7.	Data Gudang	Digunakan untuk menampilkan informasi data gudang	gudang.php
8.	Data Merek	Digunakan untuk menampilkan informasi data merek	merek.php
9.	Data Jenis Barang	Digunakan untuk menampilkan informasi data jenis barang	jenis.php
10.	Laporan Penjualan	Digunakan untuk menampilkan informasi	laporanpenjualan.php

		laporan penjualan perbulan	
11.	Laporan pembelian	Digunakan untuk menampilkan informasi laporan pembelian perbulan	laporanpembelian.php
12.	Log	Digunakan untuk menampilkan informasi histori kegiatan yang dilakukan sistem	log.php
13.	Peramalan	Digunakan untuk menampilkan informasi nilai ramalan setiap bulan	peramalan.php
14.	Barang kritis	Digunakan untuk menampilkan informasi barang yang hampir habis atau telah habis	laporanbarang.php
15.	Daftar faktur penjualan	Digunakan untuk menampilkan informasi faktur penjualan secara umum	daftarfakturpenjualan.php
16.	Daftar faktur pembelian	Digunakan untuk menampilkan informasi faktur pembelian secara umum	daftafakturpembelian.php
17.	Faktur Penjualan	Digunakan untuk menampilkan informasi secara detail faktur penjualan setiap transaksi	fakturpenjualan.php
18.	Faktur pembelian	Digunakan untuk menampilkan informasi secara detail faktur pembelian setiap transaksi	fakturpembelian.php
19.	Logout	Digunakan untuk kelaur dari sistem dan kembali ke halaman login	logout.php

Tabel 4.4 Implementasi Antarmuka Untuk Pegawai

No.	Menu	Deskripsi	Nama File
1.	Login	Halaman untuk melakukan login agar bisa masuk beranda sistem.	index.php
2.	Beranda	Halaman awal yang digunakan untuk menyajikan pilihan yang ada pada sistem.	menu.php

3.	Data Barang	Halaman yang digunakan untuk menampilkan, menambahkan, mengubah dan menghapus data barang.	barang.php
4.	Data Supplier	Digunakan untuk menampilkan, menambahkan, mengubah dan menghapus data supplier.	supplier.php
5.	Data Pelanggan	Digunakan untuk menampilkan, menambahkan, mengubah dan menghapus data pelanggan.	pelanggan.php
6.	Data Gudang	Digunakan untuk menampilkan, menambahkan, mengubah dan menghapus data gudang	gudang.php
7.	Data Merek	Digunakan untuk menampilkan, menambahkan, mengubah dan menghapus data merek	merek.php
8.	Data Jenis Barang	Digunakan untuk menampilkan, menambahkan, mengubah dan menghapus data jenis barang	jenis.php
9.	Transaksi pembelian	Digunakan untuk melakukan transaksi pembelian	transaksipembelian.php
10.	Transaksi penjualan	Digunakan untuk melakukan transaksi pembelian	transaksipenjualan.php
11.	Faktur Penjualan	Digunakan untuk menampilkan informasi secara detail faktur penjualan setiap transaksi	fakturpenjualan.php
12.	Faktur pembelian	Digunakan untuk menampilkan informasi secara detail faktur pembelian setiap transaksi	fakturpembelian.php

13	Logout	Digunakan untuk kelaur dari sistem dan kembali ke halaman login	logout.php

4.4. Pengujian Perangkat Lunak

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

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

4.4.1. Skenario Pengujian *alpha (Black box)*

Skenario pengujian *blackbox* menjelaskan skenario pengujian Sistem Informasi Monitoring bengkel Kurnia Motor tabel 4.5.

Tabel 4.5 Tabel Skenario Pengujian

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Login pengguna	*Black Box*
Data Barang	Penambahan Pengubahan penghapusan	*Black Box*
Data jenis barang	Penambahan Pengubahan penghapusan	*Black Box*
Data merek	Penambahan Pengubahan penghapusan	*Black Box*
Data pegawai	Penambahan Pengubahan penghapusan	*Black Box*
Data pelanggan	Penambahan Pengubahan penghapusan	*Black Box*
Data supplier	Penambahan Pengubahan penghapusan	*Black Box*
Transaksi pembelian	Penambahan	*Black Box*
Transaksi penjualan	Penambahan	*Black Box*
Laporan Penjualan	*Filtering*	*Black Box*
Laporan pembelian	*Filtering*	*Black Box*

4.4.2. Kasus Dan Hasil Pengujian

Pengujian dilakukan dengan menguji setiap proses untuk kemungkinan kesalahan yang terjadi.

4.4.2.1 Pengujian Login

Login digunakan untuk penentuan hak akses pengguna sebagai admin atau pegawai. Tabel pengujian login dapat dilihat pengujian data benar login pada tabel 4.6 berikut ini :

Tabel 4.6. Pengujian Login Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username **Password** Contoh Masukan 1. Data login admin Username : admin Password : admin 2. Data login pegawai Username : pegawai Password : pegawai	Mengisikan data login yang sudah terdaftar.	Data login benar dan akan masuk ke masing-masing *interface* yang telah ditentukan sesuai dengan data login.	[√]diterima [ ]ditolak

Berikut ini Pengujian data salah login yang dapat dilihat pada tabel 4.7 sebagai berikut :

Tabel 4.7.Pengujian Login Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username **Password** Contoh Masukan 1. Data login admin Username : admin Password : (kosong) 2. Data login	Menampilkan pesan "Username dan Password salah".	Dapat menampilkan pesan kesalahan "username dan password salah"	[√]diterima [ ]ditolak

pegawai Username : pegawai Password : (kosong)			
Username **Password** Contoh Masukan 1. Data login admin Username : 123456 Password : user1 2. Data login pegawai Username : 1234abc Password : pegawai	Menampilkan pesan "Username dan Password salah".	Dapat menampilkan pesan kesalahan "username dan password salah"	[√]diterima [ ]ditolak

4.4.2.2 Pengujian Penambahan Data Barang

Pengujian penambahan data barang dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.8.

Tabel 4.8. Pengujian Penambahan Data Barang Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Barang Contoh Masukan : Kategori : Suku cadang Motor Jenis barang : Accu Kode gudang : B13 Merek : GS Premium Nama barang : Accu GM3 3B GS PREMIUM Stok : 10 Harga beli : 80000 Harga jual : 90000	Mengisikan data barang baru dan menyimpan data.	Data barang dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini Pengujian data salah penambahan data barang yang salah dapat dilihat pada tabel 4.9.

Tabel 4.9 Pengujian Penambahan Data Barang Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama Barang	Menampilkan pesan "Nama Barang Harus Diisi" pada *field* Nama Barang yang kosong.	Dapat menampilkan pesan kesalahan "Nama Barang Harus Diisi" pada *field* Nama Barang yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Stok	Menampilkan pesan "Stok Harus Diisi" pada *field* stok yang kosong.	Dapat menampilkan pesan kesalahan "Stok Harus Diisi" pada *field* Stok yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Harga Beli	Menampilkan pesan "harga beli harus diisi" untuk field harga beli yang kosong.	Dapat menampilkan pesan kesalahan "Harga Beli Harus Diisi" pada *field* Harga Beli yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Harga Jual	Menampilkan pesan "harga jual harus diisi" untuk field harga jual yang kosong.	Dapat menampilkan pesan kesalahan "Harga Jual Harus Diisi" pada *field* Harga Jual yang kosong.	[√]diterima [ ]ditolak
Nilai Masukan Stok Tidak Berupa angka	Menampilkan pesan "Stok Harus Berupa Angka"	Dapat menampilkan pesan kesalahan "Stok Harus Berupa Angka".	[√]diterima [ ]ditolak
Nilai masukan Harga Beli tidak berupa angka.	Menampilkan pesan "Harga Beli Harus Berupa Angka".	Dapat menampilkan pesan kesalahan "Harga Beli Harus Berupa Ang-ka".	[√]diterima [ ]ditolak
Nilai masukan Harga Jual tidak berupa angka.	Menampilkan pesan "Harga Jual Harus Berupa Angka".	Dapat menampilkan pesan kesalahan "Stok Harus Berupa Angka".	[√]diterima [ ]ditolak

4.4.2.3 Pengujian Pengubahan Data Barang

Pengujian pengubahan input data barang yang benar dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.10.

Tabel 4.10. Pengujian Pengubahan Input Data Barang Yang Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Barang Contoh Masukan : Kategori : Suku cadang Motor Jenis barang : Accu Kode gudang : B13 Merek : GS Premium Nama barang : Accu GM3 3B	Mengisikan data barang baru dan menyimpan data.	Data barang dapat tersimpan.	[√]diterima [ ]ditolak

GS PREMIUM Stok : 10 Harga beli : 80000 Harga jual : 90000			

Berikut ini Pengujian pengubahan input data barang yang salah dapat dilihat pada tabel 4.11.

Tabel 4.11 Pengujian Pengubahan Input Data Barang Yang Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama Barang	Menampilkan pesan "Nama Barang Harus Diisi" pada *field* Nama Barang yang kosong.	Dapat menampilkan pesan kesalahan "Nama Barang Harus Diisi" pada *field* Nama Barang yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Stok	Menampilkan pesan “Stok Harus Diisi” pada *field* stok yang kosong.	Dapat menampilkan pesan kesalahan "Stok Harus Diisi" pada *field* Stok yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Harga Beli	Menampilkan pesan “harga beli harus diisi” untuk field harga beli yang kosong.	Dapat menampilkan pesan kesalahan "Harga Beli Harus Diisi" pada *field* Harga Beli yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Harga Jual	Menampilkan pesan “harga jual harus diisi” untuk field harga jual yang kosong.	Dapat menampilkan pesan kesalahan "Harga Jual Harus Diisi" pada *field* Harga Jual yang kosong.	[√]diterima [ ]ditolak
Nilai Masukan Stok Tidak Berupa angka	Menampilkan pesan "Stok Harus Berupa Angka"	Dapat menampilkan pesan kesalahan "Stok Harus Berupa Angka".	[√]diterima [ ]ditolak
Nilai masukan Harga Beli tidak berupa angka.	Menampilkan pesan "Harga Beli Harus Berupa Angka".	Dapat menampilkan pesan kesalahan "Harga Beli Harus Berupa Ang-ka".	[√]diterima [ ]ditolak
Nilai masukan Harga Jual tidak berupa angka.	Menampilkan pesan "Harga Jual Harus Berupa Angka".	Dapat menampilkan pesan kesalahan "Stok Harus Berupa Angka".	[√]diterima [ ]ditolak

4.4.2.4 Pengujian Penghapusan Data Barang

Pengujian penghapusan data barang dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.12.

Tabel 4.12. Pengujian Penghapusan Data Barang

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Barang Contoh Masukan : Mengklik *link* hapus	Menampilkan pesan bertuliskan "Anda yakin ingin menghapus data tersebut?".	Dapat menampilkan pesan bertuliskan"Anda yakin ingin menghapus data tersebut?".	[√]diterima [ ]ditolak
Jika "ya"	Menghapus data barang	Dapat menghapus data barang	[√]diterima [ ]ditolak
Jika "Tidak"	Kembali ke halaman data barang	Dapat Kembali ke halaman data barang	[√]diterima [ ]ditolak

4.4.2.5 Pengujian Penambahan Data Jenis Barang

Pengujian penambahan input data jenis barang dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.13.

Tabel 4.13 Pengujian Penambahan Input Data Jenis Barang Yang Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Barang Contoh Masukan : Nama Jenis : Accu Keterangan : Sumber Listrik Pada Kendaraan	Mengisikan data jenis barang baru dan menyimpan data.	Data jenis barang dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian data salah penambahan data Jenis Barang dapat dilihat pada tabel 4.14 sebagai berikut :

Tabel 4.14 Pengujian Penambahan Input Data Jenis Barang Yang Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama Jenis	Menampilkan pesan "Nama Jenis Harus Diisi" pada *field* Nama Jenis yang kosong.	Dapat menampilkan pesan kesalahan "Nama Jenis Harus Diisi" pada *field* Nama jenis yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Keterangan	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak

4.4.2.6 Pengujian Pengubahan Data Jenis Barang

Pengujian pengubahan Input data jenis barang yang benar dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.15.

Tabel 4.15 Pengujian Pengubahan Input Data Jenis Barang Yang Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Barang Contoh Masukan : Nama Jenis : Accu Keterangan : Sumber Listrik Pada Kendaraan	Mengisikan data jenis barang baru dan menyimpan data.	Data jenis barang dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian pengubahan Input data Jenis Barang yang salah dapat dilihat pada tabel 4.16

Tabel 4.16 Pengujian Pengubahan Input Data Jenis Barang Yang Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama Jenis	Menampilkan pesan "Nama Jenis Harus Diisi" pada *field* Nama Jenis yang kosong.	Dapat menampilkan pesan kesalahan "Nama Jenis Harus Diisi" pada *field* Nama jenis yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Keterangan	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak

4.4.2.7 Pengujian Penghapusan Data Jenis Barang

Pengujian penghapusan data jenis barang dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.17.

Tabel 4.17 Pengujian Penghapusan Data Jenis Barang

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data Barang Contoh Masukan : Mengklik *link* hapus	Menampilkan pesan bertuliskan “Anda yakin ingin menghapus data tersebut?”.	Dapat menampilkan pesan bertuliskan“Anda yakin ingin menghapus data tersebut?”.	[√]diterima [ ]ditolak
Jika “ya”	Menghapus data jenis barang	Dapat menghapus data jenis barang	[√]diterima [ ]ditolak

Jika “Tidak”	Kembali ke halaman data jenis barang	Dapat Kembali ke halaman data jenis barang	[√]diterima [ ]ditolak

4.4.2.8 Pengujian Penambahan data Merek

Pengujian penambahan data Merek dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.18.

Tabel 4.18. Pengujian Penambahan Input Data Merek Yang Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Nama Merek : GS Keterangan : Merek Accu	Mengisikan data merek baru dan menyimpan data.	Data Merek dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian data salah penambahan data merek dapat dilihat pada tabel 4.19 sebagai berikut :

Tabel 4.19 Pengujian Penambahan Input Data Merek Yang Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama Merek	Menampilkan pesan "Nama Merek Harus Diisi" pada *field* Nama Merek yang kosong.	Dapat menampilkan pesan kesalahan "Nama Merek Harus Diisi" pada *field* Nama Merek yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Keterangan	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak

4.4.2.9 Pengujian Pengubahan Data Merek

Pengujian pengubahan input data Merek yang benar dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.20.

Tabel 4.20. Pengujian Pengubahan Input Data Merek Yang Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Nama Merek : GS Keterangan : Merek Accu	Mengisikan data merek baru dan menyimpan data.	Data Merek dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian data salah pengubahan Input data merek yang salah dapat dilihat pada tabel 4.21 sebagai berikut :

Tabel 4.21 Pengujian Pengubahan Input Data Merek Yang Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama Merek	Menampilkan pesan "Nama Merek Harus Diisi" pada *field* Nama Merek yang kosong.	Dapat menampilkan pesan kesalahan "Nama Merek Harus Diisi" pada *field* Nama Merek yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Keterangan	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak

4.4.2.10 Pengujian Penghapusan Data Merek

Pengujian penghapusan data merek dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.22.

Tabel 4.22. Pengujian Penghapusan Data Merek

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Mengklik *link* hapus	Menampilkan pesan bertuliskan "Anda yakin ingin menghapus data tersebut?".	Dapat menampilkan pesan bertuliskan"Anda yakin ingin menghapus data tersebut?".	[√]diterima [ ]ditolak
Jika "ya"	Menghapus data Merek	Dapat menghapus data Merek	[√]diterima [ ]ditolak
Jika "Tidak"	Kembali ke halaman data Merek	Dapat Kembali ke halaman data Merek	[√]diterima [ ]ditolak

4.4.2.11 Pengujian Penambahan Data Pegawai

Berikut ini pengujian penambahan Data Pegawai dilakukan oleh Admin yang telah login pada sistem dapat dilihat pada pada tabel 4.23.

Tabel 4.23. Pengujian Penambahan Data Pegawai Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Nama Pengguna: Pegawai1 Kata Kunci : Pass1 Status : Pegawai	Mengisikan Data Pegawai baru dan menyimpan data.	Data Pegawai dapat tersimpan.	[√]diterima [ ]ditolak

Nama Lengkap : Puput Puspita Alamat : Kalijati No. Telepon : No. KTP :			

Berikut ini pengujian data salah penambahan data pegawai yang dapat dilihat pada tabel 4.24 sebagai berikut :

Tabel 4.24 Pengujian Penambahan Input Data Pegawai Yang Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama Pengguna	Menampilkan pesan " Nama Pengguna Harus Diisi" pada *field* Nama Pengguna yang kosong.	Dapat menampilkan pesan kesalahan " Nama Pengguna Harus Diisi" pada *field* Nama Pengguna yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Kata Kunci	Menampilkan pesan " Kata Kunci Harus Diisi" pada *field* Kata Kunci yang kosong.	Dapat menampilkan pesan kesalahan " Kata Kunci Harus Diisi" pada *field* Kata Kunci yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Status	Menampilkan pesan "Status Harus Diisi" pada *field* Status yang kosong.	Dapat menampilkan pesan kesalahan " Status Harus Diisi" pada *field* Status yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Nama Lengkap	Menampilkan pesan "Nama Lengkap Harus Diisi" pada *field* Nama Lengkap yang kosong.	Dapat menampilkan pesan kesalahan "Nama Lengkap Harus Diisi" pada *field* Nama Lengkap yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Alamat	Menampilkan pesan " Alamat Harus Diisi" pada *field* Alamat yang kosong.	Dapat menampilkan pesan kesalahan " Alamat Harus Diisi" pada *field* Alamat yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi No. Telepon	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak
Tidak Mengisi No. KTP	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak

4.4.2.12 Pengujian Pengubahan Data Pegawai

Berikut ini pengujian pengubahan input data pegawai yang Benar dilakukan oleh Admin yang telah login pada sistem dapat dilihat pada pada tabel 4.25.

Tabel 4.25. Pengujian Pengubahan Input Data Pegawai Yang Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Nama Pengguna: Pegawai1 Kata Kunci : Pass1 Status : Pegawai Nama Lengkap : Puput Puspita Alamat : Kalijati No. Telepon : No. KTP :	Mengisikan Data Pegawai baru dan menyimpan data.	Data Pegawai dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian pengubahan input data pegawai yang salah dapat dilihat pada tabel 4.26 sebagai berikut :

Tabel 4.26 Pengujian Pengubahan Input Data Pegawai Yang Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama Pengguna	Menampilkan pesan " Nama Pengguna Harus Diisi" pada *field* Nama Pengguna yang kosong.	Dapat menampilkan pesan kesalahan " Nama Pengguna Harus Diisi" pada *field* Nama Pengguna yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Kata Kunci	Menampilkan pesan " Kata Kunci Harus Diisi" pada *field* Kata Kunci yang kosong.	Dapat menampilkan pesan kesalahan " Kata Kunci Harus Diisi" pada *field* Kata Kunci yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Status	Menampilkan pesan "Status Harus Diisi" pada *field* Status yang kosong.	Dapat menampilkan pesan kesalahan " Status Harus Diisi" pada *field* Status yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Nama Lengkap	Menampilkan pesan "Nama Lengkap Harus Diisi" pada *field* Nama Lengkap yang kosong.	Dapat menampilkan pesan kesalahan "Nama Lengkap Harus Diisi" pada *field* Nama Lengkap yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Alamat	Menampilkan pesan " Alamat Harus Diisi" pada *field* Alamat yang kosong.	Dapat menampilkan pesan kesalahan " Alamat Harus Diisi" pada *field* Alamat yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi No. Telepon	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak
Tidak Mengisi No. KTP	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak

4.4.2.13 Pengujian Penghapusan Data Pegawai

Berikut ini pengujian penghapusan Data Pegawai dilakukan oleh Admin yang telah login pada sistem dapat dilihat pada pada tabel 4.27.

Tabel 4.27. Pengujian Penghapusan Data Pegawai

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Mengklik *link* hapus	Menampilkan pesan bertuliskan "Anda yakin ingin menghapus data tersebut?".	Dapat menampilkan pesan bertuliskan"Anda yakin ingin menghapus data tersebut?".	[√]diterima [ ]ditolak
Jika "ya"	Menghapus data pegawai	Dapat menghapus data pegawai	[√]diterima [ ]ditolak
Jika "Tidak"	Kembali ke halaman data pegawai.	Dapat Kembali ke halaman data pegawai	[√]diterima [ ]ditolak

4.4.2.14 Pengujian Penambahan Data Pelanggan

Berikut ini pengujian penambahan Data Pelanggan dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.28 sebagai berikut:

Tabel 4.28. Pengujian Penambahan Data Pelanggan Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Nama : Rena Yulianti Alamat : Kalijati No.Telepon : 085220380566	Mengisikan Data Pelanggan baru dan menyimpan data.	Data Pelanggan dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian data salah penambahan data pelanggan yang dapat dilihat pada tabel 4.29 sebagai berikut ;

Tabel 4.29 Pengujian Penambahan Data Pelanggan Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama	Menampilkan pesan "Nama Harus Diisi" pada *field* Nama yang kosong.	Dapat menampilkan pesan kesalahan "Nama Harus Diisi" pada *field* Nama yang kosong.	[√]diterima [ ]ditolak

Tidak Mengisi Alamat	Menampilkan pesan "Alamat Harus Diisi" pada *field* Alamat yang kosong.	Dapat menampilkan pesan kesalahan "Alamat Harus Diisi" pada *field* Alamat yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi No.Telepon	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak

4.4.2.15 Pengujian Pengubahan Data Pelanggan

Berikut ini pengujian pengubahan input data pelanggan yang benar dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.30 sebagai berikut:

Tabel 4.30. Pengujian Pengubahan Data Pelanggan Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Nama : Rena Yulianti Alamat : Kalijati No.Telepon : 085220380566	Mengisikan Data Pelanggan baru dan menyimpan data.	Data Pelanggan dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian data salah pengubahan data pelanggan yang dapat dilihat pada tabel 4.31 sebagai berikut ;

Tabel 4.31 Pengujian Pengubahan Input Data Pelanggan Yang Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama	Menampilkan pesan "Nama Harus Diisi" pada *field* Nama yang kosong.	Dapat menampilkan pesan kesalahan "Nama Harus Diisi" pada *field* Nama yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Alamat	Menampilkan pesan "Alamat Harus Diisi" pada *field* Alamat yang kosong.	Dapat menampilkan pesan kesalahan "Alamat Harus Diisi" pada *field* Alamat yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi No.Telepon	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak

4.4.2.16 Pengujian Penghapusan Data Pelanggan

Berikut ini pengujian penghapusan data pelanggan dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.32 sebagai berikut:

Tabel 4.32. Pengujian Penghapusan Data Pelanggan

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Mengklik *link* hapus	Menampilkan pesan bertuliskan "Anda yakin ingin menghapus data tersebut?".	Dapat menampilkan pesan bertuliskan"Anda yakin ingin menghapus data tersebut?".	[√]diterima [ ]ditolak
Jika "ya"	Menghapus data pelanggan	Dapat menghapus data pelanggan	[√]diterima [ ]ditolak
Jika "Tidak"	Kembali ke halaman data pelanggan.	Dapat Kembali ke halaman data pelanggan	[√]diterima [ ]ditolak

4.4.2.17 Pengujian Penambahan Data Supplier

Pengujian penambahan Data Supplier dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.33.

Tabel 4.33. Pengujian Penambahan Data Supplier Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Nama : Bernard Alamat : Bandung Perusahaan : Top1 No.Telepon : 022-6754 675	Mengisikan Data Supplier baru dan menyimpan data.	Data Supplier dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian data salah penambahan data Supplier yang dapat dilihat pada tabel 4.34 sebagai berikut :

Tabel 4.34 Pengujian Penambahan Data Supplier Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama	Menampilkan pesan "Nama Harus Diisi" pada	Dapat menampilkan pesan kesalahan "Nama Harus	[√]diterima [ ]ditolak

	field Nama yang kosong.	Diisi" pada *field* Nama yang kosong.	
Tidak Mengisi Alamat	Menampilkan pesan "Alamat Harus Diisi" pada *field* Alamat yang kosong.	Dapat menampilkan pesan kesalahan "Alamat Harus Diisi" pada *field* Alamat yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Nama Perusahaan	Menampilkan pesan "Nama Perusahaan Harus Diisi" pada *field* Nama Perusahaan yang kosong.	Menampilkan pesan " Nama Perusahaan Harus Diisi" pada *field* Nama Perusahaan yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi No.Telepon	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak

4.4.2.18 Pengujian Pengubahan Data Supplier

Pengujian pengubahan input data supplier yang benar dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.35 sebagai berikut :

Tabel 4.35. Pengujian Pengubahan Input Data Supplier Yang Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Nama : Bernard Alamat : Bandung Perusahaan : Top1 No.Telepon : 022-6754 675	Mengisikan Data Supplier baru dan menyimpan data.	Data Supplier dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian data salah pengubahan input data Supplier yang salah dapat dilihat pada tabel 4.36 sebagai berikut :

Tabel 4.36 Pengujian Penambahan Data Supplier Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tidak Mengisi Nama	Menampilkan pesan "Nama Harus Diisi" pada *field* Nama yang kosong.	Dapat menampilkan pesan kesalahan "Nama Harus Diisi" pada *field* Nama yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Alamat	Menampilkan pesan "Alamat Harus Diisi" pada *field* Alamat yang kosong.	Dapat menampilkan pesan kesalahan "Alamat Harus Diisi" pada *field* Alamat yang kosong.	[√]diterima [ ]ditolak
Tidak Mengisi Nama Perusahaan	Menampilkan pesan "Nama Perusahaan Harus	Menampilkan pesan " Nama Perusahaan Harus	[√]diterima [ ]ditolak

	Diisi" pada *field* Nama Perusahaan yang kosong.	Diisi" pada *field* Nama Perusahaan yang kosong.	
Tidak Mengisi No.Telepon	Tidak Menampilkan pesan apapun pada *field* keterangan.	Tidak Menampilkan pesan apapun pada *field* keterangan.	[√]diterima [ ]ditolak

4.4.2.19 Pengujian Penghapusan Data Supplier

Pengujian penghapusan data supplier dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.37 sebagai berikut :

Tabel 4.37. Pengujian Penghapusan Data Supplier

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Mengklik *link* hapus	Menampilkan pesan bertuliskan "Anda yakin ingin menghapus data tersebut?".	Dapat menampilkan pesan bertuliskan"Anda yakin ingin menghapus data tersebut?".	[√]diterima [ ]ditolak
Jika "ya"	Menghapus data supplier	Dapat menghapus data Supplier	[√]diterima [ ]ditolak
Jika "Tidak"	Kembali ke halaman data supplier	Dapat Kembali ke halaman data supplier	[√]diterima [ ]ditolak

4.4.2.20 Pengujian Transaksi Pembelian

Pengujian Transaksi Pembelian dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.38 sebagai berikut :

Tabel 4.38. Pengujian Transaksi Pembelian Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Supplier : Bernard Kode Barang : 101001003	Mengisikan Data Transaksi Pembelian dan menyimpan data.	Data Transaksi Pembelian dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian data salah pengujian transaksi pembelian dapat dilihat pada tabel 4.39 sebagai berikut :

Tabel 4.39 Pengujian Transaksi Pembelian Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi Nama Supplier yang tidak terdaftar	Menampilkan pesan Nama Supplier tidak ditemukan.	Menampilkan pesan Nama Supplier tidak ditemukan.	[√]diterima [ ]ditolak
Mengisi Kode Barang yang salah.	Menampilkan pesan Kode Barang tidak ditemukan.	Menampilkan pesan Kode Barang tidak ditemukan.	[√]diterima [ ]ditolak

4.4.2.21 Pengujian Transaksi Penjualan

Pengujian Transaksi Penjualan dilakukan oleh pegawai yang telah login pada sistem dapat dilihat pada pada tabel 4.40.

Tabel 4.40. Pengujian Transaksi Penjualan Data Benar

Kasus dan hasil uji (Data Benar)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Contoh Masukan : Pelanggan : Rifki Kode Barang : 1010010032	Mengisikan Data Transaksi Penjualan dan menyimpan data.	Data Transaksi Penjualan dapat tersimpan.	[√]diterima [ ]ditolak

Berikut ini pengujian data salah pengujian transaksi penjualan dapat dilihat pada tabel 4.41 sebagai berikut :

Tabel 4.41 Pengujian Transaksi Penjualan Data Salah

Kasus dan hasil uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi Nama Pelanggan yang tidak terdaftar	Menampilkan pesan Nama Pelanggan tidak ditemukan.	Menampilkan pesan Nama Pelanggan tidak ditemukan.	[√]diterima [ ]ditolak
Mengisi Kode Barang yang salah.	Menampilkan pesan Kode Barang tidak ditemukan.	Menampilkan pesan Kode Barang tidak ditemukan.	[√]diterima [ ]ditolak

4.4.2.22 Pengujian *Filtering*

Berikut ini pengujian filtering laporan pembelian dapat dilihat pada tabel 4.42 sebagai berikut :

Tabel 4.42. Pengujian *Filtering* Laporan Pembelian

Kasus dan hasil uji			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data pilih menu *filtering* perbulan	Pemilihan *filtering* perbulan	Menampilkan *filtering* data sesuai	[√]diterima [ ]ditolak

		pemilihan bulan yang diinginkan	
Data pilih menu *filtering* pertahun	Pemilihan *filtering* perbulan	Menampilkan *filtering* data sesuai pemilihan tahun yang diinginkan	[√]diterima [ ]ditolak

Berikut ini pengujian filtering laporan penjualan dapat dilihat pada tabel 4.43 sebagai berikut :

Tabel 4.43. Pengujian *Filtering* Laporan Penjualan

Kasus dan hasil uji			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data pilih menu *filtering* perbulan	Pemilihan *filtering* perbulan	Menampilkan *filtering* data sesuai pemilihan bulan yang diinginkan	[√]diterima [ ]ditolak
Data pilih menu *filtering* pertahun	Pemilihan *filtering* perbulan	Menampilkan *filtering* data sesuai pemilihan tahun yang diinginkan	[√]diterima [ ]ditolak

4.4.3. Kesimpulan Pengujian *Black Box*

Berdasarkan hasil pengujian dengan kasus *sample* uji yang telah dilakukan memberikan kesimpulan bahwa pada proses sudah benar. Penyaringan kesalahan proses dalam bentuk arahan tampilan halaman pesan sudah cukup maksimal. Secara fungsional sistem sudah dapat menghasilkan *output* yang diharapkan.

4.5. Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan, dengan menggunakan teknik wawancara kepuasan *user* terhadap aplikasi yang telah dibangun, dengan kandungan point syarat *user friendly* serta apakah sesuai dengan yang diharapkan. Adapun metode penilaian pengujian yang digunakan yaitu metode kuantitatif berdasarkan data wawancara dari *user*.

4.5.1. Skenario Pengujian Beta

Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung ditempat penelitian. Penelitian ini dilakukan di lingkungan bengkel Kurnia Motor supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif.penyelesaian permasalahan yang telah dijelaskan di awal.

Adapun beberapa poin yang diteliti adalah :

a. Tampilan antar muka yang *user friendly*.
b. Sistem informasi dapat menghasilkan angka ramalan pembelian.
c. Sistem informasi dapat memenuhi kebutuhan monitoring dan evaluasi data barang.
d. Memudahkan pengguna untuk pendataan data barang.
e. Kelayakan sistem informasi untuk digunakan**.**

Pengujian beta yang dilakukan terhadap perusahaan bengkel Kurnia Motor adalah dengan menggunakan teknik wawancara, berikut ini daftar pertanyaan dan hasil wawancara yang diberikan kepada admin (pemilik) Bengkel Kurnia Motor.:

1. Bagaimana pendapat anda mengenai Sistem Informasi ini apakah lebih memudahkan anda dalam memonitoring dan evaluasi data barang ?

   Jawaban :

   Menurut admin, sistem informasi ini sangat memudahkan penngecekan karena banyaknya data barang dan monitoringnya mampu memantau data transaksi pembelian dan data transaksi penjualan.
2. Bagaimana pendapat anda mengenai informasi yang diberikan oleh Sistem Informasi ini dengan penerapan metode peramalan ?

   Jawaban :

   Cukup membantu, terutama untuk menentukan pembelian barang yang kurang populer
3. Bagaimana laporan yang dihasilkan sistem informasi ini apakah sudah informatif?

   Jawaban :

   Laporan yang diberikan sudah cukup memenuhi apa yang saya inginkan.

4. Bagaimana pendapat anda mengenai tampilan *(interface)* Sistem Informasi ini apakah mudah digunakan dan sesuai dengan instruksi?
   Jawaban :
   Aplikasi mudah digunakan, karena memakai bahasa Indonesia yang mudah untuk dipahami sehingga mudah untuk digunakan juga.
5. Bagaimana pendapat anda mengenai tampilan *(interface)* Sistem Informasi ini apakah cukup menarik dan mudah dipahami?
   Jawaban :
   sudah cukup menarik dan mudah dipahami.
6. Bagaimana pendapat anda mengenai Sistem Informasi ini apakah sudah layak untuk digunakan ?
   Jawaban :
   Sudah cukup membantu sehingga mengurangi pekerjaan untuk membuat laporan.

Berikut hasil wawancara dengan salah satu pegawai di bengkel Kurnia Motor

1. Apakah melakukan transaksi dengan menggunakan aplikasi ini lebih mudah dilakukan?
   Jawaban :
   Transaksi cukup mudah, hanya pencetakan struk memakan waktu lama.
2. Bagaimana pendapat anda mengenai tampilan *(interface)* Sistem Informasi ini apakah mudah digunakan dan sesuai dengan instruksi?
   Jawaban :
   Mudah digunakan karena menggunakan bahasa Indonesia.
3. Bagaimana pendapat anda mengenai tampilan *(interface)* Sistem Informasi ini apakah cukup menarik dan mudah dipahami?
   Jawaban :
   sudah cukup menarik dan mudah dipahami.
4. Bagaimana pendapat anda mengenai Sistem Informasi ini apakah sudah layak untuk digunakan ?
   Jawaban :

Cukup layak, tetapi masih mempunyai kelemahan dalam melakukan transaksi. Terutama transaksi penjualan.

4.5.2. Kesimpulan Pengujian Beta

Berdasarkan pengujian beta diatas, bahwa dapat diambil kesimpulan bahwa:

1. Sistem informasi cukup membantu admin, dalam memonitoring dan evaluasi data barang.
2. Tampilan (*Interface*) Sistem informasi ini cukup menarik.
3. Sistem informasi ini dapat menghasilkan ramalan pembelian bulan berikutnya yang dapat membantu pemilik.
4. Penyimpanan di sistem informasi cukup membantu dalam pembuatan laporan pembelian dan penjualan perbulan.
5. Sistem informasi ini juga memudahkan pegawai bengkel untuk pendataan data barang di bengkel Kurnia Motor.