BAB 4
IMPLEMENTASI DAN PENGUJIAN SISTEM

Dalam bab ini akan dilakukan implementasi dan pengujian terhadap sistem yang telah dibangun. Tahapan ini dilakukan setelah analisis dan perancangan selesai dilakukan, kemudian diimplementasikan pada bahasa pemrograman yang digunakan. Setelah implementasi, dilakukan pengujian sistem dimana akan dilihat kekurangan pada sistem yang telah dibangun, untuk dilakukan perbaikan maupun pengembangan terhadap sistem tersebut.

4.1 Implementasi

Setelah sistem dianalisis dan dirancang, maka tahap selanjutnya adalah implementasi sistem. Tujuan dari tahap implementasi sistem adalah untuk mengkonfirmasikan modul program perancangan pada para pelaku sistem sehingga *user* dapat memberi masukan kepada pembangun sistem.

4.1.1 Implementasi Perangkat Keras

Perangkat keras yang digunakan untuk mengimplementasikan program ini adalah sebagai berikut.

Processor : 2.2 GHz

RAM : 1 GB

HardDisk : 80 GB

Monitor : 15 ‘’

Mouse dan Keyboard

4.1.2 Implementasi Perangkat Lunak

Perangkat lunak yang digunakan untuk mengimplementasikan program ini adalah sebagai berikut.

Sistem Operasi : Windows 7

Tools Program : Visual Basic 6.0 sebagai bahasa pemograman

: *Component Visual Basic 6.0*

: *MySQL Connector ODBC* sebagai konektor yang akan menghubungkan database ke aplikasi pengguna.

: MySQL Server sebagai database.

4.1.3 Implementasi Form

Implementasi form dilakukan untuk mengetahui setiap halaman program yang dibuat. Berikut ini adalah implementasi *form* yang dibuat.

Tabel 4.1 Implementasi Form

Menu	Deskripsi	Nama File
Login User	File program untuk proses user login.	Form1.frm
Menu Utama Admin (Administrasi)	File program untuk menampilkan menu utama Admin (Administrasi).	FrmMenuUtamaAdmin.frm
Pengolahan Data User	File program untuk proses pengolahan data user.	frmUser.frm
Ganti Password	File program untuk mengubah password.	frmgantipass.frm
Menu Utama Manager	File program untuk menampilkan menu utama Manager.	FrmMenuManager.frm
Lihat Data Peramalan	File program untuk menampilkan data peramalan.	frmdataperamalan.frm
Lihat Data Penjualan	File program untuk menampilkan data penjualan.	frmdatapenjualan.frm
Proses Peramalan	File program untuk melakukan proses peramalan jenis barang mulai dari menentukan jenis barang, memilih data, meramalkan berdasarkan jenis barang	Frmdetailperamalan.frm Frmdetailjenis.frm

	serta penyimpanan hasil peramalan dan cetak hasil peramalan.	
Pengolahan data kategori barang	File program untuk proses pengolahan data kategori barang.	frmkategori.frm
Pengolahan data jenis barang	File program untuk proses pengolahan data jenis barang.	frmjenis.frm
Pengolahan data penjualan barang	File program untuk proses pengolahan data penjualan barang.	FormDataPenjualan2.frm
Data Modul	File program untuk mengumpulkan komponen – komponen aplikasi	Connection.bas

4.1.4 Implementasi Pengguna

Pengguna atau user dimaksudkan untuk mengetahui siapa saja pengguna yang terlibat dalam sistem peramalan penjualan di distro OR-K 689, beserta karakteristiknya. Sistem yang sedang berjalan melibatkan dua orang pengguna, yaitu: Admin (Administrasi penjualan), Manager Distro OR-K 689.

4.1.5 Implementasi Basis Data

Implementasi basis data dibangun berdasarkan perancangan basis data yang dibuat sebelumnya. Basis data yang dibutuhkan akan diimplementasikan menggunakan perangkat lunak *MySQL*. Berikut ini akan dijelaskan bagaimana proses pembuatan basis data beserta tabel-tabel yang mendukung sistem.

1. Pembuatan Database

```
CREATE DATABASE `sales_predict1`;
```

2. Pembuatan table *user*

```
CREATE TABLE `user` (
        `UserID` varchar(20) NOT NULL,
        `password` varchar(32) NOT NULL,
```

```
        `level_user` tinyint(4) NOT NULL,
        PRIMARY KEY (`UserID`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Pembuatan tabel *log_user*

```
CREATE TABLE `log_user` (
        'waktu` DATETIME NOT NULL,
        `UserID` VARCHAR(20) NOT NULL,
        `actions` VARCHAR(50) NOT NULL,
        `object` VARCHAR(50) NOT NULL,
         PRIMARY KEY (`waktu`),
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `log_user`
  ADD CONSTRAINT `log_user_ibfk_1` FOREIGN KEY (`userID`)
REFERENCES `user` (`userID`);
```

4. Pembuatan tabel *kategori_barang*

```
CREATE TABLE `kategori_barang` (
        `id_kategori` varchar(10) NOT NULL,
        `nama_kategori` varchar(50) NOT NULL,
        PRIMARY KEY (`id_kategori`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5. Pembuatan tabel *jenis_barang*

```
CREATE TABLE `jenis_barang` (
        `id_jenis` varchar(10) NOT NULL,
        `id_kategori` varchar(10) NOT NULL,
        `nama_jenis` varchar(50) NOT NULL,
        PRIMARY KEY (`id_jenis`),
)ENGINE=InnoDB DEFAULT CHARSET=latin1;
ALTER TABLE `jenis_barang`
  ADD   CONSTRAINT   `jenis_barang_ibfk_1`   FOREIGN   KEY
(`id_kategori`)       REFERENCES       `kategori_barang`
(`id_kategori`);
```

6. Pembuatan tabel *penjualan*

```
CREATE TABLE `datapenjualan` (
        `id_penjualan` int(2) NOT NULL AUTO_INCREMENT,
        `UserID' varchar(20),
        `tahun` int(11) NOT NULL,
        PRIMARY KEY (`id_penjualan`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `penjualan`
```

```
  ADD    CONSTRAINT    `penjualan_ibfk_1`    FOREIGN    KEY
(`userID`) REFERENCES `user` (`userID`);
```

7. Pembuatan tabel *detaildatapenjualan*

```
CREATE TABLE `datapenjualan` (
      `id_detailpenjualan`      int(11)      NOT      NULL
      AUTO_INCREMENT,
        `id_penjualan` int(2) NOT NULL,
        `tgl_penjualan` datetime NOT NULL,
        `id_jenis` varchar(10) NOT NULL,
        `jml_penjualan` double NOT NULL,
  PRIMARY KEY (`id_detailpenjualan`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1;
ALTER TABLE `detaildatapenjualan`
  ADD CONSTRAINT `detaildatapenjualan_ibfk_3` FOREIGN KEY
(`id_penjualan`) REFERENCES `penjualan` (`id_penjualan`),
  ADD CONSTRAINT `detaildatapenjualan_ibfk_2` FOREIGN KEY
(`id_jenis`) REFERENCES `jenis_barang` (`id_jenis`);
```

8. Pembuatan tabel *peramalan*

```
CREATE TABLE `peramalan` (
       `id_peramalan` int(4) NOT NULL,
       `id_jenis` varchar(10) NOT NULL,
       `tgl_peramalan` datetime NOT NULL,
       `tahun` int(4) NOT NULL,
       `bulan_ramalan` varchar(15) NOT NULL,
       `hasil_ramalan` int(5) NOT NULL,
  PRIMARY KEY (`id_peramalan`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 ;
ALTER TABLE `peramalan`
  ADD    CONSTRAINT    `peramalan_ibfk_1`    FOREIGN    KEY
(`id_jenis`) REFERENCES `jenis_barang` (`id_jenis`);
```

4.1.6 Implementasi Antarmuka

Pada bagian implementasi antarmuka ini akan diimplementasikan antarmuka program yang telah dibangun (tampilan antarmuka program terdapat di lampiran A).

4.2 Pengujian

Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan mempresentasikan kajian pokok dari spesifikasi, desain, dan pengkodean. Pengujian dilakukan dengan cara alpha menggunakan metode black-

box dan pengujian beta dengan wawancara. Pengujian alpha menggunakan metode black-box berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian alpha dengan menggunakan metode black-box berfokus pada persyaratan fungsional perangkat lunak. Pengujian ini memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya semua persyaratan fungsional untuk suatu program.

4.2.1 Rencana Pengujian Sistem

Rencana pengujian yang akan dilakukan dengan menguji sistem secara alpha menggunakan metode black-box dan beta dengan wawancara. Pengujian sistem ini menggunakan data uji berdasarkan data yang terdapat di Distro OR-K 689. Rencana pengujian selengkapnya terlihat pada tabel 4.2 di bawah ini.

Tabel 4.2 Rencana Pengujian

Item Pengujian	Detail Pengujian	Jenis Pengujian
Login User	Verifikasi User ID, Password dan level user.	*Black Box*
Pengolahan Data user	Tambah data user	*Black Box*
	Ubah data user	*Black Box*
	Hapus data user	*Black Box*
	Cari data user	*Black Box*
Log User	Tampilkan log user	*Black Box*
Ganti Password	Ganti password user	*Black Box*
Pengolahan Data Kategori Barang	Tambah data kategori barang	*Black Box*
	Ubah data kategori barang	*Black Box*
	Hapus data kategori barang	*Black Box*
	Cari data kategori barang	*Black Box*
Pengolahan Data Jenis arang	Tambah data jenis barang	*Black Box*
	Ubah data jenis barang	*Black Box*
	Hapus data jenis barang	*Black Box*

	Cari data jenis barang	*Black Box*
Pengolahan Data Penjualan	Tambah data penjualan	*Black Box*
	Ubah data penjualan	*Black Box*
	Hapus data penjualan	*Black Box*
	Cari data penjualan kendaraan	*Black Box*
Proses Peramalan	View data penjualan	*Black Box*
	Peramalan penjualan barang	*Black Box*
	Simpan hasil peramalan	*Black Box*

4.2.2 Kasus dan Hasil Pengujian Alpha

Pengujian alpha dilakukan untuk mengetahui apakah data yang dimasukkan (input) sudah sesuai dengan yang diharapkan (Output). Berikut penjelasan dari pengujian alpha.

1. Pengujian Login

Tabel 4.3 Pengujian verifikasi User ID dan Password.

Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	*Username* ,*Password* dan level user benar. Contoh : *Username :* mgr1 *Password :* mgr1 *Level User :* 2
	Kegiatan	Klik tombol *Login*.
	Hasil	*Username*,*Password* dan *level user* yang dimasukan *valid*, masuk ke menu utama *user*.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	*Username Password* dan *Level user* dikosongkan. Contoh : *Username :* (dikosongkan) *Password :* (dikosongkan) *Level user :* (dikosongkan)
	Kegiatan	Klik tombol *Login*
	Hasil	Muncul pesan bahwa *login* gagal.
	Kesimpulan	Diterima
2.	Data Masukan	Salah satu dari *Username*, *Password* atau *Level user* dikosongkan. Contoh : *Username :* mgr1

		Password : (dikosongkan) *Level user :* (dikosongkan)
	Kegiatan	Klik tombol *Login*
	Hasil	Muncul pesan bahwa *login* gagal.
	Kesimpulan	Diterima
3.	Data Masukan	*Username*, *Password* atau *level user* salah. Contoh : *Username :* xxx *Password :* xxx *Level User :* xxx
	Kegiatan	Klik tombol *Login*
	Hasil	Muncul pesan bahwa *login* gagal.
	Kesimpulan	Diterima

2. Pengujian Pengolahan Data User

Tabel 4.4 Pengujian Pengolahan Data User

Tambah Data User		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data user yang tersedia dengan data yang benar. Contoh : *ID User* : mgr1 *Password* : mgr1 *Level User* : Pilih 2 “Manager”
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan “simpan berhasil”.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data user yang tersedia dengan data yang salah satu dari atribut kosong. Contoh : *ID User* : manager *Password* : (dikosongkan) *Level User* : Pilih 2 “Manager”
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan “id harus diisi” dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Ubah Data User		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengubah atribut data user yang sudah ada didalam *database* dengan data yang benar. Contoh : *User ID* : mgr1 *Password* : mgr1 *Level User* : Pilih 1 “Admin”
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan “simpan berhasil.

	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengubah atribut data user yang sudah ada didalam *database* dengan data yang salah atau kosong. Contoh : *ID User* : mgr1 *Password* : mgr1 *Level User* : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan "jabatan harus diisi" dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data User		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus. Contoh : Pilih *ID User* : 'mgr1'
	Kegiatan	Klik tombol Hapus, akan muncul pesan konfirmasi penghapusan dan jika memilih tombol Yes, maka data akan terhapus
	Hasil	Data berhasil terhapus di *database.*
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Data yang dihapus tidak ada.
	Kegiatan	Klik tombol Hapus, akan muncul pesan konfirmasi penghapusan dan jika memilih tombol Yes, muncul pesan bahwa data id user belum dipilih.
	Hasil	Muncul pesan bahwa data kosong.
	Kesimpulan	Diterima
Cari Data User		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memasukan data yang akan dicari. Contoh : Pilih '*ID User*' isi : mgr1
	Kegiatan	Klik tombol Cari.
	Hasil	muncul data yang dicari 'jika data yang dicari ditemukan' dan pesan data tidak ditemukan 'jika data yang dicari tidak ditemukan'.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Tidak mengisi atribut pencarian. Contoh : Pencarian : (dikosongkan) isi : mgr1
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan "id yang dicari belum diisi".
	Kesimpulan	Diterima

3. Pengujian Log User

Tabel 4.5 Pengujian Log User

Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih ID Usetr Contoh : Pilih *ID User* : ‘mgr1’
	Kegiatan	Klik tombol Cari.
	Hasil	Menampilkan data yang dicari dan muncul pesan jumlah data yang ditampilkan
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Tidak memilih *ID User* Contoh : Pilih *ID User* : (dikosongkan)
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan atribut pencarian tidak boleh kosong.
	Kesimpulan	Diterima

4. Pengujian Ganti Password

Tabel 4.6 Pengujian Ganti Password

Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut ganti *password* yang tersedia dengan data yang benar. Contoh : *Password* Lama : mgr1 *Password* Baru : mgrbaru Ulangi *Password* Baru : mgrbaru
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan Ganti password berhasil dan data berhasil ter *update* di *database*.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut ganti *password* yang tersedia dengan data yang salah. Contoh : *Password* Lama : xxx *Password* Baru : mgrbaru Ulangi *Password* Baru : mgrbaru
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan kesalahan dan data tidak berhasil ter *update* di *database*.
	Kesimpulan	Diterima
2.	Data Masukan	Mengisi atribut ganti *password* yang tersedia dengan data yang dikosongkan. Contoh : *Password* Lama : mgr1 *Password* Baru : mgrbaru Ulangi *Password* Baru : (dikosongkan)

	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan kesalahan dan data tidak berhasil ter *update* di *database*.
	Kesimpulan	Diterima

5. Pengujian Pengolahan Data Kategori Barang

Tabel 4.7 Pengujian Pengolahan Data Kategori Barang

Tambah Data Kategori Barang		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data kategori barang yang tersedia dengan data yang benar. Contoh : Id Kategori : ORK01 Nama Kategori : T-Shirt
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan "Simpan berhasil".
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data kategori barang yang tersedia dengan data yang kosong. Contoh : Id Kategori : (dikosongkan) Nama Kategori : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan "Id Kategori tidak boleh kosong" dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Ubah Data Kategori Barang		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengubah atribut data kategori barang yang sudah ada didalam *database* dengan data yang benar. Contoh : Id Kategori : ORK01 Nama Kategori : Oblong
	Kegiatan	klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan "Ubah data berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengubah atribut data kategori barang yang sudah ada didalam *database* dengan data yang kosong. Contoh : Id Kategori : (dikosongkan) Nama Jenis Kendaran : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut id kategori tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Kategori Barang		

Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus.
	Kegiatan	Klik tombol Hapus, akan muncul pesan konfirmasi dan jika memilih tombol Yes, maka data akan terhapus
	Hasil	Data berhasil terhapus di *database*.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Data yang dihapus tidak ada
	Kegiatan	Klik tombol Hapus.
	Hasil	Muncul pesan bahwa hapus data gagal.
	Kesimpulan	Diterima
Cari Data Kategori Barang		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memasukan data yang akan dicari. Contoh : Pencarian : 'Id Kategori' Isi : ORK01
	Kegiatan	Klik tombol Cari.
	Hasil	muncul data yang dicari 'jika data yang dicari ditemukan' atau muncul pesan 'data tidak ditemukan' jika data yang dicari tidak ditemukan.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Memasukan data yang akan dicari dengan salah satu atribut kosong. Contoh : Pencarian : "Id Kategori" Isi : (dikosongkan)
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan "data yang dicari belum diisi".
	Kesimpulan	Diterima

6. Pengujian Pengolahan Data Jenis Barang

Tabel 4.8 Pengujian Pengolahan Data Jenis Barang

Tambah Data Jenis Barang		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data jenis barang yang tersedia dengan data yang benar. Contoh : Pilih Id kategori : 'ORK01' Id Jenis : 'TS01' Nama Jenis : T-Shirt Dewasa Pj
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan "simpan berhasil".
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data jenis barang yang tersedia dengan data yang kosong.

		Contoh : Pilih Id kategori : ‘ORK01’ Id Jenis : (dikosongkan) Nama Jenis : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan Id jenis tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Ubah Data Jenis Barang		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengubah atribut data jenis barang yang sudah ada didalam *database* dengan data yang benar. Contoh : Pilih Id kategori : ‘ORK01’ Id Jenis : TS03 Nama Jenis : Oblong anak
	Kegiatan	klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan “ubah data berhasil”.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengubah atribut data jenis barang yang sudah ada didalam *database* dengan data yang kosong. Contoh : Pilih Id kategori : ‘ORK01’ Id Jenis : TS03 Nama Jenis : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan “nama jenis harus diisi” dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Jenis Barang		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus.
	Kegiatan	Klik tombol Hapus, akan muncul pesan konfirmasi dan jika memilih tombol Yes, maka data akan terhapus
	Hasil	Data berhasil terhapus di *database.*
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Data yang dihapus tidak ada
	Kegiatan	Klik tombol Hapus.
	Hasil	Muncul pesan bahwa “hapus data gagal”.
	Kesimpulan	Diterima
Cari Data Jenis Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memasukan data yang akan dicari. Contoh : Pencarian : ‘Id Jenis’ Isi : TS01
	Kegiatan	Klik tombol Cari.
	Hasil	munculdata yang ya ng dicari ‘ jika data ditemukan’ atau muncul pesan data tidak ditemukan jika data

		yang dicari tidak ditemukan.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Memasukan data yang akan dicari dengan salah satu atribut kosong. Contoh : Pencarian : 'Id Jenis' Isi : (dikosongkan)
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan "data yang dicari belum diisi".
	Kesimpulan	Diterima

7. Pengujian Pengolahan Data Penjualan Barang

Tabel 4.9 Pengujian Pengolahan Data Penjualan Barang

Tambah Data Penjualan Barang		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data Penjualan barang yang tersedia dengan data yang benar. Contoh : ID Penjualan Id Kategori : pilih 'ORK01' Id Jenis : pilih 'TS01' Tahun : pilih '2010' Bulan : pilih 'Januari' Jumalah Penjualan : 120 pc
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan "simpan berhasil".
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data Penjualan kendaraan yang tersedia dengan data yang kosong. Contoh : ID Penjualan Id Kategori : (dikosongkan) Id Jenis : (dikosongkan) Tahun : (dikosongkan) Bulan : (dikosongkan) Jumalah Penjualan : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan "kategori harus diisi" dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Ubah Data Penjualan Barang		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengubah atribut data penjualan barang yang sudah ada didalam *database* dengan data yang benar. Contoh : ID Penjualan

		Id Kategori : pilih ‘ORK01’ Id Jenis : pilih ‘TS01’ Tahun : pilih ‘2010’ Bulan : pilih ‘Januari’ Jumalah Penjualan : xxx
	Kegiatan	klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan “simpan berhasil”.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengubah atribut data penjualan barang yang sudah ada didalam *database* dengan data yang kosong. Contoh : ID Penjualan Id Kategori : (dikosongkan) Id Jenis : (dikosongkan) Tahun : (dikosongkan) Bulan : (dikosongkan) Jumalah Penjualan : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan “id kategori boleh kosong” dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Penjualan Barang		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus.
	Kegiatan	Klik tombol Hapus, akan muncul pesan konfirmasi dan jika memilih tombol Yes, maka data akan terhapus
	Hasil	Data berhasil terhapus di *database.*
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Data yang dihapus tidak ada
	Kegiatan	Klik tombol Hapus.
	Hasil	Muncul pesan konfirmasi dan jika memilih tombol Yes, maka akan muncuk pesan bahwa data kosong.
	Kesimpulan	Diterima
***Upload* Data Penjualan Barang**		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan diupload Contoh: Pilih : Data 2010.xls
	Kegiatan	Klik tombol open
	Hasil	Data berhasil di *upload*
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Memilih data yang akan di*upload* Contoh: Pilih : Data 2010.docx
	Kegiatan	Klik tombol *open*
	Hasil	Muncul data yang di*upload* ‘jika data yang di*upload* berhasil’ tidak muncul data ‘jika data yang di*upload* gagal’
	Kesimpulan	Diterima

Cari Data Penjualan Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memasukan data yang akan dicari. Contoh : Pencarian : ‘Id Kategori’ Isi : ORK01
	Kegiatan	Klik tombol Cari.
	Hasil	muncul data yang dicari ‘jika data yang dicari ditemukan’ atau muncul pesan ‘’data tidak ditemukan ‘jika data yang dicari tidak ditemukan’.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Memasukan data yang akan dicari dengan salah satu atribut kosong. Contoh : Pencarian : ‘Id Kategori’ Isi : (dikosongkan)
	Kegiatan	Klik tombol Cari.
	Hasil	muncul atribut pencarian tidak boleh kosong.
	Kesimpulan	Diterima

8. Pengujian Proses Peramalan

Tabel 4.10 Pengujian Proses Peramalan

Peramalan Penjualan Barang		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Masukan Periode Peramalan. Contoh : Pilih Periode : ’3 Tahun’
	Kegiatan	Memilih jenis barang, memilih data penjualan yang akan diramalkan, lalu klik Ramalkan.
	Hasil	Menampilkan hasil peramalan berdasarkan jenis barang
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Data Penjualan tidak ada
	Kegiatan	Muncul Pesan data tidak sesuai dengan periode dan tidak dapat melanjutkan proses peramalan
	Hasil	Tidak dapat melakukan proses peramalan.
	Kesimpulan	Diterima
Simpan Hasil Peramalan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Data hasil proses peramalan
	Kegiatan	Klik tombol Simpan, akan muncul pesan bahwa data telah tersimpan dan akan menanyakan cetak hasil peramalan jika memilih tombol Yes, maka data akan dicetak

	Hasil	Menyimpan hasil peramalan di *database* dan akan menampilkan preview laporan hasil peramalan yang akan dicetak..
	Kesimpulan	Diterima

4.2.3 Kesimpulan Hasil Pengujian Alpha

Berdasarkan hasil pengujian dengan kasus uji sample diatas dapat ditarik kesimpulan bahwa perangkat lunak telah bebas dari kesalahan sintaks dan secara fungsional menghasilkan hasil yang sesuai dengan yang diharapkan.

4.2.4 Kasus dan Hasil Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan wawancara langsung kepada user yang bersifat pertanyaan mengenai kepuasan user.

A. Wawancara pada bagian *Admin (Administrasi)*
    1. Apakah aplikasi yang dibangun mudah dalam pengelolaan sistem ?
    2. Apakah aplikasi dapat membantu dalam pengolahan data master ?
    3. Apakah aplikasi yang dibangun mudah dalam pemeliharaan sistem ?
    4. Apakah aplikasi yang dibangun sudah sesuai dengan kebutuhan?
    5. Apakah aplikasi yang dibangun mudah digunakan ?
    6. Apakah halaman aplikasi sudah terstruktur dengan baik ?

Dari hasil wawancara dengan bagian admin (administrasi) dapat disimpulkan bahwa sistem ini dapat membantu admin (administrasi) dalam melakukan proses pengolahan data master dan aplikasi sangat mudah dimengerti dalam penggunaanya, halaman aplikasi juga sudah terstruktur dengan baik.

B. Wawancara pada bagian *Manager*

1. Apakah aplikasi dapat membantu meramalkan penjualan barang ?
2. Apakah aplikasi dapat membantu memberikan informasi data penjualan ?
3. Apakah aplikasi yang dibangun sudah sesuai dengan kebutuhan ?
4. Apakah aplikasi yang dibangun mudah digunakan ?
5. Apakah halaman aplikasi sudah terstruktur dengan baik ?

Dari hasil wawancara dengan bagian manager dapat disimpulkan bahwa sistem yang telah dibangun sangat membantu perusahaan dalam proses meramalkan penjualan jenis barang, dapat memberikan informasi data penjualan. Sistem ini juga sudah sesuai dengan kebutuhan, mudah dalam penggunaannya dan halaman aplikasi juga sudah terstruktur dengan baik.

4.2.5 Kesimpulan Hasil Pengujian Beta

Dari hasil pengujian beta yang telah dilakukan yaitu dengan wawancara kepada user yang bersifat pertanyaan di lapangan, diperoleh kesimpulan bahwa sistem yang telah dibangun ini dinilai sudah optimal dalam hal pengelolaan sistem, mudah digunakan dan dapat membantu dalam proses pengolahan data master, data penjualan dan juga bermanfaat untuk dapat diolah kembali sebagai data historis pada proses peramalan penjualan barang, informasi hasil peramalan yang dihasilkan mudah dimengerti dan sesuai dengan yang diharapkan, memenuhi kebutuhan, mudah digunakan dan sangat bermanfaat.

4.2.6 Pengujian Peramalan

Pengujian proses peramalan dilakukan dengan membandingkan hasil perhitungan dari peramalan yang menggunakan metode peramalan Trend Variasi Musiman, Least Quared dan Dekomposisi.

Untuk mengetahui metode mana yang memiliki nilai *error* yang paling kecil antara metode peramalan Trend Variasi Musiman, Least Quared dan Dekomposisi

Pada tabel-tabel dibawah ini akan ditunjukkan hasil perhitungan kesalahan / *error* menggunakan metode MAD dan MSE dari setiap metode yang dibandingkan.

Tabel 4.11 Hasil Perhitungan Kesalahan/*Error*

Metode Peramalan	Metode Error	
	MAD	MSE
Trend Variasi Musiman	72.58	7380.91
Least Squared	92.65	11708.72
Dekomposisi	26.49	1328.42

Dari hasil perhitungan kesalahan / *error* menggunakan metode MAD dan MSE dari setiap metode yang dibandingkan pada tabel – tabel diatas, terlihat bahwa metode Dekomposisi memiliki nilai kesalahan/error yang lebih kecil dibandingkan dengan metode yang lainnya.

Dengan ini dapat disimpulkan dari hasil perhitungan kesalahan / error menggunakan metode MAD dan MSE metode Dekomposisi lah yang lebih baik digunakan pada kasus ini.