BAB 4
IMPLEMENTASI DAN PENGUJIAN

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 Peramgkat Keras

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

Processor	: Intel Pentium IV , 1,8 Ghz
RAM	: 256 MB
Harddisk	: 80 GB
Perangkat keras lainnya	: Monitor CRT 17 Inch, *Keyboard, Mouse, Printer*

4.1.2 Implementasi Perangkat Lunak

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

Sistem Operasi	: *Windows XP Service Pack 2*.
Tools program	:*Borland Delphi 7.0*, sebagai implementasi rancangan sistem.
	: *Component Borland Delphi 7.0* .

:*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.	ULogin.pas
Menu Utama Admin	File program untuk menampilkan menu utama Admin.	UMenuAdmin.pas
Pengolahan Data User	File program untuk proses pengolahan data user.	UDataUser.pas
Log User	File program untuk menampilkan Log User.	ULogUser.pas
Ganti Password Admin	File program untuk mengubah password admin.	UGantiPasswordAdmin.pas
Menu Utama Manager	File program untuk menampilkan menu utama Manager.	UMenuManager.pas
Lihat Data Peramalan	File program untuk menampilkan data peramalan.	ULihatDataPeramalan.pas, URPeramalan.pas

Lihat Data Penjualan	File program untuk menampilkan data penjualan.	ULihatDataPenjualan.pas, URPenjualan.pas
Proses Peramalan	File program untuk melakukan proses peramalan kendaraan mulai dari menentukan jenis kendaraan, tipe kendaraan, meramalkan berdasarkan tipe kendaraan serta penyimpanan hasil peramalan dan cetak hasil peramalan.	UProsesPeramalan.pas, UDetailPeramalanKendaraan.pas, UDetailPeramalanJenisKendaraan.pas, UDetailPeramalanTipeKendaraan.pas, UPerviewPeramalan.pas
Pengaturan Periode Peramalan	File program untuk proses pengolahan periode.	UPengaturanPeriode.pas
Pengaturan Alpha Peramalan	File program untuk proses pengolahan alpha.	UPengaturanAlpha.pas
Ganti Password Manager	File program untuk mengubah password manager.	UGantiPasswordManager.pas
Menu Utama Administrasi Penjualan	File program untuk menampilkan menu utama Administrasi Penjualan.	UMenuAdministration.pas
Pengolahan data pelanggan	File program untuk proses pengolahan data pelanggan.	UDataPelanggan.pas

Pengolahan data jenis kendaraan	File program untuk proses pengolahan data jenis kendaraan.	UJenisKendaraan.pas
Pengolahan data tipe kendaraan	File program untuk proses pengolahan data tipe kendaraan.	UTipeKendaraan.pas
Pengolahan data warna kendaraan	File program untuk proses pengolahan data warna kendaraan.	UWarna.pas
Pengolahan data penjualan kendaraan	File program untuk proses pengolahan data penjualan kendaraan.	UDataPenjualan.pas, UCariDataPelanggan.pas, UTambahDataPelanggan.pas
Pengolahan data pemesanan kendaraan	File program untuk proses pengolahan data pemesanan kendaraan.	UDataPemesanan.pas, UCariDataPelanggan.pas, UTambahDataPelanggan.pas
Pembuatan laporan data penjualan kendaraan	File program untuk proses pembuatan laporan data penjualan kendaraan.	UPembuatanLaporan.pas, UPerviewLaporan.pas
Ganti Password Administrasi penjualan	File program untuk mengubah password Administrasi penjualan.	UGantiPassword Administration.pas
Data Modul	File program untuk mengumpulkan komponen – komponen aplikasi	UDM.pas

4.1.4 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 `sistem_peramalan`;
```

2. Pembuatan tabel *user*

```
CREATE TABLE `user` (
‘user_id` VARCHAR(23) NOT NULL,
`user_tipe`   ENUM(   'admin',   'manager',   'administrasi
penjualan' ) NOT NULL,
`password` VARCHAR(30) NOT NULL,
PRIMARY KEY (`user_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Pembuatan tabel *log_user*

```
CREATE TABLE `log_user` (
‘waktu` DATETIME NOT NULL,
`user_id` VARCHAR(23) NOT NULL,
`actions` VARCHAR(50) NOT NULL,
`object` VARCHAR(50) NOT NULL,
PRIMARY KEY (`waktu`),
FOREIGN KEY ( `user_id` ) REFERENCES `user`(`user_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4. Pembuatan tabel *pelanggan*

```
CREATE TABLE `pelanggan` (
`id_pelanggan` VARCHAR(8) NOT NULL,
`nama` VARCHAR(50) NOT NULL,
`alamat` VARCHAR(100) NOT NULL,
PRIMARY KEY (`id_pelanggan`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5. Pembuatan tabel *jenis_kendaraan*

```
CREATE TABLE `jenis_kendaraan` (
`kd_jenis` VARCHAR(5) NOT NULL,
`nama_jenis` VARCHAR(50) NOT NULL
PRIMARY KEY (`kd_jenis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6. Pembuatan tabel *tipe_kendaraan*

```
CREATE TABLE `tipe_kendaraan` (
`kd_tipe` VARCHAR(10) NOT NULL,
`nama_tipe` VARCHAR(50) NOT NULL,
`kd_jenis` VARCHAR(5) NOT NULL,
PRIMARY KEY (`kd_tipe`),
FOREIGN KEY ( `kd_jenis` ) REFERENCES
`jenis_kendaraan`(`kd_jenis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

7. Pembuatan tabel *warna*

```
CREATE TABLE `warna` (
`kd_warna` VARCHAR(9) NOT NULL,
`nama_warna` VARCHAR(50) NOT NULL,
`kd_jenis` VARCHAR(5) NOT NULL,
PRIMARY KEY (`kd_warna`),
FOREIGN KEY ( `kd_jenis` ) REFERENCES
`jenis_kendaraan`(`kd_jenis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

8. Pembuatan tabel *penjualan*

```
CREATE TABLE `penjualan` (
`no_penjualan` VARCHAR(9) NOT NULL,
`tgl_jual` DATE NOT NULL,
`hrg_jual` INT(12) NOT NULL,
`no_lot` VARCHAR(9) NOT NULL,
`no_rangka` VARCHAR(18) NOT NULL,
`no_mesin` VARCHAR(13) NOT NULL,
`id_pelanggan` VARCHAR(8) NOT NULL,
`kd_jenis` VARCHAR(5) NOT NULL,
`kd_tipe` VARCHAR(10) NOT NULL,
`kd_warna` VARCHAR(9) NOT NULL,
`user_id` VARCHAR(23) NOT NULL,
`ket` VARCHAR(100) NOT NULL,
PRIMARY KEY (`no_penjualan`),
FOREIGN KEY ( `id_pelanggan` ) REFERENCES
`pelanggan`(` id_pelanggan`),
FOREIGN KEY ( `kd_jenis` ) REFERENCES
`jenis_kendaraan`(`kd_jenis`),
FOREIGN KEY ( `kd_tipe` ) REFERENCES
`tipe_kendaraan`(`kd_tipe`),
FOREIGN KEY ( `kd_warna` ) REFERENCES
`warna`(`kd_warna`),
FOREIGN KEY ( `user_id` ) REFERENCES
`user`(`user_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

9. Pembuatan tabel *periode*

```
CREATE TABLE `periode` (
`periode` INT(2) NOT NULL,
`ket` VARCHAR(10) NOT NULL,
PRIMARY KEY (`periode`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

10. Pembuatan tabel *alpha*

```
CREATE TABLE `alpha` (
`alpha` DOUBLE NOT NULL,
`ket` VARCHAR(10) NOT NULL,
PRIMARY KEY (`alpha`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

11. Pembuatan tabel *peramalan*

```
CREATE TABLE `peramalan` (
`no_peramalan` VARCHAR(19) NOT NULL,
`periode` INT(2) NOT NULL,
`alpha` DOUBLE NOT NULL,
`user_id` VARCHAR(23) NOT NULL,
`kd_tipe` VARCHAR(10) NOT NULL,
`tgl_peramalan` DATE NOT NULL,
`bln_ramalan` VARCHAR(15) NOT NULL,
`hasil_ramalan` INT(5) NOT NULL,
PRIMARY KEY (`no_peramalan`),
FOREIGN KEY ( `periode` ) REFERENCES
`periode`(`periode`),
FOREIGN KEY ( `alpha` ) REFERENCES
`alpha`(`alpha`),
FOREIGN KEY ( `kd_tipe` ) REFERENCES
`tipe_kendaraan`(`kd_tipe`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

1. Pembuatan tabel *pemesanan*

```
CREATE TABLE `pemesanan` (
`no_pemesanan` VARCHAR(9) NOT NULL,
`tgl_jual` DATE NOT NULL,
`u_muka` INT(12) NOT NULL,
`id_pelanggan` VARCHAR(8) NOT NULL,
`kd_jenis` VARCHAR(5) NOT NULL,
`kd_tipe` VARCHAR(10) NOT NULL,
`user_id` VARCHAR(23) NOT NULL,
`status` VARCHAR(50) NOT NULL,
PRIMARY KEY (`no_pemesanan`),
FOREIGN KEY ( `id_pelanggan` ) REFERENCES
`pelanggan`(` id_pelanggan`),
FOREIGN KEY ( `kd_jenis` ) REFERENCES
`jenis_kendaraan`(`kd_jenis`),
FOREIGN KEY ( `kd_tipe` ) REFERENCES
`tipe_kendaraan`(`kd_tipe`),
```

```
FOREIGN KEY ( `kd_warna` ) REFERENCES
`warna`(`kd_warna`),
FOREIGN KEY ( `user_id` ) REFERENCES
`user`(`user_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4.1.5 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 merupakan metode yang dilakukan untuk menjelaskan mengenai pengoperasian perangkat lunak yang terdiri dari perangkat pengujian, metode pengujian dan pelaksanaan pengujian. Pengujian yang digunakan untuk menguji sistem ini adalah menggunakan metode pengujian *black box* dan *white box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak, serta pengujian *white box* berfokus pada motode peramalan.

4.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara *alpha* dan *beta*. Pengujian sistem ini menggunakan data uji berdasarkan data yang terdapat di PT. Wicaksana Berlian Motor. 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, User Tipe, dan Password	*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 Pelanggan	Tambah data pelanggan	*Black Box*
	Ubah data pelanggan	*Black Box*
	Hapus data pelanggan	*Black Box*
	Cari data pelanggan	*Black Box*
Pengolahan Data Jenis Kendaraan	Tambah data jenis kendaraan	*Black Box*
	Ubah data jenis kendaraan	*Black Box*
	Hapus data jenis kendaraan	*Black Box*
	Cari data jenis kendaraan	*Black Box*
Pengolahan Data Tipe Kendaraan	Tambah data tipe kendaraan	*Black Box*
	Ubah data tipe kendaraan	*Black Box*
	Hapus data tipe kendaraan	*Black Box*
	Cari data tipe kendaraan	*Black Box*
Pengolahan Data Warna Kendaraan	Tambah data warna kendaraan	*Black Box*
	Ubah data warna kendaraan	*Black Box*
	Hapus data warna kendaraan	*Black Box*
	Cari data warna kendaraan	*Black Box*
Pengolahan Data Penjualan Kendaraan	Tambah data penjualan kendaraan	*Black Box*
	Ubah data penjualan kendaraan	*Black Box*
	Hapus data penjualan kendaraan	*Black Box*
	Cari data penjualan kendaraan	*Black Box*
Pengolahan Data Pemesanan Kendaraan	Tambah data pemesanan kendaraan	*Black Box*
	Ubah data pemesanan kendaraan	*Black Box*
	Hapus data pemesanan	*Black Box*

	kendaraan	
	Cari data pemesanan kendaraan	*Black Box*
Pengaturan Periode	Tambah data periode	*Black Box*
	Hapus data periode	*Black Box*
Pengaturan Alpha	Tambah data alpha	*Black Box*
	Hapus data alpha	*Black Box*
Proses Peramalan	Peramalan persediaan kendaraan	*Black Box*
	Simpan hasil peramalan	*Black Box*
Laporan data penjualan	Cetak laporan penjualan	*Black Box*
Metode peramalan	Metode *Forecasting Single Exponential Smoothing*	*White Box*

4.2.2 Kasus dan Hasil Pengujian Alpha

4.2.2.1 Pengujian *Black Box*

1. Pengujian *Login*

Tabel 4.3 Pengujian verifikasi *User ID* dan *Password*.

Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	*User ID* dan *Password* benar. Contoh : *Username :* manager *Password :* manager
	Kegiatan	Klik tombol *Login*.
	Hasil	*User ID* dan *Password* yang dimasukan *valid*, masuk ke menu utama *user*.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	*User ID* dan *Password* dikosongkan. Contoh :

		Username : (dikosongkan) *Password :* (dikosongkan)
	Kegiatan	Klik tombol *Login*
	Hasil	Muncul pesan bahwa *login* gagal.
	Kesimpulan	Diterima
2.	Data Masukan	Salah satu dari *User ID* atau *Password* dikosongkan. Contoh : *Username :* manager *Password :* (dikosongkan)
	Kegiatan	Klik tombol *Login*
	Hasil	Muncul pesan bahwa *login* gagal.
	Kesimpulan	Diterima
3.	Data Masukan	*User ID* dan *Password* salah. Contoh : *Username :* manager *Password :* 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 : *User ID* : manager *User Tipe* : Pilih “Manager”

		Password : manager
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan 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 : *User ID* : manager *User Tipe* : Pilih "Manager" *Password* : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong 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* : manager *User Tipe* : Pilih "Manager" *Password* : passwordmanagerbaru
	Kegiatan	klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan 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 : *User ID* : manager *User Tipe* : Pilih "Manager" *Password* : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong 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 *User ID* : 'manager'
	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 kosong.
	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 : Pencarian : pilih ‘User ID’ isi : manager
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan ‘data ditemukan’ 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 salah satu atribut pencarian. Contoh : Pencarian : (dikosongkan) isi : manager
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan atribut tidak boleh kosong.
	Kesimpulan	Diterima

3. Pengujian Log User

Tabel 4.5 Pengujian Log User

Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih User ID Contoh : Pilih *User ID* : ‘manager’
	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 User ID Contoh : Pilih *User ID* : (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 : manager *Password* Baru : passwordbarumanager Ulangi *Password* Baru : passwordbarumanager
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan data berhasil ter *update* 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 : passwordbarumanager

		Ulangi *Password* Baru : passwordbarumanager
	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 : manager *Password* Baru : passwordbarumanager 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 Pelanggan

Tabel 4.7 Pengujian Pengolahan Data Pelanggan

Tambah Data Pelanggan		
1.	**Kasus dan Hasil Uji (Data Normal)**	
	Data Masukan	Mengisi atribut data pelanggan yang tersedia dengan data yang benar. Contoh : Nama Pelanggan : Budi Alamat : Jln. Merseyside No.96
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		

1.	Data Masukan	Mengisi atribut data pelanggan yang tersedia dengan data yang kosong. Contoh : Nama Pelanggan : Budi Alamat : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Ubah Data Pelanggan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengubah atribut data pelanggan yang sudah ada didalam *database* dengan data yang benar. Contoh : Nama Pelanggan : Budi Haryadi Alamat : Jln. Merseyside No.96
	Kegiatan	klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengubah atribut data pelanggan yang sudah ada didalam *database* dengan data kosong. Contoh : Nama Pelanggan : (dikosongkan) Alamat : Jln. Merseyside No.96
	Kegiatan	Klik tombol Simpan.

	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Pelanggan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus. Contoh : Pilih ID Pelanggan : ‘P01’
	Kegiatan	Klik tombol Hapus, akan muncul pesan konfirmasi dan jika memilih tombol Yes, maka data akan berhasil 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 kosong.
	Hasil	Muncul pesan bahwa data kosong.
	Kesimpulan	Diterima
Cari Data Pelanggan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memasukan data yang akan dicari. Contoh : Pencarian : ‘ID Pelanggan’ Isi : P01
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan ‘data ditemukan’ 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	Tidak mengisi salah satu atribut pencarian. Contoh : Pencarian : (dikosongkan) isi : P01
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan atribut pencarian tidak boleh kosong.
	Kesimpulan	Diterima

6. Pengujian Pengolahan Data Jenis Kendaraan

Tabel 4.8 Pengujian Pengolahan Data Jenis Kendaraan

Tambah Data Jenis Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data jenis kendaraan yang tersedia dengan data yang benar. Contoh : Nama Jenis Kendaraan : Triton
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data jenis kendaraan yang tersedia dengan data yang kosong. Contoh :

		Nama Jenis Kendaran : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Ubah Data Jenis Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengubah atribut data jenis kendaraan yang sudah ada didalam *database* dengan data yang benar. Contoh : Nama Jenis Kendaraan : Maven
	Kegiatan	klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengubah atribut data jenis kendaraan yang sudah ada didalam *database* dengan data yang kosong. Contoh : Nama Jenis Kendaran : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Jenis Kendaraan		
Kasus dan Hasil Uji (Data Normal)		

1.	Data Masukan	Memilih data yang akan dihapus. Contoh : Pilih Kode Jenis Kendaraan : ‘J0001’
	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
Cari Data Jenis Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memasukan data yang akan dicari. Contoh : Pencarian : ‘Nama Jenis’ Isi : Triton
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan ‘data ditemukan’ 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 salahsatu atribut kosong.

		Contoh : Pencarian : (dikosongkan) Isi : Triton
	Kegiatan	Klik tombol Cari.
	Hasil	muncul atribut pencarian tidak boleh kosong.
	Kesimpulan	Diterima

7. Pengujian Pengolahan Data Tipe Kendaraan

Tabel 4.9 Pengujian Pengolahan Data Tipe Kendaraan

Tambah Data Tipe Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data tipe kendaraan yang tersedia dengan data yang benar. Contoh : Pilih Nama Jenis Kendaraan : ‘Triton’ Nama Tipe Kendaraan : GLX
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data tipe kendaraan yang tersedia dengan data yang kosong. Contoh : Pilih Nama Jenis Kendaran : ‘Triton’ Nama Tipe Kendaran : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di

		database.
	Kesimpulan	Diterima
Ubah Data Tipe Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengubah atribut data tipe kendaraan yang sudah ada didalam *database* dengan data yang benar. Contoh : Pilih Nama Jenis Kendaran : 'Triton' Nama Tipe Kendaran : GLX
	Kegiatan	klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengubah atribut data tipe kendaraan yang sudah ada didalam *database* dengan data yang kosong. Contoh : Pilih Nama Jenis Kendaran : 'Triton' Nama Tipe Kendaran : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Tipe Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus. Contoh :

		Pilih Kode Tipe Kendaraan : ‘J0001-T001’
	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
Cari Data Tipe Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memasukan data yang akan dicari. Contoh : Pencarian : ‘Kode Tipe’ Isi : J0001-T001
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan ‘data ditemukan’ 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 salahsatu atribut kosong. Contoh : Pencarian : (dikosongkan)

		Isi : J0001-T001
	Kegiatan	Klik tombol Cari.
	Hasil	muncul atribut pencarian tidak boleh kosong.
	Kesimpulan	Diterima

8. Pengujian Pengolahan Data warna Kendaraan

Tabel 4.10 Pengujian Pengolahan Data Warna Kendaraan

Tambah Data Warna Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data warna kendaraan yang tersedia dengan data yang benar. Contoh : Pilih Nama Jenis Kendaraan : ‘Triton’ Nama warna Kendaraan : Hitam
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data warna kendaraan yang tersedia dengan data yang kosong. Contoh : Pilih Nama Jenis Kendaran : ‘Triton’ Nama Warna Kendaran : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima

Ubah Data Warna Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengubah atribut data warna kendaraan yang sudah ada didalam *database* dengan data yang benar. Contoh : Pilih Nama Jenis Kendaran : 'Triton' Nama Warna Kendaran : GLX
	Kegiatan	klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengubah atribut data warna kendaraan yang sudah ada didalam *database* dengan data yang kosong. Contoh : Pilih Nama Jenis Kendaran : 'Triton' Nama Warna Kendaran : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Warna Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus. Contoh : Pilih Kode Warna Kendaraan : 'J0001-W01'
	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
Cari Data Warna Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memasukan data yang akan dicari. Contoh : Pencarian : ‘Kode Warna’ Isi : J0001-W01
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan ‘data ditemukan’ 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 salahsatu atribut kosong. Contoh : Pencarian : (dikosongkan) Isi : J0001-W01
	Kegiatan	Klik tombol Cari.

	Hasil	muncul atribut pencarian tidak boleh kosong.
	Kesimpulan	Diterima

9. Pengujian Pengolahan Data Penjualan Kendaraan

Tabel 4.11 Pengujian Pengolahan Data Penjualan Kendaraan

Tambah Data Penjualan Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data Penjualan kendaraan yang tersedia dengan data yang benar. Contoh : No. lot : 01/2012 ID Pelanggan : P01 Kendaraan : pilih ‘Triton’, ‘GLX’ Warna Kendaraan : pilih ‘Hitam’ Tanggal Jual : pilih ‘1-Jan-2012’ Keteranggan : cash
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data Penjualan kendaraan yang tersedia dengan data yang kosong. Contoh : No. lot : (dikosongkan) ID Pelanggan : (dikosongkan) Kendaraan : (dikosongkan) Warna Kendaraan : (dikosongkan) Tanggal Jual : (dikosongkan)

		Keteranggan : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Ubah Data Penjualan Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengubah atribut data penjualan kendaraan yang sudah ada didalam *database* dengan data yang benar. Contoh : No. lot : 01/2012 ID Pelanggan : P01 Kendaraan : pilih ‘Triton’, ‘GLX’ Warna Kendaraan : pilih ‘Hitam’ Tanggal Jual : pilih ‘1-Jan-2012’ Keteranggan : cash
	Kegiatan	klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengubah atribut data penjualan kendaraan yang sudah ada didalam *database* dengan data yang kosong. Contoh : No. lot : (dikosongkan) ID Pelanggan : (dikosongkan) Kendaraan : (dikosongkan)

		Warna Kendaraan : (dikosongkan) Tanggal Jual : (dikosongkan) Keteranggan : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Penjualan Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus. Contoh : Pilih No. Penjualan: ‘01/2012’
	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
Cari Data Penjualan Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memasukan data yang akan dicari. Contoh : Pencarian : ‘No. Penjualan’

		Isi : 01/2012
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan ‘data ditemukan’ 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 salahsatu atribut kosong. Contoh : Pencarian : (dikosongkan) Isi : 01/2012
	Kegiatan	Klik tombol Cari.
	Hasil	muncul atribut pencarian tidak boleh kosong.
	Kesimpulan	Diterima

10. Pengujian Pengolahan Data Pemesanan

Tabel 4.12 Pengujian Pengolahan Data Pemesanan

Tambah Data Pemesanan Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data Pemesanan kendaraan yang tersedia dengan data yang benar. Contoh : ID Pelanggan : P01 Kendaraan : pilih ‘Triton’, ‘GLX’ Uang Muka : 1.000.000 Tanggal Jual : pilih ‘1-Jan-2012’ Keteranggan : pesan

	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data Pemesanan kendaraan yang tersedia dengan data yang kosong. Contoh : ID Pelanggan : (dikosongkan) Kendaraan : (dikosongkan) Uang Muka : (dikosongkan) Tanggal Jual : (dikosongkan) Keteranggan : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Ubah Data Pemesanan Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengubah atribut data pemesanan kendaraan yang sudah ada didalam *database* dengan data yang benar. Contoh : ID Pelanggan : P01 Kendaraan : pilih ‘Triton’, ‘GLX’ Uang Muka : 1.000.000 Tanggal Jual : pilih ‘1-Jan-2012’ Keteranggan : pesan
	Kegiatan	klik simpan.

	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengubah atribut data pemesanan kendaraan yang sudah ada didalam *database* dengan data yang kosong. Contoh : ID Pelanggan : (dikosongkan) Kendaraan : (dikosongkan) Uang Muka : (dikosongkan) Tanggal Jual : (dikosongkan) Keteranggan : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Pemesanan Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus. Contoh : Pilih No. Pemesanan: ‘01/2012’
	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
Cari Data Pemesanan Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memasukan data yang akan dicari. Contoh : Pencarian : ‘No. Pemesanan’ Isi : 01/2012
	Kegiatan	Klik tombol Cari.
	Hasil	muncul pesan ‘data ditemukan’ 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 salahsatu atribut kosong. Contoh : Pencarian : (dikosongkan) Isi : 01/2012
	Kegiatan	Klik tombol Cari.
	Hasil	muncul atribut pencarian tidak boleh kosong.
	Kesimpulan	Diterima

11. Pengujian Pengaturan Periode

Tabel 4.13 Pengujian Pengaturan Periode

Tambah Data Periode		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data Periode yang tersedia dengan data yang benar. Contoh : Periode : 12 Bulan
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data periode yang tersedia dengan data yang kosong. Contoh : Periode : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Periode		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus. Contoh : Pilih Periode : ’12 Bulan’
	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)		
	Data Masukan	Data yang dihapus tidak ada
	Kegiatan	Klik tombol Hapus, akan muncul pesan konfirmasi dan jika memilih tombol Yes, maka muncul pesan data kosong.
	Hasil	Muncul pesan data kosong.
	Kesimpulan	Diterima

12. Pengujian Pengaturan Alpha

Tabel 4.14 Pengujian Pengaturan Alpha

Tambah Data Alpha		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Mengisi atribut data Alpha yang tersedia dengan data yang benar. Contoh : Alpha : 0.1
	Kegiatan	Klik simpan.
	Hasil	Data berhasil tersimpan di *database* dan muncul pesan berhasil.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Mengisi atribut data alpha yang tersedia dengan data yang salah. Contoh : Alpha : 2
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan bahwa nilai alpha antara 0 dan 1 dan data tidak berhasil tersimpan di

		database.
	Kesimpulan	Diterima
2.	Data Masukan	Mengisi atribut data alpha yang tersedia dengan data yang kosong. Contoh : Alpha : (dikosongkan)
	Kegiatan	Klik tombol Simpan.
	Hasil	Muncul pesan atribut tidak boleh kosong dan data tidak berhasil tersimpan di *database*.
	Kesimpulan	Diterima
Hapus Data Alpha		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Memilih data yang akan dihapus. Contoh : Pilih Alpha : ‘0.1’
	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)		
	Data Masukan	Data yang dihapus tidak ada
	Kegiatan	Klik tombol Hapus, akan muncul pesan konfirmasi dan jika memilih tombol Yes, maka muncul pesan data kosong.
	Hasil	Muncul pesan data kosong.
	Kesimpulan	Diterima

13. Pengujian Proses Peramalan

Tabel 4.15 Pengujian Proses Peramalan

Peramalan Persediaan Kendaraan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Masukan Periode Peramalan. Contoh : Pilih Periode : ’12 Bulan'
	Kegiatan	Memilih jenis kendaraan, memilih tipe kendaraan untuk diramalkan, lalu klik Ramalkan.
	Hasil	Menampilkan hasil peramalan berdasarkan tipe kendaraan
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Data Penjualan tidak ada
	Kegiatan	Muncul Pesan atribut tidak boleh kosong dan tidak dapat melanjutkan proses pemilihan jenis kendaraan dan tipe kendaraan untuk diramalkan
	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

14. Pengujian Laporan Data Penjualan

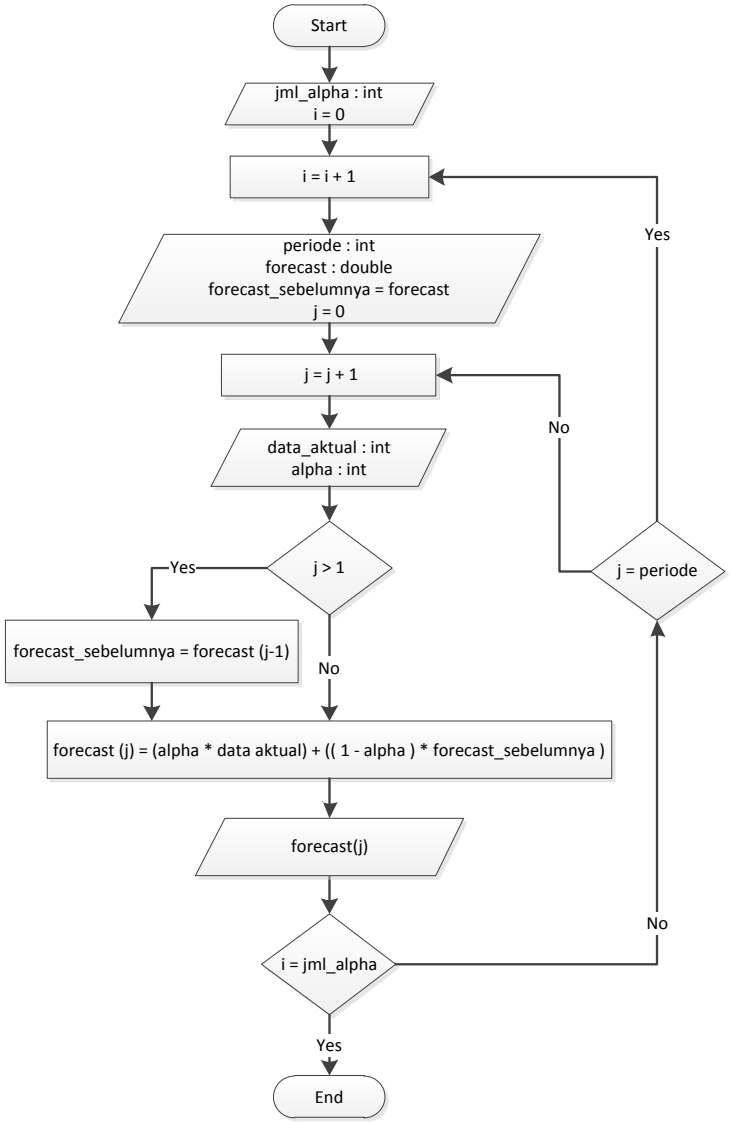
Tabel 4.16 Pengujian Laporan Data Penjualan

Cetak Laporan Penjualan		
Kasus dan Hasil Uji (Data Normal)		
1.	Data Masukan	Masukan periode waktu untuk data penjualan yang akan dicetak dengan benar. Contoh : Bulan : pilih ‘Januari’ Tahun : pilih ‘2012’
	Kegiatan	Klik Print Priview.
	Hasil	Menampilkan laporan data penjualan kendaraan berdasarkan periode waktu yang telah ditentukan.
	Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)		
1.	Data Masukan	Masukan periode waktu untuk data penjualan yang akan dicetak dengan kosong. Contoh : Bulan : (dikosongkan) Tahun : (dikosongkan)
	Kegiatan	Klik Print Priview.
	Hasil	muncul pesan kesalahan bahwa atribut tidak boleh kosong.
	Kesimpulan	Diterima

4.2.2.2 Pengujian *White Box*

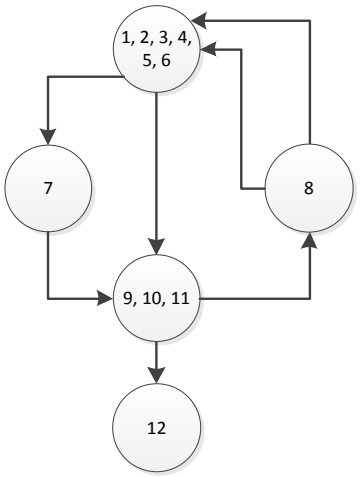
Pengujian *white box* pada perangkat lunak ini menggunakan beberapa *tools* yaitu *flowchart* pengujian dan *flowgraph* yang digunakan untuk menggambarkan alur dari metode peramalan. Berikut ini merupakan pengujian *white box* pada metode peramalan *single exponential smoothing*.

1. *Flowchart* metode *Single Exponential Smoothing*



Gambar 4.47 *Flowchart* penerapan metode *Single Exponential Smoothing*

2. *Flowgraph* metode *Single Exponential Smoothing*



Gambar 4.48 *Flowgraph* penerapan metode *Single Exponential Smoothing*

3. *Cyclomatic Complexity*

Cyclometic Complexity adalah perhitungan berdasarkan *Edge* (E) yang merupakan jumlah panah di *flowgraph* dan *Node* (N) yang merupakan jumlah lingkaran di *flowgraph*.

E = 7, N = 5

V(G) = E – N + 2

V(G) = 7 – 5 + 2

V(G) = 4

Keterangan :

E : Jumlah busur panah pada *flowgraph*

N : Jumlah lingkaran (*node*) pada *flowgraph*

V(G) : *Cyclomatic Complexity*

4. *Independent Path*

Berikut ini adalah *Independent Path* dari *flowgraph* pada gambar 4.48 :

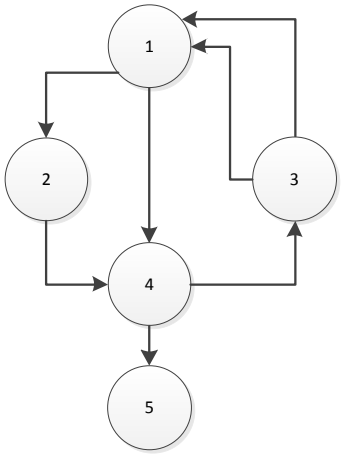
Path 1 : 1, 2, 3, 4, 5, 6, 9, 10, 11, 12

Path 2 : 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

Path 3 : 1, 2, 3, 4, 5, 6, 9, 10, 11, 8, 1, 2, 3, 4, 5, 6, 9, 10, 11, 12

Path 4 : 1, 2, 3, 4, 5, 6, 9, 10, 11, 8, 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12

5. *Graph Matrics Testing*



Gambar 4.49 *Flowgraph* penerapan metode *Single Exponential Smoothing* setelah penomoran ulang.

Tabel 4.17 *Graph Matrics Testing* penerapan metode *Single Exponential Smoothing*

Node	**1**	**2**	**3**	**4**	**5**	***Graph Matrics***
1		1		1		1
2				1		0
3	2					1
4			1		1	1
5						0
					Jumlah	**3**

V(G) = Jumlah *graph matrics* + 1

V(G) = 3 + 1

V(G) = 4

Tabel 4.18 Hasil uji *White Box* penerapan metode *Single Exponential Smoothing*

No. Path	Path	Hasil yang diharapkan	Hasil uji kasus	Keterangan
1	1, 2, 3, 4, 5, 6, 9, 10, 11, 12	Menghasilkan Path	Menghasilkan Path	Sesuai yang diharapkan
2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12	Menghasilkan Path	Menghasilkan Path	Sesuai yang diharapkan
3	1, 2, 3, 4, 5, 6, 9, 10, 11, 8, 1, 2, 3, 4, 5, 6, 9, 10, 11, 12	Menghasilkan Path	Menghasilkan Path	Sesuai yang diharapkan
4	1, 2, 3, 4, 5, 6, 9, 10, 11, 8, 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12	Menghasilkan Path	Menghasilkan Path	Sesuai yang diharapkan

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. Hasil wawancara terhadap *Admin*

1. Apakah aplikasi yang dibangun mudah dalam pengelolaan sistem ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang sudah dibangun sudah optimal, sehingga dapat memudahkan dalam pengelolaan sistem.

2. Apakah aplikasi yang dibangun mudah dalam pemeliharaan sistem ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sudah dilengkapi dengan *backup database* sehingga mudah dalam membantu proses pemeliharaan sistem.

3. Apakah aplikasi yang dibangun sudah sesuai dengan kebutuhan ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun cukup sesuai dengan kebutuhan saat ini, karena selama ini perusahaan masih banyak melakukan kegiatan perkantoran secara manual.

4. Apakah aplikasi yang dibangun mudah digunakan ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sudah sesuai dengan yang diinginkan dan diharapkan, sehingga mudah untuk digunakan.

5. Apakah halaman aplikasi sudah terstruktur dengan baik ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sudah cukup baik dalam tata letak halaman dan dalam langkah – langkah prosesnya.

B. Hasil wawancara terhadap *Manager*

1. Apakah aplikasi dapat membantu meramalkan persediaan unit mobil ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sangat membantu perusahaan dalam proses meramalkan persediaan unit mobil secara kuantitatif.

2. Apakah aplikasi dapat membantu memberikan informasi data penjualan ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sangat membantu dalam memberikan informasi data penjualan kendaraan secara cepat, dan dapat membantu pencariaan data penjualan yang dibutuhkan secara praktis dan cepat.

3. Apakah aplikasi yang dibangun sudah sesuai dengan kebutuhan ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sudah sesuai dengan kebutuhan pihak manager.

4. Apakah aplikasi yang dibangun mudah digunakan ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sudah sesuai dengan yang diharapkan, sehingga mudah untuk digunakan.

5. Apakah halaman aplikasi sudah terstruktur dengan baik ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sudah cukup baik dalam langkah – langkah prosesnya (terstruktur).

C. Hasil wawancara terhadap Administrasi Penjualan

1. Apakah aplikasi dapat membantu mengolah data penjualan ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun dapat membantu dalam proses pengolahan data penjualan.

2. Apakah aplikasi dapat membantu membuat laporan data penjualan ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun dapat membantu dalam proses pembuatan laporan data penjualan sesuai dengan yang diinginkan.

3. Apakah aplikasi yang dibangun sudah sesuai dengan kebutuhan ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sudah cukup sesuai dengan yang dibutuhkan.

4. Apakah aplikasi yang dibangun mudah digunakan ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sudah sesuai dengan yang diharapkan, sehingga mudah untuk digunakan.

5. Apakah halaman aplikasi sudah terstruktur dengan baik ?

   Hasil wawancara :

   Berdasarkan hasil wawancara dengan pertanyaan diatas maka dapat disimpulkan bahwa sistem yang telah dibangun sudah baik dalam langkah – langkah prosesnya (terstruktur).

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, menjadikan data – data yang diolahnya tersusun secara rapih didalam *database* sehingga mudah dan cepat dalam mendapatkan informasinya dan juga bermanfaat untuk dapat diolah kembali sebagai data historis pada proses peramalan kendaraan, informasi hasil peramalan yang dihasilkan mudah dimengerti dan sesuai dengan yang diharapkan, memenuhi kebutuhan, terlihat menarik, mudah digunakan dan sangat bermanfaat.

4.2.6 Pengujian Peramalan

Pengujian proses peramalan dilakukan dengan membandingkan hasil perhitungan dari aplikasi peramalan yang menggunakan metode peramalan *Single Exponential Smoothing* dengan hasil perhitungan dari metode *Double Exponential Smoothing* dan *Triple Exponential Smoothing*, serta membandingkan hasil perhitungan peramalan dengan data aktual (*real*) untuk periode yang sudah diramalkan.

1. **Pengujian Pertama**

   Pegujian pertama dilakukan untuk mengetahui metode mana yang memiliki nilai *error* yang paling kecil antara Metode *Single Exponential Smoothing, Double Exponential Smoothing* dan *Triple Exponential Smoothing*. (perhitungan metode *Double Exponential Smoothing* dan *Triple Exponential Smoothing* dilampirkan pada bagian Lampiran E)

   Pada tabel – tabel dibawah ini akan ditunjukkan hasil perhitungan kesalahan / *error* menggunakan metode (MSE) dari setiap metode yang dibandingkan.

Tabel 4.17 Hasil perhitungan kesalahan / *error* dengan mengunakan metode *Single Exponential Smoothing*

Tipe Kendaraan	***Single Exponential Smoothing***				
	α = 0.1	α = 0.3	α = 0.5	α = 0.7	α = 0.9
T 120 PU	267.87	252.47	**242.74**	246.21	253.13
L 300 PU	**40.81**	41.48	44.77	50.04	58.45

Tabel 4.18 Hasil perhitungan kesalahan / *error* dengan mengunakan metode *Double Exponential Smoothing*

Tipe Kendaraan	***Double Exponential Smoothing***				
	α = 0.1	α = 0.3	α = 0.5	α = 0.7	α = 0.9
T 120 PU	279.35	**276.97**	317.6	363.15	389.55
L 300 PU	**40.9**	50.75	66.32	94.13	149.41

Tabel 4.19 Hasil perhitungan kesalahan / *error* dengan mengunakan metode *Triple Exponential Smoothing*

Tipe Kendaraan	***Triple Exponential Smoothing***				
	α = 0.1	α = 0.3	α = 0.5	α = 0.7	α = 0.9
T 120 PU	**287.64**	336.22	422.88	500.24	551.23
L 300 PU	**42.8**	65.13	103.54	196.4	425.27

Dari hasil perhitungan kesalahan / *error* menggunakan metode (MSE) dari setiap metode yang dibandingkan pada tabel – tabel diatas, terlihat bahwa metode *Single Exponential Smoothing* memiliki nilai kesalahan / *error* yang lebih kecil dibandingkan dengan kedua metode yang lainnya.

Dengan ini dapat disimpulkan dari hasil perhitungan kesalahan / *error* menggunakan metode (MSE) antara metode *Single Exponential Smoothing*, *Double Exponential Smoothing* dan *Triple Exponential Smoothing* bahwa metode *Single Exponential Smoothing* lah yang lebih baik untuk digunakan pada kasus ini.

2. Pengujian Kedua

Pegujian kedua dilakukan untuk mengetahui hasil peramalan yang sudah dihitung dengan data aktual (*real*) untuk periode yang sudah diramalkan.

Tabel 4.20 Perbandingan hasil peramalan dengan data aktual.

Tipe Kendaraan	**Hasil Peramalan *Single Exponential Smoothing* Pada Bulan Maret 2012**	**Data Aktual di PT. Wicaksana Berlian Motor Pada Bulan Maret 2012**	**Kesalahan / *Error* Menggunakan Metode (MSE)**
T 120 PU	*56 Unit*	*58 Unit*	*0,3*
L 300 PU	*21 Unit*	*20 Unit*	*0,08*

Dengan hasil peramalan yang sudah dibandingkan diatas, terlihat bahwa tingkat *error* yang ditunjukan kecil dan hasil peramalan yang mendekati data aktualnya. Dengan ini dapat disimpulkan bahwa metode peramalan yang telah digunakan memiliki tingkat akurasi yang cukup baik terhadap kasus ini.