BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Implementasi dilakukan setelah proses perancangan dari sistem yang akan dibuat. Implementasi bertujuan untuk mengetahui apa saja yang akan diterapkan dalam sistem.

5.1.1 Batasan Implementasi

Tidak semua perangkat lunak dapat mengerjakan apa yang diperintahkan oleh *user*. Setiap kelebihan yang ada pada setiap perangkat lunak pasti mempunyai suatu batasan. Begitu pula dengan sistem informasi penjualan suku cadang dan pelayanan jasa *service* ini. Adapun batasannya adalah sebagai berikut:

1. Dalam sistem ini tidak dapat menghapus transaksi yang telah ada.
2. Implementasi *database* dari sistem ini menggunakan MySQL.
3. Sistem Informasi ini hanya untuk penjualan suku cadang dan pelayanan jasa *service* di Rayhan's toys saja

5.1.2 Impelmentasi Perangkat Lunak

Implementasi Perangkat Lunak dari Sistem ini memerlukan beberapa perangkat lunak pendukung diantaranya.

a. Kebutuhan untuk komputer server
   1. Sistem Operasi : Microsoft Windows XP SP 3/Windows 7
   2. Database menggunakan MySQL
   3. Laporan Menggunakan iReport 4.2.0
b. Kebutuhan Client
   1. Sistem Operasi : Microsoft Xp *Service* Pack 3/Windows 7
   2. Laporan Menggunakan iReport 4.2.0

5.1.3 Implementasi Perangkat Keras

Implementasi Perangkat Keras untuk mendukung dari sistem ini adalah sebagai berikut :

a. Server
   1. Menggunakan Processor Intel Core 2 Duo 2.0 Ghz *or Higher*.
   2. Menggunakan RAM minimal 2Gb
   3. Harddisk minimal 80Gb
   4. VGA minimal 256 *or Higher*
   5. Monitor, mouse, keyboard
   6. Printer
b. Client
   1. Menggunakan Processor Intel Core 2 Duo 2.0 Ghz *or Higher*.
   2. Menggunakan RAM minimal 1 Gb.
   3. Hard Disk minimal 40 Gb.
   4. VGA minimal 256 *or Higher*
   5. Monitor, mouse, keyboard
c. Konfigurasi Jaringan
   1. Port Hub
   2. Kabel UTP
   3. Connector RJ45

5.1.4 Implementasi Basis Data

Implementasi basis data sebagai berikut :

```
CREATE TABLE IF NOT EXISTS

`rayhan_detailtransaksi` (

`noTransaksi` varchar(23) NOT NULL,

`kodeSC` varchar(30) NOT NULL,

`jumlahBeli` int(23) NOT NULL,

`biayaPasang` varchar(23) NOT NULL,

`total` varchar(23) NOT NULL,

KEY `noTransaksi` (`noTransaksi`),

KEY `kodeSC` (`kodeSC`)

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

```
CREATE TABLE IF NOT EXISTS
`rayhan_detailtransaksiservice` (`noTransaksi` varchar(20) NOT NULL,
`kodeService` varchar(20) NOT NULL,
`jenisRC` varchar(20) NOT NULL,
`mekanik` varchar(20) NOT NULL,
`harga` varchar(10) NOT NULL,
`hargaSukuCadang` varchar(10) NOT NULL,
`total` varchar(10) NOT NULL,
KEY `noTransaksi` (`noTransaksi`),
KEY `noTransaksi_2` (`noTransaksi`),KEY
`kodeService` (`kodeService`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS
`rayhan_konsumenjasa` (`no` int(11) NOT NULL AUTO_INCREMENT,
`noTransaksi` varchar(20) NOT NULL,
`tanggal` date NOT NULL,
`namaKonsumen` varchar(20) NOT NULL,
`jenisRC` varchar(20) NOT NULL,
`kodeService` varchar(11) NOT NULL,
`namaService` varchar(20) NOT NULL,
`mekanik` varchar(20) NOT NULL,
 PRIMARY KEY (`no`)
```

```
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=14 ;

CREATE TABLE IF NOT EXISTS
`rayhan_pegawai` (`nama` varchar(25) NOT NULL,
`alamat` varchar(100) NOT NULL,
`telp` varchar(13) NOT NULL,
  `jabatan` varchar(10) NOT NULL) ENGINE=InnoDB DEFAULT
CHARSET=latin1;


CREATE TABLE IF NOT EXISTS
`rayhan_servis` ( `kodeServis` varchar(20) NOT NULL,
`namaServis` varchar(20) NOT NULL,
`jenisServis` varchar(30) NOT NULL,
`harga` varchar(20) NOT NULL,
PRIMARY KEY (`kodeServis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS
`rayhan_sukucadang` (`kodeSC` varchar(10) NOT NULL,
`namaSC` varchar(100) NOT NULL,
`hargaBeliSC` varchar(10) NOT NULL,
`hargaJualSC` varchar(10) NOT NULL,
```

```
`stok` int(10) NOT NULL,
`biayaPasang` varchar(10) NOT NULL,
`jenisMainan` varchar(11) NOT NULL,
PRIMARY KEY (`kodeSC`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS
`rayhan_transaksi` (`no` int(11) NOT NULL AUTO_INCREMENT,
`noTransaksi` varchar(20) NOT NULL,
`tanggalTransaksi` date NOT NULL,
`total` varchar(20) NOT NULL,
PRIMARY KEY (`noTransaksi`),
KEY `no` (`no`)) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ;

CREATE TABLE IF NOT EXISTS
`rayhan_transaksiservis` ( `noServis` varchar(20) NOT NULL,
`tanggalServis` varchar(15) NOT NULL,
`total` varchar(20) NOT NULL,
PRIMARY KEY (`noServis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS
`rayhan_user` (`id` varchar(7) NOT NULL,
```

```
`username` varchar(10) NOT NULL,
`password` varchar(15) NOT NULL,
`status` varchar(13) NOT NULL,
`nama` varchar(25) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
ALTER TABLE
`rayhan_service` ADD CONSTRAINT
`rayhan_service_ibfk_1` FOREIGN KEY (`kodeServis`) REFERENCES
`rayhan_detailtransaksiservis` (`kodeServics`) ON DELETE CASCADE ON
UPDATE CASCADE;
```

```
ALTER TABLE
`rayhan_sukucadang`ADD CONSTRAINT
`rayhan_sukucadang_ibfk_1` FOREIGN KEY (`kodeSC`) REFERENCES
`rayhan_detailtransaksi` (`kodeSC`) ON DELETE CASCADE ON UPDATE
CASCADE;
```

```
ALTER TABLE
```

```
`rayhan_transaksi`ADD CONSTRAINT
`rayhan_transaksi_ibfk_1` FOREIGN KEY (`noTransaksi`) REFERENCES
`rayhan_detailtransaksi` (`noTransaksi`) ON DELETE CASCADE ON UPDATE
CASCADE;

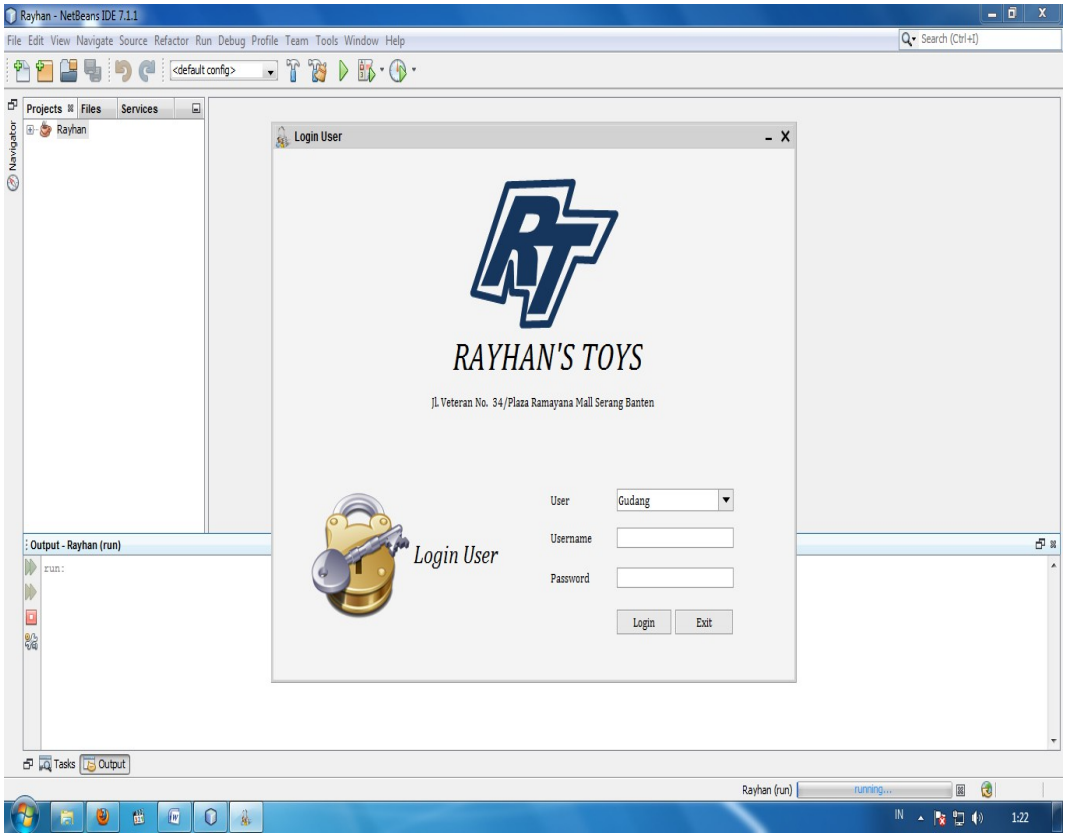
ALTER TABLE
`rayhan_transaksiservis`ADD CONSTRAINT
`rayhan_transaksiservis_ibfk_1` FOREIGN KEY (`noServis`) REFERENCES
`rayhan_detailtransaksiservis` (`noTransaksi`) ON DELETE CASCADE ON
UPDATE CASCADE;
```

5.1.5 **Implementasi Antar Muka**

Impelementasi antar muka bertujuan untuk memenuhi kebutuhan *user* agar dapat berinteraksi dengan sebuah sistem. Fasilitas yang diberikan memungkinkan *user* dapat menggunakan fungsi yang terdapat pada sistem. Adapun *interface* dalam sistem ini adalah sebagai berikut :

a. Login

Fungsi login untuk memberikan memberikan hak akses kepada user yang menggunakan sistem.



Gambar 5.1. Login

b. Tampilan Utama

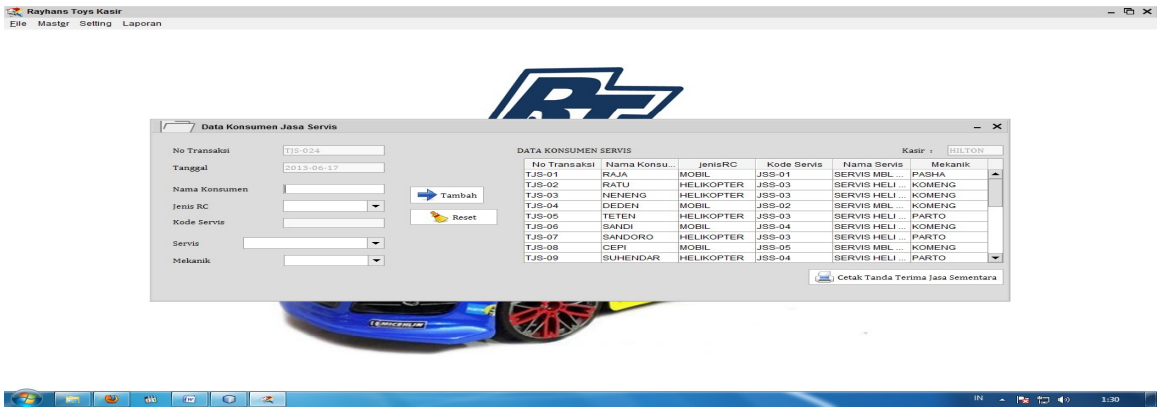
Setelah otentifikasi *user* tervalidasi akan muncul tampilan utama dari program sesuai dengan hak akses *user*. Pada tampilan menu bar terdapat beberapa kategori fungsi diantaranya File, Master, Transaksi, Laporan.



Gambar 5.2. Tampilan utama

c. Tampilan Data Konsumen Jasa *Service*

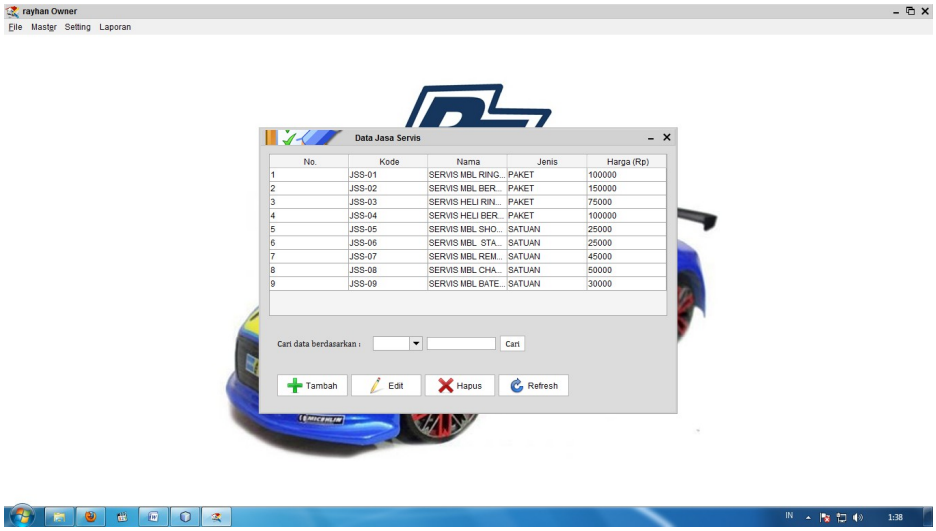
Pada tampilan data konsumen berisikan tentang data konsumen yang membeli jasa *service* pada Rayhan's toys



Gambar 5.3. Data konsumen jasa *service*

d. Tampilan Data Jasa *Service*

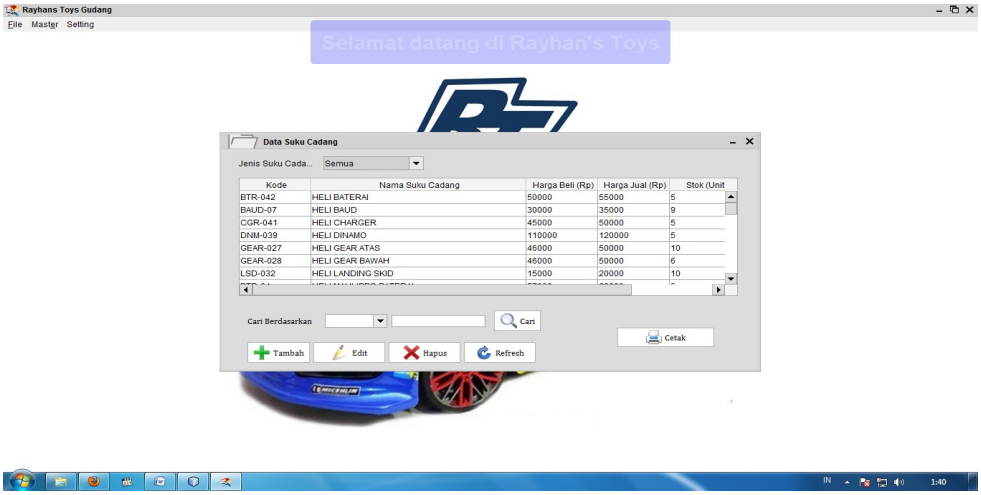
Tampilan ini dibutuhkan sebagai pengidentifikasi jenis *service* dan tarif *service* yang akan diperlukan dalam laporan *service*.



Gambar 5.4. Data Jasa *Service*

e. Tampilan Data Suku Cadang

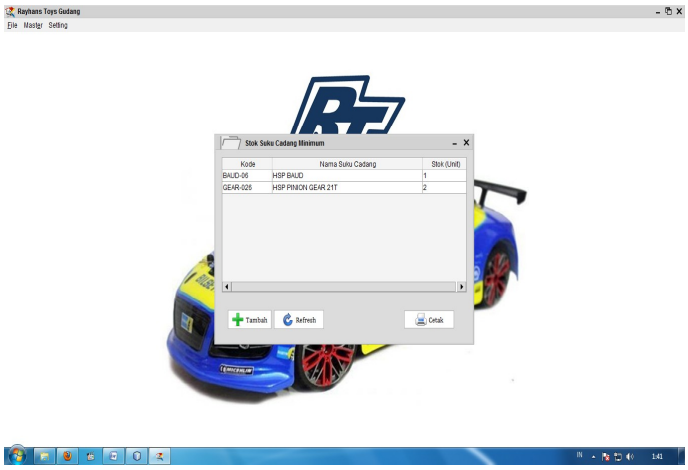
Tampilan ini dibutuhkan sebagai pengidentifikasi data suku cadang yang nantinya akan diperlukan dalam *form* penjualan suku cadang.



Gambar 5.5. Data suku cadang

f. Tampilan Cek Stok Suku Cadang

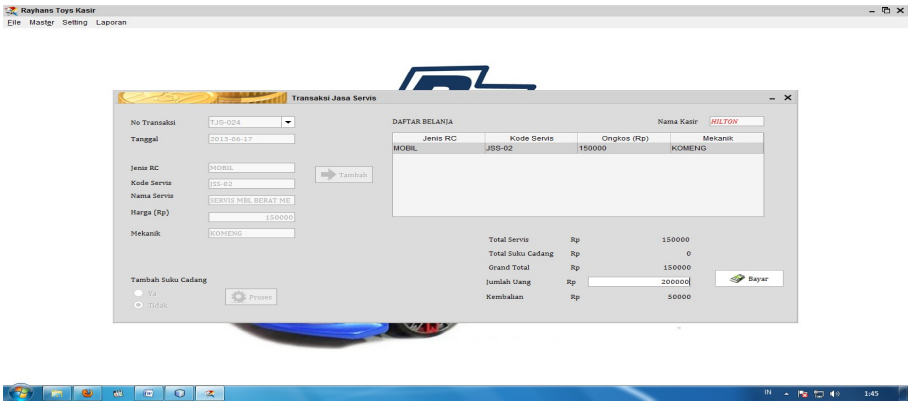
Tampilan cek stok suku cadang berfungsi untuk mengontrol suku cadang yang habis atau akan segera habis, sehingga kekosongan suku cadang di gudang dapat ditangani dengan cepat.



Gambar 5.6. Cek stok suku cadang

g. Tampilan Transaksi Jasa *Service*

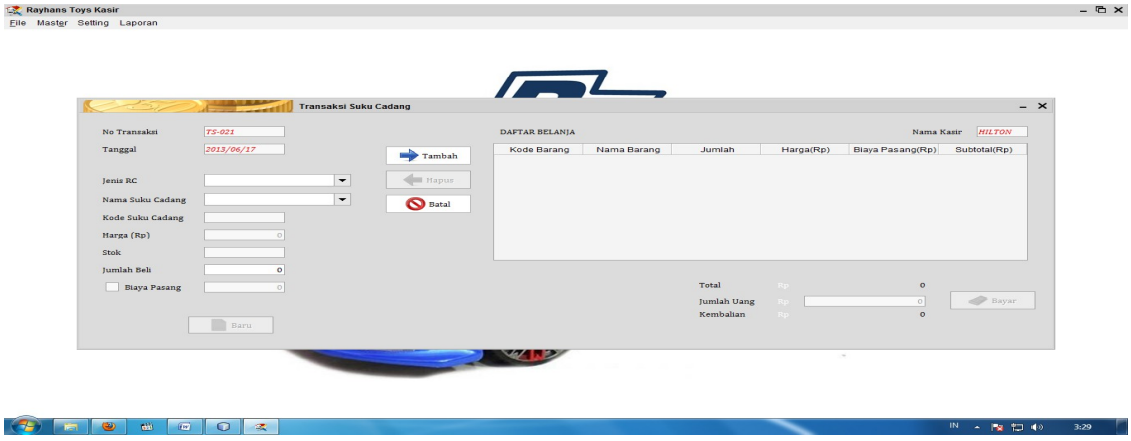
Tampilan transaksi jasa *service* berperan setelah mekanik selesai melakukan operasi *service* kemudian konsumen melakukan transaksi berdasarkan tanda terima jasa *service* sementara.



Gambar 5.7. Transaksi jasa *service*

h. Tampilan Transaksi Suku Cadang

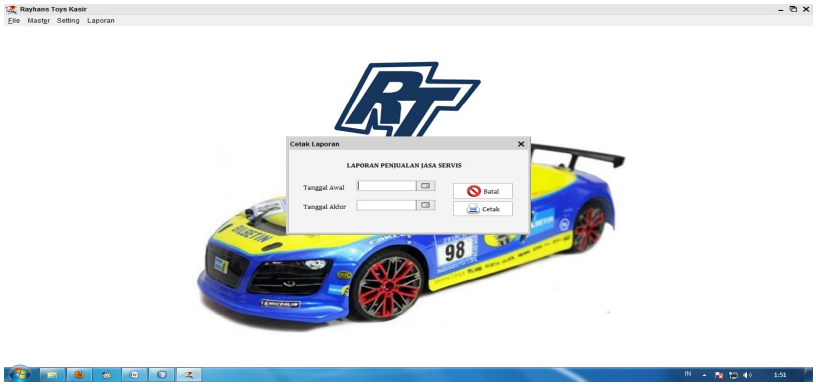
Tampilan Penjulan Suku Cadang berfungsi sebagai *form* transaksi penjualan suku cadang pada Rayhan’s toys.



Gambar 5.8. Transaksi suku cadang

i. Tampilan Laporan Penjualan Jasa *Service*

Tampilan ini dioperasikan oleh kasir dan owner sebagai pengecekan penjualan jasa *service* yang telah dilakukan.

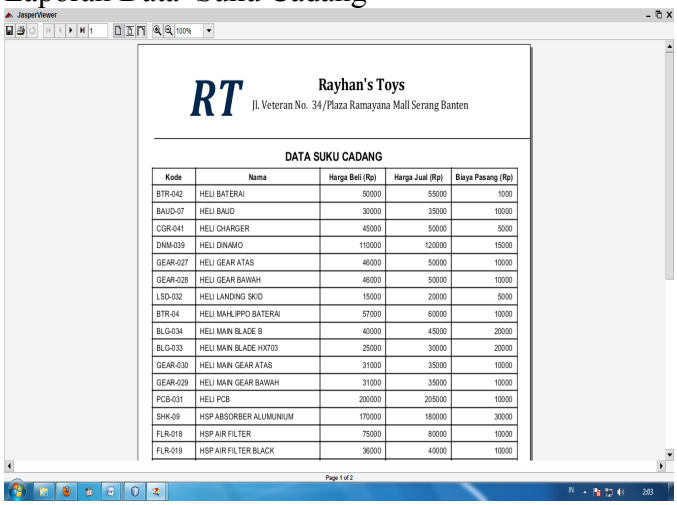


Gambar 5.9. Laporan penjualan jasa *service*

j. Tampilan Laporan

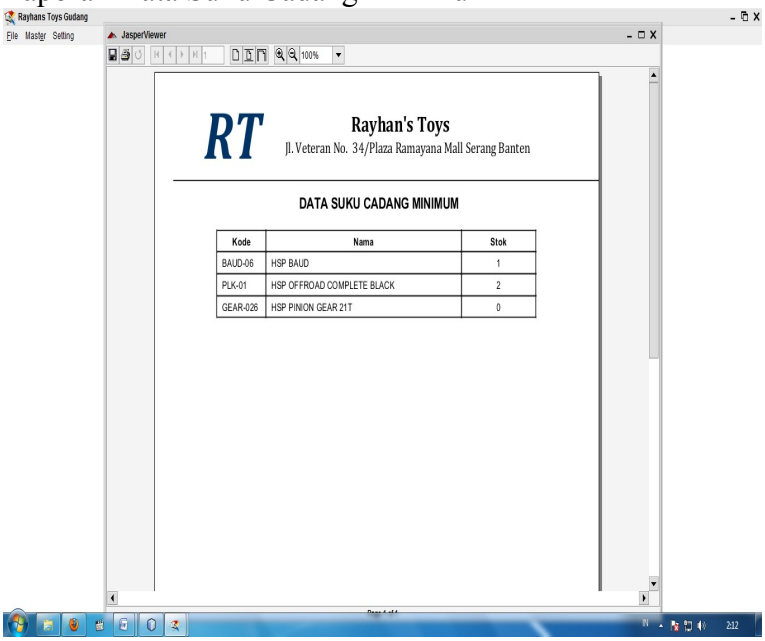
Tampilan laporan penjualan barang ini digunakan oleh bagian kasir, gudang dan owner untuk mengetahui laporan penjualan suku cadang dan jasa *service* yang sudah dilakukan dan semua data barang.

1. Laporan Data Suku Cadang



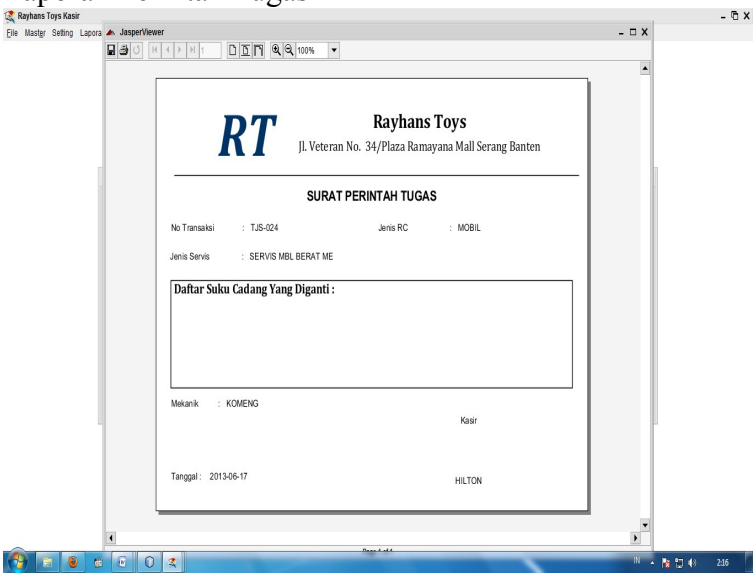
Gambar 5.10. Laporan Data Suku Cadang

2. Laporan Data Suku Cadang Minimum



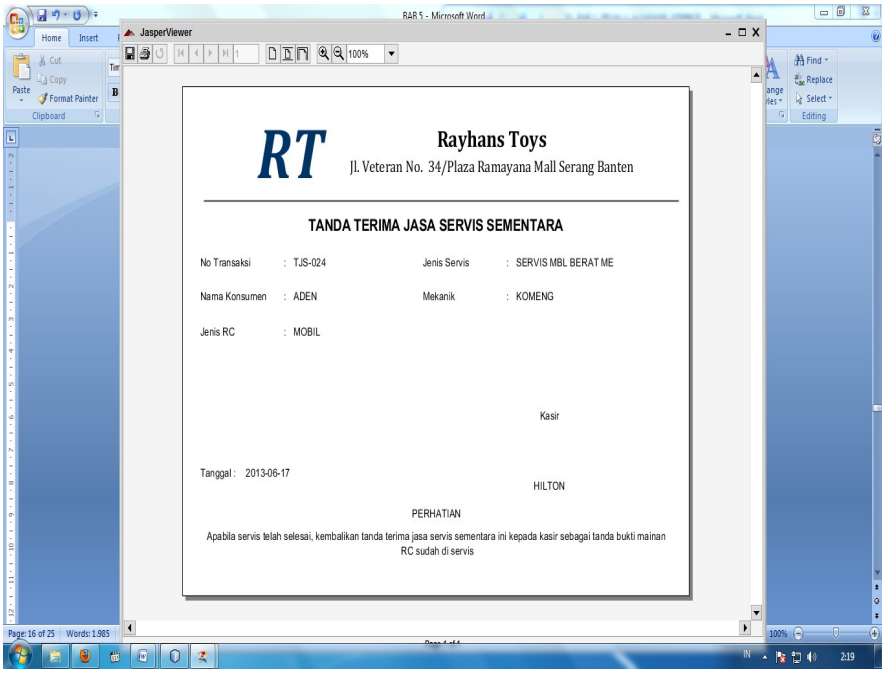
Gambar 5.11. Laporan Data Suku Minimum

3. Laporan Perintah Tugas



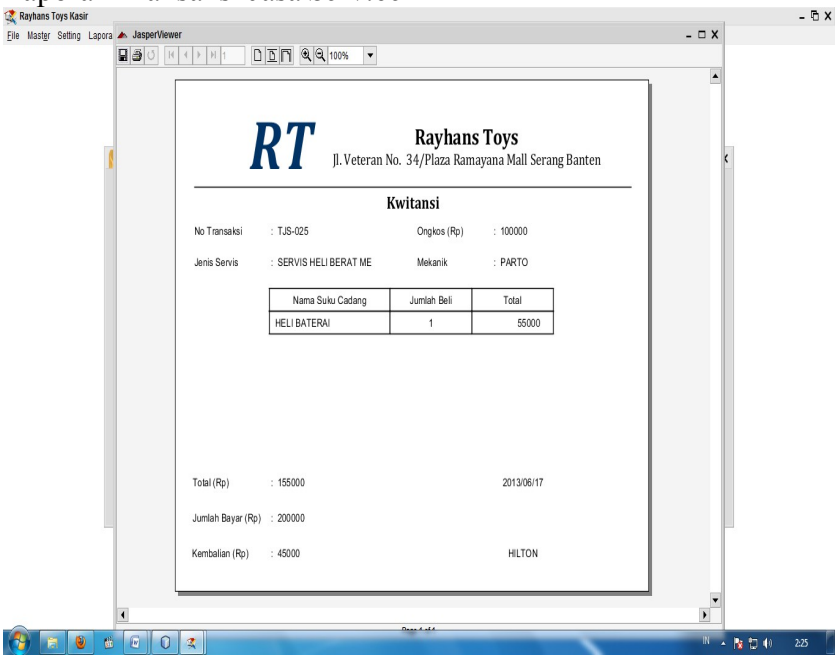
Gambar 5.12. Laporan Perintah Tugas

4. Laporan Tanda Terima Jasa *Service* Sementara



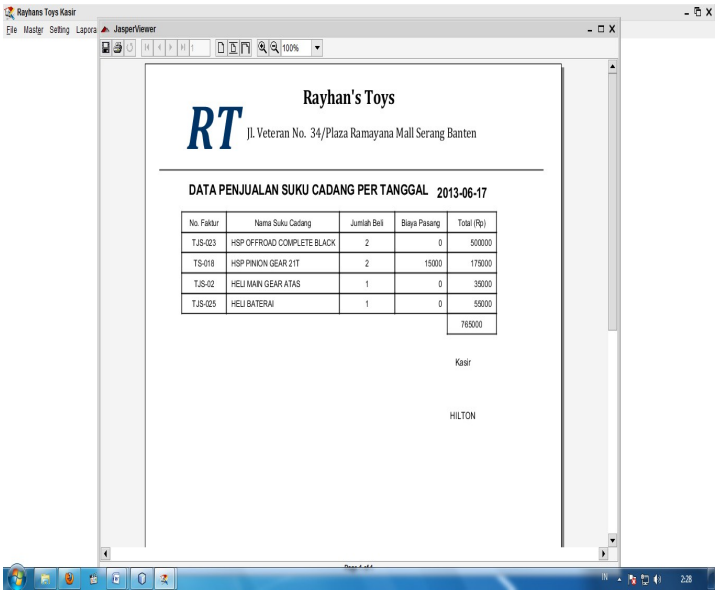
Gambar 5.13. Laporan Tanda Terima Jasa *Service* Sementara

5. Laporan Transaksi Jasa *Service*



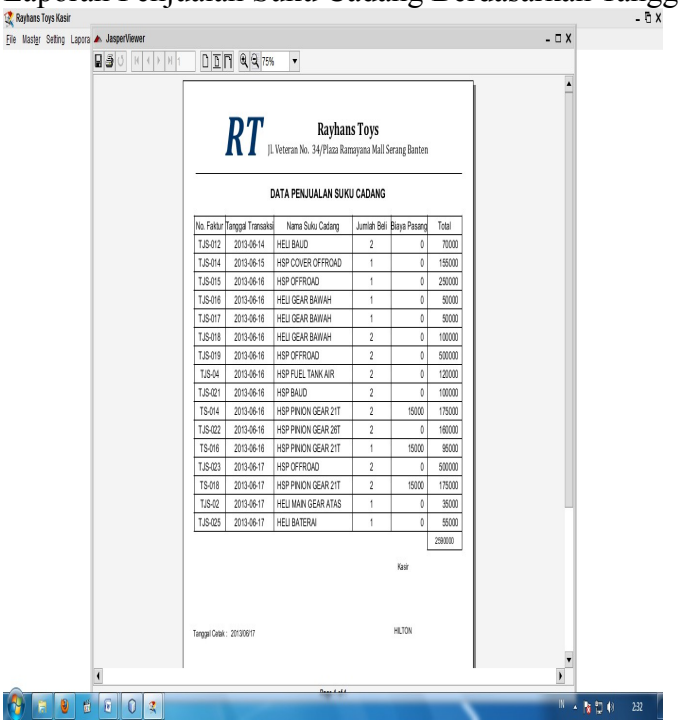
Gambar 5.14. Laporan Transaksi Jasa *Service*

6. Laporan Penjualan Suku Cadang Hari Ini



Gambar 5.15. Laporan Penjualan Suku Cadang Hari Ini

7. Laporan Penjualan Suku Cadang Berdasarkan Tanggal



Gambar 5.16. Laporan Penjualan Suku Cadang Berdasarkan Tanggal

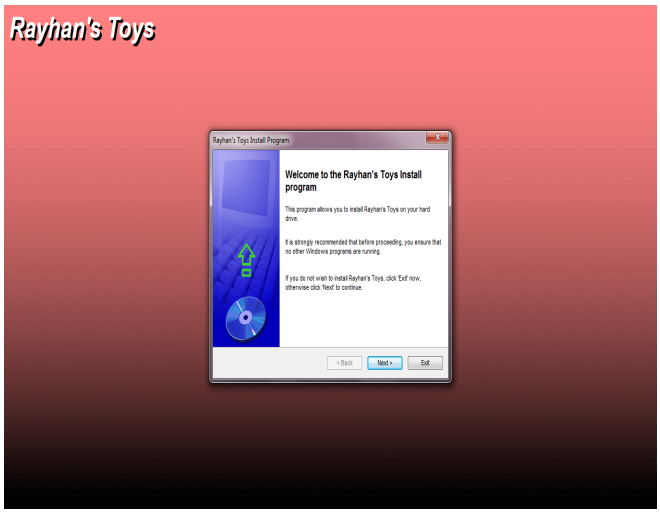
8. Laporan Penjualan Suku Cadang Semua

Gambar 5.17. Laporan Penjualan Suku Cadang Semua

5.1.6 Implementasi Instalasi Program

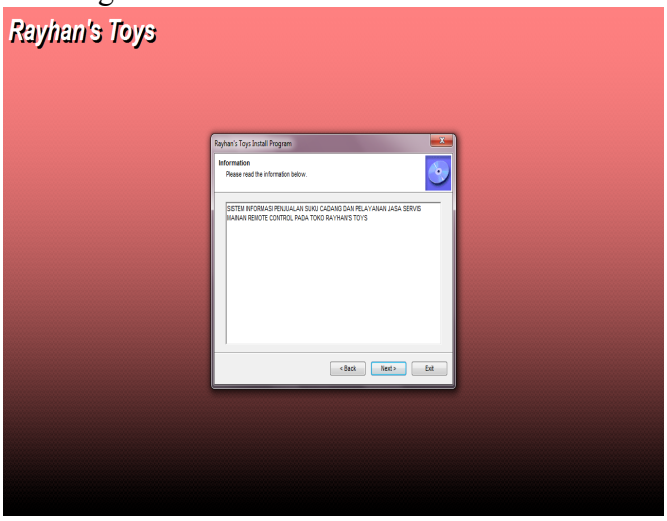
1. Pilih SETUP, kemudian klik 2 kali atau tekan Enter.

2. Kemudian muncul gambar di bawah ini.



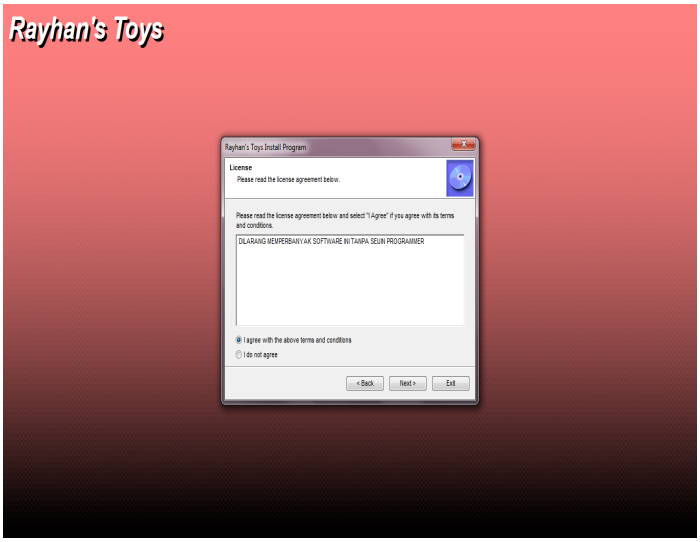
Gambar 5.18. Install tahap 1

3. Klik Next dan muncul gambar di bawah ini.



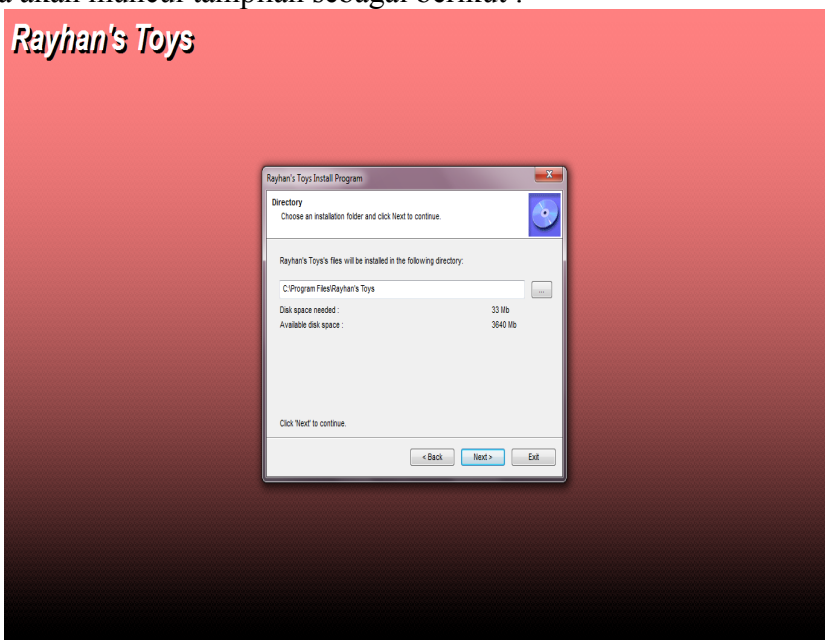
Gambar 5.19. Instal tahap 2

4. Klik next dan muncul gambar dibawah ini.



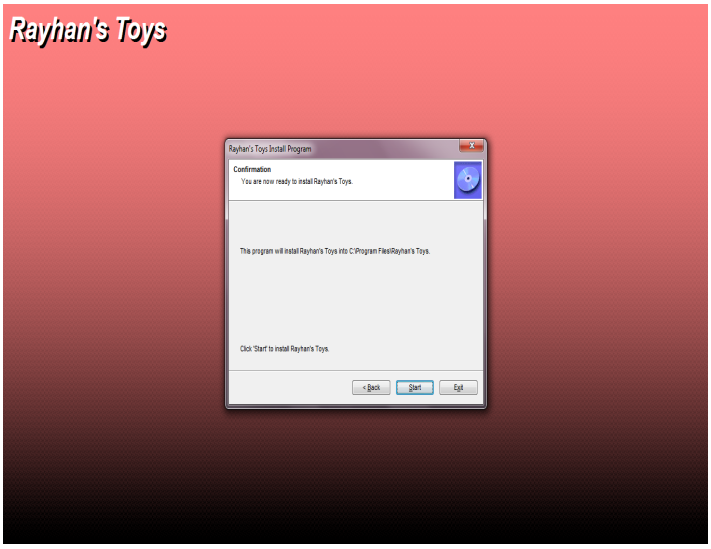
Gambar 5.20. Instal tahap 3

5. Pilih "*I agree with the above terms and conditions*", kemudian klik next, maka akan muncul tampilan sebagai berikut :



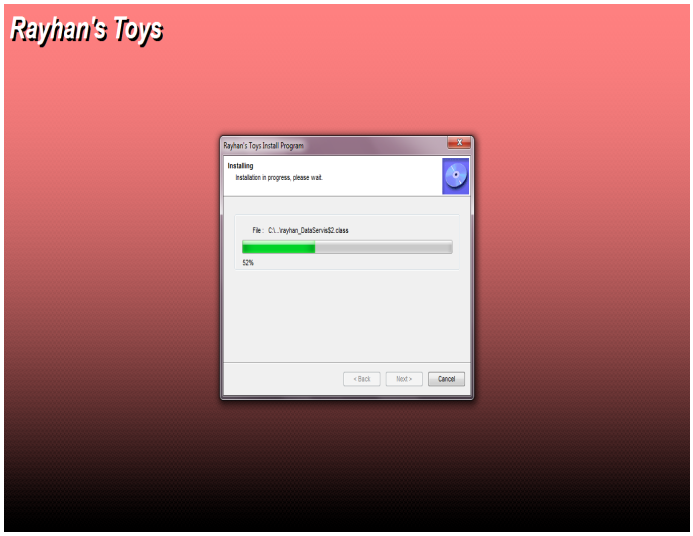
Gambar 5.21. Instal tahap 4

6. Klik next kemudian mucul gambar dibawah ini.



Gambar 5.22. Instal tahap 5

7. Klik Start tunggu proses instalasi selesai kemudian muncul gambar dibawah ini.



Gambar 5.23. Instal tahap 6

8. Klik *Finish* setelah proses instalasi selesai.

5.1.7 Penggunaan Program

Untuk prosedur pelayanan *service*, pada form data konsumen jasa *service*, user meginputkan data konsumen dan data jenis RC dengan menekan tombol tambah lalu mengisikan *field – field* yang tersedia kemudian klik simpan. Setelah

mengisikan data konsumen dan jenis RC saatnya user mencetak tanda terima jasa *service* sementara untuk konsumen dan surat perintah tugas untuk mekanik. Prosedur selanjutnya user menjalankan form transaksi *service* lalu klik tombol baru dan menginputkan nomor transaksi pada *field* sesuai dengan tanda terima jasa *service* sementara, kemudian tambahkan penjualan ke tabel daftar beli, lalu lakukan proses transaksi pembayaran *service* dengan klik tombol bayar kemudian akan keluar faktur transaksi jasa *service* untuk dicetak.

Prosedur selanjutnya yaitu Penjualan Suku Cadang, user hanya mengoperasikan form transaksi suku cadang, klik tombol tambah dan user menyesuaikan *field* - *field* yang tersedia kemudian klik tombol simpan setelah itu tombol bayar untuk mencetak transaksi.

Prosedur terakhir yaitu prosedur pencetakkan laporan *service* dan suku cadang yang dikelola oleh manajer, user menggunakan dua form laporan *service* dan laporan penjualan suku cadang. Masing – masing form terdapat tanggal awal dan tanggal akhir yang harus disesuaikan oleh kebutuhan user dalam mencetak laporan setelah itu klik cetak.

5.2 Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak. Dengan dilakukannya pengujian terhadap sistem yang dibuat, akan meminimalisir kesalahan yang mungkin akan muncul saat program

dijalankan. Pengujian dilakukan untuk mengetahui keandalan program yang dibuat. Agar fungsional sistem berjalan baik, dan tidak mengalami kesalahan fungsi atau *error* saat penggunaan program. tujuan pengujian perangkat lunak ada dua macam yaitu tujuan langsung dan tidak langsung.

A. **Tujuan langsung**

1. Identifikasi dan menemukan beberapa kesalahan yang mungkin ada dalam perangkat lunak yang diuji.
2. Setelah perangkat lunak dibetulkan, diidentifikasi lagi kesalahan dan diuji kembali untuk menjamin kualitas dari sistem tersebut.
3. Membentuk tes yang efisien dan efektif dengan anggaran dan jadwal yang terbatas

B. **Tujuan tidak langsung**

Mengumpulkan daftar kesalahan untuk digunakan dalam daftar pencegahan kesalahan (tindakan *corrective* dan *preventive*).

5.2.1 Rencana Pengujian

Rencana pengujian yang dilakukan adalah pengujian terhadap faktor akses kontrol, kebenaran data, integrasi database, dan kelanjutan pemrosesan data.

Tabel 5.1. Rencana Pengujian Sistem Informasi Pelayanan Jasa

Kelas Uji	Butir Uji	Tingkat Pengujian	Jenis Pengujian
Akses kontrol	Pengecekan LOGIN user yang telah terdaftar	Integrasi	*Black box*
Integrasi Database	Data Pada Tabel Master dengan Tabel Transaksi Sama	Integrasi	*Black Box*
Kebenaran Data	Data yang diinputkan sesuai data yang pada output	Integrasi	*Black Box*

5.2.2 Kasus dan Hasil Pengujian

Kasus uji yang dilakukan disesuaikan dengan rencana pengujian yang telah dijabarkan sebelumnya. Dibawah ini sampel kasus uji dan hasil pengujian yang dilakukan.

Tabel 5.2. Pengujian Akses Kontrol

Kasus Uji	Langkah Pengujian	Hasil Yang Diharapkan	Hasil Aktual	Kesimpulan
LOGIN (SUKSES)	1. Pengguna harus berada pada halaman login 2. Pada *text box* username dan password masukan *username* dan *password* yang valid Misalnya *username* : kasi *password* : kasir 3. Klik tombol login	Akan tampil sistem dengan menu-menu sesuai dengan hak akses pengguna	Akan tampil sistem dengan menu-menu sesuai hak akses pengguna	[√] Diterima [ ] Ditolak
LOGIN (jenis admin, username atau password	1. Pengguna harus berada pada halaman login 2. Pada *text box username* dan *password* masukan *username* dan *password* yang tidak valid	Akan tampil pesan "Username /password Salah"	Akan tampil pesan Username/ password Salah"	[ ] Diterima [√] Ditolak

yang dimasukan salah) 3.	Misalnya *username* : abcde *password* : 1234 Klik tombol login			

Tabel 5.3. Pengujian Integritas *Database* (Sinkronisasi)

Kasus Uji	Langkah Pengujian	Hasil Yang Diharapkan	Hasil Aktual	Kesimpulan
Data pada 1. tabel master sesuai 2. dengan tabel transkasi	User memasukan data pada tabel master User lainnya melihat data pada tabel transaksi	Jika data pada tabel master telah tersedia, maka pada tabel transaksi telah terdapat data sesuai tabel master	Data dapat disimpan	[√] Diterima [ ] Ditolak
Data ada 1. tabel master tidak sesuai dengan tabel 2. transkasi	Misalnya user memasukan data master User melihat data pada tabel transaksi (Jika tabel master tidak terdapat data, maka dalam tabel transaksi tidak akan dapat melihat hasil dari pemrosesan dari tabel master)	misalnya tabel master tidak diisi data, maka pada tabel transaksi tidak akan muncul data yang dibutuhkan.	Data yang diminta tidak muncul	[ ] Diterima [√] Ditolak

Tabel 5.4. Pengujian Kebenaran Data

Kasus Uji	Langkah Pengujian	Hasil Yang Diharapkan	Hasil Aktual	Kesimpulan
Kebenaran 1. data (data 2. diinputkan 3. secara	User berada pada form master (konsumen) Klik tombol tambah Inputkan data pada semua *text box* yang berfungsi	Akan tampil pesan bahwa data telah berhasil ditambahkan	Akan tampil bahwa data telah berhasil ditambahkan	[√] Diterima [ ] Ditolak

lengkap)	untuk menginputkan data 4. Klik simpan			
Kebenaran Data (data diinputkan tidak lengkap)	1. User berada pada form master (konsumen) 2. Klik tombol tambah 3. Inputkan data pada beberapa *text box* saja yang berfungsi menginputkan data, misalnya no polisi tidak diinputkan 4. Klik simpan	Data tidak akan tersimpan dan akan ada tampilan pesan "Data salah/belum diisi"	Data tidak akan tersimpan dan akan ada tampilan pesan "Data salah/belum diisi"	[ ] Diterima [√] Ditolak

5.2.3 **Kesimpulan Hasil Pengujian**

Seperti yang telah dijelaskan sebelumnya dari tujuan pengujian, dengan sampel diatas diharapkan pengujian yang dilakukan akan menghasilkan sistem yang memiliki fungsional baik, sehingga kesalahan yang mungkin akan muncul menjadi lebih kecil kemungkinannya.