BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Berdasarkan tahapan paradigma yang digunakan dalam penelitian ini, maka tahapan berikutnya stetelah melakukan perncangan adalah tahapan implementasi dan pengujian dari perangkat lunak yang telah dibuat. Ini bertujuan agar sistem yang dibuat dapat sesuai dengan keinginan dan tujuan, maka harus disusunlah suatu langkah ataupun prosedur di dalam penerapan sistem berupa suatu perencanaan implementasi dan pengujian perangkat lunak.

5.1.1 Batasan Implementasi

Dalam mengimplementasi perangkat lunak pemantauan dan pengendalian aplikasi ini ada beberapa hal yang menjadi batasan implementasi yaitu:

1. Tidak semua kebutuhan sistem yang ada di toko buku doa walisongo dapat dikerjakan, tetapi hanya subsistem-subsistem yang mendukung khususnya pada proses penjualan.
2. Database yang digunakan dalam pengimplementasian ini adalah Mysql. Pengaksesan *database* dilakukan dengan mengaktifkan phpmyadmin.

5.1.2 Implementasi Perangkat Lunak

Dalam penelitian ini, perancangan sistem informasi penjualan dibangun dengan menggunakan bahasa pemrograman PHP yang dinamis dan dapat mudah untuk dikembangkan. Sedangkan untuk mengelola seluruh data yang ada dalam

sistem ini, penulis menggunakan SQL Server, karena dianggap kompetibel dengan bahasa pemrograman yang digunakan.

5.1.3 Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi antara lain:

1. Processor minimal intel Pentium IV 2.2 Ghz (atau sejenisnya).
2. Harddisk minimal 250 GB
3. Memori (RAM) minimal 2 GB
4. VGA Card terpasang minimal 512 MB

5.1.4 Implemntasi Basis Data

Sebuah basis data atau *database*, sangat dibutuhkan dalam sebuah sistem informasi yang terkomputerisasi. Pada sistem penjualan berbasis *website* ini dibangun database demi terciptanya sistem informasi yang terintegrasi. Berikut adalah basis data menggunakan bahasa SQL dalam sistem informasi penjualan berbasis *website* :

```
CREATE TABLE IF NOT EXISTS `konsumen` (
 `id_konsumen` varchar(15) NOT NULL,
 `uname_k` varchar(30) NOT NULL,
 `pass_k` varchar(30) NOT NULL,
 `nama_k` varchar(30) NOT NULL,
 `id_provinsi` varchar(15) NOT NULL,
 `id_kota` varchar(15) NOT NULL,
 `alamat` varchar(50) NOT NULL,
 `kd_pos` varchar(30) NOT NULL,
```

```
  `email` varchar(30) NOT NULL,
  `tlpn` int(20) NOT NULL,
  PRIMARY KEY (`id_konsumen, id_provinsi, id_kota`)
  KEY `id_konsumen` (`id_konsumen`)
  KEY `id_provinsi` (`id_provinsi`)
  KEY `id_kota` (`id_kota`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `buku` (
  `kd_buku` varchar(15) NOT NULL,
  `kat_buku` varchar(30) NOT NULL,
  `penulis` varchar(30) NOT NULL,
  `kd_penerbit` varchar(30) NOT NULL,
  `judul` varchar(100) NOT NULL,
  `isbn` varchar(30) NOT NULL,
  `tgl_terbit` date NOT NULL,
  `hal` varchar(30) NOT NULL,
  `deskripsi` varchar(300) NOT NULL,
  `harga` int(20) NOT NULL,
  `stok` int(20) NOT NULL
  `gambar` varchar(70) NOT NULL
  PRIMARY KEY (`kd_buku, kat_buku, kd_penerbit`)
  KEY `kd_buku` (`kd_buku`)
  KEY `kat_buku` (`kat_buku`)
  KEY `kd_penerbit` (`kd_penerbit`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `pembelian` (
  `no_pembelian` varchar(15) NOT NULL,
```

```
  `id_konsumen` varchar(15) NOT NULL,
  `sub_tot` int(30) NOT NULL,
  PRIMARY KEY (`no_pembelian`)
  KEY `no_pembelian` (`no_pembelian`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `detail pembelian` (
  `no_urut` tinyint NOT NULL,
  `no_pembelian` varchar(15) NOT NULL,
  `kd_buku` varchar(15) NOT NULL,
  `jumlah` int(20) NOT NULL,
  `harga_s` int(20) NOT NULL,
  PRIMARY KEY (`no_urut`,`no_pembelian, kd_buku`)
  KEY `no_urut` (`no_urut`)
  KEY `no_pembelian` (`no_pembelian`)
  KEY `kd_buku` (`kd_buku`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `verifikasi` (
  `no_verifikasi` varchar(15) NOT NULL,
  `nik` varchar(15) NOT NULL,
  `status_ver` varchar(20) NOT NULL,
  PRIMARY KEY (`no_verifikasi`)
  PRIMARY KEY (`nik `)
  KEY `no_verifikasi` (`no_verifikasi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `pembayaran` (
  `no_pembayaran` varchar(15) NOT NULL,
```

```
  `id_konsumen` varchar(15) NOT NULL,
  `nik` varchar(15) NOT NULL,
  `no_pembelian` varchar(15) NOT NULL,
  `no_rek` varchar(20) NOT NULL,
  `tgl_bayar` date NOT NULL,
  `jam_bayar` time NOT NULL,
  `total_bayar` int(30) NOT NULL,
  `no_verifikasi` varchar(15) NOT NULL,
  PRIMARY KEY (`no_pembayaran, id_konsumen, no_pembelian,
                    no_verifikasi`)
  KEY `no_pembayaran` (`no_pembayaran`)
  KEY `id_konsumen` (`id_konsumen`)
  KEY `no_pembelian` (`no_pembelian`)
  KEY `no_verifikasi` (`no_verifikasi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `periode` (
  `no_periode` varchar(15) NOT NULL,
  `no_pembayaran` varchar(15) NOT NULL,
  `nik` varchar(15) NOT NULL,
  `tgl_jual` date NOT NULL,
  PRIMARY KEY (`no_periode, kd, no_pembayaran, nik`)
  KEY `no_periode` (`no_periode`)
  KEY `kd_pembayaran` (`no_pembayaran`)
  KEY `nik` (`nik`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `provinsi` (
  `id_provinsi` varchar(15) NOT NULL,
```

```
  `nama_provinsi` varchar(50) NOT NULL,
  PRIMARY KEY (`id_provinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `kota` (
  `id_kota` varchar(15) NOT NULL,
  `id_provinsi` varchar(15) NOT NULL,
  `nama_kota` varchar(50) NOT NULL,
  PRIMARY KEY (`id_kota , id_provinsi`)
  KEY `id_kota` (`id_kota`)
  KEY `id_provinsi` (`id_provinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `petugas` (
  `nik` varchar(15) NOT NULL,
  `uname_p` varchar(15) NOT NULL,
  `pass_p` varchar(50) NOT NULL,
  PRIMARY KEY (`nik`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `kategori` (
  `kat_buku` varchar(30) NOT NULL,
  `ket` varchar(40) NOT NULL,
  PRIMARY KEY (`kat_buku`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `penerbit` (
  `kd_penerbit` varchar(15) NOT NULL,
  `penerbit` varchar(40) NOT NULL,
```

```
  `kota_t` varchar(50) NOT NULL,
  PRIMARY KEY (`kd_penerbit`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
-- Constraints for dumped tables
--
--
-- Constraints for table ``
--
ALTER TABLE `konsumen`
  ADD CONSTRAINT `konsumen_ibfk_1` FOREIGN KEY (`id_provinsi`) REFERENCES `provinsi` (`id_provinsi`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `konsumen_ibfk_2` FOREIGN KEY (`id_kota`) REFERENCES `kota` (`id_kota`) ON DELETE CASCADE ON UPDATE CASCADE;
-- Constraints for table ``
--
ALTER TABLE `buku`
  ADD CONSTRAINT `buku_ibfk_1` FOREIGN KEY (`kat_buku`) REFERENCES `kategori` (`kat_buku`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `buku_ibfk_2` FOREIGN KEY (`kd_penerbit`) REFERENCES `penerbit` (`kd_penerbit`) ON DELETE CASCADE ON UPDATE CASCADE;
-- Constraints for table ``
--
ALTER TABLE `pembelian`
  ADD CONSTRAINT `pembelian_ibfk_1` FOREIGN KEY (`id_konsumen`) REFERENCES `konsumen` (`id_konsumen`) ON DELETE CASCADE ON UPDATE CASCADE,
-- Constraints for table ``
```

```
--

ALTER TABLE `detail pembelian`

  ADD CONSTRAINT `pembelian_ibfk_1` FOREIGN KEY (`no_pembelian`) REFERENCES `pembelian` (`no_pembelian`) ON DELETE CASCADE ON UPDATE CASCADE,

ADD CONSTRAINT `pembelian_ibfk_2` FOREIGN KEY (`kd_buku`) REFERENCES `buku` (`kd_buku`) ON DELETE CASCADE ON UPDATE CASCADE,

-- Constraints for table ``

--

ALTER TABLE `pembayaran`

  ADD CONSTRAINT `pembayaran_ibfk_1` FOREIGN KEY (`id_konsumen`) REFERENCES `konsumen` (`id_konsumen`) ON DELETE CASCADE ON UPDATE CASCADE,

ADD CONSTRAINT `pembayaran_ibfk_2` FOREIGN KEY (`no_pembelian`) REFERENCES `pembelian` (`no_pembelian`) ON DELETE CASCADE ON UPDATE CASCADE,

ADD CONSTRAINT `pembayaran_ibfk_3` FOREIGN KEY (`no_verifikasi `) REFERENCES `pembelian` (`no_pembelian`) ON DELETE CASCADE ON UPDATE CASCADE,

-- Constraints for table ``

--

ALTER TABLE `periode`

ADD CONSTRAINT `periode_ibfk_1` FOREIGN KEY (`no_pembayaran`) REFERENCES `pembayaran` (`no_pembayaran`) ON DELETE CASCADE ON UPDATE CASCADE

ADD CONSTRAINT `periode_ibfk_2` FOREIGN KEY (`nik`) REFERENCES `petugas` (`nik `) ON DELETE CASCADE ON UPDATE CASCADE

-- Constraints for table ``

--

ALTER TABLE `kota`
```

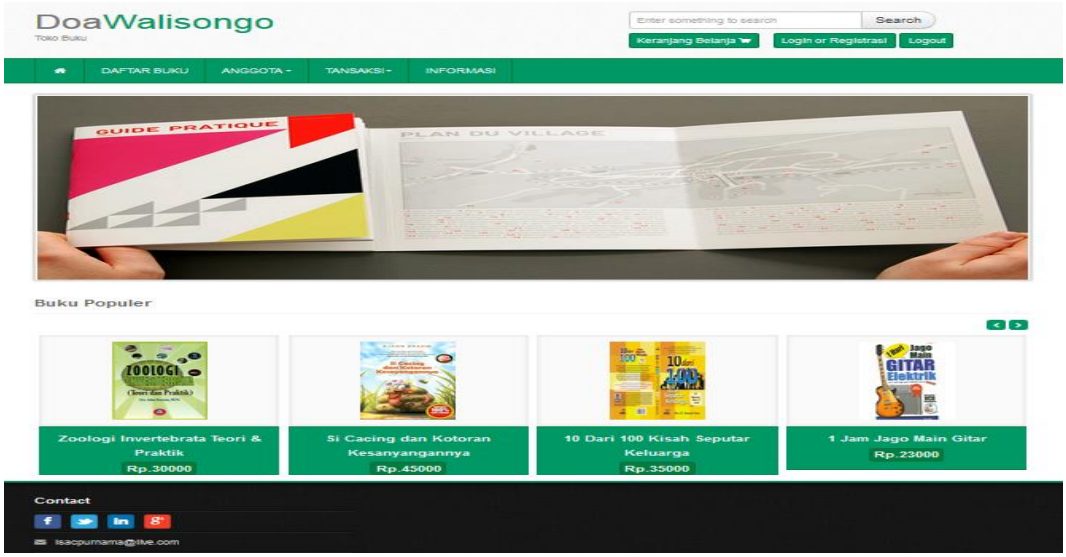
```
 ADD CONSTRAINT `periode_ibfk_1` FOREIGN KEY (`id_provinsi`)
REFERENCES `provinsi` (`id_provinsi`) ON DELETE CASCADE ON
UPDATE CASCADE,
SET FOREIGN_KEY_CHECKS=1;
COMMIT;
```

5.1.5 Implementasi Antar Muka

Antar muka sebuah perangkat lunak adalah hal yang sangat penting, karena merupakan sarana interaksi antara user dengan suatu aplikasi komputer. Setelah melakukan implementasi *Database* maka tahap berikutnya adalah melakukan pengkodean untuk menciptakan antar muka dan memproses berbagai macam data yang ada di dalam *database*. Dalam penelitian ini, sebagaimana telah dipaparkan sebelumnya pengkodean dilakukan dengan menggunakan bahasa pemrograman PHP.

5.1.5.1 Halaman Utama

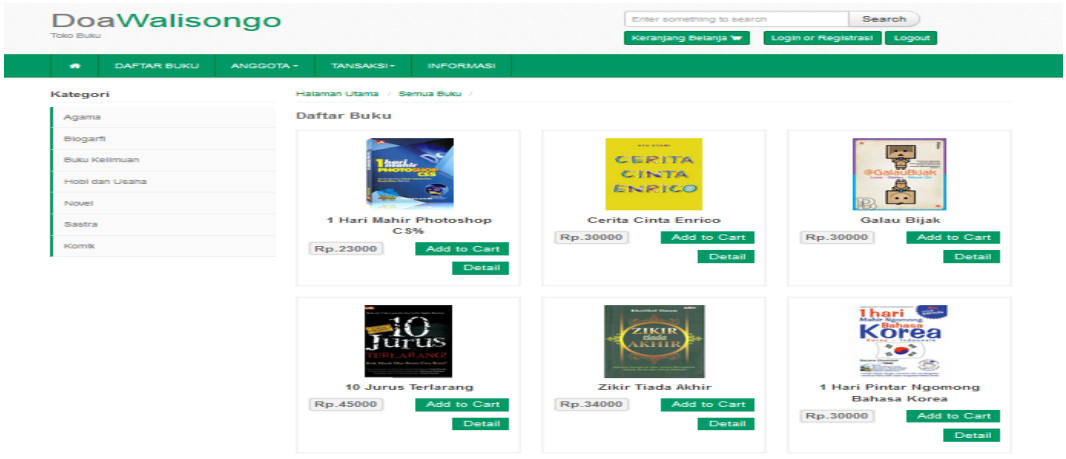
Halaman ini akan muncul ketika *konsumen* pertama kali membuka website. Menu ini merupakan tampilan pertama saat aplikasi ini dijalankan. Halaman ini menampilkan produk buku populer dan juga menampilkan menu-menu untuk mengetahui halaman-halaman informasi yang tersedia seperti daftar buku, anggota ,transaksi, informasi, keranjang belanja, logout dan menu lain yang mengacu pada halaman lain. Terdapat pula hal login user yang hanya dapat dikunjungi oleh member tersebut dengan memasukkan user name dan password.



Gambar 5.1 Halaman Utama

5.1.5.2 Daftar Buku

Pada halaman produk ini menampilkan semua produk-produk berdasarkan kategori yang dipilih pada submenu produk. Dan nantinya akan terhubung ke halaman detail produk.



Gambar 5.2 Daftar Buku

5.1.5.3 Detail Buku

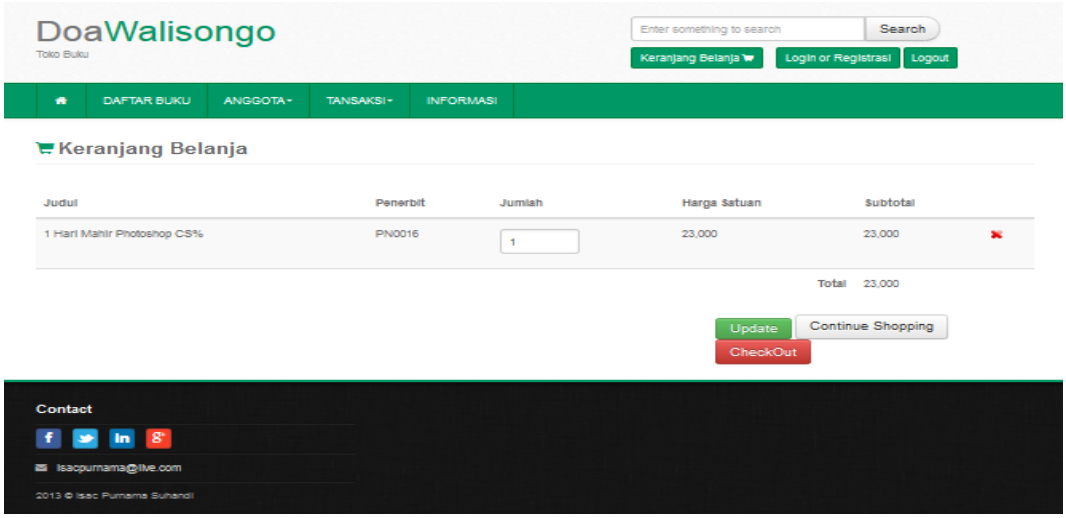
Halaman ini berisikan detail tentang produk yang telah di pilih oleh pengunjung website. Detail yang ditampilkan berupa foto, nama produk, harga dan deskripsi buku. Pada halaman ini terdapat button ‘Add to cart’ untuk memasukkan produk kedalam keranjang belanja yang nantinya akan menghubungkan kehalaman keranjang belanja.



Gambar 5.3 Detail Buku

5.1.5.4 Keranjang Belanja

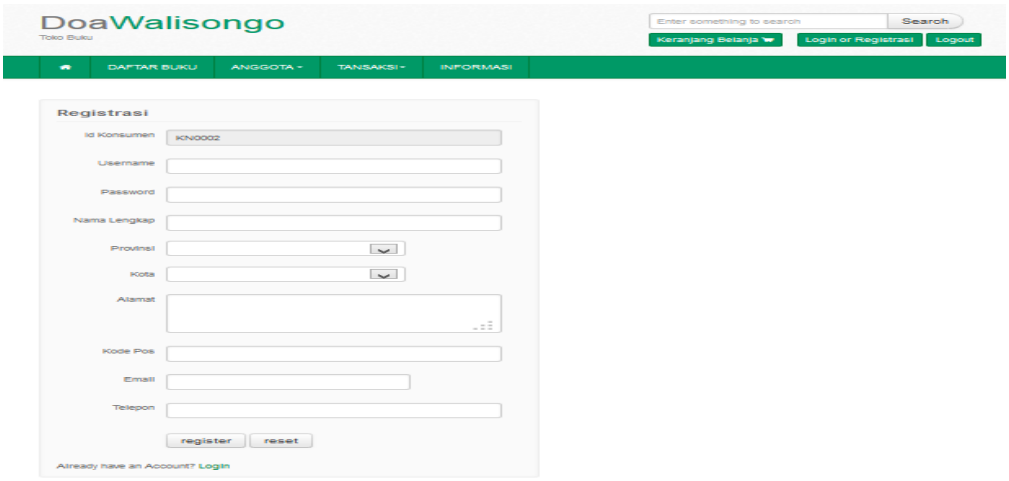
Halaman ini dapat diakses oleh semua pengunjung, namun untuk melakukan *check out* / selesai belanja, harus melakukan registrasi jika belum pernah mendaftar sebagai member. Sedangkan member yang sudah terdaftar proses pembelian dapat langsung *login* dan bisa melakukan proses transaksi pembelian tanpa harus melakukan registrasi ulang.



Gambar 5.4 Keranjang Belanja

5.1.5.5 Registrasi

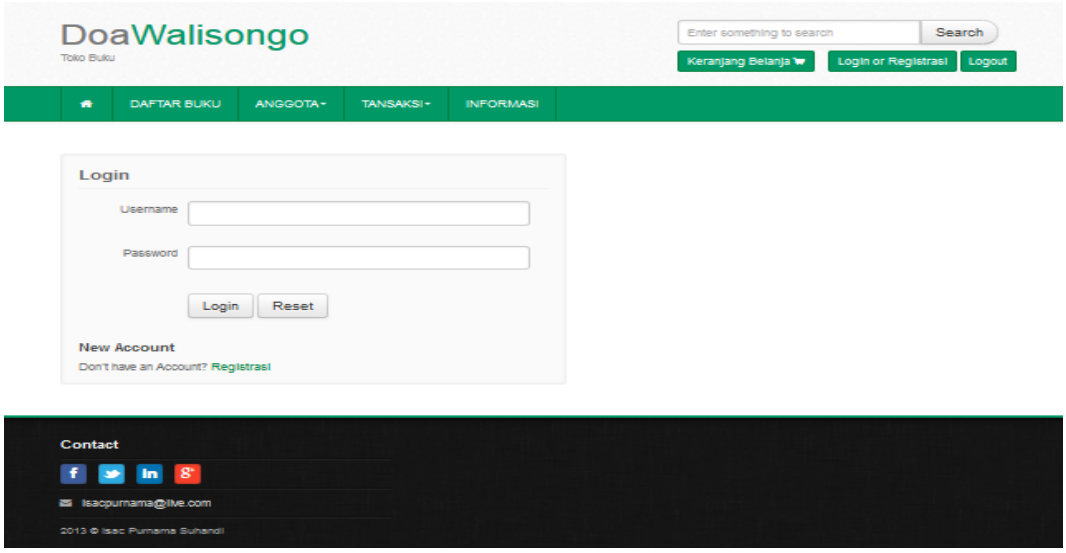
Halaman daftar ini berfungsi untuk *konsumen* yang ingin menjadi member atau ingin membeli produk yang ditawarkan oleh Toko Buku Doa Walisongo. Karena untuk melakukan proses pembelian tanpa harus melakukan pengisian data lagi pengunjung harus melakukan login. Dihalaman ini pengunjung diharuskan mengisi biodata secara lengkap karena nantinya data biodata yang telah di isi akan disimpan, dan *customer* tidak perlu menuliskan alamat nya kembali jika ingin transasi pembelian buku.



Gambar 5.5 Registrasi

5.1.5.6 Login Konsumen

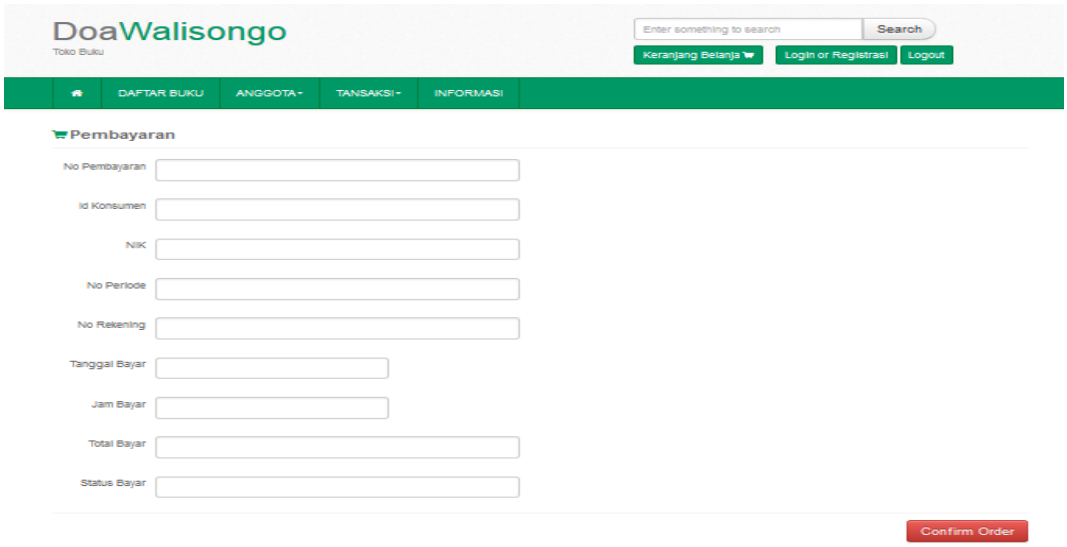
Halaman ini dilakukan setelah melakukan proses registrasi.



Gambar 5.6 Login

5.1.5.7 Pembayaran

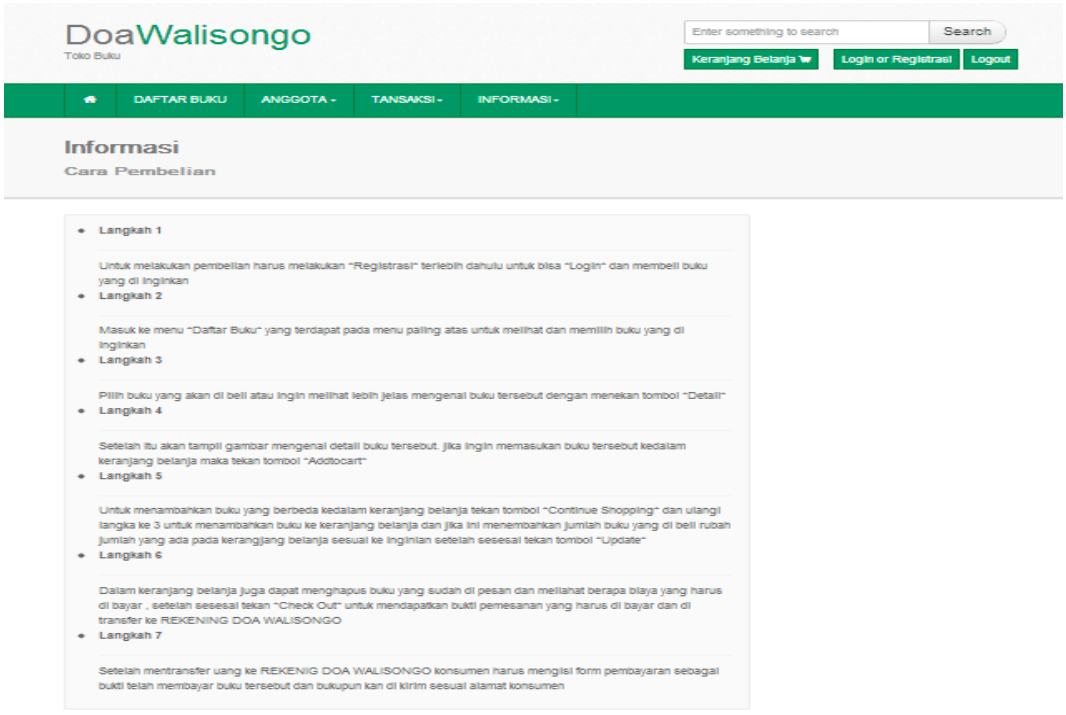
Halaman ini di isi oleh konsumen setelah mentransfer uang sesuai total pembelian buku yang di tawarkan Toko Buku Doa Walisongo.



Gambar 5.7 Pembayaran

5.1.5.8 Cara Belanja

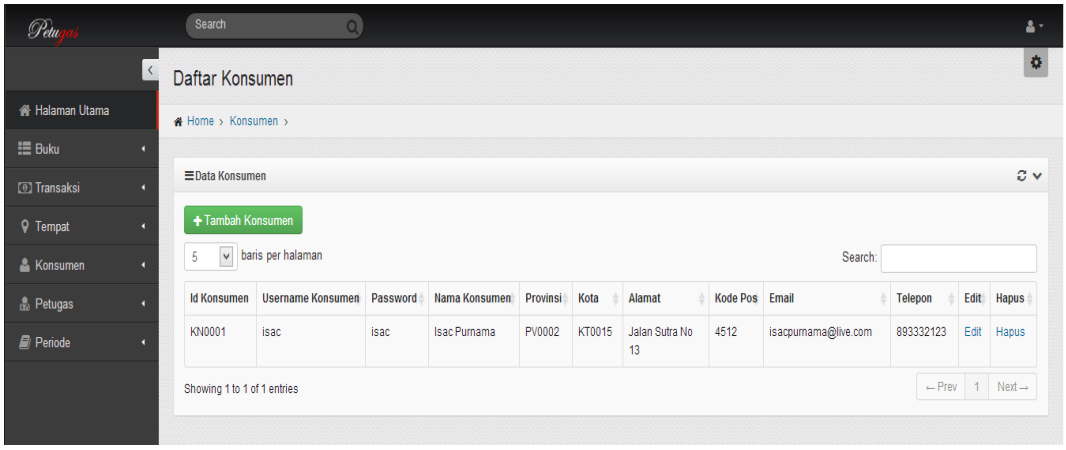
Pada halaman Cara Belanja ini menampilkan tata cara untuk melakukan belanja pada Toko Buku Doa Walisongo.



Gambar 5.8 Cara Belanja

5.1.5.9 Daftar Konsumen

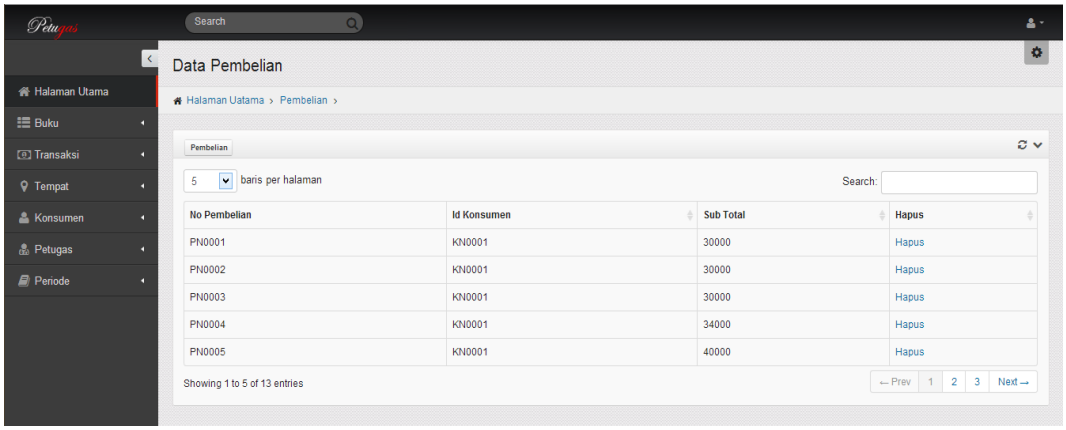
Pada halaman Daftar Konsumen ini menampilkan daftar data konsumen yang telah menjadi anggota di websiteToko Buku Doa Walisongo halaman ini hanya biasa di akses oleh petugas.



Gambar 5.9 Daftar Konsumen

5.1.5.10 Data Pembelian

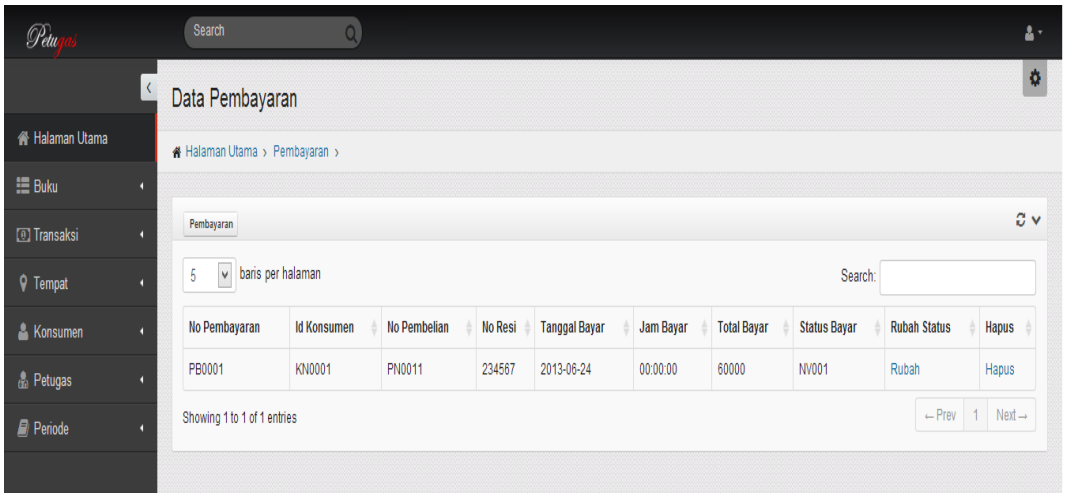
Pada halaman Data Pembelian ini menampilkan data pembelian konsumen yang telah membeli produk di websiteToko Buku Doa Walisongo halaman ini hanya biasa di akses oleh petugas.



Gambar 5.10 Data Pembelian

5.1.5.11 Data Pembayaran

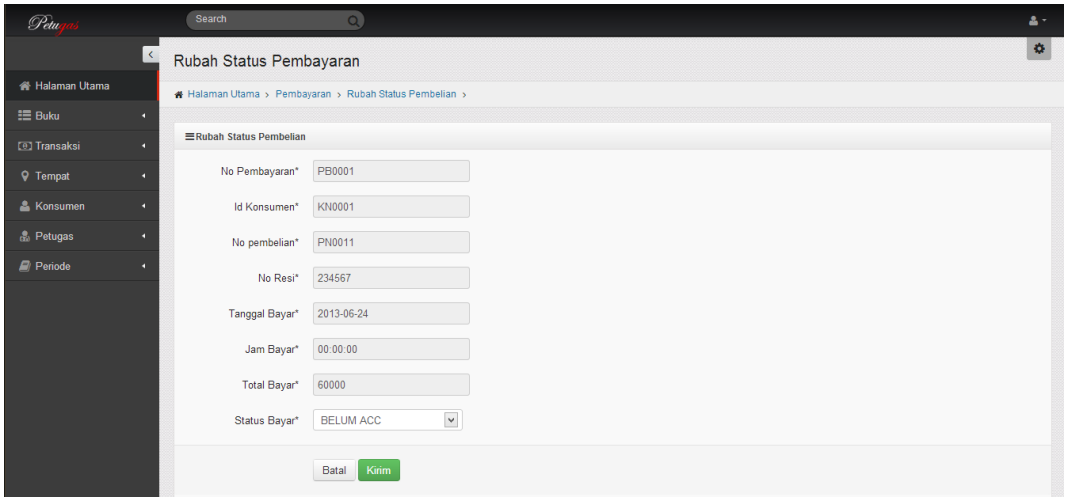
Pada halaman Data Pembayaran ini menampilkan data pembayaran yang telah dibayar oleh konsumen .halaman ini hanya biasa di akses oleh petugas.



Gambar 5.11 Data Pembayaran

5.1.5.12 Rubah Status Pembayaran

Pada halaman Rubah Status Bayar ini menampilkan data pembayaran yang akan dirubah statusnya oleh petugas sesuai data pembayaran.



Gambar 5.12 Data Status Pembayaran

5.1.6 Implementasi Instalasi Program

Implementasi instalasi program merupakan aspek awal untuk menjelaskan bagaimana perancangan aplikasi dapat dijalankan secara utuh didalam sistem komputer. Berikut langkah-langkah yang dilakukan dalam melakukan proses instalasi program agar berjalan pada kondisi *offline*.

1. Pastikan sudah memiliki software Wamp untuk servernya.
2. Lakukan konfigurasi Software Wamp (Wamp-control.exe), dengan mengaktifkan Apache dan MySQL
3. Konfigurasi database yang ada pada PhpMyAdmin, dengan menuju link, http://127.0.0.1/phpmyadmin/
4. Konfigurasi penyimpanan file data *web* secara *offline*, dengan menyimpan file pada folder www.

5.1.7 Penggunaan Program

Untuk menggunakan program dari perancangan penjualan berbasis *e-commerce* ini, dilakukan dengan menggunakan server offline yang disediakan software Wamp, dengan sebelumnya harus mengaktifkan beberapa data, kemudian masuk kedalam internet browser dan ketikan http://localhost/TokoBuku kemudian akan muncul tampilan perancangan *web* yang sudah di buat, dalam keadaan offline sistem.

5.2 Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian ini adalah

untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang baik, yaitu sesuai dengan analisis, perancangan dan pengkodean serta mampu memenuhi kebutuhan pengguna.

5.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan terhadap sistem informasi pelayanan reservasi tiket ini menggunakan data uji berupa input data yang dilakukan oleh pengguna sistem, dalam hal ini adalah petugas dan pemesan. Pada tahap pengujian ini, penulis menggunakan metode pengujian *black box* yang berfokus pada persyaratan fungsional perangkat lunak yang dibuat.Berikut adalah rencana pengujian yang akan dilakukan :

Tabel 5.1 Tabel Rencana Pengujian Tampilan Konsumen

Kasus Uji	Butir Uji	Jenis Pengujian
Registrasi	Melakukan Pengisian data pendaftaran konsumen.	*Blackbox*
Login Konsumen	Melakukan proses *login* konsumen.	*Blackbox*
Addtocart	Memasukkan pemesanan data barang ke keranjang belanja.	*Blackbox*
	Melakukan *update* jumlah pemesanan di keranjang belanja.	
Input data pembayaran	Melakukan penginputan data konfirmasi pembayaran	*Blackbox*

Tabel 5.2 Tabel Rencana Pengujian Tampilan Petugas

Kasus Uji	Butir Uji	Jenis Pengujian
Login petugas	Melakukan proses *login* sebagai *petugas*	*Blackbox*

Daftar Buku	Melakukan *input*, *edit*, serta hapus buku	*Blackbox*
Kategori	Melakukan *input*, *edit*, serta hapus kategori	*Blackbox*
Penerbit	Melakukan *input*, *edit*, serta hapus penerbit	*Blackbox*
Provinsi	Melakukan *input*, *edit*, dan hapus	*Blackbox*
Kota	Melakukan *input*, *edit*, dan hapus	*Blackbox*
Konsumen	Melakukan hapus	*Blackbox*
Petugas	Melakukan *input*, *edit*, dan hapus	*Blackbox*
Periode	Melakukan *input*, dan hapus	*Blackbox*

5.2.2 Kasus dan Hasil Pengujian

Hasil pengujian yang dilakukan terkadang tidak sesuai dengan apa yang diharapkan. Pada subbab ini kita dapat mengetahui bagaimana kelebihan dan kekurangan dari aplikasi yang telah kita buat.

Pengujian Tampilan Konsumen

1) Pengujian Registrasi Konsumen

Tabel 5.3 Pengujian Registrasi *Konsumen*

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi lengkap data pendaftaran konsumen	Berhasil mendaftar dan memunculkan halaman konfirmasi sukses mendaftar	Berhasil dan masuk ke halaman konfirmasi sukses mendaftar	[ ] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			

Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Tidak mengisi lengkap *form* pendaftaran	Registrasi tidak berhasil dan muncul pesan untuk mengisi lengkap *form* pendaftaran	Muncul pesan "Isi terlebih dahulu "	[ ] diterima [ ] ditolak
Mengisi *Form* dengan format data yang tidak *valid*	Muncul validasi data berdasarkan *form* yang diisi	Muncul pesan validasi berdasarkan *form* yang salah diisi	[ ] diterima [ ] ditolak

2) Pengujian Login Konsumen

Tabel 5.4 Pengujian Login *Konsumen*

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi *Username* dan *Password Valid*	Berhasil *login* dan masuk ke menu utama member	*Login* berhasil dan masuk ke halaman menu utama member	[ ] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi Kombinasi *Password* dan *Username* dengan data yang tidak *valid*	Gagal *login* dan muncul pesan "silahkan login lagi, password dan username yang anda masukkan salah"	*Login* gagal dan muncu pesan "silahkan login lagi, password dan username yang anda masukkan salah"	[ ] diterima [ ] ditolak

3) Pengujian Addtocart

Tabel 5.5 Pengujian *Addtocart*

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukkan data pemesanan	Data pemesanan berhasil masuk ke	Data pemesanan berhasil masuk ke	[ ] diterima [ ] ditolak

ke keranjang belanja dengan menekan tombol "addtocart"	keranjang belanja dan muncul halaman keranjang belanja	keranjang belanja dan muncul halaman keranjang belanja.	
Mengupdate jumlah pesanan pada field jumlah di keranjang belanja	Jumlah data dikeranjang belanja berhasil di *update*.	Berhasil mengupdate jumlah pemesanan barang	[ ] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Memasukkan *field* jumlah bukan angka	*Update* jumlah tidak dapat dilakukan dan *field* jumlah kosong	*Field* jumlah di keranjang belanja kosong	[ ] diterima [ ] ditolak

4) Pengujian Pembayaran

Tabel 5.6 Pengujian Pembayaran

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi lengkap data pembayaran dengan format yang benar	Data Pembayaran berhasil dimasukkan dan muncul halaman Konfirmasi Pembayaran	Muncul halaman konfirmasi pembayaran dengan data pembayaran yang di inputkan	[ ] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan salah satu *field*	muncul pesan untuk mengisi lengkap form pendaftaran	Muncul pesan "Isi terlebih dahulu"	[ ] diterima [ ] ditolak

Pengujian Tampilan Petugas

1) Pengujian Login Petugas

Tabel 5.7 Pengujian Login Petugas

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian *username dan Password* dengan data valid	Berhasil Masuk *form* Petugas	Berhasil dan masuk ke halaman Petugas tercantum pada *text box*	[ ] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan salah satu *field*	muncul pesan untuk mengisi lengkap form login	Muncul pesan "tidak boleh kosong"	[ ] diterima [ ] ditolak

2) Pengujian Login Daftar Buku

Tabel 5.8 Pengujian Daftar Buku

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Input buku valid	Berhasil menambahkan data buku	Data buku berhasil masuk ke from buku	[ ] diterima [ ] ditolak
Edit buku	Masuk Ke *form edit*	Data buku berhasil di edit	[ ] diterima [ ] ditolak
Hapus buku	muncul pesan bahwa buku berhasil di hapus	Data buku berhasil di hapus	[ ] diterima [ ] ditolak

3) Pengujian Kategori

Tabel 5.9 Pengujian Kategori

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Input kategori valid	Berhasil menambahkan data kategori	Data kategori berhasil masuk ke from kategori	[ ] diterima [ ] ditolak
Edit kategori	Masuk Ke *form edit*	Data kategori berhasil di edit	[ ] diterima [ ] ditolak
Hapus kategori	muncul pesan bahwa kategori berhasil di hapus	Data kategori berhasil di hapus	[ ] diterima [ ] ditolak

4) Pengujian Penerbit

Tabel 5.10 Pengujian Penerbit

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Input penerbit valid	Berhasil menambahkan data penerbit	Data penerbit berhasil masuk ke from penerbit	[ ] diterima [ ] ditolak
Edit kategori	Masuk Ke *form edit*	Data penerbit berhasil di edit	[ ] diterima [ ] ditolak
Hapus penerbit	muncul pesan bahwa penerbit berhasil di hapus	Data penerbit berhasil di hapus	[ ] diterima [ ] ditolak

5) Pengujian Provinsi

Tabel 5.11 Pengujian Provinsi

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

Input provinsi valid	Berhasil menambahkan data provinsi	Data provinsi berhasil masuk ke from provinsi	[ ] diterima [ ] ditolak
Edit provinsi	Masuk Ke *form edit*	Data provinsi berhasil di edit	[ ] diterima [ ] ditolak
Hapus provinsi	muncul pesan bahwa provinsi berhasil di hapus	Data provinsi berhasil di hapus	[ ] diterima [ ] ditolak

6) Pengujian Kota

Tabel 5.12 Pengujian Kota

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Input kota valid	Berhasil menambahkan data kota	Data kota berhasil masuk ke from kota	[ ] diterima [ ] ditolak
Edit kota	Masuk Ke *form edit*	Data kota berhasil di edit	[ ] diterima [ ] ditolak
Hapus kota	muncul pesan bahwa kota berhasil di hapus	Data kota berhasil di hapus	[ ] diterima [ ] ditolak

7) Pengujian Konsumen

Tabel 5.13 Pengujian Konsumen

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Hapus konsumen	muncul pesan bahwa konsumen berhasil di hapus	Data konsumen berhasil di hapus	[ ] diterima [ ] ditolak

8) Pengujian Petugas

Tabel 5.14 Pengujian Petugas

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Input petugas valid	Berhasil menambahkan data petugas	Data kota berhasil masuk ke from petugas	[ ] diterima [ ] ditolak
Edit petugas	Masuk Ke *form edit*	Data petugas berhasil di edit	[ ] diterima [ ] ditolak
Hapus petugas	muncul pesan bahwa petugas berhasil di hapus	Data petugas berhasil di hapus	[ ] diterima [ ] ditolak

9) Pengujian Periode

Tabel 5.15 Pengujian Periode

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Input data periode valid	Berhasil menambahkan data periode	Data periode berhasil masuk ke from periode	[ ] diterima [ ] ditolak
Hapus data periode	muncul pesan bahwa periode berhasil di hapus	Data periode berhasil di hapus	[ ] diterima [ ] ditolak

5.2.3 Kesimpulan Hasil Pengujian

Melihat hasil pengujian dengan kasus uji sample di atas dapat disimpulkan bahwa secara garis besar perangkat lunak *relative* bebas dari kesalahan sintak dan secara fungsional dapat berjalan dengan baik serta mengeluarkan hasil yang sesuai dengan yang diharapkan