BAB V.

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Tahap implementasi sistem merupakan tahap menterjemahkan perancangan berdasarkan hasil analisis dalam bahasa yang dapat dimengerti oleh mesin serta penerapan perangkat lunak pada keadaan yang sesungguhnya.

5.1.1. Batasan Implementasi (optional)

Batasan implementasi pada Sistem Informasi penjualan di Bunga Art Gallery Bandung ini berdasarkan penelitian penulis di Bunga Art Gallery Bandung yang membutuhkan adanya Sistem Informasi penjualan yang dapat mengelola dan menginformasikan semua data seperti laporan transaksi,keuangan dan lain lain. Dengan demikian penulis merancang sebuah Sistem Informasi Bunga Art Gallery berbasis website dimana penulis dalam membangun Sistem Informasi Bunga Art Gallery Berbasis Website ini masih dalam jaringan lokal karena masih sebatas dalam komputer penulis saja (localhost sebagai lokal *server*) penulis dapat bekerja secara *offline* tanpa harus takut menghadapi masalah biaya, waktu, dan kenyamanan saat perancangan dan pembangunan sistem informasi ini.

5.1.2. Implementasi Perangkat Lunak

Perangkat lunak yang dipasang pada sistem komputer yang digunakan untuk mengembangkan sistem penerapan *e-commerce* untuk sistem penjualan ini adalah sebagai berikut :

1. Sistem Operasi Windows XP/ Vista.

2. MySQL untuk penyimpanan *database.*

3. PhpMyAdmin untuk *interface database.*

4. Macromedia dreamweaver 8 untuk penulisan kode dan perancangan tampilan program.

5.1.3. Implementasi Perangkat Keras

Perangkat keras yang digunakan untuk mengembangkan Sistem Penerapan *E-commerce* untuk Sistem Penjualan adalah sebagai berikut :

1. *Personal Computer* dan *Processor* minimal pentium 4.

2. *RAM* minimal 512 Mb

3. *Harddisk* minimal 80 Gb

5.1.4. Implementasi Basis Data (Sintaks SQL)

Pada tahap awal pembangunan sebuah aplikasi, dibutuhkan media penyimpanan data (*database*), implementasi database untuk Sistem Penerapan *E-commerce* untuk Sistem Penjualan adalah sebagai berikut :

1. Tabel Admin

```
CREATE TABLE IF NOT EXISTS `admin` (
`id_admin` int(11) NOT NULL AUTO_INCREMENT,
`nama_admin` varchar(50) NOT NULL,
`alamat_admin` text NOT NULL,
`email_admin` varchar(50) NOT NULL,
```

```
`telp_admin` varchar(20) NOT NULL,
`password_admin` varchar(32) NOT NULL,
`verification_admin` varchar(32) NOT NULL,
PRIMARY KEY (`id_admin`),
UNIQUE KEY `email_admin` (`email_admin`)
FOREIGN KEY `id_pembelian` REFERENCES pembelian (`id_pembelian`)
    ) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

2. Tabel Member

```
CREATE TABLE IF NOT EXISTS `member` (
`id_member` int(11) NOT NULL AUTO_INCREMENT,
`id_kota` int(11) NOT NULL,
`nama_member` varchar(50) NOT NULL,
`alamat_member` text NOT NULL,
`ttl_member` date NOT NULL,
`telp_member` varchar(20) NOT NULL,
`kodepos_member` varchar(6) NOT NULL,
`email_member` varchar(50) NOT NULL,
`password_member` varchar(32) NOT NULL,
`verificationcode_member` varchar(32) NOT NULL,
`status_member` enum('0','1') NOT NULL,
PRIMARY KEY (`id_member`),
UNIQUE KEY `email_member` (`email_member`)
```

FOREIGN KEY `id_kota` REFERENCES kota (`id_kota`)) ENGINE=MyISAM DEFAULT CHARSET=latin1;

3. Tabel Provinsi

```
CREATE TABLE IF NOT EXISTS `provinsi` (
`id_provinsi` int(10) NOT NUL AUTO_INCREMENT,
`nama_provinsi` varchar(30) DEFAULT NULL,
    KEY `id_provinsi` (`id_provinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4. Tabel Ongkos Kirim

```
CREATE TABLE IF NOT EXISTS `ongkoskirim` (
`id_ongkoskirim` int(11) NOT NULL AUTO_INCREMENT,
`id_kota` int(11) NOT NULL,
`id_jenispengiriman` int(11) NOT NULL,
`harga_ongkoskirim` int(11) NOT NULL,
PRIMARY KEY (`id_ongkoskirim`)'
FOREIGN KEY `id_kota` REFERENCES kota (`id_kota`)
FOREIGN KEY 2 `id_jenispengiriman` REFERENCES (`id_jenispengiriman`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

5. Tabel Jenis Pengiriman

```
CREATE TABLE IF NOT EXISTS `jenispengiriman` (
`id_jenispengiriman` int(11) NOT NULL AUTO_INCREMENT,
`nama_jenispengiriman` varchar(50) NOT NULL,
`deskripsi_jenispengiriman` text NOT NULL,
```

```
PRIMARY KEY (`id_jenispengiriman`)
    ) ENGINE=MyISAM DEFAULT CHARSET=latin1
```

6. Tabel Kota

```
CREATE TABLE IF NOT EXISTS `kota` (
`id_kota` int(10) NOT NULL AUTO_INCREMENT,
`id_provinsi` int(10) DEFAULT NULL,
`nama_kota` varchar(50) DEFAULT NULL,
`kabkota` varchar(20) DEFAULT NULL,
UNIQUE KEY `kota#PX` (`id_kota`),
FOREIGN KEY`id_provinsi` REFERENCES kota (`id_provinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

7. Tabel Warna

```
CREATE TABLE IF NOT EXISTS `warna` (
`id_warna` int(11) NOT NULL AUTO_INCREMENT,
`nama_warna` varchar(50) NOT NULL,
`format_warna` varchar(10) DEFAULT NULL,
PRIMARY KEY (`id_warna`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

8. Tabel Ukuran

```
CREATE TABLE IF NOT EXISTS `ukuran` (
`id_ukuran` int(11) NOT NULL AUTO_INCREMENT,
`nama_ukuran` varchar(10) NOT NULL,
`deskripsi_ukuran` text NOT NULL,
```

```
PRIMARY KEY (`id_ukuran`)
    ) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

9. Tabel Kategori

```
CREATE TABLE IF NOT EXISTS `kategori` (
`id_kategori` int(11) NOT NULL AUTO_INCREMENT,
`nama_kategori` varchar(50) NOT NULL,
`deskripsi_kategori` text NOT NULL,
PRIMARY KEY (`id_kategori`),
UNIQUE KEY `nama_kategori` (`nama_kategori`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

10. Tabel Barang

```
CREATE TABLE IF NOT EXISTS barang (
`id_barang` int(4) unsigned zerofill NOT NULL AUTO_INCREMENT,
`id_kategori` int(11) NOT NULL,
`nama_barang` varchar(50) NOT NULL,
`harga_barang` int(11) NOT NULL,
`deskripsi_barang`text NOT NULL,
`diskon_barang` int(11) NOT NULL,
`rating_barang` float NOT NULL,
`voterrating_barang` int(11) NOT NULL,
    `viewcounter_barang` int(11) NOT NULL,
gambar_barang varchar(200) NOT NULL,
PRIMARY KEY (`id_barang`)
```

FOREIGN KEY (`id_kategori`) REFERENCES kategori (`id_kategori`))

ENGINE=MyISAM DEFAULT CHARSET=latin1

11. Tabel Detail Barang

CREATE TABLE IF NOT EXISTS `detailbarang` (

`id_detailbarang` int(11) NOT NULL AUTO_INCREMENT,

`id_barang` int(11) NOT NULL,

`id_warna` int(11) DEFAULT NULL,

`id_ukuran` int(11) NOT NULL,

`tanggal_detailbarang` datetime NOT NULL,

`stok_detailbarang` int(11) NOT NULL,

`berat_detailbarang` float NOT NULL,

`promo_detailbarang` enum('0','1') NOT NULL,

PRIMARY KEY (`id_detailbarang`)

KEY `fk_1` (`id_warna`)

KEY`fk_ 2` (`id_barang`)

KEY`fk_3 ` (`id_ukuran`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

12. Tabel Konfirmasi Pembayaran

CREATE TABLE IF NOT EXISTS `konfirmasipembayaran` (

`id_konfirmasipembayaran` int(11) NOT NULL AUTO_INCREMENT,

`id_pembelian` int(11) NOT NULL,

`nama_bank_pengirim` varchar(25) NOT NULL,

`no_rekening_pengirim` varchar(25) NOT NULL,

`total_bayar` int(11) NOT NULL,

`tanggal_transfer` datetime NOT NULL,

PRIMARY KEY (`id_konfirmasipembayaran`)

CONSTRAINT 'fk_id_pembelian'

FOREIGN KEY 'fk_id_pembelian'('id_pembelian')

REFERENCES 'pembelian'('id_pembelian')

ON DELETE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

13. Tabel Detail Pembelian

CREATE TABLE IF NOT EXISTS detailpembelian (

`id_detailpembelian` int(11) NOT NULL AUTO_INCREMENT,

`id_pembelian` int(11) NOT NULL,

`tanggal_detailpembelian` datetime NOT NULL,

`jenis_pembayaran` enum('paypal','transfer bank') DEFAULT NULL,

status_pengiriman enum('dipesan','dikonfirmasi','dibayar','dikirm','diterima') NOT NULL,

`keterangan_detailpembelian` text NOT NULL,

`nama_pemesan` varchar(50) NOT NULL,

`alamat_pemesan` text NOT NULL,

`email_pemesan` varchar(50) NOT NULL,

`no_telp_pemesan` varchar(20) NOT NULL,

`kodepos_pemesan` varchar(6) NOT NULL,

`nama_bank_pemesan` varchar(50) NOT NULL,

`no_rekening_pemesan` varchar(20) NOT NULL,

`no_resi_pemesan `varchar(32) NOT NULL,

`id_rekening` int(11) NOT NULL,

`id_ongkir` int(11) NOT NULL,

`status_pemesanan` enum('ok','cancel') NOT NULL,

`session_id` varchar(32) NOT NULL,

PRIMARY KEY (id_detailpembelian)

FOREIGN KEY `id_pembelian` REFERENCES pembelian (`id_pembelian`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1

14. Tabel Pembelian

CREATE TABLE IF NOT EXISTS `pembelian` (

`id_pembelian` int(11) NOT NULL AUTO_INCREMENT,

`id_member` int(11) DEFAULT NULL,

`id_detailbarang` int(11) NOT NULL,

`harga_temp` int(11) NOT NULL,

`stok_temp` int(11) NOT NULL,

`berat_temp` float NOT NULL,

`session_id` varchar(32) NOT NULL,

PRIMARY KEY (`id_pembelian`)

KEY `fk_1` (`id_detailpembelian`)

KEY `fk_2` (‘id_admin`)

KEY `fk_3` (`id_detailbarang`)

KEY `fk_4` (`id_member`)

KEY `fk_5` (`id_ongkoskirim`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

15. Tabel Kurs

CREATE TABLE IF NOT EXISTS `kurs` (

`id_kurs` int(11) NOT NULL AUTO_INCREMENT,

`kode_kurs` varchar(5) NOT NULL,

`symbol_kurs` varchar(5) NOT NULL,

`harga_kurs` int(11) NOT NULL,

`deskripsi_kurs` int(11) NOT NULL,

PRIMARY KEY (`id_kurs`),

UNIQUE KEY `nama_currency` (`kode_kurs`,`symbol_kurs`)

FOREIGN KEY `id_pembelian` REFERENCES pembelian (`id_pembelian`)

KEY ‘fk_1` (‘id_admin`)

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

5.1.5. Implementasi Antar Muka

Implementasi antar muka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna. Adapun untuk tampilan implementasi akan dilampirkan pada halaman lampiran.

Tabel 5.1 Implementasi antarmuka admin

Menu	Deskripsi	Nama File
Masuk Admin	Login sebagai admin	login.php
Lupa Kata Sandi	Digunakan jika admin lupa terhadapkata sandinya.	lupapassword.php
Edit Akun: - Ubah Biodata Ganti Kata Sandi	Digunakan jika administrator ingin merubah biodata Digunakan jika administrator ingin merubah *password*	edit.account.php

Pengolahan Data Master:		
- Pengolahan Data Kategori	Digunakan untuk melihat, menambah, merubah dan menghapus data kategori.	kategori.php
-Pengolahan Data Barang	Digunakan untuk melihat, menambah, merubah dan menghapus data barang.	barang.php
-Pengolahan Data Ukuran	Digunakan untuk melihat, menambah, merubah dan menghapus data ukuran.	ukuran.php
- Pengolahan Data Warna	Digunakan untuk melihat, menambah, merubah dan menghapus data warna.	warna .php
- Pengolahan Data Barang Detail	Digunakan untuk melihat, menambah, merubah dan menghapus data barang detail.	detailbarang.php
- Pengolahan Data Provinsi	Digunakan untuk melihat, menambah, merubah dan menghapus data provinsi.	provinsi.php
- Pengolahan Data Kota	Digunakan untuk melihat, menambah, merubah dan menghapus data kota.	kota.php
- Pengolahan Data Ongkos Kirim	Digunakan untuk melihat, menambah, merubah dan menghapus data ongkos kirim.	ongkir.php
- Pengolahan Data Member	Digunakan untuk melihat, dan menghapus data member.	member.php
- Pengolahan Data	Digunakan untuk melihat,	jenispengiriman.php

Jenis Pengiriman	dan menghapus data jenis pengiriman	
- Pengolahan Data Rekening	Digunakan untuk melihat, dan menghapus data rekening	rekening.php
Pengolahan Data Transaksi	Digunakan untuk melihat data transaksi yang masuk dan untuk merubah status	datatransaksi.php
Pengolahan Laporan: - Laporan penjualan harian - Laporan penjualan mingguan - Laporan penjualan bulanan - Laporan penjualan tahunan	 Digunakan untuk melihat laporan penjualan perhari Digunakan untuk melihat laporan penjualan perminggu Digunakan untuk melihat laporan penjualan perbulan Digunakan untuk melihat laporan penjualan pertahun	 laporanharian.php laporanmingguan.php laporanbulanan.php laporantahunan.php
Keluar	Digunakan admin untuk keluar dari menu admin	logout.php



Gambar 5.1 Tampilan admin

Tabel 5. 2 Implementasi antarmuka pengunjung dan member

Menu	Deskripsi	Nama File
Registrasi *member*	Digunakan untuk melakukan pendaftaran *member* baru.	register.php
Masuk *member*	Digunakan sebagai halaman login *member*	left.php
Lupa Kata Sandi	Digunakan jika *member* lupa dengan kata sandi	lupapassword.php
Ubah Akun: - Ubah Biodata - Ganti kata sandi	Digunakan untuk mengubah profil *member*. Digunakan jika *member* ingin merubah kata sandi	edit.account.php edit.account.php
Kategori	Digunakan untuk melihat barang perkategori	kategori.php
Produk : produk terbaru	Digunakan untuk melihat seluruh barang terbaru tiap	barang.php

	kategori	
Katalog	Digunakan untuk melihat seluruh barang tiap kategori	barang.php
Produk terlaris	Digunakan untuk melihat seluruh barang terlaris	barang.php
Produk paling banyak dilihat	Digunakan untuk melihat seluruh barang paling banyak dilihat	barang.php
Bantuan :		
Cara pembelian	Digunakan untuk melihat Cara melakukan pembelian	carapembelian.php
Data Kontak	Digunakan untuk melihat data kontak	kontak.php
Cara pembayaran	Digunakan untuk melihat cara pembayaran	carapembayaran.php
Konfirmasi Pembayaran	Digunakan untuk melakukan konfirmasi pembayaran	konfirmasi.php
Keranjang belanja	Digunakan untuk melihat, merubah, menghapus detail transaksi yang akan dilakukan.	cart.php
Keluar	Digunakan *member* untuk keluar dari akunnya.	logout.php



‘**Gambar 5.2** Tampilan Pengunjung

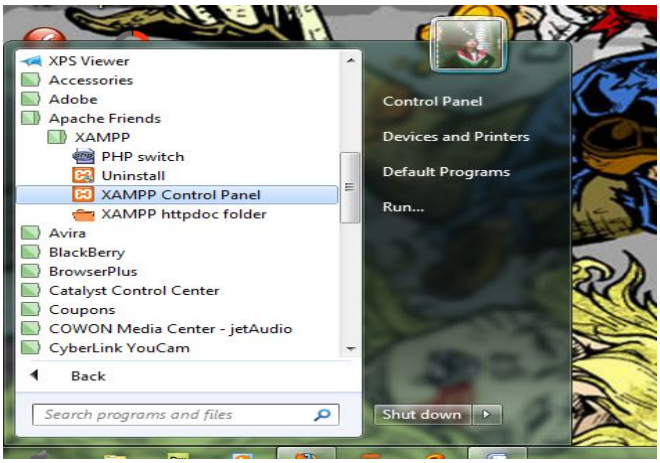


Gambar 5.3 Tampilan Member

5.1.6. Implementasi Instalasi Program

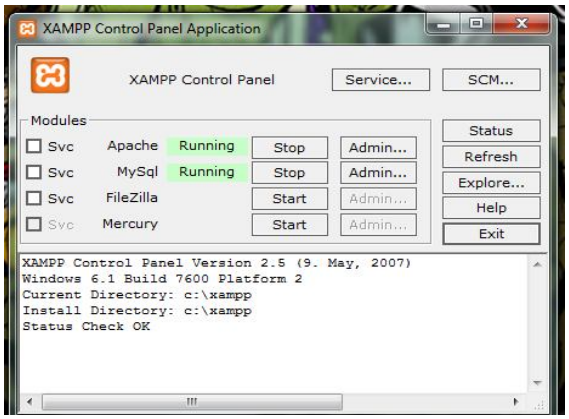
Berikut merupakan implementasi instalasi program untuk menggunakan program e-commerce di Bunga Art Gallery, Langkah-langkahnya adalah sebagai berikut :

1. Klik menu start kemudian pilih Apache Friend,XAMPP,XAMPP control panel.



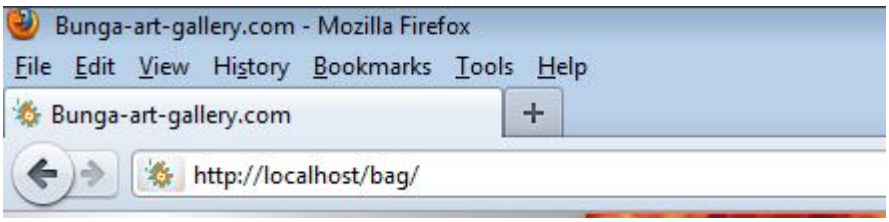
Gambar 5.4 Mengaktifkan XAMPP Control Panel Application

2. Aktifkan XAMPP Control Panel Aplication seperti gambar dibawah ini



Gambar 5.5 XAMPP Control Panel Application

3. Buka *Internet Explorer* atau *browser* lain yang sudah tersambung xampp, lalu ketikan alamat http://localhost/bag/



Gambar 5.6 Tampilan *URL*

5.1.7. Penggunaan Program

Berikut ini merupakan cara penggunaan program agar berjalan dan berfungsi sebagai mana mestinya dengan penjelasan fungsi dapat setiap bagian-bagian. Adapun cara pengoperasianya sebagai berikut :

1. **Tampilan Halama Pertama Saat masuk Bunga Art Gallery Bandung**



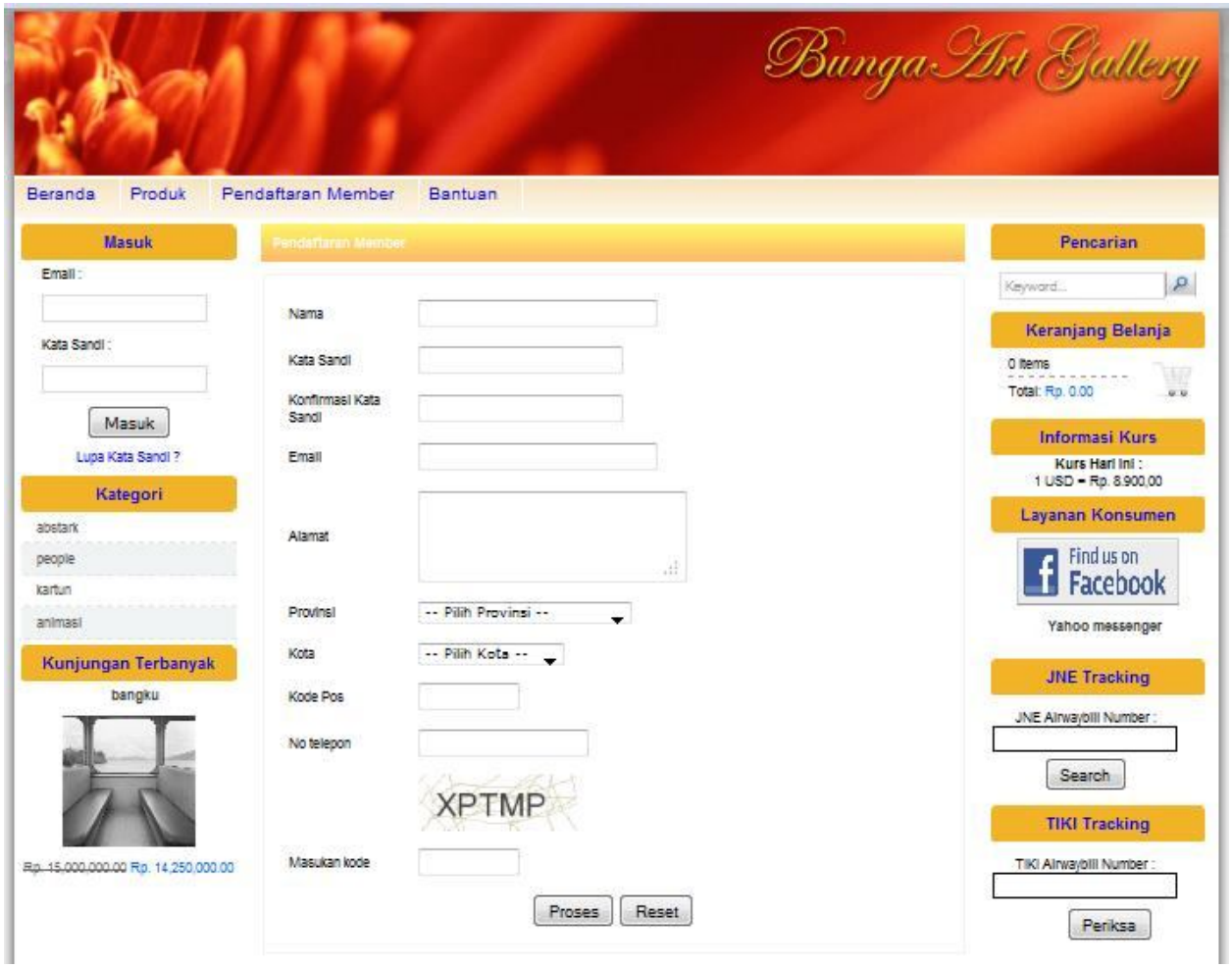
Keterangan:

1.Pengunjung dapat melihat lukisan yang tersedia di Kategori

2.Pengunjung dapat mendaftar jadi member

3. Pengunjung dapat melihat info pembelia, pembayaran, informasi Bunga Art Gallery

2. Tampilan Daftar Member Bunga Art Gallery



Keterangan:

1. Semua field di menu pendaftaran wajib di isi
2. Email harus aktif agar dapat vertification
3. Setelah itu cek email untuk buka pesan dari Bunga Art Gallery
4. Klik link yang di berikan pada email untuk melakukan aktivasi untuk jadi member.

3. Tampilan Member Bunga Art Gallery



Keterangan:

1.Member dapat mengatur akun untuk melakukan perubahan data.

2.Member dapat meliat histori pembelian yang pernah di lakukan.

3.Member melakukan konfirmasi Pembayaran untuk membayar barang yang di pesan.

4. Tampilan Belanja Bunga Art Gallery



Keterangan :

1. Member telah memilih lukisan yang akan di beli
2. Pilih warna untuk frame yang tersedia
3. Pilih ukuran yang tersedia (ukuran berdasarkan cm x cm)
4. Setelah selesai masukan ke keranjang belanja

5. **Tampilan informasi Pembayaran Bunga Art Gallery**

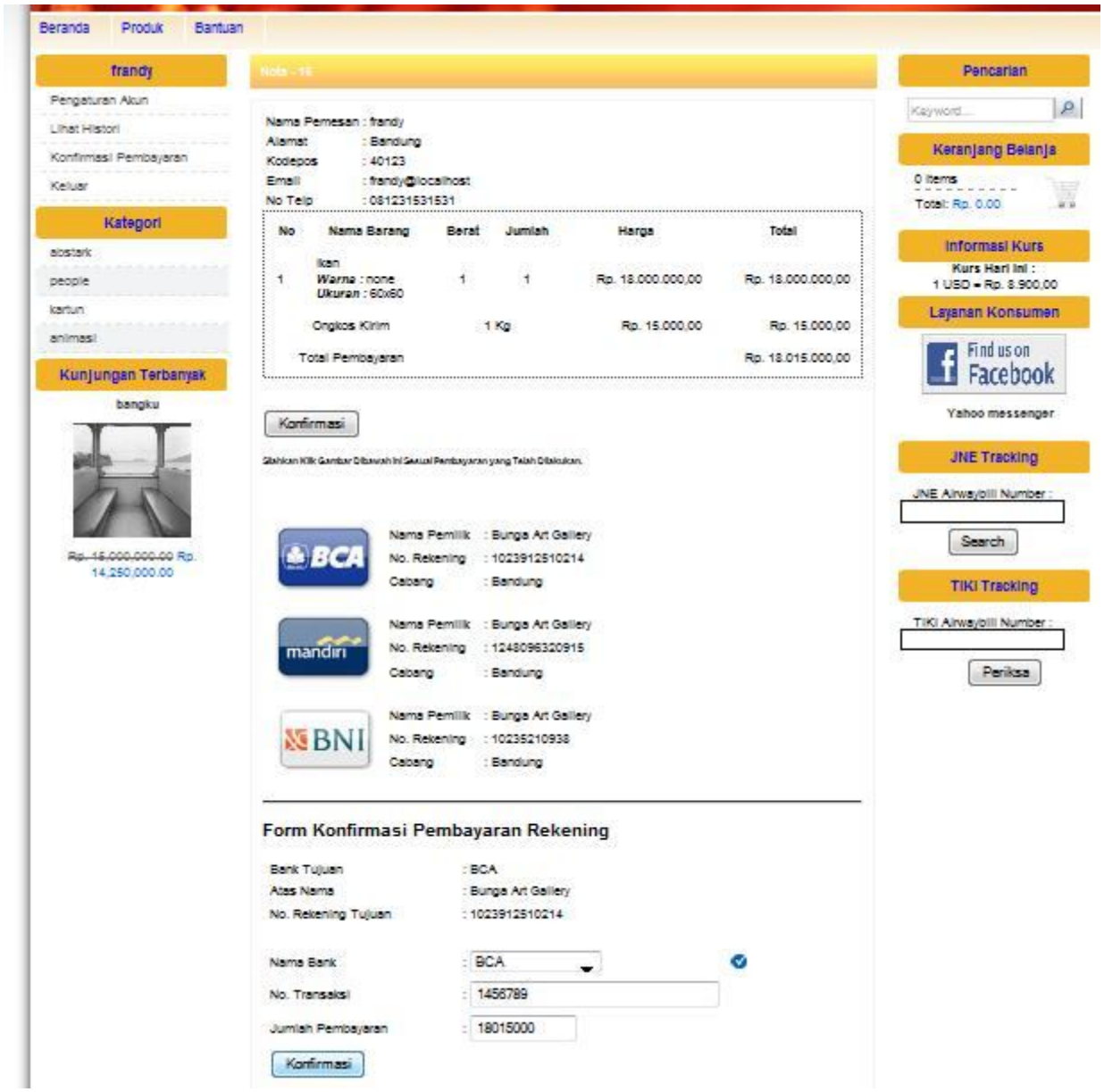


Keterangan:

1.Member wajib mengisi forwarder pengiriman antara Jne ata TIKi

2.Memilih jenis pengiriman berdasarkan waktu

6. Tampilan Konfirmasi Pembayaran



Keterangan:

1. Memasukuan no transaksi yang telah di kirim di email
2. Memilih Bank yang di pilih

5.2. Pengujian

Pengujian yang digunakan untuk menguji sistem yang baru adalah metode pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

5.2.1. Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara *alpha* dan *beta*. Pengujian situs e-commerce pada Bunga Art Gallery berikut menggunakan data uji berdasarkan data dari Bunga Art Gallery.

Tabel 5.3 Rencana Pengujian Aplikasi Admin

Item Uji	Detail Pengujian	Jenis Pengujian
Masuk	Input Login Admin	*Black Box*
	Verifikasi Login Admin	*Black Box*
Pengolahan data kategori	Tambah Kategori	*Black Box*
	Ubah Kategori	*Black Box*
	Hapus Kategori	*Black Box*
	Pencarian	*Black Box*
Pengolahan data barang	Tambah Barang	*Black Box*
	Ubah Barang	*Black Box*
	Hapus Barang	*Black Box*
	Pencarian	*Black Box*
Pengolahan data ukuran	Tambah ukuran	*Black Box*
	Ubah ukuran	*Black Box*
	Hapus ukuran	*Black Box*
	Pencarian	*Black Box*
Pengolahan data warna	Tambah warna	*Black Box*
	Ubah warna	*Black Box*
	Hapus warna	*Black Box*
	Pencarian	*Black Box*
Pengolahan data barang detail	Tambah barang detail	*Black Box*
	Ubah barang detail	*Black Box*
	Hapus barang detail	*Black Box*
	Pencarian	*Black Box*
Pengolahan data provinsi	Tambah barang provinsi	*Black Box*
	Ubah barang provinsi	*Black Box*
	Hapus barang provinsi	*Black Box*
	Pencarian	*Black Box*
Pengolahan data kota	Tambah barang kota	*Black Box*
	Ubah barang kota	*Black Box*

	Hapus barang kota	*Black Box*
	Pencarian	*Black Box*
Pengolahan data jenis pengiriman	Tambah jenis pengiriman	*Black Box*
	Ubah jenis pengiriman	*Black Box*
	Hapus jenis pengiriman	*Black Box*
	Pencarian	*Black Box*
Pengolahan data ongkir	Tambah ongkir	*Black Box*
	Ubah ongkir	*Black Box*
	Hapus ongkir	*Black Box*
	Pencarian	*Black Box*
Pengolahan data rekening	Tambah rekening	*Black Box*
	Ubah rekening	*Black Box*
	Hapus rekening	*Black Box*
	Pencarian	*Black Box*

Tabel 5.4 Skenario pengujian aplikasi member

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Registarsi	Input Registrasi	*Black Box*
	Simpan Data	*Black Box*
	Verifikasi Email	*Black Box*
Masuk	Input Login	*Black Box*
	Verifikasi Login	*Black Box*
Keluar	Menghapus *session user*	*Black Box*
Ubah Biodata	Ubah profile	*Black Box*
Ganti kata sandi	Menampilkan *form* ubah kata sandi	*Black Box*
	Verifikasi kata sandi lama	*Black Box*
	Verifikasi kata sandi baru	*Black Box*
	Verifikasi konfirmasi kata sandi	*Black Box*
Kategori	Menampilkan data kategori	*Black Box*
	Menampilkan data detail kategori	*Black Box*

Produk	Menampilkan data barang	*Black Box*
	Menampilkan data detai l barang	*Black Box*
Menu keranjang belanja	Menampilkan data detail transaksi	*Black Box*
	Mengubah data detail transaksi	*Black Box*
	Menghapus data detail transaksi	*Black Box*
Menu Bantuan	Menampilkan cara pemesanan dan cara pembelian	*Black Box*
Form Konfirmasi Pembayaran	Menampilkan form Konfirmasi Pembayaran	*Black Box*
	Isi form pengiriman	*Black Box*

5.2.2. Kasus dan Hasil Pengujian

Berikut ini adalah kasus untuk menguji perangkat lunak yang sudah dibangun menggunakan metode *BlackBox* berdasarkan Rencana pengujian yang telah dibuat sebelumnya.

1. **Tabel 5.5 Pengujian Registrasi member**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu registrasi *member*	Tampil form registrasi *member*	Dapat menampilkan form registrasi *member*	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh masukan : Nama : Melia Alamat Email : melia.skripsi@gmail.com Provinsi : Jawa Barat Kota : Bandung Kodepos : 40514 Telepon : 085720016479 Kata Sandi : 123456 Konfirmasi kata sandi : 123456	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Klik tombol simpan	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi. Jika data masukan benar muncul pesan "Anda telah berhasil melakukan registrasi"	Tampil pesan "Anda telah berhasil melakukan registrasi" jika data masukan benar dan dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Memasukan data kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

2. **Tabel 5.6 Pengujian Masuk *Member***

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Alamat *email* : melia.skripsi@gmail.com	Mengisikan data alamat *email* pada *field* alamat *email*. Jika salah maka akan muncul pesan "alamat email invalid"	Dapat mengisi *field* alamat *email* dan muncul pesan "invalid alamat email" jika salah dalam pengisian data	[√] Diterima [ ] Ditolak
Password : 123456	Mengisikan data kata sandi pada *field* kata sandi *d*. Jika kata sandi belum diisi maka akan muncul pesan " kata sandi harus diisi dan kata sandi minimal 6 karakter"	Dapat mengisi *field* kata sandi dan muncul pesan " kata sandi minimal 6 karakter diperbolehkan" jika salah dalam pengisian data	[√] Diterima [ ] Ditolak
Klik tombol masuk	Data *email* dan data kata sandi dicari	Tombol login dapat berfungsi	[√] Diterima [ ] Ditolak
Data *email* atau kata sandi tidak terdaftar	Tidak dapat login	Muncul pesan "*email* atau kata sandi salah"	[√] Diterima [ ] Ditolak
Data *email* atau kata sandi kosong	Tidak dapat login	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

3. **Tabel 5.7 Pengujian Keluar**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
KlikKeluar	Dapat keluar dan menghapus session pengguna	*User* keluar dari aplikasi	[√] Diterima [ ] Ditolak

4. **Tabel 5.8 Pengujian Ubah Biodata Member**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Ubah Bioadata	Menampilkan *form* Ubah Biodata	Dapat menampilkan *form* ubah biodata	[√] Diterima [ ] Ditolak
Data biodata yang akan diubah	Tercantum pada text box data yang akan diubah	Dapat mengisi *field* data yang akan diubah dengan benar	[√] Diterima [ ] Ditolak
Klik ubah	Data berhasil diubah di *database*	Tampil pesan "Data sedang diubah"	[√] Diterima [ ] Ditolak
Data masukan ada yang tidak diisi	Menampilkan pesan kesalahan bahwa field tersebut harus terisi.	Dapat menampilkan pesan kesalahan 'apakah anda yakin mengubah data'.	[√] Diterima [ ] Ditolak
Data masukan semuanya tidak diisi	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " apakah anda yakin mengubah data""	[√] Diterima [ ] Ditolak

5. **Tabel 5.9 Pengujian ganti kata sandi *member***

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Ganti kata sandi	Menampilkan *form* ganti kata sandi	Dapat menampilkan *form* ganti kata sandi	[√] Diterima [ ] Ditolak
Data masukan semuanya benar	kata sandi baru berhasil disimpan di *database*	Dapat menyimpan data kata sandi yang baru dengan muncul pesan " Data sedang diubah"	[√] Diterima [ ] Ditolak
Salah satu field belum terisi	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan "*field* ini harus diisi" bila ada field yang belum diisi	[√] Diterima [ ] Ditolak
Data kata sandi lama salah	muncul pesan " kata sandi salah"	Dapat menampilkan pesan kesalahan " kata sandi salah"	[√] Diterima [ ] Ditolak
Data kata sandi baru dan konfirmasi kata sandi baru tidak sama	Muncul pesan " kata sandi baru dan konfirmasi kata sandi harus sama"	Dapat menampilkan pesan " kata sandi baru dan konfirmasi kata sandi harus sama"	[√] Diterima [ ] Ditolak
Data kosong	Muncul pesan kesalahan field harus diisi	Dapat menampilkan pesan kesalahan field harus diisi	√] Diterima [ ] Ditolak

6. **Tabel 5.10 Pengujian menu produk**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu produk	Menampilkan data produk yang ada di *database*	Dapat menampilkan data produk yang ada di *data base*	[√] Diterima [ ] Ditolak
Klik detail pada data produk	Menampilkan detail produk	Dapat menampilkan detail produk	[√] Diterima [ ] Ditolak
Klik Tambah Ke Keranjang	Menuju ke keranjang belanja	Dapat menampilkan keranjang belanja	[√] Diterima [ ] Ditolak

7. **Tabel 5.11 Pengujian Admin Login**

Kasus dan Hasil Uji (data normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
email. kata sandi	Dapat terisi pada textbox(email). Dapat terisi pada textbox(kata sandi).	Dapat mengisi nama email sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Klik Tombol Masuk.	Pengecekan apakah data *valid* atau *invalid*	Tombol masuk berfungsi sesuai dengan yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (data salah)			
Nama email dan kata sandi salah	Tidak dapat login dan menampilkan menu utama admin.	admin tidak dapat masuk dan mendapatkan pesan admin maaf akun tidak ditemukan.	[√] Diterima [ ] Ditolak

8. **Tabel 5.12 Verifikasi Masuk Admin**

Kasus dan Hasil Uji (data normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
email kata sandi	Mengecek email dan kata sandi kosong atau tidak.	Dapat diterima sesuai harapan	[√] Diterima [ ] Ditolak
Klik Tombol Masuk.	Pengecekan dan jika sama dengan data masuk maka admin masuk ke menu utama admin	Tombol login berfungsi sesuai dengan yang diharapkan.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (data salah)			
Data email dan kata sandi salah	Tidak dapat masuk dan menampilkan menu utama admin.	admin tidak dapat masuk dan mendapatkan pesan kesalahan	[√] Diterima [ ] Ditolak

9. **Pengujian Menu Utama di Halaman Admin**

1. **Tabel 5.13 Pengujian ganti kata sandi admin**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik ganti kata sandi	Menampilkan *form* ganti kata sandi	Dapat menampilkan *form* ganti kata sandi	[√] Diterima [ ] Ditolak
Data masukan semuanya benar	Kata sandi baru berhasil disimpan di *database*	Dapat menyimpan data kata sandi yang baru dengan muncul pesan " Data sedang diubah"	[√] Diterima [ ] Ditolak
Salah satu field belum terisi	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan "maaf kata sandi tidak cocok dengan kata sandi lama"	[√] Diterima [ ] Ditolak
Data kata sandi lama salah	muncul pesan " kata sandi salah"	Dapat menampilkan pesan kesalahan " kata sandi salah"	[√] Diterima [ ] Ditolak
Data kata sandi baru dah konfirmasi kata sandi baru tidak sama	Muncul pesan " maaf kata sandi dan konfirmasi tidak cocok "	Dapat menampilkan pesan " maaf kata sandi dan konfirmasi tidak cocok"	[√] Diterima [ ] Ditolak
Data kosong	Muncul pesan kesalahan field harus diisi	Dapat menampilkan pesan kesalahan field harus diisi	√] Diterima [ ] Ditolak

2. **Tabel 5.14 Pengujian Pengolahan Data Provinsi**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik provinsi	Menampilkan data provinsi	Dapat menampilkan data provinsi	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah provinsi	Dapat menampilkan *form* ubah provinsi	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data provinsi kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data provinsi harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di *database*	Tampil pesan "Apakah anda yakin akan mengubah data"	[√] Diterima [ ] Ditolak
Klik batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

3. **Tabel 5.15 Pengujian tambah provinsi**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah provinsi	Menampilkan *form* tambah provinsi	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data provinsi Contoh: Nama Provinsi : Jawa Barat	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data provinsi masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data provinsi harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan "data sedang di tambah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4. **Tabel 5.16 Pengujian Pengolahan Data Kota**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik kota	Menampilkan data kota	Dapat menampilkan data kota	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah kota	Dapat menampilkan *form* ubah kota	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data kota kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data kota harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan "Apakah anda yakin mengubah data"	[√] Diterima [ ] Ditolak
Klik batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

5. **Tabel 5.17 Pengujian tambah kota**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Tambah Kota	Menampilkan *form* tambah kota	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data kota Contoh: Provinsi : Jawa Barat Nama Kota : Subang	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data kota masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "data kota harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan "data sedang di tambah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

6. **Tabel 5.18 Pengujian Pengolahan Data ukuran**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik ukuran	Menampilkan data ukuran	Dapat menampilkan data ukuran	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah ukuran	Dapat menampilkan *form* ubah ukuran	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data ukuran kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data ukuran harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan "Apakah anda yakin akan mengubah data"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

7. **Tabel 5.19 Pengujian tambah ukuran**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah ukuran	Menampilkan *form* tambah ukuran	Dapat menampilkan *form* tambah ukuran	[√] Diterima [ ] Ditolak
Mengisi data ukuran Contoh: Kode : A001 Ukuran : 60x60	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data ukuran masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data ukuran harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan "data sedang ditambah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

8. **Tabel 5.20 Pengujian Pengolahan Data kategori**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik kategori	Menampilkan *form* lihat kategori	Dapat menampilkan *form* lihat kategori	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah kategori	Dapat menampilkan *form* ubah kategori	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data kategori kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data kategori harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan "Apakah anda yakin akan mengubah data"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

9. **Tabel 5.21 Pengujian tambah kategori**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Tambah Kategori	Menampilkan *form* tambah kategori	Dapat menampilkan *form* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data kategori Contoh: Nama kategori: Abstrack	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data kategori masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data kategori harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan "data sedang ditambah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

10. **Tabel 5.22 Pengujian Pengolahan Data barang**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik barang	Menampilkan data barang	Dapat menampilkan data barang	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah barang	Dapat menampilkan *form* ubah barang	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data barang kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data barang harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan "Apakah anda yakin akan mengubah data"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

11. **Tabel 5.23 Pengujian tambah barang**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Tambah Barang	Menampilkan *form* tambah barang	Dapat menampilkan *form* tambah barang	[√] Diterima [ ] Ditolak
Mengisi data barang Contoh: kategori: Abstrack Nama : onepiece Deskripsi Stok : 1 Harga : 12000000 Berat : 1 Kg Gambar :	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data barang masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data barang harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan "data berhasil sedang di tambah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

12. **Tabel 5.24 Pengujian Pengolahan Data member**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik member	Menampilkan data member	Dapat menampilkan data member	[√] Diterima [ ] Ditolak
Klik Hapus pada data yang akan dihapus hapus	Memilih data yang akan dihapus	Data yang akan dihapus telah dipilih	[√] Diterima [ ] Ditolak

13. **Tabel 5.25 Pengujian Pengolahan Data warna**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik kategori	Menampilkan *form* lihat kategori	Dapat menampilkan *form* lihat kategori	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah kategori	Dapat menampilkan *form* ubah kategori	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data kategori kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data kategori harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan "Apakah anda yakin akan mengubah data"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

14. **Tabel 5.26 Pengujian tambah warna**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Tambah Warna	Menampilkan *form* tambah warna	Dapat menampilkan *form* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data kategori Contoh: Nama Warna : Merah	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi iika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data warna masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "data warna harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan "data sedang ditambah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali k emenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

15. **Tabel 5.27 Pengujian Pengolahan Data Jenis Pengiriman**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik jenis pengiriman	Menampilkan *form* lihat jenis pengiriman	Dapat menampilkan *form* lihat jenis pengiriman	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah jenis pengiriman	Dapat menampilkan *form* ubah jenis pengiriman	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data jenis pengiriman kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data jenis pengiriman harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan "Apakah anda yakin akan mengubah data"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

16. **Tabel 5.28 Pengujian Tambah Jenis Pengiriman**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Tambah Jenis Pengiriman	Menampilkan *form* tambah jenis pengiriman	Dapat menampilkan *form* tambah jenis pengiriman	[√] Diterima [ ] Ditolak
Mengisi data jenis pengiriman Contoh: Nama Jenis Pengiriman : SS(*Spesial Service*)	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data jenis pengiriman masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "data jenis pengiriman harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan "data sedang ditambah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

17. **Tabel 5.29 Pengujian Pengolahan Data Ongkos Kirim**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Ongkos Kirim	Menampilkan *form* lihat ongkos kirim	Dapat menampilkan *form* lihat jenis pengiriman	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah ongkos kirim	Dapat menampilkan *form* ubah jenis pengiriman	[√] Diterima [ ] Ditolak
Salah satu *field* kosong	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Data ongkos kirim kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan " data ongkos kirim harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan "Apakah anda yakin akan mengubah data"	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

18. **Tabel 5.30 Pengujian Tambah Ongkos Kirim**

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Tambah Ongkos Kirim	Menampilkan *form* tambah ongkos kirim	Dapat menampilkan *form* tambah ongkos kirim	[√] Diterima [ ] Ditolak
Mengisi data ongkos kirim Contoh: Nama Provinsi: Jawa Barat Nama Kota: Bandung Nama Jenis Pengiriman : SS(*Spesial Service*) Harga Ongkos Kirim: Rp.6000.00	Jika salah satu field tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian.	[√] Diterima [ ] Ditolak
Data jenis pengiriman masih kosong	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan "data jenis pengiriman harus diisi"	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan "data sedang ditambah"	[√] Diterima [ ] Ditolak
Klik batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan pengujian kasus perangkat lunak diatas maka penilis menarik kesimpulan bahwa perangkat lunak ini sudah berjalan secara fungsional dan mengeluarkan informasi sesuai dengan yang diharapkan.