BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

4. 1. Implementasi

Analisis dan perancangan sistem secara terperinci telah dilakukan, maka langkah selanjutnya adalah menuju tahap implementasi. Tujuan implementasi adalah untuk menjelaskan tentang manual modul kepada semua *user* yang akan menggunakan sistem. Sehingga *user* tersebut dapat merespon apa yang ditampilkan di sistem dan memberikan masukan kepada pembuat sistem untuk dilakukan perbaikan agar sistem lebih baik lagi.

4.1.1. Lingkungan Implementasi

Untuk mendukung aplikasi yang akan diterapkan pada lingkungan implementasi, maka dalam hal ini harus memperhatikan perangkat keras dan perangkat lunak yang menunjang dalam Pembangunan Aplikasi *E-commerce* pada Toko Musik Hafidz Pro Ard.

4.1.2. Implementasi Perangkat Keras

Spesifikasi perangkat keras minimal yang dibutuhkan untuk mengimplementasikan aplikasi yang di bangun antara lain terdapat pada tabel 4.1.

Tabel 4.1 Perangkat Keras yang dibutuhkan

No	Perangkat Keras	Spesifikasi
1	Prosessor	AMD X2 555 3.0 GHz
2	Monitor	ukuran 14” Resolusi 1024x768 px
3	VGA	Onboard 64 Mb
4	Memori	256 MB

5	Hard Disk	40 GB (ruang kosong 2 GB)
6	Keyboard	Keyboard
7	Mouse	Mouse
8	Printer	Hitam putih dan warna
9	Jaringan Internet	LAN dan Internet

4.1.3. Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang dibutuhkan untuk mengimplementasikan perangkat lunak antara lain terdapat pada tabel 4.2.

Tabel 4.2Perangkat Lunak yang digunakan

No	Perangkat Keras	Spesifikasi
1	Sistem operasi	Windows Xp
2	Web Browser	Mozilla firefox, opera, safari, internet explorer, google chrome

4.1.4. Implementasi Basis Data

Implementasi data dalam aplikasi *e-commerce* pada Toko musik hafidz pro ard ini menggunakan satu buah DBMS (Data *Base Management System*) dan sebuah GUI *database* untuk memudahkan tugas *administrator*. DBMS yang digunakan untuk menimplentasikan datanya adalah MySQL. Rancangan data beserta nama file query yang dilakukan adalah sebagai berikut:

1. Tabel barang

   Tabel barang berfungsi untuk menampung data barang.

```
CREATE TABLE `barang` (
`id_barang` int(11) NOT NULL AUTO_INCREMENT,
`nama_barang` varchar(50) NOT NULL,
`ukuran` varchar(20) NOT NULL,
`deskripsi` varchar(200) NOT NULL,
```

```
`berat` int(11) NOT NULL,
`jml_penilai` int(11) NOT NULL,
`jml_dilihat` int(11) NOT NULL,
`harga` int(11) NOT NULL,
`diskon` int(11) NOT NULL,
`id_kategori` int(11) NOT NULL,
`ditampilkan` varchar(10) NOT NULL, PRIMARY KEY
(`id_barang`), KEY `id_kategori` (`id_kategori`),
CONSTRAINT `barang_ibfk_1` FOREIGN KEY (`id_kategori`)
REFERENCES `kategori` (`id_kategori`)) ENGINE=InnoDB DEFAULT
CHARSET=latin1;
```

2. Tabel gambar

Tabel gambar digunakan untuk menyimpan data gambar.

```
CREATE TABLE `gambar` (
`id_gambar` int(11) NOT NULL AUTO_INCREMENT,
`id_barang` int(11) NOT NULL,
`nama_gambar` varchar(30) NOT NULL,
`url` varchar(100) DEFAULT NULL,
UNIQUE KEY (`id_gambar`), KEY `barang_fk_1` (`id_barang`),
CONSTRAINT `barang_fk_1` FOREIGN KEY (`id_barang`) REFERENCES
`barang` (`id_barang`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Tabel jasa

Tabel jasa digunakan untuk menyimpan semua data jasa.

```
CREATE TABLE `jasa` (
`id_jasa` int(11) NOT NULL AUTO_INCREMENT,
`nama_jasa` varchar(20) NOT NULL,
```

```
PRIMARY KEY (`id_jasa`)) ENGINE=InnoDB DEFAULT
CHARSET=latin1;
```

4. Tabel kategori

Tabel kategori digunakan untuk menyimpan data kategori.

```
CREATE TABLE `kategori` (
`id_kategori` int(11) NOT NULL AUTO_INCREMENT,
`nama_kategori` varchar(30) NOT NULL,
`lama_pengerjaan` int(11) NOT NULL,
`maks_pengerjaan` int(11) NOT NULL,
`harga_standar` int(11) NOT NULL,
`berat_standar` int(11) NOT NULL,
`status` varchar(10) NOT NULL,
PRIMARY KEY (`id_kategori`)) ENGINE=InnoDB
DEFAULT CHARSET=latin1;
```

5. Tabel keranjang

Tabel keranjang digunakan untuk menyimpan data keranjang.

```
CREATE TABLE `keranjang` (
`id_keranjang` int(11) NOT NULL AUTO_INCREMENT,
`id_member` int(11) NOT NULL,
`id_session` varchar(51) NOT NULL,
`id_barang` int(11) NOT NULL,
`qty` int(11) NOT NULL,
`warna` varchar(7) NOT NULL,
`desain_sendiri` varchar(20) NOT NULL,
`url_desain` varchar(100) NOT NULL,
```

```
UNIQUE KEY (`id_keranjang`),  KEY `id_barang` (`id_barang`),
KEY `id_member` (`id_member`),
CONSTRAINT  `keranjang_ibfk_2`  FOREIGN  KEY  (`id_member`)
REFERENCES    `member`    (`id_member`),          CONSTRAINT
`keranjang_ibfk_1`  FOREIGN  KEY  (`id_barang`)  REFERENCES
`barang` (`id_barang`)) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6. Tabel kota

   Tabel kota digunakan untuk menyimpan data kota.

```
CREATE TABLE `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,
PRIMARY KEY (`id_kota`),  KEY `kota_ibfk_1` (`id_provinsi`),
CONSTRAINT  `kota_ibfk_1`  FOREIGN  KEY  (`id_provinsi`)
REFERENCES `provinsi` (`id_provinsi`) ON DELETE CASCADE ON
UPDATE  CASCADE)  ENGINE=InnoDB  AUTO_INCREMENT=435  DEFAULT
CHARSET=latin1;
```

7. Tabel kurs

   Tabel kurs digunakan untuk menyimpan data kurs.

```
CREATE TABLE `kurs` (
`id_kurs` int(11) NOT NULL AUTO_INCREMENT,
`tgl_kurs` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
`nilai_kurs` int(11) NOT NULL,
PRIMARY  KEY  (`id_kurs`))  ENGINE=InnoDB  AUTO_INCREMENT=27
```

```
DEFAULT CHARSET=latin1;
```

8. Tabel member

Tabel member digunakan untuk menyimpan data member.

```
CREATE TABLE `member` (
`id_member` int(11) NOT NULL AUTO_INCREMENT,
`nama` varchar(50) NOT NULL,
`email` varchar(50) NOT NULL,
`password` varchar(51) NOT NULL,
`tgl_daftar` date NOT NULL,
`alamat` text NOT NULL,
`telepon` varchar(15) NOT NULL,
`id_kota` int(11) NOT NULL,
`kodepos` int(5) NOT NULL,
`status` varchar(11) NOT NULL,
`kode_verivikasi` varchar(51) NOT NULL,
PRIMARY KEY (`id_member`), UNIQUE KEY `email` (`email`))
ENGINE=InnoDB AUTO_INCREMENT=24 DEFAULT CHARSET=latin1;
```

9. Tabel ongkir

Tabel ongkir digunakan untuk menyimpan data ongkir.

```
CREATE TABLE `ongkir` (
`id_ongkir` int(11) NOT NULL AUTO_INCREMENT,
`id_paket` int(11) NOT NULL,
`id_kota` int(11) NOT NULL,
`tarif` int(11) NOT NULL,
PRIMARY KEY (`id_ongkir`), KEY `id_paket`
```

```
(`id_paket`,`id_kota`),       KEY    `id_kota`    (`id_kota`),
CONSTRAINT    `ongkir_ibfk_1`    FOREIGN    KEY    (`id_paket`)
REFERENCES `paket` (`id_paket`),
CONSTRAINT `ongkir_ibfk_2` FOREIGN KEY (`id_kota`) REFERENCES
`kota` (`id_kota`)) ENGINE=InnoDB AUTO_INCREMENT=1697 DEFAULT
CHARSET=latin1;
```

10. Tabel paket

Tabel paket digunakan untuk menyimpan data paket.

```
CREATE TABLE `paket` (
`id_paket` int(11) NOT NULL AUTO_INCREMENT,
`id_jasa` int(11) NOT NULL,
`nama_paket` varchar(50) NOT NULL,
`status` varchar(20) NOT NULL,
PRIMARY   KEY   (`id_paket`),     KEY   `id_jasa`   (`id_jasa`),
CONSTRAINT `paket_ibfk_1` FOREIGN KEY (`id_jasa`) REFERENCES
`jasa`  (`id_jasa`))  ENGINE=InnoDB  AUTO_INCREMENT=10  DEFAULT
CHARSET=latin1;
```

11. Tabel Pembayaran

Tabel pembayaran digunakan untuk menyimpan data pembayaran.

```
CREATE TABLE `pembayaran` (
`id_pembayaran`    int(11)    NOT    NULL    AUTO_INCREMENT,
`id_pemesanan` int(11) NOT NULL,
`tgl_bayar` date NOT NULL,
`tgl_konfirmasi`  timestamp  NOT  NULL  DEFAULT  '0000-00-00
00:00:00',
`metode_pembayaran` varchar(20) NOT NULL,
```

```
`bank_asal` varchar(30) NOT NULL,
`bank_tujuan` varchar(30) NOT NULL,
`jml_bayar` int(11) NOT NULL,
`status` varchar(20) NOT NULL,
`id_petugas` int(11) NOT NULL,
`tgl_dicek` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
`alasan` varchar(200) NOT NULL,
UNIQUE KEY (`id_pembayaran`), KEY `id_pemesanan`
(`id_pemesanan`),
CONSTRAINT `pembayaran_ibfk_1` FOREIGN KEY (`id_pemesanan`)
REFERENCES `pemesanan` (`id_pemesanan`)) ENGINE=InnoDB
AUTO_INCREMENT=64 DEFAULT CHARSET=latin1;
```

12. Tabel Pemesanan

Tabel pemesanan digunakan untuk menyimpan data pemesanan.

```
CREATE TABLE `pemesanan` (
`id_pemesanan` int(11) NOT NULL AUTO_INCREMENT,
`id_member` int(11) NOT NULL,
`tgl_pesan` date NOT NULL,
`tgl_selesai` date NOT NULL,
`status_pemesanan` varchar(30) NOT NULL,
`jenis_pembayaran` varchar(20) NOT NULL,
`nama_pengiriman` varchar(50) NOT NULL,
`alamat_pengiriman` text NOT NULL,
`kodepos_pengiriman` int(11) NOT NULL,
`telepon_pengiriman` int(11) NOT NULL,
`total_berat` float NOT NULL,
`total_bayar` int(11) NOT NULL,
```

```
`id_kota` int(11) NOT NULL,
`id_ongkir` int(11) NOT NULL,
`id_petugas` int(11) NOT NULL,
`noresi` varchar(15) NOT NULL,
`tgl_terima` date NOT NULL,
`nama_penerima` varchar(50) NOT NULL,
`tgl_ubah_terakhir` timestamp NOT NULL DEFAULT '0000-00-00
00:00:00',
`id_kurs` int(11) NOT NULL,
`kode_unik` int(11) NOT NULL,
PRIMARY KEY (`id_pemesanan`), KEY `id_petugas`
(`id_petugas`), KEY `id_ongkir` (`id_ongkir`), KEY
`id_kota` (`id_kota`), KEY `id_member` (`id_member`), KEY
`id_kurs` (`id_kurs`),
CONSTRAINT `pemesanan_ibfk_1` FOREIGN KEY (`id_member`)
REFERENCES `member` (`id_member`),
CONSTRAINT `pemesanan_ibfk_2` FOREIGN KEY (`id_kota`)
REFERENCES `kota` (`id_kota`),
CONSTRAINT `pemesanan_ibfk_3` FOREIGN KEY (`id_ongkir`)
REFERENCES `ongkir` (`id_ongkir`),
CONSTRAINT `pemesanan_ibfk_4` FOREIGN KEY (`id_petugas`)
REFERENCES `petugas` (`id_petugas`),
CONSTRAINT `pemesanan_ibfk_5` FOREIGN KEY (`id_kurs`)
REFERENCES `kurs` (`id_kurs`)) ENGINE=InnoDB
AUTO_INCREMENT=101 DEFAULT CHARSET=latin1;
```

13. Tabel pemesanan_detail

Tabel pemesanan_detail digunakan untuk menyimpan data pemesanan_detail.

```
CREATE TABLE `pemesanan_detail` (
`id_detail` int(11) NOT NULL AUTO_INCREMENT,
`id_pemesanan` int(11) NOT NULL,
`id_barang` int(11) NOT NULL,
`jml` int(11) NOT NULL,
`jml_blm_dikerjakan` int(11) NOT NULL,
`jml_sdg_dikerjakan` int(11) NOT NULL,
`jml_sdh_dikerjakan` int(11) NOT NULL,
`status_pengerjaan` varchar(20) NOT NULL,
`warna` varchar(7) NOT NULL,
`desain_sendiri` varchar(10) NOT NULL,
`url_desain` varchar(100) NOT NULL,
PRIMARY KEY (`id_detail`), KEY `id_pemesanan`
(`id_pemesanan`), KEY `id_barang` (`id_barang`),
CONSTRAINT `pemesanan_detail_ibfk_1` FOREIGN KEY
(`id_pemesanan`) REFERENCES `pemesanan` (`id_pemesanan`),
CONSTRAINT `pemesanan_detail_ibfk_2` FOREIGN KEY
(`id_barang`) REFERENCES `barang` (`id_barang`))
ENGINE=InnoDB AUTO_INCREMENT=115 DEFAULT CHARSET=latin1;
```

14. Tabel petugas

Tabel petugas digunakan untuk menyimpan data petugas.

```
CREATE TABLE `petugas` (
`id_petugas` int(11) NOT NULL AUTO_INCREMENT,
`username` varchar(20) NOT NULL,
`password` varchar(30) NOT NULL,
`nama` varchar(30) NOT NULL,
`level` varchar(20) NOT NULL,
```

```
`id_session` varchar(50) NOT NULL,
`status_blokir` varchar(1) NOT NULL,
PRIMARY KEY (`id_petugas`)) ENGINE=InnoDB AUTO_INCREMENT=10
DEFAULT CHARSET=latin1;
```

15. Tabel provinsi

Tabel provinsi digunakan untuk menyimpan data provinsi.

```
CREATE TABLE `provinsi` (
`id_provinsi` int(10) NOT NULL AUTO_INCREMENT,
`nama_provinsi` varchar(100) DEFAULT NULL,
PRIMARY KEY (`id_provinsi`)) ENGINE=InnoDB AUTO_INCREMENT=35
DEFAULT CHARSET=latin1;
```

16. Tabel retur

Tabel retur adalah tabel yang digunakan untuk menyimpan data retur.

```
CREATE TABLE `retur` (
`id_retur` int(11) NOT NULL AUTO_INCREMENT,
`id_pemesanan` int(11) NOT NULL,
`tgl_retur` date NOT NULL,
`status` varchar(30) NOT NULL,
`noresi_retur` varchar(15) NOT NULL,
`tgl_terima_retur` date NOT NULL,
`nama_penerima_retur` varchar(50) NOT NULL,
`tgl_ubah_retur` timestamp NOT NULL DEFAULT '0000-00-00
00:00:00',
`id_petugas` int(11) NOT NULL,
UNIQUE KEY (`id_retur`), KEY `id_petugas` (`id_petugas`),
KEY `id_pemesanan` (`id_pemesanan`),
```

```
CONSTRAINT `retur_ibfk_2` FOREIGN KEY (`id_petugas`)
REFERENCES `petugas` (`id_petugas`),
CONSTRAINT `retur_ibfk_1` FOREIGN KEY (`id_pemesanan`)
REFERENCES `pemesanan` (`id_pemesanan`)) ENGINE=InnoDB
AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;
```

17. Tabel retur_detail

Tabel retur_detail adalah tabel untuk menyimpan data retur_detail.

```
CREATE TABLE `retur_detail` (
`id_retur_detail` int(11) NOT NULL AUTO_INCREMENT,
`id_retur` int(11) NOT NULL,
`id_detail` int(11) NOT NULL,
`jml_retur` int(11) NOT NULL,
`alasan` varchar(30) NOT NULL,
UNIQUE KEY (`id_retur_detail`), KEY `id_detail`
(`id_detail`), KEY `id_retur` (`id_retur`),
CONSTRAINT `retur_detail_ibfk_1` FOREIGN KEY (`id_detail`)
REFERENCES `pemesanan_detail` (`id_detail`),
CONSTRAINT `retur_detail_ibfk_2` FOREIGN KEY (`id_retur`)
REFERENCES `retur` (`id_retur`))
ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;
```

18. Tabel Kalender

Tabel kalender digunakan untuk menyimpan data kalender.

```
CREATE TABLE `kalender` (
`id_kalender` int(11) NOT NULL AUTO_INCREMENT,
`tgl_libur` date NOT NULL,
`nama_event` varchar(100) NOT NULL,
```

```
`id_petugas` int(11) NOT NULL,
PRIMARY KEY (`id_kalender`),  KEY `id_petugas`
(`id_petugas`),
CONSTRAINT `kalender_ibfk_1` FOREIGN KEY (`id_petugas`)
REFERENCES `petugas` (`id_petugas`)) ENGINE=InnoDB
AUTO_INCREMENT=7 DEFAULT CHARSET=latin1;
```

19. Tabel Pengerjaan

Tabel pengerjaan digunakan untuk menyimpan data pengerjaan.

```
CREATE TABLE `pengerjaan` (
`id_pengerjaan` int(11) NOT NULL AUTO_INCREMENT,
`id_detail` int(11) NOT NULL,
`tgl_mulai` date NOT NULL,
`tgl_selesai` date NOT NULL,
`status_pengerjaan` varchar(30) NOT NULL,
UNIQUE KEY (`id_pengerjaan`),  KEY `id_detail` (`id_detail`),
CONSTRAINT `pengerjaan_ibfk_1` FOREIGN KEY (`id_detail`)
REFERENCES `pemesanan_detail` (`id_detail`)) ENGINE=InnoDB
AUTO_INCREMENT=30 DEFAULT CHARSET=latin1;
```

4.1.5. Implementasi Antarmuka

Implementasi antar muka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk file program. Berikut ini adalah implementasi antarmuka untuk pengguna. Implementasi antarmuka admin pada aplikasi *e-ecommerce* pada toko Toko musik hafidz pro ard dapat dilihat pada tabel 4.3.

Tabel 4.3 Implementasi Antarmuka Admin

Menu	Deskripsi	Nama File
Login Admin/Petugas	Digunakan sebagai halaman login *admin* atau petugas	adminhafidz/index.php
Halaman Utama Admin/Petugas	Halaman utama ketika *admin* atau petugas berhasil login yang memperlihatkan 3 data pemesanan terakhir, 3 data pembayaran terakhir dan 3 data retur terakhir.	adminhafidz/index.php
Pengolahan Data Petugas	Halaman *admin* yang digunakan untuk melihat, mencari, menambah, mengubah dan menghapus data user	adminhafidz/ index.php?c=petugas
Data Master:	Halaman yang digunakan untuk mengolah data master	
- Member	Halaman petugas yang digunakan untuk melihat, mencari dan menonaktifkan member.	adminhafidz/ index.php?c=member
- kategori	Halaman petugas yang digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data kategori.	adminhafidz/ index.php?c=kategori
- Barang	Halaman petugas yang digunakan untuk melihat, menambah, mencari mengubah dan menghapus data barang.	adminhafidz/ index.php?c=barang
- Gambar	Halaman petugas yang digunakan untuk melihat, menambah, mencari mengubah dan menghapus data gambar.	adminhafidz/ index.php?c=gambar
- Kota	Halaman petugas yang digunakan untuk melihat, menambah, mencari mengubah dan menghapus data kota.	adminhafidz/ index.php?c=kota

- Paket	Halaman petugas yang digunakan untuk melihat, menambah, mencari mengubah dan menghapus data paket.	adminhafidz/ index.php?c=paket
- Ongkos kirim	Halaman petugas yang digunakan untuk melihat, menambah, mencari mengubah dan menghapus data ongkos kirim.	adminhafidz/ index.php?c=ongkir
- Jasa	Halaman petugas yang digunakan untuk melihat, menambah, mencari mengubah dan menghapus data jasa.	adminhafidz/ index.php?c=jasa
Data Transaksi:	Halaman yang digunakan untuk mengolah data-data transaksi.	
- Pemesanan	Halaman petugas yang digunakan untuk melihat, menambah, mencari dan mengubah status pemesanan.	adminhafidz/ index.php?c=pemesanan
- Tabel Pengerjaan	Halaman petugas yang digunakan untuk melihat dan mengubah data pemesanan yang sedang dikerjakan.	adminhafidz/ index.php?c=pemesanan &a=pengerjaan
- Pembayaran	Halaman yang digunakan petugas melihat , menambah, mencari, dan mengubah status pembayaran.	adminhafidz/ index.php?c=pembayaran
- Retur	Halaman yang digunakan petugas melihat , menambah, mencari, dan mengubah status pembayaran.	adminhafidz/ index.php?c=retur
Laporan	Halaman yang digunakan petugas untuk melihat buku terjual dan mencetak laporan dalam periode tertentu	adminhafidz/ index.php?c=laporan
Backup	Halaman yang digunakan petugas untuk melakukan *backup* database, gambar dan aplikasi.	adminhafidz/ index.php?c=backup

Logout	Halaman yang digunakan admin atau petugas untuk menghapus session admin atau petugas dan keluar dari aplikasi.	adminhafidz/ index.php?c=logout

Tabel dari implementasi antamuka pengunjung dan member 4.4

Tabel 4.4 Implementasi Antarmuka Pengunjung dan Member

Menu	Deskripsi	Nama File
Beranda	Digunakan sebagai halaman utama pengunjung dan member berisi informasi mengenai produk terbaru, produk terlaris, kategori produk dan produk yang paling banyak dilihat.	index.php
Produk	Halaman yang menampilkan produk / barang yang dapat dipesan, dikelompokan berdasarkan kategori.	index.php?c=kategori
Hafidz Pro	Halaman yang menampilkan profil perusahaan dilengkapi dengan foto – foto kegiatan.	index.php?c=about
Member	Halaman yang menampilkan data akun member.	index.php?c=about
Kontak	Halaman untuk menampilkan data kontak persahaan yang dapat dihubungi oleh pengunjung atau member, disertai dengan alamat lengkap perusahaan.	index.php?c=kontak
Daftar Sekarang	Halaman untuk pengunjung yang ingin memdaftar menjadi member baru.	index.php?c=member&a=daftar
Login	Halaman login untuk member yang sudah terdaftar dan aktif.	index.php?c=member&a=login
Lupa Password	Halaman untuk member yang lupa password.	index.php?c=member&a=lupa
Cari Produk	Halaman untuk mencari produk yang ada pada situs.	index.php?c=barang &a=cari
Detail	Halaman untuk menampilkan data produk secara detail.	index.php?c=barang &a=detail
Pesan Desain Sendiri	Halaman untuk memesan produk dengan desain sendiri.	index.php?c=pemesanan&a=sendiri
Keranjang	Halaman untuk menampilkan data keranjang yang berisi data produk yang	index.php?c=keranjang

	dipesan dan siap di*checkout*.	
Data Akun	Halaman untuk menampilkan data akun member.	index.php?c=member&a=akun
History Pemesanan	Halaman untuk menampilkan data pemesanan yang pernah dilakukan oleh member.	index.php?c=member&a=history
Logout	Halaman untuk *logout* bagi member.	Index.php?c=member&a=logout

4. 2. Pengujian Sistem

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan kesalahan-kesalahan atau kekurangan-kekurangan pada perangkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

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

4.2.1. Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara *alpha* dan *beta*. Pengujian pembangunan *e-commerce* di Toko Alat Musik Hafidz Pro Ard menggunakan data uji berdasarkan data dari Toko Alat Musik Hafidz Pro Ard. Rencana pengujian selengkapnya terlihat pada tabel 4.5.

Tabel 4.5 Rencana Pengujian Sistem

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Validasi username and password	*Black Box*
Pengolahan Petugas	Tambah Petugas	*Black Box*
	Ubah Petugas	*Black Box*
	Hapus Petugas	*Black Box*
	Cari Petugas	*Black Box*

Pengolahan Member	Cari Member	*Black Box*
Pengolahan Kategori	Tambah Kategori	*Black Box*
	Ubah Kategori	*Black Box*
	Hapus Kategori	*Black Box*
	Cari Kategori	*Black Box*
Pengolahan Barang	Tambah Barang	*Black Box*
	Ubah Barang	*Black Box*
	Hapus Barang	*Black Box*
	Cari Barang	*Black Box*
Pengolahan Gambar	Tambah gambar	*Black Box*
	Ubah gambar	*Black Box*
	Hapus gambar	*Black Box*
	Cari gambar	*Black Box*
Pengolahan Kota	Tambah Kota	*Black Box*
	Ubah Kota	*Black Box*
	Hapus Kota	*Black Box*
	Cari Kota	*Black Box*
Pengolahan Provinsi	Tambah Provinsi	*Black Box*
	Ubah Provinsi	*Black Box*
	Cari Provinsi	*Black Box*
Pengolahan Paket	Tambah Paket	*Black Box*
	Ubah Paket	*Black Box*
	Hapus Paket	*Black Box*
	Cari Paket	*Black Box*
Pengolahan Ongkos Kirim	Tambah Ongkos Kirim	*Black Box*
	Ubah Ongkos Kirim	*Black Box*
	Hapus Ongkos Kirim	*Black Box*
	Cari Ongkos Kirim	*Black Box*
Pengolahan Jasa	Tambah Jasa	*Black Box*
	Ubah Jasa	*Black Box*
	Cari Jasa	*Black Box*
Pengolahan Pemesanan	Ubah status pemesanan	*Black Box*
Pengolahan Pembayaran	Cek status pembayaran	*Black Box*
Pengolahan Retur	Ubah status retur	*Black Box*
Pembuatan laporan	Pembuatan laporan	*Black Box*
Backup	Pembuatan *Backup*	*Black Box*
Daftar member	Pendaftaran sebagai member	*Black Box*
*Login M*ember	Masuk ke dalam sistem sebagai member	*Black Box*
Ubah akun	Mengubah akun	*Black Box*
Detail Produk	Menampilkan detail produk	*Black Box*
Keranjang Belanja	Tambah data produk ke keranjang	*Black Box*

	ubah data produk di keranjang	*Black Box*
	Hapus data produk di keranjang	*Black Box*
Kontak	Tampil informasi mengenai nama, alamat, no.telp, dan email Toko Musik Hafidz Pro Ard	*Black Box*
Konfirmasi Pembayaran	*Input* konfirmasi pembayaran	*Black Box*
Cari Produk	Menampilkan data produk yang dicari	*Black Box*

4.2.2. Pengujian Alpha

Hasil implementasi perangkat lunak dari aplikasi *e-commerce* yang dibangun harus diuji terlebih dahulu agar sistem yang dibangun dapat berjalan dengan baik. Pengujian perangkat lunak sistem penerapan *e-commerce* pada Toko Musik Hafidz Pro Ard menggunakan data uji berdasarkan form yang terdapat didalam aplikasi.

4.2.2.1. Pengujian aplikasi

Pengujian yang dilakukan adalah menguji semua proses yang ada di halaman website diantaranya adalah sebagai berikut :

1. Pengujian *login*

   Pengujian *login* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian *login* dapat dilihat pada tabel 4.6.

Tabel 4.6 Pengujian login

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Login berupa *Username* : admin *Password* : admin	Data login dicari dalam *database*, *username* dan data *password* benar login *valid* dan	*Username* dan *Password* benar dapat masuk ke menu utama	[√] Diterima [ ] Ditolak

	dapat masuk ke menu utama		
Kasus dan Hasil Uji salah (Data Salah)			
Memasukan data *username* dan *password* yang tidak sesuai	Tidak dapat login/masuk dan menampilkan pesan kesalahan	Muncul pesan "*Username* atau *password* salah"	[√] Diterima [ ] Ditolak
Memasukan data *username* atau *password* yang tidak terdaftar	Tampilkan pesan kesalahan "*username* atau *password* yang Anda masukkan salah"	Dapat menampilkan pesan kesalahan "*username* atau *password* yang Anda masukkan salah"	[√] Diterima [ ] Ditolak
Memasukkan *username* atau *password* yang belum aktif	Tampilkan pesan kesalahan "*username* atau *password* yang Anda masukkan salah"	Dapat menampilkan pesan kesalahan "*username* atau *password* yang Anda masukkan salah"	[√] Diterima [ ] Ditolak

2. Pengujian tambah petugas

Pengujian tambah petugas dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah petugas dilihat pada tabel 4.7.

Tabel 4.7 Pengujian tambah petugas

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah petugas			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah petugas	Menampilkan *form* tambah petugas	Sistem berhasil menampilkan *form* tambah petugas	[√] Diterima [ ] Ditolak
Mengisi data petugas contoh : *username* : petugas *password* : 12345 *level* : petugas	Tercantun dalam masing-masing *field* kemudian data petugas tersimpan dalam	Data *user* berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak

	database.		
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah petugas			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data petugas dengan mengosongkan *password*	Data petugas tidak dapat disimpan dalam *database* karena data *password* belum terisi	Sistem tidak dapat menyimpan data petugas dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data petugas dengan level petugas belum diisi	Data petugas tidak dapat disimpan dalam *database* karena level belum terisi	Sistem tidak dapat menyimpan data petugas dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data petugas dengan menggunakan *username* yang telah tersedia	Data petugas tidak dapat disimpan dalam *database* karena *username* sudah di gunakan	Sistem tidak dapat menyimpan data petugas dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

3. Pengujian ubah petugas

Pengujian ubah petugas dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah petugas dilihat pada tabel 4.8.

Tabel 4.8 Pengujian ubah *petugas*

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah petugas			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah petugas	Menampilkan form ubah petugas beserta data yang akan diubah	Dapat menampilkan form ubah petugas beserta data yang akan diubah	[√] Diterima [ ] Ditolak

Data petugas lama yang akan diubah contoh : *username* : petugas *password* : 12345 *level* : petugas	Tercantun dalam *field* petugas berupa *username*, *password*, dan *level*	Dapat menampilkan field data petugas yang akan diubah.	[√] Diterima [ ] Ditolak
Mengisi data petugas contoh : *username* : petugas 2 *password* : 54321 *level* : petugas	Dapat mengisi *field* data yang akan diubah dengan benar dan data petugas berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data petugas berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah petugas			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data petugas yang akan diubah contoh : *username* : admin itu adalah administrator yang punya hak akses *password* : admin *level* : petugas	Data petugas tidak berhasil diubah dan menampilkan pesan kesalahan, maxlength *username* adalah 20	Sistem tidak dapat menyimpan data petugas dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Data petugas yang akan diubah contoh : *username* : petugas *password* : petugas *level* : petugas	Data petugas tidak berhasil diubah dan menampilkan pesan kesalahan nama petugas sudah ada	Sistem tidak dapat menyimpan data petugas dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Data petugas yang akan diubah contoh : *username* : - *password* : - *level* : -	Data petugas tidak berhasil diubah karena *username*, *password* dan *level* belum di isi	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

4. Pengujian hapus petugas

Pengujian ubah petugas dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus *user* dilihat pada tabel 4.9.

Tabel 4.9 Pengujian hapus petugas

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus petugas			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data petugas yang akan dihapus contoh : *username* : admin *password* : admin *level* : admin	Tercantum data petugas yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data petugas yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

5. Pengujian tambah kategori

Pengujian tambah kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah kategori dilihat pada tabel 4.10.

Tabel 4.10 Pengujian tambah kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah data kategori	Menampilkan *form* tambah kategori	Sistem menampilkan *form* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data kategori contoh : Nama Kategori: gitar akustik Lama Pengerjaan : 21 hari Jml pengerjaan: 5 per hari Harga standar: 850000	Tercantun dalam masing-masing *field* kemudian data kategori tersimpan dalam *database*.	Data kategori berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak

Berat standar: 1200 gram Ditampilkan: ya			
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori contoh : Nama Kategori: gitar akustik Lama Pengerjaan : 21 Jml pengerjaan: 5 Harga standar: 850000 Berat standar: 1200 Ditampilkan: ya	Kategori tidak dapat disimpan dalam *database* karena data kategori sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data kategori dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data kategori dengan mengosongkan nama kategori	Data kategori tidak akan disimpan karena nama kategori dikosongkan dan akan muncul pesan kesalahan	Sistem tidak akan menyimpan data kategori dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data kategori dengan mengosongkan harga standar	Data kategori tidak akan disimpan dan akan muncul pesan kesalahan	Sistem tidak akan menyimpan data dalam *database* dan akan muncul pesan kesalhan	[√] Diterima [ ] Ditolak

6. Pengujian ubah kategori

Pengujian ubah kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah kategori dilihat pada tabel 4.11.

Tabel 4.11 Pengujian ubah kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah kategori	Menampilkan form ubah kategori beserta data yang akan diubah	Dapat menampilkan form ubah kategori beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data kategori lama contoh : Nama Kategori: gitar akustik Lama Pengerjaan: 21	Tercantun dalam *field* kategori berupa nama kategori.	Dapat menampilkan field data kategori yang akan diubah.	[√] Diterima [ ] Ditolak
Data kategori yang akan diubah contoh : Nama Kategori: gitar elektrik Lama Pengerjaan: 22	Dapat mengisi *field* data yang akan diubah dengan benar dan data kategori berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data kategori berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategori yang akan diubah contoh : Nama Kategori: gitar elektrik Jml produksi harian : CR7	Data kategori tidak berhasil diubah dan menampilkan pesan kesalahan, tipe data lama pengerjaan adalah integer	Sistem tidak dapat menyimpan data kategori dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengosongkan nama kategori	Data kategori tidak berhasil di ubah dan akan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data dalam *database* dan akan menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi nama kategori dengan nama yang	Data kategori tidak berhasil di ubah dan akan	Sistem tidak dapat menyimpan data dalam *database* dan	[√] Diterima [ ] Ditolak

sama	menampilkan pesan kesalahan	akan menampilkan pesan kesalahan	

7. Pengujian cari kategori

Pengujian cari kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari kategori dilihat pada tabel 4.12.

Tabel 4.12 Pengujian cari kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Cari kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategori	Menampilkan data kategori	Berhasil menampilkan data kategori	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama kategori contoh : gitar	Menampilkan data kategori berdasarkan nama kategori yang dicari	Berhasil Menampilkan data kategori berdasarkan nama kategori yang dicari	[√] Diterima [ ] Ditolak

8. Pengujian tambah barang

Pengujian tambah barang dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah barang dilihat pada tabel 4.13.

Tabel 4.13 Pengujian tambah barang

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah barang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah barang	Menampilkan *form* barang	Sistem menampilkan *form* tambah barang	[√] Diterima [ ] Ditolak
Mengisi data barang contoh : Kategori : gitar akustik Nama Produk : Yamaha C315	Tercantun dalam masing-masing *field* kemudian data produk tersimpan dalam *database*.	Data produk berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak

Deskripsi: gitar yamaha terbaru Harga: 850000			
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah barang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data barang contoh : Kategori : gitar akustik Nama Produk : Yamaha C315 Deskripsi: gitar yamaha terbaru Harga: 850000	barang tidak dapat disimpan dalam *database* karena nama barang sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data barang dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data barang dan mengosongkan nama produk	Barang tidak dapat disimpan dalam *database* dan akan menampilkan pesan kesalahan	Sistem tidak akan menyimpan dalam *database* dan akan menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak
Mengosongkan semua data barang	Barang tidak akan disimpan dan akan menampilkan pesan kesalahan	Sistem tidak akan menympan dalam *database* dan akan menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak

9. Pengujian ubah barang

Pengujian ubah barang dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah barang dilihat pada tabel 4.14.

Tabel 4.14 Pengujian ubah barang

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah barang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah barang	Menampilkan form ubah barang beserta data yang	Dapat menampilkan form ubah barang beserta data yang	[√] Diterima [ ] Ditolak

	akan diubah	akan diubah	
Data barang lama contoh : Kategori : gitar akustik Nama Produk : Yamaha C315 Deskripsi: gitar yamaha terbaru Harga: 850000	Tercantun dalam *field* kategori, nama barang, deskripsi, dan harga.	Dapat menampilkan field data produk yang akan diubah.	[√] Diterima [ ] Ditolak
Data barang yang akan diubah contoh : Kategori : gitar akustik Nama Produk : Yamaha AA51 Deskripsi: gitar yamaha Harga: 850000	Dapat mengisi *field* data yang akan diubah dengan benar dan disimpan dalam *database*,	Dapat mengisi *field* data yang akan diubah dengan benar dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah barang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data barang yang akan diubah contoh : Kategori : gitar akustik Nama Produk : Yamaha AA51 Deskripsi: gitar yamaha Harga: 8500zz	Data barang tidak berhasil diubah dan menampilkan pesan kesalahan, tipe data dari harga adalah integer	Sistem tidak dapat menyimpan data subkategori dalam ke dalam *database* dan menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak
Data barang yang diubah dengan mengosongkan kategori	Data barang tidak berhasil di ubah dan akan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data dalam *database* dan akan manampilkan pesan kesalahan	[√] Diterima [ ] Ditolak
Data barang yang diubah dengan mengosongkan semua ubah	Data barang tidak berhasil diubah dan akan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data dalam *database* dan akan menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak

data barang			

10. Pengujian cari barang

Pengujian cari barang dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari barang dilihat pada tabel 4.15.

Tabel 4.15 Pengujian cari barang

Kasus dan Hasil Uji Benar (Data Benar)			
Cari barang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data barang	Menampilkan data barang	Berhasil menampilkan data barang	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama barang contoh : gitar	Menampilkan data barang berdasarkan nama barang yang dicari	Berhasil Menampilkan data barang berdasarkan nama barang yang dicari	[√] Diterima [ ] Ditolak

11. Pengujian cari *member*

Pengujian cari *member* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari *member* dilihat pada tabel 4.16.

Tabel 4.16 Pengujian cari *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Cari member			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Member	Menampilkan data *member*	Berhasil menampilkan data member	[√] Diterima [ ] Ditolak

Data yang akan dicari berdasarkan nama *member* contoh : Rizki	Menampilkan data *member* berdasarkan nama member yang dicari	Berhasil Menampilkan data *member* berdasarkan nama *member* yang dicari	[√] Diterima [ ] Ditolak

12. Pengujian tambah gambar

Pengujian tambah gambar dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah gambar dilihat pada tabel 4.17.

Tabel 4.17 Pengujian tambah gambar

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah gambar			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah gambar	Menampilkan *form* gambar	Sistem menampilkan *form* tambah gambar	[√] Diterima [ ] Ditolak
Mengisi data gambar contoh : Nama gambar: gitar 2 URL: img_produk/gitar2.jpg	Tercantun dalam masing-masing *field* kemudian data gambar tersimpan dalam *database*.	Data gambar berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah gambar			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data gambar contoh : Nama gambar: gitar 2 URL: img_produk/gitar2.jpg	Data gambar tidak dapat disimpan dalam *database* karena nama gambar sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data gambar ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Mengisi data tambah gambar dengan mengosongkan nama gambar	Data barang tidak dapat disimpan dan akan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data ke dalam database dan akan menampilkan pesan kesahan	[√] Diterima [ ] Ditolak
Mengisi tambah barang dengan mengosongkan tambah gambar	Data barang tidak akan disimpan dan akan muncul pesan kesalahan	Sistem tidak akan menyimpan dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

13. Pengujian ubah gambar

Pengujian ubah gambar dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah gambar dilihat pada tabel 4.18.

Tabel 4.18 Pengujian ubah gambar

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah gambar			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah gambar	Menampilkan form ubah gambar beserta data yang akan diubah	Dapat menampilkan form ubah gambar beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data warna lama contoh : Nama gambar: gitar 2 URL: img_produk/gitar2.jpg	Tercantun dalam *field* gambar berupa nama gambar dan URL.	Dapat menampilkan field data gambar yang akan diubah.	[√] Diterima [ ] Ditolak
Data warna yang akan diubah contoh : Nama gambar:	Dapat mengisi *field* data yang akan diubah dengan benar	Dapat mengisi *field* data yang akan diubah dengan benar dan data gambar	[√] Diterima [ ] Ditolak

gitar 3 URL: img_produk/gitar3.jpg	dan data gambar berhasil diubah dan disimpan dalam *database*.	berhasil diubah dan disimpan dalam *databsase*	
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah gambar			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data gambar yang akan diubah contoh : Nama gambar: gitar 2 URL: img_produk/gitar2.psd	Data gambar tidak berhasil diubah dan menampilkan pesan kesalahan, format gambar harus jpg atau png atau gif	Sistem tidak dapat menyimpan data warna dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data ubah gambar dengan mengosongkan nama gambar	Data barang tidak dapat disimpan dan akan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data ke dalam database dan akan menampilkan pesan kesahan	[√] Diterima [ ] Ditolak
Mengisi ubah barang dengan mengosongkan ubah gambar	Data barang tidak akan disimpan dan akan muncul pesan kesalahan	Sistem tidak akan menyimpan dalam *database* dan akan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

14. Pengujian hapus gambar

Pengujian hapus gambar dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus gambar dilihat pada tabel 4.19.

Tabel 4.19 Pengujian hapus gambar

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus gambar			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Data gambar yang akan dihapus contoh : Nama gambar: gitar 2 URL: img_produk/gitar2.jpg	Tercantum data gambar yang akan dihapus, setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data gambar yang akan dihapus, setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

15. Pengujian cari gambar

Pengujian cari gambar dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari gambar dilihat pada tabel 4.20.

Tabel 4.20 Pengujian cari gambar

Kasus dan Hasil Uji Benar (Data Benar)			
Cari gambar			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data gambar	Menampilkan data gambar	Berhasil menampilkan data gambar	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama gambar contoh : Gitar 2	Menampilkan data gambar berdasarkan nama gambar yang dicari	Berhasil Menampilkan data gambar berdasarkan nama gambar yang dicari	[√] Diterima [ ] Ditolak

16. Pengujian tambah kota

Pengujian tambah kota dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah kota dilihat pada tabel 4.21.

Tabel 4.21 Pengujian tambah kota

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah kota			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah kota	Menampilkan *form* kota	Sistem menampilkan *form* tambah	[√] Diterima [ ] Ditolak

		kota	
Mengisi data kota contoh : Provinsi: Jawa Barat Nama kota: Bandung	Tercantun dalam masing-masing *field* kemudian data kota tersimpan dalam *database*.	Data kota berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah kota			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kota contoh : Provinsi: Jawa Barat Nama kota: Bandung	Data kota tidak dapat disimpan dalam *database* karena nama kota sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data kota ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi tambah kota dengan mengosongkan provinsi	Data kota tidak dapat disimpan dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data kota ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data kota dengan mengosongkan tambah kota	Data kota tidak dapat disimpan dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data kota ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

17. Pengujian ubah kota

Pengujian ubah kota dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah kota dilihat pada tabel 4.22.

Tabel 4.22 Pengujian ubah kota

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah kota			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah kota	Menampilkan form ubah kota beserta data yang	Dapat menampilkan form ubah kota	[√] Diterima [ ] Ditolak

	akan diubah	beserta data yang akan diubah	
Data kota lama contoh : Provinsi: Jawa Barat Nama kota: Bandung	Tercantun dalam *field* kota berupa nama kota dan URL.	Dapat menampilkan field data kota yang akan diubah.	[√] Diterima [ ] Ditolak
Data kota yang diubah contoh : Provinsi: Jawa Tengah Nama kota: Semarang	Dapat mengisi *field* data yang akan diubah dengan benar dan data kota berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data kota berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah kota			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kota yang akan diubah contoh : Provinsi: Bandung Nama kota: Bandung	Data kota tidak berhasil diubah dan menampilkan pesan kesalahan, nama provinsi tidak ada	Sistem tidak dapat menyimpan data kota dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Data kota yang akan di ubah dengan mengosongkan provinsi	Data kota tidak berhasil diubah dan menampilkan pesan kesalahan, nama provinsi kosong	Sistem tidak dapat menyimpan data kota dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Data kota yang akan di ubah dengan mengosongkan nama kota	Data kota tidak berhasil diubah dan menampilkan pesan kesalahan, nama kota kosong	Sistem tidak dapat menyimpan data kota dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

18. Pengujian hapus kota

Pengujian hapus kota dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus kota dilihat pada tabel 4.23.

Tabel 4.23 Pengujian hapus kota

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus kota			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kota yang akan dihapus contoh : Nama kota: Bandung	Tercantum data kota yang akan dihapus, setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data kota yang akan dihapus, setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

19. Pengujian cari kota

Pengujian cari kota dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari kota dilihat pada tabel 4.24.

Tabel 4.24 Pengujian cari kota

Kasus dan Hasil Uji Benar (Data Benar)			
Cari kota			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kota	Menampilkan data kota	Berhasil menampilkan data kota	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama kota contoh : Bandung	Menampilkan data kota berdasarkan nama kota yang dicari	Berhasil Menampilkan data kota berdasarkan nama kota yang dicari	[√] Diterima [ ] Ditolak

20. Pengujian tambah provinsi

Pengujian tambah provinsi dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah provinsi dilihat pada tabel 4.25.

Tabel 4.25 Pengujian tambah provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah provinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah provinsi	Menampilkan *form* provinsi	Sistem menampilkan *form* tambah provinsi	[√] Diterima [ ] Ditolak
Mengisi data provinsi contoh : Nama provinsi: Jawa Barat	Tercantun dalam *field* kemudian data provinsi tersimpan dalam *database*.	Data provinsi berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah provinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data provinsi contoh : Nama provinsi: Jawa Barat	Data provinsi tidak dapat disimpan dalam *database* karena nama provinsi sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data provinsi ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengosongkan data provinsi	Data provinsi tidak dapat disimpan dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data provinsi ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

21. Pengujian ubah provinsi

Pengujian ubah provinsi dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah provinsi dilihat pada tabel 4.26.

Tabel 4.26 Pengujian ubah provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah provinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah provinsi	Menampilkan form ubah	Dapat menampilkan form ubah provinsi	[√] Diterima [ ] Ditolak

	provinsi beserta data yang akan diubah	beserta data yang akan diubah	
Data provinsi lama contoh : Nama provinsi: Jawa Barat	Tercantun dalam *field* provinsi berupa nama provinsi	Dapat menampilkan field data provinsi yang akan diubah.	[√] Diterima [ ] Ditolak
Data provinsi yang akan diubah contoh : Nama provinsi: Jawa Timur	Dapat mengisi *field* data yang akan diubah dengan benar dan data provinsi berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data provinsi berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah provinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengosongkan nama provinsi	Data provinsi tidak berhasil diubah dan menampilkan pesan kesalahan, nama provinsi harus diisi	Sistem tidak dapat menyimpan data warna dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data provinsi contoh : Nama provinsi: Jawa Barat	Data provinsi tidak dapat disimpan dalam *database* karena nama provinsi sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data provinsi ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

22. Pengujian cari provinsi

Pengujian cari provinsi dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari provinsi dilihat pada tabel 4.27.

Tabel 4.27 Pengujian cari provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Cari provinsi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data provinsi	Menampilkan data provinsi	Berhasil menampilkan data provinsi	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama provinsi contoh : DKI JAKARTA	Menampilkan data provinsi berdasarkan nama provinsi yang dicari	Berhasil Menampilkan data provinsi berdasarkan nama provinsi yang dicari	[√] Diterima [ ] Ditolak

23. Pengujian tambah paket

Pengujian tambah paket dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah paket dilihat pada tabel 4.28.

Tabel 4.28 Pengujian tambah paket

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah paket			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah paket	Menampilkan *form* paket	Sistem menampilkan *form* tambah paket	[√] Diterima [ ] Ditolak
Mengisi data paket contoh : Jasa : JNE Nama paket: Reguler	Tercantun dalam masing-masing *field* kemudian data paket tersimpan dalam *database*.	Data paket berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah paket			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Mengisi data paket contoh : Jasa : JNE Nama paket: Reguler	Data paket tidak dapat disimpan dalam *database* karena Jasa dengan nama paket sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data paket ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengosongkan data paket jasa	Data paket didak dapat disimpan dan akan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data paket ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

24. Pengujian ubah paket

Pengujian ubah paket dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah paket dilihat pada tabel 4.29.

Tabel 4.29 Pengujian ubah paket

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah paket			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah paket	Menampilkan form ubah paket beserta data yang akan diubah	Dapat menampilkan form ubah paket beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data paket lama contoh : Jasa : JNE Nama paket: Reguler	Tercantun dalam *field* paket berupa nama paket dan Jasa.	Dapat menampilkan field data paket yang akan diubah.	[√] Diterima [ ] Ditolak
Data paket yang akan diubah contoh : Jasa: JNE Nama paket: YES	Dapat mengisi *field* data yang akan diubah dengan benar dan data paket berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data paket berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah paket			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Data paket yang akan diubah contoh : Jasa : JNE Nama paket: *null*	Data paket tidak berhasil diubah dan menampilkan pesan kesalahan, nama paket harus diisi.	Sistem tidak dapat menyimpan data paket dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Ubah data paket dengan mengosongkan jasa	Data paket tidak berhasil diubah dan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data paket dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Memasukan data paket dengan nama jasa sudah ada	Data paket tidak berhasil diubah dan menapilkan pesan kesalahan	Sistem tidak dapat menyimpan data paket dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

25. Pengujian hapus paket

Pengujian hapus paket dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus paket dilihat pada tabel 4.30.

Tabel 4.30 Pengujian hapus paket

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus paket			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data paket yang akan dihapus contoh : Jasa: JNE Nama paket: YES	Tercantum data paket yang akan dihapus, setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data paket yang akan dihapus, setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

26. Pengujian cari paket

Pengujian cari paket dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari paket dilihat pada tabel 4.31.

Tabel 4.31 Pengujian cari paket

Kasus dan Hasil Uji Benar (Data Benar)			
Cari paket			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data paket	Menampilkan data paket	Berhasil menampilkan data paket	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama paket contoh : YES	Menampilkan data paket berdasarkan nama paket yang dicari	Berhasil Menampilkan data paket berdasarkan nama paket yang dicari	[√] Diterima [ ] Ditolak

27. Pengujian tambah ongkos kirim

Pengujian tambah ongkos kirim dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah ongkos kirim dilihat pada tabel 4.32.

Tabel 4.32 Pengujian tambah ongkos kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah ongkos kirim			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah ongkos kirim	Menampilkan *form* ongkos kirim	Sistem menampilkan *form* tambah ongkos kirim	[√] Diterima [ ] Ditolak
Mengisi data ongkos kirim contoh : Paket: JNE Reguler Kota: Jakarta Tarif: 8000	Tercantun dalam masing-masing *field* kemudian data ongkos kirim tersimpan dalam *database*.	Data ongkos kirim berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah ongkos kirim			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Mengisi data ongkos kirim contoh : Paket: JNE Reguler Kota: Jakarta Tarif: 8000	Data ongkos kirim tidak dapat disimpan dalam *database* karena nama ongkos kirim sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data ongkos kirim ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengosongkan paket ongkos kirim	Data ongkos kirim tidak dapat disimpan dan akan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data ongkos kirim ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengosongkan kota ongkos kirim	Data ongkos kirim tidak dapat disimpan dan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data ongkos kirim ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

28. Pengujian ubah ongkos kirim

Pengujian ubah ongkos kirim dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah ongkos kirim dilihat pada tabel 4.33.

Tabel 4.33 Pengujian ubah ongkos kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah ongkos kirim			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah ongkos kirim	Menampilkan form ubah ongkos kirim beserta data yang akan diubah	Dapat menampilkan form ubah ongkos kirim beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data ongkos kirim lama contoh : Paket: JNE	Tercantun dalam *field* ongkos kirim berupa nama ongkos kirim dan	Dapat menampilkan field data ongkos kirim yang akan diubah.	[√] Diterima [ ] Ditolak

Reguler Kota: Jakarta Tarif: 8000	URL.		
Data ongkos kirim yang diubah contoh : Paket: JNE Reguler Kota: Cirebon Tarif: 9000	Dapat mengisi *field* data yang akan diubah dengan benar dan data ongkos kirim berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data ongkos kirim berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah ongkos kirim			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data ongkos kirim contoh : Paket: JNE Reguler Kota: Jakarta Tarif: 0	Data ongkos kirim tidak berhasil diubah dan menampilkan pesan kesalahan, Tarif tidak boleh 0 nol	Sistem tidak dapat menyimpan data warna dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengosongkan paket ongkos kirim	Data ongkos kirim tidak dapat disimpan dan akan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data ongkos kirim ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengosongkan kota ongkos kirim	Data ongkos kirim tidak dapat disimpan dan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data ongkos kirim ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

29. Pengujian hapus ongkos kirim

Pengujian hapus ongkos kirim dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus ongkos kirim dilihat pada tabel 4.34.

Tabel 4.34 Pengujian hapus ongkos kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus ongkos kirim			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data ongkos kirim contoh : Paket: JNE Reguler Kota: Jakarta Tarif: 8000	Tercantum data ongkos kirim yang akan dihapus, setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data ongkos kirim yang akan dihapus, setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

30. Pengujian cari ongkos kirim

Pengujian cari ongkos kirim dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari ongkos kirim dilihat pada tabel 4.35.

Tabel 4.35 Pengujian cari ongkos kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Cari ongkos kirim			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data ongkos kirim	Menampilkan data ongkos kirim	Berhasil menampilkan data ongkos kirim	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan ID ongkos kirim contoh : 25	Menampilkan data ongkos kirim berdasarkan nama ongkos kirim yang dicari	Berhasil Menampilkan data ongkos kirim berdasarkan nama ongkos kirim yang dicari	[√] Diterima [ ] Ditolak

31. Pengujian tambah jasa

Pengujian tambah jasa dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah jasa dilihat pada tabel 4.36.

Tabel 4.36 Pengujian tambah jasa

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah jasa			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah jasa	Menampilkan *form* jasa	Sistem menampilkan *form* tambah jasa	[√] Diterima [ ] Ditolak
Mengisi data jasa contoh : Nama jasa: JNE	Tercantun dalam masing-masing *field* kemudian data jasa tersimpan dalam *database*.	Data jasa berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah jasa			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data jasa contoh : Nama jasa: JNE	Data jasa tidak dapat disimpan dalam *database* karena nama jasa sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data jasa ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengosongkan nama jasa	Data tidak dapat disimpan dan akan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data jasa ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

32. Pengujian ubah jasa

Pengujian ubah jasa dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah jasa dilihat pada tabel 4.37.

Tabel 4.37 Pengujian ubah jasa

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah jasa			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah jasa	Menampilkan form ubah jasa beserta data yang akan diubah	Dapat menampilkan form ubah jasa beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data jasa lama contoh : Nama jasa: JNE	Tercantun dalam *field* jasa berupa nama jasa dan URL.	Dapat menampilkan field data jasa yang akan diubah.	[√] Diterima [ ] Ditolak
Data jasa yang akan diubah contoh : Nama jasa: POS	Dapat mengisi *field* data yang akan diubah dengan benar dan data jasa berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data jasa berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah jasa			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data jasa yang akan diubah contoh : Nama jasa: *null*	Data jasa tidak berhasil diubah dan menampilkan pesan kesalahan, nama jasa tidak boleh kosong	Sistem tidak dapat menyimpan data warna dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Ubah jasa dengan nama sama	Data jasa tidak berhasil diubah dan menampilkan pesan kesalahan	Sistem tidak dapat menyimpan data warna dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

33. Pengujian cari jasa

Pengujian cari jasa dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari jasa dilihat pada tabel 4.38.

Tabel 4.38 Pengujian cari jasa

Kasus dan Hasil Uji Benar (Data Benar)			
Cari jasa			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data jasa	Menampilkan data jasa	Berhasil menampilkan data jasa	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama jasa contoh : JNE	Menampilkan data jasa berdasarkan nama jasa yang dicari	Berhasil Menampilkan data jasa berdasarkan nama jasa yang dicari	[√] Diterima [ ] Ditolak

34. Pengujian ubah status pemesanan

Pengujian ubah status pemesanan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah status pemesanan dapat dilihat pada tabel 4.39.

Tabel 4.39 Pengujian ubah status pemesanan

Kasus dan Hasil Uji Benar (Data Benar)			
Pengolahan data pemesanan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Pilih Data Transaksi Pemesanan	Tampil data pemesanan beserta status pemesanan.	Sistem berhasil menampilkan data pemesanan beserta status pemesanan	[√] Diterima [ ] Ditolak
Pilih detail pemesanan	Tampil data pemesanan sesuai dengan detail pemesanan yang dipilih	Sistem berhasil menampilkan data pemesanan sesuai dengan detail pemesanan yang dipilih	[√] Diterima [ ] Ditolak
Mengubah status pemesanan sesuai dengan kondisi pemesanan sekarang	Status Pemesanan berubah sesuai dengan status yang dimasukan	Sistem dapat mengubah status pemesanan sesuai dengan status yang dimasukkan	[√] Diterima [ ] Ditolak

35. Pengujian ubah status pembayaran

Pengujian ubah status pembayaran dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah status pembayaran dapat dilihat pada tabel 4.40.

Tabel 4.40 Pengujian ubah status pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah status pembayaran			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Pengolahan Data Transaksi pembayaran	Tampil data pembayaran yang menampilkan data pembayaran dari member.	Sistem berhasil menampilkan data pembayaran.	[√] Diterima [ ] Ditolak
Tandai Diterima	Status pembayaran dari pemesanan menjadi diterima	Sistem berhasil merubah status pembayaran dari belum dicek menjadi diterima.	[√] Diterima [ ] Ditolak

36. Pengujian ubah status retur

Pengujian ubah status retur dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah status retur dapat dilihat pada tabel 4.41.

Tabel 4.41 Pengujian ubah status retur

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah status retur			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Pilih Data Transaksi Retur	Tampil data retur beserta status retur.	Sistem berhasil menampilkan data retur beserta status retur	[√] Diterima [ ] Ditolak
Pilih detail retur	Tampil data retur sesuai dengan detail retur yang dipilih	Sistem berhasil menampilkan data retur sesuai dengan detail retur yang dipilih	[√] Diterima [ ] Ditolak
Mengubah status retur	Status Retur berubah sesuai dengan status	Sistem dapat mengubah status	[√] Diterima [ ] Ditolak

sesuai dengan kondisi retur sekarang	yang dimasukan	retur sesuai dengan status yang dimasukkan	

37. Pengujian pembuatan laporan

Pengujian pembuatan laporan penjualan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian pembuatan laporan penjualan dilihat pada tabel 4.42.

Tabel 4.42 Pengujian pembuatan laporan

Kasus dan Hasil Uji Benar (Data Benar)			
Pembuatan laporan			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Cetak laporan pemesanan	Menampilkan form laporan penjualan	Dapat menampilkan form laporan penjualan	[√] Diterima [ ] Ditolak
Memasukan bulan dan tahun contoh : Bulan : Februari Tahun : 2012	Tercantun dalam *field* laporan berupa bulan dan tahun	Dapat menampilkan data laporan pada periode yang dipilih.	[√] Diterima [ ] Ditolak
Proses laporan	Menampilkan laporan pemesanan pada periode tertentu dalam bentuk *file* pdf.	Dapat menampilkan laporan pemesanan pada periode tertentu dalam benuk *file* pdf	[√] Diterima [ ] Ditolak

38. Pengujian ubah akun

Pengujian ubah akun dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah akun dilihat pada tabel 4.43.

Tabel 4.43 Pengujian ubah akun

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah akun			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah akun	Menampilkan form ubah akun beserta data yang akan diubah	Dapat menampilkan form ubah akun beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data akun lama yang	Tercantun dalam	Dapat	[√]

akan diubah contoh : *username* : admin *password* : admin	*field* akun berupa *username*, *password*	menampilkan field data akun yang akan diubah.	Diterima [ ] Ditolak
Mengisi data akun contoh : *username* : admin1 *password* : admin1	Dapat mengisi *field* data yang akan diubah dengan benar dan data akun berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data akun berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah akun			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data akun yang akan diubah contoh : *username* : admin itu adalah administrator punya semua hak akses *password* : admin *level* : admin	Data akun tidak berhasil diubah dan menampilkan pesan kesalahan, maxlength *username* adalah 20	Sistem tidak dapat menyimpan data akun dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Ubah data akun *username* tidak isi	Data akun tidak berhasil diubah dan menampilkan pesan kesalahan,	Sistem tidak dapat menyimpan data akun dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Ubah data akun *password* tidak isi	Data akun tidak berhasil diubah dan menampilkan pesan kesalahan,	Sistem tidak dapat menyimpan data akun dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

39. Pengujian pendaftaran sebagai *member*

Pengujian pendaftaran sebagai *member* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian pendaftaran sebagai *member* dilihat pada tabel 4.44.

Tabel 4.44 Pengujian pendaftaran sebagai *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Pendaftaran sebagai *member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Daftar *member*	Menampilkan form daftar *member*	Dapat menampilkan form daftar member	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh masukan : Nama : Rizki Ariwijaya Email : ryusatoshi@yahool.com password : 12345 Kota : Bandung Kodepos : 40514 Telepon : 085720016479 konfirmasi password : 12345	Sistem menyimpan data pendaftaran kedalam *database*	Berhasil menyimpan data pendaftaran member kedalam database	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
pendaftaran sebagai *member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Masukan data pendaftaran, contoh masukan : Nama : Rizki Ariwijaya Email : ryusatoshi@yahoo.com password : 12345 Kota : Bandung Kodepos : 40514	Tampil pesan kesalahan field *password* dengan field *password* konfirmasi tidak sama	Dapat pesan kesalahan field *password* dengan field *password* konfirmasi tidak sama	[√] Diterima [ ] Ditolak

Telepon : 085720016479 konfirmasi password : 787878			
Masukan data pendaftaran *password* tidak diisi	Tampil pesan kesalahan *password* belum diisi	Sistem tidak dapat menyimpan data dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data pendaftaran nama tidak diisi	Tampil pesan kesalahan nama belum diisi	Sistem tidak dapat menyimpan data dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

40. Pengujian ubah akun

Pengujian ubah akun dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah akun dilihat pada tabel 4.45.

Tabel 4.45 Pengujian ubah akun

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah akun			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah akun	Menampilkan form ubah akun beserta data yang akan diubah	Dapat menampilkan form ubah akun beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data akun lama yang akan diubah contoh : Nama : Rizki Ariwijaya Email : ryusatoshi@yahool.co	Tercantun dalam *field* akun berupa *username*, *password*, nama, email, jenis-kelamin, kota,	Dapat menampilkan field data akun yang akan diubah.	[√] Diterima [ ] Ditolak

m password : 12345 Kota : Bandung Kodepos : 40514 Telepon : 085720016479 konfirmasi password : 12345	kodepos, telepon		
Mengisi data akun contoh : Nama : Rizki AW Email : ryusatoshi@yahool.com password : 12345 Kota : Jakarta Kodepos : 32000 Telepon : 085720016479 konfirmasi password : 12345	Dapat mengisi *field* data yang akan diubah dengan benar dan data akun berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data akun berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah akun			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data akun contoh : Nama : Rizki AW Email : ryusatoshi@yahool.com password : 12345 Kota : Jakarta Kodepos : 32000 Telepon : 085720016479 konfirmasi password : 56565	Data akun tidak berhasil diubah dan menampilkan pesan kesalahan, password dengan password konfirmasi tidak sama	Sistem tidak dapat menyimpan data akun dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Masukan data *email* tidak diisi	Tampil pesan kesalahan *email* belum diisi	Sistem tidak dapat menyimpan data dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Masukan data nama tidak diisi	Tampil pesan kesalahan nama belum diisi	Sistem tidak dapat menyimpan data dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

41. Pengujian tampil data kategori

Pengujian menampilkan data kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian menampilkan data kategori dilihat pada tabel 4.46.

Tabel 4.46 Pengujian tampil data kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Tampil data kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
kategori	Menampilkan data kategori yang dipilih	Dapat menampilkan data kategori	[√] Diterima [ ] Ditolak

42. Tampil detail produk

Pengujian tampil detail produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tampil detail produk dilihat pada tabel 4.47.

Tabel 4.47 Pengujian tampil detail produk

Kasus dan Hasil Uji Benar (Data Benar)			
Tampil detail produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Pilih Produk	Menampilkan data detail produk yang dipilih	Dapat menampilkan detail dari produk yang dipilih	[√] Diterima [ ] Ditolak

43. Pengujian tambah produk ke keranjang

Pengujian tambah produk ke keranjang dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah produk ke keranjang dilihat pada tabel 4.48.

Tabel 4.48 Pengujian tambah produk ke keranjang

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah produk ke keranjang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Detail produk	Menampilkan detail produk yang akan dibeli	Sistem menampilkan detail produk yang kaan dibeli	[√] Diterima [ ] Ditolak
membeli produk dengan memasukan data produk contoh : produk : Jimbe Tali	Buku masuk kedalam keranjang belanja.	Sistem berhasil memasukan data buku kedalam keranjang belanja	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah produk ke keranjang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
membeli produk dengan memasukan data julah produk contoh : Jumlah : 12	Barang gagal masuk ke keranjang, dan keluar pesan bahwa jml maksimal adalah 10	Sistem gagal memasukan data produk ke keranjang dan menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak
membeli produk dengan memasukan data julah produk contoh : Jumlah : 0	Barang gagal masuk ke keranjang	Sistem gagal memasukan data produk ke keranjang dan menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak

44. Pengujian *update* keranjang

Pengujian *update* keranjang dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian *update* keranjang dilihat pada tabel 4.49.

Tabel 4.49 Pengujian *update* keranjang

Kasus dan Hasil Uji Benar (Data Benar)			
Update keranjang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Keranjang belanja	Menampilkan halaman keranjang belanja yang berisi buku yang telah dibeli	Sistem Menampilkan halaman keranjang belanja yang berisi buku yang telah dibeli	[√] Diterima [ ] Ditolak
Data keranjang lama contoh : produk : jimbe Tali Jumlah :10	Tercantun dalam *field* keranjang berupa nama produk dan jumlah pemesanan	Dapat menampilkan field data keranjang yang akan diubah.	[√] Diterima [ ] Ditolak
Data keranjang yang akan diubah contoh : produk : Jimbe Tali Jumlah : 5	data produk yang telah diubah disimpan di dalam keranjang	Dapat mengisi *field* data yang akan diubah dengan benar dan data keranjang berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Update keranjang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data keranjang yang akan diubah contoh : produk : Jimbe Tali Jumlah : 30B	Data keranjang tidak berhasil diubah dan menampilkan pesan kesalahan, jumlah mempunyai tipe data integer	Sistem tidak dapat menyimpan data kategori dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

Data keranjang yang akan diubah contoh : produk : Jimbe Tali Jumlah : 12	Data keranjang tidak berhasil diubah dan menampilkan pesan kesalahan, jumlah jumlah max 10	Sistem tidak dapat menyimpan data kategori dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

45. Pengujian hapus data produk dari keranjang

Pengujian hapus data barang dari keranjang dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus data barang dari keranjang dilihat pada tabel 4.50.

Tabel 4.50 Pengujian hapus data barang dari keranjang

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus data produk dari keranjang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data buku dalam keranjang yang akan dihapus : contoh : produk : Jimbe Tali	Tercantum data buku dalam keranjang yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data buku dalam keranjang yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

46. Pengujian kontak

Pengujian kontak dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian kontak dilihat pada tabel 4.51.

Tabel 4.51 Pengujian kontak

Kasus dan Hasil Uji Benar (Data Benar)			
Tampil Kontak			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Kontak	Menampilkan Data Kontak	Dapat menampilkan Data Kontak	[√] Diterima [ ] Ditolak

47. Tampil Detail Produk

Pengujian tampil detail produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tampil detail produk dilihat pada tabel 4.52.

Tabel 4.52 Pengujian tampil detail produk

Kasus dan Hasil Uji Benar (Data Benar)			
Tampil detail produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data Produk	Menampilkan Detail Produk yang dipilih	Dapat menampilkan detail produk yang diplih	[√] Diterima [ ] Ditolak

48. Pengujian konfirmasi pembayaran

Pengujian konfirmasi pembayaran dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian konfirmasi pembayaran dilihat pada tabel 4.53.

Tabel 4.53 Pengujian konfirmasi pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Konfirmasi pembayaran			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Konfirmasi pembayaran	Menampilkan *form* konfirmasi pembayaran	Sistem menampilkan *form* konfirmasi pembayaran	[√] Diterima [ ] Ditolak
Mengisi data konfirmasi pembayaran contoh : Tgl Bayar : 10-02-2012 Bank Asal: BCA Bank Tujuan: BNI Atas nama : Iky	Tercantun dalam masing-masing *field* kemudian data konfirmasi dan tersimpan dalam *database*.	Data konfirmasi pembayaran berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Konfirmasi pembayaran			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan

Mengisi data konfirmasi pembayaran contoh : Tgl Bayar : 10-2012-02 Bank Asal: BCA Bank Tujuan: BNI Atas nama : Iky	data konfirmasi pembayaran tidak dapat disimpan dalam *database* karena format tanggal salah	Sistem tidak dapat menyimpan data konfirmasi pembayaran dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak
Mengisi data konfirmasi pembayaran dengan bank asal tidak diisi	Data konfirmasi pembayaran tidak dapat disimpan dalam *database* dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data konfirmasi pembayaran dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

49. Pengujian cari produk

Pengujian cari produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari produk dilihat pada tabel 4.54.

Tabel 4.54 Pengujian cari produk

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data produk	Menampilkan data produk	Berhasil menampilkan data produk	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama produk contoh : Gitar	Menampilkan data produk berdasarkan produk buku yang dicari	Berhasil Menampilkan data produk berdasarkan nama produk yang dicari	[√] Diterima [ ] Ditolak

4.2.2.2. Kesimpulan Hasil Pengujian Alpha

Berdasarkan hasi pengujian alpha (fungsional) dengan kasus uji diatas dapat ditarik kesimpulan bahwa pembangunan *e-commerce* di Toko Musik Hafidz Pro Ard

secara fungsional sistem sudah dapat digunakan dan menghasilkan *output* yang diharapkan.

4.2.3. Pengujian Beta

Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung di tempat penelitian dengan membagikan kuesioner kepada 10 orang dari masyarakat umum dan melakukan wawancara kepada pengguna sistem atau petugas di Toko musik hafidz pro ard. Penelitian ini dilakukan di dua lingkungan yaitu :

1. Lingkungan masyarakat umum

   Penelitian di lingkungan pada masyarakat umum khususnya yang sudah terbiasa dengan teknologi internet, dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat memberikan kemudahan masyarakat umum untuk melakukan transaksi pembelian barang secara online.

2. Lingkungan Toko Musik Hafidz Pro Ard

   Penelitian di lingkungan Toko Musik Hafidz Pro Ard dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang ada.

Berdasarkan data hasil kusioner tersebut, dapat dicari presentase masing-masing jawaban dengan menggunakan rumus :

$$Y = P/Q * 100\ \%$$

Keterangan :
P = Banyaknya jawaban responden tiap soal
Q = Jumlah responden
Y = Nilai prosentase

Berikut ini adalah hasil *presentase* masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuesioner ini diujikan kepada 10 orang.

4.2.3.1. Kuesioner member

Kuesioner untuk member terdiri dari 5 pertanyaan (contoh kuesioner dapat diliihat pada lampiran) dengan menggunakan rumus Y=P/Q*100% , dengan pertanyaan sebagai berikut :

1. Apakah anda mudah mendapatkan informasi barang beserta detailnya melalui *website* Toko Musik Hafidz Pro Ard ?

Tabel 4.55 Hasil kuesioner pertanyaan nomor 1 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
1	1	Sangat Mudah	3	30
	2	Mudah	6	60
	3	Cukup Mudah	1	10
	4	Biasa Saja	-	-
	5	Tidak Mudah	-	-
Jumlah			10	100

Berdasarkan hasil prosentase pada tabel 4.45 maka dapat disimpulkan sebanyak 3 atau 30% menyatakan sangat mudah, 6 atau 60% menyatakan mudah dan 1 atau 10% menyatakan cukup mudah untuk mendapatkan informasi barang beserta detail *website* Toko Musik Hafidz Pro Ard.

2. Apakah anda setuju bahwa *website* ini mudah dioperasikan dan menarik ?

Tabel 4.56 Hasil kuesioner pertanyaan nomor 2 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
2	1	Sangat Setuju	1	10
	2	Setuju	8	80
	3	Cukup Setuju	1	10
	4	Biasa Saja	-	-

	5	Tidak Setuju	-	-
Jumlah			10	100

Berdasarkan hasil prosentase pada tabel 4.46 maka dapat disimpulkan sebanyak 1 orang atau 10% menyatakan sangat setuju, 8 orang atau 80% menyatakan setuju dan 1 orang atau 10% menyatakan cukup setuju saja bahwa tampilan dari website *e-commerce* pada toko Toko musik hafidz pro ard menarik dan *user friendly*.

3. Apakah anda setuju bahwa *website* ini lebih memudahkan dalam melakukan pemesanan barang khususnya dari tempat yang berjauhan dengan toko ?

Tabel 4.57 Hasil kuesioner pertanyaan nomor 3 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
3	1	Sangat Setuju	3	30
	2	Setuju	5	50
	3	Cukup Setuju	1	10
	4	Biasa Saja	1	10
	5	Tidak Setuju	-	-
Jumlah			10	100

Berdasarkan hasil prosentase pada tabel 4.47 maka dapat disimpulkan sebanyak 3 orang atau 30% menyatakan sangat setuju 5 orang atau 50% menyatakan setuju, 1 orang atau 10% menyatakan cukup setuju dan 1 orang atau 10% menyatakan biasa saja bahwa *website* ini lebih memudahkan dalam melakukan pemesanan barang khususnya dari tempat yang berjauhan dengan toko.

4. Apakah anda setuju bahwa sistem pembayaran *online* yang digunakan lebih memudahkan dalam proses transaksi ?

Tabel 4.58 Hasil kuesioner pertanyaan nomor 4 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
4	1	Sangat Setuju	3	30
	2	Setuju	6	60
	3	Cukup Setuju	1	10
	4	Biasa Saja	-	-
	5	Tidak Setuju	-	-
Jumlah			10	100

Berdasarkan hasil prosentase pada tabel 4.48 maka dapat disimpulkan sebanyak 3 orang atau 30% menyatakan sangat setuju, 6 orang atau 60% menyatakan setuju dan 1 orang atau 10% menyatakan cukup setuju bahwa sistem pembayaran *online* yang digunakan lebih memudahkan dalam proses transaksi.

5. Apakah anda setuju bahwa *website* ini memudahkan dalam mendapatkan informasi mengenai status pemesanan barang yang dilakukan ?

Tabel 4.59 Hasil kuesioner pertanyaan nomor 5 member

Pertanyaan	No	Keterangan	Responden	Prosentase (%)
5	1	Sangat Setuju	4	40
	2	Setuju	4	40
	3	Cukup Setuju	1	10
	4	Biasa Saja	1	10
	5	Tidak Setuju	-	-
Jumlah			10	100

Berdasarkan hasil prosentase pada tabel 4.49 maka dapat disimpulkan sebanyak 4 orang atau 40% menyatakan sangat setuju, 4 orang atau 40% menyatakan setuju, 1 orang atau 10% menyatakan setuju dan 1 orang atau 10% menyatakan biasa saja bahwa *website* ini memudahkan dalam mendapatkan informasi mengenai status pemesanan barang yang dilakukan.

4.2.3.2. Wawancara administrator

Wawancara dilakukan kepada petugas yang telah menggunakan sistem dan melakukan proses-proses yang ada dalam sistem tersebut. Wawancara dilakukan dengan mengajukan 4 pertanyaan kepada petugas, hasil dari wawancara dapat dilihat sebagai berikut :

1. Apakah dengan adanya *website e-commerce* ini dapat memudahkan dalam melakukan promosi produk di Toko Musik Hafidz Pro Ard ?

   Jawaban:

   Website yang dibangun dapat membantu dalam mempromosikan produk kepada masyarakat luas khususnya kepada para pengguna internet.

2. Apakah halaman administrator dari *website* ini mudah dipahami dan digunakan ?

   Jawaban:

   Secara keseluruhan, halaman administrator mudah untuk dipahami dan digunakan untuk melakukan proses-proses pengolahan data hingga pembuatan laporan.

3. Apakah *website* dapat memudahkan dalam mengelola data transaksional Toko Musik Hafidz Pro Ard ?

   Jawaban:

   Data transaksional khusunya data pemesanan lebih mudah untuk dikelola dan proses produksi yang dilakukan lebih mudah untuk dikontrol.

4. Apakah aplikasi yang dibangun dapat memudahkan dalam membuat laporan pemesanan produk ?

Jawaban:

Laporan lebih mudah untuk dibuat sehingga mudah untuk dicek dan mengurangi kesalahan yang terjadi.

5. Dengan mengguanakan aplikasi ini, apa dalam pemasaran produk memiliki waktu yang lebih efektif?

   Jawaban:

   Ya, karena pemesaran produk dapat dipasarkan dalam waktu 24jam dikarenakan secara online.

4.2.3.3. Kesimpulan Pengujian Beta

Berdasarkan hasil prosentase diatas yang didapatkan dari pengujian betha, maka sistem penerapan *e-commerce* untuk sistem pemesanan ini sudah sesuai dengan tujuan yaitu dapat membantu mempromosikan produk, mengelola data pemesanan dan memudahkan pembuatan laporan pada Toko Musik Hafidz Pro Ard.