BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

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

Untuk mendukung aplikasi yang akan diterapkan pada lingkungan implementasi, maka dalam hal ini menggunakan perangkat keras dan perangkat lunak yang menunjang dalam membangun aplikasi *E-Commerce* di Toko Amalinda Collection.

4.1.1 Perangkat Keras yang digunakan

Spesifikasi perangkat keras yang digunakan dalam membangun aplikasi *E-Commerce* di Amalinda Collection adalah seperti tercantum pada tabel 4.1 sebagai berikut :

Tabel 4.1 Perangkat keras yang digunakan

Perangkat	Spesifikasi
Processor	Intel Pentium T4400 (2.2 Ghz, 800 MHz FSB)
RAM	1Gb
Hardisk	160 Gb
Monitor	Resolusi 1280 x 768
VGA	Intel GMA 4500M

4.1.2 Perangkat Lunak yang digunakan

Spesifikasi perangkat lunak yang digunakan dalam membangun aplikasi *E-Commerce* di Amalinda Collection adalah sebagai berikut :

Tabel 4.2 Perangkat Lunak yang digunakan

Perangkat Lunak	Keterangan
Sistem Operasi	Windows *7* Ultimate
Bahasa Pemrograman	PHP
DBMS	MySQL 5.0.27
Web browser	*Mozilla firefox 3.5.5*
Code editor	Adobe Dreamweaver CS5
DFD modeler	Microsoft Visio 2007

4.2 Implementasi Database

Pembangunan *database* dilakukan dengan menggunakan aplikasi *DBMS MySQL*. Implementasi *database* dalam bahasa *SQL* adalah sebagai berikut :

1. Tabel admin

```
CREATE TABLE `t_admin` (
  `id_admin`        int AUTO_INCREMENT NOT NULL,
  `nama_admin`      varchar(50) NOT NULL,
  `username`        varchar(10) NOT NULL,
  `password`        varchar(41) NOT NULL,
  `status_admin`    int NOT NULL,
  `id_kota`         int NOT NULL,
  `alamat_lengkap`  text NOT NULL,
  `kodepos`         char(5) NOT NULL,
  `telepon`         varchar(10) NOT NULL,
  `mobile_phone`    varchar(12) NOT NULL,
  `email`           varchar(50),
  /* Keys */
  PRIMARY KEY (`id_admin`),
  /* Foreign keys */
  CONSTRAINT `FKkotaadmin`
    FOREIGN KEY (`id_kota`)
    REFERENCES `t_kota`(`id_kota`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
CREATE INDEX `FKkotaadmin`
  ON `t_admin`
```

```
(`id_kota`); Tabel bank
CREATE TABLE IF NOT EXISTS `bank` (
  `id_bank` int(10) NOT NULL auto_increment,
  `nama_bank` varchar(255) NOT NULL,
  PRIMARY KEY  (`id_bank`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

2. Tabel biaya pengiriman

```
CREATE TABLE `t_biaya_pengiriman` (
  `id_biaya`           int AUTO_INCREMENT NOT NULL,
  `id_paket`           int NOT NULL,
  `id_kota`            int NOT NULL,
  `lama_pengiriman`    varchar(10) NOT NULL,
  `ongkos_pengiriman`  int NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_biaya`),
  /* Foreign keys */
  CONSTRAINT `FKkotabiaya`
    FOREIGN KEY (`id_kota`)
    REFERENCES `t_kota`(`id_kota`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT,
  CONSTRAINT `FKpaketbiaya`
    FOREIGN KEY (`id_paket`)
    REFERENCES `t_paket_pengiriman`(`id_paket`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
CREATE INDEX `FKkotabiaya`
  ON `t_biaya_pengiriman`
  (`id_kota`);
CREATE INDEX `FKpaketbiaya`
  ON `t_biaya_pengiriman`
  (`id_paket`);
```

3. Tabel kontak

```
CREATE TABLE `t_kontak` (
  `id_kontak`   int AUTO_INCREMENT NOT NULL,
  `nama`        varchar(50) NOT NULL,
  `email`       varchar(50) NOT NULL,
  `perusahaan`  varchar(50) NOT NULL,
  `telepon`     varchar(12) NOT NULL,
  `pesan`       text NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_kontak`)
) ENGINE = InnoDB;
```

4. Tabel kota

```
CREATE TABLE `t_kota` (
  `id_kota`      int AUTO_INCREMENT NOT NULL,
  `id_provinsi`  int NOT NULL,
  `nama_kota`    varchar(100) NOT NULL,
```

```
  /* Keys */
  PRIMARY KEY (`id_kota`),
  /* Foreign keys */
  CONSTRAINT `FKprovinsi`
    FOREIGN KEY (`id_provinsi`)
    REFERENCES `t_provinsi`(`id_provinsi`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
CREATE INDEX `FKprovinsi`
  ON `t_kota`
  (`id_provinsi`);
```

5. Tabel layanan pengiriman

```
CREATE TABLE `t_layanan_pengiriman` (
  `id_layanan`       int AUTO_INCREMENT NOT NULL,
  `nama_layanan`     varchar(20) NOT NULL,
  `alamat_layanan`   text NOT NULL,
  `telepon_layanan`  varchar(12) NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_layanan`)
) ENGINE = InnoDB;
```

6. Tabel member

```
CREATE TABLE `t_member` (
  `id_member`       int AUTO_INCREMENT NOT NULL,
  `email`           varchar(50) NOT NULL,
  `password`        varchar(41) NOT NULL,
  `nama_depan`      varchar(25) NOT NULL,
  `nama_belakang`   varchar(25) NOT NULL,
  `jenis_kelamin`   varchar(15),
  `id_kota`         int NOT NULL,
  `alamat_lengkap`  text NOT NULL,
  `kodepos`         char(5) NOT NULL,
  `telepon`         varchar(10) NOT NULL,
  `mobile_phone`    varchar(12) NOT NULL,
  `status`          int NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_member`),
  /* Foreign keys */
  CONSTRAINT `FKkotamember`
    FOREIGN KEY (`id_kota`)
    REFERENCES `t_kota`(`id_kota`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;

CREATE INDEX `FKkotamember`
  ON `t_member`
  (`id_kota`);
```

7. Tabel order

```
CREATE TABLE `t_order` (
  `id_order`            int AUTO_INCREMENT NOT NULL,
  `id_member`           int NOT NULL,
  `invoice_order`       int NOT NULL,
  `tanggal_pesan`       date NOT NULL,
  `tanggal_bayar`       date,
  `tanggal_kirim`       date NOT NULL,
  `tanggal_terkirim`    date NOT NULL,
  `tanggal_expired`     date NOT NULL,
  `total_order`         int NOT NULL,
  `nama_penerima`       varchar(5) NOT NULL,
  `alamat_penerima`     text,
  `telepon_penerima`    varchar(12),
  `handphone_penerima`  varchar(12),
  `kodepos_penerima`    varchar(5),
  `id_kota`             int,
  `biaya_pengiriman`    int NOT NULL,
  `jenis_pembayaran`    varchar(10) NOT NULL,
  `order_status`        int NOT NULL,
  `status_bayar`        int(1) NOT NULL,
  `no_resi`             varchar(30) NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_order`),
  /* Foreign keys */
  CONSTRAINT `FKordermember`
    FOREIGN KEY (`id_member`)
    REFERENCES `t_member`(`id_member`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
CREATE INDEX `FKordermember`
  ON `t_order`
  (`id_member`);
```

8. Tabel order detail

```
CREATE TABLE `t_order_detail` (
  `id_detail`            int AUTO_INCREMENT NOT NULL,
  `id_order`             int NOT NULL,
  `id_stok`              int NOT NULL,
  `jumlah_order`         int NOT NULL,
  `harga_produk_pesan`   int NOT NULL,
  `diskon_produk_pesan`  int NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_detail`),
  /* Foreign keys */
  CONSTRAINT `FKoderstok`
    FOREIGN KEY (`id_stok`)
    REFERENCES `t_produk_stok`(`id_stok`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT,
  CONSTRAINT `FKorderdetail`
    FOREIGN KEY (`id_order`)
```

```
    REFERENCES `t_order`(`id_order`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
CREATE INDEX `FKoderstok`
  ON `t_order_detail`
  (`id_stok`);
CREATE INDEX `FKorderdetail`
  ON `t_order_detail`
  (`id_order`);
```

9. Tabel order pesan

```
CREATE TABLE `t_order_pesan` (
  `id_pesan`       int AUTO_INCREMENT NOT NULL,
  `id_order`       int NOT NULL,
  `isi_pesan`      text NOT NULL,
  `tanggal_pesan`  date NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_pesan`),
  /* Foreign keys */
  CONSTRAINT `FKorderpesan`
    FOREIGN KEY (`id_pesan`)
    REFERENCES `t_order`(`id_order`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
```

10. Tabel paket pengiriman

```
CREATE TABLE `t_paket_pengiriman` (
  `id_paket`    int AUTO_INCREMENT NOT NULL,
  `id_layanan`  int NOT NULL,
  `nama_paket`  varchar(20) NOT NULL,
  `keterangan`  text NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_paket`),
  /* Foreign keys */
  CONSTRAINT `FKlayananpaket`
    FOREIGN KEY (`id_layanan`)
    REFERENCES `t_layanan_pengiriman`(`id_layanan`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
CREATE INDEX `FKlayananpaket`
  ON `t_paket_pengiriman`
  (`id_layanan`);
```

11. Tabel pembayaran

```
CREATE TABLE `t_pembayaran` (
  `id_pembayaran`      int AUTO_INCREMENT NOT NULL,
  `id_order`           int NOT NULL,
  `nama_bank_asal`     varchar(10),
  `nama_bank_tujuan`   varchar(10) NOT NULL,
  `no_rekening`        varchar(20) NOT NULL,
  `atas_nama`          varchar(50) NOT NULL,
  `jumlah_pembayaran`  int,
  /* Keys */
  PRIMARY KEY (`id_pembayaran`),
  /* Foreign keys */
  CONSTRAINT `FKpembayaranorder`
    FOREIGN KEY (`id_order`)
    REFERENCES `t_order`(`id_order`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
CREATE INDEX `FKpembayaranorder`
  ON `t_pembayaran`
  (`id_order`);
```

12. Tabel Produk

```
CREATE TABLE `t_produk` (
  `id_produk`          int AUTO_INCREMENT NOT NULL,
  `id_kategori`        int NOT NULL,
  `jenis_kelamin`      char NOT NULL,
  `nama_produk`        varchar(50) NOT NULL,
  `ringkasan_produk`   varchar(20) NOT NULL,
  `deskripsi_produk`   text NOT NULL,
  `harga_produk`       int NOT NULL,
  `diskon_produk`      float NOT NULL,
  `berat_produk`       float NOT NULL,
  `status_produk`      int NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_produk`)
) ENGINE = InnoDB;
```

13. Tabel produk gambar

```
CREATE TABLE `t_produk_gambar` (
  `id_gambar`    int AUTO_INCREMENT NOT NULL,
  `id_produk`    int NOT NULL,
  `nama_gambar`  varchar(100) NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_gambar`),
  /* Foreign keys */
  CONSTRAINT `FKprodukgambar`
    FOREIGN KEY (`id_produk`)
    REFERENCES `t_produk`(`id_produk`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
```

```
) ENGINE = InnoDB;
CREATE INDEX `FKprodukgambar`
  ON `t_produk_gambar`
  (`id_produk`);
```

14. Tabel produk kategori

```
CREATE TABLE `t_produk_kategori` (
  `id_kategori`    int AUTO_INCREMENT NOT NULL,
  `nama_kategori`  varchar(25) NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_kategori`)
) ENGINE = InnoDB;
```

15. Tabel retur

```
CREATE TABLE `t_produk_retur` (
  `id_retur`       int AUTO_INCREMENT NOT NULL,
  `id_order`       int NOT NULL,
  `status_retur`   int NOT NULL,
  `tanggal_retur`  date NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_retur`),
  /* Foreign keys */
  CONSTRAINT `FKorderretur`
    FOREIGN KEY (`id_order`)
    REFERENCES `t_order`(`id_order`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
CREATE INDEX `FKorderretur`
  ON `t_produk_retur`
  (`id_order`);
```

16. Tabel retur detail

```
CREATE TABLE `t_produk_retur_detail` (
  `id_retur_detail`  int AUTO_INCREMENT NOT NULL,
  `id_retur`         int NOT NULL,
  `id_stok`          int NOT NULL,
  `jumlah_retur`     int NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_retur_detail`),
  /* Foreign keys */
  CONSTRAINT `FKdetailretur`
    FOREIGN KEY (`id_retur`)
    REFERENCES `t_produk_retur`(`id_retur`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT,
  CONSTRAINT `FKstokdetail`
    FOREIGN KEY (`id_stok`)
    REFERENCES `t_produk_stok`(`id_stok`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
```

```
) ENGINE = InnoDB;
CREATE INDEX `FKdetailretur`
  ON `t_produk_retur_detail`
  (`id_retur`);
CREATE INDEX `FKstokdetail`
  ON `t_produk_retur_detail`
  (`id_stok`);
```

17. Tabel stok

```
CREATE TABLE `t_produk_stok` (
  `id_stok`      int AUTO_INCREMENT NOT NULL,
  `id_produk`    int NOT NULL,
  `id_ukuran`    int NOT NULL,
  `id_warna`     int NOT NULL,
  `jumlah_stok`  int NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_stok`),
  /* Foreign keys */
  CONSTRAINT `FKprodukstok`
    FOREIGN KEY (`id_produk`)
    REFERENCES `t_produk`(`id_produk`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT,
  CONSTRAINT `FKukuranstok`
    FOREIGN KEY (`id_ukuran`)
    REFERENCES `t_produk_ukuran`(`id_ukuran`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT,
  CONSTRAINT `FKwarnastok`
    FOREIGN KEY (`id_warna`)
    REFERENCES `t_warna`(`id_warna`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
CREATE INDEX `FKprodukstok`
  ON `t_produk_stok`
  (`id_produk`);
CREATE INDEX `FKukuranstok`
  ON `t_produk_stok`
  (`id_ukuran`);
CREATE INDEX `FKwarnastok`
  ON `t_produk_stok`
  (`id_warna`);
```

18. Tabel ukuran

```
CREATE TABLE `t_produk_ukuran` (
  `id_ukuran`    int AUTO_INCREMENT NOT NULL,
  `nama_ukuran`  varchar(5) NOT NULL,
  `id_kategori`  int,
  /* Keys */
  PRIMARY KEY (`id_ukuran`),
  /* Foreign keys */
  CONSTRAINT `FKkategoriukuran`
```

```
    FOREIGN KEY (`id_kategori`)
    REFERENCES `t_produk_kategori`(`id_kategori`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT
) ENGINE = InnoDB;
CREATE INDEX `FKkategoriukuran`
  ON `t_produk_ukuran`
  (`id_kategori`);
```

19. Tabel provinsi

```
CREATE TABLE `t_provinsi` (
  `id_provinsi`    int AUTO_INCREMENT NOT NULL,
  `nama_provinsi`  varchar(100) NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_provinsi`)
) ENGINE = InnoDB;
```

20. Tabel warna

```
CREATE TABLE `t_warna` (
  `id_warna`    int AUTO_INCREMENT NOT NULL,
  `nama_warna`  varchar(20) NOT NULL,
  /* Keys */
  PRIMARY KEY (`id_warna`)
) ENGINE = InnoDB;
```

21. Tabel temp

```
CREATE TABLE `temp` (
  `id_temp`     int AUTO_INCREMENT NOT NULL,
  `id_session`  char(32),
  `id_stok`     int,
  `qty`         int,
  `id_member`   int,
  /* Keys */
  PRIMARY KEY (`id_temp`)
) ENGINE = InnoDB
  CHECKSUM = 1
  ROW_FORMAT = DYNAMIC
  DELAY_KEY_WRITE = 1;
```

4.3 Implementasi Antar Muka

Implementasi antar muka dilakukan dengan setiap tampilan aplikasi yang dibangun dan pengkodeannya dalam bentuk *file*. Berikut ini adalah implementasi antarmuka untuk pengguna. Adapun tampilan implementasi aplikasi *E-Commerce* di Amalinda Collection akan dilampirkan pada halaman lampiran.

Tabel 4.3 Implementasi antar muka

Menu	Deskripsi	Nama File
Login administrator dan member	Digunakan sebagai halaman login administrator dan member	index.php
Ganti *password* administrator	Digunakan jika administrator ingin mengubah *password*	admin.php
Ganti password member	Digunakan jika member ingin mengubah *password*	member_password.php
Kategori	Digunakan untuk mencari, menambah, mengubah dan menghapus data kategori	kategori.php
Warna	Digunakan untuk mencari, menambah, mengubah dan menghapus data warna	warna.php
Detail Produk	Digunakan untuk mencari, menambah, mengubah dan menghapus data detail barang	detail_barang.php
member	Digunakan untuk mencari dan melihat data member	member.php
Provinsi	Digunakan untuk mencari, menambah, mengubah dan menghapus data provinsi	provinsi.php
Kota	Digunakan untuk mencari, menambah, mengubah dan menghapus data kota	kota.php

Layanan Pengiriman	Digunakan untuk menambah, mengubah dan menghapus data jasa pengiriman	layanan_pengiriman.php
Paket Pengiriman	Digunakan untuk menambah, mengubah dan menghapus data detail jasa pengiriman	Paket_pengiriman.php
Registrasi member	Digunakan untuk melakukan pendaftaran member	daftar_member.php
Lupa password	Digunakan untuk melakukan fasilitas lupa password	
Produk kami	Digunakan untuk melihat seluruh data barang yang ada	produk_kami.php
Cara registrasi	Digunakan untuk melihat cara registrasi member	cara_registrasi.php
Cara pemesanan	Digunakan untuk melihat cara pemesanan barang	cara_pemesanan.php
Cara pembayaran	Digunakan untuk melihat cara pembayaran	cara_pembayaran.php
Add to cart	Digunakan untuk melakukan pemesanan barang	addtocart2.php
Keranjang belanja	Digunakan untuk melihat, mengubah dan menghapus data pemesanan barang	shopping_cart.php
Detail Pembelian	Digunakan untuk melihat dan melakukan pembayaran	detail_pembayaran.php
Histori order	Digunakan untuk melihat data histori order	daftar_pemesanan.php

Retur member	Digunakan untuk mengirim data retur.	retur.php
Member profil	Digunakan untuk melihat dan mengubah data member	member_profile.php
Order	Digunakan untuk melihat dan mengolah data order	penjualan.php
Detail order administrator dan detail order member	Digunakan untuk melihat detail order.	detail_order.php
Laporan member	Digunakan untuk melihat dan mencari laporan data member	laporan_datamember.php
Laporan dipesan	Digunakan untuk melihat dan mencari laporan data barang dipesan	laporan_barangdipesan.php
Laporan barang dikirim	Digunakan untuk melihat dan mencari laporan data barang yang sedang dikirim	laporan_barangdikirim.php
Laporan Penjualan	Digunakan untuk melihat dan mencari laporan data penjualan	laporan_penjualan.php
Laporan retur	Digunakan untuk melihat dan mencari laporan data retur	laporan_dataretur.php
Restore dan backup	Digunakan untuk restore dan backup database	restore_backup.php
Logout	Digunakan user (administrator dan member) untuk dapat keluar dari aplikasi	logout.php

4.4 Pengujian Perangkat Lunak

Pengujian adalah proses untuk menemukan *error* pada perangkat lunak sebelum di-*delivery* kepada pengguna. Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunakdan merefresentasikan spesifikasi, design dan pengkodean. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

Metode pengujian pada perangkat lunak ini menggunakan metode pengujian *black box.* Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

4.5 Pengujian Alpha

Pengujian alpha untuk membangun aplikasi *E-Commerce* pada toko Amalinda Collection ini dilakukan dengan menggunakan metode pengujian *black box.* Untuk pengujian alpha ini yaitu pada pengujian sebagai pengguna.

4.5.1 Skenario Pengujian Alpha

Berikut adalah skenario pengujian alpha aplikasi pada *member* dapat dilihat pada tabel 4.4 dibawah ini :

Tabel 4.4 Skenario pengujian aplikasi *member*

Kelas Uji	Butir Uji	Jenis Pengujian
Registrasi member	Menampilkan *form* user baru	*Black Box*
	Isi *form* pendaftaran	*Black Box*
	Simpan data baru	*Black Box*
	Mengirim Konfirmasi ke email	*Black Box*
Lupa *password*	Menampilkan *form* lupa	*Black Box*

	password	
	Mengirim informasi password ke email member	*Black Box*
Login	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Logout	Menghapus *session user*	*Black Box*
Beranda	Menampilakan data barang	*Black Box*
	Menampilkan data detail barang	*Black Box*
	Menampilkan rating barang	*Black Box*
Testimoni	Mengirim testimoni	*Black Box*
	Menampilakan testimoni	*Black Box*
Tentang kami	Menampilakan data kontak kami	*Black Box*
Cara pemesanan	Menampilkan cara pemesanan barang	*Black Box*
Cara Pembayaran	Menampilkan cara pembayaran	*Black Box*
Histori order	Menampilkan data detail histori order	*Black Box*
Retur	Memasukkan no order yang akan direturkan	*Black Box*
Member profil	Menampilkan data member	*Black Box*
	Mengubah data member	*Black Box*
	Menyimpan perubahan	*Black Box*
Ganti *password*	Menampilkan *form* ubah *password*	*Black Box*
	Verifikasi password lama	*Black Box*
	Verifikasi password baru	*Black Box*
	Verifikasi konfirmasi password	*Black Box*

Berikut adalah skenario pengujian alpha aplikasi pada *member* dapat dilihat pada tabel 4.5 dibawah ini :

Tabel 4.5 Skenario pengujian aplikasi *administrator*

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Mengisi data login	*Black Box*
	Verifikasi username	*Black Box*
	Verifikasi password	*Black Box*
Halaman admin	Menampilkan halaman utama admin	*Black Box*
Data produk kategori	Menampilkan data kategori	*Black Box*
	Menambah data kategori	*Black Box*
	Mengubah data kategori	*Black Box*
	Mencari data kategori	*Black Box*
Data produk	Menampilkan data produk	*Black Box*
	Menambah data produk	*Black Box*
	Mengubah data produk	*Black Box*
	Mencari data produk	*Black Box*
Data akun member	Menampilkan data member	*Black Box*
Data Provinsi	Menampilkan data provinsi	*Black Box*
	Menambah data provinsi	*Black Box*
	Mengubah data provinsi	*Black Box*
	Mencari data provinsi	*Black Box*
Data kota	Menampilkan data kota	*Black Box*
	Menambah data kota	*Black Box*
	Mengubah data kota	*Black Box*
	Mencari data kota	*Black Box*
Data layanan pengiriman	Menampilkan data layanan pengiriman	*Black Box*
	Menambah data layanan pengiriman	*Black Box*
	Mengubah data layanan pengiriman	*Black Box*
	Mencari data layanan	*Black Box*

	pengiriman	
Data kontak kami	Menampilkan data kontak kami	*Black Box*
	Menambah data kontak kami	*Black Box*
	Mengubah data kontak kami	*Black Box*
	Mencari data kontak kami	*Black Box*
Data pengolahan order	Menampilkan data order	*Black Box*
	Mengelola data transaksi	*Black Box*
	Print data pengiriman	*Black Box*
	Menambah nomor resi pengiriman	*Black Box*
Laporan barang dipesan	Menampilkan laporan data barang dipesan	*Black Box*
	Mencari laporan data barang dipesan	*Black Box*
Laporan barang dikirim	Menampilkan laporan data barang dikirm	*Black Box*
	Mencari laporan barang dikirim	*Black Box*
Laporan penjualan	Menampilkan laporan data penjualan	*Black Box*
	Mencari laporan data penjualan	*Black Box*
Laporan data retur	Menampilkan laporan data retur barang	*Black Box*
	Mencari laporan data retur barang	*Black Box*
Restore dan *Backup*	Menampilkan *form* restore dan *backup*	*Black Box*
	Restore database	*Black Box*
	Backup database	*Black Box*
Ganti password admin	Menampilkan *form* ubah password admin	*Black Box*
	Verifikasi password lama	*Black Box*
	Verifikasi password baru	*Black Box*
	Verifikasi konfirmasi password	*Black Box*

Logout	Menghapus *session user*	*Black Box*
Data paket pengiriman	Menampilkan data paket pengiriman	*Black Box*
	Menambah data paket pengiriman	*Black Box*
	Mengubah data paket pengiriman	*Black Box*
	Mencari data paket pengiriman	*Black Box*
Data biaya pengiriman	Menampilkan data biaya pengiriman	*Black Box*
	Menambah data biaya pengiriman	*Black Box*
	Mengubah data biaya pengiriman	*Black Box*
	Mencari data biaya pengiriman	*Black Box*

4.5.2 Kasus dan Hasil Pengujian

4.5.2.1 Pengujian Registrasi member

Berikut dapat dilihat pengujian dari registrasi *member* pada tabel 4.6 dibawah ini :

Tabel 4.6 Pengujian Registrasi *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu *registrasi member*	Tampil *forn registrasi member*	Dapat menampilkan *form registrasi member*	[√] Diterima [ ] Ditolak
Memasukkan data pendaftaran, contoh masukan : Email : ariesafriandayani@yahoo.com	Jika salah satu field tidak terisi atau salah dalam	Dapat memunculkan pesan kesalahan bahwa field tersebut harus	[√] Diterima [ ] Ditolak

Password :testo12 Confirm Password :testo12 Nama Lengkap : aries Jenis Kelamin : Laki-laki Email : ariesafriandayani@yahoo.com Tanggal Lahir : 01-04-1988 Alamat : cicaheum Provinsi : Jawa Barat Kota : Bandun Kode Pos :14325	pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus diisi dengan benar.	terisi jika salah satu field tidak terisi atau salah dalam pengisian.	
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 "Berhasil".	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	Tampil pesan kesalahan	Dapat menampilkan pesan kesalahan "*field* ini harus diisi"	[√] Diterima [ ] Ditolak

4.5.2.2 Pengujian Login Member

Berikut dapat dilihat pengujian login *member* pada tabel 4.7 dibawah ini :

Tabel 4.7 Pengujian login *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan *email* : ariesafriandayani	Mengisikan data alamat *email* pada *field* alamat *email*. Jika salah maka akan muncul pesan "invalid alamat email"	Dapat mengisi *field email* dan muncul pesan "*email* dan *password* anda salah. Silahkan Ulangi !!" jika salah dalam pengisian data	[√] Diterima [ ] Ditolak
Memasukan *Password* : testo12	Mengisikan data *password* pada *field password*. Jika salah maka akan muncul pesan "diantara 6 sampai 11 karakter diperbolehkan"	Dapat mengisi *field password* dan muncul pesan "*email* dan *password* anda salah. Silahkan Ulangi !!"" jika salah dalam pengisian data	[√] Diterima [ ] Ditolak
Klik tombol login	Data *email* dan data *password* dicari	Tombol login dapat berfungsi	[√] Diterima [ ] Ditolak
Memasukan data *email atau password yang tidak terdaftar*	Tidak dapat login	Muncul pesan "*email* dan *password* anda salah. Silahkan Ulangi !!"	[√] Diterima [ ] Ditolak
Mengosongkan data *email* atau *password*	Tidak dapat login	Dapat menampilkan pesan kesalahan "*email* dan *password* anda	[√] Diterima [ ] Ditolak

		salah. Silahkan Ulangi !!”	

4.5.2.3 Pengujian Lupa *Password*

Berikut dapat dilihat pengujian dari lupa *password* pada tabel 4.8 dibawah ini :

Tabel 4.8 Pengujian Lupa *Password*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan alamat *email* : ariesafriandayani@yahoo.com	Mengisikan alamat *email* pada *field* alamat *email.*	Dapat mengisi *field* alamat *email* dan mengirimkan informasi *password* pada email yang dimasukan	[√] Diterima [ ] Ditolak
Klik tombol kirim	Data *password* dikirimkan ke alamat *email.*	Tombol login dapat berfungsi	[√] Diterima [ ] Ditolak

4.5.2.4 Pengujian *Logout*

Berikut dapat dilihat pengujian dari *logout* pada tabel 4.9 dibawah ini :

Tabel 4.9 Pengujian *logout*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik *logout*	Dapat keluar dan menghapus *session* pengguna	*User* keluar darin aplikasi	[√] Diterima [ ] Ditolak

4.5.2.5 Pengujian Ubah profile *member*

Berikut dapat dilihat pengujian dari ubah profil member pada tabel 4.10 dibawah ini :

Tabel 4.10 Pengujian ubah profil *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik update profile member	Menampilkan *form* profil *member*	Dapat menampilkan *form* profil *member*	[√] Diterima [ ] Ditolak
Klik Edit	Profil menampilkan *form* ubah profil	Dapat menampilkan *form* ubah profil	[√] Diterima [ ] Ditolak
Data profil yang akan diubah	Tercantum pada text box data yang akan di ubah	Dapat mengisi *field* data yang akan diubah dengan benar	[√] Diterima [ ] Ditolak
Klik simpan	Data berhasil diubah di *database*	Tampil pesan "Update berhasil"	[√] Diterima [ ] Ditolak
Mengosongkan salah satu *field*	Menampilkan pesan kesalahan bahwa *field* tersebut harus terisi	Dapat menampilkan pesan kesalahan bahwa *field* tersebut harus terisi.	[√] Diterima [ ] Ditolak

4.5.2.6 Pengujian ganti *password member*

Berikut dapat dilihat pengujian dari ganti *password member* pada tabel 4.11 dibawah ini :

Tabel 4.11 Pengujian ganti *password member*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data yang benar	*Password* baru berhasil disimpan di *database*	Dapat menyimpan data *password* yang baru dengan muncul pesan "Update berhasil"	[√] Diterima [ ] Ditolak
Mengosongkan salah satu field	Muncul pesan kesalahan *field* tersebut harus diiisi	Dapat menampilkan pesan kesalahan "*field* ini tidak boleh kosong" bila ada *field* yang belum diisi	[√] Diterima [ ] Ditolak
Data *password* lama salah	Muncul pesan "*password* salah"	Dapat menampilkan pesan kesalahan "*Password* lama salah"	[√] Diterima [ ] Ditolak
Data *password* baru dan konfirmasi *password* baru tidak sama	Muncul pesan "*Password* baru dan *konfirmasi password* harus sama"	Dapat menampilkan pesan "*Password* baru dan *konfirmasi password* harus sama"	[√] Diterima [ ] Ditolak
Mengosongkan semua *field*	Muncul pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan *field* harus diisi	[√] Diterima [ ] Ditolak

4.5.2.7 Pengujian Beranda

Berikut dapat dilihat pengujian dari menu beranda pada tabel 4.12 dibawah ini :

Tabel 4.12 Pengujian beranda

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu beranda	Menampilkan data produk kami yang ada di *database*	Dapat menampilkan data produk kami yang ada di *database*	[√] Diterima [ ] Ditolak
Klik gambar barang	Menampilkan detail barang	Dapat menampilkan detail barang	[√] Diterima [ ] Ditolak
Klik beli	Menuju ke halaman addtocart	Dapat menampilkan halaman addtocart	[√] Diterima [ ] Ditolak

4.5.2.8 Pengujian Testimoni

Berikut dapa tdilihat pengujian dari menu berita pada tabel 4.13 dibawah ini :

Tabel 4.13 Pengujian Testimoni

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik Testimoni	Menampilkan *form* testimoni	Dapat menampilkan *form* Testimoni	[√] Diterima [ ] Ditolak
Memasukkan data testimoni	Mengisikan data testimoni	Dapat mengisikan dapat testimoni	[√] Diterima [ ] Ditolak
Klik kirim	Data testimoni terkirim	Tombol kirim berfungsi	[√] Diterima [ ] Ditolak

Mengosongkan salah satu field	Muncul pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan "*field* ini tidak boleh kosong" bila ada *field* yang belum diisi	[√] Diterima [ ] Ditolak
Mengosongkan semua field	Muncul pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan *field* harus diisi	[√] Diterima [ ] Ditolak
Klik tombol bersihkan	Data testimoni dibersihkan	Tombol bersihkan berfungsi	[√] Diterima [ ] Ditolak

4.5.2.9 Pengujian menu tentang kami

Berikut dapat dilihat pengujian dari menu tentang kami pada tabel 4.14 dibawah ini :

Tabel 4.14 Pengujian menu tentang kami

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tentang kami	Menampilkan *form* tentang kami	Menampilkan *form* tentang kami	[√] Diterima [ ] Ditolak

4.5.2.10 Pengujian menu Keranjang

Berikut dapat dilihat pengujian dari menu Keranjang pada tabel 4.15 dibawah ini :

Tabel 4.15 Pengujian menu Keranjang

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu	Menampilkan	Dapat	[√] Diterima

Keranjang	*form* Keranjang	menampilkan *form* Keranjang	[ ] Ditolak
Menampilkan data jumlah pesanan	Tercantum pada *text box* data yang akan diubah	Dapat mengisi data jumlah dengan benar	[√] Diterima [ ] Ditolak
Klik tombol ubah qty	Data jumlah yang diubah tersimpan di *database*	Dapat menampilkan perubahan data jumlah	[√] Diterima [ ] Ditolak
Klik tombol hapus	Data yang dihapus terhapus dari Keranjang	Dapat menghapus data pada Keranjang	[√] Diterima [ ] Ditolak
Klik tombol belanja lagi	Menampilkan *form* produk kami	Dapat menampilkan *form* produk kami	[√] Diterima [ ] Ditolak
Klik tombol next	Menampilkan *form* data pengiriman	Dapat menampilkan *form* data pengiriman	[√] Diterima [ ] Ditolak

4.5.2.11 Pengujian *form* data pengiriman

Berikut dapat dilihat dari *form* pengiriman pada tabel 4.16 dibawah ini :

Tabel 4.16 Pengujian *form* data pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data yang benar lalu meng serta memilih jasa pengiriman yang disediakan	Data pengiriman baru, akan disimpan di dalam *database*	Data pengiriman baru dapat disimpan di dalam *database*	[√] Diterima [ ] Ditolak
Mengosongkan salah satu *field*	Menampilkan pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan ”*field* ini harus diisi”	[√] Diterima [ ] Ditolak

Klik back	Menampilkan *form* sebelumnya yaitu *form* Keranjang	Dapat menampilkan *form* sebelumnya yaitu *form* Keranjang	[√] Diterima [ ] Ditolak
Klik tombol lanjutkan ke pembayaran	Menampilkan *form* detail pembelian	Dapat menampilkan *form* detail pembelian	[√] Diterima [ ] Ditolak

4.5.2.12 Pengujian *form* detail pembelian

Berikut dapat dilihat pengujian dari *form* detail pembelian pada tabel 4.17 dibawah ini :

Tabel 4.17 Pengujian *form* detail pembelian

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang akan dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik transfer	Menampilkan *form* pemberitahuan jenis pembayaran transfer	Dapat menampilkan *form* pemberitahuan jenis pembayaran transfer	[√] Diterima [ ] Ditolak
Klik Paypal	Menuju ke halaman paypal	Dapat melanjutkan proses ke halaman paypal	[√] Diterima [ ] Ditolak
Klik simpan dihistory order	Menyimpan data transaksi	Dapat menyimpan data transaksi	[√] Diterima [ ] Ditolak

4.5.2.13 Pengujian menu histori order

Berikut dapat dilihat pengujian dari *form* histori order pada tabel 4.18 dibawah ini :

Tabel 4.18 Pengujian *form* histori order

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik menu histori order	Menampilkan histori order	Dapat menampilkan histori order	[√] Diterima [ ] Ditolak
Klik detail	Menampilkan *form* detail order	Dapat menampilkan *form* detail order	[√] Diterima [ ] Ditolak

4.5.2.14 Pengujian detail order

Berikut dapat dilihat pengujian dari *form* detail order pada tabel 4.19 dibawah ini :

Tabel 4.19 Pengujian *form* detail order

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik no resi	Menuju halaman *web* jasa pengiriman	Dapat menuju halaman *web* jasa pengiriman	[√] Diterima [ ] Ditolak

4.5.2.15 Pengujian *form* Retur

Berikut dapat dilihat pengujian dari *form* retur pada tabel 4.20 dibawah ini :

Tabel 4.20 Pengujian *form* retur

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan no order yang akan direturkan	Mengisikan no order yang akan direturkan	Dapat mengisikan no order yang akan direturkan	[√] Diterima [ ] Ditolak
Klik tombol cari	Menampilkan data barang yang sudah memenuhi syarat untuk direturkan. Jika status barang belum diterima atau barang sudah lebih dari 7 hari dari diterima, maka akan muncul pesan "Maaf no order Anda tidak ditemukan. Pastikan no order anda benar dan barang sudah diterima serta status barang tidak lebih dari 7 hari dari waktu barang diterima"	Dapat menampilkan data barang yang sudah memenuhi syarat untuk direturkan. Jika status barang belum diterima atau barang sudah lebih dari 7 hari dari diterima, maka akan muncul pesan "Maaf no order Anda tidak ditemukan. Pastikan no order anda benar dan barang sudah diterima serta status barang tidak lebih dari 7 hari dari waktu barang diterima"	[√] Diterima [ ] Ditolak
Klik *link* histori retur	Menampilkan histori retur	Dapat menampilkan histori retur	[√] Diterima [ ] Ditolak

4.5.2.16 Pengujian Login Administrator

Berikut dapat dilihat pengujian login administrator pada tabel 4.21 dibawah ini:

Tabel 4.21 Pengujian *login administrator*

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
username : aries	Tercantum pada text box *username*	Dapat mengisi field *username* dengan benar	[√] Diterima [ ] Ditolak
Password : admin	Tercantum pada text box *password*	Dapat mengisi field *password* dengan benar	[√] Diterima [ ] Ditolak
Klik tombol login	Data *username* dan data *password* dicari	Tombol login dapat berfungsi	[√] Diterima [ ] Ditolak
Memasukan data *username* atau *password* yang tidak terdaftar	Tidak dapat login	Muncul pesan "Username tidak teridentifikasi."	[√] Diterima [ ] Ditolak
Tidak memasukan data *username*	Tidak dapat login dan muncul pesan "Silahkan masukan Username"	Dapat menampilkan pesan kesalahan "Silahkan masukan Username"	[√] Diterima [ ] Ditolak
Tidak memasukan data *password*	Tidak dapat login dan muncul pesan "Silahkan masukan Password"	Dapat menampilkan pesan kesalahan "Silahkan masukan Password"	[√] Diterima [ ] Ditolak

4.5.2.17 Pengujian ganti *password* Admin

Berikut dapat dilihat pengujian dari ganti *password* admin pada tabel 4.22 dibawah ini:

Tabel 4.22 Pengujian ganti *password* admin

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik ubah *password*	Menampilkan *form* ubah *password*	Dapat menampilkan *form* ubah *password*	[√] Diterima [ ] Ditolak
Memasukan data yang benar	*Password* baru berhasil disimpan di *database*	Dapat menyimpan data *password* yang baru dengan muncul pesan " Data berhasil diubah"	[√] Diterima [ ] Ditolak
Mengosongkan salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan "*field* ini harus diisi" bila ada field yang belum diisi	[√] Diterima [ ] Ditolak
Memasukan *password* yang salah	muncul pesan "password salah"	Dapat menampilkan pesan kesalahan "*Password* salah"	[√] Diterima [ ] Ditolak
Memasukan data *password* baru dan konfirmasi *password* baru yang tidak sama	Muncul pesan "password baru dan konfirmasi password harus sama"	Dapat menampilkan pesan "password baru dan konfirmasi password harus sama"	[√] Diterima [ ] Ditolak
Mengosongkan semua *field*	Muncul pesan kesalahan *field* harus diisi	Dapat menampilkan pesan kesalahan *field* harus diisi	[√] Diterima [ ] Ditolak

4.5.2.18 Pengujian data produk kategori

Berikut dapat dilihat pengujian dari data produk kategori pada tabel 4.23 dibawah ini:

Tabel 4.23 Pengujian data produk kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik produk kategori	Menampilkan data produk kategori	Dapat menampilkan data produk kategori	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah produk kategori	Dapat menampilkan *form* ubah kategori	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	menampilkan pesan kesalahan *field* tersebut harus diisi	Dapat menampilkan pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data harus diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Data dapat tersimpan kedalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.19 Pengujian tambah kategori

Berikut dapat dilihat pengujian dari tambah kategori pada tabel 4.24 dibawah ini:

Tabel 4.24 Pengujian tambah kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah kategori	Menampilkan *form* tambah kategori	Dapat menampilkan *form* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data kategori Contoh: Nama Kategori : kaos	Jika salah satu field tidak terisi, 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
Tidak mengisi nama kategori	Menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Masukan Kategori”	[√] Diterima [ ] Ditolak
Tidak mengisi nama kategori	Menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Masukan Kategori”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan	[√] Diterima [ ] Ditolak

		muncul pesan ”data berhasil di tambah”	
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.20 Pengujian data produk warna

Berikut dapat dilihat pengujian dari data produk warna pada tabel 4.25 dibawah ini:

Tabel 4.25 Pengujian data produk warna

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik data produk warna	Menampilkan data data produk warna	Dapat menampilkan data produk warna	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah data produk warna	Dapat menampilkan *form* data produk warna	[√] Diterima [ ] Ditolak
Tidak mengisi *field*	Menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”masukan nama warna”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di *database*	Data dapat diubah di *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.21 Pengujian tambah warna

Berikut dapat dilihat pengujian dari tambah warna pada tabel 4.26 dibawah ini:

Tabel 4.26 Pengujian tambah warna

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah warna	Menampilkan *form* tambah warna	Dapat menampilkan *form* tambah warna	[√] Diterima [ ] Ditolak
Mengisi warna barang Contoh: Warna barang : Hitam	Jika 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 field tidak terisi.	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil disimpan dalam *database*	Data warna akan tersimpan dalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.22 Pengujian data produk

Berikut dapat dilihat pengujian dari data produk pada tabel 4.27 dibawah ini:

Tabel 4.27 Pengujian lihat produk

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik data produk	Menampilkan data	Dapat	[√] Diterima

	produk	menampilkan data produk	[ ] Ditolak
Klik ubah	Menampilkan *form* ubah produk	Dapat menampilkan *form* ubah produk	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field* kategori	Menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Kategori belum diisi”	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field* kategori	Menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Kode barang belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah kedalam *database*	Data berhasil diubah kedalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.23 Pengujian tambah produk

Berikut dapat dilihat pengujian dari tambah produk pada tabel 4.28 dibawah ini:

Tabel 4.28 Pengujian tambah produk

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah produk	Menampilkan *form* tambah produk	Dapat menampilkan *form* tambah produk	[√] Diterima [ ] Ditolak
Mengisi data produk Contoh: kategori: kaos	Jika salah satu field tidak terisi, 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.	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database* dan muncul pesan ”data berhasil di tambah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.24 Pengujian produk

Berikut dapat dilihat pengujian dari detail produk pada tabel 4.29 dibawah ini:

Tabel 4.29 Pengujian produk

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik produk	Menampilkan data produk	Dapat menampilkan data produk	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah detail produk	Dapat menampilkan *form* ubah detail produk	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”data belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah ke *database*	Data dapat berhasil diubah ke *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.25 Pengujian tambah produk

Berikut dapat dilihat pengujian dari tambah produk pada tabel 4.31 dibawah ini:

Tabel 4.30 Pengujian tambah produk

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah produk	Menampilkan *form* tambah produk	Dapat menampilkan *form* tambah produk	[√] Diterima [ ] Ditolak
Mengisi data produk Contoh: Jumlah barang : 1 Ukuran : L Warna : Hitam Berat : 0,5 Harga : Rp 100000 Discount : 10% Deskripsi : Terbuat dari katun Gambar pertama : Gambar kedua : Gambar ketiga : Gambar keempat : Gambar kelima : Gambar keenam :	Jika salah satu field tidak terisi, 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.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data detail barang akan tersimpan didalam *database*	Data detail barang dapat tersimpan didalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali ke menu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.26 Pengujian data member

Berikut dapat dilihat pengujian dar data member pada tabel 4.31 dibawah ini:

Tabel 4.31 Pengujian lihat member

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik data member	Menampilkan data member	Dapat menampilkan data member	[√] Diterima [ ] Ditolak
Masukan nama pada *edit box* pencarian	Mengisi nama member yang akan dicari	Dapat mengisi nama member yang akan dicari	[√] Diterima [ ] Ditolak

4.5.2.27 Pengujian data provinsi

Berikut dapat dilihat pengujian dari data provinsi pada tabel 4.32 dibawah ini:

Tabel 4.32 Pengujian data provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik data provinsi	Menampilkan data provinsi	Dapat menampilkan data provinsi	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah provinsi	Dapat menampilkan *form* ubah provinsi	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan	[√] Diterima [ ] Ditolak

		pesan kesalahan ”Data belum diisi”	
Klik tombol simpan	Data berhasil diubah di *database*	Data dapat berhasil diubah di *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.28 Pengujian tambah provinsi

Berikut dapat dilihat pengujian dari tambah kota pada tabel 4.33 dibawah ini:

Tabel 4.33 Pengujian tambah provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah provinsi	Menampilkan *form* tambah provinsi	Dapat menampilkan *form* tambah provinsi	[√] Diterima [ ] Ditolak
Mengisi data provinsi Contoh: Nama Provinsi : Jawa Barat	Jika field tidak terisi, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika field tidak terisi.	[√] Diterima [ ] Ditolak
Tidak mengisi *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data provinsi akan tersimpan didalam *database*	Data provinsi akan tersimpan didalam *database.*	[√] Diterima [ ] Ditolak

Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.29 Pengujian data kota

Berikut dapat dilihat pengujian dari data kota pada tabel 4.34 dibawah ini:

Tabel 4.34 Pengujian data kota

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik data kota	Menampilkan data kota	Dapat menampilkan data kota	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah kota	Dapat menampilkan *form* ubah kota	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di *database*	Data dapat berhasil diubah di *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.30 Pengujian tambah kota

Berikut dapat dilihat pengujian dari tambah kota pada tabel 4.35 dibawah ini:

Tabel 4.35 Pengujian tambah kota

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah kota	Menampilkan *form* tambah kota	Dapat menampilkan *form* tambah kota	[√] Diterima [ ] Ditolak
Mengisi data kota Contoh: Provinsi : Jawa Tengah Nama Kota : Semarang	Jika salah satu field tidak terisi, 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.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data kota akan tersimpan didalam *database*	Data kota akan tersimpan didalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.31 Pengujian layanan pengiriman

Berikut dapat dilihat pengujian dari layanan pengiriman pada tabel 4.36 dibawah ini:

Tabel 4.36 Pengujian layanan pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik layanan pengiriman	Menampilkan *form* layanan pengiriman	Dapat menampilkan *form* layanan pengiriman	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah layanan pengiriman	Dapat menampilkan *form* ubah layanan pengiriman	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di *database*	Data dapat berhasil diubah di *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.32 Pengujian tambah layanan pengiriman

Berikut dapat dilihat pengujian dari tambah layanan pengiriman pada tabel 4.37 dibawah ini:

Tabel 4.37 Pengujian tambah layanan pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah layanan pengiriman	Menampilkan *form* tambah layanan pengiriman	Dapat menampilkan *form* tambah layanan pengiriman	[√] Diterima [ ] Ditolak
Mengisi data layanan pengiriman Contoh: Nama Jasa Pengiriman: Tiki Jenis Paket : Regular Logo Perusahaan : jne.png	Jika salah satu field tidak terisi, 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.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Menampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data jasa pengiriman akan tersimpan didalam *database*	Data jasa pengiriman akan tersimpan didalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.33 Pengujian paket pengiriman

Berikut dapat dilihat pengujian dari paket pengiriman pada tabel 4.38 dibawah ini:

Tabel 4.38 Pengujian paket pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik paket pengiriman	Menampilkan *form* paket pengiriman	Dapat menampilkan *form* paket pengiriman	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah paket pengiriman	Dapat menampilkan *form* ubah paket pengiriman	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data detail jasa pengiriman berhasil diubah di*database*	Data detail jasa pengiriman dapat berhasil diubah di *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.34 Pengujian tambah paket pengiriman

Berikut dapat dilihat pengujian dari lihat kategori pada tabel 4.39 dibawah ini:

Tabel 4.39 Pengujian tambah paket pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tambah paket pengiriman	Menampilkan *form* tambah paket pengiriman	Dapat menampilkan *form* tambah detail jasa pengiriman	[√] Diterima [ ] Ditolak
Mengisi data paket pengiriman Contoh: Nana layanan : Tiki Jenis paket : reguler keterangan : 2 hari	Jika salah satu field tidak terisi, 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.	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data detail jasa pengiriman akan tersimpan didalam *database*	Data pengiriman akan tersimpan didalam *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.35 Pengujian tentang kami

Berikut dapat dilihat pengujian dari tentang kami pada tabel 4.40 dibawah ini:

Tabel 4.40 Pengujian tentang kami

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tentang kami	Menampilkan *form* tentang kami	Dapat menampilkan *form* tentang kami	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah tentang kami	Dapat menampilkan *form* ubah tentang kami	[√] Diterima [ ] Ditolak
Tidak mengisi salah satu *field*	Muncul pesan kesalahan *field* tersebut harus diisi	Dapat muncul pesan kesalahan *field* tersebut harus diisi	[√] Diterima [ ] Ditolak
Tidak mengisi semua *field*	Tampilkan pesan kesalahan	Dapat menampilkan pesan kesalahan ”Data belum diisi”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data kontak kami berhasil diubah di *database*	Data kontak kami berhasil diubah di *database*	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.5.2.36 Pengujian testimoni

Berikut dapat dilihat pengujian dari testimoni pada tabel 4.41 dibawah ini:

Tabel 4.41 Pengujian testimoni

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik testimoni	Menampilkan *form* testimoni	Dapat menampilkan *form* testimoni	[√] Diterima [ ] Ditolak
Memasukan nama member	Mengisikan nama member pada *edit box* pencarian	Dapat mengisikan nama member pada *edit box* pencarian	[√] Diterima [ ] Ditolak
Klik cari	Menampilkan data testimoni dari member yang dicari dan akan muncul pesan kesalahan jika nama yang dicari tidak ada	Dapat menampilkan data testimoni	[√] Diterima [ ] Ditolak
Klik hapus	Menghapus data testimoni dari *database*	Dapat menghapus data testimoni dari *database*	[√] Diterima [ ] Ditolak

4.5.2.37 Pengujian data order

Berikut dapat dilihat pengujian dari data order pada tabel 4.42 dibawah ini:

Tabel 4.42 Pengujian data order

Kasus dan Hasil Uji Benar (Data Benar)			
Aktifitas yang dilakukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik data order	Menampilkan *form* data order	Dapat menampilkan *form* data order	[√] Diterima [ ] Ditolak

Klik *detail order*	Menampilkan *form* detail order	Dapat menampilkan *form* detail order	[√] Diterima [ ] Ditolak
Klik print data pengiriman	Menampilkan *form* detail order	Dapat menampilkan *form* detail order	[√] Diterima [ ] Ditolak
Klik ubah	Menampilkan *form* ubah berita	Dapat menampilkan *form* ubah berita	[√] Diterima [ ] Ditolak
Klik jika barang akan dikirimkan	Menampilkan *form* barang akan dikirim	Dapat menampilkan *form* barang akan dikirim	[√] Diterima [ ] Ditolak
Klik jika barang telah sampai di tujuan	Menampilkan *form* barang sudah diterima	Dapat menampilkan pesan kesalahan ”ada kesalahan didalam pengisian data, silahkan cek kembali”	[√] Diterima [ ] Ditolak
Klik tombol simpan	Data berhasil diubah di*database*	Tampil pesan “Data berhasil diubah”	[√] Diterima [ ] Ditolak
Klik batal	Kembali kemenu sebelumnya	Tampil menu sebelumnya	[√] Diterima [ ] Ditolak

4.6 Pengujian Beta

4.6.1 Skenario Pengujian Beta

Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung ditempat penelitian. Penelitian ini dilakukan di dua lingkungan :

1. Lingkungan petugas di Amalinda Collection

Penelitian di lingkungan petugas Amalinda Collection dilakukan supaya dapat mengetahui sejauh mana sistem yang akan dibangun dapat menjadi alternatif penyelesaian permasalahan yang telah dijelaskan di awal.

2. Lingkungan Masyarakat Umum

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

   Adapun beberapa poin yang diteliti adalah :

   a. Tampilan antar muka yang *user friendly*
   b. *Website* dapat memenuhi kebutuhan informasi pelanggan
   c. *Website* menjadi sarana memperluas pangsa pasar toko Amalinda Collection.
   d. Proses pemesanan barang yang mudah dilakukan.
   e. Penggunaan jasa pengiriman TIKI mempermudah pelacakan keberadaan barang.
   f. Sistem pembayaran yang mudah digunakan.
   g. Proses retur barang mudah dilakukan.
   h. Kelayakan *website* untuk dipublikasikan.
   i. Berdasarkan data hasil kuisioner tersebut, dapat dicari prosentase masing-masing jawaban dengan menggunakan rumus : Y = P/Q * 100%

Keterangan :

P = Banyaknya jawaban responden tiap soal.

Q = Jumlah responden

Y = Nilai persentase

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

1. Apakah anda setuju bahwa tampilan *website* ini sudah *user friendly* (mudah digunakan, tampilannya menarik dan tidak membosankan)?

Tabel 4.50 Hasil pengujian kuesioner soal nomor 1

Pertanyaan	**No.**	**Keterangan**	**Responden**	**Prosentase (%)**
1	1	Sangat Setuju	3	30
	2	Setuju	6	60
	3	Biasa Saja	1	10
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 atau 30% menyatakan setuju, 6 atau 60% menyatakan setuju, dan 1 atau 10% menyatakan biasa saja bahwa tampilan *website* cukup menarik dan mudah untuk dipahami.

2. Apakah anda setuju bahwa informasi barang yang diberikan oleh *website* ini sesuai dengan informasi yang anda butuhkan?

Tabel 4.51 Hasil pengujian kuesioner soal nomor 2

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	3	30
	2	Setuju	7	70
	3	Biasa Saja	0	0
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 atau 30% menyatakan setuju, dan 7 atau 70% menyatakan setuju bahwa informasi barang yang diberikan pada *website* sudah sesuai dengan yang member butuhkan.

3. Apakah anda setuju dengan adanya *website* ini, Toko Snowmount dapat memperluas pangsa pasarnya?

Tabel 4.52 Hasil pengujian kuesioner soal nomor 3

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	2	20
	2	Setuju	6	60
	3	Biasa Saja	2	20
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 atau 20% menyatakan setuju, 6 atau 60% menyatakan setuju, dan 2 atau 20% menyatakan biasa saja bahwa Toko Snowmount dapat memperluas pangsa pasarnya dengan adanya *website ini.*

4. Apakah anda setuju bahwa proses pemesanan barang pada *website* ini sudah mudah dilakukan oleh anda selaku member?

Tabel 4.53 Hasil pengujian kuesioner soal nomor 4

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	2	20
	2	Setuju	7	70
	3	Biasa Saja	1	10
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 atau 20% menyatakan setuju, 7 atau 70% menyatakan setuju, dan 1 atau 10% menyatakan biasa saja bahwa proses pemesanan barang pada *website* ini sudah mudah dilakukan.

5. Apakah anda setuju bahwa penggunaan jasa pengiriman TIKI pada *website* ini dapat mempermudah member dalam melakukan pelacakan keberadaan barang?

Tabel 4.54 Hasil pengujian kuesioner soal nomor 5

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	4	40
	2	Setuju	6	60
	3	Biasa Saja	0	0
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 4 atau 40% menyatakan setuju, dan 6 atau 60% menyatakan setuju bahwa penggunaan jasa pengiriman JNE pada *website* ini dapat mempermudah member dalam melakukan pelacakan keberadaan barang.

6. Apakah anda setuju bahwa sistem pembayaran paypal, transfer bank dan COD yang diterapkan pada *website* ini dapat mempermudah member dalam melakukan pembayarannya?

Tabel 4.55 Hasil pengujian kuesioner soal nomor 6

Pertanyaan	**No.**	**Keterangan**	**Responden**	**Prosentase (%)**
1	1	Sangat Setuju	2	20
	2	Setuju	6	60
	3	Biasa Saja	2	20
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 atau 20% menyatakan setuju, 6 atau 60% menyatakan setuju, dan 2 atau 20% menyatakan biasa saja bahwa sistem pembayaran paypal, transfer bank dan COD mempermudah member dalam melakukan pembayaran.

7. Apakah anda setuju bahwa semua informasi transaksi yang dikirimkan melalui email oleh *website* ini sudah lengkap?

Tabel 4.56 Hasil pengujian kuesioner soal nomor 7

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	1	10
	2	Setuju	6	60
	3	Biasa Saja	3	30
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 1 atau 10% menyatakan setuju, 6 atau 60% menyatakan setuju, dan 3 atau 30% menyatakan biasa saja bahwa informasi yang dikirimkan melalui email member sudah lengkap.

8. Apakah anda setuju bahwa semua informasi cara-cara mendaftar, memesan dan melakukan pembayaran pada menu bantuan pada *website* ini sudah jelas?

Tabel 4.57 Hasil pengujian kuesioner soal nomor 8

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	3	30
	2	Setuju	5	50
	3	Biasa Saja	2	20
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 atau 30% menyatakan setuju, 5 atau 50% menyatakan setuju, dan 2 atau

20% menyatakan biasa saja bahwa informasi tentang cara-cara registrasi, pemesanan dan pembayaran sudah jelas.

9. Apakah anda setuju bahwa proses retur barang yang diberikan oleh *website* ini mudah dilakukan?

Tabel 4.58 Hasil pengujian kuesioner soal nomor 9

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	3	30
	2	Setuju	5	50
	3	Biasa Saja	2	20
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 3 atau 30% menyatakan setuju, 5 atau 50% menyatakan setuju, dan 2 atau 20% menyatakan biasa saja bahwa proses retur barang yang ada sudah mudah dilakukan.

10. Apakah anda setuju bahwa *website* ini sudah layak untuk di-*online*-kan?

Tabel 4.59 Hasil pengujian kuesioner soal nomor 10

Pertanyaan	No.	Keterangan	Responden	Prosentase (%)
1	1	Sangat Setuju	2	20
	2	Setuju	6	60
	3	Biasa Saja	2	20
	4	Tidak Setuju	0	0
	5	Sangat Tidak Setuju	0	0
Jumlah			**10**	**100**

Berdasarkan hasil prosentase diatas maka dapat disimpulkan sebanyak 2 atau 20% menyatakan setuju, 6 atau 60% menyatakan setuju, dan 2 atau 20% menyatakan biasa saja bahwa *website* ini sudah layak untuk di-*online*-kan.

4.6.2 Kesimpulan Pengujian Betha

Berdasarkan hasil prosesntase diatas yang didapatkan dari pengujian betha pengguna admin dan member, maka sistem penerapan E-Commerce untuk sistem penjualan ini sudah sesuai dengan tujuan yaitu Amalinda Collection dapat mempermudah dalam proses promosi dan penjualan barang yang ditawarkan, member dapat melakukan transaksi pembelian barang secara online, tampilannya *user friendly*, memudahkan dalam pengelolaan data barang dan memudahkan dalam membuat laporan penjualan.