BAB IV

IMPLEMENTASI DAN PENGUJIAN

IV.1. Implementasi

Tahap implementasi merupakan rangkaian pelaksanaan kegiatan yang dilakukan setelah tahap perancangan sistem selesai dilaksanakan, dengna tujuan yang ingin dicapai adalah dapat dioprasikannya hasil perancangan sistem yang telah dibuat.

IV.1.1. Implementasi Perangkat Lunak

Perangkat Lunak yang digunakan dalam membangun Sistem Penjualan *online* di LOSTMANNEQUIN adalah sebagai berikut :

1. Sistem Operasi MacOS Snow Leopard 10.6.8
2. Bahasa Pemrograman menggunakan PHP Version 5.2.17
3. *Database* menggunakan MySQL
4. *Local Server* menggunakan Apache
5. *Code editor* menggunakan Adobe Dreamweaver CS 5
6. System Modeler menggunakan Microsoft Visio 2007,ConceptDraw Office Pro

IV.1.2. Implementasi Perangkat Keras

Perangkat keras yang digunakan dalam membangun Sistem Penjualan *online* di LOSTMANNEQUIN adalah sebagai berikut :

1. Processor mengunakan Intel Core 2 Duo 2.00 Ghz
2. RAM sebesar 4 GB
3. Harddisk sebesar 250 Gb
4. Monitor dengan resolusi 1280 x 800
5. VGA NVIDIA GFORCE 320m
6. Keyboard, Mouse

IV.1.3.3 Implementasi Database

Tabel IV.1 *Table* Bank

***Table* Bank**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`bank` (`bankId` INT(11) NOT NULL AUTO_INCREMENT , `nama_bank` VARCHAR(45) NOT NULL , `rekeningLM` VARCHAR(45) NOT NULL , `atasnama` VARCHAR(100) NOT NULL , PRIMARY KEY (`bankId`)) ENGINE = InnoDB AUTO_INCREMENT = 5 DEFAULT CHARACTER SET = latin1;

Tabel IV.2 *Table* Kategori

***Table* Kategori**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`kategori` (`kdkategori` INT(11) NOT NULL AUTO_INCREMENT , `nama_kategori` VARCHAR(45) NOT NULL , `seokategori` VARCHAR(100) NOT NULL , PRIMARY KEY (`kdkategori`)) ENGINE = InnoDB AUTO_INCREMENT = 4 DEFAULT CHARACTER SET = latin1;

Tabel IV.3 *Table* Produk

***Table* Produk**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`produk` (`produkId` INT(11) NOT NULL AUTO_INCREMENT , `kdkategori` INT(11) NOT NULL , `nama_produk` VARCHAR(100) NOT NULL , `seoproduk` VARCHAR(100) NOT NULL , `stok` INT(11) NOT NULL DEFAULT '0' , `berat` FLOAT NOT NULL DEFAULT '0' , `harga` INT(11) NOT NULL DEFAULT '0' , `deskripsi` VARCHAR(255) NOT NULL , `last_update` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP , `insert_date` TIMESTAMP NOT NULL DEFAULT '0000-00-00 00:00:00' , `view` INT(11) NOT NULL DEFAULT '0' , `diskon` DECIMAL(5,2) NOT NULL DEFAULT '0.00' , PRIMARY KEY (`produkId`) , INDEX `kdkategori` (`kdkategori` ASC) , INDEX `nama_produk` (`nama_produk` ASC) , CONSTRAINT `fk_kategori_produk` FOREIGN KEY (`kdkategori`) REFERENCES `lostm810_lostmannequin`.`kategori` (`kdkategori`) ON DELETE CASCADE ON UPDATE CASCADE) ENGINE = InnoDB AUTO_INCREMENT = 37 DEFAULT CHARACTER SET = latin1;

Tabel IV.4 *Table* User

***Table* User**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`user` (`userID` INT(11) NOT NULL AUTO_INCREMENT , `email` VARCHAR(75) NOT NULL , `password` VARCHAR(75) NOT NULL , `first_name` VARCHAR(255) NOT NULL DEFAULT '' , `last_name` VARCHAR(255) NOT NULL DEFAULT ' ' , `gender` ENUM('male','female') NOT NULL DEFAULT 'male' , `level` CHAR(1) NOT NULL DEFAULT '5' , `blocked` ENUM('y','n') NOT NULL DEFAULT 'n' , `kode` VARCHAR(255) NOT NULL , `insert_date` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP , PRIMARY KEY (`userID`) ,

```
  UNIQUE INDEX `email_UNIQUE` (`email` ASC) ,
  INDEX `insert_date` (`insert_date` ASC) ,
  INDEX `last_name` (`last_name` ASC) )
ENGINE = InnoDB
AUTO_INCREMENT = 27
DEFAULT CHARACTER SET = latin1;
```

Tabel IV.5 *Table Comment*

Table Comment

```
CREATE  TABLE IF NOT EXISTS `lostm810_lostmannequin`.`comment` (
  `commentId` INT(11) NOT NULL AUTO_INCREMENT ,
  `userID` INT(11) NOT NULL ,
  `produkId` INT(11) NOT NULL ,
  `comment` VARCHAR(255) NOT NULL ,
  `insert_date` DATETIME NOT NULL ,
  PRIMARY KEY (`commentId`) ,
  INDEX `userID` (`userID` ASC) ,
  INDEX `produkId` (`produkId` ASC) ,
  INDEX `insert_date` (`insert_date` ASC) ,
  CONSTRAINT `fk_produk_comment`
    FOREIGN KEY (`produkId` )
    REFERENCES `lostm810_lostmannequin`.`produk` (`produkId` )
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `fk_user_comment`
    FOREIGN KEY (`userID` )
    REFERENCES `lostm810_lostmannequin`.`user` (`userID` )
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
AUTO_INCREMENT = 25
DEFAULT CHARACTER SET = latin1;
```

Tabel IV.6 *Table* Favorit

***Table* Favorit**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`favorit` (`idfav` INT(11) NOT NULL AUTO_INCREMENT , `userID` INT(11) NOT NULL , `produkId` INT(11) NOT NULL , `insert_date` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP , PRIMARY KEY (`idfav`) , INDEX `userID` (`userID` ASC) , INDEX `produkId` (`produkId` ASC) , CONSTRAINT `fk_produk_favorit` FOREIGN KEY (`produkId`) REFERENCES `lostm810_lostmannequin`.`produk` (`produkId`) ON DELETE CASCADE ON UPDATE CASCADE, CONSTRAINT `fk_user_favorit` FOREIGN KEY (`userID`) REFERENCES `lostm810_lostmannequin`.`user` (`userID`) ON DELETE CASCADE ON UPDATE CASCADE) ENGINE = InnoDB AUTO_INCREMENT = 30 DEFAULT CHARACTER SET = latin1;

Tabel IV.7 *Table* Negara

***Table* Negara**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`negara` (`kdnegara` INT(11) NOT NULL AUTO_INCREMENT , `nama_negara` VARCHAR(45) NOT NULL , PRIMARY KEY (`kdnegara`) , INDEX `nama_negara` (`nama_negara` ASC)) ENGINE = InnoDB AUTO_INCREMENT = 3 DEFAULT CHARACTER SET = latin1;

Tabel IV.8 *Table* Provinsi

***Table* Provinsi**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`provinsi` (`kdprovinsi` INT(11) NOT NULL AUTO_INCREMENT , `kdnegara` INT(11) NOT NULL , `nama_provinsi` VARCHAR(45) NOT NULL , PRIMARY KEY (`kdprovinsi`) , INDEX `kdnegara` (`kdnegara` ASC) , INDEX `nama_provinsi` (`nama_provinsi` ASC) , CONSTRAINT `fk_negara_provinsi` FOREIGN KEY (`kdnegara`) REFERENCES `lostm810_lostmannequin`.`negara` (`kdnegara`) ON DELETE CASCADE ON UPDATE CASCADE) ENGINE = InnoDB AUTO_INCREMENT = 36 DEFAULT CHARACTER SET = latin1;

Tabel IV.9 *Table* Kota

***Table* Kota**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`kota` (`kdkota` INT(11) NOT NULL AUTO_INCREMENT , `kdprovinsi` INT(11) NOT NULL , `nama_kota` VARCHAR(45) NOT NULL , PRIMARY KEY (`kdkota`) , INDEX `kdprovinsi` (`kdprovinsi` ASC) , INDEX `nama_kota` (`nama_kota` ASC) , CONSTRAINT `fk_provinsi_kota` FOREIGN KEY (`kdprovinsi`) REFERENCES `lostm810_lostmannequin`.`provinsi` (`kdprovinsi`) ON DELETE CASCADE ON UPDATE CASCADE) ENGINE = InnoDB AUTO_INCREMENT = 100 DEFAULT CHARACTER SET = latin1;

Tabel IV.10 *Table* Lostmannequin_log

***Table* Lostmannequin_log**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`lostmannequin_log` (`logId` INT(11) NOT NULL AUTO_INCREMENT , `activity` VARCHAR(45) NOT NULL DEFAULT '' , `waktu` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP , `status` ENUM('read','unread') NOT NULL DEFAULT 'unread' , PRIMARY KEY (`logId`) , INDEX `status` (`status` ASC)) ENGINE = InnoDB AUTO_INCREMENT = 162 DEFAULT CHARACTER SET = latin1;

Tabel IV.11 *Table* Matauang

***Table* Matauang**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`matauang` (`kode` VARCHAR(4) NOT NULL , `kurs` DOUBLE NOT NULL , PRIMARY KEY (`kode`)) ENGINE = MyISAM DEFAULT CHARACTER SET = latin1;

Tabel IV.12 *Table* Ongkos_kirim

***Table* Ongkos_kirim**
CREATE *TABLE* IF NOT EXISTS `lostm810_lostmannequin`.`ongkos_kirim` (`id_ongkir` INT(11) NOT NULL AUTO_INCREMENT , `kdnegara` INT(11) NOT NULL , `kdkota` INT(11) NOT NULL , `ongkos` INT(11) NOT NULL DEFAULT '0' , PRIMARY KEY (`id_ongkir`) , INDEX `kdnegara` (`kdnegara` ASC) , INDEX `kdkota` (`kdkota` ASC) , CONSTRAINT `ongkos_kirim_ibfk_1` FOREIGN KEY (`kdnegara`) REFERENCES `lostm810_lostmannequin`.`negara` (`kdnegara`) ON DELETE CASCADE ON UPDATE CASCADE, CONSTRAINT `ongkos_kirim_ibfk_2` FOREIGN KEY (`kdkota`) REFERENCES `lostm810_lostmannequin`.`kota` (`kdkota`)

```
   ON DELETE CASCADE
   ON UPDATE CASCADE)
ENGINE = InnoDB
AUTO_INCREMENT = 197
DEFAULT CHARACTER SET = latin1;
```

Tabel IV.13 *Table ShippingInfo*

Table ShippingInfo

```
CREATE  TABLE IF NOT EXISTS `lostm810_lostmannequin`.`shippingInfo` (
  `sid` INT(11) NOT NULL AUTO_INCREMENT ,
  `userID` INT(11) NOT NULL ,
  `notelp` VARCHAR(45) NOT NULL ,
  `kdnegara` INT(11) NOT NULL ,
  `kdkota` INT(11) NOT NULL ,
  `alamat` TEXT NOT NULL ,
  `kodepos` VARCHAR(45) NOT NULL ,
  `nama` VARCHAR(100) NOT NULL ,
  PRIMARY KEY (`sid`) ,
  INDEX `kdnegara` (`kdnegara` ASC) ,
  INDEX `kdkota` (`kdkota` ASC) ,
  INDEX `fk_user_shipping` (`userID` ASC) ,
  CONSTRAINT `fk_kota_shipping`
    FOREIGN KEY (`kdkota` )
    REFERENCES `lostm810_lostmannequin`.`kota` (`kdkota` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_negara_shipping`
    FOREIGN KEY (`kdnegara` )
    REFERENCES `lostm810_lostmannequin`.`negara` (`kdnegara` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_user_shipping`
    FOREIGN KEY (`userID` )
    REFERENCES `lostm810_lostmannequin`.`user` (`userID` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB
AUTO_INCREMENT = 54
DEFAULT CHARACTER SET = latin1;
```

Tabel IV.14 *Table* Orders

***Table* Orders**

```
CREATE  TABLE IF NOT EXISTS `lostm810_lostmannequin`.`orders` (
  `orderId` INT(11) NOT NULL AUTO_INCREMENT ,
```

```
  `userID` INT(11) NOT NULL ,
  `sid` INT(11) NOT NULL ,
  `total_bayar` DOUBLE NOT NULL ,
  `status`
ENUM('PAID','PAID(UNCONFIRMED)','UNPAID','SHIPPED','CANCELLED')    NOT
NULL DEFAULT 'UNPAID' ,
  `noresi` VARCHAR(15) NOT NULL DEFAULT '' ,
  `insert_date` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ,
  PRIMARY KEY (`orderId`) ,
  INDEX `userID` (`userID` ASC) ,
  INDEX `sid` (`sid` ASC) ,
  CONSTRAINT `fk_user_order`
    FOREIGN KEY (`userID` )
    REFERENCES `lostm810_lostmannequin`.`user` (`userID` )
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `orders_ibfk_1`
    FOREIGN KEY (`sid` )
    REFERENCES `lostm810_lostmannequin`.`shippinginfo` (`sid` )
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
AUTO_INCREMENT = 49
DEFAULT CHARACTER SET = latin1;
```

Tabel IV.15 *Table* Order_Detail

Table Order_detail

```
CREATE  TABLE IF NOT EXISTS `lostm810_lostmannequin`.`order_detail` (
  `orderdetailId` INT(11) NOT NULL AUTO_INCREMENT ,
  `orderId` INT(11) NOT NULL ,
  `produkId` INT(11) NOT NULL ,
  `jumlah` INT(11) NOT NULL DEFAULT '0' ,
  PRIMARY KEY (`orderdetailId`) ,
  INDEX `produkId` (`produkId` ASC) ,
  INDEX `orderId` (`orderId` ASC) ,
  CONSTRAINT `fk_order_orderdetail`
    FOREIGN KEY (`orderId` )
    REFERENCES `lostm810_lostmannequin`.`orders` (`orderId` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_produk_orderdetail`
    FOREIGN KEY (`produkId` )
```

```
  REFERENCES `lostm810_lostmannequin`.`produk` (`produkId` )
  ON DELETE NO ACTION
  ON UPDATE NO ACTION)
ENGINE = InnoDB
AUTO_INCREMENT = 48
DEFAULT CHARACTER SET = latin1;
```

Tabel IV.16 *Table* Pembayaran

Table Pembayaran

```
CREATE  TABLE IF NOT EXISTS `lostm810_lostmannequin`.`pembayaran` (
  `paymentId` INT(11) NOT NULL AUTO_INCREMENT ,
  `userID` INT(11) NOT NULL ,
  `orderID` INT(11) NOT NULL ,
  `carabayar` ENUM('PAYPAL','BANK') NOT NULL ,
  `atasnama` VARCHAR(255) NOT NULL ,
  `Bankid` INT(11) NOT NULL ,
  `tanggal_bayar` DATE NOT NULL ,
  `jumlah` DOUBLE NOT NULL ,
  `kode_transaksi_paypal` VARCHAR(100) NOT NULL ,
  `pembayar` VARCHAR(100) NOT NULL ,
  `matauangbayar` VARCHAR(4) NOT NULL ,
  `waktubayar` DATETIME NOT NULL ,
  `insert_date` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP
ON UPDATE CURRENT_TIMESTAMP ,
  PRIMARY KEY (`paymentId`) ,
  INDEX `userID` (`userID` ASC) ,
  INDEX `bankId` (`Bankid` ASC) ,
  INDEX `orderID` (`orderID` ASC) ,
  CONSTRAINT `fk_bank_pembayaran`
    FOREIGN KEY (`Bankid` )
    REFERENCES `lostm810_lostmannequin`.`bank` (`bankId` )
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `fk_user_pembayaran`
    FOREIGN KEY (`userID` )
    REFERENCES `lostm810_lostmannequin`.`user` (`userID` )
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `pembayaran_ibfk_1`
    FOREIGN KEY (`orderID` )
    REFERENCES `lostm810_lostmannequin`.`orders` (`orderId` )
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
```

```
AUTO_INCREMENT = 26
DEFAULT CHARACTER SET = latin1;
```

Tabel IV.17 *Table* Restock

Table Restock

```
CREATE  TABLE IF NOT EXISTS `lostm810_lostmannequin`.`restock` (
  `restockId` INT(11) NOT NULL AUTO_INCREMENT ,
  `userID` INT(11) NOT NULL ,
  `produkId` INT(11) NOT NULL ,
  `stock` INT(11) NOT NULL DEFAULT '0' ,
  `insert_date` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP
,
  PRIMARY KEY (`restockId`) ,
  INDEX `userID` (`userID` ASC) ,
  INDEX `produkId` (`produkId` ASC) ,
  INDEX `insert_date` (`insert_date` ASC) ,
  CONSTRAINT `fk_produk_restok`
    FOREIGN KEY (`produkId` )
    REFERENCES `lostm810_lostmannequin`.`produk` (`produkId` )
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `fk_user_restok`
    FOREIGN KEY (`userID` )
    REFERENCES `lostm810_lostmannequin`.`user` (`userID` )
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
AUTO_INCREMENT = 2
DEFAULT CHARACTER SET = latin1;
```

Tabel IV.18 *Table* Return

***Table* Return**

```
CREATE  TABLE IF NOT EXISTS `lostm810_lostmannequin`.`retur` (
  `id` INT(11) NOT NULL AUTO_INCREMENT ,
  `userID` INT(11) NOT NULL ,
  `orderID` INT(11) NOT NULL ,
  `alasan` TEXT NOT NULL ,
  PRIMARY KEY (`id`) ,
  INDEX `orderId` (`orderID` ASC) ,
  INDEX `userID` (`userID` ASC) ,
  CONSTRAINT `fk_order_return`
```

```
    FOREIGN KEY (`orderID` )
    REFERENCES `lostm810_lostmannequin`.`orders` (`orderId` )
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `fk_user_return`
    FOREIGN KEY (`userID` )
    REFERENCES `lostm810_lostmannequin`.`user` (`userID` )
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;
```

Tabel IV.19 Tabel retur_detail

***Table* Retur_detail**
CREATE *TABLE* IF NOT EXISTS `retur_detail` (`returdetailID` int(11) NOT NULL AUTO_INCREMENT, `returID` int(11) NOT NULL, `produkID` int(11) NOT NULL, `qty` int(11) NOT NULL, `alasan` enum('Wrong','Damaged') NOT NULL, PRIMARY KEY (`returdetailID`), KEY `returID` (`returID`), KEY `produkID` (`produkID`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=31 ;

Tabel IV.20 Trigger *iComment*

Trigger i*Comment*
DELIMITER $$ USE `lostm810_lostmannequin`$$ CREATE DEFINER=`root`@`localhost` TRIGGER `lostm810_lostmannequin`.`i*Comment*` AFTER INSERT ON `lostm810_lostmannequin`.`*comment*` FOR EACH ROW insert into lostmannequin_log(activity,waktu) values ('i*Comment*',now())$$

Tabel IV.21 Trigger iFavorit

Trigger iFavorit
DELIMITER ; DELIMITER $$

```
USE `lostm810_lostmannequin`$$
CREATE
DEFINER=`root`@`localhost`
TRIGGER `lostm810_lostmannequin`.`iFavorit`
AFTER INSERT ON `lostm810_lostmannequin`.`favorit`
FOR EACH ROW
insert into lostmannequin_log(activity,waktu) value ('iFavorit',now())$$
```

Tabel IV.22 Trigger iOrder

Trigger iOrder
DELIMITER ; DELIMITER $$ USE `lostm810_lostmannequin`$$ CREATE DEFINER=`root`@`localhost` TRIGGER `lostm810_lostmannequin`.`iOrder` AFTER INSERT ON `lostm810_lostmannequin`.`orders` FOR EACH ROW insert into lostmannequin_log(activity,waktu) values ('iOrder',now())$$

Tabel IV.23 Trigger iPayment

Trigger iPayment
DELIMITER ; DELIMITER $$ USE `lostm810_lostmannequin`$$ CREATE DEFINER=`root`@`localhost` TRIGGER `lostm810_lostmannequin`.`iPayment` AFTER INSERT ON `lostm810_lostmannequin`.`pembayaran` FOR EACH ROW insert into lostmannequin_log(activity,waktu) values ('iPayment',now())$$

Tabel IV.24 Trigger iRestock

Trigger iRestock
DELIMITER ; DELIMITER $$ USE `lostm810_lostmannequin`$$ CREATE DEFINER=`root`@`localhost` TRIGGER `lostm810_lostmannequin`.`iRestock` AFTER INSERT ON `lostm810_lostmannequin`.`restock` FOR EACH ROW insert into lostmannequin_log(activity,waktu) value ('iRestock',now())$$

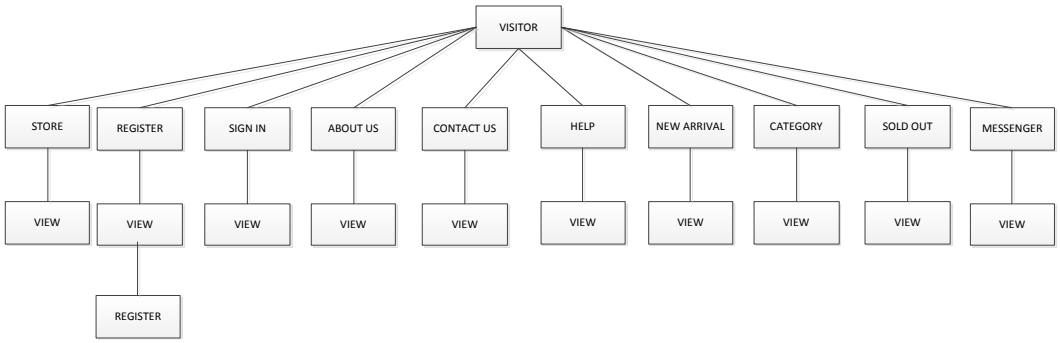
Tabel IV.25 Trigger iUser

Tigger iUser
DELIMITER ; DELIMITER $$ USE `lostm810_lostmannequin`$$ CREATE DEFINER=`root`@`localhost` TRIGGER `lostm810_lostmannequin`.`iUser` AFTER INSERT ON `lostm810_lostmannequin`.`user` FOR EACH ROW insert into lostmannequin_log(activity,waktu) values ('iUser',now())$$

IV.1.4. Struktur Menu

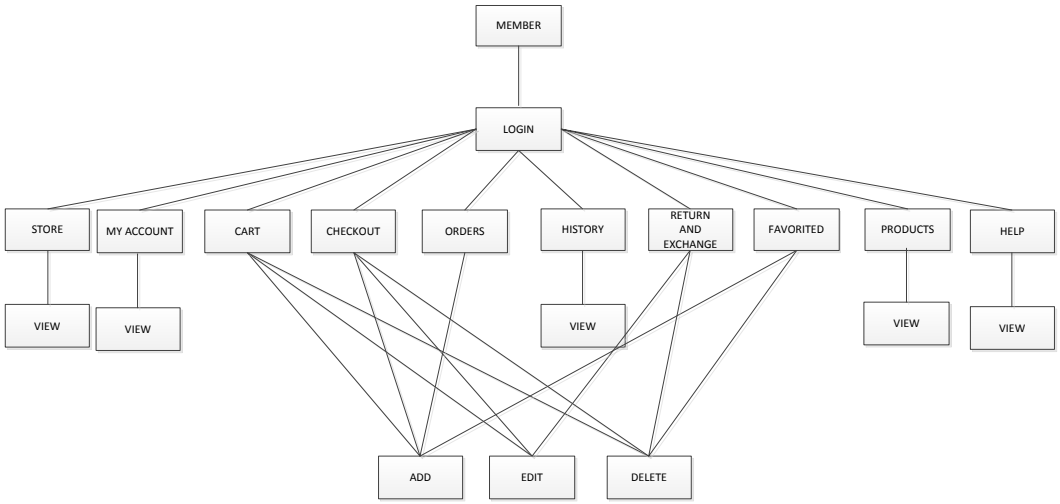
Berikut ini adalah struktur menu dari sistem penjualan *online* LOSTMANNEQUIN

a. Struktur Menu Pengunjung (Visitor)



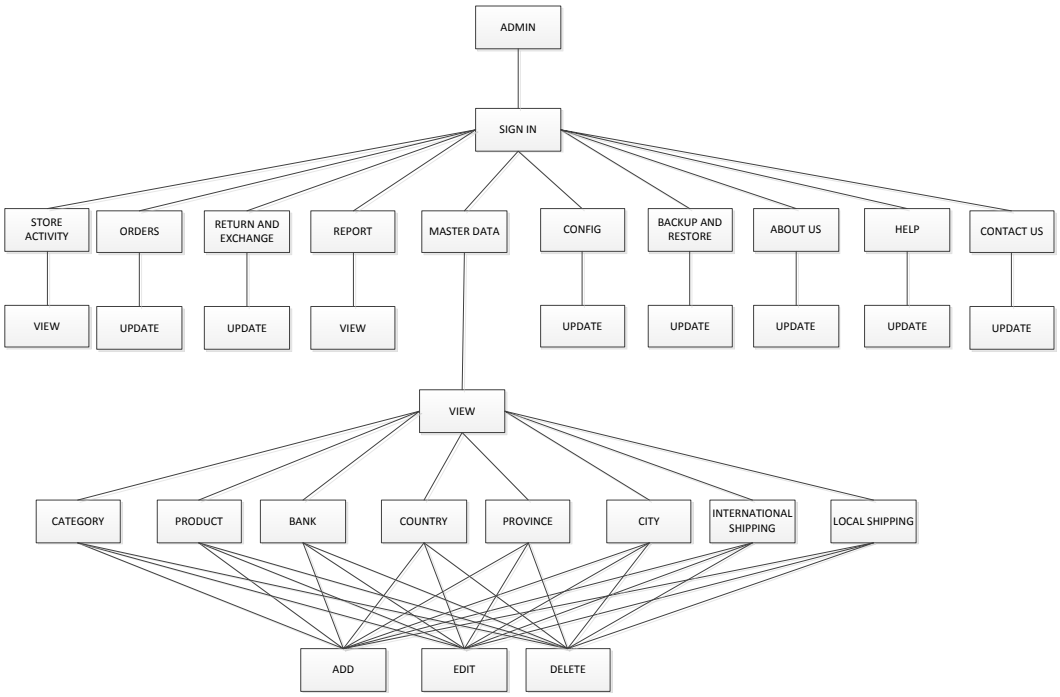
Gambar IV.1 Struktur Menu Pengunjung

b. Struktur Menu Member



Gambar IV.2 Struktur Menu Member

c. Struktur Menu Admin



Gambar IV.3 Struktur Menu Admin

IV.1.5. Implementasi Antarmuka

Implementasi antarmuka menggambarkan tampilan dari aplikasi yang dibangun yaitu implementasi antarmuka sistem penjualan *online* pada LOSTMANNEQUIN . Implementasi antarmuka sistem penjualan *online* pada LOSTMANNEQUIN dapat dilihat pada tabel IV.26.

Tabel IV.26 Implementasi Antarmuka

Menu	Deskripsi	Nama *file*
Halaman *Store*	File Program untuk menampilkan menu utama dari website LOSTMANNEQUIN	index.php
Halaman *Sign In*	*File* Program untuk menampilkan *form* login dan lupa password *member*	Login.php

Halaman *Register*	*File* Program untuk menampilkan form registrasi *member*	Register.php
Halaman *Cart*	*File* Program untuk menampilkan keranjang belanja *member*	Cart.php
Halaman *Checkout*	*File* Program menampilkan form checkout *member*	Checkout.php
Halaman *Help*	*File* Program untuk menampilkan halaman bantuan *member* tentang cara pesan, cara pembayaran, pengiriman serta cara retur	Help.php
Halaman *About us*	*File* Program untuk menampilkan halaman yang berisi tentang profil LOSTMANNEQUIN	Aboutus.php
Halaman *Contactus*	*File* Program untuk menampilkan halaman yang berisi cara untuk berkomunikasi dengan pihak LOSTMANNEQUIN	Contactus.php
Halaman *New Arrival*	*File* Program untuk menampilkan informasi produk produk paling baru yang ada pada website LOSTMANNEQUIN	new.php
Halaman *Soldout*	*File* Program untuk menampilkan informasi produk yang sudah *sold out*	soldout.php
Halaman *Categories*	*File* program untuk menampilkan informasi untuk produk yang ada pada kategori di LOSTMANNEQUIN	kategori.php
Halaman *Myaccount*	*File* program untuk menampilkan halaman menu utama *member*	*Member*.php
Halaman produk	*File* program untuk menampilkan halaman yang berisi gambar dan detail produk	Produk.php
Halaman *edit your profile*	*File* Program untuk menampilkan form *update* password member	Update_your_password.php
Halaman *favorited list*	*File* program untuk menampilkan halaman daftar produk yang telah difavoritkan	Favorited_list.php
Halaman *order*	*File* program untuk daftar *order* yang sedang dilakukan	Order_list.php
Halaman *history*	*File* program untuk daftar *order* yang sudah pernah dilakukan	History.php
Halaman *return and exchange*	*File* program untuk menampilkan halaman retur	Retur_proses.php

Halaman *invoice*	*File* program untuk menampilkan detail dari pesanan yang dilakukan oleh member	Invoice.php
Halaman *invoice* retur	*File* program untuk menampilkan detail dari retur yang dilakukan oleh member	Invoice_retur.php
Halaman Checkout	*File* program untuk menampilkan form pengisian alamat pengiriman	Checkout.php
Halaman *payment confirmation*	*File* program untuk menampilkan form konfirmasi pembayaran oleh member	Confirm.php
Halaman *Forgot* Password	*File* program untuk menampilkan form lupa password member	Forgot_your_password.php
Halaman *login* administrator	*File* program untuk menampilkan form login administrator	Serviceauthlogin.php
Halaman *administrator panel*	*File* program untuk menampilkan halaman utama admin	Index.php
Halaman *activity*	*File* program untuk menampilkan list aktivitas yang sedang tejadi di LOSTMANNEQUIN	Activity.php
Halaman *orders*	*File* program untuk menampilkan halaman pengolahan data order	History.php
Halaman *restock*	*File* program untuk menampilkan halaman produk yang telah *direquest restock oleh member*	Restock.php
Halaman *Return and Exchange*	*File* program untuk menampilkan halaman pengolahan retur produk	Retur_list.php
Halaman *report*	*File* program untuk menampilkan halaman pengolahan laporan produk	Laporan.php
Halaman *Categories*	*File* program untuk menampilkan halaman pengolahan data kategori	kategori.php
Halaman *Products*	*File* program untuk menampilkan halaman pengolahan data produk	Produk.php
Halaman *member list*	*File* program untuk menampilkan halaman pengolahan data member	Member.php
Halaman Bank	*File* program untuk menampilkan halaman pengolahan data bank	Bank.php
Halaman *Country*	*File* program untuk menampilkan halaman pengolahan data negara	negara.php
Halaman *province*	*File* program untuk menampilkan halaman pengolahan data *province*	Provinsi.php
Halaman *City*	*File* program untuk menampilkan halaman pengolahan data kota	Kota.php
Halaman *Local Shipping*	*File* program untuk menampilkan halaman pengolahan data ongkos kirim local	Shipping.php
Halaman *International Shipping*	*File* program untuk menampilkan halaman pengolahan data ongkos kirim luar negeri	Ln_shipping.php

Halaman *Config*	*File* program untuk menampilkan halaman *update* password admin	Password.php
Halaman *Help*	*File* program untuk menampilkan halaman pengolahan data *help*	Help.php
Halaman *About us*	*File* program untuk menampilkan halaman pengolahan data *about us*	Aboutus.php
Halaman *Contact us*	*File* program untuk menampilkan pengolahan data *contact us*	Contactus.php

IV. 2 Pengujian *Black Box*

Proses pengujian *black box* adalah pengujian yang dilakukan dengan cara mencoba program aplikasi dengan memasukkan data ke dalam form-form yang telah disediakan. Pada tahap ini merupakan kelanjutan dari tahap implementasi. Pengujian yang dilakukan dengan cara alpha yaitu dengan metode pengujian *black box* yang berfokus pada persyaratan fungsional perangkat lunak. Pengujian ini memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya semua persyaratan fungsional untuk suatu program.

IV.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara alpha menggunakan metode *black box* dan beta dengan kuesioner.

1. Rencana pengujian halaman pengunjung

   Rencana pengujian pengunjung dapat dilihat pada tabel IV.27.

Tabel IV.27 Rencana Pengujian halaman pengunjung

Item Pengujian	Detail Pengujian	Jenis Pengujian
Register	Menampilkan *form* registrasi	*Black Box*
	Isi *form* registrasi	*Black Box*

	Simpan data registrasi	*Black Box*
Aktivasi	Buka email	*Black Box*
	Aktivasi pendaftaran	*Black Box*

2. Rencana pengujian halaman *member*

Rencana pengujian halaman *member* dapat dilihat pada tabel IV.28.

Tabel IV.28 Rencana Pengujian Halaman *Member*

Item Pengujian	Detail Pengujian	Jenis Pengujian
Login *Member*	Isi data login	*Black Box*
	Verifikasi *username*	*Black Box*
	Verifikasi password	*Black Box*
Forgot Password	Menampilkan *form forgot* password	*Black Box*
	Masukkan email *member*	*Black Box*
	Kirim password *member* ke email pemesan	*Black Box*
Edit your Profile	Menampilkan *form profile member*	*Black Box*
	Verifikasi data profile *member*	*Black Box*
Cart	Menampilkan pemesanan sementara	*Black Box*
	Mengedit data detail pemesanan sementara	*Black Box*
	Menghapus data detail pemesanan sementara	*Black Box*
Checkout	Menampilkan form alamat tujuan pengiriman produk	*Black Box*
	Isi form alamat tujuan pengiriman produk	*Black Box*
Invoice	Menampilkan total bayar	*Black Box*

	Menampilkan cara pembayaran	*Black Box*
	Menyimpan data pembayaran	*Black Box*
Konfirmasi Pembayaran	Menampilkan form konfirmasi pembayaran	*Black Box*

3. Rencana pengujian halaman admin

Rencana pengujian halaman admin dapat dilihat pada tabel IV.29.

Tabel IV.29 Rencana Pengujian Halaman Admin

Item Pengujian	Detail Pengujian	Jenis Pengujian
Login Admin	Isi data login	*Black Box*
	Verifikasi data login admin	*Black Box*
Update password admin	Menampilkan *form* update password Admin	*Black Box*
	Verifikasi data *profile* Admin	

IV.2.2 Kasus dan Hasil Pengujian Alpha

Proses pengujian alpha dilakukan untuk mengetahui apakah data yang dimasukkan (*Input*) sudah sesuai dengan yang diharapkan (*output*). Berdasarkan rencana pengujian, maka dapat dilakukan pengujian alpha pada sistem penjualan *online* di LOSTMANNEQUIN.

1. Pengujian Login Admin

Pengujian login admin dapat dilihat pada tabel IV.30

Tabel IV.30 Pengujian Login Admin

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	*Email* : administrator@lostmannequin.com Password : admin
Yang Diharapkan	Proses akan dilanjutkan dengan menampilkan menu utama admin
Pengamatan	Admin memasukkan *email* dan *password* lalu tekan tombol login maka akan tampil halaman utama admin, sesuai dengan yang diharapkan
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	*Username* : admin, Password : 12345
Yang Diharapkan	Muncul pesan "*wrong email or password*"
Pengamatan	Muncul pesan kesalahan, kombinasi email dan password salah.
Kesimpulan	Diterima

2. Pengujian Data *Add New Products*

Pengujian data konten web dapat dilihat pada tabel IV.30.

Tabel IV.30 Pengujian Data Konten Web

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	*Category : Bag and Purse* *Product Name : Thor The Shy Creature* *Price : 500000* *Discount : 10%* *Stock : 3* *Weight : 2* *Description : Thor the shy creature !* *images :*

	
Yang Diharapkan	Data Konten Web disimpan ke *database* dan tampil pesan perubahan konten web berhasil
Pengamatan	administrator mengisi form add new product lalu memilih tombol simpan maka akan muncul pesan penambahan produk telah berhasil dilakukan
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	*Category : Bag and Purse* *Product Name : -* *Price : -* *Discount : 10%* *Stock : 3* *Weight : 2* *Description : Thor the shy creature !* *images :* 
Yang Diharapkan	Muncul pesan kesalahan, "*a value required*"
Pengamatan	Muncul pesan kesalahan, " *a value required*"

Kesimpulan	Diterima

3. Pengujian Block *Member*

Pengujian block *member* dapat dilihat pada tabel IV.31.

Tabel IV.31 Pengujian Data *Member*

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	Status : Blocked
Yang Diharapkan	Data *member* disimpan ke *database* dan muncul pesan "*this member has blocked from store*"
Pengamatan	Muncul pesan block berhasil
Kesimpulan	Diterima

4. Pengujian Pengolahan Data Kategori

Pengujian pengolahan data kategori dapat dilihat pada tabel IV.34.

Tabel IV.34 Pengujian Data Kategori

Kasus Data dan Hasil Uji (Data Normal)	
Tambah	
Data Masukkan	Kategori : *Accessories*
Yang Diharapkan	Data Kategori tersimpan ke *database* dan menampilkan pesan "*a new category has added*"
Pengamatan	Admin mengisi form tambah kategori lalu pilih simpan, dan muncul pesan "*a new category has added*"
Kesimpulan	Diterima
Edit	
Data Masukkan	Kategori : *Accessories*
Yang Diharapkan	Data Kategori tersimpan ke *database* dan menampilkan pesan "*category has edited*"
Pengamatan	Admin mengisi form edit kategori lalu tekan simpan, dan muncul pesan "*category has edited*"
Kesimpulan	Diterima

Kasus Data dan Hasil Uji (Data Salah)	
Tambah	
Data Masukkan	Kategori : *Accessories*
Yang Diharapkan	Muncul pesan kesalahan, " *this category already on system* "
Pengamatan	Muncul pesan kesalahan, "*this category already on system*".
Kesimpulan	Diterima

5. Pengujian Pengolahan Data *Country*

Pengujian pengolahan data *country* dapat dilihat pada tabel IV.32.

Tabel IV.32 Pengujian Data *Country*

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	*Country Name : Alaska* *Shipping Cost : 1000000*
Yang Diharapkan	Data *Country* tersimpan ke *database* dan muncul pesan *"New Country Has added to Store*"
Pengamatan	Data *Country* tersimpan ke *database* dan muncul pesan *"New Country Has added to Store*"
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	*Country Name : -* *Shipping Cost : 100000*
Yang Diharapkan	Muncul pesan kesalahan "*a value required*"
Pengamatan	Muncul pesan kesalahan "*a value required"*
Kesimpulan	Diterima

6. Pengujian Pengolahan Data *Province*

Pengujian pengolahan data *province* dapat dilihat pada tabel IV.33.

Tabel IV.33 Pengujian Data Provinsi

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	Province Name : Lombok
Yang Diharapkan	Data *Province* tersimpan ke *database* dan muncul pesan "*new province has added to store*"
Pengamatan	Data *Province* tersimpan ke *database* dan muncul pesan "*new province has added to store*"
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	*Province Name : -*
Yang Diharapkan	Muncul pesan kesalahan "*a value required*"
Pengamatan	Muncul pesan kesalahan "*a value required"*
Kesimpulan	Diterima

7. Pengujian Pengolahan Data Kota

Pengujian pengolahan data kota dapat dilihat pada tabel IV.34.

Tabel IV.34 Pengujian Data *City*

Kasus Data dan Hasil Uji (Data Normal)	
Tambah	
Data Masukkan	*City Name* : Bali, *Province Name* : Bali, *Shipping Cost* : 10000
Yang Diharapkan	Data Kota tersimpan ke *database* dan muncul pesan "*new city has been added* "
Pengamatan	Data Kota tersimpan ke *database* dan muncul pesan "*new city has been added*"
Kesimpulan	Diterima
Edit	
Data Masukkan	*City Name* : Bali,

	Province Name : Bali, *Shipping Cost* : 10000
Yang Diharapkan	Data Kota tersimpan ke *database* dan muncul pesan "*city edited*"
Pengamatan	Data Kota tersimpan ke *database* dan muncul pesan "*City edited*"
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Tambah	
Data Masukkan	*City Name* : -, *Province Name* : Bali, *Shipping Cost* : 10000
Yang Diharapkan	Data *City* tidak tersimpan ke *database* dan muncul pesan kesalahan "*a value required*"
Pengamatan	Data *City* tidak tersimpan ke database dan muncul pesan kesalahan "*a value required*"
Kesimpulan	Diterima

8. Pengujian Pendaftaran *member*

Pengujian pendaftaran dapat dilihat pada tabel IV.47.

Tabel IV.35 Pengujian Pendaftaran

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	*First Name* : Aam *Last Name* : Agus Mucharam *Email :* skoolkaholic@gmail.com *Gender* : Male *Password* : BGQRUR45PK *Confirm Password* : BGQRUR45PK
Yang Diharapkan	Data *member* disimpan ke *database*, dan sistem mengirimkan link konfirmasi ke alamat email yang didaftarkan untuk mengaktifkan status menjadi aktif.
Pengamatan	Data *member* disimpan ke *database*, dan sistem

	mengirimkan link konfirmasi ke alamat email yang didaftarkan untuk mengaktifkan status menjadi aktif.
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	*First Name* : Aam *Last Name* : Agus Mucharam *Email :* - *Gender* : Male *Password* : BGQRUR45PK *Confirm Password* : BGQRUR45PK
Yang Diharapkan	Muncul pesan kesalahan "*a value required*"
Pengamatan	Muncul pesan kesalahan "*a value required"*
Kesimpulan	Diterima

9. Pengujian Login *Member*

Pengujian login *member* dapat dilihat pada tabel IV.39.

Tabel IV.39 Pengujian Login *Member*

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	*Email :* skoolkaholic@gmail.com *Password :* BGQRUR45PK
Yang Diharapkan	Proses akan dilanjutkan dengan menampilkan halaman utama *member*
Pengamatan	Menampilkan halaman utama *member*
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	*Email :* skoolkaholic@gmail.com *Password :* BGQRUR45PKck
Yang Diharapkan	Muncul pesan kesalahan, *"wrong email or password*"
Pengamatan	Muncul pesan kesalahan*"wrong email or password*"
Kesimpulan	Diterima

10. Pengujian Lupa password

Pengujian lupa password dapat dilihat pada tabel IV.40.

Tabel IV.36 Pengujian Lupa Password

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	*Email* : skoolkaholic@gmail.com
Yang Diharapkan	Muncul pesan "*an email has been sent to your inbox ☺*"
Pengamatan	Muncul pesan "*an email has been sent to your inbox ☺*"
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	*Email* : 12345@yahoo.com
Yang Diharapkan	Muncul pesan kesalahan "*wrong email !*"
Pengamatan	Muncul pesan kesalahan "*wrong email !*"
Kesimpulan	Diterima

11. Pengujian Edit Password *Member*

Pengujian edit password *member* dapat dilihat pada tabel IV.43.

Tabel IV.43 Pengujian Edit Password *Member*

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	*Old Password :* BGQRUR45PK *New Password :* NOMNOMNOM *Confirm :* NOMNOMNOM
Yang Diharapkan	Muncul pesan "*your password succesfully updated*"
Pengamatan	Muncul pesan "*your password succesfully updated*"
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	*Old Password :* Bcaaaaaa *New Password :* NOMNOMNOM *Confirm :* NOMNOMNOM

Yang Diharapkan	Muncul pesan "*wrong old password !*"
Pengamatan	Muncul pesan "*wrong old password !*"
Kesimpulan	Diterima

12. Pengujian Cart

Pengujian keranjang belanja dapat dilihat pada tabel IV.44.

Tabel IV.37 Pengujian *Cart*

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	qty : 1 (stock produk adalah 5)
Yang Diharapkan	Proses akan dilanjutkan ke form alamat tujuan pengiriman produk.
Pengamatan	*Member* mengisi jumlah produk yang dipesan yaitu 1 sedangkan stock produk adalah 5 lalu tekan *update* dilanjutkan dengan tekan tombol lanjutkan, menuju form *checkout*
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	Jumlah : 10 (stock produk adalah 5)
Yang Diharapkan	Muncul pesan "*we only have 5 qty for this items*"
Pengamatan	*Member* mengisi jumlah produk yang dipesan yaitu 10 sedangkan stock produk adalah 5 lalu tekan update muncul pesan "*we only have 5 qty for this items*". Dan qty yang ada di cart menjadi 5
Kesimpulan	Diterima

13. Pengujian *Shipping Information*

Pengujian pengisian *Shipping Information* dapat dilihat pada tabel IV.38.

Tabel IV.38 pengisian *Shipping Information*

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	*Name :* Kezia hapsari *Telephone :* 092846839289 *Country :* Indonesia *Province :* Jawa Barat *City :* Depok *Address :* Kuningan No 12 *Postal code :*45363
Yang Diharapkan	Proses dilanjutkan ke halaman *invoice*
Pengamatan	Proses dilanjutkan ke halaman *invoice*
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	*Name :* Kezia hapsari *Telephone :* - *Country :* Indonesia *Province :* Jawa Barat *City :* Depok *Address :* Kuningan No 12 *Postal code :*45363
Yang Diharapkan	Muncul pesan kesalahan "*a value required*"
Pengamatan	Muncul pesan kesalahan "*a value required*"
Kesimpulan	Diterima

14. Pengujian Form Pembayaran

Pengujian form pembayaran dapat dilihat pada tabel IV.39.

Tabel IV.39 Pengujian Form Pembayaran

Kasus Data dan Hasil Uji (Data Normal)	
Data Masukkan	*Name* : Aam Agus Mucharam *From* : Bank Mandiri *To* : Bank Mandiri - 100102010230 *Date* : Aug 5, 2011 *Amount* : 150000,-
Yang Diharapkan	Data Pembayaran Disimpan ke database
Pengamatan	Data Pembayaran Disimpan ke database
Kesimpulan	Diterima
Kasus Data dan Hasil Uji (Data Salah)	
Data Masukkan	*Name* : - *From* : Bank Mandiri *To* : Bank Mandiri - 100102010230 *Date* : Aug 5, 2011 *Amount* : 150000,-
Yang Diharapkan	Muncul pesan kesalahan "*a value required*"
Pengamatan	Muncul pesan kesalahan "*a value required*"
Kesimpulan	Diterima

IV.2.3 Kesimpulan Hasil Pengujian *Alpha*

Berdasarkan hasil pengujian *alpha* yang telah dilakukan dapat ditarik kesimpulan bahwa sistem penjualan *online* pada LOSTMANNEQUIN sudah berjalan cukup maksimal, tetapi tidak menutup kemungkinan dapat terjadi kesalahan pada suatu saat aplikasi digunakan. Sehingga membutuhkan proses *maintenance* untuk lebih mengetahui kekurangan dari sistem penjualan *online* ini.

IV.2.4 Kasus dan Hasil Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana program aplikasi diuji secara langsung ke lapangan yaitu dengan membuat *kuesioner* yang ditujukan kepada pengguna sistem penjualan *online* pada LOSTMANNEQUIN selanjutnya ditanyakan langsung kepada responden. Hasil pengujian beta disertakan dalam lampiran. *Kuesioner* disebar kepada 10 orang pengguna, *kuesioner* ini masing-masing terdiri dari 5 pertanyaan (contoh *kuesioner* dapat dilihat pada lampiran).

Daftar pertanyaan yang diberikan untuk umum adalah sebagai berikut :

1. Apakah anda setuju tampilan website www.lostmannequin.com mudah dipahami / *user friendly ?*

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

2. Apakah anda setuju informasi produk yang diberikan oleh *website* www.lostmannequin.com sesuai dengan informasi yang anda butuhkan?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

3. Apakah anda setuju layanan pembayaran *online* (paypal) dan transfer bank yang diterapkan pada *website e-commerce* www.lostmannequin.com dapat mempermudah anda dalam melakukan pembayaran?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

4. Apakah anda setuju dengan *website*www.lostmannequin.com bisa memudahkan layanan transaksi penjualan jarak jauh sehingga perusahaan dalam memperluas pangsa pemasaran ?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

5. Apakah anda setuju bahwa *website*www.lostmannequin.com telah layak untuk di-*online*-kan?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

Daftar pertanyaan yang diberikan untuk toko adalah sebagai berikut :

1. Apakah anda setuju tampilan *website* www.lostmannequin.comcukup menarik dan mudah dipahami untuk pengolahan data?
   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

2. Apakah anda setuju dengan cara penggunaan *website* www.lostmannequin.com dalam mengelola transaksi pemesanan?
   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

3. Apakah anda setuju dengan cara penggunaan *website* www.lostmannequin.comdalam mengelola data produk?
   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

4. Apakah anda setuju dengan adanya *website* www.lostmannequin.com membantu perusahaan dalam pengolahan data yang selama ini masih manual menjadi terstruktur ?
   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

5. Apakah anda setuju dengan adanya *website* www.lostmannequin.com dapat membantu calon konsumen dalam melakukan pembelian dan pembayaran secara *online*?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

6. Apakah anda setuju bahwa website www.lostmannequin.com sudah layak untuk di *online* kan?

   A. Sangat Setuju
   B. Setuju
   C. Cukup Setuju
   D. Kurang Setuju
   E. Tidak Setuju

Berdasarkan data hasil kuisioner, dapat dicari prosentase masing – masing jawaban dengan menggunakan rumus : Y = P/Q * 100%.

Keterangan : Y= nilai prosentase

P = banyaknya jawaban tiap pertanyaan

Q = jumlah responden

Berikut ini adalah hasil prosentase masing-masing jawaban yang ditujukan untuk umum yang sudah dihitung nilainya dengan menggunakan rumus diatas.

1. Apakah anda setuju tampilan website www.lostmannequin.com mudah dipahami / *user friendly ?*

Tabel IV.47 Hasil pengujian *kuesioner* pengunjung/*member* soal no 1

Pilihan	Keterangan	Responden	Prosentase (%)
a	Sangat Setuju	1	10 %
b	Setuju	3	30 %
c	Cukup Setuju	3	30%
d	Kurang Setuju	3	30%
e	Tidak Setuju	-	-

2. Apakah anda setuju informasi produk yang diberikan oleh *website* www.lostmannequin.com sesuai dengan informasi yang anda butuhkan?

Tabel IV.48 Hasil pengujian *kuesioner* pengunjung/*member* soal no 2

Pilihan	Keterangan	Responden	Prosentase (%)
a	Sangat Setuju	1	10 %
b	Setuju	4	40 %
c	Cukup Setuju	3	30%
d	Kurang Setuju	2	20%
e	Tidak Setuju	-	-

3. Apakah anda setuju layanan pembayaran *online* (paypal) dan transfer bank yang diterapkan pada *website e-commerce* www.lostmannequin.com dapat mempermudah anda dalam melakukan pembayaran?

Tabel IV.49 Hasil pengujian *kuesioner* pengunjung/ *member* soal no 3

Pilihan	Keterangan	Responden	Prosentase (%)
a	Sangat Setuju	1	10 %
b	Setuju	4	40 %
c	Cukup Setuju	3	30%
d	Kurang Setuju	2	20%
e	Tidak Setuju	-	-

4. Apakah anda setuju layanan pembayaran *online* (paypal) dan transfer bank yang diterapkan pada *website e-commerce* www.lostmannequin.com dapat mempermudah anda dalam melakukan pembayaran?

Tabel IV.50 Hasil pengujian *kuesioner* pengunjung/ *member* soal no 4

Pilihan	Keterangan	Responden	Prosentase (%)
a	Sangat Setuju	1	10%
b	Setuju	4	40 %
c	Cukup Setuju	3	30%
d	Kurang Setuju	2	20%
e	Tidak Setuju	-	-

5. Apakah anda setuju dengan *website* www.lostmannequin.com bisa memudahkan layanan transaksi penjualan jarak jauh sehingga perusahaan dalam memperluas pangsa pemasaran ?

Tabel IV.51 Hasil pengujian *kuesioner* pengunjung/ *member* soal no 5

Pilihan	Keterangan	Responden	Prosentase (%)
a	Sangat Setuju	-	-
b	Setuju	3	30 %
c	Cukup Setuju	4	40%
d	Kurang Setuju	3	30%
e	Tidak Setuju	-	-

Berikut ini adalah hasil persentase masing-masing jawaban yang ditujukan untuk pegawai dari pihak LOSTMANNEQUIN yang sudah dihitung nilainya dengan menggunakan rumus diatas.

1. Apakah anda setuju tampilan *website* www.lostmannequin.com cukup menarik dan mudah dipahami untuk pengolahan data?

Tabel IV.53 Hasil pengujian *kuesioner* pegawai soal no 1

Pilihan	Keterangan	Responden	Prosentase (%)
a	Sangat Setuju	2	100%
b	Setuju	-	-
c	Cukup Setuju	-	-
d	Kurang Setuju	-	-
e	Tidak Setuju	-	-

2. Apakah anda setuju dengan cara penggunaan *website* www.lostmannequin.com dalam mengelola transaksi pemesanan?

Tabel IV.54 Hasil pengujian *kuesioner* pegawai soal no 2

Pilihan	Keterangan	Responden	Prosentase (%)
a	Sangat Setuju	1	50%
b	Setuju	1	50%
c	Cukup Setuju	-	-
d	Kurang Setuju	-	-
e	Tidak Setuju	-	-

3. Apakah anda setuju dengan cara penggunaan *website* www.lostmannequin.com dalam mengelola data produk?

Tabel IV.55 Hasil pengujian *kuesioner* pegawai soal no 3

Pilihan	Keterangan	Responden	Prosentase (%)
a	Sangat Setuju	2	100 %
B	Setuju	-	-
C	Cukup Setuju	-	-
D	Kurang Setuju	-	-
E	Tidak Setuju	-	-

4. Apakah merasa mudah jika menangani transaksi di *e-commerce* pada LOSTMANNEQUIN?

Tabel IV.56 Hasil pengujian kuesioner pegawai soal no 4

Pilihan	Keterangan	Responden	Prosentase (%)
a	Sangat Setuju	2	100 %
b	Setuju	-	-
c	Cukup Setuju	-	-
d	Kurang Setuju	-	-
e	Tidak Setuju	-	-

5. Apakah *e-commerce* pada LOSTMANNEQUIN memudahkan pihak toko dalam pembuatan laporan ?

Tabel IV.57 Hasil pengujian *kuesioner* pegawai soal no 5

Pilihan	Keterangan	Responden	Prosentase (%)
a	Sangat Setuju	1	50 %
b	Setuju	1	50%
c	Cukup Setuju	-	-
d	Kurang Setuju	-	-
e	Tidak Setuju	-	-

IV.2.5 Kesimpulan Hasil Pengujian Beta

Berdasarkan hasil prosentase diatas yang didapatkan dari pengujian beta, yang dibagikan kepada 12 orang mengatakan bahwa perangkat lunak yang dibangun ini cukup mudah dipelajari, cukup mudah digunakan, dan tampilan antarmuka yang menarik. Selain itu proses yang terdapat di dalam aplikasi dapat berjalan dengan baik.