BAB 4
IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini akan dilakukan implementasi dan pengujian terhadap sistem yang baru. Dimana akan di lihat kekurangan-kekurangan pada system yang baru untuk pengembangan system selanjutnya dan Tahapan ini dilakukan setelah perancangan selesai dan selanjutnya akan diimplementasikan pada bahasa pemrograman yang akan digunakan.

4.1 Implementasi Sistem

4.1.1 Perangkat Lunak Pembangun

Perangkat lunak yang digunakan dalam membangun aplikasi *e-commerce* di toko Moon dapat dilihat pada tabel berikut.

Tabel 4.1 Perangkat lunak pembangunan

Jenis Perangkat Lunak	**Nama Perangkat Lunak**
Operating System	Microsoft Windows 7
Code Editor	Adobe Dreamweaver CS5
Local Server	Wamp Server 2.1 dan XAMPP 1.7.0
System Modeler	Microsoft Visio 2010
Browser	Mozilla Firefox (aurora 16), Google Chrome, Internet Explorer
Image Editor	Adobe Photoshop

4.1.2 Perangkat Keras Pembangun

Perangkat keras yang digunakan dalam membangun aplikasi *e-commerce* di toko boneka Moon *Clothing* dapat dilihat pada tabel berikut.

Tabel 4.2 Perangkat Keras Pembangun

Jenis Perangkat Keras	**Spesifikasi**
Processor	Intel Core i3 2.53 Ghz
RAM	1.00 Gb

Harddisk	64 bit
Monitor	Resolusi 1366x766
VGA	Intel
Aksesoris	Keyboard, Mouse

4.1.3 Implementasi *Database*

Pembangunan *database* dilakukan dengan menggunakan aplikasi *DBMS MySQL*. Implementasi *database* dalam bahasa *SQL* dan berikut adalah langkah-langkah dalam pembuatan database beserta tabel :

1. Pilih tab *databases*, lalu klik MySQL Databases



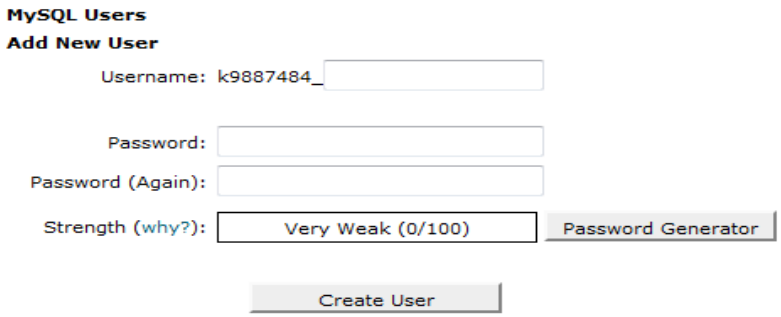
Gambar 4.1 MySQL Databases

2. Isi nama *database* yang diinginkan, lalu klik *Create Database*



Gambar 4.2 Create Databases

3. Isi nama *user* yang akan mengakses *database* yang telah dibuat lengkap dengan *password* sebagai *security*, lalu klik *Create User*



Gambar 4.3 *Create User*

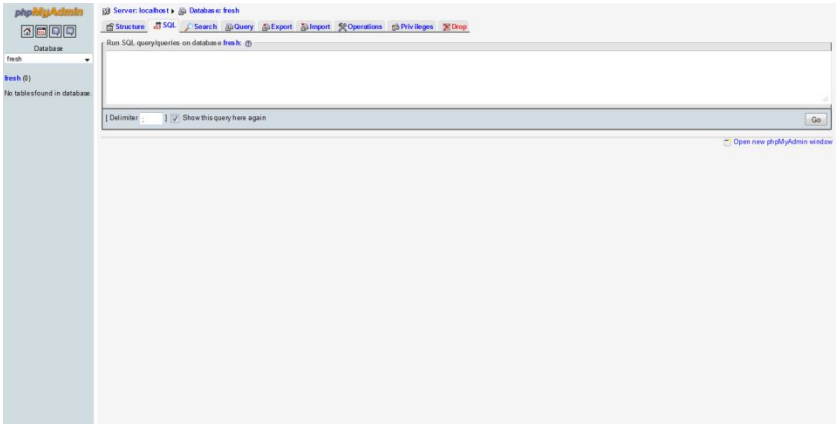
Langkah selanjutnya adalah pembuatan tabel. Berikut merupakan langkah – langkah pembuatan tabel.

1. Pilih tab *databases*, lalu klik phpMyAdmin



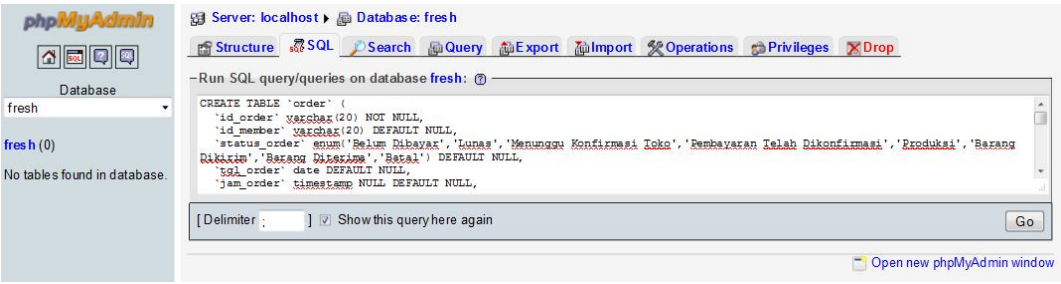
Gambar 4.4 phpMyAdmin

2. Pilih *database* yang akan dibuatkan tabel, lalu klik tab SQL



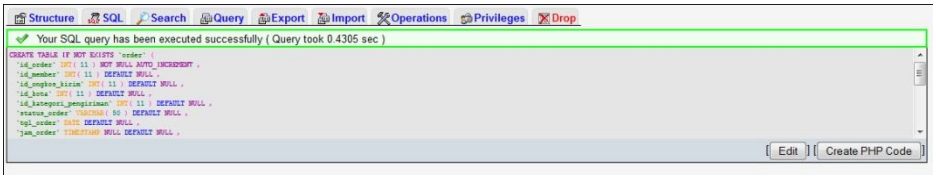
Gambar 4.5 Pilih *database*

3. Isikan *Query* (sebagai contoh *query* untuk tabel order), untuk membuat tabel klik go.



Gambar 4.6 Contoh *Query* Tabel Order

4. Tabel (sebagai contoh *Query* untuk tabel order) telah berhasil dibuat.



Gambar 4.7 Tabel Order berhasil dibuat

Dibawah ini merupakan query untuk setiap tabel yang akan diimplementasikan dalam aplikasi *e-commerce*

4.1.4 Implementasi Basis Data

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

1. Tabel admin

Tabel 4.3 Implementasi basis data admin

```
CREATE TABLE IF NOT EXISTS `admin` (
  `admin_username` varchar(20) NOT NULL,
  `password` varchar(40) NOT NULL,
  `nama` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
  `level` enum('Admin','Gudang','Pemilik') NOT NULL DEFAULT 'Gudang',
  `status` enum('Aktif','Tidak Aktif') NOT NULL DEFAULT 'Aktif',
  PRIMARY KEY (`admin_username`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

2. Tabel bank

Tabel 4.4 Implementasi basis data bank

```
CREATE TABLE IF NOT EXISTS `bank` (
  `id_bank` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nama_bank` varchar(100) NOT NULL,
  `tipe` enum('Paypal','Bank') NOT NULL DEFAULT 'Bank',
  `nomor_rekening` varchar(100) NOT NULL,
  `nama_pemilik` varchar(100) DEFAULT NULL,
  PRIMARY KEY (`id_bank`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;;
```

3. Tabel member

Tabel 4.5 Implementasi basis data member

```
CREATE TABLE IF NOT EXISTS `member` (
  `email_member` varchar(100) NOT NULL,
  `password` varchar(40) NOT NULL,
  `nama` varchar(100) NOT NULL,
  `alamat` tinytext NOT NULL,
  `id_kota` varchar(6) NOT NULL,
  `telepon` varchar(20) DEFAULT NULL,
  `kodepos` varchar(5) DEFAULT NULL,
  `tgl_daftar` date NOT NULL,
  `kode_aktivasi` varchar(40) NOT NULL,
  `status` enum('Belum Aktif','Aktif') NOT NULL DEFAULT 'Belum Aktif',
  `blokir` enum('Ya','Tidak') NOT NULL DEFAULT 'Tidak',
  `session_id` varchar(40) DEFAULT NULL,
  `last_seen` datetime DEFAULT NULL,
  PRIMARY KEY (`email_member`),
  KEY `kota` (`id_kota`)
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4. Tabel kota

Tabel 4.6 Implementasi basis data kota

```
CREATE TABLE IF NOT EXISTS `kota` (
  `id_kota` varchar(6) NOT NULL,
  `nama_kota` varchar(100) NOT NULL,
  `id_propinsi` tinyint(2) unsigned NOT NULL,
  PRIMARY KEY (`id_kota`),
  KEY `propinsi` (`id_propinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5. Tabel retur

Tabel 4.7 Implementasi basis data retur

```
CREATE TABLE IF NOT EXISTS `retur` (
  `id_retur` int(4) unsigned zerofill NOT NULL AUTO_INCREMENT,
  `id_order` int(4) unsigned zerofill NOT NULL,
  `waktu_masuk` datetime NOT NULL,
  `status_retur` enum('Baru','Retur Sudah Direspon','Paket Retur telah Dikirim','Paket Retur telah Sampai') NOT NULL DEFAULT 'Baru',
  `resi_retur` varchar(20) DEFAULT NULL,
  `tgl_kirim_retur` date DEFAULT NULL,
  `waktu_terima_retur` datetime DEFAULT NULL,
  `penerima_retur` varchar(100) DEFAULT NULL,
  `admin_username` varchar(20) DEFAULT NULL,
  `rekening_info` text,
  `status_uang_retur` enum('Belum Ditransfer','Sudah Ditransfer') DEFAULT NULL,
```

```
  `uang_retur` int(10) unsigned DEFAULT '0',
  PRIMARY KEY (`id_retur`),
  KEY `admin_username` (`admin_username`),
  KEY `id_order` (`id_order`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;
```

6. Tabel Kategori

Tabel 4.8 Implementasi basis data kategori

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `id_kategori` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nama_kategori` varchar(100) NOT NULL,
  PRIMARY KEY (`id_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

7. Tabel konfirm bayar

Tabel 4.9 Implementasi basis data konfirm bayar

```
CREATE TABLE IF NOT EXISTS `konfirm_bayar` (
  `id_konfirm_bayar` mediumint(4) unsigned zerofill NOT NULL
AUTO_INCREMENT,
  `id_order` int(4) unsigned zerofill NOT NULL,
  `cara_transfer` enum('Setor Tunai','ATM','Internet Banking','SMS
Banking','PayPal') NOT NULL,
  `tgl_pembayaran` date NOT NULL,
  `waktu_konfirmasi` datetime NOT NULL,
  `id_bank` int(10) unsigned NOT NULL,
  `id_kurs` int(10) unsigned DEFAULT NULL,
  `besar_transfer` float NOT NULL,
  `kode_validasi` varchar(14) NOT NULL,
```

```
  `pemilik_rekening` varchar(255) NOT NULL,
  `status` enum('Valid','Baru','Tidak Valid') NOT NULL DEFAULT 'Baru',
  `admin_username` varchar(20) DEFAULT NULL,
  PRIMARY KEY (`id_konfirm_bayar`),
  KEY `admin_username` (`admin_username`),
  KEY `id_kurs` (`id_kurs`),
  KEY `rekening_tujuan` (`id_bank`),
  KEY `id_order` (`id_order`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;
```

8. Tabel Harga Paket

Tabel 4.10 Implementasi basis data harga paket

```
CREATE TABLE IF NOT EXISTS `harga_paket` (
  `id_kota` varchar(6) NOT NULL,
  `id_paket_pengiriman` mediumint(3) unsigned NOT NULL,
  `harga` int(7) unsigned NOT NULL,
  `lama` tinyint(2) unsigned DEFAULT NULL,
  PRIMARY KEY (`id_kota`,`id_paket_pengiriman`),
  KEY `id_paket_pengiriman` (`id_paket_pengiriman`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

9. Tabel orders

Tabel 4.11 Implementasi basis data orders

```
CREATE TABLE IF NOT EXISTS `orders` (
  `id_order` int(4) unsigned zerofill NOT NULL,
  `status_order` enum('Batal','Belum Bayar','Sudah Konfirmasi','Sudah Lunas','Sudah Dikirim','Telah Sampai','Retur','Selesai') NOT NULL DEFAULT 'Belum Bayar',
  `email_member` varchar(100) NOT NULL,
```

```
  `ongkos_kirim` int(8) unsigned DEFAULT NULL,
  `id_paket_pengiriman` mediumint(3) unsigned NOT NULL,
  `id_kota` varchar(6) NOT NULL,
  `waktu_order` datetime NOT NULL,
  `jatuh_tempo` datetime NOT NULL,
  `penerima` varchar(100) DEFAULT NULL,
  `kodepos` int(10) unsigned DEFAULT NULL,
  `telepon` varchar(20) DEFAULT NULL,
  `alamat` mediumtext,
  `resi_order` varchar(20) DEFAULT NULL,
  `tgl_kirim_order` date DEFAULT NULL,
  `waktu_terima_order` datetime DEFAULT NULL,
  `penerima_order` varchar(100) DEFAULT NULL,
  `admin_username` varchar(20) DEFAULT NULL,
  PRIMARY KEY (`id_order`),
  KEY `paket_pengiriman` (`id_paket_pengiriman`),
  KEY `member` (`email_member`),
  KEY `admin` (`admin_username`),
  KEY `kota` (`id_kota`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

10. Tabel orders_detail

Tabel 4.12 Implementasi basis data order detail

```
CREATE TABLE IF NOT EXISTS `orders_detail` (
  `id_produk` int(10) unsigned NOT NULL,
  `id_order` int(4) unsigned zerofill NOT NULL,
  `harga` int(8) unsigned DEFAULT NULL,
  `banyak` tinyint(3) unsigned NOT NULL,
  `diskon` tinyint(2) unsigned DEFAULT '0',
```

```
  PRIMARY KEY (`id_produk`,`id_order`),
  KEY `id_order` (`id_order`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

11. Tabel produk

Tabel 4.13 Implementasi basis data produk

```
CREATE TABLE IF NOT EXISTS `produk` (
  `id_produk` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nama_produk` varchar(100) NOT NULL,
  `id_kategori` int(10) unsigned NOT NULL,
  `harga` int(8) NOT NULL,
  `diskon` tinyint(2) unsigned NOT NULL,
  `tgl_input` date NOT NULL,
  `admin_username` varchar(20) NOT NULL,
  `berat` int(6) unsigned NOT NULL DEFAULT '0',
  `keterangan` mediumtext,
  `status` enum('Aktif','Non Aktif') NOT NULL DEFAULT 'Aktif',
  `hit` int(10) unsigned NOT NULL DEFAULT '0',
  `stok_terakhir` int(10) unsigned DEFAULT NULL,
  `stok_sekarang` int(10) unsigned NOT NULL,
  `tgl_update` date DEFAULT NULL,
  PRIMARY KEY (`id_produk`),
  KEY `kategori` (`admin_username`),
  KEY `admin` (`admin_username`),
  KEY `id_kat` (`id_kategori`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;
```

12. Tabel provinsi

Tabel 4.14 Implementasi basis data provinsi

```
CREATE TABLE IF NOT EXISTS `propinsi` (
```

```
  `id_propinsi` tinyint(2) unsigned NOT NULL AUTO_INCREMENT,
  `nama_propinsi` varchar(50) NOT NULL,
  PRIMARY KEY (`id_propinsi`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=43 ;
```

13. Tabel Paket pengiriman

Tabel 4.15 Implementasi basis data paket pengiriman

```
CREATE TABLE IF NOT EXISTS `paket_pengiriman` (
  `id_paket_pengiriman` mediumint(3) unsigned NOT NULL
AUTO_INCREMENT,
  `nama_paket_pengiriman` varchar(255) NOT NULL,
  `id_jasa_pengiriman` int(10) unsigned NOT NULL,
  PRIMARY KEY (`id_paket_pengiriman`),
  KEY `jasa` (`id_jasa_pengiriman`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;
```

14. Tabel jasa pengiriman

Tabel 4.16 Implementasi basis data jasa pengiriman

```
CREATE TABLE IF NOT EXISTS `jasa_pengiriman` (
  `id_jasa_pengiriman` int(11) unsigned NOT NULL AUTO_INCREMENT,
  `nama_jasa_pengiriman` varchar(100) NOT NULL,
  PRIMARY KEY (`id_jasa_pengiriman`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

15. Tabel rating

Tabel 4.17 Implementasi basis data rating

```
CREATE TABLE IF NOT EXISTS `rating` (
```

```
    `email_member` varchar(100) NOT NULL,
    `id_produk` int(10) unsigned NOT NULL,
    `rating` enum('1','2','3','4','5') NOT NULL,
    PRIMARY KEY (`email_member`,`id_produk`),
    KEY `id_produk` (`id_produk`)
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

16. Tabel Kurs

Tabel 4.18 Implementasi basis data retur detail

```
CREATE TABLE IF NOT EXISTS `kurs` (
  `id_kurs` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `nilai_kurs` int(6) unsigned NOT NULL,
  `last_update` datetime NOT NULL,
  PRIMARY KEY (`id_kurs`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=266 ;
```

17. Tatabel retur detail

Tabel 4.19 Implementasi basis data retur detail

```
  CREATE TABLE IF NOT EXISTS `retur_detail` (
    `id_produk` int(6) unsigned NOT NULL,
    `id_retur` int(4) unsigned zerofill NOT NULL,
    `alasan` enum('Barang Salah Kirim','Barang Cacat') NOT NULL,
    `banyak` int(3) unsigned NOT NULL,
    `acc_barang` int(3) unsigned NOT NULL,
    `acc_uang` int(3) unsigned DEFAULT NULL,
    `status` enum('Diterima','Ditolak') DEFAULT NULL,
    `keterangan` text,
    PRIMARY KEY (`id_produk`,`id_retur`,`alasan`),
    KEY `id_retur` (`id_retur`)
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4.1.5 Upload File

Langkah selanjutnya setelah implementasi database adalah *upload file* dari *file local* ke *hosting*.

1. Pilih Tab *files* dan kemudian pilih *File Manager*.



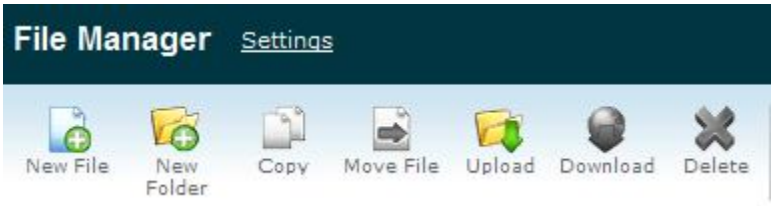
Gambar 4.8 Langkah *Upload File*

2. Pilih *Web Root(public_html)* kemudian klik *Go*.



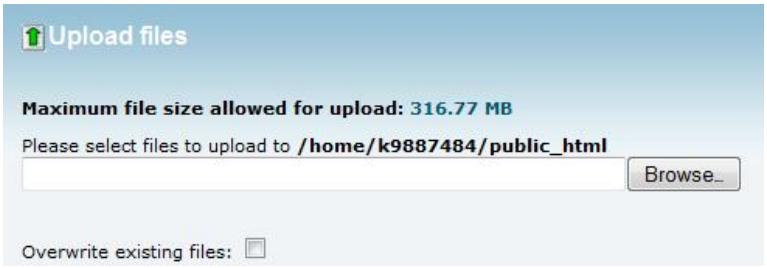
Gambar 4.9 Langkah *Upload File*

3. Pilih ikon *Upload* .



Gambar 4.10 Langkah *Upload File*

4. Klik *Choose File* dan kemudian pilih *file* yang akan di*upload* missal index.php.

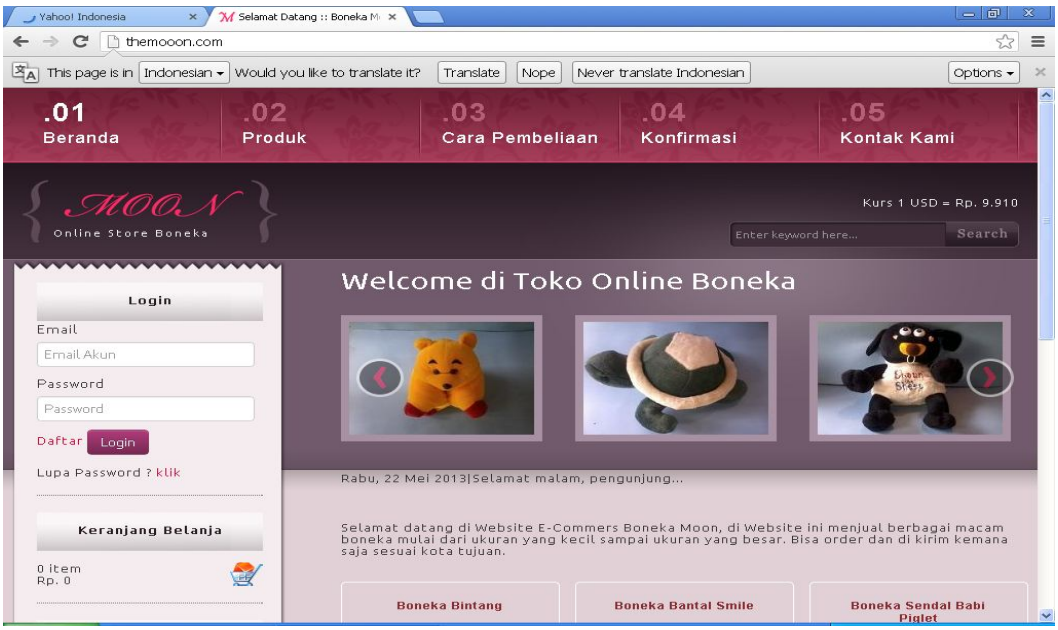


Gambar 4.11 Langkah *Upload File*

4.1.6 Implementasi Antarmuka

1.1.6.1 Implementasi Antarmuka Pengunjung

Implementasi antarmuka beranda yang terdapat pada perangkat lunak yang ditujukan untuk pengunjung dapat dilihat pada gambar berikut ini:

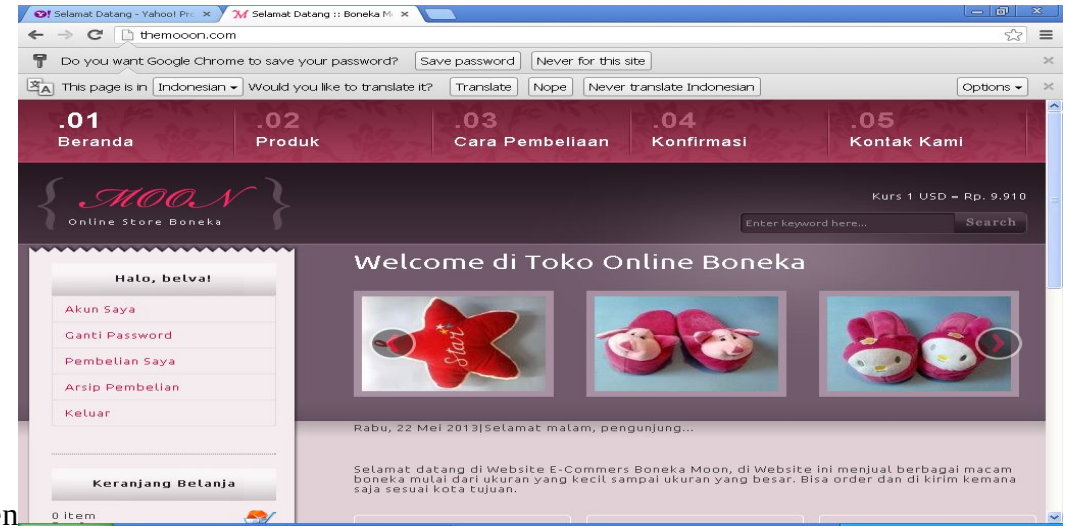


Gambar 4.12 Implementasi antarmuka beranda

1.1.6.2 Implementasi Antarmuka *Member*

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk pelanggan dapat dilihat pada gambar berikut ini:

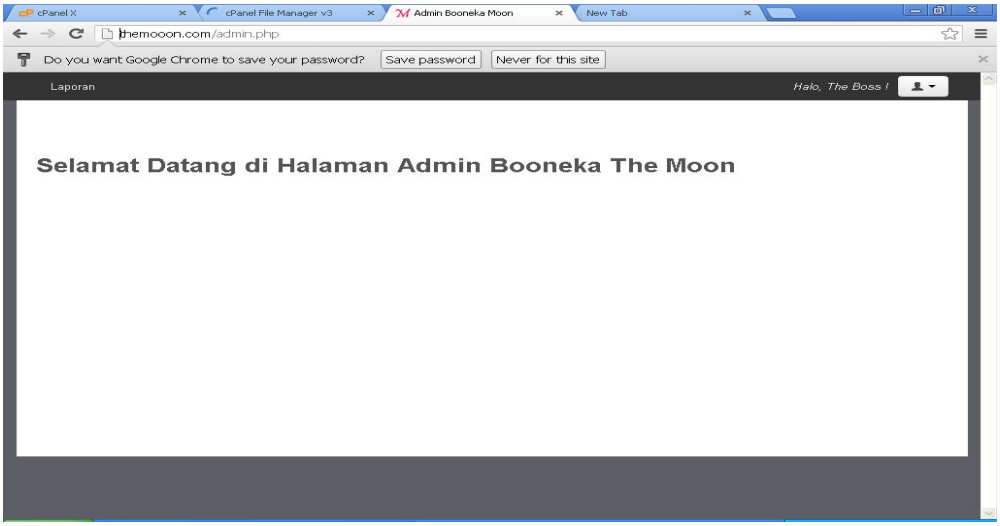
Gambar 4.13

Implemen 

Gambar 4.13 Implementasi antarmuka *member*

1.1.6.3 Implementasi Antarmuka Pemilik

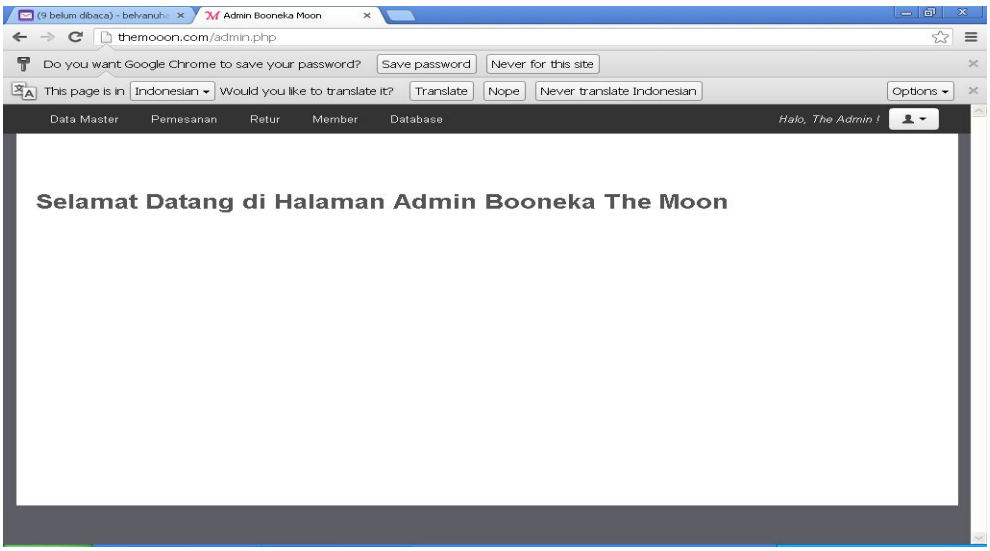
Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk pemimpin dapat dilihat pada gambar berikut ini:



Gambar 4.14 Implementasi antarmuka pemilik

1.1.6.4 Implementasi Antarmuka Administrasi

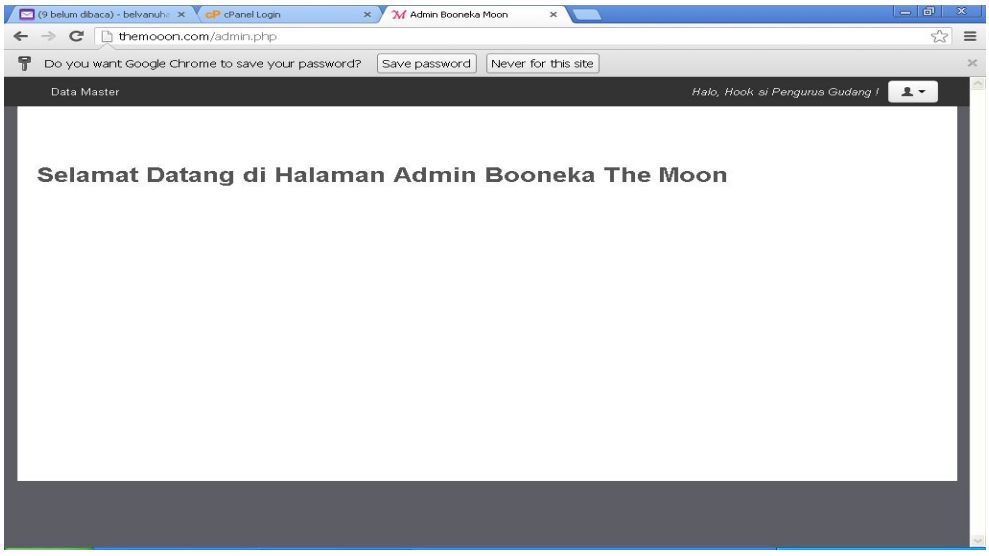
Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk *staff* DBA dapat dilihat pada gambar berikut ini:



Gambar 4.15 Implementasi antarmuka administrasi

1.1.6.5 Implementasi Antarmuka Gudang

Implementasi antarmuka yang terdapat pada perangkat lunak yang ditujukan untuk gudang dapat dilihat pada gambar berikut ini:



Gambar 4.16 Implementasi antarmuka gudang

Daftar struktur *file* yang dibangun untuk membuat sistem ini dapat dilihat pada tabel di bawah ini.

Tabel 4.20 Daftar struktur *file*

Struktur File	**Pengunjung dan Member**

Menu	**Deskripsi**	**Nama File**
Beranda	Halaman awal pengunjung	http://www.themooon.com/?
produk	Digunakan sebagai halaman tampilan produk yang ada di *web*	http://www.themooon.com/?zone=produk
Cara Pembelian	Digunakan sebagai halaman untuk melihat cara pembelian suatu produk dan pembayaran atas pembelian tersebut	http://www.themooon.com/?zone=artikel&action=carabeli
Konfirmasi	Digunakan sebagai halaman untuk konfirmasi pembayaran bagi *member* yang melakukan pembelian	http://www.themooon.com/?zone=member&file=konfirmasi
Kontak Kami	Digunakan sebagai halaman kontak toko	http://www.themooon.com/?zone=artikel&action=kontakkami
Akun saya	Digunakan sebagai halaman untuk data diri member yang da hnn pat di edit oleh member itu sendiri	http://www.themooon.com/?zone=member
Keranjang Belanja	Digunakan sebagai halaman untuk menampilkan catatan produk yang telah dipilih.	
Retur	Digunakan sebagai halaman untuk retur	
Logout	Digunakan sebagai menu untuk keluar dari sistem.	

1.2 Pengujian Sistem

Pengujian yang dilakukan yaitu pengujian secara fungsional *(alpha)* dan *betha*. Metode yang digunakan untuk pengujian *alpha* adalah pengujian *black box* yang berfokus pada persyaratan fungsional dari sistem yang dibangun. Sedangkan untuk pengujian beta dilakukan dengan metode kuantitatif.

4.2.1 Pengujian Alpha

Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan merepresentasikan kajian pokok dari spesifikasi, perancangan, dan pengkodean. Pengujian yang digunakan untuk menguji sistem ini adalah metode pengujian *black-box*. Pengujian *black-box*berfokus pada persyaratan fungsional perangkat lunak.

4.2.1.1 Rencana Pengujian Alpha

Rencana pengujian sistem dibedakan berdasarkan fungsional pengguna yang akan menggunakan aplikasi ini. Pengujian ini dilakukan bagi pengguna sebagai pengunjung, member, pemilik, administrasi, gudang.

4.2.1.1.1 Rencana Pengujian Sistem Pemilik

Tabel 4.21 Rencana Pengujian Sistem Pemimpin

Item Uji	**Detail Pengujian**	**Jenis Pengujian**
Login	Pengecekan data login	*Black Box*
Ubah Password	Merubah data Password	*Black Box*
Edit Profile	Merubah data profile	*Black Box*
Laporan	1. Menampilkan data laporan produk, transaksi	*Black Box*
Logout	Menghapus data *session login*	*Black Box*

4.2.1.1.2 Rencana Pengujian Sistem Gudang

Tabel 4.22 Rencana Pengujian Sistem Gudang

Item Uji	Detail Pengujian	Jenis Pengujian
Login	Pengecekan data login	*Black Box*
Ubah Password	Merubah data *Password*	*Black Box*
Edit Profile	Merubah data profile	*Black Box*
Olah data produk	1. Tambah data produk 2. Ubah data produk 3. Hapus data produk	*Black Box*
Logout	Menghapus data *session login*	*Black Box*

4.2.1.1.3 Rencana Pengujian Sistem Administrasi

Daftar rencana pengujian untuk pengguna aplikasi sebagai administrasi dapat dilihat pada tabel dibawah ini.

Tabel 4.23 Rencana Pengujian Sistem Administrasi

Item Uji	Detail Pengujian	Jenis Pengujian
Login	Pengecekan data login	*Black Box*
Ubah Password	Merubah data Password	*Black Box*
Edit Profile	Merubah data profile	*Black Box*
Olah data jasa pengiriman	1. Tambah data jasa pengiriman 2. Ubah data jasa pengiriman 3. hapus data jasa pengiriman	
Olah data paket pengiriman	1. Tambah data paket pengiriman 2. Ubah data paket pengiriman 3. hapus data paket	

	pengiriman	
Olah data harga paket	1. Tambah data harga paket 2. Ubah data harga paket 3. Hapusdata harga paket	
Olah data Provinsi	1. Tambah data provinsi 2. Hapus data provinsi 3. Ubah data provinsi	*Black Box*
Olah data kota	1. Tambah data kota 2. ubah data kota 4. Hapus data kota	*Black Box*
Logout	Menghapus data *session login*	*Black Box*

4.2.1.1.4 Rencana Pengujian Sistem Pengunjung

Daftar rencana pengujian untuk pengguna aplikasi sebagai pengguna dapat dilihat pada tabel di bawah ini.

Tabel 4.24 Rencana Pengujian Sistem Pengunjung

Item Uji	**Detail Pengujian**	**Jenis Pengujian**
Daftar	1. Menampilkan halaman form pendaftaran untuk member 2. Mengisi form pendaftaran 3. Kirim pendaftaran Konfirmasi pendaftaran ke email	*Black Box*
Lupa Password	1. Pengecekan email	*Black Box*

	2. Kirim *password* baru ke email	

4.2.1.1.5 Rencana Pengujian Sistem *Member*

Tabel 4.25 Rencana Pengujian Sistem Member

Item Uji	**Detail Pengujian**	**Jenis Pengujian**
Login	Pengecekan data login yang di masukan oleh member	*Black Box*
Profil Saya	1. Merubah Data diri	*Black Box*
	2. Merubah Password	
Keranjang Belanja	1. Tambah produk ke keranjang belanja 2. Hapus produk keranjang belanja 3. Update jumlah produk (ready stok) keranjang belanja 4. Hapus jumlah produk (pesan) keranjang belanja	*Black Box*
Isi Alamat pengiriman	1. Pengisian Alamat pengiriman 2. Pengisian Jenis Pengiriman	*Black Box*
Konfirmasi Pembayaran	1. Mengkonfirmasi pembayaran	*Black Box*
Retur	1. Pilih Barang yang diretur 2. Konfirmasi pengiriman retur	*Black Box*

4.2.1.1 Kasus Dan Hasil Pengujian Alpha

Berdasarkan rencana pengujian yang disusun, maka dilakukan pengujian sebagai berikut:

4.2.1.1.1 Pengujian Data Pemimpin

Tabel 4.26 Pengujian login Pemimpin

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data login pemimpin, dengan contoh : Username : pemilik Password : 123456	Menuju halaman pemimpin	1. Memeriksa data login 2. Dapat menuju ke halaman pemimpin	**[√]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *username* dan *password*	Login gagal, tidak menuju ke halaman pemimpin	Menampilkan pesan *username* dan *password* tidak benar	**[√]** Diterima **[ ]** Ditolak
Mengisi *Username* dan *password* dengan data yang salah Contoh: *Username:*pemilik1 *Password*:12345	Login gagal, tidak menuju ke halaman pemimpin	Menampilkan pesan *username* dan *password* tidak benar	**[√]** Diterima **[ ]** Ditolak

a. Pengujian ubah data *password*

Tabel 4.27 Pengujian Rubah Password

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data rubah password pemimpin dengan contoh : password lama: 123456 coba *Password* baru: cobacoba ulangi password baru:cobacoba	Password berhasil dirubah	1. Memeriksa data password 2. Password berhasil dirubah	**[√]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan data Password	password tidak berhasil dirubah	Menampilkan pesan data password masih kosong	**[√]** Diterima [ ] Ditolak
pengisian password baru berbeda. Contoh: password coba coba1234 Password baru: coba coba ulangi password baru:cobacoba	password tidak berhasil dirubah	Menampilkan data password baru tidak sama	**[√]** Diterima [ ] Ditolak
pengisian password lama salah. Contoh: password lama: coba Pengisian *password* lama: coba1234 Password cobacoba. ulangi *password*	password tidak berhasil dirubah	Menampilkan data password lama salah	**[√]** Diterima [ ] Ditolak

baru:cobacoba.			

4.2.1.1.2 Pengujian Data Administratif

Berdasarkan rencana pengujian yang disusun, maka dilakukan pengujian untuk pengguna level administrasi, yaitu sebagai berikut:

a. Pengujian login Administrasi

Tabel 4.28 Pengujian Login Administrasi

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data login administrasi dengan contoh :	Menuju halaman administratif	1. Memeriksa data login	**[**√**]** Diterima
Username : admin Password : 123456		2. Dapat menuju ke halaman administrasi	**[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *username* dan *password*	Login gagal, tidak menuju	Menampilkan pesan	**[**√**]** Diterima

	ke halaman administrasi	*username* dan *password* tidak benar	
			[ ] Ditolak
Mengisi Username dan password dengan data yang salah Contoh: Username : adminadmin Password : qwerttyyuuii	Login gagal, tidak menuju ke halaman administrasi	Menampilkan pesan *username* dan *password* tidak benar	**[**√**]** Diterima **[ ]** Ditolak

b. Pengujian Rubah *Password*

Tabel 4.29 Pengujian Rubah *Password*

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data rubah password administrasi dengan contoh : *password* lama: 123456 *Password* baru: adminadmin. ulangi password baru: adminadmin	Password berhasil dirubah	1. Memeriksa data password	**[**√**]** Diterima
		2. Password berhasil dirubah	**[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan data *Password*	*password* tidak berhasil dirubah	Menampilkan pesan data password masih kosong	**[**√**]** Diterima [ ] Ditolak

pengisian password baru berbeda. Contoh: password lama: pass Password baru: adminqwerty ulangi password baru:adminqwerty	password tidak berhasil dirubah	Menampilkan data password baru tidak sama	**[**√**]** Diterima [ ] Ditolak
pengisian password lama salah. Contoh : password lama: pass Pengisian password lama: pass Password baru: admin.admin ulangi password baru:adminadmin	password tidak berhasil dirubah	Menampilkan data password lama salah	**[**√**]** Diterima [ ] Ditolak

c. Pengujian Rubah status Order

Tabel 4.30 Pengujian Rubah Status Pesanan

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pemilihan data status pesanan:	Status Berubah	1. Memeriksa data status 2. Statusberhasil dirubah	**[**√**]** Diterima **[ ]** Ditolak

d. Pengujian Isi no. resi pengiriman

Tabel 4.31 Pengujian isi no. resi

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Isi no resi pengiriman dengan contoh:	No resi tersimpan	1. Memeriksa data no. resi	**[**√**]** Diterima
No. resi : BDOF300425473512		2. No. resi berhasil	**[ ]** Ditolak

		tersimpan	
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi no resi dengan data yang sudah pernah dipakai. Contoh: No. Resi : BDOF300425473512	no resi tidak berhasil disimpan	Menampilkan pesan no resi tidak benar	**[**√**]** Diterima **[ ]** Ditolak

e. Pengujian Rubah status Retur

Tabel 4.34 Pengujian status retur

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pemilihan data status retur	Status Berubah	1. Memeriksa data status 2. Status berhasil dirubah	**[**√**]** Diterima **[ ]** Ditolak

f. Pengujian Isi no. resi pengiriman retur

Tabel 4.32 Pengujian Isi No. Resi Pengiriman Retur

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Isi no resi pengiriman retur dengan contoh: No. resi : BDOF300425483312	No resi tersimpan	1. Memeriksa data no. resi 2. No. resi berhasil	**[**√**]** Diterima **[ ]** Ditolak

		tersimpan	
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi no resi dengan data yang sudah pernah dipakai. Contoh No. resi : BDOF300425483312	no resi tidak berhasil disimpan	Menampilkan pesan no resi tidak benar	**[**√**]** Diterima **[ ]** Ditolak

g. Pengujian Tambah data Kategori

Tabe 4.33 Pengujian Tambah Kategori

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data kategori. Contoh: Nama kategori : T-Shirt	Menambahkan data kategori	Data kategori bertambah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data kategori.	Muncul pesan form harus diisi.	Menampilkan pesan harus diisi.	**[**√**]** Diterima **[ ]** Ditolak

h. Pengujian ubah data Kategori

Tabel 4.34 Pengujian edit data kategori

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
mengubah data kategori. Contoh: Nama kategori sebelumnya : boneka bantal Nama Kategori baru : asdfg	Mengubah data kategori	Data kategori dapat diubah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian ubah data kategori.	Muncul pesan harus *form* harus diisi.	Menampilkan pesan harus diisi.	**[**√**]** Diterima **[ ]** Ditolak

i. Pengujian Tambah data jasa pengiriman

Tabel 4.35 Pengujian tambah data jasa_pengiriman

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data jasa pengiriman. Contoh : Nama : JNE	Menambahkan data jasa_pengiriman	Data jasa_pengiriman bertambah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			

Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data jasa_pengiriman.	Muncul pesan form harus diisi.	Menampilkan pesan harus diisi.	**[**√**]** Diterima **[ ]** Ditolak

j. Pengujian ubah data jasa pengiriman

Tabel 4.36 Pengujian Ubah data jasa pengiriman

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
mengubah data jasa_pengiriman. Contoh: Nama Lama: JNE Nama baru : JNE2	Mengubah data jasa pengiriman.	Data jasa pengiriman dapat diubah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian ubah data jasa pengiriman.	Muncul pesan harus *form* harus diisi.	Menampilkan pesan harus diisi.	**[**√**]** Diterima **[ ]** Ditolak

k. Pengujian Tambah data provinsi

Tabel 4.37 Pengujian tambah data provinsi

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

Pengisian data provinsi. Contoh : Banten	Menambahkan data provinsi	Data provinsi bertambah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data provinsi.	Muncul pesan form harus diisi.	Menampilkan pesan harus diisi.	**[**√**]** Diterima **[ ]** Ditolak

l. Pengujian ubah data provinsi

Tabel 4.38 Pengujian Ubah Data Provinsi

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
mengubah data provinsi. Contoh Nama Provinsi lama : Banten Nama Provinsi baru : Bantan2	Mengubah data provinsi	Data provinsi dapat diubah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian ubah data provinsi.	Muncul pesan harus *form* harus diisi.	Menampilkan pesan harus diisi.	**[**√**]** Diterima **[ ]** Ditolak

m. Pengujian Tambah data kota

Tabel 4.39 Pengujian tambah data kota

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data kota. Contoh: Nama Provinsi : Banten Kota :Serang	Menambahkan data kota	Data kota bertambah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data kota.	Muncul pesan form harus diisi.	Menampilkan pesan harus diisi.	**[**√**]** Diterima **[ ]** Ditolak
Mengisi data kota dengan data yang tidak benar. Contoh: Provinsi belum dipilih	Muncul pesan data pengisian data tidak benar	Muncul pesan data pengisian data tidak benar	**[**√**]** Diterima **[ ]** Ditolak

n. Pengujian ubah data kota

Tabel 4.40 Pengujian Ubah data kota

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
mengubah data kota. Contoh: Nama Provinsi : Banten	Mengubah data kota	Data kota dapat diubah	**[**√**]** Diterima **[ ]** Ditolak

Kota lama : Serang Kota baru : Serang Baru			
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian ubah data kota.	Muncul pesan harus *form* harus diisi.	Menampilkan pesan harus diisi.	**[**√**]** Diterima **[ ]** Ditolak
Mengisi data kota dengan data yang salah. Contoh: Provinsi belum dipilih	Muncul pesan data pengisian data tidak benar	Muncul pesan data pengisian data tidak benar.	**[**√**]** Diterima **[ ]** Ditolak

o. Pengujian Tambah data harga paket pengiriman

Tabel 4.41 Pengujian tambah data harga paket pengiriman

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data harga paket pengiriman. Contoh : Provinsi: jawab barat Kota:Kota Bandung Jasa Pengiriman:JNE Kategori Pengiriman:REG harga paket pengiriman:8000	Menambahkan data ongkos_kirim	Data ongkos_kirim bertambah	**[**√**]** Diterima **[ ]** Ditolak

Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data harga paket pengiriman	Muncul pesan form harus diisi.	Menampilkan pesan harus diisi.	**[√]** Diterima **[ ]** Ditolak
Mengisi data harga paket pengiriman dengan data yang tidak benar Contoh : harga paket pengiriman =asdfghj	Muncul pesan data pengisian data tidak benar	Muncul pesan data pengisian data tidak benar	**[√]** Diterima **[ ]** Ditolak

p. Pengujian ubah data paket pengiriman

Tabel 4.42 Pengujian Ubah Data paket pengiriman

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
mengubah data harga paket pengiriman Contoh: harga paket pengiriman lama:8000 harga paket pengiriman baru : 10000	Mengubah data ongkos_kirim.	Data ongkos_kirim dapat diubah.	**[√]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian ubah data harga paket pengiriman.	Muncul pesan harus *form* harus diisi.	Menampilkan pesan harus diisi.	**[√]** Diterima **[ ]** Ditolak

Mengisi data harga paket pengiriman dengan data yang salah Contoh : harga paket pengiriman =asdfghjk	Muncul pesan data pengisian data tidak benar	Muncul pesan data pengisian data tidak benar.	**[**√**]** Diterima **[ ]** Ditolak

4.2.1.1.3 Pengujian Data Gudang

Berdasarkan rencana pengujian yang disusun, maka dilakukan pengujian untuk pengguna gudang, yaitu sebagai berikut:

a. Pengujian login gudang

Tabel 4.43 Pengujian Login Gudang

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data login gudang,dengan contoh : Username : gudang Password : 123456	Menuju halaman gudang	1. Memeriksa data login 2. Dapat menuju ke halaman gudang	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *username* dan *password*	Login gagal, tidak menuju ke halaman gudang	Menampilkan pesan *username* dan *password* tidak benar	**[**√**]** Diterima **[ ]** Ditolak
Mengisi Username dan password dengan data yang salah	Login gagal, tidak menuju ke halaman gudang	Menampilkan pesan *username* dan *password* tidak benar	**[**√**]** Diterima **[ ]** Ditolak

b. Pengujian ubah *password* Gudang

Tabel 4.44 Pengujian ubah *Password*Gudang

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data rubah password Gudang,dengan contoh : password lama: 123456 Password baru: gudang123. ulangi password baru:gudang123	Password berhasil dirubah	1. Memeriksa data password 2. Password berhasil dirubah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan data Password	password tidak berhasil dirubah	Menampilkan pesan data password masih kosong	**[**√**]** Diterima [ ] Ditolak
pengisian password baru berbeda. Contoh: password lama: gudang. Password baru: gudang123. ulangi password baru:gudang1	password tidak berhasil dirubah	Menampilkan data password baru tidak sama	**[**√**]** Diterima [ ] Ditolak
pengisian password lama salah. Contoh : password lama:	password tidak berhasil dirubah	Menampilkan data password lama salah	**[**√**]** Diterima [ ] Ditolak

gudang. Pengisian password lama: gudang1 Password baru: gudang123. ulangi password baru:gudang123			

c. Pengujian Tambah data produk

Tabel 4.45 Pengujian tambah data Produk

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
data produk Contoh : Nama Produk : kura-kura Harga : Rp. 60.000 Kategori : boneka Berat : 0.5 Kemampuan	Menambahkan data produk	Data produk bertambah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian data produk.	Muncul pesan harus diisi.	Menampilkan pesan harus diisi.	**[**√**]** Diterima **[ ]** Ditolak

Mengisi data produk dengan data yang tidak benar Contoh : stok : -1	Muncul pesan data pengisian data tidak benar	Muncul pesan data pengisian data tidak benar	**[**√**]** Diterima **[ ]** Ditolak

d. Pengujian Edit data produk

Tabel 4.46 Pengujian Edit Data produk

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
mengubah data produk. Contoh: Nama Produk : kura-kura Harga sebelumnya: Rp. 60.000 Harga setelahnya: Rp. 100.000	Mengubah data produk	Data produk dapat diubah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form pengisian ubah data produk.	Muncul pesan harus *form* harus diisi.	Menampilkan pesan harus diisi.	**[**√**]** Diterima **[ ]** Ditolak
Mengisi data produk dengan data yang salah Contoh :	Muncul pesan data pengisian data tidak benar	Muncul pesan data pengisian data tidak benar.	**[**√**]** Diterima **[ ]** Ditolak

diskon:120%			

a. Pengujian lupa password

Tabel 4.47 Pengujian Lupa password

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi email sesuai dengan email pelangan dengan contoh Email : belvanuhaekaputra@yahoo.com	1. Mengecek email valid atau tidak 2. Mengirimkan password baru jika email valid 3. Jika email tidak valid maka akan muncul pesan email tidak terdaftar	1. Mengecek email valid atau tidak 2. Mengirimkan password baru jika email valid 3. Jika email tidak valid maka akan muncul pesan email tidak terdaftar	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Mengosongkan form isian email atau memasukan email yang tidak terdaftar Contoh : Email:alvhiee@yahoo.com	1. Muncul pesan bahwa email tidak terdaftar 2. Password baru tidak terkirim ke email yang di inputkan	Muncul pesan bahwa email tidak terdaftar	**[**√**]** Diterima **[ ]** Ditolak

4.2.1.1.4 Pengujian data Member

Berdasarkan rencana pengujian yang disusun, maka dilakukan pengujian untuk pengguna level member, yaitu sebagai berikut:

a. Login *Member*

Tabel 4.48 Pengujian login *member*

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Pengisian data login *member*, dengan contoh : Email: bellvanuhaekaputra@yahoo.com Password : qwerty	Menuju halaman pelanggan	1. Memeriksa data login 2. Dapat menuju ke halaman pelanggan	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan *username* dan *password*	Login gagal, tidak menuju ke halaman pelanggan	Menampilkan pesan *username* dan *password* tidak benar	**[**√**]** Diterima **[ ]** Ditolak
Mengisi *Username* dan *password* dengan data yang salah	Login gagal, tidak menuju ke halaman pelanggan	Menampilkan pesan *username* dan *password* tidak benar	**[**√**]** Diterima **[ ]** Ditolak

b. Pengujian profile saya

Tabel 4.49 Pengujian Profile saya

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

Mengisi data profil Contoh: Nama lama:belva Nama baru : belva nuha	Data profil pelangganberhasil di ubah	Data profil pelangganberhasil di ubah	**[**√**]** Diterima **[ ]** Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan data profil	Data profil pelanggangagal di ubah dan menampilkan pesan kesalahan pengisian form	Data profil pelanggangagal di ubah dan menampilkan pesan kesalahan pengisian form	**[**√**]** Diterima **[ ]** Ditolak

c. Pengujian Update jumlah produk (ready stok) keranjang belanja

Tabel 4.50 Pengujian Update jumlah produk (*ready* stok) keranjang belanja

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Merubah kuantitas produk (ready stok) yang akan dibeli. Kemudian klik update.	jumlah produk(ready stok) bertambah	jumlah produk(ready stok) dapat bertambah	**[**√**]** Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**

Merubah kuantitas produk (ready stok) yang akan dibeli dengan data yang salah. Kemudian klik update. Contoh : jumlah=-1	jumlah produk(ready stok) tidak bertambah	Muncul pesan format data salah	**[√]** Diterima [ ] Ditolak

d. Pengujian Pengisian alamat pengiriman

Tabel 4.51 Pengujian Pengisian alamat pengiriman

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi alamat pengiriman	Alamat pengiriman terisi, menuju halaman selanjutnya	Dapat menuju halaman selanjutnya	**[√]** Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan alamat pengiriman	tidak dapat menuju halaman selanjutnya	ada pesan data alamat pengiriman kosong	**[√]** Diterima [ ] Ditolak
Mengisi alamat pengiriman dengan data yang salah Kode pos=abcde	tidak dapat menuju halaman	ada pesan data alamat pengiriman	**[√]** Diterima [ ] Ditolak

	selanjutnya	salah	

e. Pengujian Konfirmasi pembayaran

Tabel 4.52 Pengujian Konfirmasi Pembayaran

Kasus dan Data Hasil Uji (data benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Isi form konfirmasi pembayaran Contoh: Nama pembayar : belva	Menampilkan pesan konfirmasi pembayaran telah diterima	Dapat menampilkan pesan konfirmasi pembayaran telah diterima	**[√]** Diterima [ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan form konfirmasi pembayaran	Menampilkan pesan form harus diisi	Dapat Menampilkan pesan form harus diisi	**[√]** Diterima [ ] Ditolak

f. Pengujian Konfirmasi Pengiriman Retur

Tabel 4.53 Pengujian Konfirmasi Pengiriman Retur

Kasus dan Data Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengisi data konfirmasi	Konfirmasi retur	Konfirmasi	**[√]** Diterima

pengiriman retur. Contoh: Jasa pengiriman retur : JNE	berhasil	retur berhasil	[ ] Ditolak
Kasus dan Data Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Mengosongkan data konfirmsi pengiriman	Konfirmasi retur tidak berhasil	ada pesan data konfirmasi masih kosong	**[√]** Diterima [ ] Ditolak

4.2.1.3 Kesimpulan Hasil Pengujian Alpha

Dari hasil pengujian yang dilakukan dapat diketahui bahwa fungsional dari aplikasi dapat berjalan sesuai dengan yang diharapkan dan hasil pengujian dirasakan sudah sesuai dengan fungsional yang dibutuhkan.

4.2.2 Pengujian *Beta*

Pengujian *beta* merupakan pengujian yang dilakukan dengan beruan untuk mengetahui sejauh mana kualitas sistem.

1. Lingkungan pegawai toko moon dengan cara wawancara
2. Lingkungan masyarakat umum dengan cara kuesioner

4.2.2.1 Pengujian Beta untuk Pengunjung (Kuesioner)

1. Apakah anda setuju dengan *web* ini mudah pengoprasiannya bagi pengunjung baru?
2. Apakah anda setuju jika fitur-fitur dalam web ini tidak asing lagi baigi pengunjung baru?

3. Apakah anda setuju jika pengunjung membuka aplikasi akan mendapatkan informasi mengenai produk?
4. Apakah anda setuju pencarian produk dalam *website* ini mudah?
5. Apakah anda setuju bagi para pengunjung jika inig jadi member aplikasi ini harus mendaftar terlebih dahulu?

Berdasarkan data hasil kuesioner tersebut, dapat dicari persentasemasing-masing jawaban dengan menggunakan rumus dibawah ini :

$$Y = \frac{X}{skorideal} \times 100\%$$

Keterangan :

X = Jumlah frekuensi dikalikan dengan nilai kategori jawaban ($\sum$ = N x R)

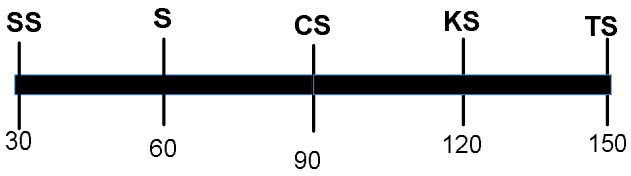
N = Nilai dari setiap jawaban

R = Frekuensi

Skor ideal = Nilai tertinggi dikalikan dengan jumlah sampel (5 x 30 = 150)

Y = Nilai persentase yang dicari

Setelah diperoleh nilai Y maka interpretasi nilai dapat dilihat sebagai berikut :



Berikut ini adalah hasil persentase masing-masing jawaban dari lima pertanyaan yang sudah dihitung nilainya dengan menggunakan rumus diatas.

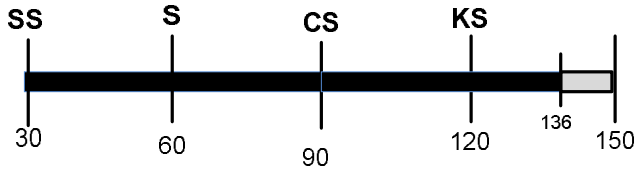
1. Apakah anda setuju weh ini mudah pengoprasiannya bagi pengunjung baru?

Tabel 4.54 Hasil Kuesioner Pertanyaan No. 1

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
1	Sangat setuju	5	20	100
	Setuju	4	6	24

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
	Cukup Setuju	3	4	12
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah				136

$$Y = \frac{117}{30 \times 5} \times 100\% = 90{,}6\%$$

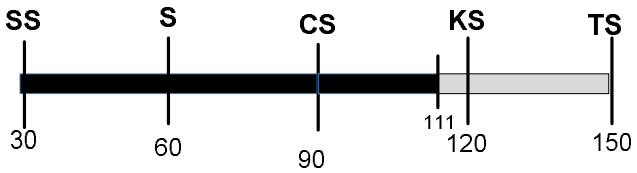


2. Apakah anda setuju jika fitur-fitur dalam web ini tidak asing lagi bagi pengnjung baru?

Tabel 4.55 Hasil Kuesioner Pertanyaan No. 2

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
2	Sangat setuju	5	3	15
	Setuju	4	15	60
	Cukup Setuju	3	12	36
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			30	111

$$Y = \frac{111}{30 \times 5} \times 100\% = 74\%$$

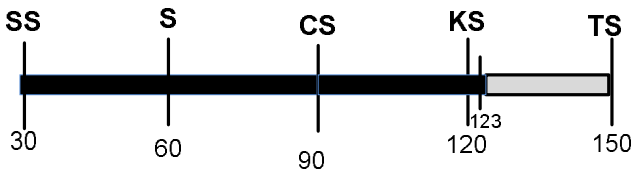


3. Apakah anda setuju jika pengunjung membuka aplikasi akan mendapatka informasi mengenai produk?

Tabel 4.56 Hasil Kuesioner Pertanyaan No. 3

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
3	Sangat setuju	5	3	15
	Setuju	4	27	108
	Cukup Setuju	3	0	0
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			30	123

$$Y = \frac{119}{30 \times 5} \times 100\% = 82\%$$

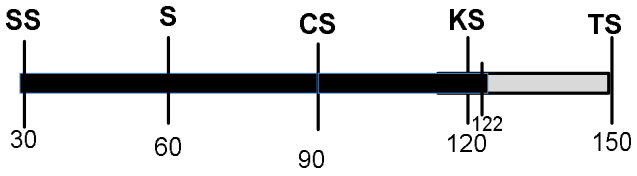


4. Apakah anda setuju pencarian produk dalam *website* ini mudah?

Tabel 4.57 Hasil Kuesioner Pertanyaan No. 4

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
4	Sangat setuju	5	7	35
	Setuju	4	18	72
	Cukup Setuju	3	5	15
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			30	122

$$Y = \frac{122}{30 \times 5} \times 100\% = 81\%$$

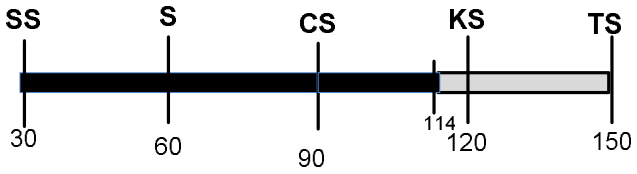


5. Apakah anda setuju bagi para pengunjung jika ingin jadi member aplikasi ini harus mendaftar terlebih dulu?

Tabel 4.56 Hasil Kuesioner Pertanyaan No. 5

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
4	Sangat setuju	5	1	5
	Setuju	4	22	88
	Cukup Setuju	3	7	21
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			30	114

$$Y = \frac{114}{30 \times 5} \times 100\% = 76\%$$



4.2.2.2 Pengujian Beta untuk *Member* (Kuesioner)

Kuesioner ini terdiri dari lima pertanyaan (contoh kuesioner dapat dilihat pada lampiran) dengan menggunkan skala *likert* dengan skala satu sampai lima, dengan jumlah responden sebanyak 30 responden, dengan ketentuan pertanyaan dan skala sebagai berikut :

1. Apakah anda setuju dengan adanya web ini akan memudahkan pelanggan untuk mendapatkan produk?

2. Apakah anda setuju dengan aplikasi e-commerce ini akan memudahkan pelanggan dalam mendapatkan informasi mengenai produk?
3. Apakah anda setuju dengan aplikasi e-commerce ini akan memudahkan pelanggan melakukan transaksi pembelian?
4. Apakah anda setuju kalau web ini mudah digunakan?
5. Apakah anda setuju jika web ini mudah untuk di akses?

Berdasarkan data hasil kuesioner tersebut, dapat dicari persentasemasing-masing jawaban dengan menggunakan rumus dibawah ini :

$$Y = \frac{X}{skorideal} \times 100\%$$

Keterangan :

X = Jumlah frekuensi dikalikan dengan nilai kategori jawaban ($\sum$ = N x R)

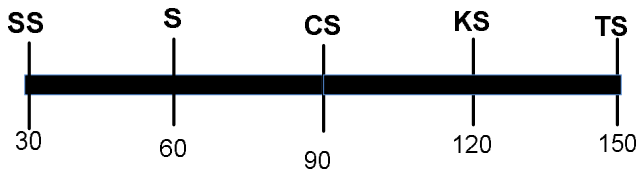
N = Nilai dari setiap jawaban

R = Frekuensi

Skor ideal = Nilai tertinggi dikalikan dengan jumlah sampel (5 x 30 = 150)

Y = Nilai persentase yang dicari

Setelah diperoleh nilai Y maka interpretasi nilai dapat dilihat sebagai berikut :



Berikut ini adalah hasil persentase masing-masing jawaban dari empat pertanyaan yang sudah dihitung nilainya dengan menggunakan rumus diatas.

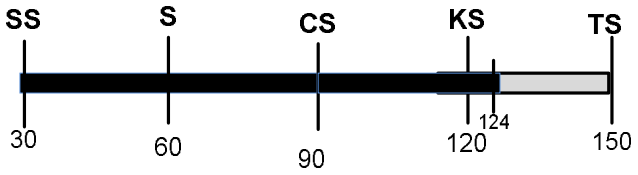
1. Apkah anda setuju dengan adanya web ini akan memudahkan pelanggan untuk mendapatkan produk?

Tabel 4.57 Hasil Kuesioner Pertanyaan No. 1

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
1	Sangat setuju	5	5	25
	Setuju	4	24	96
	Cukup Setuju	3	1	3
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah				124

$$Y = \frac{124}{30 \times 5} \times 100\% = 82{,}6\%$$

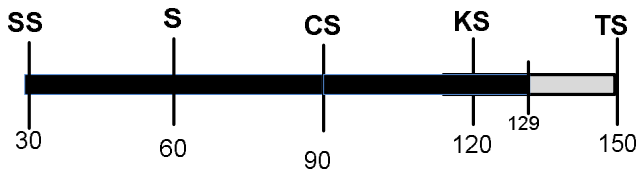


2. Apakah anda setuju dengan apikasi ecommerce ini akan memudahkan pelanggan dalam mendapatkan informasi mengenai produk?

Tabel 4.58 Hasil Kuesioner Pertanyaan No. 2

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
2	Sangat setuju	5	9	45
	Setuju	4	21	84
	Cukup Setuju	3	0	0
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			30	129

$$Y = \frac{129}{30 \times 5} \times 100\% = 86\%$$

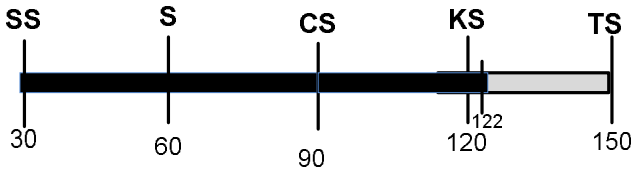


3. Apakah anda setuju dengan adanya adanya aplikasi e-commerce ini akan memudahkan pelanggan dalam melakukan transaksi pembelian?

Tabel 4.59 Hasil Kuesioner Pertanyaan No. 3

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
3	Sangat setuju	5	7	35
	Setuju	4	20	80
	Cukup Setuju	3	3	9
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			30	122

$$Y = \frac{122}{30 \times 5} \times 100\% = 81{,}3\%$$



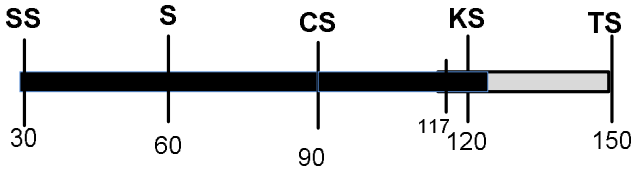
4. Apakah anda setuju kalaw web ini mudah digunakan?

Tabel 4.60 Hasil Kuesioner Pertanyaan No. 4

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
4	Sangat setuju	5	0	0
	Setuju	4	27	108

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
	Cukup Setuju	3	3	9
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			30	117

$$Y = \frac{117}{30 \times 5} \times 100\% = 78\%$$

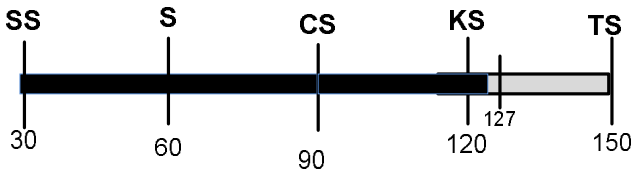


5. Apakah anda setuju jika web ini mudah untuk di akses?

Tabel 4.61 Hasil Kuesioner Pertanyaan No. 5

Pertanyaan	Keterangan	Skala (N)	Responden (R)	N x R
4	Sangat setuju	5	10	50
	Setuju	4	17	68
	Cukup Setuju	3	3	9
	Kurang Setuju	2	0	0
	Tidak Setuju	1	0	0
Jumlah			30	127

$$Y = \frac{127}{30 \times 5} \times 100\% = 84{,}7\%$$



4.2.2.3 Pengujian Beta untuk Pemimpin (Wawancara)

Wawancara dilakukan dengan menggunakan teknik kualitatif melalui wawancara tak berstruktur atau terbuka.

a. Apakah setuju dengan adanya *website* yang dibangun ini dapat membantu dalam penerimaan laporan produk dan transaksi jual beli?
   Jawaban : Tentu saya sangat setuju
b. Apakah anda setuju jika tampilan *website* ini tidak asing lagi bagi anda?
   Jawaban : Ya, karena fitur-fitur di dalam *website* tidak ribet dan tidak aneh-aneh.
c. Apakah anda setuju jika aplikasi ini menarik?
   Jawaban : Ya, saya setuju

4.2.2.4 Pengujian Beta untuk Administrasi (Wawancara)

a. Apakah anda setuju jika aplikasi ini memudahkan dalam melakukan transaksi dengan pelanggan?
   Jawaban : Ya, saya sangat setuju karena dapat melakukan transaksi kapan aja dan dimana saja, tanpa saya harus ada di toko.
b. Apakah anda setuju dengan adanya *web e-commerce* ini sangat membantu dalam proses pekerjaan anda?
   Jawaban : Ya, saya sangat setuju sekali
c. Apakah anda setuju *web* ini sangat mudah untuk di aksesnya?
   Jawaban : Ya, saya setuju karena tidak terpaku di satu tempat untuk mengakses aplikasi ini kapan pun dan di mana pun bisa mengakses aplikasi ini.
d. Apakah Anda setuju dengan adanya *web e-commerce* ini memudahkan dalam pembuatan laporan mengenai transaksi?
   Jawaban: Ya, saya setuju, ketika melakukan transaksi jual beli di aplikasi ini maka pemilik toko dapat langsung melihat laporan-laporan dari setiap transaksi tanpa ada batas.

4.2.2.5 Pengujian Beta untuk Gudang (Wawancara)

a. Apakah anda setuju jika aplikasi ini membantu dalam pekerjaan anda?
   Jawaban : Ya, saya setuju

b. Apakah anda setuju dengan adanya *web e-commerce* ini membantu dalam pembuatan laporan mengenai produk?

   Jawaban : Ya, saya setuju, karena saya tinggal meng-update produk baru maka laporan dapat langsg di akses oleh pemilik toko

c. Apakah anda setuju jika *web* ini mudah di gunakan?

   Jawaban : Tentu saya setuju, karena fitur yang digunakan dalam aplikasi ini sudah tidak asing lagi untuk saya.

d. Apakah anda setuju jika fitur di dalam aplikasi ini tidak asing lg untuk anda?

   Jawaban : Ya, saya sangat setuju

4.2.2.6 Kesimpulan Pengujian Beta

Berdasarkan hasil wawancara yang telah di lakukan, dan telah dipaparkan diatas diperoleh dari pengujian *betha* di lingkungan toko Moon dan lingkungan masyarakat umum, maka aplikasi ini sesuai dengan tujuan yaitu: memudahkan pelanggan dalam melakukan transaksi pembelian dan mudah dalam mendapatkan informasi mengenai produk, memudahkan bagian gudang dalam pengolahan data produk dan dalam pembuatan laporan produk, memudahkan bagian administrasi dalam melakukan transaksi dengan pelanggan dan juga dalam pembuatan laporannya, dan memudahkan pemilik toko dalam penerimaan laporan mengenai produk dan transaksi