BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Tahap implementasi merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan tahap dimana sistem siap untuk dioperasikan, yang terdiri dari penjelasan mengenai lingkungan implementasi, dan implementasi program.

4.1.1 Perangkat Lunak Yang Digunakan

Spesifikasi perangkat lunak yang digunakan dalam membangun sistem penjualan *online website Distroblackshoes.com* adalah sebagai berikut:

Tabel 4.1 Implementasi spesifikasi perangakat lunak

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Linux
2	Bahasa Pemrograman	PHP
3	*DBMS*	MySQL
4	Web *browser*	Mozilla firefox 4.0
5	*Code editor*	Adobe Dreamweaver CS3

4.1.2 Perangkat Keras Yang Digunakan

Spesifikasi perangkat keras yang digunakan dalam pembangunan sistem penjualan di*website Distroblackshoes* adalah tercantum pada tabel 4.2 sebagai berikut :

Tabel 4.2 Implementasi spesifikasi perangkat keras

Perangkat	Spesifikasi
Processor	Kecepatan 1.6 GHz
RAM	1 Gb
Harddisk	320 Gb
Monitor	LCD 14”
VGA	256 Mb

4.2 Implementasi

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

1. Tabel barang

```
CREATE TABLE IF NOT EXISTS `barang` (
  `id_barang` int(11) NOT NULL AUTO_INCREMENT,
  `id_kategori` int(5) NOT NULL,
  `nama_barang` varchar(50) NOT NULL,
  `harga_barang` int(11) NOT NULL,
  `diskon_barang` int(2) NOT NULL,
  `rating_barang` int(2) NOT NULL,
  `voterrating_barang` int(11) NOT NULL,
  `viewcounter_barang` int(11) NOT NULL,
  PRIMARY KEY (`id_barang`),
  KEY `id_barang` (`id_barang`),
  KEY `id_kategori` (`id_kategori`),
CONSTRAINT `barang_ibfk_1` FOREIGN KEY (`id_kategori`)
```

```
          REFERENCES `kategori` (`id_kategori`)
)  ENGINE=InnoDB  DEFAULT  CHARSET=latin1  COMMENT='Tabel
Barang';
```

2. Tabel *detail* barang

```
    CREATE TABLE IF NOT EXISTS `detailbarang` (

      `id_detailbarang` int(11) NOT NULL AUTO_INCREMENT,

      `id_barang` int(11) NOT NULL,

      `id_warna` int(5) NOT NULL,

      `id_ukuran` int(5) DEFAULT NULL,

      `tanggal_detailbarang` date NOT NULL,

      `stok_detailbarang` int(5) NOT NULL,

      `berat_detailbarang` float(4,2) NOT NULL,

      UNIQUE KEY `id_detailbarang_2` (`id_detailbarang`),

      KEY `id_barang` (`id_barang`),

      KEY `id_warna` (`id_warna`),

      KEY `id_ukuran` (`id_ukuran`)

      CONSTRAINT   `detailbarang_ibfk_3`   FOREIGN   KEY
          (`id_ukuran`)        REFERENCES        `ukuran`
          (`id_ukuran`),

      CONSTRAINT   `detailbarang_ibfk_1`   FOREIGN   KEY
          (`id_barang`)        REFERENCES        `barang`
          (`id_barang`),

      CONSTRAINT   `detailbarang_ibfk_2`   FOREIGN   KEY
          (`id_warna`) REFERENCES `warna` (`id_warna`)

)  ENGINE=InnoDB  DEFAULT  CHARSET=latin1  COMMENT='Tabel
Detail Barang';
```

3. Tabel *detail*_pembelian

```
CREATE TABLE IF NOT EXISTS `detail_pembelian` (
  `id_detail_pembelian` int(11) NOT NULL AUTO_INCREMENT,
  `id_pembelian` int(11) DEFAULT NULL,
  `id_detailbarang` int(11) NOT NULL,
  `harga` int(11) NOT NULL,
  `jumlah` int(5) NOT NULL,
  `berat` float(4,2) NOT NULL,
  `session_id` varchar(32) NOT NULL,
  PRIMARY KEY (`id_detail_pembelian`),
  KEY `id_pembelian` (`id_pembelian`),
  KEY `id_detailbarang` (`id_detailbarang`),
  CONSTRAINT `detail_pembelian_ibfk_2` FOREIGN KEY
      (`id_detailbarang`) REFERENCES `detailbarang`
      (`id_detailbarang`),
  CONSTRAINT `detail_pembelian_ibfk_1` FOREIGN KEY
      (`id_pembelian`) REFERENCES `pembelian`
      (`id_pembelian`))
ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Tabel
      Pembelian';
```

4. Tabel *detail*_retur

```
CREATE TABLE IF NOT EXISTS `detail_retur` (
  `id_detailretur` int(11) NOT NULL AUTO_INCREMENT,
  `id_retur` int(11) NOT NULL,
```

```
      `id_detailbarang` int(11) NOT NULL,

      `qty` int(5) NOT NULL,

      `ket` varchar(80) NOT NULL,

      `tindakan` varchar(20) DEFAULT NULL,

      UNIQUE KEY `id_detailretur` (`id_detailretur`),

      KEY `id_retur` (`id_retur`,`id_detailbarang`),

      KEY `id_retur_2` (`id_retur`),

      KEY `id_detailbarang` (`id_detailbarang`),

     CONSTRAINT     `detail_retur_ibfk_2`     FOREIGN     KEY
          (`id_detailbarang`)    REFERENCES    `detailbarang`
          (`id_detailbarang`),

     CONSTRAINT     `detail_retur_ibfk_1`     FOREIGN     KEY
          (`id_retur`) REFERENCES `retur` (`id_retur`)

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

5. Tabel gambar

```
    CREATE TABLE IF NOT EXISTS `gambar` (

      `id_gambar` int(11) NOT NULL AUTO_INCREMENT,

      `id_barang` int(11) NOT NULL,

      `path_gambar` varchar(200) NOT NULL,

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

      `path_gambar2` varchar(200) DEFAULT NULL,

      PRIMARY KEY (`id_gambar`),

      KEY `id_barang` (`id_barang`),

     CONSTRAINT  `gambar_ibfk_1`  FOREIGN  KEY  (`id_barang`)
          REFERENCES `barang` (`id_barang`)

    )  ENGINE=InnoDB  DEFAULT  CHARSET=latin1  COMMENT='Tabel
          Gambar';
```

6. Tabel hubungi

```
    CREATE TABLE IF NOT EXISTS `hubungi` (
      `id_hubungi` int(11) NOT NULL AUTO_INCREMENT,
      `id_petugas` int(5) DEFAULT NULL,
      `nama_pelanggan` varchar(50) NOT NULL,
      `email_pelanggan` varchar(50) NOT NULL,
      `keterangan` varchar(600) NOT NULL,
      `tgl_hubungi` date NOT NULL,
      PRIMARY KEY (`id_hubungi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

7. Tabel jasapengiriman

```
    CREATE TABLE IF NOT EXISTS `jasapengiriman` (
      `id_jasapengiriman` int(5) NOT NULL AUTO_INCREMENT,
      `nama_jasapengiriman` varchar(50) NOT NULL,
      `deskripsi_jasapengiriman` varchar(50) NOT NULL,
      PRIMARY KEY (`id_jasapengiriman`),
      UNIQUE KEY `nama_jasapengiriman`
          (`nama_jasapengiriman`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

8. Tabel jenispengiriman

```
    CREATE TABLE IF NOT EXISTS `jenispengiriman` (
      `id_jenispengiriman` int(5) NOT NULL AUTO_INCREMENT,
      `id_jasapengiriman` int(5) NOT NULL,
      `nama_jenispengiriman` varchar(50) NOT NULL,
      `deskripsi_jenispengiriman` varchar(50) NOT NULL,
```

```
      PRIMARY KEY (`id_jenispengiriman`),

      KEY `id_jasapengiriman` (`id_jasapengiriman`),

      KEY `id_jasapengiriman_2` (`id_jasapengiriman`),

      CONSTRAINT   `jenispengiriman_ibfk_1`   FOREIGN   KEY
          (`id_jasapengiriman`) REFERENCES `jasapengiriman`
          (`id_jasapengiriman`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Tabel Jenis
Pengiriman';
```

9. Tabel kategori

```
    CREATE TABLE IF NOT EXISTS `kategori` (

      `id_kategori` int(5) NOT NULL AUTO_INCREMENT,

      `nama_kategori` varchar(50) NOT NULL,

      PRIMARY KEY (`id_kategori`),

      UNIQUE KEY `nama_kategori` (`nama_kategori`)
)   ENGINE=InnoDB   DEFAULT   CHARSET=latin1   COMMENT='Tabel
Kategori';
```

10. Tabel kota

```
    CREATE TABLE IF NOT EXISTS `kota` (

      `id_kota` int(5) NOT NULL AUTO_INCREMENT,

      `id_provinsi` int(5) NOT NULL,

      `nama_kota` varchar(50) NOT NULL,

      PRIMARY KEY (`id_kota`),

      KEY `id_provinsi` (`id_provinsi`),

     CONSTRAINT  `kota_ibfk_1`  FOREIGN  KEY  (`id_provinsi`)
          REFERENCES `provinsi` (`id_provinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

11. Tabel kurs

```
CREATE TABLE IF NOT EXISTS `kurs` (
  `id_kurs` int(5) NOT NULL,
  `dollar` float(6,2) NOT NULL,
  `id_petugas` int(5) NOT NULL,
  PRIMARY KEY (`id_kurs`),
  KEY `id_petugas` (`id_petugas`),
CONSTRAINT `kurs_ibfk_1` FOREIGN KEY (`id_petugas`)
REFERENCES `petugas` (`id_petugas`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

12. Tabel *member*

```
    CREATE TABLE IF NOT EXISTS `member` (
      `id_member` int(11) NOT NULL AUTO_INCREMENT,
      `id_kota` int(5) NOT NULL,
      `nama_member` varchar(50) NOT NULL,
      `alamat_member` varchar(80) NOT NULL,
      `ttl_member` date NOT NULL,
      `telp_member` varchar(20) NOT NULL,
      `kodepos_member` varchar(6) NOT NULL,
      `email_member` varchar(50) NOT NULL,
      `password_member` varchar(32) NOT NULL,
      `verificationcode_member` varchar(32) NOT NULL,
      `status_member` enum('0','1') NOT NULL,
      PRIMARY KEY (`id_member`),
      UNIQUE KEY `email_member` (`email_member`),
      KEY `id_kota` (`id_kota`),
      CONSTRAINT `member_ibfk_1` FOREIGN KEY (`id_kota`)
          REFERENCES `kota` (`id_kota`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Tabel
Member';
```

13. Tabel ongkoskirim

```
    CREATE TABLE IF NOT EXISTS `ongkoskirim` (

      `id_ongkoskirim` int(11) NOT NULL AUTO_INCREMENT,

      `id_kota` int(5) NOT NULL,

      `id_jenispengiriman` int(5) NOT NULL,

      `harga_ongkoskirim` int(11) NOT NULL,

      PRIMARY KEY (`id_ongkoskirim`),

      KEY `id_kota_2` (`id_kota`),

      KEY `id_jenispengiriman` (`id_jenispengiriman`),

      CONSTRAINT     `ongkoskirim_ibfk_2`     FOREIGN     KEY
          (`id_jenispengiriman`)                   REFERENCES
          `jasapengiriman` (`id_jasapengiriman,

      CONSTRAINT     `ongkoskirim_ibfk_1`     FOREIGN     KEY
          (`id_kota`) REFERENCES `kota` (`id_kota`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Tabel Ongkos
Kirim';
```

14. Tabel pembelian

```
    CREATE TABLE IF NOT EXISTS `pembelian` (

      `id_pembelian` int(11) NOT NULL AUTO_INCREMENT,

      `id_member` int(11) NOT NULL,

      `jenis_pembayaran`
          enum('','paypal','atm','internetbanking','rekenin
          g','belumbayar') NOT NULL DEFAULT 'belumbayar',

      `status_pengiriman`
          enum('dipesan','dibayar','lunas','dikirim','diter
          ima') NOT NULL,

      `nama_pemesan` varchar(50) NOT NULL,

      `alamat_pemesan` varchar(80) NOT NULL,
```

```
      `email_pemesan` varchar(50) NOT NULL,
      `no_telp_pemesan` varchar(20) NOT NULL,
      `kodepos_pemesan` varchar(6) NOT NULL,
      `nama_pengirim_transfer` varchar(50) NOT NULL,
      `total_kirim` float(10,2) NOT NULL,
      `tgl_beli` datetime NOT NULL,
      `tgl_bayar` datetime NOT NULL,
      `tgl_terima` datetime NOT NULL,
      `no_resi` varchar(32) NOT NULL,
      `id_rekening` int(5) NOT NULL,
      `id_ongkoskirim` int(5) NOT NULL,
      `status_pemesanan` enum('ok','cancel') NOT NULL,
      `nama_bank_pemesan` varchar(50) NOT NULL,
      `session_id` varchar(32) NOT NULL,
      PRIMARY KEY (`id_pembelian`),
      KEY `id_member` (`id_member`),
      KEY `id_member_3` (`id_member`),
      KEY `id_rekening` (`id_rekening`),
      KEY `id_ongkoskirim` (`id_ongkoskirim`),
    CONSTRAINT      `pembelian_ibfk_3`      FOREIGN      KEY
          (`id_ongkoskirim`)    REFERENCES    `ongkoskirim`
          (`id_jenispengiriman`),
    CONSTRAINT `pembelian_ibfk_1` FOREIGN KEY (`id_member`)
          REFERENCES `member` (`id_member`),
    CONSTRAINT      `pembelian_ibfk_2`      FOREIGN      KEY
          (`id_rekening`)       REFERENCES       `rekening`
          (`id_rekening`)
)  ENGINE=InnoDB  DEFAULT  CHARSET=latin1  COMMENT='Tabel
pembelian';
```

15. Tabel petugas

```
    CREATE TABLE IF NOT EXISTS `petugas` (
      `id_petugas` int(5) NOT NULL AUTO_INCREMENT,
      `username` varchar(20) NOT NULL,
      `nama_petugas` varchar(50) NOT NULL,
      `alamat_petugas` varchar(50) NOT NULL,
      `email_petugas` varchar(50) NOT NULL,
      `telp_petugas` varchar(20) NOT NULL,
      `password_petugas` varchar(32) NOT NULL,
      `level` enum('admin','gudang','kasir') NOT NULL,
      PRIMARY KEY (`id_petugas`),
      UNIQUE KEY `email_admin` (`email_petugas`),
      UNIQUE KEY `username` (`username`)
)  ENGINE=InnoDB  DEFAULT  CHARSET=latin1  COMMENT='Tabel
petugas';
```

16. Tabel provinsi

```
    CREATE TABLE IF NOT EXISTS `provinsi` (
      `id_provinsi` int(5) NOT NULL AUTO_INCREMENT,
      `nama_provinsi` varchar(50) NOT NULL,
      PRIMARY KEY (`id_provinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

17. Tabel rekening

```
    CREATE TABLE IF NOT EXISTS `rekening` (
      `id_rekening` int(5) NOT NULL AUTO_INCREMENT,
      `nama_rekening` varchar(50) NOT NULL,
```

```
  `bank_rekening` varchar(20) NOT NULL,
  `cabang_rekening` varchar(20) NOT NULL,
  `no_rekening` varchar(32) NOT NULL,
  `gambar_rekening` varchar(255) NOT NULL,
  PRIMARY KEY (`id_rekening`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Tabel
    Rekening';
```

18. Tabel retur

```
CREATE TABLE IF NOT EXISTS `retur` (
  `id_retur` int(11) NOT NULL AUTO_INCREMENT,
  `id_pembelian` int(11) NOT NULL,
  `id_member` int(11) NOT NULL,
  `tgl_kirim` date DEFAULT NULL,
  `resi_kirim` varchar(20) DEFAULT NULL,
  `kurir_kirim` int(5) DEFAULT NULL,
  `tgl_retur` date DEFAULT NULL,
  `resi_retur` varchar(20) DEFAULT NULL,
  `kurir_retur` int(5) DEFAULT NULL,
  `tgl_terima` date DEFAULT NULL,
  `status_retur` varchar(20) NOT NULL,
  `bank` varchar(20) DEFAULT NULL,
  `no_rek` varchar(20) DEFAULT NULL,
  `nama` varchar(30) DEFAULT NULL,
  UNIQUE KEY `id_retur` (`id_retur`),
  KEY `id_pembelian` (`id_pembelian`),CONSTRAINT
    `retur_ibfk_1` FOREIGN KEY (`id_pembelian`)
    REFERENCES `pembelian` (`id_pembelian`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

19. Tabel ukuran

```
    CREATE TABLE IF NOT EXISTS `ukuran` (
      `id_ukuran` int(5) NOT NULL AUTO_INCREMENT,
      `nama_ukuran` varchar(10) NOT NULL,
      PRIMARY KEY (`id_ukuran`)
)  ENGINE=InnoDB  DEFAULT  CHARSET=latin1  COMMENT='Tabel
Ukuran';
```

20. Tabel warna

```
    CREATE TABLE IF NOT EXISTS `warna` (
      `id_warna` int(5) NOT NULL AUTO_INCREMENT,
      `nama_warna` varchar(20) NOT NULL,
      PRIMARY KEY (`id_warna`)
    ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4.3 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk *file* program. Berikut ini adalah implementasi antarmuka untuk pengguna. Adapun untuk tampilan implementasi pembangunan sistem penjualan *online* di *website Distroblackshoes* akan dilampirkan pada halaman lampiran.

Tabel 4.3 Implementasi antar muka

Menu	Deskripsi	Nama File
Login admin	Digunakan sebagai halaman *login* admin	*login*.php
Member	Digunakan untuk melihat, menambah, dan merubah data *member*	*member*.php
Kota	Digunakan untuk melihat, menambah, dan merubah data kota	kota. php
Provinsi	Digunakan untuk melihat, menambah, dan merubah data provinsi	provinsi. php
Barang	Digunakan untuk melihat, menambah, dan merubah data barang	barang. php
Jasa pengiriman	Digunakan untuk melihat, menambah, dan merubah data pengiriman	jasapengiriman. php
Jenis pengiriman	Digunakan untuk melihat, menambah, dan merubah data pengiriman	jenispengiriman. php
Ongkos Kirim	Digunakan untuk melihat, menambah, dan merubah data Ongkos Kirim	ongkoskirim. php
Warna	Digunakan untuk melihat, menambah, dan merubah data warna barang	warna.php
Ukuran	Digunakan untuk melihat, menambah, dan merubah data ukuran barang	ukuran.php
Kategori	Digunakan untuk melihat, menambah, dan merubah data kategori barang	kategori.php
Retur	Digunakan untuk melihat data retur dan mengubah	retur.php

	status retur	
Laporan barang	Digunakan untuk melihat laporan stok barang	Laporanbarang.php
Laporan penjualan	Digunakan untuk melihat laporan laporan	laporanantarwaktu.php
Halaman Depan	Digunakan untuk menampilkan halaman utama	index.php
Registrasi *member*	Digunakan untuk melakukan pendaftaran *member*	daftar.php
Tentang kami	Digunakan untuk menampilkan halaman tentang kami	about.php
Hubungi kami	Digunakan untuk menampilkan halaman hubungi kami	hubungi.php
Barang	Digunakan untuk melihat seluruh data barang yang ada	barang.php
Detail pembelian	Digunakan untuk melihat seluruh data pemesanan yang ada secara *detail*.	*detail*.php
Keranjang belanja	Digunakan untuk melihat, merubah, menghapus *detail* transaksi yang akan dilakukan	*cart*.php

4.4 Pengujian Perangkat Lunak

Pengujian sistem merupakan hal terpenting yang bertujuan untuk menemukan kesalahan – kesalahan atau kekurangan – kekurangan pada perangkat lunak yang diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang

dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

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

4.5 Pengujian Alpha

Pengujian alpha dilakukan dengan menggunakan metode *black box*. Untuk pengujian alpha ini yaitu pada pengujian sebagai pengguna.

4.5.1 Skenario Pengujian Alpha

Tabel 4.4 menjelaskan skenario pengujian aplikasi *member*.

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Registrasi *member*	Isi *form* pendaftaran	*Black Box*
	Simpan data baru	*Black Box*
	Isi data *login*	*Black Box*
Login	Verifikasi *username*	*Black Box*
	Verifikasi *password*	*Black Box*
Ganti *profile*	Ubah data *member*	*Black Box*
	Simpan perubahan	*Black Box*
Ganti *password*	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verifikasi konfirmasi *password*	*Black Box*
Pengiriman Barang	Menampilkan data pengiriman	*Black Box*

Tabel 4.5 menjelaskan skenario pengujian aplikasi petugas.

Kelas Uji	**Butir Uji**	**Jenis Pengujian**
Login	Verifikasi *username*	*Black Box*
	Verivikasi *password*	*Black Box*
Ganti P*assword*	Verifikasi *password* lama	*Black Box*
	Verifikasi *password* baru	*Black Box*
	Verfikasi konfirmasi *password*	*Black Box*
Kota	Tambah data kota	*Black Box*
	Ubah data kota	*Black Box*
Provinsi	Tambah data Provinsi	*Black Box*
	Ubah data Provinsi	*Black Box*
Jasa Pengiriman	Tambah data Jasa pengiriman	*Black Box*
	Ubah data Jasa Pengiriman	*Black Box*
Jenis Pengiriman	Tambah data jenis pengiriman	*Black Box*
	Ubah data jenis pengiriman	*Black Box*
Warna	Ubah data Warna	*Black Box*
	Lihat data Warna	*Black Box*
Ukuran	Tambah data Ukuran	*Black Box*
	Ubah data Ukuran	*Black Box*
Kategori	Tambah data Kategori	*Black Box*
	Ubah data Kategori	*Black Box*
Barang	Tambah data barang	*Black box*
	Ubah data barang	*Black box*

4.5.2 Kasus Dan Hasil Pengujian

4.5.2.1 Pengujian Registrasi Member

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

Tabel 4.6 Pengujian Registrasi *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data pendaftaran, contoh masukan: Email: reydota@yahoo.co.id *Password*: reysa *Confirm password* : reysa Nama lengkap: Reysa Randy Alamat : Jl. Kubangsari 7 no 57 Telpon: 08132242424 Provinsi : Jawa Barat Kota : Bandung Kodepos : 40134 Kode konfirmasi: 12GhsL	Jika semua *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses daftar.	Form pendaftaran terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses daftar.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data pendaftaran, contoh masukan : Email: (kosong) *Password*: (kosong) *Confirm password* : (kosong) Nama lengkap: (kosong) Alamat : (kosong) Telpon: (kosong) Provinsi : (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi	pesan kesalahan dapat tampil dan muncul pesan kesalahan bahwa data harus diisi	[√]Diterima [ ] Ditolak

Kota : (kosong) Kodepos : (kosong) Kode konfirmasi: (kosong)			

4.5.2.2 Pengujian Login Member

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

Tabel 4.7 Pengujian *login member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data *login*, contoh masukan: Email: reydota@yahoo.co.id Kata sandi: 123456	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses *login*.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses *login*.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data *login*, contoh masukan: Email: (kosong) Kata sandi: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan maaf *account* tidak ditemukan	pesan kesalahan dapat tampil dan muncul pesan kesalahan maaf *account* tidak ditemukan	[√]Diterima [ ]Ditolak

4.5.2.3 Pengujian Ubah Profile *Member*

Berikut dapat dilihat pengujian dari Ubah *profile member* pada tabel 4.8 dibawah ini:

Tabel 4.8 Pengujian ubah *profile member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data *edit profile*, contoh masukan: Email: reydota@yahoo.co.id Nama: reysa Alamat: kubangsari Telp: 08132424242 Provinsi: Jawa Barat Kota: Bandung Kodepos: 45536	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses *edit profile*.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses *edit profile*.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data *edit profile*, contoh masukan: Email: (kosong) Nama: (kosong) Alamat: (kosong) Telp: (kosong) Provinsi: (kosong) Kota: (kosong) Kodepos: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi semua.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi semua.	[√]Diterima [ ]Ditolak

4.5.2.4 Pengujian Ganti *Password Member*

Berikut dapat dilihat pengujian dari ganti *password member* pada tabel 4.9 dibawah ini:

Tabel 4.9 Pengujian ganti *password member*

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data ganti *password*, contoh masukan: Kata sandi lama: rey Kata sandi baru: reysa Ulangi kata sandi: reysa *Code*: Gfr323	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ganti *password*.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ganti *password*.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data ganti *password*, contoh masukan: Kata sandi lama: (kosong) Kata sandi baru: (kosong) Ulangi kata sandi: (kosong) *Code*: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi semua.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi semua.	[√]Diterima [ ]Ditolak

4.5.2.5 Pengujian Menu Pengiriman

Tabel 4.10 Pengujian menu pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data pengiriman, contoh masukan: Nama lengkap: reysa Alamat: kubangsari Provinsi: Jawa Barat Kota: Bandung Kodepos: 52422 Email: reydota@yahoo.co.id Telpon: 0813422622	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses pengiriman.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses pengiriman.	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data pengiriman, contoh masukan: Nama lengkap: (kosong) Alamat: (kosong) Provinsi: (kosong) Kota: (kosong) Kodepos: (kosong) Email: (kosong) Telpon: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi semua.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi semua.	[√]Diterima [ ]Ditolak

4.5.2.6 Pengujian Login Admin

Berikut dapat dilihat pengujian *login* adminpada tabel 4.11

Tabel 4.11 Pengujian *login* admin

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data *login* admin, contoh masukan: *Username*: admin *Password*: reysa	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses *login* admin.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses *login* admin.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data *login* admin, contoh masukan: *Username*: (kosong) *Password*: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi.	[√]Diterima [ ]Ditolak

4.5.2.7 Pengujian Ubah *Password* Admin

Berikut dapat dilihat pengujian dari ganti *password* admin pada tabel 4.12

Tabel 4.12 Pengujian ganti *password* admin

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data ganti *password* admin, contoh masukan:	Jika *field* terisi atau benar dalam pengisian, maka	Jika terisi atau benar dalam pengisian, maka	[√]Diterima [ ]Ditolak

Password lama: rey *Password* baru: reysa *Confirm password*: reysa	akan dapat melanjutkan proses ganti *password* admin.	akan dapat melanjutkan proses ganti *password* admin.	
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data ganti *password* admin, contoh masukan: *Password* lama: (kosong) *Password* baru: (kosong) *Confirm password*: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi.	[√]Diterima [ ]Ditolak

4.5.2.8 Pengujian Ubah Data Provinsi

Berikut dapat dilihat pengujian ubah data provinsi pada tabel 4.13

Tabel 4.13 Pengujian ubah data provinsi

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data provinsi, contoh masukan: Provinsi: Jawa Barat	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah provinsi.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah provinsi.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data provinsi, contoh masukan: Provinsi: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi.	[√]Diterima [ ]Ditolak

4.5.2.9 Pengujian Data Ubah Kota

Berikut dapat dilihat pengujian data ubah kota pada tabel 4.14

Tabel 4.14 Pengujian data kota

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data kota, contoh masukan: Nama Kota: Bandung	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah kota.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah kota.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data kota, contoh masukan: Kota: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi.	[√]Diterima [ ]Ditolak

4.5.2.10 Pengujian Ubah Data Jasa Pengiriman

Berikut dapat dilihat pengujian ubah data Jasa pengiriman pada tabel 4.15

Tabel 4.15 Pengujian ubah data Jasa pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data jasa pengiriman, contoh masukan:	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan	Jika terisi atau benar dalam pengisian, maka akan dapat	[√]Diterima [ ]Ditolak

Jasa pengiriman: JNE	proses ubah jasa pengiriman.	melanjutkan proses ubah jasa pengiriman.	
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data jasa pengiriman, contoh masukan: Jasa pengiriman: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi.	[√]Diterima [ ]Ditolak

4.5.2.11 Pengujian Ubah Data Jenis Pengiriman

Berikut dapat dilihat pengujian dari ubah pada tabel 4.16

Tabel 4.16 Pengujian Ubah Data Jenis Pengiriman

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data jenis pengiriman, contoh masukan: Jenis pengiriman: YES	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah jenis pengiriman.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah jenis pengiriman.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data jenis pengirimna, contoh masukan: Jenis pengiriman:	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data	[√]Diterima [ ]Ditolak

(kosong)		harus diisi.	

4.5.2.12 Pengujian Ubah Data Ongkos Kirim

Berikut dapat dilihat pengujian databiaya pengiriman pada tabel 4.17

Tabel 4.17 Pengujian data Ongkos Kirim

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data ongkos kirim, contoh masukan: Provinsi: S Kota: Bandung Jasa pengiriman: JNE Jenis pengiriman: YES Harga Kirim: 250	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah ongkos kirim	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah ongkos kirim	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data ongkos kirim, contoh masukan: Provinsi: (kosong) Kota: (kosong) Jasa pengiriman: (kosong) Jenis pengiriman: (kosong) Harga Kirim: (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi.	

4.5.2.13 Pengujian Ubah Data Warna

Berikut dapat dilihat pengujian data warna pada tabel 4.18

Tabel 4.18 Pengujian data warna

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data jasa pengiriman, contoh masukan: Warna : merah	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah warna.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah jasa pengiriman.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data jasa pengiriman, contoh masukan: Warna : (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi.	[√]Diterima [ ]Ditolak

4.5.2.14 Pengujian Ubah Data Ukuran

Berikut dapat dilihat pengujian data ukuran pada tabel 4.19

Tabel 4.19 Pengujian data ukuran

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data ukuran, contoh masukan:	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan	[√]Diterima [ ]Ditolak

Ukuran : 38	proses ubah ukuran.	proses ubah ukuran.	
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data jasa pengiriman, contoh masukan: Ukuran : (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi.	[√]Diterima [ ]Ditolak

4.5.2.15 Pengujian Ubah Data Kategori

Berikut dapat dilihat pengujian datakategori pada tabel 4.20

Tabel 4.20 Pengujian data kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data ukuran, contoh masukan: Kategori : Vans	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah kategori.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah kategori.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data jasa pengiriman, contoh masukan: Kategori : (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi.	[√]Diterima [ ]Ditolak

4.5.2.16 Pengujian Ubah Data Barang

Berikut dapat dilihat pengujian data barang pada tabel 4.21

Tabel 4.21 Pengujian data barang

Kasus dan Hasil Uji Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data ukuran, contoh masukan: Kategori : Vans Nama Barang : Vans Blue Harga : 200000 Diskon : 10	Jika *field* terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah barang.	Jika terisi atau benar dalam pengisian, maka akan dapat melanjutkan proses ubah barang.	[√]Diterima [ ]Ditolak
Kasus dan Hasil Uji Benar (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Memasukan data jasa pengiriman, contoh masukan: Kategori : (kosong) Nama Barang : (kosong) Harga : (kosong) Diskon : (kosong)	Tampilkan pesan kesalahan dalam pengisian, maka akan muncul pesan kesalahan data harus diisi.	pesan kesalahan dapat tampil dan muncul pesan kesalahan data harus diisi.	[√]Diterima [ ]Ditolak

4.5.3 Kesimpulan Pengujian Alpha

Berdasarkan hasi pengujian alpha (fungsional) dengan kasus uji diatas dapat ditarik kesimpulan bahwa pembangunan sistem penjualan *online*di *website Distroblackshoes* bebas kesalahan penulisan dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

4.6 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan dengan membuat kuisioner mengenai kepuasan *user* dengan kandungan poin syarat *user friendy* untuk selanjutnya dibagikan kepada sebagian *user*. Dari hasil kuisioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru.

4.6.1.1 Wawancara Pengujian Beta Untuk Admin

1. Apakah *website* ini dapat membantu dalam pengolahan data *user* ?
2. Apakah anda setuju dengan adanya *website* ini, pemilik dapat memperluas pangsa pasarnya ?
3. Apakah anda mengerti dengan instruksi yang terdapat pada *website* tersebut ?
4. Apakah anda setuju dengan tampilan *website* admin untuk pengolahan data *user* ?

4.6.1.2 Hasil Pengujian Beta Untuk Admin

1. Apakah *website* ini dapat membantu dalam pengolahan data *user* ?

Jawaban : sangat membantu, karena datanya jadi lebih mudah dikelola

2. Apakah anda setuju dengan adanya *website* ini, toko Black shoes dapat memperluas pangsa pasarnya ?

Jawaban : setuju, karena dengan adanya *website* ini toko saya akan berkembang.

3. Apakah anda mengerti dengan instruksi yang terdapat pada *website* tersebut ?

Jawaban : setelah anda jelaskan, sekarang saya mengerti, ternyata pengelolaannya mudah saja.

4. Apakah anda setuju dengan tampilan *website* admin untuk pengolahan data *user* ?

Jawaban : setuju, tampilan *website* nya simpel dan mudah dipahami.

4.6.1.3 Wawancara Pengujian Beta Untuk Petugas Kasir

1. Apakah *website* ini dapat membantu anda dalam pengolahan data transaksi ?
2. Apakah *website* ini dapat membantu dalam pengolahan data penjualan secara *online* ?
3. Apakah *website* ini dapat membantu dalam pembuatan laporan penjualan?
4. Apakah anda mengerti dengan instruksi yang terdapat pada *website* tersebut?

5. Apakah anda setuju dengan tampilan *website* pada halaman petugas kasir?

4.6.1.4 Hasil Pengujian Beta Untuk Petugas Kasir

1. Apakah *website* ini dapat membantu anda dalam pengolahan data transaksi ?

Jawaban: sangat membantu, karena lebih mudah bagi saya untuk mengelola pembelian.

2. Apakah *website* ini dapat membantu dalam pengolahan data penjualan secara *online* ?

Jawaban: tentu saja, sekarang penjualan dapat dilakukan secara *online*

3. Apakah *website* ini dapat membantu dalam pembuatan laporan penjualan?

Jawaban : dengan adanya website ini saya tidak perlu capek menulis laporan, sehingga menghemat waktu dan meringankan beban kerja saya.

4. Apakah anda mengerti dengan instruksi yang terdapat pada *website* tersebut?

Jawaban: setelah tadi anda demokan kepada saya, sekarang saya mengerti dan ternyata tidak terlalu sulit.

5. Apakah anda setuju dengan tampilan *website* pada halaman petugas kasir?

Jawaban : ya, saya setuju

4.6.1.5 Wawancara Pengujian Beta Untuk Petugas Gudang

1. Apakah *website* ini dapat membantu anda dalam pengolahan data master?
2. Apakah *website* ini dapat membantu anda dalam pembuatan laporan barang ?
3. Apakah anda mengerti dengan instruksi yang terdapat pada *website* tersebut ?
4. Apakah anda setuju dengan tampilan *website* pada halaman petugas gudang?

4.6.1.6 Hasil Wawancara Pengujian Beta Untuk Petugas Gudang

1. Apakah *website* ini dapat membantu anda dalam pengolahan data master?

Jawaban: tentu saja membantu, dengan begini saya dapat mengelola data barang dengan mudah.

2. Apakah *website* ini dapat membantu anda dalam pembuatan laporan barang ?

Jawaban: setelah anda jelaskan, menurut saya *website* ini cukup membantu saya dalam pembuatan laporan barang.

3. Apakah anda mengerti dengan instruksi yang terdapat pada *website* tersebut ?

Jawaban: ya saya mengerti sekarang setelah anda jelaskan kepada saya.

4. Apakah anda setuju dengan tampilan *website* pada halaman petugas gudang?

Jawaban: setuju saja, tampilannya mudah dipahami sih.

4.6.1.7 Kuisioner Pengujian Beta Untuk Member

1. Apakah anda setuju bahwa *website e-commerce* ini dapat membantu anda untuk mendapatkan informasi terbaru tentang barang yang ada di Toko Black shoes ?

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

2. Jika rumah anda jauh, apakah anda setuju bahwa *website* ini akan lebih mempermudah anda dalam melakukan transaksi pembelian dibandingkan transaksi secara manual ?

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

3. Jika rumah anda jauh, apakah anda setuju bahwa sistem pembayaran yang digunakan akan lebih memudahkan anda ?

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

4. Apakah anda setuju dengan *website e-commerce* ini pemilik toko Black shoes akan dapat memperluas pangsa pasarnya ?

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

5. Apakah anda setuju bahwa tampilan *website* Black shoes cukup menarik dan mudah dipahami / *user friendly* ?

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

Berdasarkan data hasil kuisioner tersebut, dapat dicari presentase masing-masing jawaban dengan menggunakan rumus : Y = (P/Q) * 100%

Keterangan :

P = Banyaknya jawaban responden tiap soal

Q = Jumlah responden

Y = Nilai persentase

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumus diatas. Kuisioner ini diujikan kepada 15 orang.

1. Apakah anda setuju bahwa *website e-commerce* ini dapat membantu anda untuk mendapatkan informasi terbaru tentang barang yang ada di Toko Black shoes ?

Tabel 4.22 Hasil pertanyaan kuisioner pertanyaan nomor 1

Pertanyaan	No	Keterangan	Responden	Presentase %
1	1	Sangat Setuju	6	30
	2	Setuju	8	40
	3	Cukup Setuju	6	30
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
	Jumlah		**20**	**100**

2. Jika rumah anda jauh, apakah anda setuju bahwa *website* ini akan lebih mempermudah anda dalam melakukan transaksi pembelian dibandingkan transaksi secara manual?

Tabel 4.23 Hasil pertanyaan kuisioner pertanyaan nomor 2

Pertanyaan	No	Keterangan	Responden	Presentase %
2	1	Sangat Setuju	5	25
	2	Setuju	7	35
	3	Cukup Setuju	8	40
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
	Jumlah		**20**	**100**

3. Jika rumah anda jauh, apakah anda setuju bahwa sistem pembayaran yang digunakan akan lebih memudahkan anda ?

Tabel 4.24 Hasil pertanyaan kuisioner pertanyaan nomor 3

Pertanyaan	No	Keterangan	Responden	Presentase %
3	1	Sangat Setuju	5	25
	2	Setuju	8	40
	3	Cukup Setuju	6	30
	4	Kurang Setuju	1	5
	5	Tidak Setuju	0	0
	Jumlah		**20**	**100**

4. Apakah anda setuju dengan *website e-commerce* ini pemilik toko Black shoes akan dapat memperluas pangsa pasarnya ?

Tabel 4.25 Hasil pertanyaan kuisioner pertanyaan nomor 4

Pertanyaan	No	Keterangan	Responden	Presentase %
4	1	Sangat Setuju	8	40
	2	Setuju	6	30
	3	Cukup Setuju	6	30
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
	Jumlah		**20**	**100**

5. Apakah anda setuju bahwa tampilan *website* Black shoes cukup menarik dan mudah dipahami / *user friendly* ?

Tabel 4.26 Hasil pertanyaan kuisioner pertanyaan nomor 5

Pertanyaan	No	Keterangan	Responden	Presentase %
5	1	Sangat Setuju	7	35
	2	Setuju	8	40
	3	Cukup Setuju	5	5
	4	Kurang Setuju	0	0
	5	Tidak Setuju	0	0
	Jumlah		**20**	**100**

4.6.2 Kesimpulan Pengujian Beta

Berdasarkan pengujian beta diatas, bahwa dapat diambil kesimpulan bahwa:

1. Aplikasi ini memudahkan pembeli dalam mendapatkan informasi barang yang tersedia.
2. Pembeli dapat lebih mudah dalam melakukan transaksi pembelian barang di toko Black shoes.
3. Aplikasi ini juga memudahkan pihak toko dalam mengelola data master dan membuat laporan barang dan laporan penjualan.