BAB IV
IMPLEMENTASI DAN PENGUJIAN

4. 1. Implementasi

Pada bab ini akan dilakukan implementasi dan pengujian terhadap web yang telah dirancang. Tujuan tahapan implementasi adalah untuk menerapkan hasil perancangan agar maksud dan tujuan pembuatan aplikasi dapat tercapai. Setelah implementasi dilakukan, maka tahap berikutnya adalah pengujian terhadap pembuatan aplikasi, untuk mengetahui apakah maksud dan tujuan yang ingin dicapai telah terpenuhi sehingga dapat ditarik kesimpulan.

4.1.1 Implementasi Perangkat Keras

Spesifikasi perangkat keras minimal yang dibutuhkan untuk mengimplementasikan perangkat keras antara lain terdapat pada tabel 4.1 sebagai berikut:

Tabel 4.1 Perangkat Keras yang Dibutuhkan

No	Perangkat Keras	Spesifikasi
1	Processor	Processor AMD Phenom II 3,4 GHz
2	Monitor	Monitor LCD 17 inch
4	Memori	2048 MB
6	Keyboard	Logitech
7	Mouse	Logitech
8	Printer	Hitam putih dan warna

4.1.2 Implementasi Perangkat Lunak

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

Tabel 4.2 Perangkat Lunak yang digunakan

No	Perangkat Keras	Spesifikasi
1	Sistem Operasi	Windows 7 Ultimate 32-bit
2	Web Browser	Mozilla Firefox, Internet Explorer, Google Chrome
3	Code Editor	Macromedia Dreamweaver 8
4	Database	Mysql versi 5.0
5	Web server	XAMPP 2.5.8

4.1.3 Implementasi Basis Data

Pembuatan basis data dilakukan dengan menggunakan aplikasi pemrograman *MySQL*. Implementasi basis data yang ada yaitu :

1. Tabel Barang

Tabel barang digunakan untuk menyimpan data barang

```
CREATE TABLE IF NOT EXISTS `barang` (
  `barang_id` int(11) NOT NULL AUTO_INCREMENT,
  `kategori` int(11) NOT NULL,
  `sub_kategori` int(11) NOT NULL,
  `kode_barang` varchar(10) NOT NULL,
  `nama_barang` varchar(50) NOT NULL,
  `berat` float NOT NULL,
  `harga` double NOT NULL,
  `harga_reseller` double NOT NULL,
  `stok` int(11) NOT NULL,
  `deskripsi_barang` text NOT NULL,
  `rekomendasi` varchar(10) NOT NULL,
  `status_aktif` varchar(15) NOT NULL,
  `status_barang` varchar(20) NOT NULL,
  `gambar_1` varchar(100) NOT NULL,
  `gambar_2` varchar(100) NOT NULL,
  `gambar_3` varchar(100) NOT NULL,
  `gambar_4` varchar(100) NOT NULL,
  `input_by` varchar(30) NOT NULL,
  `input_date` datetime NOT NULL,
  `update_by` varchar(30) NOT NULL,
  `update_date` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`barang_id`)
```

2. Tabel *Member*

Tabel member digunakan untuk menyimpan data pelanggan.

```
CREATE TABLE IF NOT EXISTS `member` (
  `customer_id` int(11) NOT NULL AUTO_INCREMENT,
  `nama_lengkap` varchar(30) NOT NULL,
  `tempat_lahir` varchar(30) NOT NULL,
  `tanggal_lahir` date NOT NULL,
  `jenis_kelamin` varchar(15) NOT NULL,
  `alamat` text NOT NULL,
  `provinsi` varchar(30) NOT NULL,
  `kabupaten_kota` varchar(30) NOT NULL,
  `kodepos` varchar(6) NOT NULL,
  `telepon` varchar(15) NOT NULL,
  `handphone` varchar(15) NOT NULL,
  `paypall` varchar(50) DEFAULT NULL,
  `email` varchar(150) DEFAULT NULL,
  `nama_bank` varchar(30) DEFAULT NULL,
  `nama_rekening` varchar(30) DEFAULT NULL,
  `no_rekening` varchar(30) DEFAULT NULL,
  `user_name` varchar(50) NOT NULL,
  `password` varchar(100) NOT NULL,
  `foto` varchar(50) NOT NULL,
  `status_customer` tinyint(4) NOT NULL,
  `status_confirm` int(11) NOT NULL,
  PRIMARY KEY (`customer_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=37 ;
```

3. Tabel diskon

Tabel diskon digunakan untuk mengatur diskon dari suatu barang.

```
CREATE TABLE IF NOT EXISTS `diskon` (
  `diskon_id` int(11) NOT NULL AUTO_INCREMENT,
  `judul` varchar(35) NOT NULL,
  `tanggal_awal` date NOT NULL,
  `tanggal_akhir` date NOT NULL,
  `minimal_beli` double NOT NULL,
  `besar_diskon` float NOT NULL,
  `status_diskon` varchar(15) NOT NULL,
  `deskripsi` text NOT NULL,
  PRIMARY KEY (`diskon_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

4. Tabel Kategori

Tabel kategori digunakan untuk menyimpan data kategori ustu barang.

```
CREATE TABLE IF NOT EXISTS `kategori` (
  `kategori_id` int(11) NOT NULL AUTO_INCREMENT,
  `nama_kategori` varchar(25) NOT NULL,
  `deskripsi_kategori` text NOT NULL,
  PRIMARY KEY (`kategori_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;
```

5. Tabel Master Kabupaten

Tabel ongkos kirim digunakan untuk menyimpan data kabupaten

```
CREATE TABLE IF NOT EXISTS `master_kabupaten` (
  `id_kabupaten` int(3) NOT NULL,
  `nama_kabupaten` varchar(50) NOT NULL,
  `id_propinsi` int(2) NOT NULL,
  PRIMARY KEY (`id_kabupaten`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

6. Tabel Master Propinsi

Tabel paket digunakan untuk menyimpan data propinsi.

```
CREATE TABLE IF NOT EXISTS `master_propinsi` (
  `id_propinsi` int(2) NOT NULL,
  `nama_propinsi` varchar(50) NOT NULL,
  PRIMARY KEY (`id_propinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

7. Ongkos Kirim

Tabel paket detail digunakan untuk menyimpan data ongkos kirim

```
CREATE TABLE IF NOT EXISTS `ongkos_kirim` (
  `ongkir_id` int(11) NOT NULL AUTO_INCREMENT,
  `paket` int(11) NOT NULL,
  `paket_detail` int(11) NOT NULL,
  `kota_asal` varchar(35) NOT NULL,
  `kota_tujuan` varchar(35) NOT NULL,
  `tarif` double NOT NULL,
  PRIMARY KEY (`ongkir_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=92 ;
```

10. Tabel Paket

Tabel pembelian digunakan untuk menyimpan data paket

```
CREATE TABLE IF NOT EXISTS `paket` (
  `paket_id` int(11) NOT NULL AUTO_INCREMENT,
  `nama_paket` varchar(35) NOT NULL,
  `deskripsi_paket` text NOT NULL,
  PRIMARY KEY (`paket_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=9 ;
```

11. Tabel Pembelian

Tabel pembelian digunakan untuk menyimpan data pembelian.

```
CREATE TABLE IF NOT EXISTS `pembelian` (
  `pembelian_id` int(11) NOT NULL AUTO_INCREMENT,
  `tanggal_pembelian` date NOT NULL,
  `supplier` int(11) NOT NULL,
  `nama_supplier` varchar(30) NOT NULL,
  `total_pembelian` double NOT NULL,
  `status_pembelian` int(11) NOT NULL,
  PRIMARY KEY (`pembelian_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

12. Tabel Pembelian Detail

Tabel pengguna digunakan untuk menyimpan data pembelian yang lebih detail.

```
CREATE TABLE IF NOT EXISTS `pembelian_detail` (
  `pembelian_detail_id` int(11) NOT NULL AUTO_INCREMENT,
  `pembelian` int(11) NOT NULL,
  `barang` int(11) NOT NULL,
  `nama_barang` varchar(40) NOT NULL,
  `harga` double NOT NULL,
  `jumlah` int(11) NOT NULL,
  `subtotal` double NOT NULL,
  PRIMARY KEY (`pembelian_detail_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=12 ;
```

13. Tabel Admin

Tabel profil digunakan untuk menyimpan data admin

```
CREATE TABLE IF NOT EXISTS `pengguna` (
  `user_id` int(11) NOT NULL AUTO_INCREMENT,
  `akses` int(11) NOT NULL,
  `nama` varchar(35) NOT NULL,
  `username` varchar(30) NOT NULL,
  `password` varchar(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `kata_kunci` varchar(40) NOT NULL,
  PRIMARY KEY (`user_id`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

14. Tabel Profil

Tabel sub katagori digunakan untuk menyimpan data profil

```
CREATE TABLE IF NOT EXISTS `profil` (
  `nama_perusahaan` varchar(30) NOT NULL,
  `alamat_perusahaan` text NOT NULL,
  `provinsi_perusahaan` varchar(20) NOT NULL,
  `kota_perusahaan` varchar(20) NOT NULL,
  `telepon_perusahaan` varchar(15) NOT NULL,
  `email_perusahaan` varchar(30) NOT NULL,
  `keterangan` text NOT NULL,
  `logo` varchar(50) NOT NULL,
  `bank` varchar(50) NOT NULL,
  `atas_nama` varchar(50) NOT NULL,
  `no_rekening` varchar(30) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

15. Tabel Retur

Tabel transaksi digunakan untuk menyimpan data transaksi.

```
CREATE TABLE IF NOT EXISTS `retur` (
  `id_retur` int(11) NOT NULL AUTO_INCREMENT,
  `customer_id` varchar(150) NOT NULL,
  `transaksi_id` varchar(150) NOT NULL,
  `nama_barang` varchar(150) NOT NULL,
  `keterangan_retur` varchar(150) NOT NULL,
  `tanggal_retur` datetime NOT NULL,
  `tanggal_kirim_balik` datetime NOT NULL,
  `status` varchar(10) NOT NULL,
  PRIMARY KEY (`id_retur`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

16. Tabel Transaksi

Tabel transaksi digunakan untuk menyimpan data transaksi.

```
CREATE TABLE IF NOT EXISTS `transaksi` (
`transaksi_id` int(11) NOT NULL AUTO_INCREMENT,
`tanggal_transaksi` datetime NOT NULL,
`customer` int(11) NOT NULL,
`nama_customer` varchar(30) NOT NULL,
`no_rekening` varchar(20) NOT NULL,
`atas_nama` varchar(50) NOT NULL,
`ongkos_kirim` float NOT NULL,
`diskon` float NOT NULL,
`total_transaksi` double NOT NULL,
`total_transaksi_dolar` double NOT NULL,
`jenis_pengiriman` varchar(10) DEFAULT NULL,
`jenis_paket` varchar(15) DEFAULT NULL,
`kota_tujuan` varchar(20) NOT NULL,
`alamat_tujuan` varchar(150) NOT NULL,
`jenis_pembayaran` varchar(20) DEFAULT NULL,
`tanggal_pembayaran` varchar(20) NOT NULL,
`status` int(11) NOT NULL,
```

17. Tabel Trate

Tabel transaksi digunakan untuk menyimpan data rate

```
CREATE TABLE IF NOT EXISTS `trate` (
  `customer_id` varchar(50) DEFAULT NULL,
  `barang_id` varchar(50) DEFAULT NULL,
  `rate` varchar(50) DEFAULT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

4.1.4 Implementasi *Class*

Pembangunan aplikasi ini didasarkan pada perancangan yang telah dibahas pada bab 3. Dari perancangan tersebut, terdapat *class-class* yang telah terbentuk dari *use case*. *Class-class* tersebut juga diimplementasikan dalam bentuk file-file *.php yang nantinya file-file *.php yang berisi *class-class* tersebut digunakan dalam pembuatan dan jalannya proses dalam aplikasi. Berikut implementasi *class* yang telah terdefinisi dan digunakan dalam aplikasi dapat dilihat pada tabel 4.3.

Tabel 4.3 Implementasi *Class*

No	Nama *Class*	Nama File Fisik
1	admin_control	admin_control.php
2	barang_control	barang_control.php
3	diskon_control	diskon_control.php
4	kategori_control	kategori_control.php
5	pemesanan_control	pemesanan_control.php
6	profil_control	profil_control.php
7	supplier_control	supplier_control.php
8	users_control	users_control.php
9	barang_model	barang_model.php
10	diskon_model	diskon_model.php
11	kategori_model	kategori_model.php
12	pemesanan_model	pemesanan_model.php
13	profil_model	profil_model.php
14	supplier_model	supplier_model.php

15	user_model	user_model.php
16	users_model	users_model.php
17	admin_view	admin_view.php
18	barang_view	barang_view.php
19	member_view	member_view.php
20	diskon_view	diskon_view.php
21	kategori_view	kategori_view.php
22	profil_view	profil_view.php
23	supplier_view	supplier_view.php
24	users_view	users_view.php
25	pemesanan_view	pemesanan_view.php

4.1.5 Implementasi Antarmuka

Implementasi antar muka dilakukan dengan setiap tampilan program yang dibangun dan pengkodeannya dalam bentuk file program. Berikut ini adalah implementasi antarmuka untuk pengguna dapat dilihat pada tabel 4.4.

Tabel 4.4 Implementasi Antarmuka Admin

Menu	Deskripsi	Nama File Fisik
Login Admin	Digunakan sebagai halam login admin	admin_control.php
Halaman Utama Admin	Halalman utama admin	application\views\admin\index.php
Kategori	Digunakan untuk mengolah data kategori	application\views\admin\kategori\daftar_kategori.php
Barang	Digunakan untuk mengolah data barang	application\views\admin\barang\daftar_barang,php
Member	Digunakan untuk megolah data memeber	application\views\admin\member\daftar_member.php

Reseller	Digunakan untuk mengolah data reseller	application\views\admin\member\daftar_reseller.php
Profil Perusahaan	Digunakan untuk mengatur profil perusahaan	application\views\admin\profil\profil.php
Diskon	Digunakan untuk mengatur diskon	application\views\admin\diskon\daftar_diskon.php
Bantuan	Digunakan untuk mengatur bantuan	application\views\admin\bantuan\bantuan.php
Paket Ongkir	Digunakan untuk mengatur paket ongkir	application\views\admin\paket\daftar_paket.php
Daftar Harga Ongkir	Digunakan untuk mengatur daftar harga ongkir	application\views\admin\ongkos_kirim\daftar_ongkos_kirim.php
Pemesanan	Digunakan untuk mengolah data pemesanan	application\views\admin\penjualan\daftar_pemesanan.php
Penjualan	Digunakan untuk mengolah data penjualan	application\views\admin\penjualan\daftar_penjualan.php
retur	Digunakan untuk mengolah data retur	application\views\admin\retur\daftar_retur.php
Laporan	Digunakan untuk mengolah data laporan	application\views\admin\laporan\
Back up dan Restore	Digunakan untuk back up dan restore database	application\views\admin\backup

Implementasi antarmuka pengunjung dan member pada aplikasi *ecommerce* pada Mitra Tekni kdapat dilihat pada tabel 4.5.

Tabel 4.5 Implementasi Antarmuka Pengunjung

Menu	Deskripsi	Nama File Fisik
Beranda	Digunakan sebagai halam login admin	application\views\users\index.php
Cara	Berisi informasi cara pembelian	application\views\users\pemes

Pemberlian		anan\cara_pemesanan.php
Barang	Halaman untuk melihat barang	daftar_barang,php
Katagori	Halaman untuk melihat barang katagori	daftar_kategori.php
tracking	Digunakan untuk tracking	Tracking.php
Profil perusahaan	Halaman untuk melihat profil perusahaan	profil.php
bantuan	Digunakan untuk bantuan	Bantuan.php
Daftar member	Digunakan untuk pendaftaran member	daftar_member.php
Login member	Digunakan untuk login member	Login.php
Reset password	Digunakan untuk merest password	Resetpassword.php
keranjang	Digunakan untuk melihat daftar belanja	shopping_cart.php
konfirmasi	Digunakan untuk melakukan konfirmasi	Konfirmasipembayaran.php
Pengaturan akun	Digunakan untuk mengatur akun member	application\views\users\user\edit.php

4.2 Pengujian Sistem

Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut. Pengujian yang digunakan untuk menguji sistem yang baru adalah metode pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

4.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan dengan menguji sistem secara *alpha* dan *beta*. Pengujian pembangunan *e-commerce* pada Mitra

Teknikmenggunakan data uji berdasarkan data dari toko. Rencana pengujian selengkapnya terlihat pada tabel 4.6.

Tabel 4.6 Rencana Pengujian Aplikasi

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi Data Login	Blackbox
	Validasi username dan password	Blackbox
Barang	Tambah Barang	Blackbox
	Hapus barang	Blackbox
	Edit barang	Blackbox
	Cari barang	Blackbox
Katagori	Tambah katagori	Blackbox
	Hapus katagori	Blackbox
	Edit katagori	Blackbox
member	Lihat member	Blackbox
	Hapus member	Blackbox
	Cari member	Blackbox
pengaturan	profil	Blackbox
	diskon	Blackbox
	bantuan	Blackbox
Paket ongkir	Daftar harga ongkir	Blackbox
	Paket ongkir	Blackbox
penjualan	Pemesanan member	Blackbox
	Pemesanan reseller	Blackbox
	Penjualan member	Blackbox
	Penjualan reseller	Blackbox
retur	mengolah data retur	Blackbox
laporan	penjualan	Blackbox
	pembelian	Blackbox
	Stok barang	Blackbox

Konfirmasi pembayaran	Konfirmasi bayar	Blackbox
Back up dan restore	backup	Blackbox
	restore	Blackbox
logout	logout	Blackbox
Pengaturan akun	Pengaturan akun	Blackbox

4.2.2 Pengujian Alpha

Pengujian alpha merupakan pengujian yang menitikberatkan pada hasil *output* dari kendali *input* yang dimasukkan pada tampilan kendali *input form* tampilan, dimana pengujian dikatakan berhasil apabila *output* sesuai dengan kendali *input* yang dimasukkan pada setiap tampilan.

4.2.2.1 Pengujian Aplikasi

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

1. Pengujian *login*

Pengujian *login* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian *login* dilihat pada tabel 4.7

Tabel 4.7 Pengujian login

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username : admin *Password* : admin	Tercantum pada *text field username* dan *password*	Dapat mengisi data *Login* sesuai yang diharapkan.”	[√] Diterima [ ] Ditolak
- Nama : abc	Menampilkan pesan “Nama minimal harus 4 karakter”	Data tidak berhasil disimpan dan menampilkan pesan “Nama minimal harus 4 karakter”	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Username : admin *password* : 123456	Tidak dapat ogin/masuk dan enampilkan pesan kesalahan	Muncul pesan "*Username* atau *password* salah"	[√] Diterima [ ] Ditolak

2. Pengujian *logout*

Pengujian *logout* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian *logout* dilihat pada tabel 4.8

Tabel 4.8 Pengujian logout

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Admin yang sudah *login* memilih menu *logout*	Sistem akan menampilkan menu login admin	Sistem akan menampilkan menu login admin	[√] Diterima [ ] Ditolak

3. Pengujian Tambah Kategori

Pengujian tambah kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah kategori dilihat pada tabel 4.9

Tabel 4.9 Pengujian Tambah Kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tambah data kategori	Menampilkan *form* tambah kategori	Sistem menampilkan f*orm* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data kategori	Tercantun dalam	Data kategori	[√] Diterima

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
	masing-masing *field* kemudian data kategori tersimpan dalam *database*.	berhasil disimpan dalam *database*	[ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tambah kategori Data Masukan Yang diharapkan Pengamatan Kesimpulan	Mengisi data kategori Kategori tidak dapat disimpan dalam *database* karena ada data yang tidak diisi	Sistem tidak dapat menyimpan data kategori dalam ke dalam *database* dan muncul pesan kesalahan salah”	[√] Diterima [ ] Ditolak

4. Pengujian Edit Kategori

Pengujian edit kategori dilakukan dengan mengubah data kategori untuk mendapatkan hasil yang diinginkan. Pengujian tambah kategori dilihat ada tabel 4.10

Tabel 4.10 Pengujian Edit kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Edit kategori	Menampilkan form ubah kategori beserta data yang akan diubah	Dapat menampilkan form ubah kategori beserta data yang akan	[√] diterima [ ] ditolak

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
		diubah	
Data kategori lama	Tercantun dalam *field* kategori berupa data kategori lama	Dapat menampilkan field data kategori yang akan diubah.	[√] Diterima [ ] Ditolak
Data kategori yang akan diubah	Dapat mengisi *field* Data yang akan diubah dengan benar dan data kategori berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data kategori berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak

5. Pengujian Hapus Kategori

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

Tabel 4.11 Pengujian hapus kategori

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Ceklist data kategori yang akan dihapus dan menekan tombol hapus	Terdapat pesan untuk melakukan hapus kategori	Terdapat pesan untuk melakukan hapus kategori	[√] Diterima [ ] Ditolak

6. Pengujian Hapus Member

Pengujian hapus *member* dilakukan dengan menghapus data member untuk mendapatkan hasil yang diinginkan. Pengujian hapus *member* dilihat pada table 4.14

Tabel 4.12 Pengujian Hapus Member

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Ceklist data member yang akan dihapus dan menekan tombol hapus	Terdapat pesan untuk melakukan hapus member	Terdapat pesan untuk melakukan hapus member	[√] Diterima [ ] Ditolak

7. Pengujian Cari Member

Pengujian cari *member* dilakukan dengan memasukkan data member yang akan dicari untuk mendapatkan hasil yang diinginkan. Pengujian cari *member* dilihat pada table 4.15

Tabel 4.13 Pengujian Cari Member

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama member yang akan dicari dan menekan tombol filter	Menampilkan data member yang dicari	Menampilkan data member yang dicari	[√] Diterima [ ] Ditolak

8. Pengujian Tambah Barang

Pengujian tambah barang dilakukan dengan memasukan data baru untuk mendapatkan hasil yang diinginkan. Pengujian tambah barang dilihat pada tabel 4.16

Tabel 4.14 Pengujian Tambah Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tambah data barang	Menampilkan *form* tambah barang	Sistem menampilkan f*orm* tambah barang	[√] Diterima [ ] Ditolak
Mengisi data barang	Tercantun dalam masing-masing *field* kemudian data barang tersimpan dalam *database*.	Data barang berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tambah barang Data Masukan Yang diharapkan Pengamatan Kesimpulan	Mengisi data barang Barang tidak dapat disimpan dalam *database* karena ada data yang tidak diisi	Sistem tidak dapat menyimpan data barang dalam ke dalam *database* dan muncul pesan kesalahan salah”	[√] Diterima [ ] Ditolak

9. Pengujian Hapus Barang

Pengujian hapus barang dilakukan dengan menghapus data barang untuk mendapatkan hasil yang diinginkan. Pengujian hapus barang dilihat pada table 4.17

Tabel 4.15 Pengujian Hapus Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Ceklist data barang yang akan dihapus dan menekan tombol hapus	Terdapat pesan untuk melakukan hapus barang	Terdapat pesan untuk melakukan hapus barang	[√] Diterima [ ] Ditolak

10. Pengujian Edit Barang

Pengujian edit barang dilakukan dengan memasukan data baru di data barang lama untuk mendapatkan hasil yang diinginkan. Pengujian edit barang dilihat pada table 4.18

Tabel 4.16 Pengujian Edit Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Edit barang	Menampilkan form ubah barang beserta data yang akan diubah	Dapat menampilkan form ubah barang beserta data yang akan diubah	[√] diterima [ ] ditolak
Data barang lama	Tercantun dalam *field* barang berupa data barang lama	Dapat menampilkan field data barang yang akan diubah.	[√] Diterima [ ] Ditolak
Data barang yang akan diubah	Dapat mengisi *field* Data yang akan diubah dengan benar dan data barang berhasil diubah dan	Dapat mengisi *field* data yang akan diubah dengan benar dan data barang	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
	disimpan dalam *database*.	berhasil diubah dan disimpan dalam *databsase*	

11. Pengujian Cari Barang

Pengujian cari barang dilakukan dengan memasukkan data barang yang akan dicari untuk mendapatkan hasil yang diinginkan. Pengujian cari barang dilihat pada table 4.19

Tabel 4.17 Pengujian Cari Barang

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama barang yang akan dicari dan menekan tombol filter	Menampilkan data barang yang dicari	Menampilkan data barang yang dicari	[√] Diterima [ ] Ditolak

12. Pengujian Pembuatan laporan

Pengujian pembuatan laporan dilakukan dengan memasukkan batas waktu laporan barang yang akan ditampilkan untuk mendapatkan hasil yang diinginkan. Pengujian pembuatan laporan dilihat pada table 4.20

Tabel 4.18 Pengujian Pembuatan Laporan

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Memilih tanggal awal dan tanggal akhir untuk batasan laporan	Menampilkan data tanggal sesuai pilihan	Sistem mampu Menampilkan data tanggal sesuai pilihan	[√] Diterima [ ] Ditolak
Menekan filter untuk	Menampilkan data	Sistem mampu	[√] Diterima

Kasus dan Hasil Uji (Data Benar)			
Data Masukkan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
menampilkan laporan sesuai tanggal yang sudah dipilih	laporan sesuai tanggal yang sudah dipilih	Menampilkan data laporan sesuai tanggal yang sudah dipilih	[ ] Ditolak

4.2.3 Kesimpulan Hasil Pengujian Alpha

Berdasarkan hasil pengujian dengan kasus *sample* uji yang telah dilakukan memberikan kesimpulan bahwa secara fungsional sistem sudah dapat menghasilkan *output* yang diharapkan.

4.2.4 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan yaitu instansi yang bersangkutan mengenai kepuasan user dengan kandungan poin yaitu pemenuhan kebutuhan dari tujuan awal pembangunan *trouble ticket system* dan tampilan antarmuka dari *trouble ticket system*. Pengujian beta dilakukan melalui dua teknik pengambilan data, yaitu melalui wawancara dan kuesioner.

Pengujian beta melalui wawancara dilakukan untuk halaman *helpdesk*, *teknisi* dan *manager* pada aplikasi *trouble ticket system*, karena terbatasnya jumlah pengguna untuk *helpdesk*, *teknisi* dan *manager*.

Pengujian beta melalui kuisioner dilakukan dengan membuat kuesioner mengenai kepuasan *member* dengan kandungan poin syarat user friendly untuk selanjutnya dibagikan kepada sebagian *member* dengan mengambil sample sebanyak 10 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan terhadap penilaian

penerapan sistem yang baru. Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung di tempat penelitian.

4.2.5 Pengujian Beta Untuk Perusahaan (Situs *Backend*)

Pengujian Beta untuk perusahaan dilakukan dengan cara wawancara kepada setiap pegawai yang akan memakai sistem, berikut adalah hasil dari wawancara kepada setiap pegawai dan pemilik dari perusahaan :

4.2.6 Pengujian Beta untuk Admin/Pemilik

1. Apakah anda setuju *website* ini dapat membantu dalam memperluas promosi dan penjualan barang ?

   Jawaban : setuju, karena dengan adanya *website* ini barang dapat lebih dikenal secara luas oleh masyarakat serta membantu dalam penjualan barang.

2. Apakah anda setuju dengan adanya *website* ini anda dapat dengan mudah mendapatkan informasi menganai laporan?

   Jawaban :setuju, karena dengan adanya *websit e* ini saya dapat mendapatkan informasi hasil penjualan dengan mudah.

3. Apakah *website* bagian admin/pemilik ini mudah untuk digunakan ?

   Jawaban :pada awalnya cukup kesulitan tetapi setelah beberapa lama menggunakan *website* bagian admin/pemilik ini ternyata cukup mudah digunakan.

4.2.7 Kuesioner Pengujian Beta Untuk Pengunjung (Situs *Frontend*)

Kuesioner disebarkan menggunakan teknik *Simple Random Sampling*dimana anggota *sample* dari populasi dilakukan secara acak tanpa memperhatikan strata yang ada dalam populasi. Kuisioner disebarkan kepada 10 orang masyarakat umum, dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat diambil kesimpulan terhadap penilaian penerapan sistem yang baru. Kuesioner terdiri dari 5 pernyataan dengan menggunakan skala 1 sampai 5 (skala likert).

Untuk mencari nilai persentase dari masing-masing jawaban kuesioner digunakan rumus skala likert sebagai berikut :

$$Y = \frac{S}{skorideal} \times 100$$

Keterangan :

Y = Nilai persentase yang dicari

S = Jumlah responden dikalikan dengan nilai yang di tetapkan jawaban

Skorideal= Nilai tertinggi dikalikan dengan jumlah sampel / responden

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

1. Apakah anda setuju , tampilan antarmuka/*interface* website ini menarik?

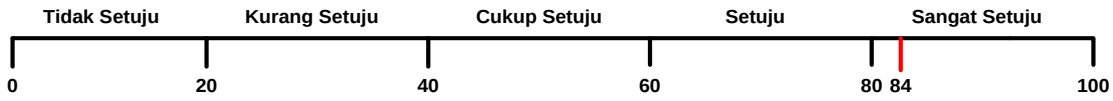
Tabel 4. 19 Hasil Pengujian Kuesioner soal nomor 1 (Situs *Frontend*)

Pertanyaan	Nilai	Jawaban	Responden	Nilai x Responden
1	5	Sangat Setuju	3	15
	4	Setuju	6	24
	3	Cukup Setuju	1	3
	2	Kurang Setuju	0	0
	1	Tidak Setuju	0	0
Jumlah			10	42

Berdasarkan data hasil kusioner tersebut, dapat dicari prosentase jawaban dengan menggunakan rumus :

$$Y = \frac{42}{10\ x\ 5} \times 100 = 84\ \%$$

Skala



Berdasarkan hasil persentase diatas maka dapat disimpulkan 84% menyatakan sangat setuju bahwa *website* yang dibangun tampilan antarmuka/*interface* website ini menarik.

2. Apakah Anda setuju penyajian informasi barang di website sudah sesuai dengan informasi yang dibutuhkan ?

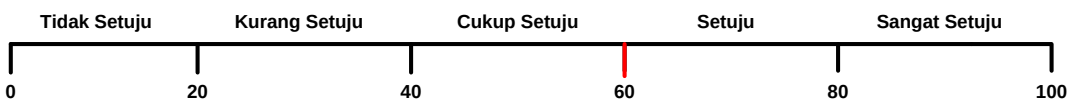
Tabel 4. 20 Hasil Pengujian Kuesioner soal nomor 2 (Situs *Frontend*)

Pertanyaan	Nilai	Jawaban	Responden	Nilai x Responden
2	5	Sangat Setuju	2	10
	4	Setuju	6	24
	3	Cukup Setuju	2	6
	2	Kurang Setuju	0	0
	1	Tidak Setuju	0	0
		Jumlah	10	30

Berdasarkan data hasil kusioner tersebut, dapat dicari prosentase jawaban dengan menggunakan rumus :

$$Y = \frac{30}{10\ x\ 5} \times 100 = 60\ \%$$

Skala



Berdasarkan hasil persentase diatas maka dapat disimpulkan 60% menyatakan cukup setuju bahwa *website* yang dibangunpenyajian informasi barang di website sudah sesuai dengan informasi yang dibutuhkan.

3. Apakah Anda setuju struktur menu di website ini mudah dimengerti ?

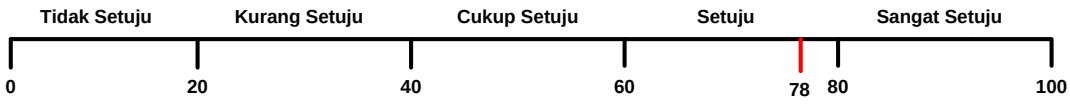
Tabel 4.21 Hasil Pengujian Kuesioner soal nomor 3 (Situs *Frontend*)

Pertanyaan	Nilai	Jawaban	Responden	Nilai x Responden
3	5	Sangat Setuju	4	20
	4	Setuju	2	8
	3	Cukup Setuju	3	9
	2	Kurang Setuju	1	2
	1	Tidak Setuju	0	0
		Jumlah	10	39

Berdasarkan data hasil kusioner tersebut, dapat dicari prosentase jawaban dengan menggunakan rumus :

$$Y = \frac{39}{10\ x\ 5} \times 100 = 78\ \%$$

Skala



Berdasarkan hasil persentase diatas maka dapat disimpulkan 78% menyatakan setuju bahwa struktur menu di website ini mudah dimengerti

4. Apakah Anda setuju website ini mempermudah Anda dalam melakukan transaksi pemesanan/pembelian barang ?

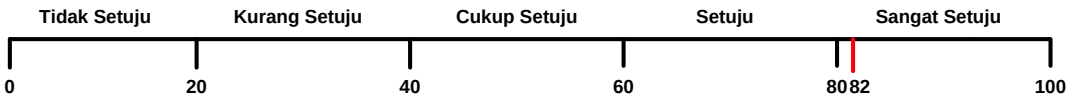
Tabel 4. 22 Hasil Pengujian Kuesioner soal nomor 4 (Situs *Frontend*)

Pertanyaan	Nilai	Jawaban	Responden	Nilai x Responden
4	5	Sangat Setuju	4	20
	4	Setuju	3	12
	3	Cukup Setuju	3	9
	2	Kurang Setuju	0	6
	1	Tidak Setuju	0	0
		Jumlah	10	41

Berdasarkan data hasil kusioner tersebut, dapat dicari prosentase jawaban dengan menggunakan rumus :

$$Y = \frac{41}{10\ x\ 5} \times 100 = 82\ \%$$

Skala



Berdasarkan hasil persentase diatas maka dapat disimpulkan 82% menyatakan sangat setuju bahwa *website* yang dibangun mempermudah anda dalam melakukan transaksi pemesanan/pembelian barang.

5. Apakah Anda setuju website ini sudah layak untuk digunakan ?

Tabel 4. 23 Hasil Pengujian Kuesioner soal nomor 5 (Situs *Frontend*)

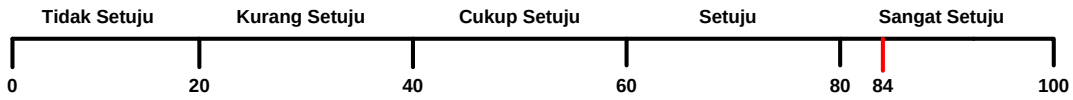
Pertanyaan	Nilai	Jawaban	Responden	Nilai x Responden
5	5	Sangat Setuju	4	20
	4	Setuju	4	16

	3	Cukup Setuju	2	6
	2	Kurang Setuju	0	0
	1	Tidak Setuju	0	0
		Jumlah	30	42

Berdasarkan data hasil kusioner tersebut, dapat dicari prosentase jawaban dengan menggunakan rumus :

$$Y = \frac{42}{10\ x\ 5} \times 100 = 84\ \%$$

Skala



Berdasarkan hasil persentase diatas maka dapat disimpulkan 84% menyatakan sangat setuju bahwa *website* yang dibangunsudah layak untuk digunakan.

Kesimpulan Pengujian Beta

Berdasarkan pengujian beta yang telah dilakukan, bahwa dapat diambil kesimpulan bahwa :

1. *Website* ini memudahkan perusahaan dalam mengolah data barang danmenjual barang.
2. *Website* inimemberikan kemudahan kepada konsumen untuk mendapat informasi tentang barang dengan lengkap.
3. *Website* ini memberikan kemudahan kepada konsumen dalam proses bertransaksi.
4. Mempermudah perusahaan dalam pembuatan laporan hasil penjualan.*webiste.*