BAB 4

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

Analisis dan desain sistem secara terperinci telah dilakukan, maka langkah selanjutnya adalah menuju tahap implementasi. Tujuan implementasi adalah untuk menjelaskan tentang manual modul kepada semua *user* yang akan menggunakan sistem. Sehingga *user* tersebut dapat merespon apa yang ditampilkan di sistem dan memberikan masukan kepada pembuat sistem untuk dilakukan perbaikan agar sistem lebih baik lagi.

4.1.1 Lingkungan Implementasi

Untuk mendukung aplikasi yang akan diterapkan pada lingkungan implementasi, maka dalam hal ini harus memperhatikan perangkat keras dan perangkat lunak yang menunjang dalam Pembangunan Aplikasi *E-commerce* pada pabrik baju 12 Titan.

4.1.2 Implementasi Perangkat Keras

Spesifikasi perangkat keras minimal yang dibutuhkan untuk mengimplementasikan aplikasi yang di bangun antara lain terdapat pada tabel 4.1.

Tabel 4.1 Perangkat Keras yang dibutuhkan

No	Perangkat Keras	Spesifikasi
1	Prosessor	AMD X2 555 3.0 GHz
2	Monitor	ukuran 14” Resolusi 1024x768 px
3	VGA	Onboard 64 Mb
4	Memori	256 MB
5	Hard Disk	40 GB (ruang kosong 2 GB)
6	Keyboard	Keyboard

7	Mouse	Mouse
8	Printer	Hitam putih dan warna
9	Jaringan Internet	LAN dan Internet

4.1.3 Implementasi Perangkat Lunak

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

Tabel 4.2 Perangkat Lunak yang digunakan

No	Perangkat Keras	Spesifikasi
1	Sistem operasi	Windows Xp
2	Web Browser	Mozilla firefox, opera, safari, internet explorer, google chrome

4.1.4 Implementasi Basis Data

Implementasi data dalam aplikasi *e-commerce* pada toko Pabrik baju 12Titan ini menggunakan satu buah DBMS (Data *Base Management System*) dan sebuah GUI *database* untuk memudahkan tugas *administrator*. DBMS yang digunakan untuk menimplentasikan datanya adalah MySQL.. Rancangan data beserta nama file query yang dilakukan adalah sebagai berikut:

1. Tabel desain

   Tabel desain digunakan untuk menyimpan data desain yang dikirimkan oleh pemesan.

```
    CREATE TABLE IF NOT EXISTS `desain` (
      `id_desain` int(11) NOT NULL auto_increment,
      `id_produk` int(11) NOT NULL,
      `gbr_desain` varchar(50) NOT NULL,
      `status` int(11) NOT NULL,
      PRIMARY KEY  (`id_desain`),
      KEY `id_produk` (`id_produk`)
      ) ENGINE=InnoDB  DEFAULT CHARSET=latin1
AUTO_INCREMENT=99 ;
```

2. Tabel detail_ukuran

Tabel detail_ukuran digunakan untuk menyimpan data ukuran dan produk.

```
CREATE TABLE IF NOT EXISTS `detail_ukuran` (
  `id_item` int(11) NOT NULL,
  `id_produk` int(11) NOT NULL,
  `id_ukuran` int(11) NOT NULL,
  `qty` int(11) NOT NULL,
  KEY `id_ukuran` (`id_ukuran`),
  KEY `id_produk` (`id_produk`),
  KEY `id_item` (`id_item`)
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

3. Tabel harga

Tabel harga berfungsi untuk menampung data harga.

```
  CREATE TABLE IF NOT EXISTS `harga` (
    `id_kategori` int(11) NOT NULL,
    `id_qty` int(11) NOT NULL,
    `harga` int(11) NOT NULL,
    KEY `id_qty` (`id_qty`),
    KEY `id_kategori` (`id_kategori`)
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

4. Tabel kabkota

Tabel kabkota digunakan untuk menyimpan data kabupaten dan kota.

```
  CREATE TABLE IF NOT EXISTS `kabkota` (
    `id_kabkota` int(10) NOT NULL default '0',
    `id_prov` int(10) default NULL,
    `nama` varchar(50) default NULL,
    `kabkota` varchar(20) default NULL,
    `ongkir` int(11) NOT NULL,
    PRIMARY KEY  (`id_kabkota`),
    KEY `id_prov` (`id_prov`),
    KEY `id_prov_2` (`id_prov`)
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5. Tabel kategori

Tabel kategori digunakan untuk menyimpan semua data kategori buku.

```
      CREATE TABLE IF NOT EXISTS `kategori` (
        `id_kategori` int(11) NOT NULL auto_increment,
        `id_parent` int(11) NOT NULL,
        `nama_kategori` varchar(30) NOT NULL,
        `gbr_kategori` varchar(100) NOT NULL,
        `status` varchar(20) NOT NULL,
        `kat_keyword` varchar(30) NOT NULL,
        `produksi_harian` int(11) default NULL,
        `maks_produksi` int(11) NOT NULL,
        PRIMARY KEY  (`id_kategori`),
        KEY `nama_kategori` (`nama_kategori`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=23 ;
```

6. Tabel keranjang

Tabel keranjang digunakan untuk menyimpan data keranjang.

```
      CREATE TABLE IF NOT EXISTS `keranjang` (
        `id_keranjang` int(11) NOT NULL auto_increment,
        `id_session` varchar(35) NOT NULL,
        `id_produk` int(11) NOT NULL,
        `teknik` varchar(10) NOT NULL,
        `kode_warna` varchar(7) NOT NULL,
        `posisi` varchar(20) NOT NULL,
        `id_desain` int(11) NOT NULL,
        `jml_s` int(11) NOT NULL,
        `jml_m` int(11) NOT NULL,
        `jml_l` int(11) NOT NULL,
        `jml_xl` int(11) NOT NULL,
        `jml_2xl` int(11) NOT NULL,
        `jml_3xl` int(11) NOT NULL,
        `jml_qty` int(11) NOT NULL,
        `berat` float NOT NULL,
        `harga` int(11) NOT NULL,
        `subtotal` int(11) NOT NULL,
        `strip` int(11) NOT NULL,
        PRIMARY KEY  (`id_keranjang`),
        KEY `kode_warna` (`kode_warna`),
        KEY `id_produk` (`id_produk`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=2 ;
```

7. Tabel *member*

Tabel *member* digunakan untuk menyimpan data *member*.

```
      CREATE TABLE IF NOT EXISTS `member` (
        `id_member` int(11) NOT NULL auto_increment,
        `nama_member` varchar(100) NOT NULL,
        `username` varchar(50) NOT NULL,
        `password` varchar(51) NOT NULL,
        `status_member` varchar(20) NOT NULL,
        `email` varchar(100) NOT NULL,
        `alamat` text NOT NULL,
        `id_kabkota` int(11) NOT NULL,
        `kodepos` varchar(5) NOT NULL,
        `notelp` varchar(15) NOT NULL,
        PRIMARY KEY  (`id_member`),
        UNIQUE KEY `email` (`email`),
        KEY `id_kabkota` (`id_kabkota`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=23 ;
```

8. Tabel *order*

Tabel *order* digunakan untuk menyimpan data *order*.

```
      CREATE TABLE IF NOT EXISTS `order` (
        `id_order` int(11) NOT NULL auto_increment,
        `tgl_order` date NOT NULL,
        `tgl_selesai` date NOT NULL,
        `jml_order` int(11) NOT NULL,
        `total_bayar` int(11) NOT NULL,
        `berat` double NOT NULL,
        `metode_pembayaran` varchar(20) NOT NULL,
        `id_member` int(11) NOT NULL,
        `nama_pemesan` varchar(100) NOT NULL,
        `notelp_pemesan` varchar(15) NOT NULL,
        `email_pemesan` varchar(100) NOT NULL,
        `nama_tujuan` varchar(100) NOT NULL,
        `alamat_tujuan` text NOT NULL,
        `id_kabkota` int(11) NOT NULL,
        `kodepos_tujuan` int(5) NOT NULL,
        `status_order` varchar(30) NOT NULL,
        `terakhir_diubah` date NOT NULL,
        `id_user` int(11) NOT NULL default '0',
        PRIMARY KEY  (`id_order`),
        KEY `id_member` (`id_member`),
        KEY `id_kabkota` (`id_kabkota`),
        KEY `id_user` (`id_user`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=102 ;
```

9. Tabel *order_item*

Tabel *order_item* digunakan untuk menyimpan data *order_item*.

```
      CREATE TABLE IF NOT EXISTS `order_item` (
        `id_item` int(11) NOT NULL auto_increment,
        `id_order` int(11) NOT NULL,
        `id_produk` int(11) NOT NULL,
        `qty` int(11) NOT NULL,
        `kode_warna` varchar(7) NOT NULL,
        `teknik` varchar(10) NOT NULL,
        `strip` int(11) NOT NULL,
        `posisi` varchar(20) NOT NULL,
        `id_desain` int(11) NOT NULL,
        `harga` int(11) NOT NULL,
        PRIMARY KEY  (`id_item`),
        KEY `id_order` (`id_order`),
        KEY `id_produk` (`id_produk`),
        KEY `kode_warna` (`kode_warna`),
        KEY `id_item` (`id_item`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=83 ;
```

10. Tabel paypal

Tabel paypal digunakan untuk menyimpan data pembayaran paypal.

```
      CREATE TABLE IF NOT EXISTS `paypal` (
        `id_paypal` int(11) NOT NULL auto_increment,
        `kode_transaksi` varchar(17) NOT NULL,
        `waktu_pembayaran` datetime NOT NULL,
        `pembayar` varchar(50) NOT NULL,
        `jml_bayar` double NOT NULL,
        `mata_uang` varchar(10) NOT NULL,
        PRIMARY KEY  (`id_paypal`)
      ) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1
;
```

11. Tabel pembayaran

Tabel pembayaran digunakan untuk menyimpan data pembayaran.

```
      CREATE TABLE IF NOT EXISTS `pembayaran` (
        `id_pembayaran` int(11) NOT NULL auto_increment,
        `id_order` int(11) NOT NULL,
        `email` varchar(50) NOT NULL,
        `bank_tujuan` varchar(10) NOT NULL,
        `jumlah_bayar` int(11) NOT NULL,
        `bank_asal` varchar(20) NOT NULL,
        `atas_nama` varchar(100) NOT NULL,
        `tgl_bayar` date NOT NULL,
```

```
        `status` varchar(15) NOT NULL,
        `tgl_konfirmasi` date NOT NULL,
        PRIMARY KEY  (`id_pembayaran`),
        KEY `id_pesanan` (`id_order`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=40 ;
```

12. Tabel produk

Tabel produk digunakan untuk menyimpan data produk.

```
      CREATE TABLE IF NOT EXISTS `produk` (
        `id_produk` int(11) NOT NULL auto_increment,
        `id_kategori` int(11) NOT NULL,
        `nama_produk` varchar(50) NOT NULL,
        `gbr_depan` varchar(100) NOT NULL,
        `gbr_belakang` varchar(100) NOT NULL,
        `desk_produk` text NOT NULL,
        `berat` float NOT NULL,
        `status` int(11) NOT NULL,
        PRIMARY KEY  (`id_produk`),
        KEY `id_kategori` (`id_kategori`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=18 ;
```

13. Tabel Provinsi

Tabel provinsi digunakan untuk menyimpan data dari provinsi.

```
      CREATE TABLE IF NOT EXISTS `provinsi` (
        `id_prov` int(10) NOT NULL default '0',
        `nama` varchar(100) default NULL,
        PRIMARY KEY  (`id_prov`)
      ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

14. Tabel Range_Qty

Tabel range_qty digunakan untuk menyimpan data range_qty.

```
      CREATE TABLE IF NOT EXISTS `range_qty` (
        `id_qty` int(11) NOT NULL auto_increment,
        `qty_min` int(11) NOT NULL,
        `qty_max` int(11) NOT NULL,
        PRIMARY KEY  (`id_qty`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=9 ;
```

15. Tabel Teknik_Cetak

Tabel teknik cetak digunakan untuk menyimpan data teknik cetak.

```
      CREATE TABLE IF NOT EXISTS `teknik_cetak` (
        `id_teknik` int(11) NOT NULL auto_increment,
        `nama_teknik` varchar(50) NOT NULL,
        `deskripsi` text NOT NULL,
        `gbr_mesin` varchar(100) NOT NULL,
        `gbr_sample` varchar(100) NOT NULL,
        `link_video` varchar(100) NOT NULL,
        PRIMARY KEY  (`id_teknik`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=6 ;
```

16. Tabel Ukuran

Tabel ukuran digunakan untuk menyimpan data ukuran.

```
      CREATE TABLE IF NOT EXISTS `ukuran` (
        `id_ukuran` int(11) NOT NULL auto_increment,
        `ukuran` varchar(5) NOT NULL,
        `gender` varchar(100) NOT NULL,
        `width` int(11) NOT NULL,
        `height` int(11) NOT NULL,
        PRIMARY KEY  (`id_ukuran`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=7 ;
```

17. Tabel *user*

Tabel *user* digunakan untuk menyimpan data *user*.

```
      CREATE TABLE IF NOT EXISTS `user` (
        `id_user` int(11) NOT NULL auto_increment,
        `username` varchar(20) NOT NULL,
        `password` varchar(51) NOT NULL,
        `email_user` varchar(50) NOT NULL,
        `level` enum('admin','operator','gudang') NOT NULL,
        `status` int(11) NOT NULL,
        PRIMARY KEY  (`id_user`)
      )     ENGINE=InnoDB          DEFAULT     CHARSET=latin1
AUTO_INCREMENT=9 ;
```

18. Tabel Warna

Tabel warna adalah tabel yang digunakan untuk menyimpan data warna.

```
      CREATE TABLE IF NOT EXISTS `warna` (
        `kode_warna` varchar(7) NOT NULL,
        `nama_warna` varchar(20) NOT NULL,
        `dihapus` char(1) NOT NULL,
        PRIMARY KEY  (`kode_warna`)
      ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

19. Tabel v_keranjang

Tabel v_keranjang adalah tabel view yang digunakan untuk melihat data keranjang.

```
CREATE OR REPLACE VIEW `v_keranjang` AS select
`r`.`id_keranjang` AS `id_keranjang`,`r`.`id_session` AS
`id_session`,`r`.`id_produk` AS `id_produk`,`k`.`id_kategori`
AS `id_kategori`,`k`.`nama_kategori` AS
`nama_kat`,`p`.`nama_produk` AS `nama_prod`,`r`.`teknik` AS
`teknik`,`r`.`kode_warna` AS `kode_warna`,`r`.`posisi` AS
`posisi`,`d`.`gbr_desain` AS `gbr_desain`,`r`.`id_desain` AS
`id_desain`,`r`.`jml_s` AS `jml_s`,`r`.`jml_m` AS
`jml_m`,`r`.`jml_l` AS `jml_l`,`r`.`jml_xl` AS
`jml_xl`,`r`.`jml_2xl` AS `jml_2xl`,`r`.`jml_3xl` AS
`jml_3xl`,`r`.`jml_qty` AS `jml_qty`,`r`.`berat` AS
`berat`,`r`.`harga` AS `harga`,`r`.`subtotal` AS
`subtotal`,`r`.`strip` AS `strip`,`k`.`gbr_kategori` AS
`gbr_kategori` from (((`keranjang` `r` join `produk` `p`
on((`r`.`id_produk` = `p`.`id_produk`))) join `kategori` `k`
on((`p`.`id_kategori` = `k`.`id_kategori`))) join `desain`
`d` on((`r`.`id_desain` = `d`.`id_desain`)));
```

20. Tabel v_member

Tabel v_keranjang adalah tabel view yang digunakan untuk melihat data member.

```
CREATE OR REPLACE VIEW `v_member` AS select `m`.`id_member`
AS `id_member`,`m`.`nama_member` AS
`nama_member`,`m`.`username` AS `username`,`m`.`password` AS
`password`,`m`.`status_member` AS `status_member`,`m`.`email`
AS `email`,`m`.`alamat` AS `alamat`,`m`.`id_kabkota` AS
`id_kabkota`,`m`.`kodepos` AS `kodepos`,`m`.`notelp` AS
`notelp`,concat(`k`.`kabkota`,_latin1' ',`k`.`nama`) AS
`nama_kabkota`,`p`.`nama` AS `nama_provinsi` from ((`member`
`m` join `kabkota` `k` on((`m`.`id_kabkota` =
`k`.`id_kabkota`))) join `provinsi` `p` on((`k`.`id_prov` =
`p`.`id_prov`)));
```

21. Tabel v_order

Tabel v_order adalah tabel view yang digunakan untuk melihat data order.

```
CREATE OR REPLACE VIEW `v_order` AS select `o`.`id_order` AS
`id_order`,date_format(`o`.`tgl_order`,_utf8'%d - %m - %Y')
AS `tgl_order`,date_format(`o`.`tgl_selesai`,_utf8'%d - %m -
%Y') AS `tgl_selesai`,`o`.`jml_order` AS
`jml_order`,`o`.`total_bayar` AS `total_bayar`,`o`.`berat` AS
`berat`,`o`.`metode_pembayaran` AS
`metode_pembayaran`,`o`.`id_member` AS
```

```
`id_member`,`o`.`nama_pemesan` AS
`nama_pemesan`,`o`.`notelp_pemesan` AS
`notelp_pemesan`,`o`.`email_pemesan` AS
`email_pemesan`,`o`.`nama_tujuan` AS
`nama_tujuan`,`o`.`alamat_tujuan` AS
`alamat_tujuan`,concat(`k`.`kabkota`,_latin1' ',`k`.`nama`)
AS `kabkota`,`p`.`nama` AS `provinsi`,`k`.`ongkir` AS
`ongkir`,`o`.`kodepos_tujuan` AS
`kodepos_tujuan`,`o`.`status_order` AS
`status_order`,`o`.`terakhir_diubah` AS
`terakhir_diubah`,`o`.`id_user` AS `id_user` from ((`order`
`o` join `kabkota` `k` on((`o`.`id_kabkota` =
`k`.`id_kabkota`))) join `provinsi` `p` on((`k`.`id_prov` =
`p`.`id_prov`)));
```

22. Tabel v_order_item

Tabel v_order_item adalah tabel view yang digunakan untuk melihat data item pada order.

```
CREATE OR REPLACE VIEW `v_order_item` AS select `o`.`id_item`
AS `id_item`,`o`.`id_order` AS `id_order`,`p`.`nama_produk`
AS `nama_produk`,`o`.`qty` AS `qty`,`o`.`kode_warna` AS
`kode_warna`,`o`.`teknik` AS `teknik`,`o`.`strip` AS
`strip`,`o`.`posisi` AS `posisi`,`o`.`id_desain` AS
`id_desain`,`o`.`harga` AS `harga` from (`order_item` `o`
join `produk` `p` on((`o`.`id_produk` = `p`.`id_produk`)));
```

23. Foreign Key

Foreign key digunakan untuk menghubungkan field yang saling berhubungan antar tabel.

```
ALTER TABLE `kabkota` ADD FOREIGN KEY ( `id_prov` )
REFERENCES `titan12_pabrikbaju`.`provinsi` (
`id_prov`
   );
ALTER TABLE `order_item` ADD FOREIGN KEY ( `id_order` )
REFERENCES `titan12_pabrikbaju`.`order` (
`id_order`
);

ALTER TABLE `order_item` ADD FOREIGN KEY ( `id_produk` )
REFERENCES `titan12_pabrikbaju`.`produk` (
`id_produk`
);

ALTER TABLE `order_item` ADD FOREIGN KEY ( `kode_warna` )
REFERENCES `titan12_pabrikbaju`.`warna` (
`kode_warna`
);
```

```
ALTER TABLE `pembayaran` ADD FOREIGN KEY ( `id_order` )
REFERENCES `titan12_pabrikbaju`.`order` (
`id_order`
);
ALTER TABLE `order` ADD FOREIGN KEY ( `id_kabkota` )
REFERENCES `titan12_pabrikbaju`.`kabkota` (
`id_kabkota`
);

ALTER TABLE `order` ADD FOREIGN KEY ( `id_user` ) REFERENCES
`titan12_pabrikbaju`.`user` (
`id_user`
);
ALTER TABLE `detail_ukuran` ADD FOREIGN KEY ( `id_produk` )
REFERENCES `titan12_pabrikbaju`.`produk` (
`id_produk`
);

ALTER TABLE `detail_ukuran` ADD FOREIGN KEY ( `id_ukuran` )
REFERENCES `titan12_pabrikbaju`.`ukuran` (
`id_ukuran`
);
ALTER TABLE `harga` ADD FOREIGN KEY ( `id_kategori` )
REFERENCES `titan12_pabrikbaju`.`kategori` (
`id_kategori`
);

ALTER TABLE `harga` ADD FOREIGN KEY ( `id_qty` ) REFERENCES
`titan12_pabrikbaju`.`range_qty` (
`id_qty`
);

ALTER TABLE `keranjang` ADD FOREIGN KEY ( `id_produk` )
REFERENCES `titan12_pabrikbaju`.`produk` (
`id_produk`
);

ALTER TABLE `keranjang` ADD FOREIGN KEY ( `kode_warna` )
REFERENCES `titan12_pabrikbaju`.`warna` (
`kode_warna`
);

ALTER TABLE `keranjang` ADD FOREIGN KEY ( `id_desain` )
REFERENCES `titan12_pabrikbaju`.`desain` (
`id_desain`
);

ALTER TABLE `order` ADD FOREIGN KEY ( `id_member` )
REFERENCES `titan12_pabrikbaju`.`member` (
`id_member`
);
```

4.1.5 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. Implementasi *Class E-commerce* di Pabrik Baju 12Titan dapat dilihat pada tabel 4.3.

Tabel 4.3 Implementasi *Class*

No	Nama *Class*	Nama File Fisik
1	Kategori_control	Kategori.php
2	Produk_control	Produk.php
3	Pesanan_control	Order.php
4	Pembayaran_control	Pembayaran.php
5	User_control	User.php
6	Member_control	Member.php
7	Login_control	Login.php
8	Logout_control	Logout.php
9	Harga_control	Harga.php
10	Warna_control	Warna.php
11	Kategori_model	Kategori_model.php
12	Produk_model	Produk_model.php
13	Pesanan_model	Order_model.php
14	Pembayaran_model	Pembayaran_model.php
15	User_model	User_model.php
16	Member_model	Member_model.php
17	Login_model	Login_model.php
18	Harga_model	Harga_model.php
19	Warna_model	Warna_model.php

4.1.6 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. Implementasi antarmuka admin pada aplikasi *e-ecommerce* pada toko Pabrik baju 12Titan dapat dilihat pada tabel 4.4.

Tabel 4.4 Implementasi Antarmuka Admin

Menu	Deskripsi	Nama File
Login Admin	Digunakan sebagai halaman login *admin*	admiiin/index.php/Login
Halaman Utama Admin	Halaman utama ketika *admin* berhasil login yang berisi 5 order terakhir dan 5 pembayaran terakhir.	admiiin/index.php/Dashboard
Pengolahan *User*	Halaman *admin* yang digunakan untuk melihat, mencari, menambah, mengubah dan menghapus data user	admiiin/index.php/User
Pengolahan kategori	Halaman *admin* yang digunakan untuk melihat, menambah, mencari, mengubah dan menghapus data kategori.	admiiin/index.php/Kategori
Pengolahan produk	Halaman *admin* yang digunakan untuk melihat, menambah, mencari mengubah dan menghapus data produk.	admiiin/index.php/Produk
Pengolahan member	Halaman *admin* yang digunakan untuk melihat, mencari, menambah, mengubah dan menonaktifkan member.	admiiin/index.php/member
Pengolahan warna	Halaman *admin* yang digunakan untuk melihat, menambah, mencari mengubah dan menghapus data warna.	admiiin/index.php/warna
Pengolahan desain	Halaman *admin* yang digunakan untuk melihat, menambah, mencari mengubah dan menghapus data desain.	admiiin/index.php/desain
Pengolahan harga	Halaman *admin* yang digunakan untuk mengubah harga baju.	admiiin/index.php/harga
Pengolahan order	Halaman yang digunakan *admin* untuk melihat, mengubah dan menambah data order	admiiin/index.php/order
Pengolahan pembayaran	Halaman yang digunakan *admin* melihat , menambah, mencari, dan mengubah status pembayaran.	admiiin/index.php/pembayaran

Menu	Deskripsi	Nama File
Pembuatan laporan	Halaman yang digunakan admin untuk melihat buku terjual dan mencetak laporan dalam periode tertentu	admiiin/index.php/laporan
Logout	Halaman yang digunakan untuk menghapus session admin	index.php/admin/destroy

Tabel dari implementasi dari pengunjung dapat dilihat pada tabel 4.5.

Tabel 4.5 Implementasi Antarmuka Pengunjung dan Member

Menu	Deskripsi	Nama File
Beranda	Digunakan sebagai halaman utama konsumen berisi informasi mengenai buku terbaru, buku terlaris, kategori buku dan konsumen dapat melakukan tracking JNE untuk mengecek keberadaan barang yang sedang dikirim.	index.php/katalog
Teknik Kami:	Digunakan untuk menampilkan infromasi mengenai teknik yang digunakan.	Pb12Titan/index.php/teknik/
- Screen Printing	Halaman yang digunakan untuk menampilkan informasi tentang screen printing	Pb12Titan/index.php/teknik/detail/1
- Transfer Printing	Halaman yang digunakan untuk menampilkan informasi tentang screen printing	Pb12Titan/index.php/teknik/detail/2
- Direct-to-Garment	Halaman yang digunakan untuk menampilkan informasi tentang screen printing	Pb12Titan/index.php/teknik/detail/3
- Embroidery	Halaman yang digunakan untuk menampilkan informasi tentang Embroidery printing	Pb12Titan/index.php/teknik/detail/4
- Digital Printing	Halaman yang digunakan untuk menampilkan informasi tentang digital printing	Pb12Titan/index.php/teknik/detail/5
Katalog	Halaman yang digunakan member untuk melihat data smua kategori yang tersedia	index.php/kategori
Order :	Halaman yang digunakan untuk menampilkan cara order, konfirmasi pembayaran dan cara pembayaran	index.php/order/
- Cara Order	Halaman yang digunakan untuk menampilkan langkah-langkah order	index.php/order/cara_order
- Konfirmasi Pembayaran	Halaman yang digunakan untuk menampilkan cara order	index.php/order/pembayaran
- Cara Pembayaran	Halaman yang digunakan untuk menampilkan cara pembayaran	index.php/order/cara_pembayaran
Tentang Kami	Halaman yang digunakan member untuk melihat prodif perusahaan pabrik baju 12 Titan	index.php/profil
Keranjang	Halaman yang dugunakan untuk melihat produk yang akan dibeli	/index.php/order/keranjang

Menu	Deskripsi	Nama File
Akun	Halaman yang digunakan untuk melihat, mengubah data akun member	/index.php/member
Peta Situs	Halaman yang digunakan untuk melihat apa saja menu yang tersedia pada website baprik baju 12 titan	/index.php/sitemap
Kontak	Halaman yang digunakan untuk mengirimkan pertanyaan kepada pabrik baju 12 Titan	/index.php/contact

4.2 Pengujian Sistem

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 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* di pabrik baju 12 Titan menggunakan data uji berdasarkan data dari pabrik baju 12 Titan. Rencana pengujian selengkapnya terlihat pada tabel 4.6.

Tabel 4.6 Rencana Pengujian Aplikasi

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Isi data Login	*Black Box*
	Validasi username and password	*Black Box*
Pengolahan Data Kategori	Tambah Kategori	*Black Box*
	Ubah Kategori	*Black Box*
	Hapus Kategori	*Black Box*
	Cari Kategori	*Black Box*
Pengolahan Data *User*	Tambah *User*	*Black Box*
	Ubah *User*	*Black Box*
	Hapus *User*	*Black Box*
	Cari *User*	*Black Box*
Pengolahan Data Produk	Tambah Produk	*Black Box*
	Ubah Produk	*Black Box*
	Hapus Produk	*Black Box*
	Cari Produk	*Black Box*
Pengolahan Data Member	Cari Member	*Black Box*
Pengolahan Data Warna	Tambah Warna	*Black Box*
	Ubah Warna	*Black Box*
	Hapus warna	*Black Box*
	Cari Warna	*Black Box*
Pengolahan Data Harga	Ubah data harga	*Black Box*
Pengolahan Data Order	Ubah status pemesanan	*Black Box*
Pengolahan Data Pembayaran	Cek status pembayaran	*Black Box*
Pembuatan laporan	Pembuatan laporan	*Black Box*
Daftar member	Pendaftaran sebagai member	*Black Box*
Pengolahan akun	Ubah akun	*Black Box*
Lihat Produk	Menampilkan detail produk	*Black Box*
Order Produk	Tampil informasi mengenai panduan cara berbelanja dan cara pembayaran	*Black Box*
Pencarian Produk	Menampilkan data produk yang dicari	*Black Box*
Backup dan Restore *Database*	Backup *Database*	*Black Box*
	Restore *Database*	*Black Box*
Lupa password	Menampilkan password member	*Black Box*
Logout	Keluar dari sistem sebagai member	*Black Box*

4.2.2 Pengujian Alpha

Hasil implementasi perangkat lunak dari aplikasi *e-commerce* yang dibangun harus diuji terlebih dahulu agar sistem yang dibangun dapat berjalan dengan baik. Pengujian perangkat lunak sistem penerapan *e-commerce* pada toko Pabrik baju 12Titan menggunakan data uji berdasarkan form yang terdapat didalam aplikasi.

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 Benar (Data Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Username : admin *Password* : pass	Tercantum pada *text field username* dan *password*	Dapat mengisi data *login* sesuai yang diharapkan.	[√] Diterima [ ] Ditolak
Data Login berupa *Username* : admin *Password* : pass	Data login dicari dalam *database*, *username* dan data *password* benar login *valid* dan dapat masuk ke menu utama	*Username* dan *Password* benar dapat masuk ke menu utama	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji salah (Data Salah)			
Username : admin *password* : admin	Tidak dapat login/masuk dan menampilkan 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 Benar (Data Benar)			
Data Masukan	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 Benar (Data Benar)			
Tambah kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah data kategori	Menampilkan *form* tambah kategori	Sistem menampilkan *form* tambah kategori	[√] Diterima [ ] Ditolak
Mengisi data kategori contoh : Kategori : T_Shirts Jml produksi harian : 10 Jml maksimal produksi : 10	Tercantun dalam masing-masing *field* kemudian data kategori tersimpan dalam *database*.	Data kategori berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kategori contoh : Kategori : T_Shirts Jml produksi harian : 10 Jml maksimal produksi : 10	Kategori tidak dapat disimpan dalam *database* karena data kategori sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data kategori dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

4. Pengujian ubah kategori

   Pengujian ubah kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah kategori dilihat pada tabel 4.10.

Tabel 4.10 Pengujian ubah kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah kategori	Menampilkan form ubah kategori beserta data yang akan diubah	Dapat menampilkan form ubah kategori beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data kategori lama contoh : Kategori : T_Shirts Jml produksi harian : 10 Jml maksimal produksi : 10	Tercantun dalam *field* kategori berupa nama kategori.	Dapat menampilkan field data kategori yang akan diubah.	[√] Diterima [ ] Ditolak
Data kategori yang akan diubah contoh : Kategori : T_Shirts 2 Jml produksi harian : 7 Jml maksimal produksi : 10	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
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategori yang akan diubah contoh : Kategori : T_Shirts 2 Jml produksi harian : A7x Jml maksimal produksi : 10	Data kategori tidak berhasil diubah dan menampilkan pesan kesalahan, tipe data jumlah produksi harian adalah integer	Sistem tidak dapat menyimpan data kategori dalam ke dalam *database* dan muncul pesan kesalahan	[√] 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 Benar (Data Benar)			
Hapus kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategori yang akan dihapus contoh : Kategori : T_Shirts 2 Jml produksi harian : 7 Jml maksimal produksi : 10	Tercantum data kategori yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data kategori yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

6. Pengujian cari kategori

Pengujian cari kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari kategori dilihat pada tabel 4.12

Tabel 4.12 Pengujian cari kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data kategori	Menampilkan data kategori	Berhasil menampilkan data kategori	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama kategori contoh : T-Shirts	Menampilkan data kategori berdasarkan nama kategori yang dicari	Berhasil Menampilkan data kategori berdasarkan nama kategori yang dicari	[√] Diterima [ ] Ditolak

7. Pengujian tambah *user*

   Pengujian tambah *user* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah *user* dilihat pada tabel 4.13

Tabel 4.13 Pengujian tambah user

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah user			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah *user*	Menampilkan *form* tambah *user*	Sistem berhasil menampilkan *form* tambah *user*	[√] Diterima [ ] Ditolak
Mengisi data *user* contoh : *username* : admin *password* : admin *level* : admin	Tercantun dalam masing-masing *field* kemudian data *user* tersimpan dalam *database*.	Data *user* berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah user			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data *user* contoh : *username* : admin *password* : admin level : -	Data *user* tidak dapat disimpan dalam *database* karena data level belum terisi	Sistem tidak dapat menyimpan data *user* dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

8. Pengujian ubah *user*

   Pengujian ubah *user* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah *user* dilihat pada tabel 4.14

Tabel 4.14 Pengujian ubah *user*

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah user			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah *user*	Menampilkan form ubah *user* beserta data yang akan diubah	Dapat menampilkan form ubah *user* beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data *user* lama yang akan diubah contoh : *username* : admin *password* : admin *level* : admin	Tercantun dalam *field user* berupa *username*, *password*, dan *level*	Dapat menampilkan field data *user* yang akan diubah.	[√] Diterima [ ] Ditolak

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data *user* contoh : *username* : admin *password* : admin *level* : admin	Dapat mengisi *field* data yang akan diubah dengan benar dan data *user* berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data *user* berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah user			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *user* yang akan diubah contoh : *username* : admin itu adalah administrator punya semua hak akses *password* : admin *level* : admin	Data *user* tidak berhasil diubah dan menampilkan pesan kesalahan, maxlength *username* adalah 20	Sistem tidak dapat menyimpan data *user* dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

9. Pengujian hapus *user*

   Pengujian ubah *user* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus *user* dilihat pada tabel 4.15

Tabel 4.15 Pengujian hapus user

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus user			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data *user* yang akan dihapus contoh : *username* : admin *password* : admin *level* : admin	Tercantum data *user* yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data *user* yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

10. Pengujian tambah produk

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

Tabel 4.16 Pengujian tambah produk

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah produk	Menampilkan *form* produk	Sistem menampilkan *form* tambah produk	[√] Diterima [ ] Ditolak
Mengisi data produk contoh : Kategori : T-Shirts Nama Produk : Kerah V Deskripsi: - Teknik : Digital Printing	Tercantun dalam masing-masing *field* kemudian data produk tersimpan dalam *database*.	Data produk berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data produk contoh : Kategori : T-Shirts Nama Produk : Kerah V Deskripsi: - Teknik : Digital Printing	produk tidak dapat disimpan dalam *database* karena data produk sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data produk dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

11. Pengujian ubah produk

Pengujian ubah produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah produk dilihat pada tabel 4.17

Tabel 4.17 Pengujian ubah produk

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah produk	Menampilkan form ubah produk beserta data yang akan diubah	Dapat menampilkan form ubah produk beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data produk lama contoh : Kategori : T-Shirts Nama Produk : Kerah V Deskripsi: - Teknik : Digital Printing	Tercantun dalam *field* nama produk, deskripsi, teknik dan subkategori.	Dapat menampilkan field data produk yang akan diubah.	[√] Diterima [ ] Ditolak

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data produk yang akan diubah contoh : Kategori : T-Shirts Nama Produk : Kerah Bagus Deskripsi: - Teknik : Digital Printing	Dapat mengisi *field* data yang akan diubah dengan benar dan data subkategori berhasil diubah dan disimpan dalam *database*,	Dapat mengisi *field* data yang akan diubah dengan benar dan data subkategori berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah subkategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data subkategori yang akan diubah contoh : Kategori : T-Shirts Nama Produk : kerah baju adalah sebuah bagian dari baju yang letaknya di leher Deskripsi: - Teknik : Digital Printing	Data produk tidak berhasil diubah dan menampilkan pesan kesalahan, maxlength nama produk adalah 50	Sistem tidak dapat menyimpan data subkategori dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

12. Pengujian hapus produk

Pengujian hapus produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus produk dilihat pada tabel 4.18

Tabel 4.18 Pengujian hapus produk

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data produk yang akan dihapus : contoh : Kategori : T-Shirts Nama Produk : kerah baju adalah sebuah bagian dari baju yang letaknya di leher Deskripsi: - Teknik : Digital Printing	Tercantum data produk yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data produk yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

13. Pengujian cari produk

Pengujian cari produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari produk dilihat pada tabel 4.19.

Tabel 4.19 Pengujian cari produk

Kasus dan Hasil Uji Benar (Data Benar)			
Cari subkategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data produk	Menampilkan data produk	Berhasil menampilkan data produk	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama produk contoh : Jaket	Menampilkan data produk berdasarkan nama produk yang dicari	Berhasil Menampilkan data produk berdasarkan nama produk yang dicari	[√] Diterima [ ] Ditolak

14. Pengujian cari *member*

Pengujian cari *member* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari *member* dilihat pada tabel 4.20

Tabel 4.20 Pengujian cari *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Cari member			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Member	Menampilkan data *member*	Berhasil menampilkan data produk	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama *member* contoh : Bobby	Menampilkan data *member* berdasarkan nama produk yang dicari	Berhasil Menampilkan data *member* berdasarkan nama *member* yang dicari	[√] Diterima [ ] Ditolak

15. Pengujian tambah warna

Pengujian tambah warna dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah warna dilihat pada tabel 4.21.

Tabel 4.21 Pengujian tambah warna

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah warna			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah warna	Menampilkan *form* warna	Sistem menampilkan *form* tambah warna	[√] Diterima [ ] Ditolak
Mengisi data warna contoh : Nama warna : putih Kode warna : ffffff	Tercantun dalam masing-masing *field* kemudian data warna tersimpan dalam *database*.	Data produk berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah warna			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data warna contoh : Nama warna : putih Kode warna : ffffff	warna tidak dapat disimpan dalam *database* karena data produk sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data warna dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

16. Pengujian ubah warna

Pengujian ubah warna dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah warna dilihat pada tabel 4.22.

Tabel 4.22 Pengujian ubah warna

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah warna			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah warna	Menampilkan form ubah warna beserta data yang akan diubah	Dapat menampilkan form ubah warna beserta data yang akan diubah	[√] Diterima [ ] Ditolak

Data warna lama contoh : Nama warna : putih Kode warna : ffffff	Tercantun dalam *field* warna berupa nama warna dan kode warna.	Dapat menampilkan field data warna yang akan diubah.	[√] Diterima [ ] Ditolak
Data warna yang akan diubah contoh : Nama warna : hitam Kode warna : 000000	Dapat mengisi *field* data yang akan diubah dengan benar dan data warna berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data warna berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah warna			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data warna yang akan diubah contoh : Nama warna : hitam Kode warna : abcdefghij	Data penerbit tidak berhasil diubah dan menampilkan pesan kesalahan, kode warna tidak ada	Sistem tidak dapat menyimpan data warna dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

17. Pengujian hapus warna

Pengujian hapus warna dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus warna dilihat pada tabel 4.23

Tabel 4.23 Pengujian hapus warna

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus penerbit			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data warna yang akan dihapus contoh : Nama warna : putih Kode warna : ffffff	Tercantum data warna yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data penerbit yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

18. Pengujian cari warna

Pengujian cari warna dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari warna dilihat pada tabel 4.24

Tabel 4.24 Pengujian cari warna

Kasus dan Hasil Uji Benar (Data Benar)			
Cari warna			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data warna	Menampilkan data warna	Berhasil menampilkan data penerbit	[√] Diterima [ ] Ditolak
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data yang akan dicari berdasarkan nama warna contoh : putih	Menampilkan data warna berdasarkan nama penerbit yang dicari	Berhasil Menampilkan data warna berdasarkan nama warna yang dicari	[√] Diterima [ ] Ditolak

19. Pengujian tambah desain

Pengujian tambah desain dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah desain dilihat pada tabel 4.25

Tabel 4.25 Pengujian tambah desain

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah desain			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tambah desain	Menampilkan *form* tambah desain	Sistem berhasil menampilkan *form* tambah desain	[√] Diterima [ ] Ditolak
Mengisi data desain contoh : nama desain : batik gambar : img1.jpg	Tercantun dalam masing-masing *field* kemudian data desain tersimpan dalam *database*.	Data desain berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Tambah desain			

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data desain contoh : nama desain : batik gambar : img1.jpg	Data desain tidak dapat disimpan dalam *database* karena data desain sudah ada dan muncul pesan kesalahan	Sistem tidak dapat menyimpan data desain dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

20. Pengujian ubah desain

Pengujian ubah desain dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah desain dilihat pada tabel 4.26

Tabel 4.26 Pengujian ubah desain

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah desain			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah desain	Menampilkan form ubah desain beserta data yang akan diubah	Dapat menampilkan form ubah desain beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data desain lama contoh : nama desain : batik gambar : img1.jpg	Tercantun dalam *field* buku berupa nama desain dan gamabar.	Dapat menampilkan field data desain yang akan diubah.	[√] Diterima [ ] Ditolak
Data desain yang akan diubah contoh : nama desain : batik cirebon gambar : img1.jpg	Dapat mengisi *field* data yang akan diubah dengan benar dan data desain berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data desain berhasil diubah dan disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah desain			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data desain yang akan diubah contoh : nama desain : batik cirebon gambar : img1.psd	Data desain tidak berhasil diubah dan menampilkan pesan kesalahan, tipe data gambar salah	Sistem tidak dapat menyimpan data desain dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

21. Pengujian hapus desain

Pengujian ubah desain dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian desain buku dilihat pada tabel 4.27

Tabel 4.27 Pengujian hapus desain

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus desain			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data desain yang akan dihapus contoh : nama desain : batik cirebon gambar : img1.jpg	Tercantum data desain yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data desain yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

22. Pengujian cari desain

Pengujian cari desain dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari desain dilihat pada tabel 4.28

Tabel 4.28 Pengujian cari desain

Kasus dan Hasil Uji Benar (Data Benar)			
Cari desain			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data desain	Menampilkan data desain	Berhasil menampilkan data desain	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama desain contoh : batik	Menampilkan data desain berdasarkan nama buku yang dicari	Berhasil Menampilkan data desain berdasarkan nama buku yang dicari	[√] Diterima [ ] Ditolak

23. Pengujian ubah harga

Pengujian ubah harga dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah harga dilihat pada tabel 4.29

Tabel 4.29 Pengujian ubah harga

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah harga			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Pengolahan harga	Menampilkan *form* pengolahan harga produk	Berhasil menampilkan form pengolahan harga produk	[√] Diterima [ ] Ditolak
Mengisi data harga produk lama contoh : nama produk : T-Shirts Harga : Rp 20000	Tercantun dalam *field* harga dan data harga tersimpan dalam *database*	Data harga berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Data harga produk yang akan diubah contoh : nama produk : T-Shirts Harga : Rp 25000	Dapat mengisi *field* data yang akan diubah dengan benar dan data harga berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data desain berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah harga			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data harga produk yang akan diubah contoh : nama produk : T-Shirts Harga : Rp 25000abc	Data harga produk tidak berhasil diubah dan menampilkan pesan kesalahan, tipe data pemesanan adalah *integer*	Sistem tidak berhasil menyimpan data harga produk	[√] Diterima [ ] Ditolak

24. Pengujian ubah status order

Pengujian ubah status order dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah status order dapat dilihat pada tabel 4.30

Tabel 4.30 Pengujian ubah status order

Kasus dan Hasil Uji Benar (Data Benar)			
Pengolahan data order			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Pilih pengolahan order	Tampil data order beserta status pembayaran.	Sistem berhasil menampilkan data order beserta status pembayaran	[√] Diterima [ ] Ditolak
Pilih detail order	Tampil data order sesuai dengan detail order yang dipilih	Sistem berhasil menampilkan data order sesuai dengan detail order yang dipilih	[√] Diterima [ ] Ditolak
Pilih pembayaran sudah dicek	Menyimpan status pembayaran menjadi lunas	Sistem berhasil menyimpan status pembayaran menjadi lunas	[√] Diterima [ ] Ditolak
Memasukan status pengiriman dan nomor resi pengiriman	Jika salah satu field yang bertanda wajib diisi tidak terisi atau salah dalam pengisian, maka akan muncul pesan kesalahan bahwa field tersebut harus terisi.	Sistem dapat memunculkan pesan kesalahan bahwa field tersebut harus terisi jika salah satu field tidak terisi atau salah dalam pengisian	[√] Diterima [ ] Ditolak
Pilih tombol Simpan	Data pemesanan berhasil disimpan dengan status pengiriman terkirim beserta nomor resi pengiriman	Data berhasil menyimpan data dengan status pengiriman terkirim beserta nomor resi pengiriman	[√] Diterima [ ] Ditolak

25. Pengujian ubah status pembayaran

Pengujian ubah status pembayaran dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah status pembayaran dapat dilihat pada tabel 4.31

Tabel 4.31 Pengujian ubah status pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Pengujian kirim promosi			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Pengolahan pembayaran	Tampil data pembayaran yang menampilkan data pembayaran dari member.	Sistem berhasil menampilkan data pembayaran.	[√] Diterima [ ] Ditolak

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Tandai Sudah dicek	Status pembayaran dari pemesanan menjadi lunas	Sistem berhasil merubah status pembayaran dari pemesanan menjadi lunas	[√] Diterima [ ] Ditolak

26. Pengujian pembuatan laporan

Pengujian pembuatan laporan penjualan dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian pembuatan laporan penjualan dilihat pada tabel 4.32

Tabel 4.32 Pengujian pembuatan laporan

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah user			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Cetak laporan penjualan	Menampilkan form laporan penjualan	Dapat menampilkan form laporan penjualan	[√] Diterima [ ] Ditolak
Memasukan bulan dan tahun contoh : Bulan : mei Tahun : 2011	Tercantun dalam *field* laporan berupa bulan dan tahun	Dapat menampilkan data laporan pada periode yang dipilih.	[√] Diterima [ ] Ditolak
Cetak laporan	Menampilkan laporan penjualan pada periode tertentu dalam bentuk *file* pdf.	Dapat menampilkan laporan penjualan pada periode tertentu dalam benuk *file* pdf	[√] Diterima [ ] Ditolak

27. Pengujian ubah akun

Pengujian ubah akun dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah akun dilihat pada tabel 4.33

Tabel 4.33 Pengujian ubah akun

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah user			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah akun	Menampilkan form ubah akun beserta data yang akan diubah	Dapat menampilkan form ubah akun beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data akun lama yang akan diubah contoh : *username* : admin *password* : admin	Tercantun dalam *field* akun berupa *username*, *password*	Dapat menampilkan field data akun yang akan diubah.	[√] Diterima [ ] Ditolak

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data akun contoh : *username* : admin1 *password* : admin1	Dapat mengisi *field* data yang akan diubah dengan benar dan data akun berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data akun berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah user			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data akun yang akan diubah contoh : *username* : admin itu adalah administrator punya semua hak akses *password* : admin *level* : admin	Data akun tidak berhasil diubah dan menampilkan pesan kesalahan, maxlength *username* adalah 20	Sistem tidak dapat menyimpan data akun dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

28. Pengujian pendaftaran sebagai *member*

Pengujian pendaftaran sebagai *member* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian pendaftaran sebagai *member* dilihat pada tabel 4.34

Tabel 4.34 Pengujian pendaftaran sebagai *member*

Kasus dan Hasil Uji Benar (Data Benar)			
Pendaftaran sebagai *member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Daftar *member*	Menampilkan form daftar *member*	Dapat menampilkan form daftar member	[√] Diterima [ ] Ditolak
Masukan data pendaftaran, contoh masukan : Nama : bobby permana Email : bobby.cool@gmail.com username : bobby password : bobby Kota : Bandung Kodepos : 40514 Telepon : 085720016479 konfirmasi password : bobby	Sistem menyimpan data pendaftaran kedalam *database*	Berhasil menyimpan data pendaftaran member kedalam *database*	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji Salah (Data Salah)			
pendaftaran sebagai *member*			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Masukan data pendaftaran, contoh masukan : Nama : bobby permana Email : bobby.cool@gmail.com username : bobby password : bobby Kota : Bandung Kodepos : 40514 Telepon : 085720016479 konfirmasi password : 123221	Tampil pesan kesalahan field password dengan field password konfirmasi tidak sama	Dapat pesan kesalahan field password dengan field password konfirmasi tidak sama	[√] Diterima [ ] Ditolak

29. Pengujian ubah akun

Pengujian ubah akun dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian ubah akun dilihat pada tabel 4.35

Tabel 4.35 Pengujian ubah akun

Kasus dan Hasil Uji Benar (Data Benar)			
Ubah akun			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Ubah akun	Menampilkan form ubah akun beserta data yang akan diubah	Dapat menampilkan form ubah akun beserta data yang akan diubah	[√] Diterima [ ] Ditolak
Data akun lama yang akan diubah contoh : Nama : bobby permana Email : if.10107259@yahoo.co username : bobby password : bobby Kota : Bandung Kodepos : 40514 Telepon : 085720016479 konfirmasi password : bobby	Tercantun dalam *field* akun berupa *username*, *password*, nama, email, jenis-kelamin, kota, kodepos, telepon	Dapat menampilkan field data akun yang akan diubah.	[√] Diterima [ ] Ditolak

Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data akun contoh : Nama : bobby permana Email : if.10107259@yahoo.com username : bobby password : 12345 Kota : Cimahi Kodepos : 40514 Telepon : 085720016479 konfirmasi password : 12345	Dapat mengisi *field* data yang akan diubah dengan benar dan data akun berhasil diubah dan disimpan dalam *database*.	Dapat mengisi *field* data yang akan diubah dengan benar dan data akun berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah akun			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data akun yang akan diubah contoh : Nama : bobby permana Email : if.10107259@yahoo.com username : bobby password : 12345 Kota : Cimahi Kodepos : 40514 Telepon : 085720016479 konfirmasi password : 12345678	Data akun tidak berhasil diubah dan menampilkan pesan kesalahan, password dengan password konfirmasi tidak sama	Sistem tidak dapat menyimpan data akun dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

30. Pengujian tampil data kategori

Pengujian menampilkan data kategori dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian menampilkan data kategori dilihat pada tabel 4.36.

Tabel 4.36 Pengujian tampil data kategori

Kasus dan Hasil Uji Benar (Data Benar)			
Tampil data kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
kategori	Menampilkan data kategori yang dipilih	Dapat menampilkan data kategori	[√] Diterima [ ] Ditolak

31. Tampil detail produk

Pengujian tampil detail produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tampil detail produk dilihat pada tabel 4.37.

Tabel 4.37 Pengujian tampil detail produk

Kasus dan Hasil Uji Benar (Data Benar)			
Tampil detail produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Pilih Produk	Menampilkan data detail produk yang dipilih	Dapat menampilkan detail dari produk yang dipilih	[√] Diterima [ ] Ditolak

32. Pengujian tambah produk ke keranjang

Pengujian tambah produk ke keranjang dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tambah produk ke keranjang dilihat pada tabel 4.38.

Tabel 4.38 Pengujian tambah produk ke keranjang

Kasus dan Hasil Uji Benar (Data Benar)			
Tambah produk ke keranjang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Detail produk	Menampilkan detail produk yang akan dibeli	Sistem menampilkan detail produk yang kaan dibeli	[√] Diterima [ ] Ditolak
membeli produk dengan memasukan data produk contoh : produk : V neck T-Shirt Jumlah : 25	Buku masuk kedalam keranjang belanja.	Sistem berhasil memasukan data buku kedalam keranjang belanja	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji Salah (Data Salah)			
Tambah produk ke keranjang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
membeli produk dengan memasukan data julah produk contoh : Jumlah : 36B	Buku gagal masuk ke keranjang, dan keluar pesan bahwa tipe data jumlah adalah integer.	Sistem gagal memasukan data produk ke keranjang dan menampilkan pesan kesalahan	[√] Diterima [ ] Ditolak

33. Pengujian *update* keranjang

Pengujian *update* keranjang dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian *update* keranjang dilihat pada tabel 4.39

Tabel 4.39 Pengujian *update* keranjang

Kasus dan Hasil Uji Benar (Data Benar)			
Update keranjang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Keranjang belanja	Menampilkan halaman keranjang belanja yang berisi buku yang telah dibeli	Sistem Menampilkan halaman keranjang belanja yang berisi buku yang telah dibeli	[√] Diterima [ ] Ditolak
Data keranjang lama contoh : produk : V neck T-Shirt Jumlah : 25	Tercantun dalam *field* keranjang berupa nama produk dan jumlah pemesanan	Dapat menampilkan field data keranjang yang akan diubah.	[√] Diterima [ ] Ditolak
Data keranjang yang akan diubah contoh : produk : V neck T-Shirt Jumlah : 30	data produk yang telah diubah disimpan di dalam keranjang	Dapat mengisi *field* data yang akan diubah dengan benar dan data keranjang berhasil diubah dan disimpan dalam *databsase*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Ubah kategori			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data keranjang yang akan diubah contoh : produk : V neck T-Shirt Jumlah : 30B	Data keranjang tidak berhasil diubah dan menampilkan pesan kesalahan, jumlah mempunyai tipe data integer	Sistem tidak dapat menyimpan data kategori dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

34. Pengujian hapus data produk dari keranjang

Pengujian hapus data barang dari keranjang dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian hapus data barang dari keranjang dilihat pada tabel 4.40

Tabel 4.40 Pengujian hapus data barang dari keranjang

Kasus dan Hasil Uji Benar (Data Benar)			
Hapus data produk dari keranjang			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data buku dalam keranjang yang akan dihapus : contoh : produk : V neck T-Shirt Jumlah : 30B	Tercantum data buku dalam keranjang yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	Tercantum data buku dalam keranjang yang akan dihapus berupa setelah memilih hapus maka tampil pesan peringatan apakah data akan dihapus	[√] Diterima [ ] Ditolak

35. Pengujian kontak

Pengujian kontak dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian kontak dilihat pada tabel 4.41

Tabel 4.41 Pengujian kontak

Kasus dan Hasil Uji Benar (Data Benar)			
Kontak			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Kontak kami	Menampilkan *form* kontak kami	Sistem menampilkan *form* kontak kami	[√] Diterima [ ] Ditolak
Mengisi data kontak kamu contoh : nama : bobby email : if.10107259@yahoo.com alamat : bandung pesan : hai hai hai telepon : 92344283	Tercantun dalam masing-masing *field* kemudian data kontak dan tersimpan dalam *database*.	Data konfirmasi kontak berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak

Kasus dan Hasil Uji Salah (Data Salah)			
Kontak			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data kontak contoh : nama : bobby email : if.10107259.yahoo.com alamat : bandung pesan : hai hai hai telepon : 92344283	data kontak tidak dapat disimpan dalam *database* karena format *email* salah	Sistem tidak dapat menyimpan data kontak pembayaran dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

36. Tampil *order*

Pengujian tampil *order* dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian tampil *order* dilihat pada tabel 4.42

Tabel 4.42 Pengujian tampil detail produk

Kasus dan Hasil Uji Benar (Data Benar)			
Tampil detail produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Order	Menampilkan tata cara pemesanan	Dapat menampilkan tata cara pemesanan	[√] Diterima [ ] Ditolak

37. Pengujian konfirmasi pembayaran

Pengujian konfirmasi pembayaran dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian konfirmasi pembayaran dilihat pada tabel 4.43

Tabel 4.43 Pengujian konfirmasi pembayaran

Kasus dan Hasil Uji Benar (Data Benar)			
Konfirmasi pembayaran			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Konfirmasi pembayaran	Menampilkan *form* konfirmasi pembayaran	Sistem menampilkan *form* konfirmasi pembayaran	[√] Diterima [ ] Ditolak
Mengisi data konfirmasi pembayaran contoh : nomor faktur : 30 Tanggal transfer : 10-05-2011 Atas nama : bobby	Tercantun dalam masing-masing *field* kemudian data konfirmasi dan tersimpan dalam *database*.	Data konfirmasi pembayaran berhasil disimpan dalam *database*	[√] Diterima [ ] Ditolak
Kasus dan Hasil Uji Salah (Data Salah)			
Konfirmasi pembayaran			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengisi data konfirmasi pembayaran contoh : nomor faktur : 30 Tanggal transfer : 10-05-2011 Atas nama : bobby	data konfirmasi pembayaran tidak dapat disimpan dalam *database* karena format tanggal salah	Sistem tidak dapat menyimpan data konfirmasi pembayaran dalam ke dalam *database* dan muncul pesan kesalahan	[√] Diterima [ ] Ditolak

38. Pengujian cari produk

Pengujian cari produk dilakukan dengan memasukan data untuk mendapatkan hasil yang diinginkan. Pengujian cari produk dilihat pada tabel 4.44

Tabel 4.44 Pengujian cari produk

Kasus dan Hasil Uji Benar (Data Benar)			
Cari Produk			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Data produk	Menampilkan data produk	Berhasil menampilkan data produk	[√] Diterima [ ] Ditolak
Data yang akan dicari berdasarkan nama produk contoh : T-Shirts	Menampilkan data produk berdasarkan produk buku yang dicari	Berhasil Menampilkan data produk berdasarkan nama produk yang dicari	[√] Diterima [ ] Ditolak

4.2.2.2 Kesimpulan Hasil Pengujian Alpha

Berdasarkan hasi pengujian alpha (fungsional) dengan kasus uji diatas dapat ditarik kesimpulan bahwa pembangunan *e-commerce* di pabrik Baju 12 Titan secara fungsional sistem sudah dapat digunakan dan menghasilkan output yang diharapkan.

4.2.3 Pengujian Beta

Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung di tempat penelitian dengan membagikan kuesioner kepada 10 orang dari masyarakat umum dan melakukan wawancara kepada pengguna sistem atau petugas di Pabrik Baju 12Titan. Penelitian ini dilakukan di dua lingkungan yaitu :

1. Lingkungan masyarakat umum

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

2. Lingkungan pabrik baju 12 Titan

   Penelitian di lingkungan pabrik baju 12Titan dilakukan supaya dapat mengetahui sejauh mana sistem yang dibangun dapat menjadi alternatif penyelesaian permasalahan yang ada.

Berdasarkan data hasil kusioner tersebut, dapat dicari persentase 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.

4.2.3.1 Kuesioner member

Kuesioner untuk member terdiri dari 5 pertanyaan (contoh kuesioner dapat diliihat pada lampiran) dengan menggunakan skala likert dengan skala 1 sampai 5, dengan ketentuan skala untuk tiap pertanyaan sebagai berikut :

1. Apakah anda mudah mendapatkan informasi mengenai produk beserta detailnya di situs *E-commerce* Pabrik Baju 12Titan ?

Tabel 4.45 Hasil kuisioner pertanyaan nomor 1 member

Pertanyaan	No	Keterangan	Responden	Persentase (%)
1	1	Sangat Mudah		
	2	Mudah	7	46,67
	3	Cukup Mudah	6	40
	4	Biasa Saja	2	13,33
	5	Tidak Mudah		
Jumlah			15	100

Berdasarkan hasil persentase pada tabel 4.45 maka dapat disimpulkan sebanyak 7 atau 46,67% menyatakan sesuai, 6 atau 40% menyatakan cukup sesuai dan 2 atau 13,33% menyatakan biasa saja bahwa informasi barang yang diberikan sudah sesuai dengan informasi yang dibutuhkan.

2. Apakah anda setuju bahwa tampilan *website* cukup menarik dan *user friendly*?

Tabel 4.46 Hasil kuisioner pertanyaan nomor 2 member

Pertanyaan	No	Keterangan	Responden	Persentase (%)
2	1	Sangat Setuju		
	2	Setuju	6	40
	3	Cukup Setuju	6	40
	4	Biasa Saja	3	20
	5	Tidak Setuju		
Jumlah			15	100

Berdasarkan hasil persentase pada tabel 4.46 maka dapat disimpulkan sebanyak 6 orang atau 40 % menyatakan setuju, 6 orang atau 40% menyatakan cukup setuju dan 3 orang atau 20% menyatkan biasa saja bahwa tampilan dari website *e-commerce* pada toko Pabrik baju 12Titan menarik dan *user friendly*.

3. Apakah anda setuju bahwa *website* ini lebih memudahkan dalam melakukan pemesanan barang?

Tabel 4.47 Hasil kuisioner pertanyaan nomor 3 member

Pertanyaan	No	Keterangan	Responden	Persentase (%)
3	1	Sangat Setuju	1	6,67
	2	Setuju	9	60
	3	Cukup Setuju	5	33,33
	4	Biasa Saja		
	5	Tidak Setuju		
Jumlah			15	100

Berdasarkan hasil persentase pada tabel 4.47 maka dapat disimpulkan sebanyak 1 orang atau 6,67% menyatakan sangat setuju 9 orang atau 60 % menyatakan setuju, 5 orang atau 33,33% menyatakan cukup setuju bahwa

akan lebih memudahkan pelanggan dalam melakukan transaksi pembelian barang.

4. Apakah anda setuju dengan penentuan harga dan tanggal penyelesaian berdasarkan produk yang dipesan ?

Tabel 4.48 Hasil kuisioner pertanyaan nomor 4 member

Pertanyaan	No	Keterangan	Responden	Persentase (%)
4	1	Sangat Setuju		
	2	Setuju	7	46,67
	3	Cukup Setuju	8	53,33
	4	Biasa Saja		
	5	Tidak Setuju		
Jumlah			15	100

Berdasarkan hasil persentase pada tabel 4.48 maka dapat disimpulkan sebanyak 7 orang atau 46,67 % menyatakan setuju, 8 orang atau 53,33% menyatakan cukup setuju dengan penentuan harga dan tangga selesai produk pesanan.

5. Apakah anda setuju bahwa sistem pembayaran yang digunakan lebih memudahkan dalam proses transaksi?

Tabel 4.49 Hasil kuisioner pertanyaan nomor 5 member

Pertanyaan	No	Keterangan	Responden	Persentase (%)
5	1	Sangat Setuju		
	2	Setuju	11	73,33
	3	Cukup Setuju	4	26,67
	4	Biasa Saja		
	5	Tidak Setuju		
Jumlah			15	100

Berdasarkan hasil persentase pada tabel 4.49 maka dapat disimpulkan sebanyak 11 orang atau 73,33 % menyatakan setuju dan 4 orang atau 26,67% menyatakan

cukup setuju bahwa sistem pembayaran yang digunakan akan lebih memudahkan dalam proses transaksi.

4.2.3.2 Wawancara Admin

Wawancara dilakukan kepada admin yang telah menggunakan sistem dan melakukan proses-proses yang ada dalam sistem tersebut. Wawancara dilakukan dengan mengajukan 4 pertanyaan kepada admin, hasil dari wawancara dapat dilihat sebagai berikut :

1. Apakah dengan adanya *website e-commerce* ini dapat membantu dalam sarana promosi dan memperkenalkan pabrik baju 12 Titan kepada masyarakat luas ?

   Jawaban : *Website* yang dibangun dapat membantu dalam mempromosikan produk kepada masyarakat luas khususnya kepada para pengguna internet.

   Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website yang dibangun dapat membantu dalam hal promosi produk kepada masyarakat.

2. Apakah website dapat membantu untuk memudahkan penentuan harga dan tanggal penyelesaian pesanan pada pabrik baju 12 Titan ?

   Jawaban : Harga yang ditentukan sudah sesuai dengan standar perusahaan dan tanggal penyelesaian dapat ditangani oleh perusahaan.

   Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website dapat membantu dalam proses estimasi harga dan tanggal penyelesaian pesanan.

3. Apakah website dapat memudahkan dalam mengelola data pesanan pabrik baju 12 Titan?

   Jawaban : Data pesanan lebih mudah untuk dikelola dan proses produksi yang dilakukan lebih mudah untuk dikontrol.

   Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website dapat mempermudah dalam mengelola data pesanan.

4. Apakah website dapat memudahkan dalam membuat laporan penjualan produk?

   Jawaban : Laporan lebih mudah untuk dibuat sehingga mudah untuk dicek dan mengurangi kesalahan yang terjadi.

   Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website memudahkan dalam pembauatan laporan.

4.2.3.3 Wawancara Petugas

Wawancara dilakukan kepada petugas yang telah menggunakan sistem dan melakukan proses-proses yang ada dalam sistem tersebut. Wawancara dilakukan dengan mengajukan 4 pertanyaan kepada petugas, hasil dari wawancara dapat dilihat sebagai berikut :

1. Apakah dengan adanya *website e-commerce* ini dapat membantu dalam sarana promosi dan memperkenalkan pabrik baju 12 Titan kepada masyarakat luas ?

   Jawaban : website membantu mempromosikan ke masyarakat.

Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website yang dibangun dapat membantu dalam hal promosi produk kepada masyarakat.

2. Apakah website dapat membantu untuk memudahkan penentuan harga dan tanggal penyelesaian pesanan pada pabrik baju 12 Titan ?

   Jawaban : Harga lebih mudah ditentukan dan tanggal penyelesaian dapat lebih mudah untuk dikelola.

   Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website dapat membantu dalam proses estimasi harga dan tanggal penyelesaian pesanan.

3. Apakah website dapat memudahkan dalam mengelola data pesanan pabrik baju 12 Titan?

   Jawaban : Data pesanan lebih mudah untuk dikelola dan proses produksi yang dilakukan lebih mudah untuk dikontrol.

   Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website dapat mempermudah dalam mengelola data pesanan.

4. Apakah website dapat memudahkan dalam membuat laporan penjualan produk?

   Jawaban : Laporan lebih mudah untuk dibuat sehingga mudah untuk dicek dan mengurangi kesalahan yang terjadi.

   Berdasarkan hasil wawancara yang telah dilakukan maka dapat disimpulkan bahwa website memudahkan dalam pembauatan laporan.

4.2.3.4 Kesimpulan Pengujian Beta

Berdasarkan hasil persentase diatas yang didapatkan dari pengujian betha, maka sistem penerapan *e-commerce* untuk sistem penjualan ini sudah sesuai dengan tujuan yaitu dapat membantu mengelola data pemesanan, mengoptimalkan proses produksi dan meningkatkan penjualan produk pada pabrik baju 12 Titan.