

BAB V PERANCANGAN SISTEM

Setelah tahap analisa sistem dilakukan, maka sistem telah mendapatkan gambaran apa yang harus dilakukan. Langkah selanjutnya adalah merancang bagaimana sistem tersebut dibentuk melalui perancangan sistem. Perancangan sistem terdiri dari tiga tahap yaitu *design*, *implementation* dan *testing*.

5.1 System Design / Desain Sistem

Desain sistem adalah fase ketiga dari lima fase dalam siklus hidup pengembangan sistem. Tujuan dari desain sistem adalah untuk menciptakan model fisik dari sistem yang memenuhi persyaratan desain yang ditetapkan selama fase analisa sebelumnya. Pada tahap analisa sistem sudah dikembangkan sebuah model logis dari sistem yang baru. Pada tahap ini akan dirancang desain fisik yang akan memenuhi spesifikasi yang dijelaskan dalam *system requirements document*. Tahap desain sistem mencakup desain *user interface*, desain *database*, dan desain algoritma sistem.

5.1.1 Desain Database

Untuk tahap desain *database* secara umum, yang perlu dilakukan oleh analis adalah mengidentifikasi terlebih dahulu file-file yang diperlukan oleh sistem informasi. Langkah-langkah untuk mendesain *database* adalah sebagai berikut:

5.1.1.1 Logical Model

Logical model dari sistem informasi lebih menjelaskan kepada *user* bagaimana nantinya fungsi-fungsi di sistem informasi secara logika akan bekerja. Logical model dapat digambar dengan menggunakan ERD (*Entity Relation Diagram*). ERD menggambarkan entitas dan atribut yang terlibat dalam sistem. Sebaiknya daftar entitas dan atribut dibuat terlebih dahulu agar mempermudah pembuatan ERD. Berikut ini adalah tabel daftar entitas dan atribut:

Tabel 5.1 Tabel Entitas dan atribut ERD

Entitas	Atribut
Admin	<u>Username</u> admin , password
user_data	<u>id</u> user , password, nama_user, alamat, kode_pos, kota, email, No_Telephon
Produk	<u>Id</u> produk , nama_produk, tanggal, harga, keterangan, jenis_bahan, ukuran, ukuran_hak, Gambar_1, gambar_2, gambar_3, gambar_4, gambar_5, gambar_6
Detail_pesanan	<u>Id</u> detail , id_user, id_produk, bahan_pesan, ukuran_pesan, ukuran_hak, Jumlah, harga_pesan, jumlah_pesan, jumlah_kirim
Kategori produk	<u>Id</u> kategori , id_produk,
Jenis bahan	<u>Jenis</u> bahan , gambar_bahan
Ukuran sepatu	<u>Ukuran</u>
Hak	<u>Ukuran</u> hak

5.1.1.2 Kardinalitas Relasi

Setelah daftar entitas dan atribut dibuat berikutnya adalah menentukan jenis relasi pada masing-masing entitas yang terlibat pada tabel berikut:

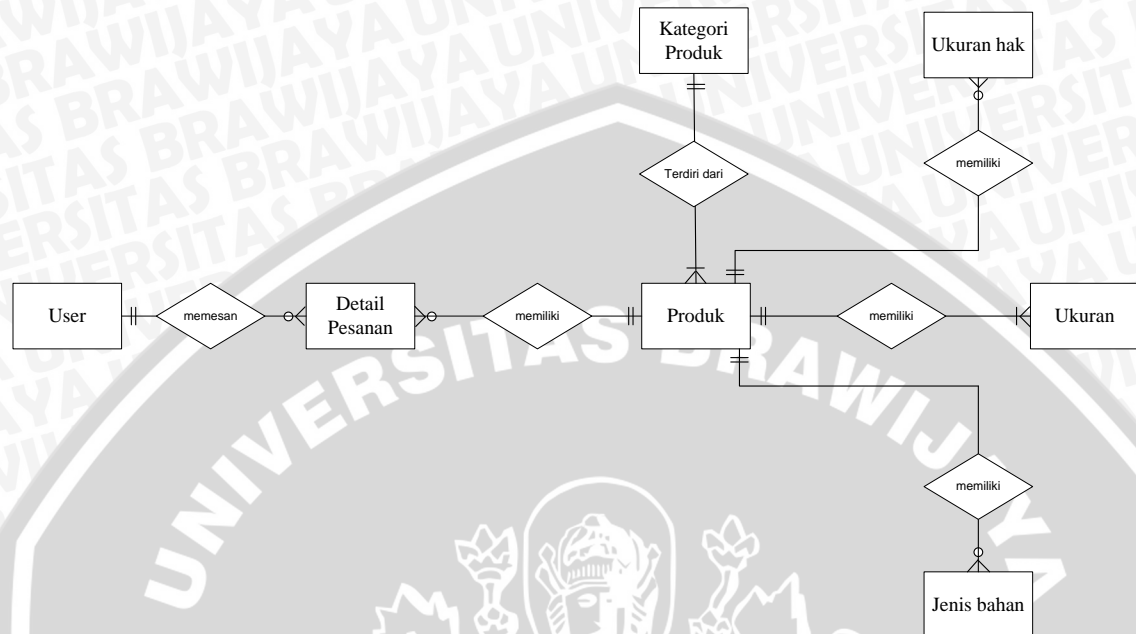
Tabel 5.2 Tabel Identifikasi Relasi

Entitas	Relasi	Entitas	Jenis Relasi
User	Memesan	Detail Pesanan	1:M
Produk	Dipesan	Detail Pesanan	1:M
Kategori produk	Terdiri dari	Produk	1:M
Produk	Memiliki	Ukuran	1:M
Produk	Memiliki	Ukuran Hak	1:M
Produk	Memiliki	Jenis Bahan	1:M

Berdasarkan Identifikasi Relasi yang telah dilakukan, maka yang perlu dilakukan adalah mengidentifikasi hubungan antar masing-masing entitas yang terlibat sebagai berikut:

- Setiap satu *user* memesan banyak detail pesanan atau tidak memesan.
- Setiap kategori produk memiliki satu atau beberapa produk di dalamnya.
- Setiap satu produk dipesan untuk banyak detail pesanan atau tidak dipesan. Setiap satu detail pesanan memiliki satu produk.
- Setiap satu produk memiliki beberapa ukuran sepatu.
- Setiap satu produk memiliki beberapa ukuran atau tidak memiliki hak sepatu.
- Setiap satu produk memiliki beberapa pilihan jenis bahan atau tidak memiliki pilihan bahan.

Berdasarkan identifikasi yang telah dilakukan, maka ERD selanjutnya dapat dibuat dengan jenis relasinya pada gambar 5.1 berikut ini:

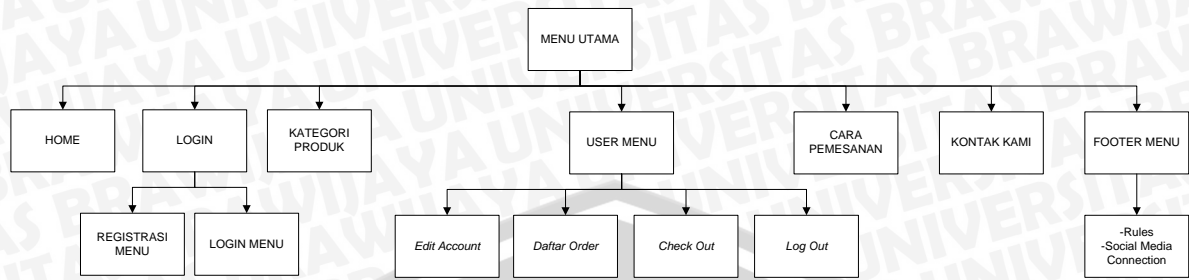


Gambar 5.1 Relasi ERD

5.2 Desain Antar Muka (*User Interface Design*)

Tampilan visual merupakan hal yang penting dalam interaksi pengguna dengan sistem. Sistem dalam tampilan visual harus memenuhi beberapa ketentuan yang telah dianalisa sebelumnya agar mudah mengerti dan digunakan oleh admin dan pengguna. Perancangan *user interface* akan ditunjukkan pada gambar berikut ini.

5.2.1 Bagan Hierarki Menu *User Interface*

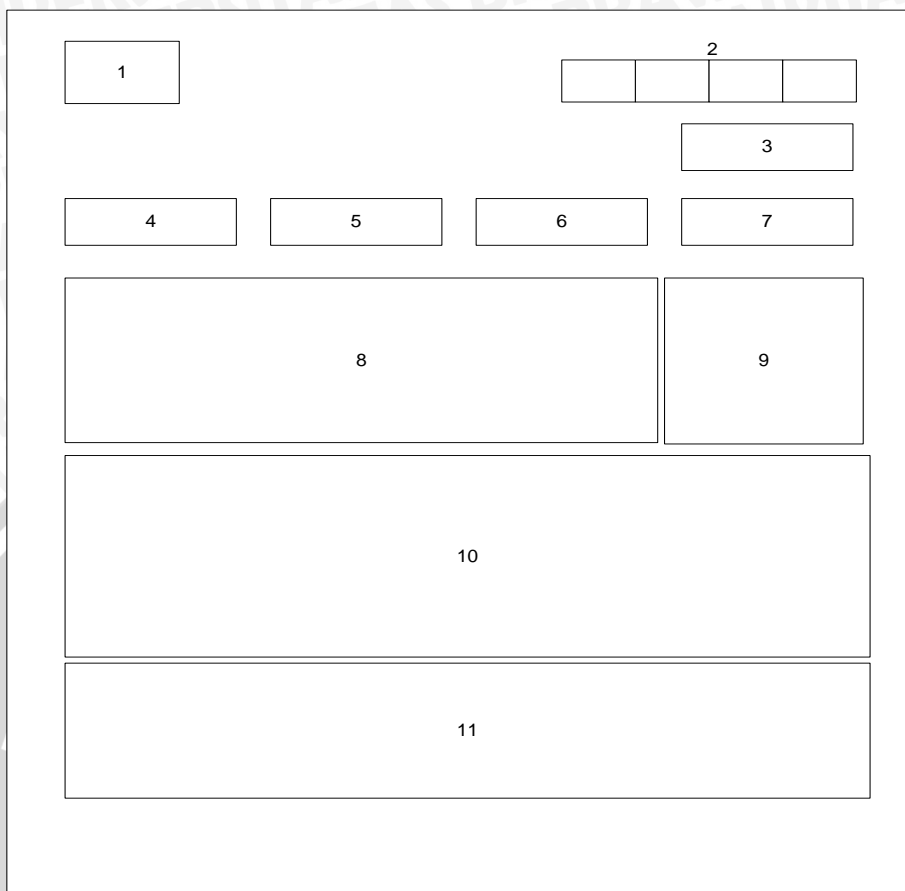


Gambar 5.2 Bagan Hierarki Menu *User Interface*

Dari gambar diatas dapat dilihat bahwa menu utama dari website e-commerce ini terdiri dari enam menu yaitu *Home*, *Login*, *User Menu*, *Kategori*, *Cara Pemesanan*, *Footer Menu* dan *Kontak Kami*. Masing-masing menu terdiri dari beberapa sub-menu. Sub-menu login berisi form registrasi dan form login. *Footer page* berisi peraturan-peraturan yang ada di dalam website dan terkoneksi dengan beberapa *social media* yang ada pada saat ini. *User Menu* tersusun atas sub menu *edit account*, *daftar order*, *check out* dan *log out*.

5.2.2 Desain *Interface*

Desain *interface* adalah perancangan tampilan *website* yang nantinya digunakan oleh *user* (dalam hal ini *customer*). Desain *interface* ini harus diatur sedemikian rupa sehingga pengguna (*user*) tidak kesulitan ketika menggunakan *website e-commerce* ini nantinya. Desain *interface website e-commerce* Mr.Pienk bisa dilihat pada Gambar 5.4 di bawah ini.



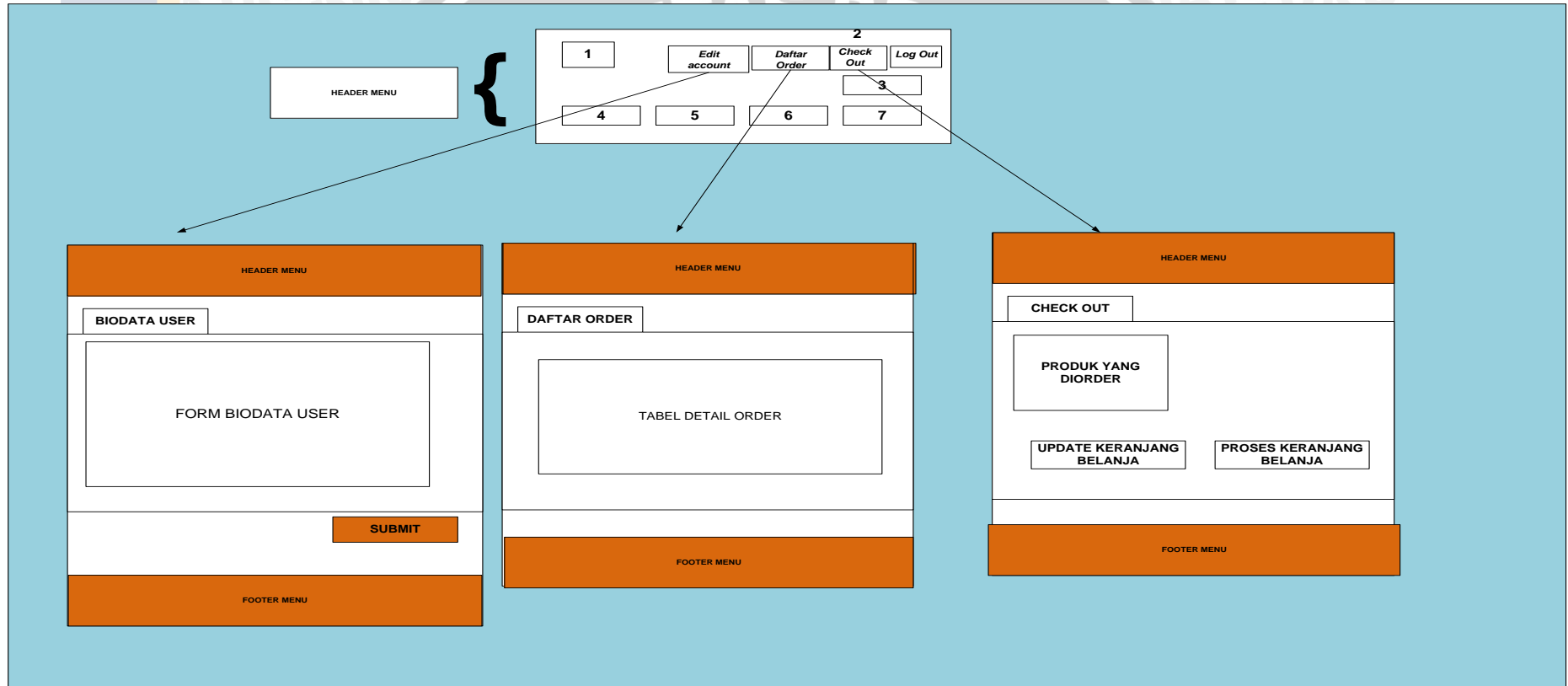
Gambar 5.3 Desain *Interface*

Keterangan:

- | | |
|--|-------------------------|
| 1. Logo | 7. Kontak Kami |
| 2. <i>User Menu (My Account/ Edit Account ,My Order/ Daftar Order, Check out, Log out)</i> | 8. <i>Banner</i> |
| 3. Keranjang Belanja / <i>Cart</i> | 9. <i>User Guide</i> |
| 4. Home | 10. <i>Product Page</i> |
| 5. Kategori | 11. <i>Footer Menu</i> |
| 6. Cara Pemesanan | |

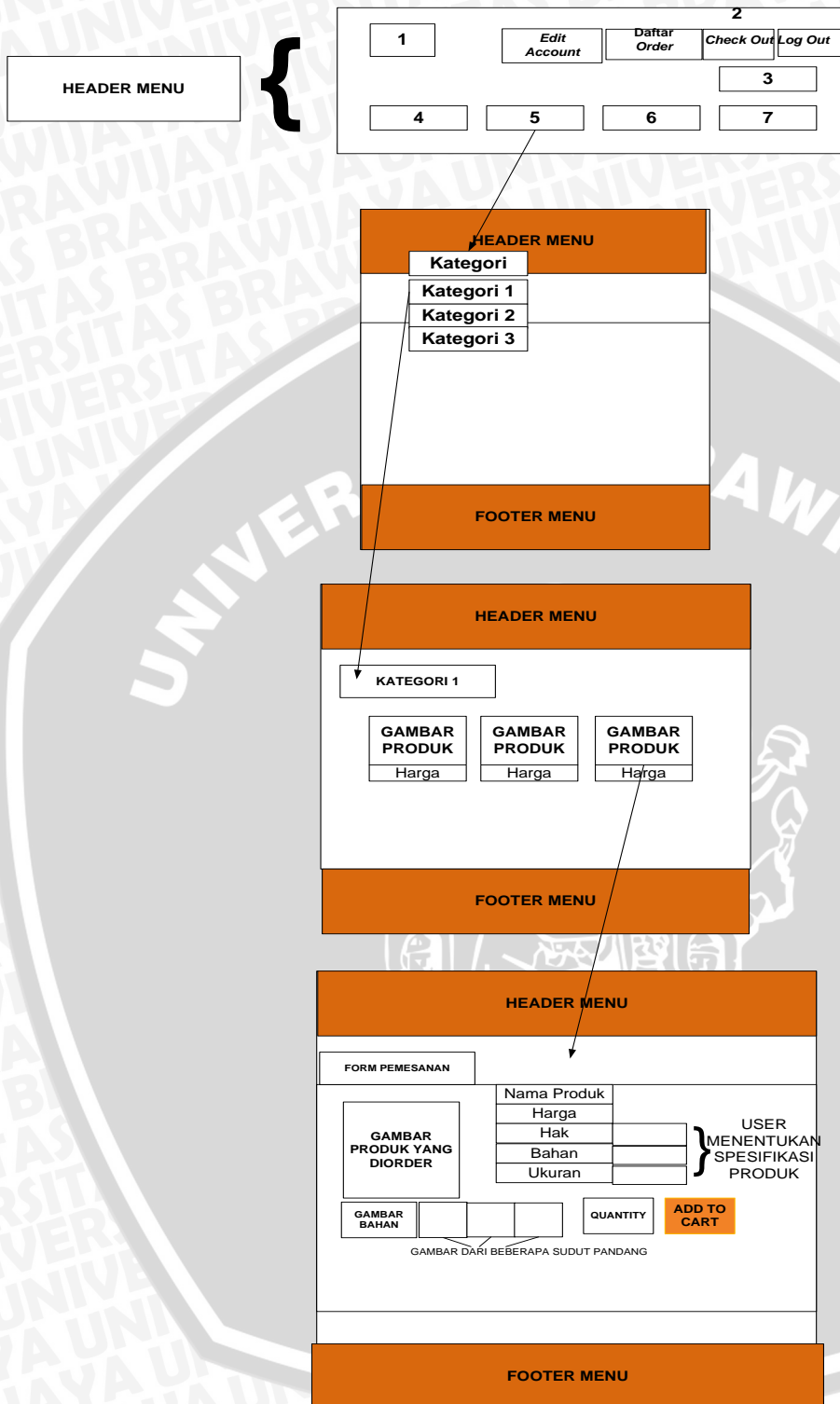


5.2.2.1 Story Board User menu



Gambar 5.4 Story Board User menu

5.2.2.2 Story Board Form Pemesanan Produk



Gambar 5.5 Story Board Form Pemesanan Produk



5.3 Desain Algoritma

Algoritma adalah otak dari software, yang melukiskan langkah demi langkah yang harus dilakukan untuk mencapai tujuan. Algoritma pada umumnya berupa flowchart atau pseudocode. Berikut ini algoritma dari bagian terpenting dari aplikasi dalam bentuk pseudocode.

5.3.1 Desain Algoritma Admin

1. Proses Input Produk oleh Admin

Berikut ini adalah algoritma langkah proses input produk pada sistem:

Start

Masuk Halaman Utama Website

Klik menu Log in

Masukkan *username* dan *password* admin

Klik Log in

Log in admin sukses

Tampilkan Halaman Admin

Klik Input Product

Pilih Kategori Produk

Isi Kolom Atribut Produk (Nama Produk, Harga, Keterangan)

Input Hak Sepatu

Jika ada hak sepatu pilih "ada"

Jika tidak ada hak sepatu pilih "tidak ada"

Pilih ukuran hak sepatu yang akan ditampilkan pada list "ukuran hak"

Input Ukuran Sepatu

Pilih ukuran sepatu yang akan ditampilkan pada list "ukuran"

Input Jenis bahan Sepatu

Pilih jenis bahan sepatu yang akan ditampilkan pada list "jenis bahan"

Pilih gambar yang akan ditampilkan

Klik submit

Data Produk disimpan pada *database* produk

Data produk ditampilkan pada halaman produk

Selesai

Dari algoritma diatas dapat dijelaskan bahwa untuk input admin yang harus dilakukan terlebih dahulu adalah masuk ke halaman utama website. Kemudian admin log in untuk dapat menampilkan halaman admin. Langkah berikutnya adalah memilih input produk, dalam menginput produk yang harus dilakukan adalah memilih kategori produk, mengisi kolom atribut produk, kemudian memilih ada hak sepatu atau tidak ada hak sepatu dan memilih gambar produk yang akan ditampilkan. Setelah semua atribut diinput maka admin dapat melakukan submit untuk menyimpan data produk dan ditampilkan pada halaman produk.

Proses Verifikasi *OrderUser* oleh Admin

Berikut ini adalah algoritma langkah proses cek *order* user pada sistem:

Start

Masuk halaman Utama Admin

Klik menu Daftar *Order*

Tampilkan tabel Daftar *Order*

Jika *user* sudah membayar dan *order* diproses klik “proses”

Jika *user* belum membayar klik “hapus *order*”

Jika *order* sudah diproses dan telah dikirim klik “kirim”

Jika tidak *order* belum dapat diproses klik “belum proses”

Data akan tersimpan pada *database* pesanan

Tampilkan detail pesanan pada halaman *my orderuser*

Selesai

Logout

Dari algoritma diatas dapat dijelaskan bahwa untuk memproses *orderuser* oleh admin yang dilakukan terlebih dahulu adalah masuk ke halaman utama admin. Kemudian klik menu daftar *order*, langkah berikutnya adalah klik proses jika *user* sudah membayar, klik hapus *order* jika *user* belum membayar, klik kirim jika *order* sudah diproses semua dan telah dikirim, jika *order* belum dapat diproses maka klik belum proses. Setelah semua langkah dilakukan data verifikasi akan disimpan pada *database* pesanan dan ditampilkan pada tabel *my order* milik *user*.

5.3.2 Desain Algoritma *User*

1. Proses registrasi dan *log in*

Berikut ini adalah algoritma langkah proses registrasi dan *log inuser* pada sistem:

Start

Masuk halaman utama website

Klik menu Log in

Klik Create account

Tampilkan Form Registrasi *User*

Isi Kolom Form registrasi (Nama, Alamat, No Telephon, *email*, *password*, *confirm password*)

Klik Create

Data akan tersimpan pada *databaseuser*

Klik menu log in

Isi kolom (*email*, *password*)

Klik Log in

Log in sukses

Tampilkan halaman *user*

Dari algoritma diatas dapat dijelaskan bahwa untuk login dan registrasi *user* yang dilakukan terlebih dahulu adalah masuk ke halaman utama website. Kemudian

klik menu log in, langkah berikutnya adalah klik menu create account, isi semua kolom pada form *user* (Nama,Alamat,No Telephon,*email,password,confirm password*) dan klik create maka data akan tersimpan pada *database user*. Untuk menampilkan halaman *user* klik menu login, isikan *email* dan *password* yang telah diregistrasi.

2. Proses Order Produk

Berikut ini adalah algoritma langkah proses *order* Produkuser pada sistem:

Start

Masuk Halaman Utama Website

Klik menu kategori

Pilih Kategori Produk

Tampilkan Hatalaman Kategori Yang dipilih

Pilih Produk yang diinginkan

Tampilkan halaman pemesanan

Jika ada hak sepatu

Isi ukuran hak sepatu

Jika ada pilihan bahan

Pilih bahan yang diinginkan

Tuliskan ukuran sepatu yang diinginkan

Tuliskan jumlah barang yang ingin dibeli pada kolom quantity

Klik add to cart

Klik menu check out

Jika ingin memproses *order* klik proses *order*

Jika tidak klik update keranjang belanja

Detail pesanan disimpan pada *database* detail pesanan

Tampilkan tabel detail pesanan pada halaman *myorder*

Selesai

Dari algoritma diatas dapat dijelaskan bahwa untuk login dan registrasi *user* yang dilakukan terlebih dahulu adalah masuk ke halaman utama website. Kemudian klik menu kategori untuk menampilkan halaman kategori yang dipilih, langkah berikutnya adalah pilih produk untuk menampilkan halaman pemesanan. Jika sepatu yang dipilih memiliki hak maka isi ukuran hak yang diinginkan. Jika produk dapat dipilih bahannya,pilih bahan yang diinginkan, kemudian isikan ukuran sepatu yang diinginkan. Pada kolom quantity isikan jumlah produk yang diinginkan. Setelah semua spesifikasi diisikan klik menu *addto cart*, selanjutnya klik menu *checkout* untuk memproses *order*, jika ingin memilih produk lain klik *update* keranjang belanja. Klik proses *order* agar *order* dapat disimpan pada *database* detail pesanan. Untuk melihat status *order* dapat dilihat pada tabel *my order*.

3. Proses Konfirmasi *Order*

Berikut ini adalah algoritma langkah konfirmasi *Order* Produk oleh *user* pada sistem:

Start

Klik menu *myorder*

Tampilkan tabel detail pesanan

 Jika sudah membayar Klik “Konfirm”

 Jika ingin menghapus *order* klik “Hapus”

Tampilkan status *order* yang telah diverifikasi admin

Selesai

Log out

Dari algoritma diatas dapat dijelaskan bahwa untuk konfirmasi *order* yang dilakukan terlebih dahulu adalah masuk ke halaman *myorder* untuk melihat tabel *my order*. Untuk proses konfirmasi, jika sudah membayar klik konfirm, jika ingin menghapus *order* klik hapus. Status *order* yang telah diverifikasi oleh admin akan ditampilkan pada tabel *my order*.

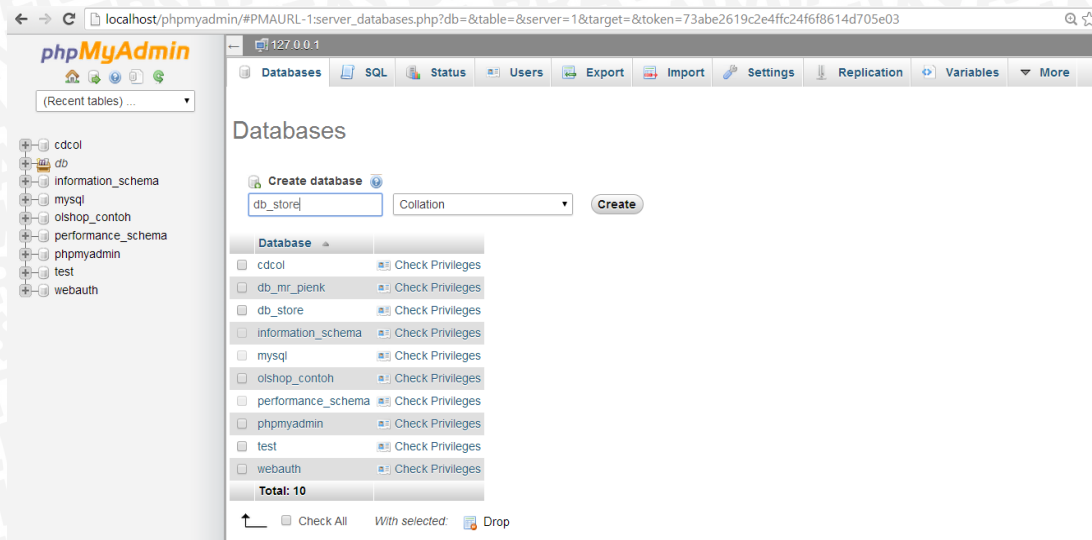
5.4 Implementasi

Pada tahapan ini semua perancangan yang telah dilakukan akan diaplikasikan ke dalam software. Implementasi bertujuan agar admin/*user* dapat menggunakan program yang telah dianalisa dan dirancang sebelumnya. Pada tahap ini terdapat dua bagian, yaitu implementasi *database* dan implementasi *user interface*.

5.4.1 Implementasi *Database*

Pembuatan *database* dilakukan sebelum penginstalan program. Pembuatan *database* ini diperlukan untuk memasukkan, menghapus, mengubah, manipulasi, dan memperoleh data atau informasi seluruh *content* yang ada di dalam *website* nantinya. Pembuatan *database* pada website Mr.Pienk dengan menggunakan MySQL dapat dilakukan dengan cara sebagai berikut:

1. Buka jendela *browser* (dalam hal ini penulis menggunakan *Google Chrome* Version 28.0.1500.95 m)
2. Ketikkan *http://localhost/phpmyadmin/* pada *address bar*.
3. Beri nama *database* store.
4. Kemudian klik *create*.
5. *Database* sudah terbentuk secara otomatis dengan tabel sekaligus jumlah *field* yang sudah dikonfigurasi oleh *phpmyadmin*.



Gambar 5.6 Tampilan PHPMyAdmin pada Localhost

5.4.1.1 Pembuatan Database

Setelah entitas tabel yang telah ditentukan dibuat langkah selanjutnya adalah membuat atribut dari tabel tersebut dan akan ditunjukkan sebagai berikut:

1. Tabel User/Customer

Berikut ini adalah implementasi tabel *database* untuk entitas *User/Customer*:

+ Options										
← T →										
	id_user	nm_user	pass	email	alamat	negara	prov	kota	pos	phone
<input type="checkbox"/>	1	Wulan	pass	wulan@gmail.com	Jalan Veteran	Indonesia	Jawa Timur	Surabaya	62115	085733128787
<input type="checkbox"/>	2	sar	123123	xartika.kd@gmail.com	sawojajar	Indonesia	Jawa Timur	Malang	65139	083848xxxx
<input type="checkbox"/>	3	Ayu Yunita	pass	ayu_yunita@gmail.com	Griya Shanta	Indonesia	Jawa Timur	Malang	62115	081234XXXX
<input type="checkbox"/>	5	nindyta	NIND321	nindyatratifania@gmail.com	watu gong 17A	Indonesia	Jawa Timur	Malang	68145	085749973262

Gambar 5.7 Prenscren Tabel Database User pada PHP my Admin

2. Tabel Detail Pesanan

Berikut ini adalah implementasi tabel *database* untuk entitas Detail Pesanan:

+ Options										
← T →										
	id_det	id_pesanan	id_produk	bahan_pesanan	ukuran_pesanan	jml	harga_pesanan	hak	jmlp	jmlj
<input type="checkbox"/>	29	34	17	Chocolate7209c	36	12	150000	0	12	12
<input type="checkbox"/>	34	37	17	Chocolate7209c	36	1	150000	0	1	1
<input type="checkbox"/>	35	38	18	Butter901	23	1	150000	0	1	1
<input type="checkbox"/>	33	36	26	Chocolate7209c	36	2	135000	0	2	2
<input type="checkbox"/>	36	39	17	Chocolate7209c	12	1	150000	0	11	11

Gambar 5.8 Prenscren Tabel Detail Pesanan pada PHP my Admin

3. Tabel Kategori Produk

Berikut ini adalah implementasi tabel *database* untuk Kategori Produk:

	id_produk	id_kategori
<input type="checkbox"/> Edit Copy Delete	6	5
<input type="checkbox"/> Edit Copy Delete	8	5
<input type="checkbox"/> Edit Copy Delete	9	5
<input type="checkbox"/> Edit Copy Delete	10	1
<input type="checkbox"/> Edit Copy Delete	11	1

Gambar 5.9 Printscreen Tabel Kategori Produk pada PHP my Admin

5.4.2 Implementasi User Interface

Implementasi *user interface* menggunakan PHP dan MySQL ditujukan supaya pengguna/*user* lebih mudah untuk mengakses *website e-commerce*. Implementasi ini didasarkan atas analisa dan desain yang telah dibuat pada tahapan sebelumnya. Implementasi *user interface website e-commerce* Mr.Pienk adalah sebagai berikut:

1. Implementasi User Interface User/Customer

Implementasi *user interface user /costumer* yang ingin membeli produk.

- a. Pertama *user/customer* harus melakukan pendaftaran terlebih dahulu dengan mengklik *log in*.



Gambar 5.10 Print screen menu log in user

- b. Kemudian klik create account

Login or Create an Account

New Customers

By creating an account with our store, you will be able to move through the checkout process faster, store multiple shipping addresses, view and track your orders in your account and more.

* Required Fields

Registered Customers

If you have an account with us, please log in.

Email Address*

Password*

* Required Fields

Gambar 5.11 Print screen halaman log in user

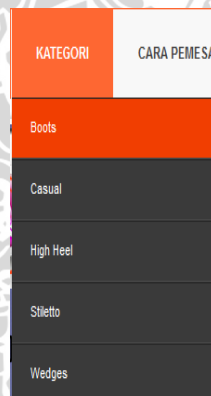
c. Data *Useryang* diisi pada form registrasi harus valid.

The screenshot shows a web registration form with the following sections:

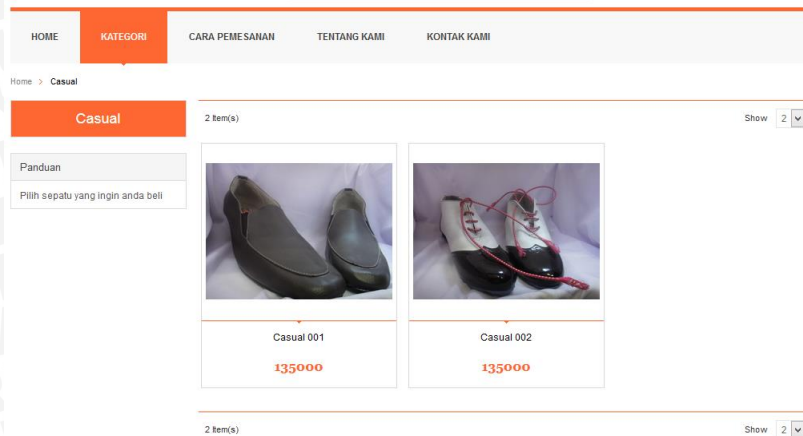
- Navigation:** HOME, KATEGORI, CARA PEMESANAN, TENTANG KAMI, KONTAK KAMI
- Create an Account:**
 - Personal Information:**
 - ID User: 00002
 - Nama Lengkap (Username)*: [input field]
 - Alamat E-mail*: [input field]
 - Alamat: [input field]
 - Kota: Jakarta (dropdown menu)
 - Provinsi: [input field]
 - Kode Pos: [input field]
 - No. handphone / Telepon: [input field]
 - Login Information:**
 - Password*: [input field]
 - Confirm Password*: [input field]
- Buttons:** Back (with arrow), Submit
- Text:** * Required Fields

Gambar 5.12 Print screen halaman form registrasi user

d. Selanjutnya *customer* melakukan *login*, lalu *customer* memilih produk pada halaman *home* atau memilih produk berdasarkan kategori.

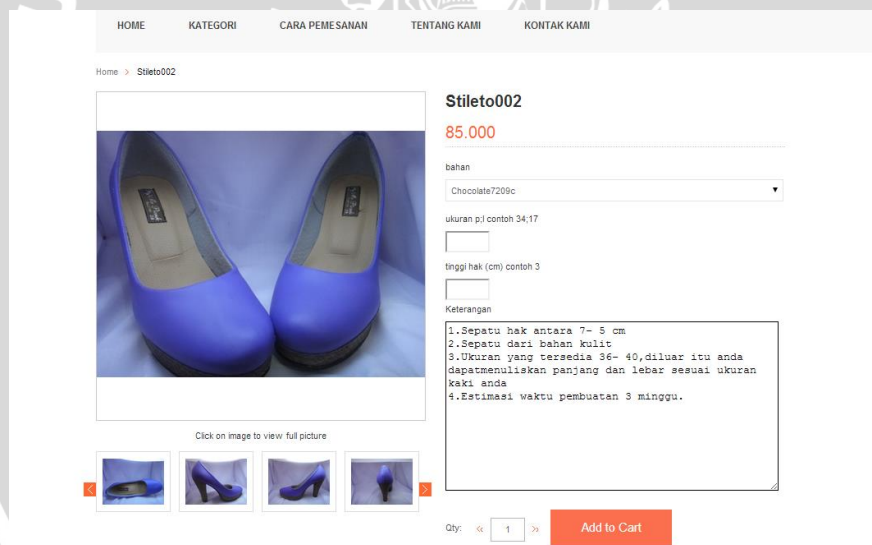


Gambar 5.13a Print screen list menu kategori produk



Gambar 5.13b Print screen halaman kategori produk

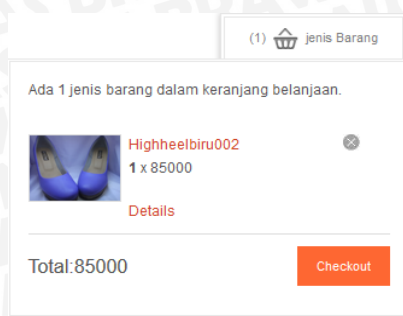
- e. Jika ada produk yang ingin dibeli, *user/customer* bisa mengklik pada produk. Kemudian memilih spesifikasi produk yang diinginkan.



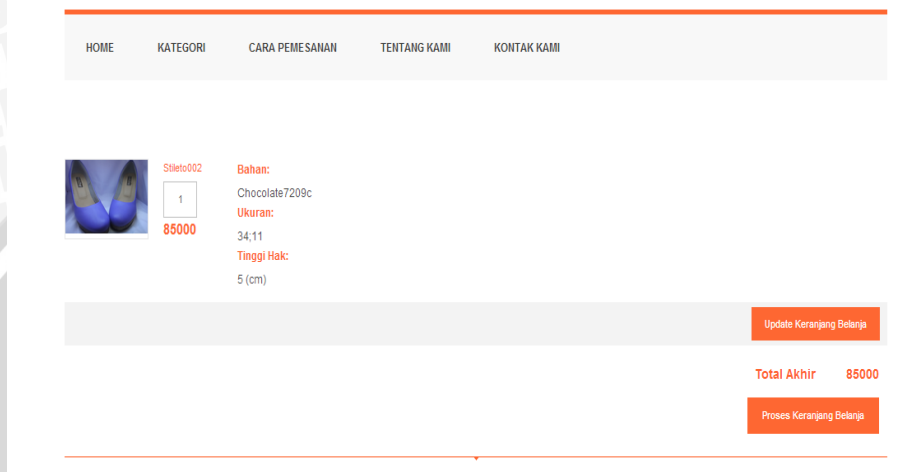
Gambar 5.14 Print screen halaman form detail pemesanan

Spesifikasi yang dapat diisikan adalah bahan, ukuran dan tinggi hak jika sepatu memiliki hak.

- f. Klik menu check out pada keranjang belanja setelah memesan, *user* dapat kembali memilih produk lagi jika ingin membeli produk lain, jika tidak maka *user* dapat melakukan “proses keranjang belanja” untuk memproses *order*.



Gambar 5.15 a *Print screen* menu check out pada keranjang belanja

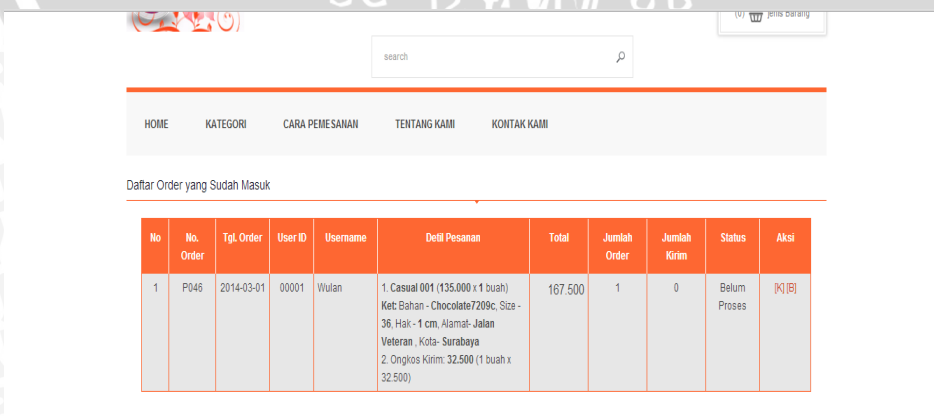


Gambar 5.15b *Print screen* halaman check out user

g. Setelah membayar sejumlah *order* yang sudah ditentukan *user* dapat melakukan konfirmasi pada menu *myorder* dengan mengklik huruf K pada kolom Aksi.



Gambar 5.16 a *Print screen* menu *myorder* user



Gambar 5.16 b *Print screen* halaman *myorder* user

d. *User* dapat mengganti alamat pengiriman pada halaman *myaccount*.



2. Implementasi *User Interface Admin*

- a. Admin melakukan login pada halaman login, pada kolom *email address* admin menuliskan *username* admin, berbeda dengan *user* yang menuliskan *email* dan *password*.

Gambar 5.17 *Print screen* halaman log in admin

- b. Admin melakukan input produk

Admin melakukan proses input produk dengan klik menu input produk pada menu admin.

Input Product

Gambar 5.18 a *Print screen* menu input produk

Gambar 5.18 b *Print screen* halaman input produk admin

Pada kolom hak admin dapat memilih pilihan ada jika sepatu memiliki hak dan tidak ada jika sepatu tidak memiliki hak.

- c. Admin melakukan verifikasi *orderuser* pada menu.

Untuk melakukan proses verifikasi *user* admin memilih menu daftar *order* pada menu admin.

Daftar Order

Gambar 5.19 aPrint screenmenu daftar *order*

No	No. Order	Tgl. Order	User ID	Username	Detil Pesanan	Total	Jumlah Order	Jumlah Kirim	Status	Aksi
1	P045	2014-02-17	00015	agfrizal	1. Wedges merah 002 (135.000 x 20 buah) Ket: Bahan - Butter901, Size - 34, Hak - 0 cm, Alamat- JL. Dirgantara C3 , Kota- Malang 2. Stileto003 (135.000 x 40 buah) Ket: Bahan - Chocolate7209c, Size - 36, Hak - 1 cm, Alamat- JL. Dirgantara C3 , Kota- Malang 3. Ongkos Kirim: 300.000 (60 buah x 5.000)	8.400.000	60	60	Kirim	[B] [P] [K] [H]
2	P044	2014-02-17	00015	agfrizal	1. BootsTali001 (150.000 x 1 buah) Ket: Bahan - Butter901, Size - 34, Hak - 0 cm, Alamat- JL. Dirgantara C3 , Kota- Malang 2. Casual 001 (135.000 x 10 buah) Ket: Bahan - Chocolate7209c, Size - 35, Hak - 1 cm, Alamat- JL. Dirgantara C3 , Kota- Malang 3. Ongkos Kirim: 55.000 (11 buah x 5.000)	9.955.000	11	11	Kirim	[B] [P] [K] [H]

Gambar 5.19 bPrint screen verifikasi *order* user oleh admin

Admin dapat melakukan verifikasi dengan mengklik pilihan aksi yaitu; jika *user* belum membayar maka klik H (Hapus pesanan), jika *user* sudah membayar tapi *order* belum bisa diproses klik B (Belum), jika *user* sudah membayar dan *order* diproses klik P (Proses) dan jika orderan telah selesai diproduksi dan dikirim klik K (Kirim).

5.4.3 Pembuatan Kode Program (Source Code)

Berikut contoh beberapa penggalan source code sesuai dengan algoritma proses yang telah disusun sebelumnya menggunakan notepad ++:

1. Admin

Untuk Kode Program Admin terdiri dari proses input produk dan proses verifikasi *order*:

a. Proses Input Produk

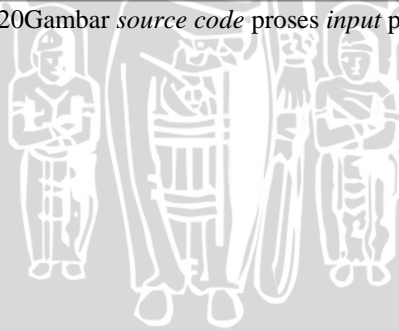
Berikut ini adalah source code PHP untuk halaman proses input produk:

```

<form method='post' action='<?php echo $_SERVER['PHP_SELF']; ?>' name='fproduk' enctype="multipart/form-
data">
<input type="hidden" name="page" value="<?php echo $_SESSION["page"]; ?>">
<div class="main">
<h1 class="page-title">Input Produk</h1>
<div class="fieldset">
<h2 class="legend">Kategori <select title="Kategori Produk" id="kat" name="kat">
<?php
$sql = mysql_query("SELECT * FROM kategori ORDER BY nm_kategori") or die(mysql_error());
while ($row = mysql_fetch_array($sql))
{
echo "<option value='".$row[0]."'>".$row[1]."</option>";
}
?>
</select><span style="float: right"><input type="checkbox" name="feat" value=""
id='feat'> -</span></h2>
<ul class="form-list">
<li class="fields">
<div class="customer-name">
<div class="input-box name-firstname">
<label for="nama">Nama Produk<em>*</em></label>
<input type="text" class="required-entry input-text" title="Nama Lengkap" value=""
name="nama" id="nama">
</div>
</div>
<div class="input-box">
<label for="harga">Harga</label>
<input type="text" title="Harga Produk" value="" id="harga"
name="harga">

```

Gambar 5.20 Gambar source code proses input produk



b. Proses verifikasi *orderuser* oleh admin

Berikut ini adalah source code PHP untuk halaman proses verifikasi *order*:

```
$nmFD = "fBelum".$i; $nmF1 = "fProses".$i; $nmF2 = "fKirim".$i; $nmF3 = "fHapus".$i;
echo "
<form name='".$nmFD.'" action="$_SERVER['PHP_SELF'].'" method='post'>
<input type='hidden' name='page' value='7'><input type='hidden'
name='id_pesan' value='".$row['id_pesan']."'><input type='hidden' name='aksi' value='belum'></form>
<form name='".$nmF1.'" action="$_SERVER['PHP_SELF'].'" method='post'>
<input type='hidden' name='page' value='7'><input type='hidden'
name='id_pesan' value='".$row['id_pesan']."'><input type='hidden' name='aksi' value='proses'></form>
<form name='".$nmF2.'" action="$_SERVER['PHP_SELF'].'" method='post'>
<input type='hidden' name='page' value='7'><input type='hidden'
name='id_pesan' value='".$row['id_pesan']."'><input type='hidden'
name='aksi' value='kirim'></form>
<form name='".$nmF3.'" action="$_SERVER['PHP_SELF'].'" method='post'>
<input type='hidden' name='page' value='7'><input type='hidden'
name='id_pesan' value='".$row['id_pesan']."'><input type='hidden' name='aksi' value='hapus'></form>
<a href='#' title='Belum Proses (default)' onclick=\"document.forms['".$nmFD."'].submit();return false;\">[B]</a>
<a href='#' title='Proses' onclick=\"document.forms['".$nmF1."'].submit();return false;\">[P]</a>
<a href='#' title='Kirim' onclick=\"document.forms['".$nmF2."'].submit();return false;\">[K]</a>
<a href='#' title='Hapus' onclick=\"document.forms['".$nmF3."'].submit();return false;\">[H]</a></td>;
```

Gambar 5.21 Gambar *source code* proses verifikasi *orderuser* oleh admin

2. User/Customer

a. Proses Pemilihan Spesifikasi Produk

Berikut ini adalah source code PHP untuk halaman pemilihan spesifikasi produk oleh *user*:




```

<div class="label-row">
<label>tinggi hak (cm) contoh 3 </label>
</div>
<div class="input-row">
<input type="text" id="hak" name="hak" maxlength="3" style="height: 20px; width: 50px">
</div>
<div class="label-row">
<label>Keterangan</label>
</div>
<div class="input-row">
<textarea id="ket" style="height: 200px; width: 400px" readonly<?php echo $rD['ket']; ?></textarea>
</div>
<?php ?>
<div class="add-to-cart-box">
<span class="qty-box">
<label for="qty">Qty:</label>
<a href="javascript:void(0)" title="" class="prev add"></a>
<input type="hidden" name="idProd" value="<?php echo $rD['id_produk']; ?>">
<input type="hidden" name="nmProd" value="<?php echo $rD['nm_produk']; ?>">
<input type="hidden" name="hrgProd" value="<?php echo $rD['harga']; ?>">
<input type="hidden" name="imgProd" value="<?php echo $rD['img!']; ?>">
<input type="text" value="1" maxlength="12" id="qty" class="input-text qty" name="qty">

```

Gambar 5.22 Gambar *source code* proses pemilihan spesifikasi produk

b. Proses Konfirmasi *order* oleh *user*

Berikut ini adalah *source code* PHP untuk halaman konfirmasi *order* oleh *user*:

```

$nmF1 = "fBayar".$; $nmF2 = "fBatal".$;
echo "
<form name="."$nmF1." Action="."$_SERVER['PHP_SELF']." Method='post'>
<input type='hidden' name='page' value='7'><input type='hidden' name='id_pesan' value="."$row['id_pesan']."><input
type='hidden'
name='aksi' value='bayar'></form>
<form name="."$nmF2." Action="."$_SERVER['PHP_SELF']." Method='post'>
<input type='hidden' name='page' value='7'><input type='hidden' name='id_pesan' value="."$row['id_pesan']."><input
type='hidden' name='aksi' value='hapus'></form>
<a href='#' title='Konfirm' onclick="\ document.forms["."$nmF1."].submit();return false;">[K]</a>
<a href='#' title='Batal' onclick="\ document.forms["."$nmF2."].submit();return false;">[B]</a></td>";

```

Gambar 5.23 Gambar *source code* proses pemilihan spesifikasi produk

5.5 Pengujian (*Testing*)

Tahapan terakhir setelah sistem sudah menjadi *prototype* adalah *testing* (pengujian). Langkah pengujian ini penting untuk melihat apakah *prototype* yang telah dibuat sudah sesuai dengan harapan atau tidak. Tahap pengujian ini ditinjau dari tiga segi, yaitu uji verifikasi, uji validasi dan uji *prototype* yang masing-masing terdapat tujuan yang saling terhubung.

5.5.1 Uji Verifikasi

Verifikasi adalah proses pengecekan atau pengujian yang dilakukan oleh perancang sistem sebelum nantinya sistem akan digunakan oleh *user*. terhadap perangkat lunak apakah sudah memenuhi spesifikasi yang ditetapkan dalam proses perancangan. Untuk pengujian verifikasi sistem e-commerce ini dibedakan menjadi 2 yaitu:

5.5.1.1 Uji Verifikasi Statis

Uji verifikasi statis yaitu berhubungan dengan analisis representasi sistematis untuk menemukan masalah, biasa disebut *Software Inspection*. Uji verifikasi statis dilakukan dengan menguji proses – proses yang ada pada sistem tidak mengalami kesalahan atau error. Dari hasil uji verifikasi statis semua proses telah sesuai dengan alur yang telah dirancang sebelumnya.

5.5.1.2 Uji verifikasi Dinamis

Uji verifikasi dinamis yaitu berhubungan dengan dengan pelaksanaan dan memperhatikan perilaku produk, biasa disebut *Software testing*. Uji verifikasi dinamis dilakukan dengan eksekusi data uji verifikasi statis sebelumnya, untuk menemukan kesalahan dalam kode. Dari uji verifikasi dinamis yang telah dilakukan sebelumnya tidak ditemukan kesalahan coding pada sistem dan sistem dapat berjalan sesuai proses pada uji statis.

5.5.2 Validasi sistem

Uji validasi dilakukan untuk mengetahui apakah sistem sudah memenuhi kebutuhan dengan beberapa tahapan sebagai berikut:

1. *Unit Testing*

Unit Testing adalah sebuah proses untuk menguji sebuah bagian atau komponen tertentu dalam kode program untuk menentukan apakah komponen

tersebut berfungsi dengan benar. Komponen program yang diuji pada tingkat unit testing adalah subprogram, subroutines atau prosedur. Berikut adalah beberapa hasil unit testing:

Tabel 5.3 *Unit Testing* Komponen Cart

NO	Testing Procedure	Expected Result	Actual result
1.	Menu "checkout" diklik.	Menu "checkout" akan menuju proses keranjang belanja	Berhasil, sistem akan masuk ke proses keranjang belanja
2.	Menu "proses keanjang belanja" diklik	Menu "proses keranjang belanja" memproses <i>order</i> yang ditampilkan pada tabel <i>order</i> / tabel <i>my order</i>	Berhasil, sistem memproses <i>order</i> dan ditampilkan pada tabel <i>order</i> / <i>my order</i> .
3.	Menu "update keranjang belanja"	Menu "proses keranjang belanja" mengembalikan sistem ke halaman utama	Berhasil, sistem kembali ke halaman utama
4.	Menu "My Order"	Menu "my order" tabel data order.	Berhasil, sistem menampilkan tabel <i>my order</i> .
5.	Menu "K" diklik.	Menu "K" akan merubah status <i>order</i> menjadi Proses/ Dikirim/ Tidak dapat Diproses	Berhasil, sistem akan merubah status <i>order</i> menjadi Proses/ Dikirim/ Tidak dapat Diproses
6.	Menu "H" diklik	Menu "H" akan menghapus order	Berhasil, sistem akan menghapus <i>order</i>

Tabel 5.4 *Unit testing* Komponen Registrasi User

NO	Testing Procedure	Expected Result	Actual result
1.	Menu "registrasi" diklik.	Menu "registrasi" akan menampilkan form registrasi user	Berhasil, sistem menampilkan form registrasi user
2.	Menu "submit" diklik	Menu "submit" menyimpan data user dan dilihat pada menu <i>My account</i>	Berhasil, data <i>user</i> disimpan dan ditampilkan pada <i>my account</i>
3.	Menu "My account"	Menu "my account" menampilkan data <i>user</i> .	Berhasil, sistem menampilkan data <i>user</i> .

Tabel 5.5 *Unit testing* Komponen Log in User

NO	Testing Procedure	Expected Result	Actual result
1.	Menu "log in" diklik.	Menu "log in" akan masuk kedalam sistem sebagai <i>user</i>	Berhasil, sistem ditampilkan sebagai <i>user</i>

2. Integration Testing

Integration testing merupakan proses pengujian yang dilakukan setelah *unit testing*. Masukan *integration testing* adalah beberapa modul yang sudah melalui proses unit testing, kemudian modul tersebut disatukan dan dilakukan

pengujian berdasarkan apa yang sudah direncanakan. Berikut adalah hasil uji *integration testing*:

Tabel 5.6 *Integration Testing* Proses Order

NO	Testing Procedure	Input Data	Expected Result	Actual result
1.	Menambahkan pesanan.	Mengklik menu update keranjang belanja dan memilih produk dan memproses cart.	Order berhasil ditambahkan	Order yang ditambahkan Sesuai dengan produk yang dipesan.
2.	Menghapus pesanan	Mengklik menu “h” pada tabel my order	Order berhasil dihapus	Order tidak ditampilkan pada tabel my order.

Tabel 5.7 Uji *integration* registrasi dan log in

NO	Testing Procedure	Input Data	Expected Result	Actual result
1.	Update data user	Mengedit biodata kemudian mengklik menu submit pada halaman my account	Data berhasil diubah	Data berhasil diubah dan dapat dilihat pada halaman my account
2.	User log in tanpa melakukan registrasi	mengklik menu log in dan memasukkan data username dan email yang belum diregistrasi	Tidak bisa melakukan order	Berhasil masuk sistem tanpa sebagai user atau hanya bisa melihat produk.

3. System Testing

System Testing adalah pengujian yang dilakukan pada sebuah sistem yang sudah terintegrasi dengan tujuan untuk mengevaluasi kemampuan sistem dalam memenuhi requirement. *System testing* dilakukan dengan metode *black-box testing*. Semua komponen yang telah diuji pada *integration testing* kemudian disatukan menjadi satu kesatuan sistem dan kemudian dilakukan *system testing*.

Tabel 5.8 *System Testing*

Item Pengujian	Deskripsi	Jenis Pengujian
Melakukan Registrasi dan Log in	Mengisi Data pada form: Nama, Alamat, No.Telephone, email, password.	Black Box
Melakukan order produk	Memilih produk, isi form spesifikasi (jenis bahan, ukuran, ukuran hak, jumlah), kemudian klik tombol checkout pada cart, kemudian klik proses keranjang belanja	Black Box
Konfirmasi pembayaran	Klik “K” untuk konfirmasi pembayaran “ H” untuk menghapus order	Black Box
Input Produk	Isi Form input (nama produk, harga, jenis bahan, ukuran hak, ukuran sepatu)	Black Box
Verifikasi Order	Klik “ B” untuk belum proses “ H” untuk hapus order, “ P” untuk proses order , “K” untuk order dikirim	Black Box

Untuk hasil *System Testing* dapat dilihat pada lampiran.1

4. Acceptance Testing

User acceptance testing adalah sebuah proses pengujian yang membandingkan perangkat lunak dengan requirement awal dan kebutuhan *end user*. *User acceptance testing* dilaksanakan oleh customer. Hasil dari pengujian ini akan memberikan tingkat kepercayaan yang tinggi di customer bahwa perangkat lunak memiliki performansi yang baik ketika digunakan. Berikut ini adalah hasil uji dari *acceptance testing*. Acceptance testing dilakukan oleh 10 orang customer dan pengelola atau pemilik toko sepatu Mr.Pienk

Tabel 5.9 *Acceptance testing*

ADMIN		
No	User requirement	Hasil Uji
1.	Sistem menyediakan form untuk admin dapat menginputkan data produk, kategori produk, jenis hak, jenis ukuran jenis dan jenis bahan.	Memenuhi
2.	Sistem dapat menampilkan informasi spesifikasi produk , list kategori produk dan katalog produk	Memenuhi
3.	Admin dapat melakukan input, edit, delete data produk.	Memenuhi
4.	Sistem dapat memungkinkan admin melakukan update pada keseluruhan data produk dalam sistem.	Memenuhi
5.	Sistem memberikan keamanan untuk akses admin dengan <i>username</i> dan <i>password</i> yang hanya dapat diakses oleh admin itu sendiri.	Memenuhi
USER/CUSTOMER		
No	User requirement	Hasil Uji
1.	Adanya form pemesanan produk dan user dapat memilih spesifikasi (jenis bahan, ukuran, ukuran hak).	Memenuhi
2.	Sistem dapat menampilkan data <i>customer</i> .	Memenuhi
3.	Sistem dapat menampilkan data <i>order</i> produk beserta spesifikasinya yang telah dipesan oleh <i>customer</i>	Memenuhi
4.	Setiap <i>customer</i> melakukan registrasi dan <i>login</i> untuk masuk ke halaman utama website.	Memenuhi
5.	<i>Customer</i> dapat melakukan proses pemesanan produk.	Memenuhi
6.	<i>Customer</i> dapat melakukan proses pengisian spesifikasi produk pada form pemesanan.	Memenuhi
7.	Sistem memungkinkan <i>customer</i> mengetahui informasi produk (katalog, kategori produk, spesifikasi produk).	Memenuhi
8.	Sistem memungkinkan <i>customer</i> memilih sendiri spesifikasi produk yang diinginkan.	Memenuhi

Dari tabel *acceptance testing* diatas sistem telah memnuhi *user requirement* yang telah dirancang sebelumnya.

5.5.3 Uji Prototype

Uji *prototype* dilakukan dengan membandingkan *prototype* dengan sistem lama

Tabel 5.10 Uji Prototype

Jenis Uji	Sistem Lama	Sistem Baru
PERFORMANCE	1. Proses pencatatan pesanan masih memerlukan waktu yang cukup lama, karena setelah memilih sepatu yang ada pada etalase, pelanggan kemudian memilih jenis bahan yang ada pada catalog atau pada sepatu yang lain kemudian spesifikasi sepatu dicatat oleh pengelola	1. Proses pemilihan produk, spesifikasi produk, pengelolaan data pemesanan membutuhkan waktu yang cepat karena dilakukan secara otomatis pada sistem.
INFORMATION	Data spesifikasi pemesanan kurang akurat karena tidak dibedakan dengan kode-kode tertentu sehingga data pemesanan untuk proses produksi sulit dikelola karena ketidaksepahaman jenis spesifikasi antara pihak pengelola data dengan pengrajin	Data spesifikasi produk dibedakan dengan kode-kode sehingga kekeliruan produksi dapat dikurangi.
ECONOMY	Biaya operasional yang dibutuhkan untuk membuat desain sepatu yang baru cukup besar dan harus memindah beberapa sepatu yang sudah ada pada etalase toko.	Biaya desain pembuatan sepatu dapat dikurangi karena sistem menampilkan desain baru melalui katalog tanpa harus membuat seri desain sepatu tersebut.
CONTROL	1. Sistem informasi pada toko keamanannya belum begitu diperhatikan karena dapat dilihat oleh pihak manapun yang sedang berada di toko. 2. Ketika sepatu telah diselesaikan dan diserahkan pada pembeli, maka kwitansi pembelian akan dihilangkan dari buku jurnal dan belum ada sistem yang menyimpan data-data lama.	1. Akses sistem disertai username dan password sehingga hanya orang tertentu yang dapat mengakses sistem. 2. Data pemesanan disimpan pada database dan dapat dicari kembali jika dibutuhkan.
EFICIENCY	Untuk pelaporan data informasi pemesanan hanya dilakukan oleh pihak pengelola toko saja jadi ketika pihak pengelola toko tidak masuk maka informasi yang sebelumnya telah direkap oleh pihak pengelola kurang dipahami dengan pengelola yang menggantikan, hal tersebut akan menghambat proses produksi dan pemesanan berikutnya.	Pelaporan data dapat dipahami oleh beberapa pengelola sehingga ketika salah satu pengelola tidak masuk, proses pelaporan tetap dapat dilakukan.
SERVICE	1. Sistem informasi pemesanan tidak bisa diakses 24 jam karena keterbatasan jam kerja toko. 2. Sistem hanya bisa dijangkau oleh warga yang berada di kota Malang saja sedangkan warga luar kota yang ingin memesan harus datang ke toko Mr. Pienk. 3. Pendistribusian informasi pemesanan antara pihak pengelola dengan konsumen kurang baik karena pihak pengelola sering terlambat memberikan informasi jika terjadi penundaan produksi pada konsumen. Pemberitahuan hanya melalui sms saja.	1. Sistem informasi dapat diakses 24 jam karena sistem tersedia secara online. 2. Sistem dapat dijangkau warga luar kota Malang 3. Penyampaian informasi pemesanan dapat dilakukan secara cepat karena status pemesanan disampaikan melalui sistem.

Contents

BAB V	56
PERANCANGAN SISTEM	56
5.1 System Design / Desain Sistem	56
5.1.1 Desain Database	56
5.1.1.1 Logical Model	56
Tabel 5.1 Tabel Entitas dan atribut ERD	57
5.1.1.2 Kardinalitas Relasi	57
Tabel 5.2 Tabel Identifikasi Relasi	57
Gambar 5.1 Relasi ERD.....	58
5.2 Desain Antar Muka (User Interface Design)	58
5.2.1 Bagan Hierarki Menu User Interface	59
Gambar 5.2 Bagan Hierarki Menu <i>User Interface</i>	59
5.2.2 Desain Interface	59
Gambar 5.3 Desain <i>Interface</i>	60
5.2.2.1 Story Board User menu	61
Gambar 5.4 <i>Story Board User menu</i>	61
5.2.2.2 Story Board Form Pemesanan Produk	62
Gambar 5.5 <i>Story Board Form Pemesanan Produk</i>	62
5.3 Desain Algoritma	63
5.3.1 Desain Algoritma Admin	63
5.3.2 Desain Algoritma User	64
5.4 Implementasi	66
5.4.1 Implementasi Database	66
Gambar 5.6 Tampilan <i>PHPMYAdmin</i> pada <i>Localhost</i>	67
5.4.1.1 Pembuatan Database	67
Gambar 5.7 <i>Printscreen</i> Tabel <i>Database User</i> pada <i>PHP my Admin</i>	67
Gambar 5.8 <i>Printscreen</i> Tabel <i>Detail Pesanan</i> pada <i>PHP my Admin</i>	67
Gambar 5.9 <i>Printscreen</i> Tabel <i>Kategori Produk</i> pada <i>PHP my Admin</i>	68
5.4.2 Implementasi User Interface	68
Gambar 5.10 <i>Print screen</i> menu log in <i>user</i>	68
Gambar 5.11 <i>Print screen</i> halaman log in <i>user</i>	68
Gambar 5.12 <i>Print screen</i> halaman form registrasi <i>user</i>	69
Gambar 5.13 a <i>Print screen</i> list menu kategori produk	69

	83
Gambar 5.13 b <i>Print screen</i> halaman kategori produk	70
Gambar 5.14 <i>Print screen</i> halaman form detail pemesanan	70
Gambar 5.15 a <i>Print screen</i> menu check out pada keranjang belanja	71
Gambar 5.15 b <i>Print screen</i> halaman check out user	71
Gambar 5.16 a <i>Print screen</i> menu myorderuser	71
Gambar 5.16 b <i>Print screen</i> halaman myorderuser	71
Gambar 5.17 <i>Print screen</i> halaman log in admin.....	72
Gambar 5.18 a <i>Print screen</i> menu input produk	72
Gambar 5.18 b <i>Print screen</i> halaman input produk admin	72
Gambar 5.19 a <i>Print screen</i> menu daftar order	73
Gambar 5.19 b <i>Print screen</i> verifikasi orderuser oleh admin.....	73
5.4.3 Pembuatan Kode Program (Source Code)	73
Gambar 5.20 Gambar <i>source code</i> proses input produk	74
Gambar 5.21 Gambar <i>source code</i> proses verifikasi orderuser oleh admin	75
Gambar 5.22 Gambar <i>source code</i> proses pemilihan spesifikasi produk.....	76
Gambar 5.23 Gambar <i>source code</i> proses pemilihan spesifikasi produk.....	76
5.5 Pengujian (Testing)	77
5.5.1 Uji Verifikasi	77
5.5.1.1 Uji Verifikasi Statis	77
5.5.1.2 Uji verifikasi Dinamis	77
5.5.2 Validasi sistem	77
Tabel 5.3 <i>Unit Testing</i> Komponen Cart	78
Tabel 5.4 <i>Unit testing</i> Komponen Registrasi User.....	78
Tabel 5.5 <i>Unit testing</i> Komponen Log in User	78
Tabel 5.6 <i>Integration Testing</i> Proses Order.....	79
Tabel 5.7 <i>Uji integration</i> registrasi dan log in	79
Tabel 5.8 <i>System Testing</i>	79
Tabel 5.9 <i>Acceptance testing</i>	80
5.5.3 Uji Prototype	81
Tabel 5.10 Uji Prototype.....	81

BAB V	56
PERANCANGAN SISTEM	56
5.1 System Design / Desain Sistem	56
5.1.1 Desain Database	56
5.1.1.1 Logical Model	56
Tabel 5.1 Tabel Entitas dan atribut ERD	57
5.1.1.2 Kardinalitas Relasi	57
Tabel 5.2 Tabel Identifikasi Relasi	57
5.2 Desain Antar Muka (User Interface Design)	58
5.2.1 Bagan Hierarki Menu User Interface	59
5.2.2 Desain Interface	59
5.2.2.1 Story Board User menu	61
5.2.2.2 Story Board Form Pemesanan Produk	62
5.3 Desain Algoritma	63
5.3.1 Desain Algoritma Admin	63
5.3.2 Desain Algoritma User	64
5.4 Implementasi	66
5.4.1 Implementasi Database	66
5.4.1.1 Pembuatan Database	67
5.4.2 Implementasi User Interface	68
5.4.3 Pembuatan Kode Program (Source Code)	73
5.5 Pengujian (Testing)	77
5.5.1 Uji Verifikasi	77
5.5.1.1 Uji Verifikasi Statis	77
5.5.1.2 Uji verifikasi Dinamis	77
5.5.2 Validasi sistem	77
Tabel 5.3 Unit Testing Komponen Cart	78

Tabel 5.4	Unit testing Komponen Registrasi User	78
Tabel 5.5	Unit testing Komponen Log in User	78
Tabel 5.6	Integration Testing Proses Order	79
Tabel 5.7	Uji integration registrasi dan log in	79
Tabel 5.8	System Testing	79
Tabel 5.9	Acceptance testing	80
5.5.3	Uji Prototype	81
Tabel 5.10	Uji Prototype	81

