BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. Implementasi

Implementasi sistem merupakan kelanjutan dari kegiatan rancangan program dan dapat dipandang sebagai usaha untuk mewujudkan sistem yang telah dirancang serta merupakan tahap meletakkan sistem supaya siap untuk diimplementasikan pada perusahaan.

5.1.1. Batasan Implementasi

Dalam mengimplementasikan perangkat lunak Sistem Informasi Penjualan jam tangan pada Mercury Watch Bandung, terdapat beberapa hal yang menjadi batasan dalam implementasi sistem, yaitu :

1. Basis data yang digunakan dalam implementasi sistem adalah MySQL
2. Sistem informasi penjualan barang ini diimplementasikan khusus untuk studi kasus pada Mercury Watch Bandung.

5.1.2. Implementasi Perangkat Lunak

Untuk mendukung kelancaran sistem informasi pendaftaran pelatihan, diperlukan perangkat lunak untuk menjalankan program yang telah dirancang. Dimana perangkat lunak berperan sebagai pendukung dari sistem. Perangkat lunak pendukung yang digunakan adalah PHP, XAMPP, dan Web Browser. PHP digunakan sebagai perangkat lunak pengembang karena menyediakan fasilitas yang cukup memadai dalam hal perangkat lunak yang dapat diakses oleh web browser. XAMPP digunakan sebagai

perangkat lunak pendukung dalam pembuatan serta pengembangan dari basis data.

5.1.3. Implementasi Perangkat Keras

Perangkat keras dibutuhkan dalam menjalankan sistem informasi penjualan barang berdasarkan kebutuhan minimal sistem yang harus dipenuhi, terdiri dari :

1. Processor Intel Pentium IV atau setara
2. RAM 512 MB
3. VGA Card Onboard 64 MB
4. Harddisk 40 GB

5.1.4. Implementasi Basis Data

Untuk pembuatan serta pengembangan dari basias data pada sistem informasi pendaftaran pelatihan, perangkat lunak yang digunakan adalah XAMPP server. Berikut adalah implementasi basis data menggunakan XAMPP :

1. Tabel admin

```
CREATE TABLE IF NOT EXISTS `admin` (
`id_admin` int(11) NOT NULL,
`username` varchar(15) COLLATE latin1_general_ci NOT NULL,
`password` varchar(50) COLLATE latin1_general_ci NOT NULL,
PRIMARY KEY (`id_admin`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;
```

2. Tabel category

```
CREATE TABLE IF NOT EXISTS `category` (
  `id_kategori` varchar(3) COLLATE latin1_general_ci NOTNULL,
  `nama_kategori` varchar(35) COLLATE latin1_general_ci NOT NULL,
  `foto` varchar(500) COLLATE latin1_general_ci NOT NULL,
  PRIMARY KEY (id_kategori`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci;
```

3. Tabel user

```
CREATE TABLE IF NOT EXISTS `member` (
  `id_user` varchar(35) COLLATE latin1_general_ci NOT NULL,
  `alamat` varchar(100) COLLATE latin1_general_ci NOT NULL,
  `telepon` varchar(12) COLLATE latin1_general_ci NOT NULL,
  `email` varchar(50) COLLATE latin1_general_ci NOT NULL,
  `nama` varchar(10) COLLATE latin1_general_ci NOT NULL,
  `password` varchar(50) COLLATE latin1_general_ci NOT NULL,
  `id_kota` varchar(50) COLLATE latin1_general_ci NOT NULL,
  PRIMARY KEY (`id_user`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_general_ci;
```

4. Tabel kota

```
CREATE TABLE IF NOT EXISTS `kota` (
  `idkota` int(11) NOT NULL,
```

```
  `nama` varchar(25) COLLATE latin1_general_ci NOT NULL,
  `ongkos` double NOT NULL,
  PRIMARY KEY (`id_kota`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;
```

5. Tabel order

```
CREATE TABLE IF NOT EXISTS `order` (
  `id_order` int(11) NOT NULL,
  `temstamp` varchar(6) COLLATE latin1_general_ci NOT NULL,
  `total` double NOT NULL,
  `status` int(11) NOT NULL,
`id_user` int(11) NOT NULL,
`alamat_pengirim` int(11) NOT NULL,
`id_kota` int(11) NOT NULL,
`ongkir` int(11) NOT NULL,
`grad` int(11) NOT NULL,
  PRIMARY KEY (`id_order`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;
```

6. Tabel order detail

```
CREATE TABLE IF NOT EXISTS `order_detail` (
  `id_order` int(11) NOT NULL,
`total` int(11) NOT NULL,
`id_barang` int(11) NOT NULL,
```

```
  `qty` varchar(500) COLLATE latin1_general_ci NOT NULL,
  PRIMARY KEY (`id_order`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;
```

7. Tabel barang

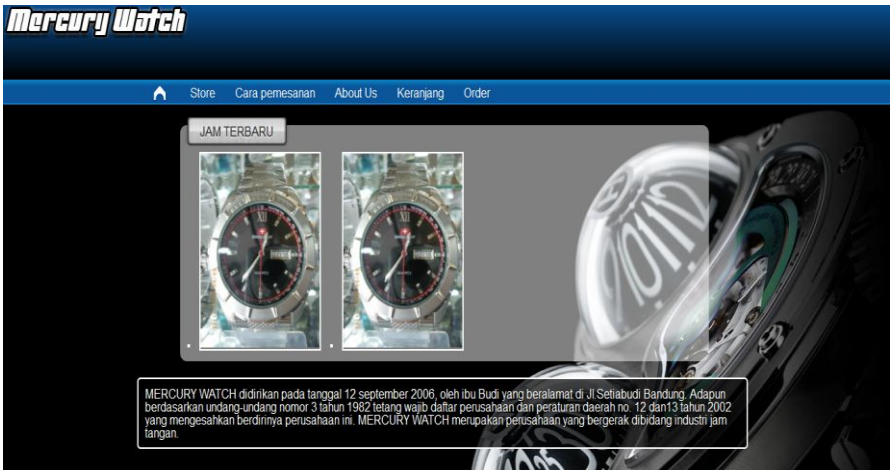
```
CREATE TABLE IF NOT EXISTS `barang` (
  `id_barang` varchar(3) COLLATE latin1_general_ci NOT NULL,
  `id_kategori ` varchar(6) COLLATE latin1_general_ci NOT NULL,
  `harga` double NOT NULL,
`stok` int(11) NOT NULL,
  `foto` varchar(500) COLLATE latin1_general_ci NOT NULL,
   PRIMARY KEY (`id_barang`)
)ENGINE=MyISAM DEFAULT CHARSET=latin1
COLLATE=latin1_general_ci;
```

5.1.5. Implementasi Antar Muka

Berikut adalah implementasi antar muka yang telah dirancang dan dibangun dengan menggunakan bahasa pemrograman PHP.

1. Halaman home

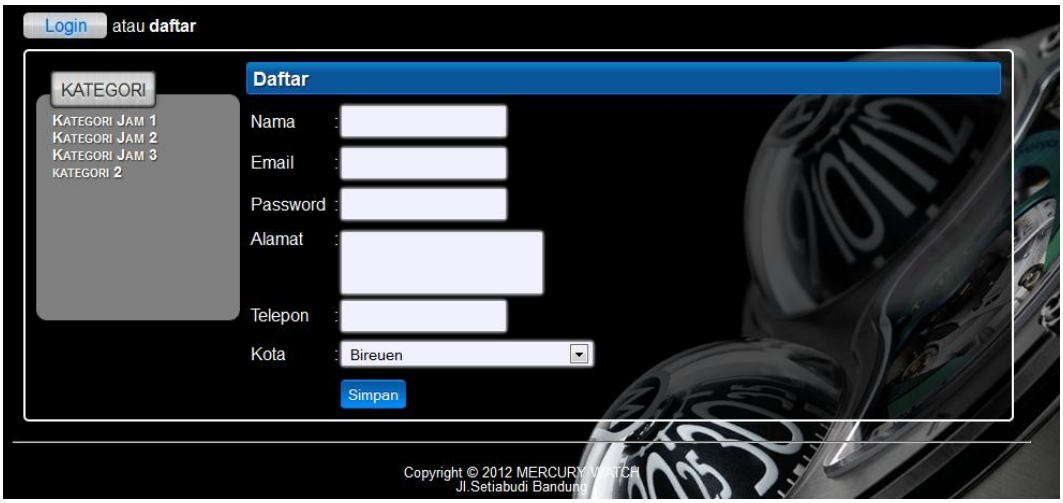
Tampilan menu utama merupakan tampilan utama dari sistem informasi penjualan barang pada Mercury Watch



Gambar 5.0 Halaman home

2. Halaman daftar member

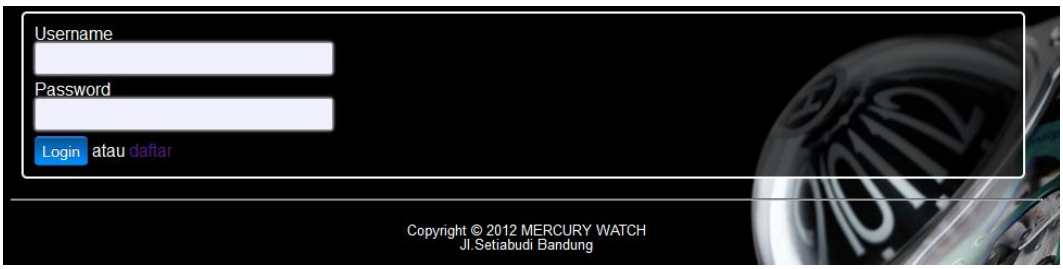
Halaman untuk pendaftaran member baru yang akan memesan produk, di halaman ini member baru dapat mengisi data diri lengkap, seperti : nama lengkap, jenis kelamin, alamat, telepon, *e-mail*, *username* dan *password.* Setelah data diri lengkap calon member dapat mengklik tombol Daftar untuk menyimpan dan tombol reset untuk mengkoreksi data diri



Gambar 5.1 Halaman Daftar member

3. Halaman Login Member

Halaman ini merupakan halaman untuk member yang akan memesan produk, dengan mengisikan username dan password member yang benar sesuai data yang telah disimpan. Kemudian klik tombol Login untuk dapat memesan produk.



Gambar 5.2 Halaman Login member

4. Halaman Products

Halaman ini merupakan halaman yang menampilkan produk-produk yang tersedia yang berdasarkan tiga kategori produk, yaitu :

kategori1, kategori2, kategori3, member cukup mengklik gambar kategori untuk melakukan pemesanan.



Gambar 5.3 Halaman Produk

5. Halaman Pemesanan Produk

Halaman ini merupakan halaman untuk memesan produk yang telah dipilih, yang menampilkan nomor peasanan, informasi produk, dan data member, lalu member mengisi alamat tujuan, kode pos, ukuran produk, kota tujuan dan jumlah produk yang akan dipesan sebagai daftar pesanan, lalu klik tombol simpan.



Gambar 5.4 Halaman Pemesanan Produk

6. Tampilan daftar pesanan

Pada tampilan ini menampilkan peringatan mengenai cara pembayaran melalui rekening antar bank dan juga menampilkan daftar pesanan, yang meliputi : tanggal pesanan, nomor pesanan, data member dan total harga yang harus dibayar. Klik DEAL bila akan memesan dan batl untuk membatalkan.



Gambar 5.5 Tampilan Daftar pesanan

7. Tampilan Konfirmasi pembayaran

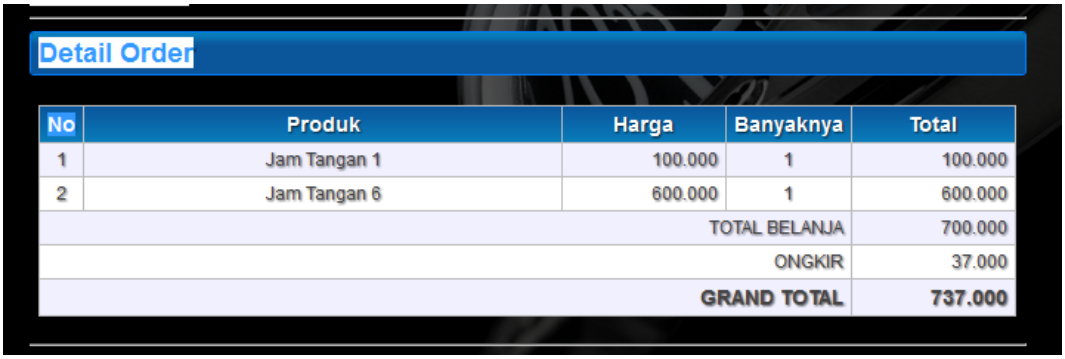
Merupakan tampilan untuk mengkonfirmasi bahwa member telah melakukan pembayaran, dengan mengisikan nomor pesanan sebelumnya, nomor bukti transfer, dan rekening atas nama pengirim. Kilik konfirmasi jika telah melakukan pembayaran.



gambar 5.6 Tampilan konfirmasi pembayaran

8. Bukti pemesanan

Merupakan bukti bahwa member telah melakukan pembayaran.



Gambar 5.7 Bukti pemesanan

9. List Pesanan

Merupakan halaman pada admin yang menampilkan daftar pesanan dari member, dengan status “konfirmasi” jika telah melakukan pembayaran dan “belum konfirmasi” jika belum melakukan pembayaran.



Gambar 5.8 List Pesanan

5.1.6. Implementasi Instalasi Program

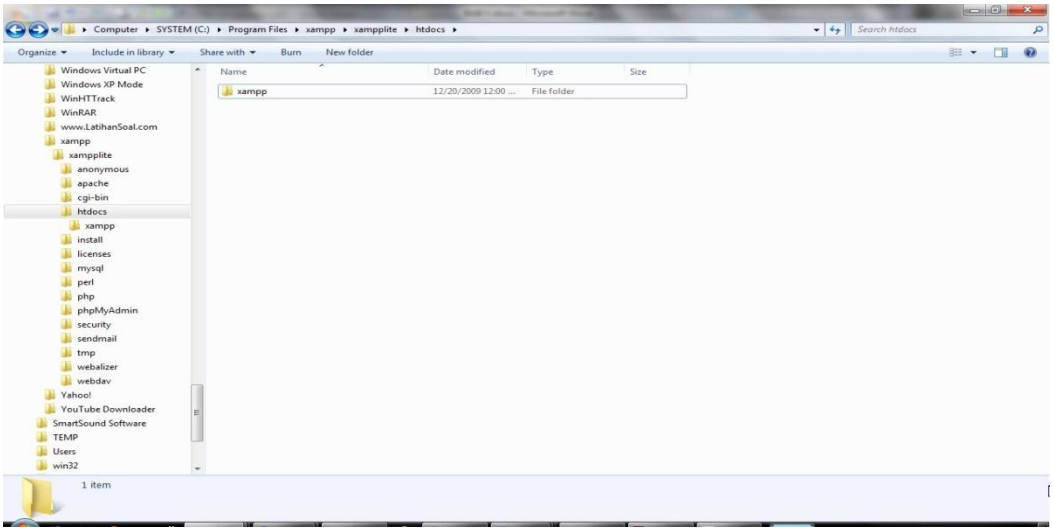
Berikut adalah tahapan-tahapan dalam instalasi program Sistem Informasi Penjualan Barang pada Mercury Watch Bandung :

1. Terlebih dahulu install XAMPP sebagai perangkat lunak pendukung. Cara penginstalannya tidak jauh berbeda dengan cara penginstalan perangkat lunak pada umumnya.



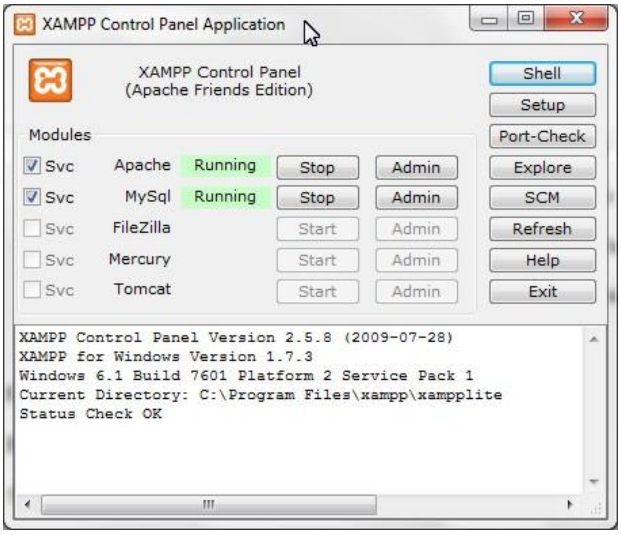
Gambar 5.9 Tampilan *Installer* XAMPP

2. Kemudian buka subfolder htdocs yang berada pada folder xampp, dan copy folder yang berisi data Sistem Informasi Penjualan Barang yang berekstensi .php kedalamnya.



Gambar 5.10 Tampilan subfolder htdocs pada folder xampp

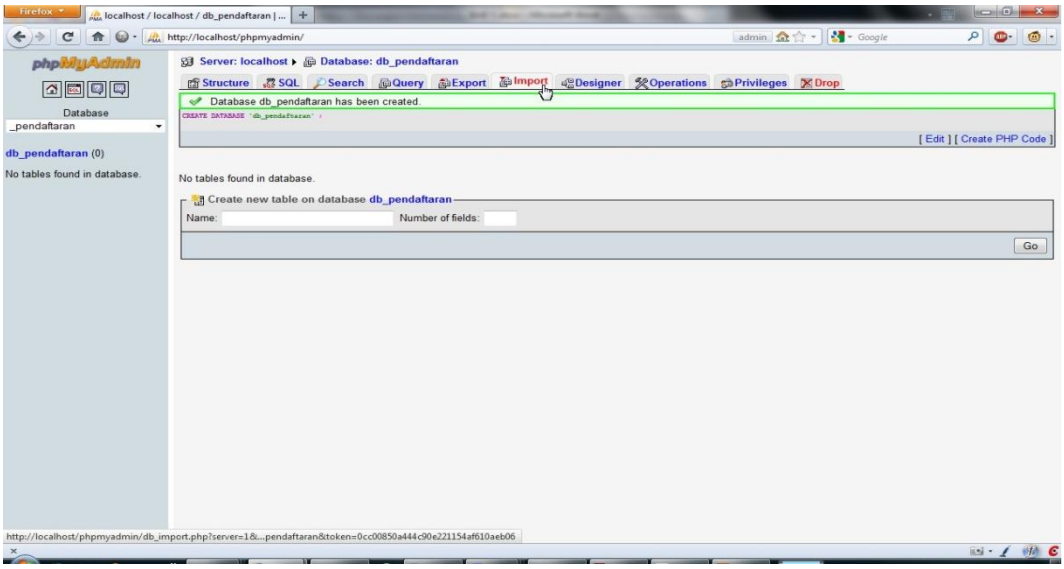
3. Buka xampp control panel dan pastikan Apache dan MySql dalam kondisi *running.*



Gambar 5.11 Tampilan XAMPP *Control Panel*

4. Setelah itu buka web browser, dan ketik "http://localhost/phpmyadmin/", dan buat database Mercury

Watch . Kemudian import tabel-tabel yang sebelumnya telah dibuat. Ke dalam database Mercury Watch.



Gambar 5.12 Tampilan halaman phpMyAdmin

5.1.7. Penggunaan Program

Program ini digunakan dengan tujuan untuk membantu proses penjualan, input dan mengelola data barang, dan laporan penjualan. Program ini hanya bisa diakses oleh dua pengguna yaitu admin dan member.

5.2. Pengujian

Pengujian program adalah pengujian dimana pengguna program memasukkan data ke dalam sistem agar dapat diproses sistem tersebut, dan untuk mempermudah user. Pengujian dilakukan untuk menjamin kualitas serta untuk mengetahui kekurangan dari program. Metode pengujian program yang digunakan adalah dengan menggunakan metode *black box*. Metode ini hanya bekerja pada fungsional program, tidak pada alur internal program.

5.2.1. Rencana Pengujian

Adapun rencana pengujian yang dilakukan adalah :

1. Pengujian *login user*.
2. Pengujian *input* data produk.
3. Pengujian *input* data order.
4. Pengujian *input* data *user*.

5.2.2. Kasus dan Hasil Pengujian

Berikut adalah beberapa kasus dan hasil pengujian yang dilakukan, diantaranya :

1. **Pengujian *Login***

Tabel 5.1 Pengujian *Login*

Data Masukan	Yang Di Harapkan	Pengamatan	Kesimpilan
User Name : Admin Password : Admin1	Tercantum pada Text Box Username dan Password	Dapat Mengisi Login sesuai dengan yang di harapkan	[ ] diterima [ ] ditolak
Klik Login	Dapat masuk ke menu	Tombol login dapat berfungsi sesuai yang di harapkan	[ ] diterima [ ] ditolak
Klik Logout	Dapat keluar dari menu	Tombol Logout dapat berfungsi sesuai yang di harapkan	[ ] diterima [ ] ditolak

2. Pengujian input data

Tabel 5.2 Pengujian input data user

Data Masukan	Yang Di Harapkan	Pengamatan	Kesimpilan
Memasukan data baru yang belum tersimpan dalam table user	Data Tersimpan	Di simpan dalam database pada tabel user	[ ] diterima [ ] ditolak
Memasukan data baru yang tersimpan dalam table user	Data tidak dapat di simpan di table user	Tidak dapat di simpan di database pada tabel user	[ ] diterima [ ] ditolak
Memasukan data kosong	Validasi "sudah ada sebelumnya"	Tidak dapat di simpan di database pada tabel user	[ ] diterima [ ] ditolak

3. Pengujian Input Data Barang

Tabel 5.3 Pengujian Input Data Barang

Data Masukan	Yang Di Harapkan	Pengamatan	Kesimpilan
Memasukan data baru yang belum tersimpan dalam tablebarang	Data Tersimpan	Di simpan dalam database pada tabel barang	[ ] diterima [ ] ditolak
Memasukan data baru yang tersimpan dalam table barang	Data tidak dapat di simpan di table barang	Tidak dapat di simpan di database pada tabel barang	[ ] diterima [ ] ditolak
Memasukan data yang kosong	Validasi "sudah ada sebelumnya"	Tidak dapat di simpan di database pada tabel barang	[ ] diterima [ ] ditolak

4. Pengujian input data order

Tabel 5.4 Pengujian input data order

Data Masukan	Yang Di Harapkan	Pengamatan	Kesimpilan
Memasukan data baru yang belum tersimpan dalam table order	Data Tersimpan	Di simpan dalam database pada tabel order	[ ] diterima [ ] ditolak
Memasukan data baru yang tersimpan dalam table order	Data tidak dapat di simpan di table	Tidak dapat di simpan di database pada tabel order	[ ] diterima [ ] ditolak
Memasukan data baru yang kosong	Validasi nama sudah ada sebelumnya	Tidak dapat di simpan di database pada tabel order	[ ] diterima [ ] ditolak

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian sample di atas dapat ditarik kesimpulan bahwa perangkat lunak secara fungsional mengeluarkan hasil sesuai dengan yang diharapkan.