BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Implementasi adalah proses penerapan rancangan program yang telah dibuat pada bab sebelumnya atau aplikasi dalam melaksanakan sistem informasi pemrograman yang telah dibuat, hasil dari tahapan implementasi ini adalah suatu sistem pengolahan data yang sudah dapat berjalan dengan baik. Dengan demikian dapat diketahui apakah perangkat lunak ini dapat menghasilkan system informasi pemasaran dan penjualan produk foodmart secara *online* pada S28 yang sesuai dengan tujuan yang diharapkan.

5.1.1. Batas Implementasi

1. Tidak semua kebutuhan sistem yang ada di S28 dapat dikerjakan, tetapi hanya subsistem-subsistem yang mendukung khususnya pada proses pemesanan produk foodmart.
2. Perangkat lunak masih merupakan *prototype*.

5.1.2. Implementasi Perangkat Lunak.

Untuk mendukung kelancaran sistem informasi ini selain diperlukan *hardware* juga diperlukan perangkat lunak (*software*) yang terdiri dari sistem operasi dan aplikasi *database*. Adapun perangkat lunak yang diperlukan adalah sebagai berikut:

1. Sistem Operasi Windows menggunakan Microsoft Windows 7
2. Web Server menggunakan Apache
3. Database menggunakan MySQL

4. Script Server menggunakan PHP
5. Design menggunakan Macromedia Dreamweaver CS5, Adobe Photoshop CS5, Artisteer
6. Browser menggunakan Mozilla Firefox.
7. *File transfer protocol* menggunakan *CuteFTP 8*

5.1.3 Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi antara lain:

Server :

1. Processor yang digunakan (Intel Xeon 8x 2.26 GHz) atau setara
2. RAM terpasang 96 GB, rekomendasi minimum 256 MB.
3. Harddisk rekomendasi minimum 80Gb
4. Mouse, Keyboard, monitor dan Printer sebagai peralatan antarmuka.

Client :

1. Processor yang dipergunakan adalah (intel Dual Core 1,86 GHz) atau setara.
2. RAM terpasang 1 GB, rekomendasi minimum adalah 128 MB.
3. Harddisk rekomendasi minimum 20GB
4. Mouse, Keyboard, monitor dan Printer sebagai peralatan antarmuka

5.1.4 Implementasi Basis Data (Sintaks SQL)

-- phpMyAdmin SQL Dump

-- version 2.11.9.2

-- http://www.phpmyadmin.net

--

```
-- Host: localhost
-- Generation Time: Jun 29, 2011 at 04:01 PM
-- Server version: 5.0.67
-- PHP Version: 5.2.6
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
--
-- Database: `dbs28`
--
-- Table structure for table `admins`
--
CREATE TABLE IF NOT EXISTS `admins` (
  `username` varchar(50) collate latin1_general_ci NOT NULL,
  `password` varchar(50) collate latin1_general_ci NOT NULL,
  `nama_lengkap` varchar(100) collate latin1_general_ci NOT NULL,
  `email` varchar(100) collate latin1_general_ci NOT NULL,
  `no_telp` varchar(20) collate latin1_general_ci NOT NULL,
  `level` varchar(20) collate latin1_general_ci NOT NULL default 'user',
  `blokir` enum('Y','N') collate latin1_general_ci NOT NULL default 'N',
  PRIMARY KEY  (`username`)
  ) ENGINE=MyISAM DEFAULT CHARSET=latin1
  COLLATE=latin1_general_ci;
--
-- Table structure for table `berita`
--
```

```
CREATE TABLE IF NOT EXISTS `berita` (
  `id` int(5) NOT NULL auto_increment,
  `judul` varchar(100) NOT NULL,
  `tanggal` varchar(25) NOT NULL default '0000-00-00',
  `isi` text NOT NULL,
  `username` varchar(30) NOT NULL,
  PRIMARY KEY  (`id`)
  )  ENGINE=MyISAM  DEFAULT  CHARSET=latin1
  AUTO_INCREMENT=10 ;
--
-- Table structure for table `bukutamu`
--
CREATE TABLE IF NOT EXISTS `bukutamu` (
  `id` int(11) NOT NULL auto_increment,
  `nama` varchar(30) NOT NULL,
  `email` varchar(50) NOT NULL,
  `tanggal` varchar(20) NOT NULL default '0000-00-00',
  `komentar` text NOT NULL,
  PRIMARY KEY  (`id`)
  )  ENGINE=MyISAM  DEFAULT  CHARSET=latin1
  AUTO_INCREMENT=36 ;
--
-- Table structure for table `bukutamu_reply`
--
```

```
CREATE TABLE IF NOT EXISTS `bukutamu_reply` (
  `id` int(11) NOT NULL,
  `nama` varchar(30) NOT NULL,
  `komentar_reply` text NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
-- Table structure for table `fasilitas`
--
CREATE TABLE IF NOT EXISTS `fasilitas` (
  `id` int(5) NOT NULL auto_increment,
  `judul` varchar(50) NOT NULL,
  `foto` varchar(100) NOT NULL,
  `keterangan` text NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
AUTO_INCREMENT=5 ;
--
-- Table structure for table `foto`
--
CREATE TABLE IF NOT EXISTS `foto` (
  `id` int(11) NOT NULL auto_increment,
  `nama` varchar(40) NOT NULL,
  `gambar` varchar(100) NOT NULL,
  `username` varchar(30) NOT NULL,
```

```
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
  AUTO_INCREMENT=9 ;
--
-- Table structure for table `job`
--
CREATE TABLE IF NOT EXISTS `job` (
  `id` int(11) NOT NULL,
  `tanggal` varchar(20) NOT NULL default '0000-00-00',
  `job` varchar(20) NOT NULL,
  `isi` text NOT NULL,
  `username` varchar(30) NOT NULL,
 PRIMARY KEY  (`id`)
  ) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
-- Table structure for table `kategori`
--
CREATE TABLE IF NOT EXISTS `kategori` (
  `id` int(5) NOT NULL auto_increment,
  `kategori` varchar(20) NOT NULL,
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
AUTO_INCREMENT=5 ;
--
```

```
-- Stand-in structure for view `kondisi`
--
CREATE TABLE IF NOT EXISTS `kondisi` (
`id_orders` int(5)
,`nama_konsumen` varchar(100)
,`alamat` text
,`telpon` varchar(50)
,`email` varchar(50)
,`status` varchar(50)
,`jam_order` time
,`tgl_order` date
,`id_kota` int(5)
,`username` varchar(30)
,`kode` varchar(20)
,`jumlah` int(20)
);
--
-- Table structure for table `kota`
--
CREATE TABLE IF NOT EXISTS `kota` (
  `id_kota` int(5) NOT NULL auto_increment,
  `id_propinsi` int(5) NOT NULL,
  `nama_kota` varchar(30) NOT NULL,
  `biaya` varchar(30) NOT NULL,
```

```
  PRIMARY KEY  (`id_kota`),
  KEY `id_propinsi` (`id_propinsi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=18 ;
--
-- Table structure for table `member`
--
CREATE TABLE IF NOT EXISTS `member` (
  `email` varchar(50) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `password` varchar(50) NOT NULL,
  `alamat` text NOT NULL,
  `provinsi` varchar(50) NOT NULL,
  `kota` varchar(50) NOT NULL,
  `telp` varchar(15) NOT NULL,
  `kode_pos` varchar(5) NOT NULL,
  PRIMARY KEY  (`email`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
-- Table structure for table `orders`
--
CREATE TABLE IF NOT EXISTS `orders` (
  `id_orders` int(5) NOT NULL auto_increment,
  `nama_konsumen` varchar(100) NOT NULL,
```

```
  `alamat` text NOT NULL,
  `telpon` varchar(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `status` varchar(50) NOT NULL,
  `no_resi` varchar(20) NOT NULL,
  `jam_order` time NOT NULL,
  `tgl_order` date NOT NULL,
  `id_kota` int(5) NOT NULL,
  `username` varchar(30) NOT NULL,
  PRIMARY KEY  (`id_orders`),
  KEY `id_kota` (`id_kota`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=134 ;
--
-- Table structure for table `orders_detail`
--
CREATE TABLE IF NOT EXISTS `orders_detail` (
  `id_orders` int(5) NOT NULL,
  `kode` varchar(20) NOT NULL,
  `jumlah` int(20) NOT NULL,
  KEY `id_orders` (`id_orders`),
  KEY `kode` (`kode`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
--
```

```
-- Table structure for table `orders_temp`
--
CREATE TABLE IF NOT EXISTS `orders_temp` (
  `id_orders_temp` int(5) NOT NULL auto_increment,
  `kode` varchar(20) NOT NULL,
  `id_session` varchar(100) NOT NULL,
  `jumlah` int(20) NOT NULL,
  `tgl_order_temp` date NOT NULL,
  `jam_order_temp` varchar(100) NOT NULL,
  PRIMARY KEY  (`id_orders_temp`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ;
--
-- Table structure for table `pembayaran`
--
CREATE TABLE IF NOT EXISTS `pembayaran` (
  `id_orders` int(11) NOT NULL,
  `nama` varchar(50) NOT NULL,
  `tanggal` varchar(10) NOT NULL,
  `total` int(11) NOT NULL,
  `sudah_baca` varchar(1) NOT NULL,
  PRIMARY KEY  (`id_orders`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
--
```

```
-- Table structure for table `produk`
--
CREATE TABLE IF NOT EXISTS `produk` (
  `kode` varchar(20) NOT NULL,
  `nama_produk` varchar(50) NOT NULL,
  `harga` int(20) NOT NULL,
  `stok` int(5) NOT NULL,
  `diskripsi` text NOT NULL,
  `foto` varchar(100) NOT NULL,
  `id_kat` int(5) NOT NULL,
  PRIMARY KEY  (`kode`)
) ENGINE=MyISAM  DEFAULT CHARSET=latin1;
--
-- Table structure for table `propinsi`
--
CREATE TABLE IF NOT EXISTS `propinsi` (
  `id_propinsi` int(5) NOT NULL auto_increment,
  `id_pulau` int(11) NOT NULL,
  `nama_propinsi` varchar(30) NOT NULL,
  PRIMARY KEY  (`id_propinsi`),
  KEY `id_pulau` (`id_pulau`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=41 ;
--
```

```
-- Stand-in structure for view `sisa_stok`
--
CREATE TABLE IF NOT EXISTS `sisa_stok` (
`kode` varchar(20)
,`nama_produk` varchar(50)
,`request` decimal(32,0)
,`stok` int(5)
,`stok_tersedia` decimal(33,0)
);
--
-- Structure for view `kondisi`
--
DROP TABLE IF EXISTS `kondisi`;
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY DEFINER VIEW `dbs28`.`kondisi` AS select `a`.`id_orders` AS `id_orders`,`a`.`nama_konsumen` AS `nama_konsumen`,`a`.`alamat` AS `alamat`,`a`.`telpon` AS `telpon`,`a`.`email` AS `email`,`a`.`status` AS `status`,`a`.`jam_order` AS `jam_order`,`a`.`tgl_order` AS `tgl_order`,`a`.`id_kota` AS `id_kota`,`a`.`username` AS `username`,`b`.`kode` AS `kode`,`b`.`jumlah` AS `jumlah` from (`dbs28`.`orders_detail` `b` join `dbs28`.`orders` `a` on((`a`.`id_orders` = `b`.`id_orders`)));
--
-- Structure for view `sisa_stok`
--
```

```
DROP TABLE IF EXISTS `sisa_stok`;

CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL SECURITY DEFINER VIEW `dbs28`.`sisa_stok` AS select `k`.`kode` AS `kode`,`p`.`nama_produk` AS `nama_produk`,sum(`k`.`jumlah`) AS `request`,`p`.`stok` AS `stok`,(`p`.`stok` - sum(`k`.`jumlah`)) AS `stok_tersedia` from (`dbs28`.`kondisi` `k` join `dbs28`.`produk` `p` on((`p`.`kode` = `k`.`kode`))) where (`k`.`status` <> _latin1'dikirim') group by `k`.`kode`;
```

5.1.5. Implementasi Antar Muka

Implementasi antarmuka merupakan tahapan dalam memenuhi kebutuhan pengguna (*user*) dalam berinteraksi dengan komputer. Fasilitas antar muka yang baik sangat membantu pemakai dalam memahami proses yang sedang dilakukan oleh sistem tersebut dan dapat meningkatkan kinerja sistem. Berikut ini adalah implementasi antar muka yang di rancang

1. Tampilan halaman utama



Gambar 5.1 Tampilan Halaman Utama

Halaman utama atau index menampilkan gambar slide yang bertujuan agar produk / event terbaru dapat dilihat oleh semua pengunjung website. Selain produk / event terbaru juga menampilkan sekilas tentang perusahaan S28 agar pengunjung lebih mengenal S28.

2. Tampilan Halaman Produk



Gambar 5.2 Tampilan Halaman Produk Foodmart

Halaman produk foodmart menampilkan semua sekilas isi dari produk yang berkategori *foodmart* dan halaman ini dapat diakses oleh semua pengunjung. Dan nantinya akan terhubung ke halaman detail produk.

3. Tampilan Halaman Detail Produk



Gambar 5.3 Tampilan Detail Produk

Halaman ini berisikan detail tentang produk yang telah di pilih oleh pengunjung website. Detail yang ditampilkan berupa foto, nama produk, harga dan keterangan tentang produk. Untuk melanjutkan proses belanja, pengunjung dapat menekan tombol masukkan ke keranjang belanja atau icon keranjang belanja yang nantinya akan menghubungkan kehalaman keranjang belanja.

4. Tampilan Halaman Keranjang Belanja



Gambar 5.4 Tampilan Halaman Keranjang Belanja

Halaman ini dapat diakses oleh semua pengunjung, namun untuk melakukan *check out* / selesai belanja, hanya bisa dilakukan oleh pengunjung yang sudah terdaftar sebagai member S28. Sedangkan pengunjung yang belum terdaftar proses pembelian hanya sampai dikeranjang belanja dan tidak bisa melakukan proses transaksi pembelian.

5. Tampilan Halaman Registrasi *Customer*



Gambar 5.5 Tampilan Halaman Registrasi *Customer*

Halaman registrasi ini berfungsi untuk pengunjung yang ingin menjadi member atau ingin membeli produk yang ditawarkan oleh S28. Karena untuk melakukan proses pembelian pengunjung harus melakukan login. Dihalaman ini pengunjung diharuskan mengisi biodata secara lengkap karena nantinya data biodata yang telah di isi akan menjadi alamat pengiriman barang pesanan yang sudah dipesan oleh *customer*.

6. Tampilan Halaman Profil



Gambar 5.6 Tampilan Halaman Profil *Customer*

Halaman ini menampilkan data pribadi *customer* yang sudah melakukan login, di halaman ini *customer* bisa melakukan edit data pribadi, melihat info status pemesanan barang yang sudah dipesan dan melakukan konfirmasi pembayaran apabila telah melakukan pembayaran terhadap produk yang dipesan. *Customer* juga bisa melihat kembali list pesanan barang yang sudah dipesan apabila belum melakukan *check out* dan juga melakukan interaksi dengan pengelola website dengan fasilitas *message* yang ada didalam halaman profil.

7. Tampilan Halaman *Check out*



Gambar 5.7 Tampilan Halaman *check out* / Selesai Belanja

Halaman ini hanya dapat diakses oleh *customer* yang sudah terdaftar, halaman ini merupakan proses dari halaman keranjang belanja. Setelah *customer* memilih produk pesanannya, maka secara otomatis data tujuan pengiriman akan muncul apabila *customer* telah menekan tombol selesai belanja. Disini *customer* bisa memilih opsi pengiriman pesanan, yaitu mengirim ke alamat yang sudah ada dan mengirim kealamat lain.

8. Tampilan Halaman Proses Selesai Belanja



Gambar 5.8 Tampilan Halaman Selesai Belanja

Apabila *customer* telah menyelesaikan semua proses pesanannya, maka secara otomatis halaman detail semua dari detail balanja akan ditampilkan, mulai dari tujuan pengiriman, data barang yang dipesan dan total jumlah transaksi akan ditampilkan. Untuk data nomor rekening, dan ketentuan khusus akan dikirim ke email *customer* sesuai dengan email yang sudah didaftarkan ke database member S28.

9. Tampilan Halaman Info Status Pesanan



Gambar 5.9 Tampilan Halaman Info Status Pesanan

Halaman info status pesanan berfungsi untuk setiap *customer* melihat info status pesanan mereka. Data yang ditampilkan disini adalah data order *customer* mulai dari status pesanan baru, lunas dan dikirim. Status baru adalah status pesanan yang belum melakukan pembayaran, status lunas adalah status pesanan yang sudah melakukan pembayaran, namun barang belum dikirim oleh pihak S28 dan belum terdapat nomor resi pengiriman. Status barang dikirim adalah pesanan yang sudah dikirim oleh pihak S28 dan sudah terdapat nomor resi pengiriman, sehingga *customer* bisa melakukan *tracking* barang pesanannya.

10. Tampilan Halaman Konfirmasi Pembayaran



Gambar 9.10 Tampilan Halaman Konfirmasi Pembayaran

Halaman konfirmasi pembayaran berfungsi untuk *customer* melakukan konfirmasi bahwa sudah melakukan konfirmasi pembayara. Disini *customer* harus mengisi data yang valid sesuai dengan transaksi pembayaran yang telah dilakukan, validasi yang paling utama adalah nama, nama harus diisi sesuai dengan nama yang tercantum pada saat melakukan pembayaran. Selebihnya validasi akan dilakukan oleh sistem, ketidakcocokan data akan ditolak oleh sistem, untuk itu data yang di input harus benar-benar valid sesuai dengan data yang ada.

11. Tampilan Halaman Admin



Gambar 5.11 Tampilan Halaman Utama Admin

Admin yang telah berhasil melakukan login, maka secara otomatis akan masuk ke halaman utama admin. Halaman ini berfungsi untuk mengelola data website, baik itu data konten dan data transaksi pembelian.

12. Tampilan Halaman Validasi Pembayaran



Gambar 5.12 Tampilan Halaman Validasi Pembayaran

Halaman validasi pembayaran berfungsi untuk admin melakukan pengecekan pembayaran ke rekening S28. Halaman ini akan menampilkan data yang diinput oleh *customer* di halaman konfirmasi pembayaran. Selanjutnya admin akan melakukan proses ubah status menjadi lunas.

13. Tampilan Halaman Ubah Status



Gambar 9.13 Tampilan Halaman Ubah Status

Halaman ini berfungsi untuk admin mengelola pengiriman barang, namun untuk proses pengiriman belum dilakukan, admin hanya melakukan konfirmasi ke *customer* bahwa pesanan *customer* sedang diproses oleh pihak S28.

14. Tampilan Halaman Konfirmasi Pengiriman



Gambar 5.14 Tampilan Halaman Konfirmasi Pengiriman

Halaman konfirmasi pengiriman ini berfungsi untuk admin melakukan proses pengiriman barang, yang tadinya status pesanan lunas, dihalaman ini admin menggantikan status pesanan menjadi dikirim dan di sertakan proses penginputan data nomor resi, sehingga nantinya *customer* yang ingin melakukan *tracking* barang, bisa melakukan validasi nomor resi. Dalam hal ini, pihak penyedia jasa pengiriman adalah JNE.

5.1.6. Implementasi Instalasi Program

Untuk dapat menggunakan aplikasi *website* sistem informasi pemasaran dan penjualan produk *foormart* secara online ini maka kita harus melakukan proses mulai dari membeli hosting, dan melakukan upload data website kedalam hostingan. Dalam melakukan upload data kedalam hostingan, penulis

menggunakan *software file transfer protocol* yang bernama *CuteFTP 8 Profesional*. Penggunaan *software* ini mempermudah pengelola website dalam melakukan manajemen website. Adapun tahapan-tahapan yang dilakukan dalam proses upload data ke penyedia hostingan antara lain.

1. Melakukan Login kedalam hostingan yang telah terdaftar dengan menggunakan web broser dan mengakases halaman hostingan tersebut dan melakukan input *username* dan *password*. Hosting yang digunakan adalah http://namadomain/cpanel



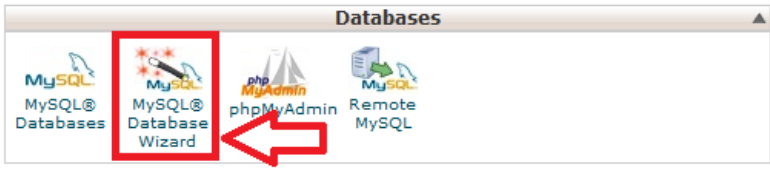
Gambar 5.15 Tampilan halaman login cPanel

2. Setelah melakukan login, maka secara otomatis halaman akan *redirect* kehalaman cPanel hosting yang berisikan seluruh konfigurasi untuk perangkat lunak yang akan dibangun.



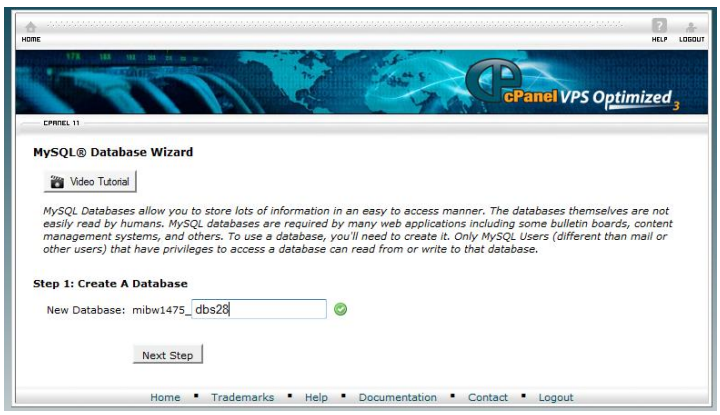
Gambar 5.16 Tampilan Halaman Konfigurasi cPanel

3. Setelah tampil halaman Cpanel langkah berikutnya yang harus dilakukan adalah membuat database pada hosting dengan memilih menu MySQL yang ada pada cPanel hosting.



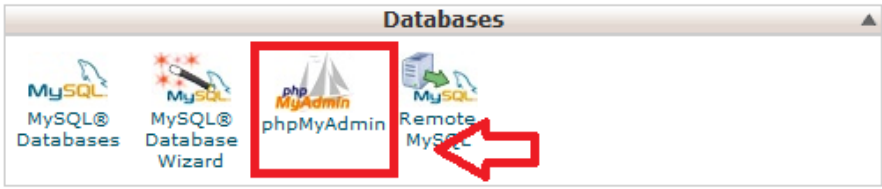
Gambar 5.17 Tampilan Halaman MySQL Database Wizard

4. Langkah berikutnya adalah membuat nama database, disini kita menginputkan nama database sesuai dengan apa yang akan kita buat



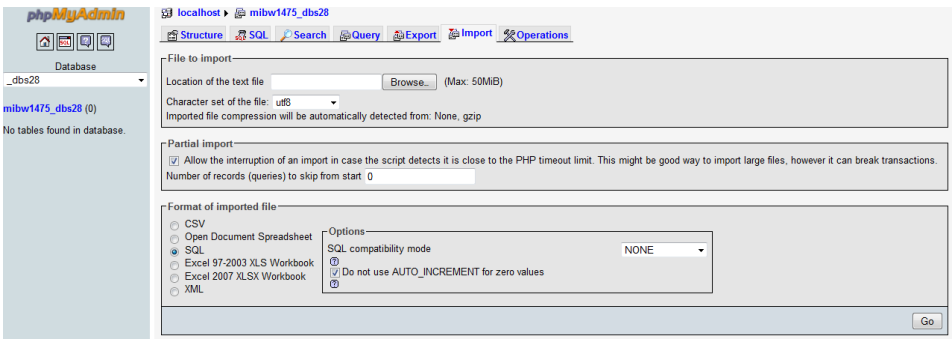
Gambar 5.18 Tampilan Halaman Database Wizard

5. Setelah mengisi nama database, langkah berikutnya adalah menekan tombol *next step* dan akan memunculkan halaman pembuatan database *user*. Proses yang dilakukan disini adalah menginputkan nama user dan password untuk akses kedalam database. Langkah berikutnya adalah memberikan hak askes user kedalam database.
6. Setelah pembuatan database selesai, langkah berikutnya adalah melakukan *import* database yang sebelumnya telah kita buat di server localhost. Untuk memulai proses import, kita memilih menu phpMyAdmin yang ada di halaman konfigurasi cPanel hosting.



Gambar 5.19 Tampilan Halaman Konfigurasi phpMyAdmin

7. Setelah memilih halaman phpMyAdmin yang ada di halaman konfigurasi cPanel, maka menampilkan halaman utama phpMyAdmin, layaknya halaman phpMyAdmin yang ada di localhost. Langkah berikutnya adalah memilih menu import untuk melakukan import database yang sudah kita export sebelumnya dengan menggunakan localhost.



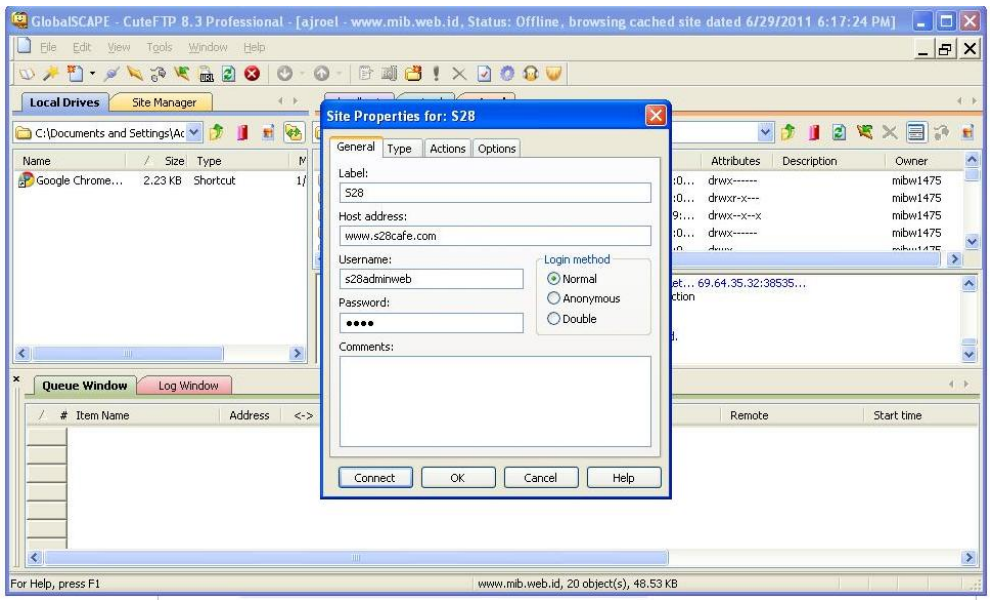
Gambar 5.20 Tampilan Halaman phpMyAdmin Hosting

8. Setelah database berhasil diimport, maka proses selanjutnya adalah mencocokkan kembali file koneksi.php yang ada di dalam folder S28Cafe/config/koneksi.php yang ada di server local.

   Kemudian mengubah data username dan password sesuai dengan username dan password yang telah di tambahkan dalam proses pembuatan database tadi.

9. Langkah berikutnya adalah melakukan upload data website yang ada di server local ke server hostingan, dalam kasus ini penulis menggunakan *software CuteFTP 8 Profesional.*

10. Langkah pertama adalah instalasi *software CuteFTP 8* ke dalam komputer local, kemudia mengisikan label, *host address* diisi sesuai dengan host website sesuai dengan nama hostingan. Kemudia mengisi username dan password sesuai dengan yang diberikan oleh penyedia hostingan.



Gambar 5.21 Tampilan Halaman *Software CuteFTP*

11. Setelah proses login dengan menggunakan *CuteFTP* selesai, maka langkah berikutnya adalah upload seluruh file dan folder perangkat lunak sistem informasi pemasaran dan penjualan pada S28 yang ada didalam komputer local kedalam server hostingan.
12. Setelah upload file selesai maka untuk membuktikan apakah dapat berjalan dengan baik dan benar adalah dengan melakukan test menggunakan web broser dan mengisi alamat yang sesuai dengan nama domain yang ada kedalam address bar http://www.namadomain.com.

5.1. Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak .Pengujian dilakukan untik menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal, yaitu merepresentasikan kajian pokok dari spesifikasi, analis perancangan dan pengkodean dari perangkat lunak itu sendiri.

5.2.1. Rencana Pengujian

Rencana pengujian pengelolaan Sistem Informasi Pemasaran dan Penjualan berikut menggunakan data uji berupa sebuah data masukan dari user pelanggan, dan admin pada website yang telah dibuat.

Tabel 5.1 Rencana Pengujian

Item Pengujian	Deskripsi	Jenis Pengujian
Login	Pengecekan login semua pengguna	*Black Box*
Proses Registrasi member	Memeriksa kelengkapan data inputan	*Black Box*

Tabel Lanjutan		
Item Pengujian	**Deskripsi**	**Jenis Pengujian**
Proses Penginputan konfirmasi pembayaran	Memeriksa proses konfirmasi pembayaran	*Black Box*
Proses mengelola data produk	Memeriksa proses pengolahan data produk	*Black Box*

5.2.2. Kasus dan Hasil Pengujian

5.2.2.1. Pengujian Login

Pengujian login terbagi menjadi tiga bagian yaitu login admin, login user pengelola dan login *customer*.

1. Login Admin dan User

Berikut ini adalah table pengujian login Admin dan user pengelola website S28 :

Tabel 5.2 Pengujian Login Admin

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username : Admin Password : admin Hak akses : administrator	Masuk ke halaman admin dengan hak akses administrator	Dapat mengakses halaman admin sesuai dengan apa yang diharapkan	[X] diterima [ ] ditolak

Tabel Lanjutan			
Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Username : Budi Password : budilili Hak Akses : user	Masuk kehalaman admin dengan hak akses user	Dapat mengakses halaman admin, namun dengan hak akses sebagai user yang hanya bisa mengakses beberapa proses saja	[X] diterima [ ] ditolak
Kasus Dan Hasil Uji (Data Salah)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Username : bebas Password : 12345	Tidak dapat melakukan login	Menampilkan pesan login gagal “username atau password salah	[X] diterima [ ] ditolak

2. Pengecekan Login *Customer*

Berikut ini adalah table pengujian login untuk pengecekan *customer* yang telah terdaftar :

Tabel 5.3 Pengecekan logi *customer*

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Email : siddiq.dizzy@mail.com Password : achmad	Dapat masuk ke menu profil, keranjang belanja, logout	Dapat melakukan pemesanan dan konfirmasi pembayaran	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Email : bebas@gmail.com Password : 12345	Tidak dapat melakukan login	Menampilkan pesan login gagal “maaf	[X] diterima [ ] ditolak
		Anda belum terdaftar sebagai customer kami	

4.2.2.2. Proses Registrasi *Customer*

Tabel 5.4 Pengecekan registrasi *customer*

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi lengkap data *customer* baru	Data tersimpan kedalam database	Data *customer* tersimpan sesuai dengan yang diharapkan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi lengkap data *customer* baru, namun untuk data email mengisi email yang sudah terdaftar sebagai *custumer*	Data tidak tersimpan kedalam database	Menampilkan pesan kesalahan "Email anda telah terdaftar"	[X] diterima [ ] ditolak
Kasus Dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pengisian data *customer* tidak lengkap	Data tidak tersimpan kedalam database	Menampilkan pesan eror disetiap field yang tidak lengkap	[X] diterima [ ] ditolak

4.2.2.3. Proses Penginputan Konfirmasi Pembayaran

Tabel 5.5 Pengecekan pengelolaan data konfirmasi pembayaran

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi lengkap dan sesuai dengan data transaksi	Data tersimpan kedalam database	Data konfirmasi tersimpan dan sesuai dengan yang diharapkan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi lengkap data, namun tidak sesuai dengan data transaksi	Data tidak tersimpan kedalam database	Menampilkan pesan kesalahan “Data tidak valid”	[X] diterima [ ] ditolak
Kasus Dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pengisian data konfirmasi tidak lengkap	Data tidak tersimpan kedalam database	Menampilkan pesan eror disetiap field yang tidak lengkap	[X] diterima [ ] ditolak

4.2.2.4. Proses Mengelola Data Produk

Tabel 5.6 Pengecekan pengelolaan data produk

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi data produk secara lengkap	Data tersimpan kedalam database	Data produk tersimpan dan sesuai dengan yang diharapkan	[X] diterima [ ] ditolak
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Mengisi lengkap data, namun mengisi kode produk yang sudah ada	Data tidak tersimpan kedalam database	Menampilkan pesan kesalahan “Data produk sudah ada”	[X] diterima [ ] ditolak
Kasus Dan Hasil Uji (Data Salah)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Pengisian data produk tidak lengkap	Data tidak tersimpan kedalam database	Menampilkan pesan eror disetiap field yang tidak lengkap	[X] diterima [ ] ditolak

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus uji *sample* diatas dapat ditarik kesimpulan bahwa perangkat lunak berjalan dengan baik dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan perusahaan.