BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. Implementasi Sistem

Implementasi prototype sistem informasi pemesanan tiket online di bioskop braga 21 dilakukan dengan menggunakan bahasa dan lingkungan pemrograman PHP dengan macromedia dreamweaver 8 sebagai perangkat lunaknya, dan juga SQL sebagai basis data dengan MySQL sebagai perangkat lunaknya. Bahasa pemograman PHP tersebut dapat dijalankan pada berbagai platform sistem operasi dan perangkat keras, tetapi implementasi dan pengujian sepenuhnya hanya dilakukan di perangkat keras PC (personal computer) dengan sistem operasi Microsoft Windows 7.

5.1.1. Batasan Implementasi (optional)

Dalam mengimplementasikan perangkat lunak, batasan sebagai pengendalian aplikasi ini ada beberapa hal yang perlu dibatasai dalam implementasi ini yaitu :

1. Perangkat lunak yang digunakan dalam pengimplementasian basis data ini adalah MySQL yang telah terkompilasi kedalam sebuah perangkat lunak bebas yaitu wamp.

2. Tampilan untuk web based menggunakan bahasa Indonesia dan Inggris.

5.1.2. Implementasi Perangkat Lunak

Untuk implementasi perangkat lunak ini menggunakan Wamp 2.0 , Macromedia dreamweaver 8, MySQL 5.0.45 dan PHP version 5.2.5. Wamp 2.0 dipilih sebagai perangkat lunak untuk webserver karena faktor kecepatan, kinerja yang stabil, dan performansi PHP dan Macromedia dreamweaver 8 dipilih sebagai perangkat lunak pengembang karena menyediakan fasilitas yang memadai dan membuat perangkat lunak yang berbasis web. Sementara itu MySQL digunakan sebagai pengembang dalam pembuatan basis data.

5.1.3. Implementasi Perangkat Keras

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

1. Processor Minimal Pentium atau yang sekelasnya.
2. Menggunakan minimal RAM 256 MB.
3. Harddisk 40 Gb.
4. Keyboard, Monitor dan Mouse.

5.1.4. Implementasi Basis Data

```
CREATE TABLE IF NOT EXISTS `admin` (
  `Nama` varchar(30) NOT NULL,
```

```
  `Username` varchar(25) NOT NULL,
  `Pass` varchar(30) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
CREATE TABLE IF NOT EXISTS `berita` (
  `kode_berita` varchar(10) NOT NULL,
  `judul_berita` varchar(60) NOT NULL,
  `isi_berita` text NOT NULL,
  `tgl_posting_berita` date NOT NULL,
  `jam_posting_berita` time NOT NULL,
  `foto_berita` varchar(200) NOT NULL,
  PRIMARY KEY (`kode_berita`),
  UNIQUE KEY `judul_berita` (`judul_berita`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
-- --------------------------------------------------------

CREATE TABLE IF NOT EXISTS `film` (
  `id_film` varchar(10) NOT NULL,
  `judul_film` varchar(60) NOT NULL,
```

```
  `kd_genre` varchar(10) NOT NULL,
  `kd_klasifikasi` varchar(10) NOT NULL,
  `tahun_produksi` year(4) NOT NULL,
  `rumah_produksi` varchar(60) NOT NULL,
  `sinopsis` text NOT NULL,
  `foto_film` varchar(200) NOT NULL,
  `durasi_film` varchar(10) NOT NULL,
  `kd_theater` varchar(10) NOT NULL,
  PRIMARY KEY (`id_film`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

CREATE TABLE IF NOT EXISTS `ganti` (
  `no` int(2) NOT NULL AUTO_INCREMENT,
  `nama_p` varchar(30) NOT NULL,
  `nama_p2` varchar(30) NOT NULL,
  `nama_p3` varchar(30) NOT NULL,
  `nama_p4` varchar(30) NOT NULL,
  `logo` varchar(200) NOT NULL,
  PRIMARY KEY (`no`)
```

```
) ENGINE=MyISAM  DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

-- --------------------------------------------------------

CREATE TABLE IF NOT EXISTS `genre` (
  `kd_genre` varchar(10) NOT NULL,
  `nama_genre` varchar(30) NOT NULL,
  PRIMARY KEY (`kd_genre`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

CREATE TABLE IF NOT EXISTS `jenis_kelamin` (
  `kode_jk` varchar(10) NOT NULL,
  `nama_jk` varchar(20) NOT NULL,
  PRIMARY KEY (`kode_jk`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

CREATE TABLE IF NOT EXISTS `kategori_theater` (
  `kd_kategori_th` varchar(10) NOT NULL,
  `nama_kategori` varchar(60) NOT NULL,
  `kapasitas_th` varchar(5) NOT NULL,
  `harga` varchar(20) NOT NULL,
```

```
  PRIMARY KEY (`kd_kategori_th`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
CREATE TABLE IF NOT EXISTS `klasifikasi_film` (
  `kd_klasifikasi_film` varchar(10) NOT NULL,
  `nama_klasifikasi_film` varchar(30) NOT NULL,
  `inisial_klasifikasi_film` varchar(10) NOT NULL,
  PRIMARY KEY (`kd_klasifikasi_film`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
CREATE TABLE IF NOT EXISTS `manifest` (
  `no_manifest` varchar(100) NOT NULL,
  `id_film` varchar(15) NOT NULL,
  `kd_theater` varchar(20) NOT NULL,
  `kursi_tersedia` int(11) NOT NULL,
  PRIMARY KEY (`no_manifest`,`id_film`,`kd_theater`),
  KEY `kode_bus` (`id_film`),
  KEY `kode_jadwal` (`kd_theater`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
-- --------------------------------------------------------

CREATE TABLE IF NOT EXISTS `pengguna` (
  `id_user` varchar(15) NOT NULL,
  `password_us` varchar(30) NOT NULL,
  `username_us` varchar(30) NOT NULL,
  `nama_us` varchar(30) NOT NULL,
  `kode_jk` varchar(10) NOT NULL,
  `tgl_lahir_us` date NOT NULL,
  `alamat_us` varchar(60) NOT NULL,
  `kota_us` varchar(60) NOT NULL,
  `email_us` varchar(60) NOT NULL,
  `no_telp` varchar(20) DEFAULT NULL,
  PRIMARY KEY (`id_user`),
  UNIQUE KEY `username_us` (`username_us`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

CREATE TABLE IF NOT EXISTS `theater` (
  `kd_theater` varchar(10) NOT NULL,
  `nama_theater` varchar(60) NOT NULL,
```

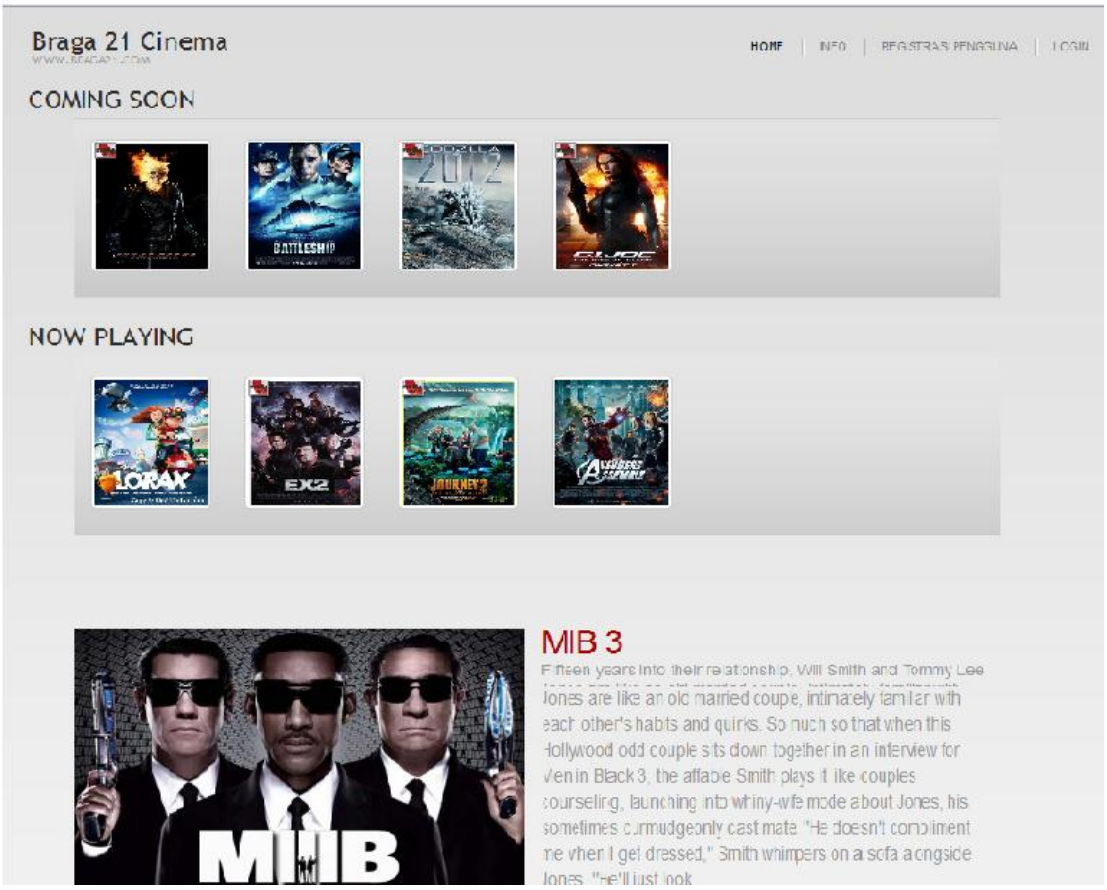
```
  `kd_katagori_th` varchar(10) NOT NULL,
  PRIMARY KEY (`kd_theater`),
  UNIQUE KEY `nama_theater` (`nama_theater`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
CREATE TABLE IF NOT EXISTS `tiket` (
  `no_tiket` varchar(15) NOT NULL,
  `no_manifest` varchar(100) NOT NULL,
  `id_pengguna` varchar(10) NOT NULL,
  `kd_theater` varchar(15) NOT NULL,
  `status_tiket` varchar(20) NOT NULL,
  `no_kursi` varchar(10) NOT NULL,
  PRIMARY KEY (`no_tiket`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

5.1.5. Implementasi Antar Muka

Implementasi antar muka merupakan tahapan dalam memenuhi kebutuhan pengguna (user) dalam berinteraksi dengan komputer. Dalam tahapan implementasi ini sebuah website tiket bioskop mempunyai fasilitas antarmuka yang baik untuk membantu pemesan dalam melihat, mencari, mendapatkan apa yang sekiranya diperlukan.

5.1.5.1. Menu User

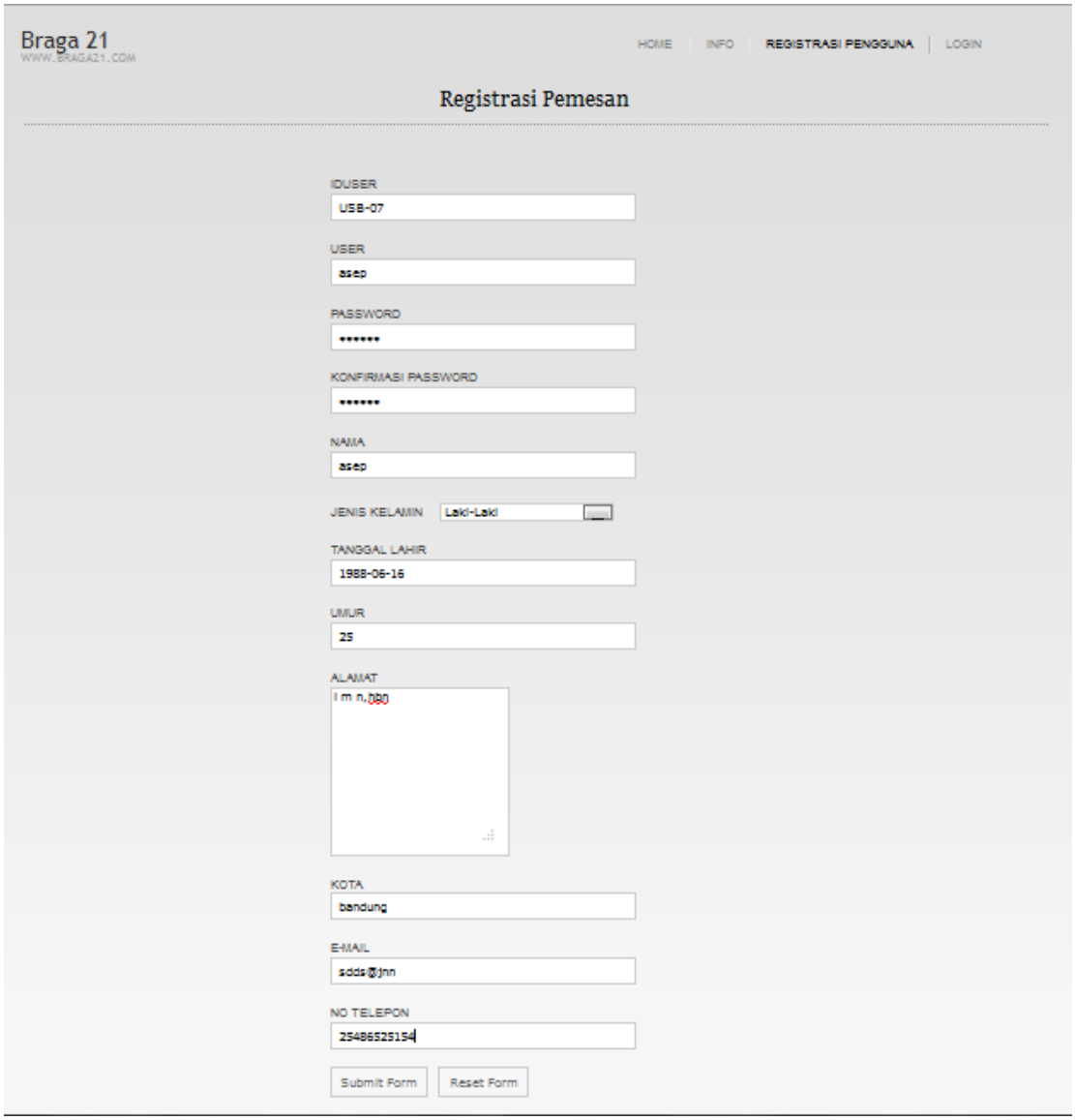
Pada pertama kita menggunakan aplikasi ini, kita langsung dihadapkan dengan tampilan halaman utama, tampilannya seperti terlihat pada gambar berikut ini:



Gambar 5.1 Menu user

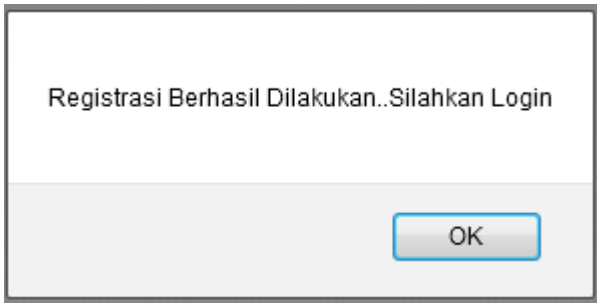
Setelah diperlihatkan halaman utama website lalu kita memilih atau klik registrasi pengguna, registrasi pengguna ini berfungsi sebagai proses pertama kali dalam

melakukakan pemesanan, registrasi pengguna ini untuk mendapatkan hak akses dalam melakukan pemesanan. Seperti yang terlihat gambar dibawah ini :



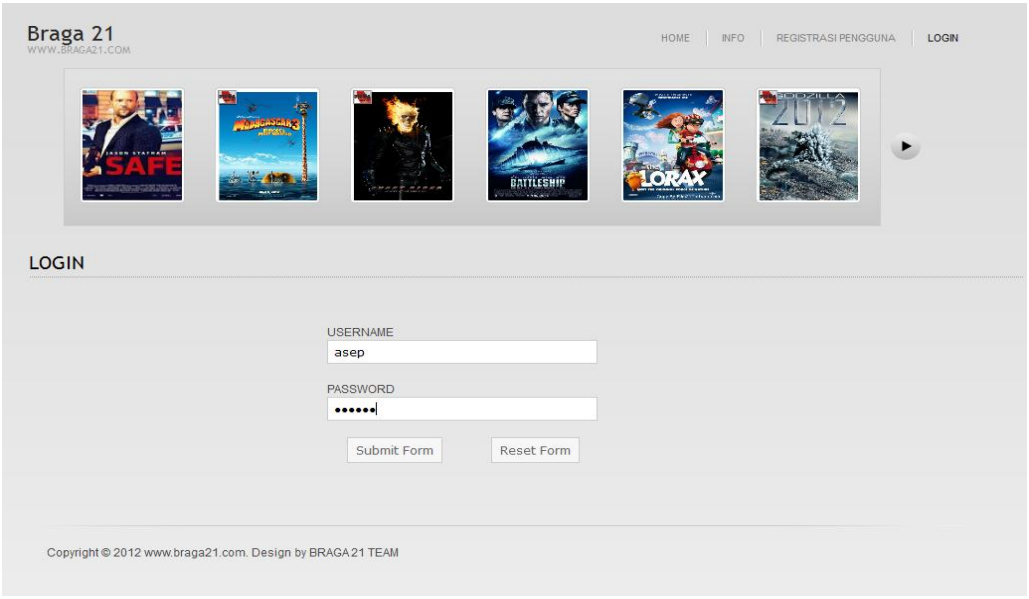
Gambar 5.2 Registrasi pemesan

Setelah selesai mengisi registrasimaka akan muncul pemberitahuan, seperti gambar dibawah ini :



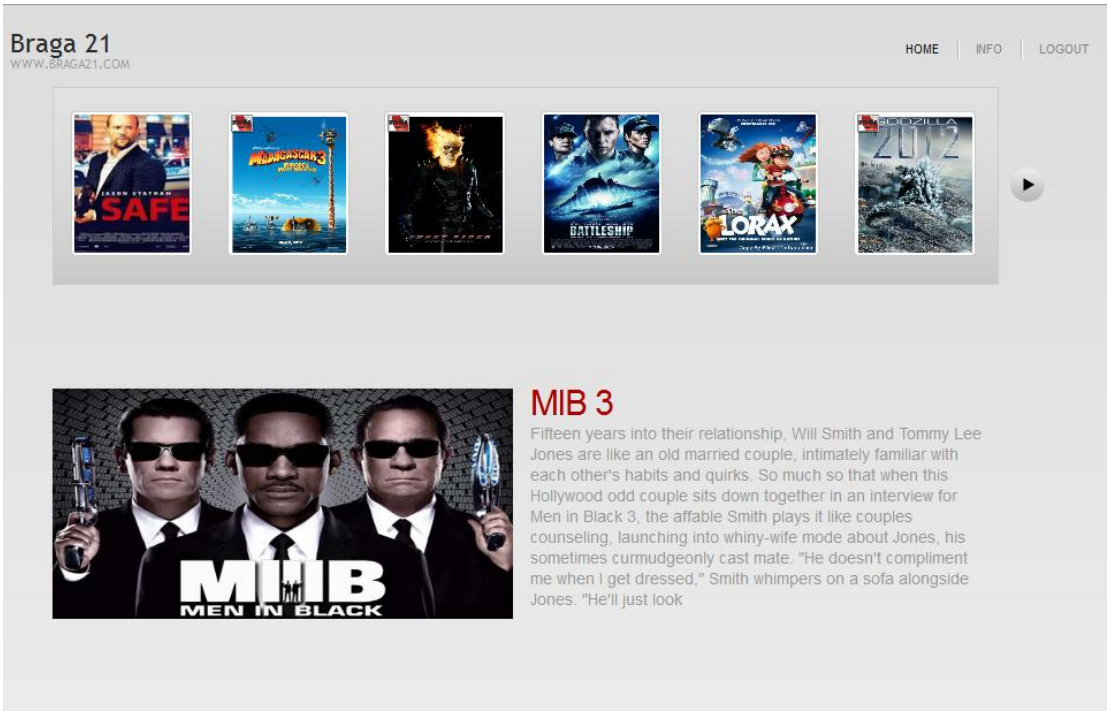
Gambar 5.3 Pemberitahuan registrasi berhasil

Klik tombol ok maka user akan dihadapkan pada tampilan halaman login, lalu isi form login tersebut sesuai dengan data registrasi. Seperti yang ada pada gambar dibawah ini :



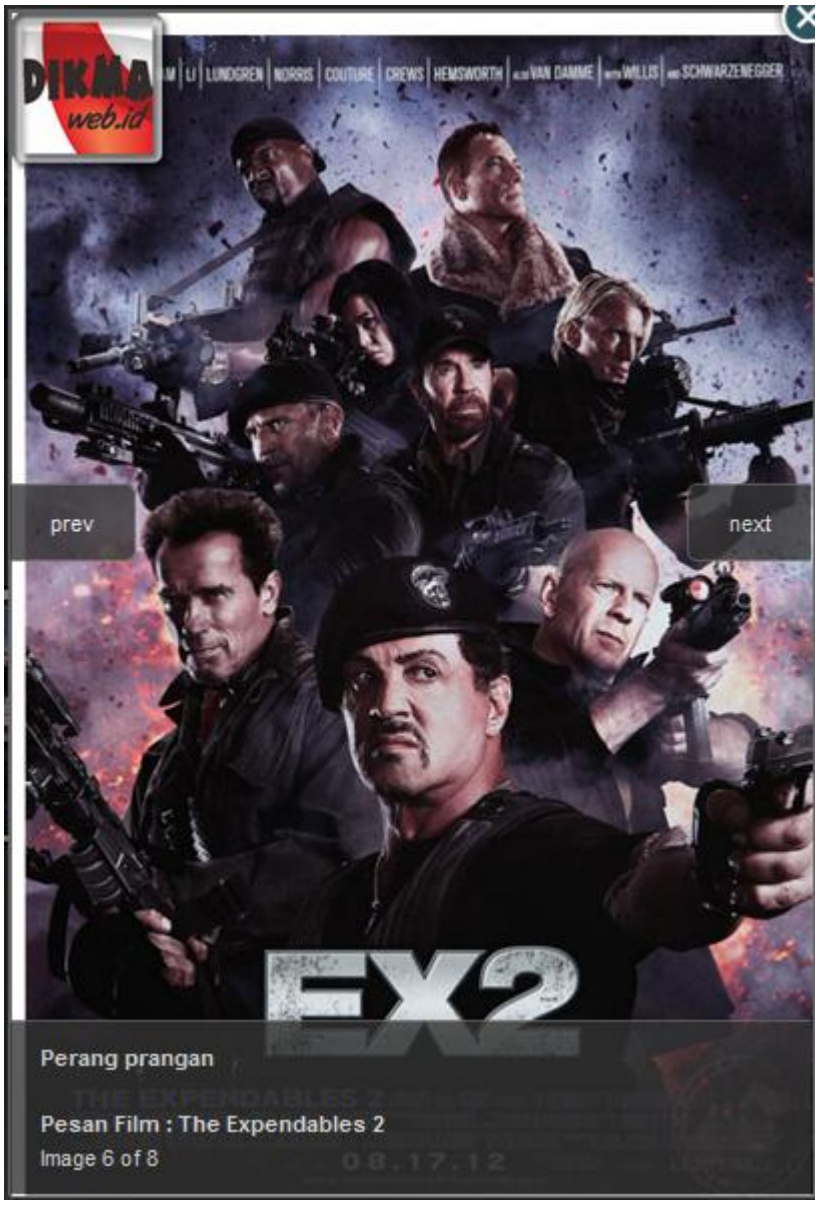
Gambar 5.4 Login

Setelah diisi username dan password nya lalu klik tombol submit untuk masuk ke halaman pengguna atau user. Seperti yang ada pada gambar dibawah ini :



Gambar 5.5 Halaman user

Lalu pilih salah satu film yang mau anda pesan yang ada di deretan atas, maka akan muncul seperti gambar dibawah ini :



Gambar 5.6 Film yang dipesan

Setelah muncul gambar tersebut maka jika kita tempelkan kursor pada gambar akan muncul tulisan di bawah gambar tersebut maka ada tulisan pesan film klik disitu,

maka akan muncul form informasi jadwal tayang film tersebut seperti gambar dibawah ini :

JUDUL FILM : **The Expendables 2**



Sinopsis

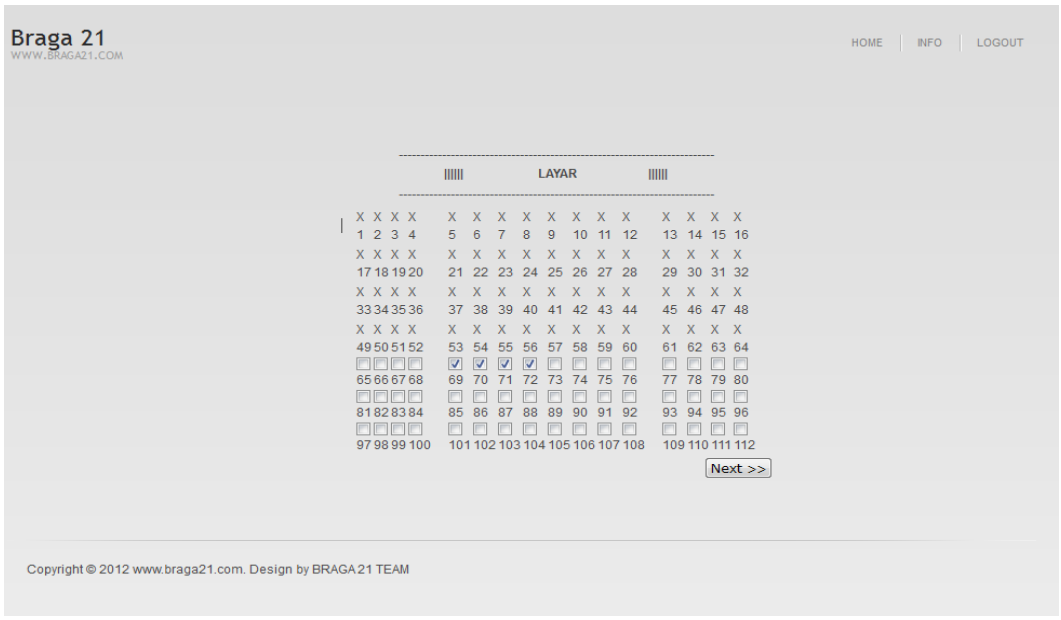
Perang prangan

JADWAL TAYANG FILM

Nama Theater	Tanggal Tayang	Jam Tayang	Kursi Tersedia	Harga Tiket	
Theater 3	2012-07-11	11:00:00	104	30000	pesan
Theater 3	2012-07-11	14:00:00	112	30000	pesan
Theater 3	2012-07-11	17:00:00	112	30000	pesan
Theater 3	0000-00-00	19:00:00	112	30000	pesan
Theater 3	2012-07-11	21:00:00	112	30000	pesan
Theater 3	2012-07-11	23:00:00	112	30000	pesan

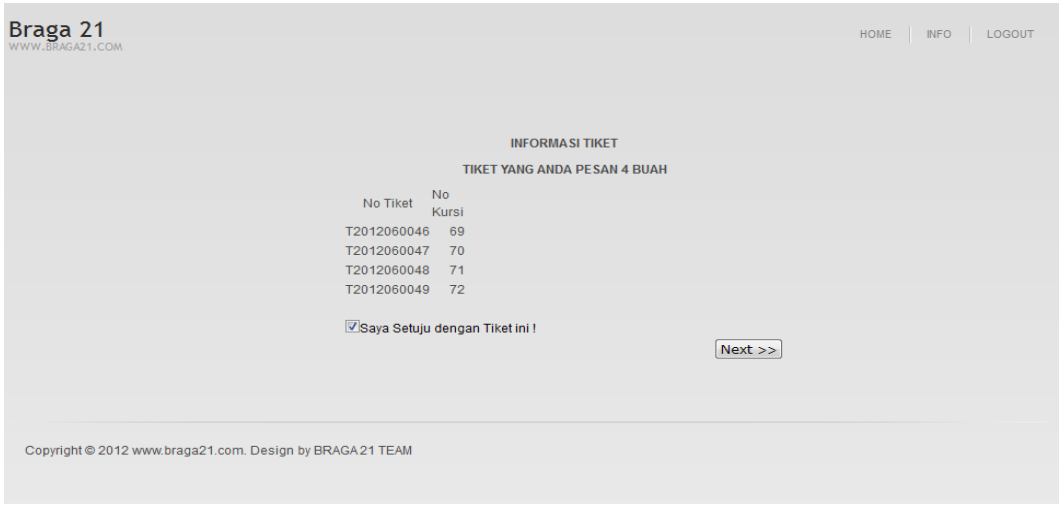
Gambar 5.7 pemesanan tiket

Setelah muncul form informasi jika anda akan memesan film tersebut maka di ujung kanan tiap kolom terdapat tulisan pesan lalu klik sesuai dengan jam tayang yang anda inginkan, setelah itu anda akan diperlihatkan kursi penonton, seperti gambar di bawah ini :



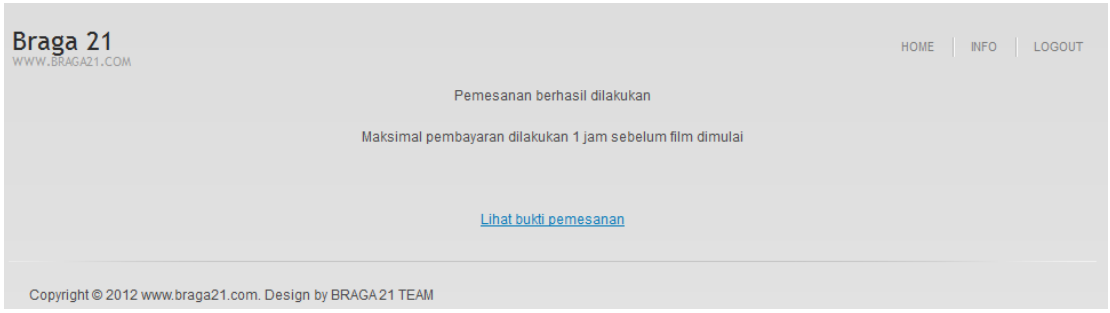
Gambar 5.8 Pemesanan kursi

Setelah itu pilih kursi yang mau dipesan lalu klik next, setelah di klik maka akan muncul form persetujuan. Seperti gambar di bawah ini :



Gambar 5.9 Informasi tiket yang dipesan

Setelah muncul form informasi persetujuan lalu centang persetujuan pemesanan tiket lalu klik next.



Gambar 5.10 Informasi pemesanan

Lalu akan muncul informasi maksimal pembayaran paling telat. Ada juga tombol unuk mencetak bukti pemesanan.



Braga 21 www.braga21.com

BUKTI PEMESANAN

Status Tiket : dipesan

Nomor Tiket : T2012060050
No Mainfest : 2012-06-23/13:00:00/Godzilla 2
Id Pengguna : USB-07

No	Nomor Tiket	Nomor Kursi	Harga Tiket
1.	T2012060050	79	25000

Total Bayar : Rp. 25000

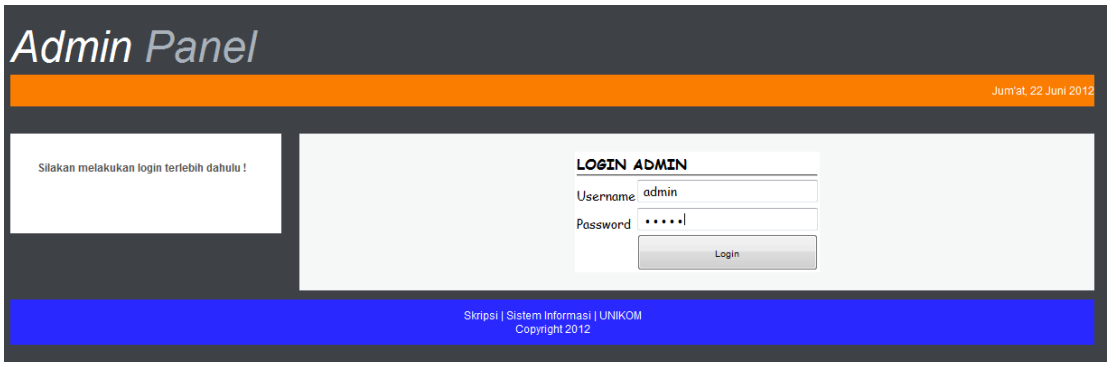
Dimohon untuk datang 1 Jam sebelum Film Di Mulai
Jika anda mengalami keterlambatan, tiket tidak berlaku lagi
Jagalah barang bawaan anda

Keterangan :
1. Bukti Pemesanan ini, dibawa pada saat melakukan pembayaran

Gambar 5.11 Bukti pemesanan

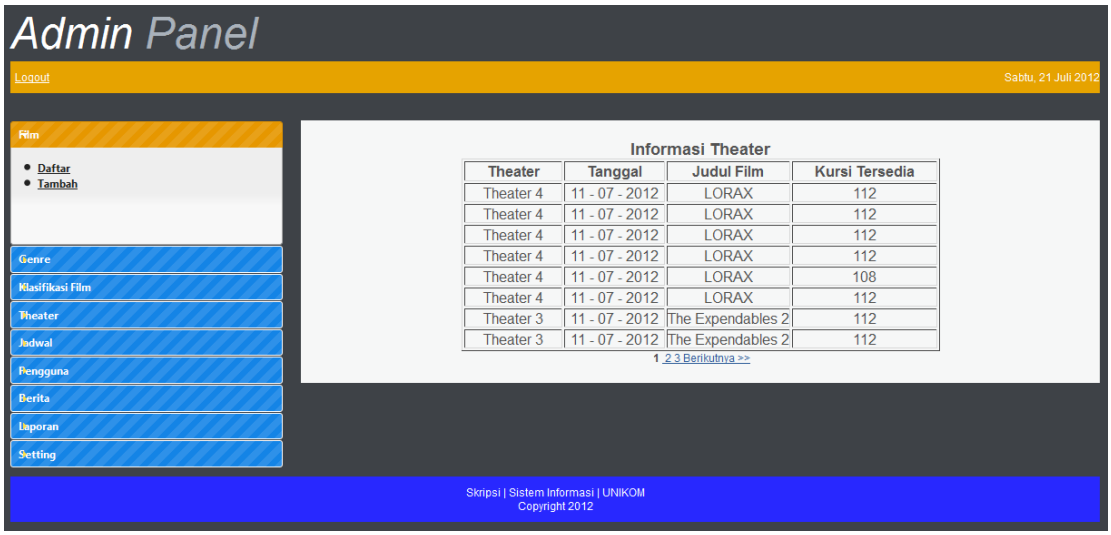
5.1.5.2. Menu Admin

Ini adalah tampilan halaman login admin



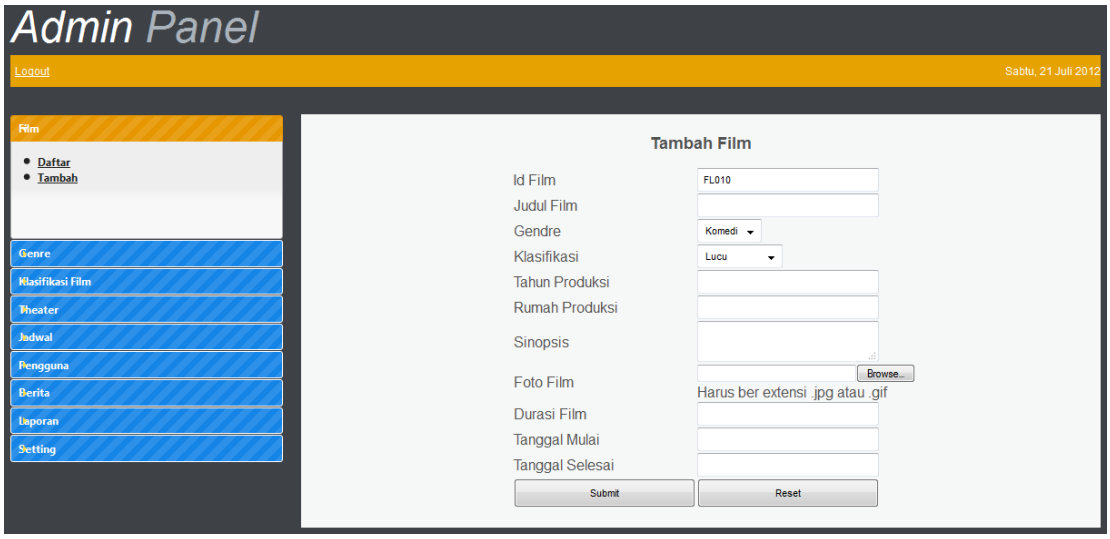
Gambar 5.12 Admin panel

Tampilan halaman admin



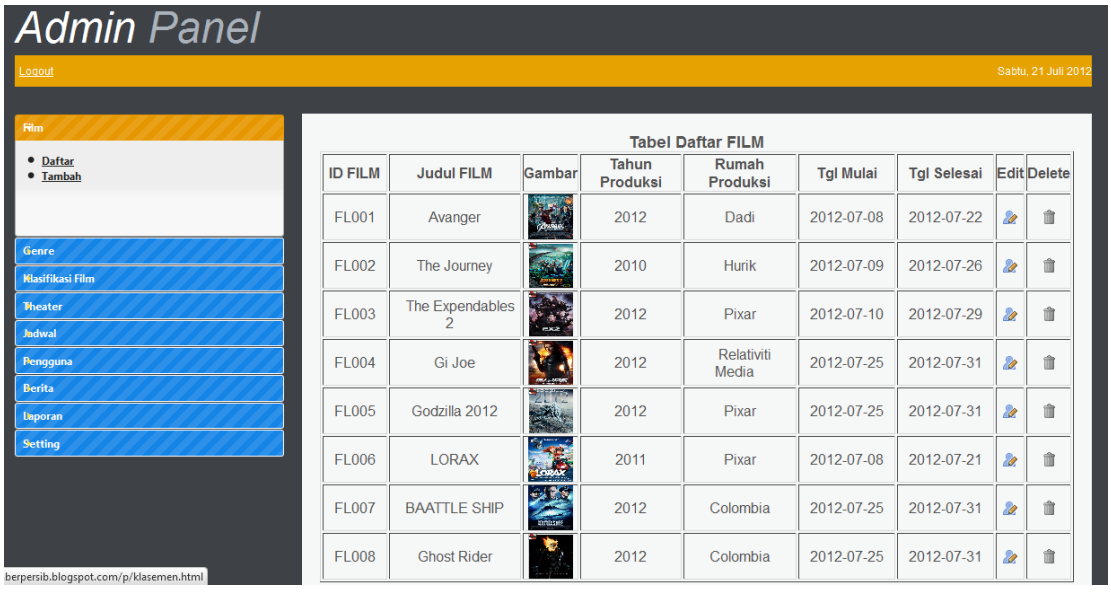
Gambar 5.13 Halaman admin

Tampilan halaman tambah film



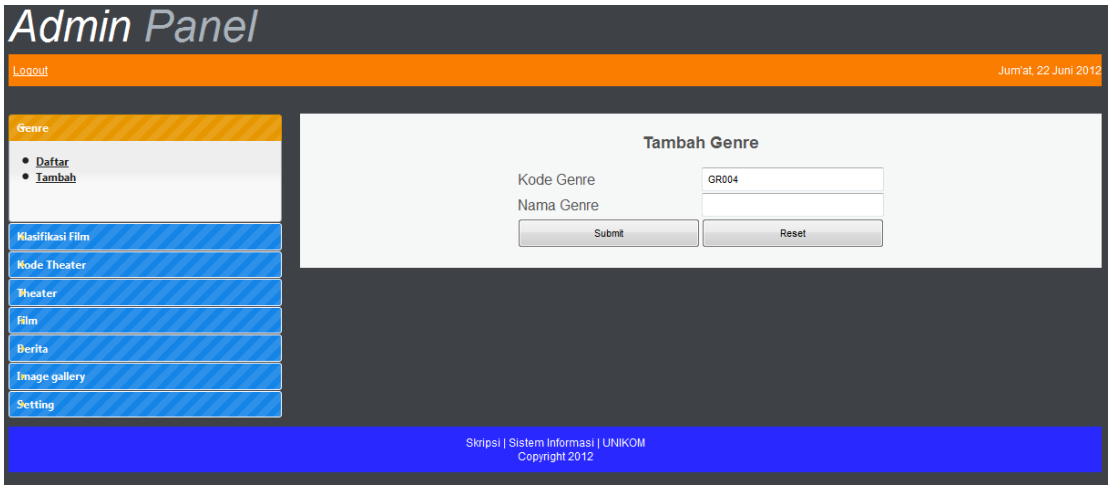
Gambar 5.14 Tambah film

Tampilan halaman tabel film



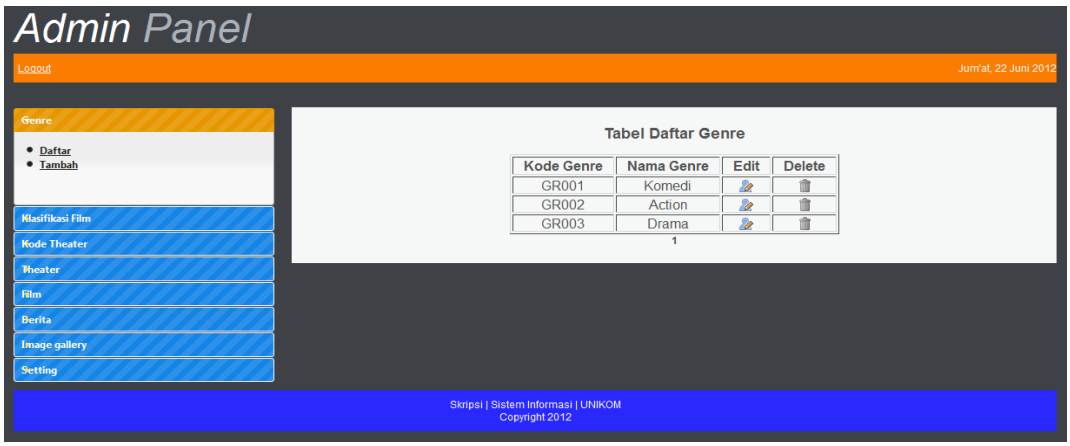
Gambar 5.15 Tabel daftar film

Tampilan halaman tambah genre



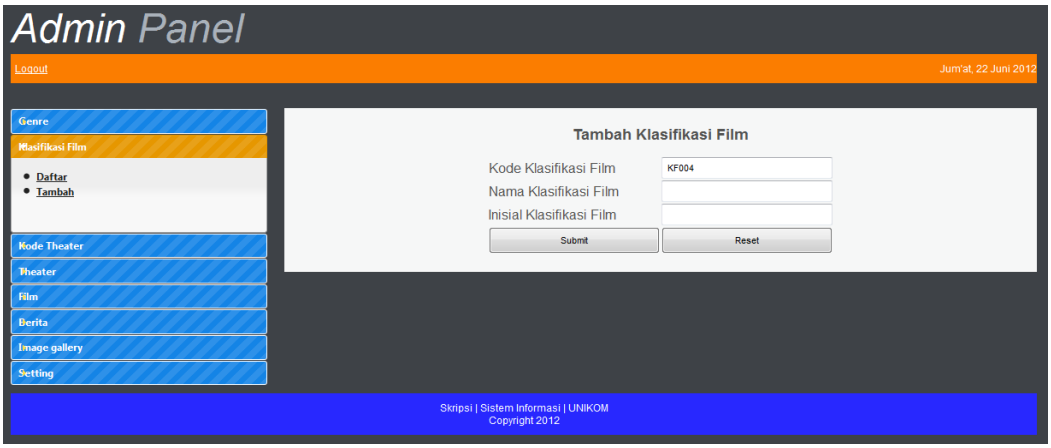
Gambar 5.16 Tambah genre

Tampilan halaman tabel genre



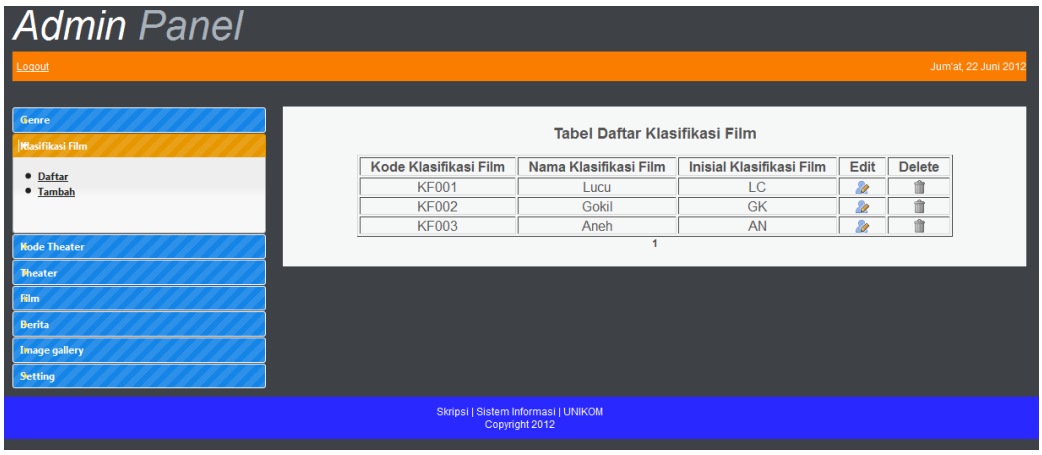
Gambar 5.17 tabel daftar genre

Tampilan halaman tambah klasifikasi film



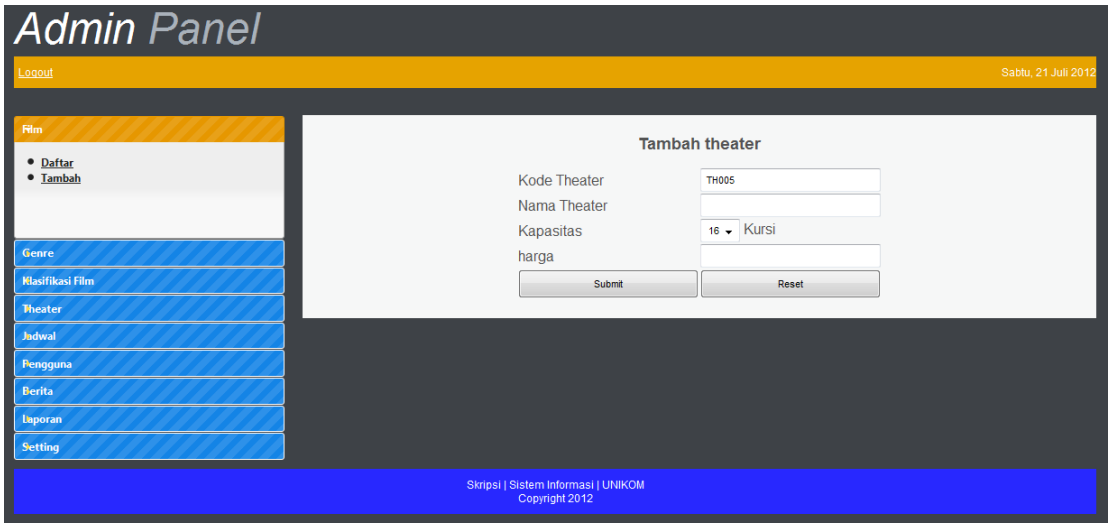
Gambar 5.18 Tambah klasifikasi film

Tampilan halaman tabel klasifikasi film



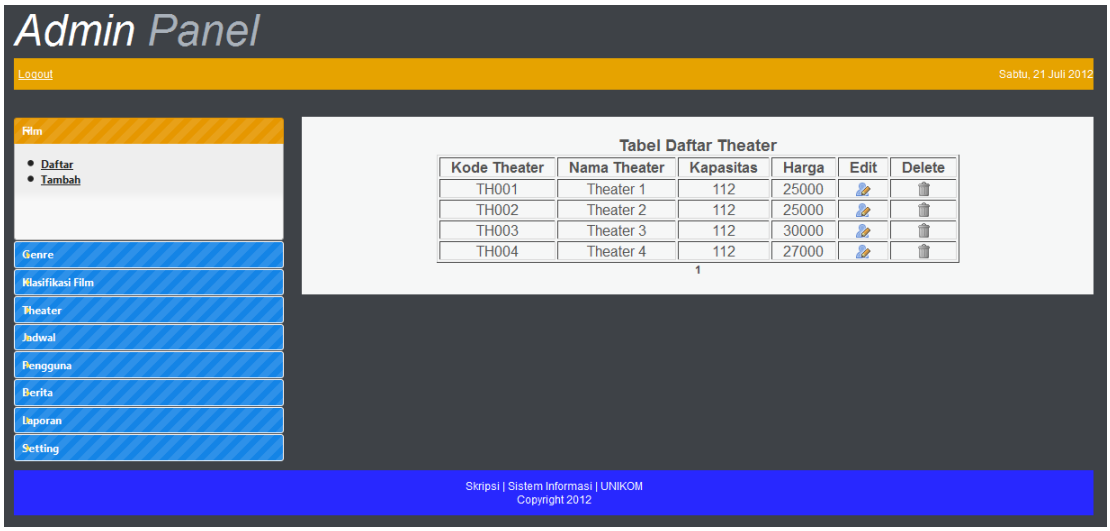
Gambar 5.19 Tabel daftar klasifikasi film

Tampilan halaman tambah theater



Gambar 5.20 Tambah theater

Tampilan halaman tabel theater



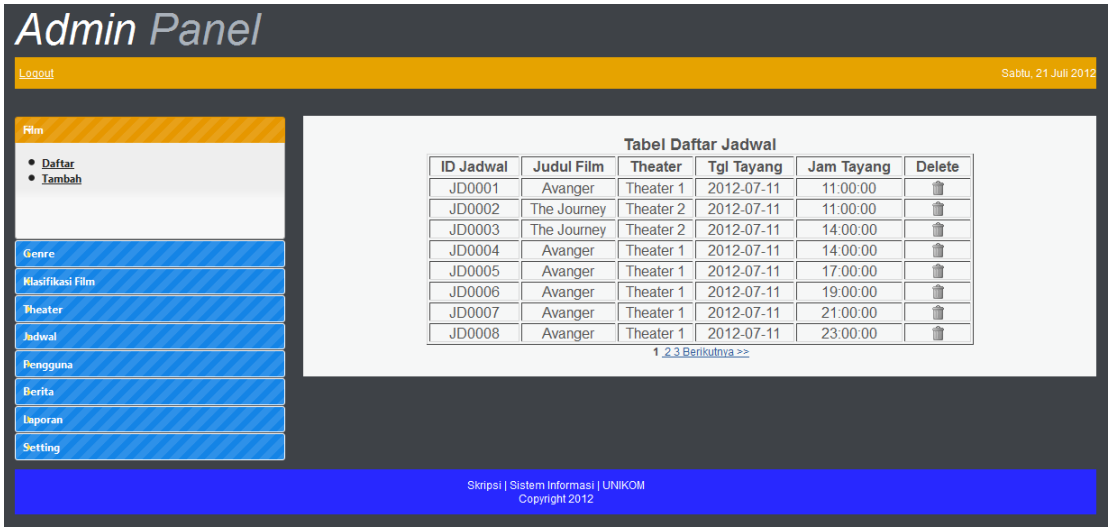
Gambar 5.21 Tabel daftar theater

Tampilan halaman tambah jadwal



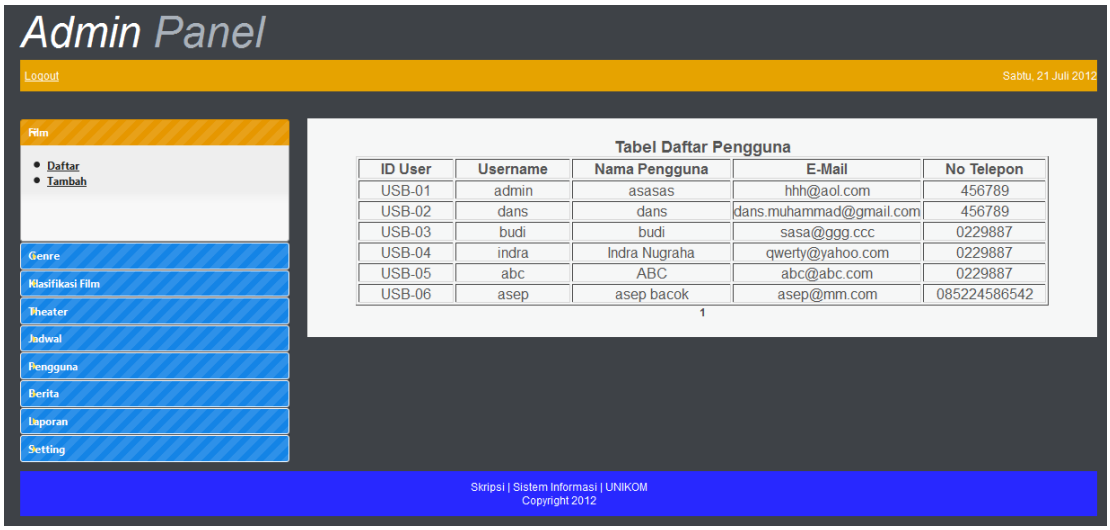
Gambar 2.22 Tambah theater

Tampilan halaman tabel daftar jadwal



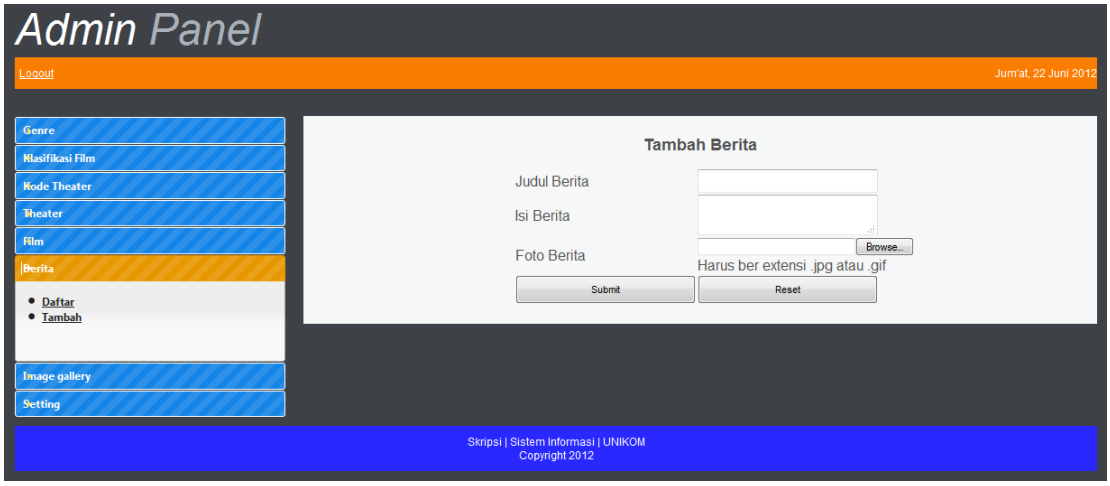
Gambar 2.23 Tabel Daftar jadwal

Tampilan halaman daftar pengguna



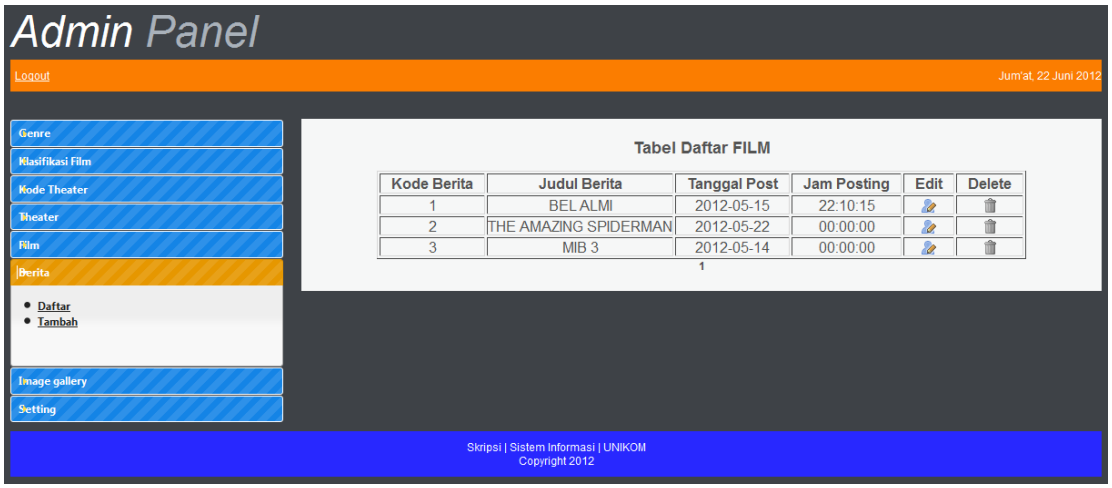
Gambar 2.24 Tabel daftar pengguna

Tampilan halaman tambah berita



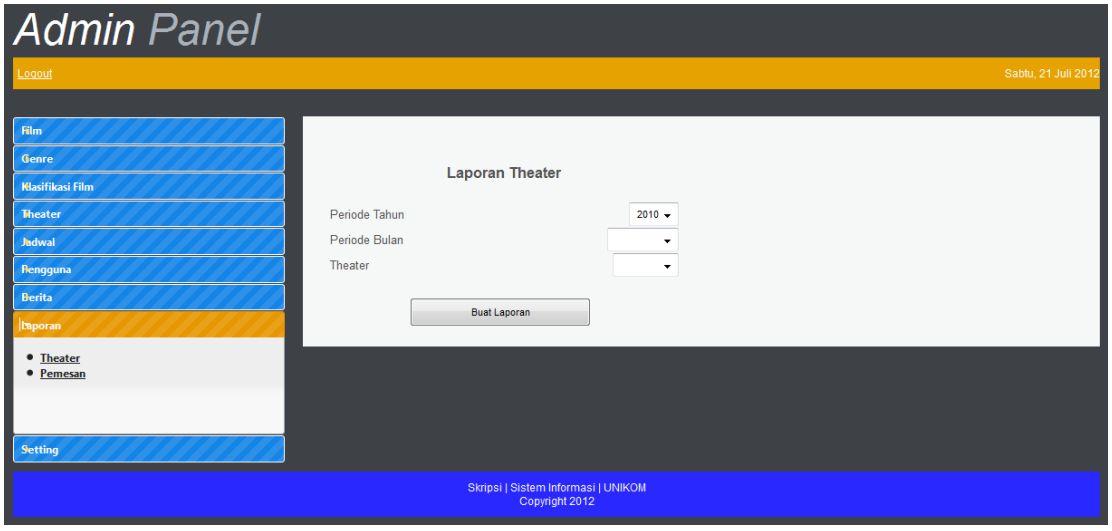
Gambar 5.25 Tambah berita

Tampilan halaman tabel berita



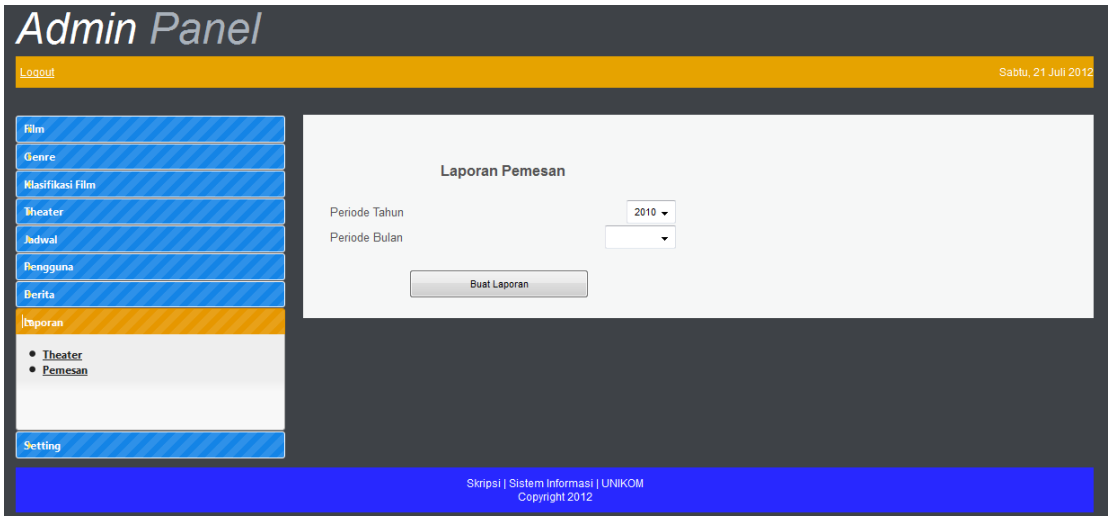
Gambar 5.26 Tabel daftar berita

Tampilan halaman pembuatan laporan theater



Gambar 2.27 Laporan Theater

Tampilan halaman pembuatan laporan pemesanan



Gambar 2.28 Laporan pemesanan

5.1.6. Implementasi Instalasi Program

Cara instalasi program :

1. Instalasi program **WAMP Server** yang terdapat dalam folder pada CD
2. Copy dan Paste folder **"bioskop**" ke direktori **C:/wamp/www/**. Alamat direktori ini akan muncul setelah WAMP Server berhasil diinstalasi.
3. Aktifkan Browser, dan masuk ke alamat phpmyadmin: **"http//127.0.0.1/phpmyadmin"** (ditulis tanpa kutip).
4. Setelah halaman phpmyadmin muncul di layar komputer, buat sebuah database baru (create new database) dengan nama **"bioskop"** (ditulis tanpa kutip).

5. Setelah database berhasil dibuat, pilih menutab **import** dan browse file **bioskop.sql** yang ada dalam CD.
6. Setelah proses import berhasil maka Program Aplikasi siap digunakan.
7. Untuk membuka halaman user masuk ke alamat **http//localhost/bioskop**.
8. Untuk membuka halaman admin masuk ke alamat **http// localhost/bioskop /Admin**.

5.1.7. Penggunaan Program

Sistem informasi pemesanan tiket online di bioskop braga 21 ini merupakan aplikasi yang dirancang untuk mempermudah user dalam melakukan pemesanan tiket kapanpun dan dimanapun tidak harus mengantri di loket.

Adapun tahap-tahap dalam melakukan pemesanan adalah sebagai berikut

1. User melakukan login terlebih dahulu
2. Jika sudah masuk halaman pemesanan User memilih film yang diinginkan
3. Setelah memilih maka user akan dihadapkan dengan synopsis ,jumlah kursi yang tersedia dan pesan film.
4. Jika kursi masih tersedia lalu pilih pesan film lalu user harus memilih waktu dan jam tayang film tersebut.
5. Setelah itu user harus memesan denah kursi yang diinginkan.
6. Lalu user harus cetak bukti pemesanan supaya bisa ditukarkan dengan tiket di bioskopnya.

5.2. Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak (software). Tujuan dari pengujian adalah untuk menjamin perangkat lunak yang di bangun memilki kualitas yang handal, yaitu mampu mempresentasikan kajian pokok dari spesifikasi, analisis, perancangan dan pengkodean dari perangkat lunak itu sendiri. Dalam analisis dan perancangan ini, penulis menggunakan 3 (tiga) faktor pengujian yang dilakukan secara internal terhadap sistem informasi tiket online di bioskop ini.

5.2.1. Rencana Pengujian

Rencana Pengujian sistem informasi pemesanan tiket online di bioskop braga 21 adalah sebagai berikut :

Item Pengujian	Deskripsi	Jenis Pengujian
Proses Registrasi User	Memasukan data user	Black Box
Proses Login User	Memeriksa login user	Black Box
Proses Pemesanan	Memilih waktu dan jam tayang	Black Box

Tabel 5.1 Rencana Pengujian

5.2.2. Kasus dan Hasil Pengujian

1. Pengujian Registrasi User

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Semua field diisi dengan benar. Klik tombol '*submit*'.	Data tersimpan dalam database	Data tersimpan ke dalam database.	[ ] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Semua field kosong. Klik tombol '*submit'*.	Data tidak tersimpan, program menampilkan pesan kesalahan	Data tidak tersimpan	[ ] Diterima [ ] Ditolak
Field '*user* ' tidak diisi. Klik tombol '*submit'*.	Data tidak tersimpan, program menampilkan pesan kesalahan	Data tidak tersimpan	[ ] Diterima [ ] Ditolak
Field '*password*' tidak diisi. Klik tombol '*submit'*.	Data tidak tersimpan, program menampilkan pesan kesalahan	Data tidak tersimpan	[ ] Diterima [ ] Ditolak
Field '*konfirmasi password*' tidak diisi.	Data tidak tersimpan, program	Data tidak tersimpan	[ ] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Klik tombol '*submit'*.	menampilkan pesan kesalahan		
Field '*nama'* tidak diisi. Klik tombol '*submit'*.	Data tidak tersimpan, program menampilkan pesan kesalahan	Data tidak tersimpan	[ ] Diterima [ ] Ditolak
Field '*tanggal lahir'* tidak diisi. Klik tombol '*submit'*.	Data tidak tersimpan, program menampilkan pesan kesalahan	Data tidak tersimpan	[ ] Diterima [ ] Ditolak
Field '*umur'* tidak diisi. Klik tombol '*submit'*.	Data tidak tersimpan, program menampilkan pesan kesalahan	Data tidak tersimpan	[ ] Diterima [ ] Ditolak
Field '*alamat'* tidak diisi. Klik tombol '*submit'*.	Data tidak tersimpan, program menampilkan pesan kesalahan	Data tidak tersimpan	[ ] Diterima [ ] Ditolak
Field '*kota'* tidak diisi. Klik tombol '*submit'*.	Data tidak tersimpan, program menampilkan pesan kesalahan	Data tidak tersimpan	[ ] Diterima [ ] Ditolak
Field '*email'* tidak diisi. Klik tombol '*submit'*	Data tidak tersimpan, tampil pesan kesalahan	Data tidak tersimpan	[ ] Diterima [ ] Ditolak

Kasus dan Hasil Uji (Data Salah)			
Field '*no telp'* tidak diisi. Klik tombol '*submit'*.	Data tidak tersimpan, program menampilkan pesan kesalahan	Data tidak tersimpan	[ ] Diterima [ ] Ditolak

Tabel 5.2 Pengujian registrasi user

2. Proses Login Admin

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data login admin Username : admin Password : admin	Login berhasil Status login	Login berhasil Status login	[ ] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Login admin salah Username : xxxxxxxx Password : xxxxxxxxxx	Tidak dapat login	Admin tidak dapat login	[ ] Diterima [ ] Ditolak

Tabel 5.3 Pengujian login admin

3. Proses Pemesanan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Data pemesanan diisi dan sesuai	Pemesanan berhasil	Pemesanan berhasil	[ ] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data pemesanan tidak diisi dan tidak sesuai	Pemesanan tidak berhasil	Pemesanan tidak berhasil	[ ] Diterima [ ] Ditolak

Tabel 5.4 Pengujian proses pemesanan

5.2.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus uji *sample* di atas dapat ditarik kesimpulan bahwa sistem informasi ini bebas dari kesalahan sintaks dan secara fungsional mengeluarkan hasil sesuai dengan yang diharapkan.