

## Lampiran 1: Pembuatan Database dan Tabel dengan SQL Server 7.0

### 1. Pembuatan Database

*Database* adalah sekumpulan tabel, view, indeks, trigger, prosedur, dan objek-objek lain. SQL Server 7.0 memiliki kemampuan untuk memelihara beberapa *database*. Membuat *database*, terutama struktur dasarnya adalah pekerjaan yang sangat sederhana pada SQL Server 7.0 karena SQL Server sudah menggunakan antarmuka grafis. Membuat sebuah *database* meliputi menyediakan karakteristik fisik dari *database*, seperti ukuran, tingkat kenaikan, nama, identifikasi pemilik, serta identifikasi grup. Secara umum, ada 3 buah *file* yang membentuk sebuah *database*, yaitu :

#### 1. File Primer

File ini mengandung informasi yang dibutuhkan untuk menu *shortcut-load* dan melakukan inisialisasi *database*, serta penyimpanan data. File primer ini merupakan file *default* untuk setiap *database* yang kita buat.

#### 2. File Sekunder

File sekunder hanya akan ada apabila file primer tidak cukup besar untuk memelihara semua file data. Bergantung pada ukuran dari *database*, mungkin kita memerlukan beberapa file sekunder ini.

#### 3. File Log

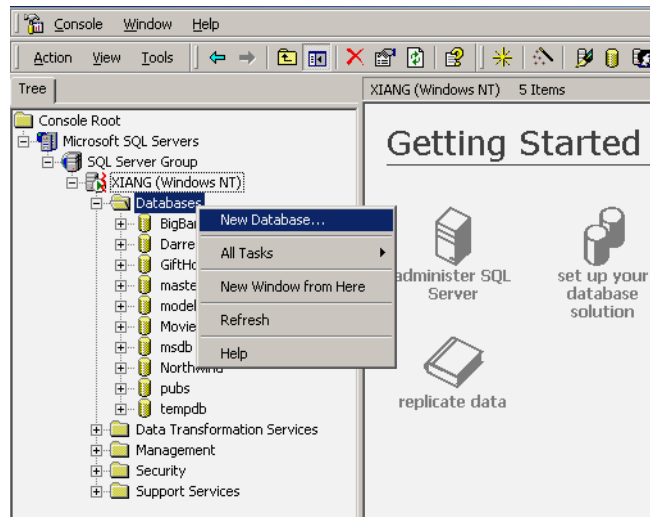
File log digunakan untuk memulihkan *database*. Setidaknya ada satu file log dalam setiap *database*, dan file log yang kedua selalu bisa dibuat. Ukuran minimal dari file log adalah 512 kB.

*Database* dapat dibuat secara manual dengan menggunakan Enterprise Manager, dengan menggunakan Create Database Wizard, maupun dengan menggunakan bahasa Transact-SQL.

#### 1. Membuat *Database* secara Manual

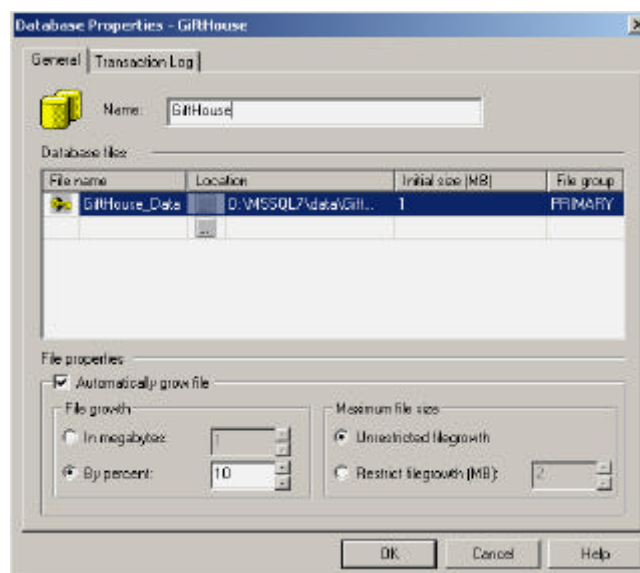
Untuk membuat *database* secara manual, langkah pertama yang harus dilakukan adalah meng-klik *icon* Microsoft SQL Server pada Enterprise Manager, lalu membuka *server*. Klik kanan pada *folder* Database lalu pilih New Database, seperti ditampilkan pada Gambar 1. berikut ini:

## Lampiran 1: Pembuatan Database dan Tabel dengan SQL Server 7.0 (sambungan)



Gambar 1. Membuat Database dengan Enterprise Manager

Ketikkan nama untuk *database* yang akan kita buat pada kotak dialog yang muncul, seperti terlihat pada Gambar 2. Pada contoh ini, ketikkanlah GiftHouse.

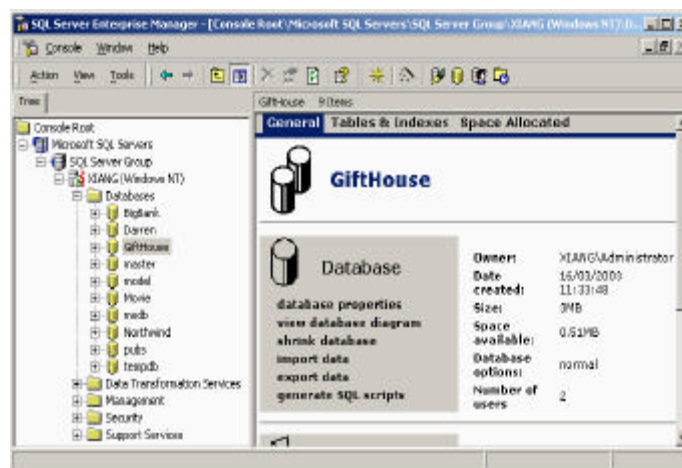


Gambar 2. Database Properties

Database primer dan file-file log transaksi akan dibuat dengan menggunakan nama yang telah kita tentukan sebagai awalnya. Pada contoh yang kita buat, akan terbentuk GiftHouse.mdf dan GiftHouse.ldf. Ukuran awal dari file utama

## Lampiran 1: Pembuatan Database dan Tabel dengan SQL Server 7.0 (sambungan)

dan file log didasarkan pada *database model*. Pada Tab Option, pilihan DBO Use Only memberikan hak akses hanya kepada pembuat *database*, sedangkan pilihan Single User hanya akan memberikan hak akses kepada satu *user* saja. Untuk membuat *database*, kita dapat meng-klik tombol OK. *Database* akan terbentuk dengan menggunakan nilai-nilai yang telah kita tentukan. Untuk lebih jelasnya, lihat pada Gambar 3. di bawah ini:

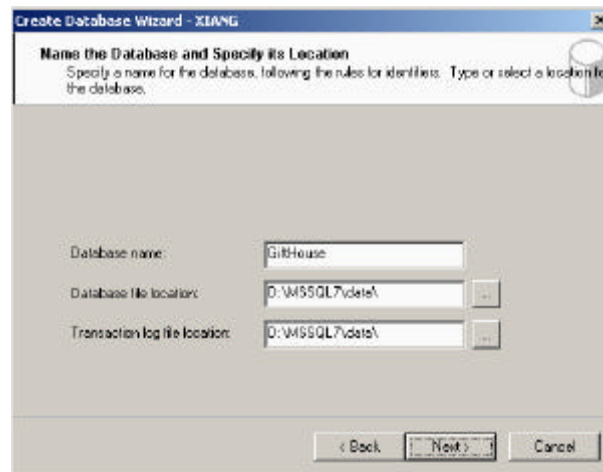


Gambar 3. Database GiftHouse

### 2. Membuat *Database* dengan Wizard

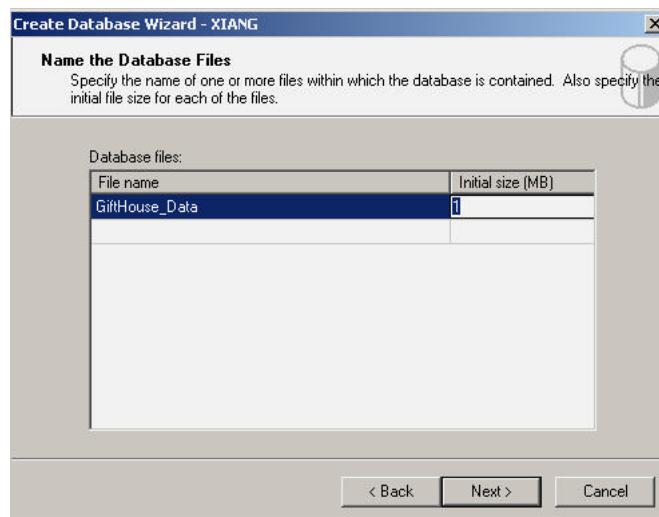
Database Wizard dapat diaktifkan melalui perintah Tools|Wizard|Database. Klik ganda pada Create Database Wizard. Selain itu, Database Wizard juga dapat diaktifkan dengan mengklik kanan pada folder Database dan memilih menu Tool|Wizard. Layar pertama dari Wizard hanya akan berupa data informasi. Tekan Next untuk melanjutkan ke tahap berikutnya. Pada layar kedua, kita diminta untuk menyetikkan nama *database* dan menentukan lokasi penyimpanan, seperti terlihat pada Gambar 4.

## Lampiran 1: Pembuatan Database dan Tabel dengan SQL Server 7.0 (sambungan)



Gambar 4. Membuat Database dengan Wizard

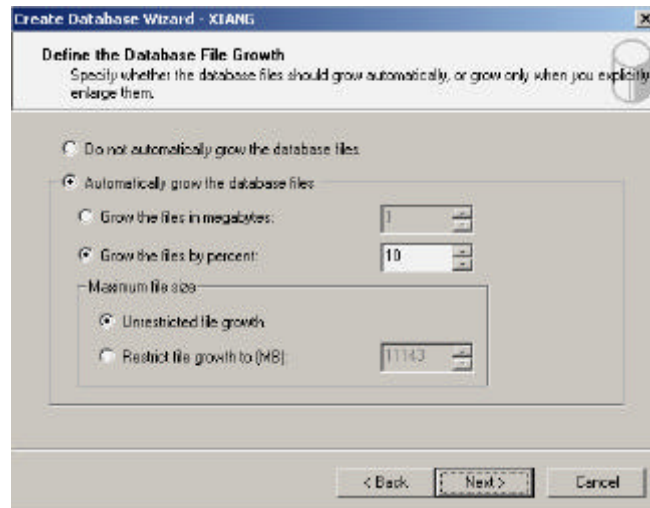
Pada langkah berikutnya, kita dapat menerima nilai yang diberikan untuk nama file *database* dan ukurannya, atau ketikkan ukuran awal untuk file *database*. Lihat Gambar 5. untuk lebih jelasnya.



Gambar 5. Konfigurasi Nama File dan Ukurannya

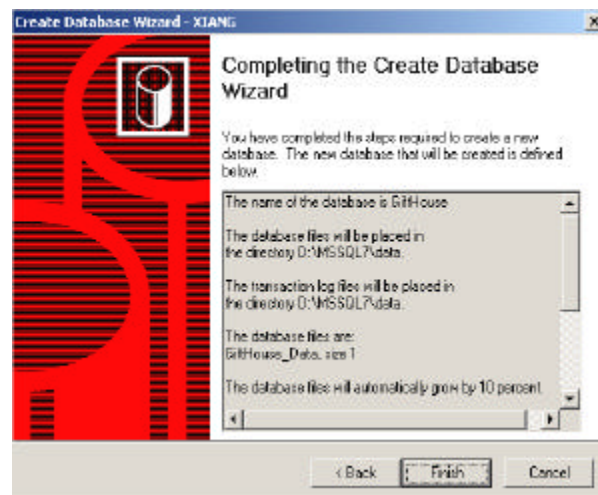
Setelah menekan Next, kita akan diminta untuk mengetikkan data mengenai pertumbuhan file *database*, seperti terlihat pada Gambar 6. Secara *default*, *database* secara otomatis akan tumbuh dalam kenaikan 10 persen dari ukuran awalnya. Kita dapat mengetikkan data sesuai dengan keinginan kita, atau tetap menerima nilai yang ditampilkan.

## Lampiran 1: Pembuatan Database dan Tabel dengan SQL Server 7.0 (sambungan)



Gambar 6. Konfigurasi Pertumbuhan File

Pada saat tombol Next ditekan, akan muncul *window* baru yang akan menampilkan rangkuman dari karakteristik *database* yang telah kita tentukan pada langkah-langkah sebelumnya, seperti pada Gambar 7. Tombol Finish digunakan untuk menyelesaikan Wizard.



Gambar 7. Rangkuman Karakteristik Database

### 3. Membuat *Database* menggunakan Transact-SQL

Pada SQL Server 7.0, *database* dapat juga dibuat dengan menggunakan perintah SQL CREATE DATABASE. Berikut ini adalah contoh dari perintah untuk membuat *database* dengan nama test1. File primer akan

## Lampiran 1: Pembuatan Database dan Tabel dengan SQL Server 7.0 (sambungan)

bernama test1dat. Ukuran awal dari file primer adalah 20 MB sedangkan ukuran maksimalnya adalah 50 MB, dengan kenaikan pertumbuhan sebesar 5 MB. File log akan memiliki ukuran awal 6 MB dan ukuran maksimal sebesar 20 MB, dengan kenaikan pertumbuhan sebesar 4 MB.

```
USE master
GO
CREATE DATABASE Test1
ON
(
FILENAME = 'c:\mssql7\data\test1dat.mdf',
SIZE = 20MB,
MAXSIZE = 50MB,
FILEGROWTH = 5MB
)
LOG ON
(
NAME = 'test1_log',
FILENAME = 'c:\mssql7\data\test1log.ldf',
SIZE = 6MB,
MAXSIZE = 20MB,
FILEGROWTH = 4MB
)
```

### 2. Pembuatan Tabel

Berikut ini akan dijelaskan beberapa fasilitas yang ada di dalam SQL Server 7.0 untuk membuat, melihat data, dan menghapus tabel serta melakukan manipulasi terhadap sebuah tabel.

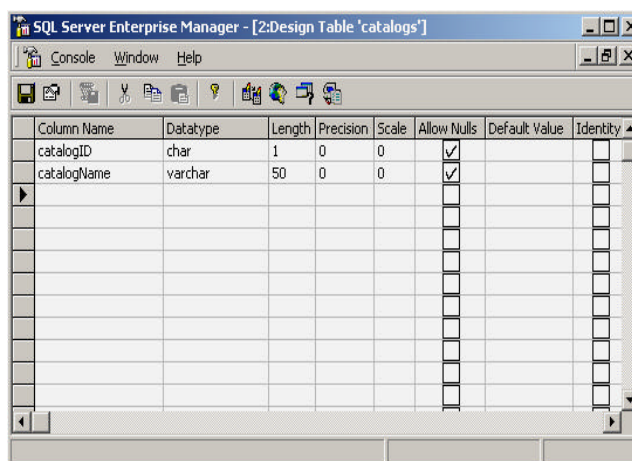
#### 1. Membuat Tabel dengan Enterprise Manager

Kita dapat membuat sebuah tabel melalui Enterprise Manager, yaitu dengan mengklik tanda plus pada Database (*expand*) dan kemudian mengklik kanan pada folder Tables. Pada saat kita klik kanan pada folder Tables, kita

## Lampiran 1: Pembuatan Database dan Tabel dengan SQL Server 7.0 (sambungan)

dapat memilih menu New Tables. Sebuah kotak dialog akan muncul yang berfungsi untuk menentukan nama dari tabel yang akan kita buat.

Setelah kita masukkan nama dari tabel yang akan kita buat dan menekan tombol OK, maka akan muncul sebuah *window* baru yang memungkinkan kita untuk menginputkan karakteristik dari kolom-kolom, seperti dapat kita lihat pada Gambar 8. berikut ini.



Gambar 8. Desain Tabel

Berikut ini adalah penjelasan dari setiap kolom yang ada pada *window* tersebut di atas:

Column Name	Menunjukkan nama untuk kolom atau <i>field</i> .
Datatype	Menunjukkan tipe data yang akan disimpan oleh kolom.
Length	Menunjukkan ukuran dari <i>field</i> .
Precision	Menunjukkan jumlah digit di dalam angka
Scale	Menunjukkan jumlah dari angka pecahan
Allow Nulls	Apabila kolom ini dipilih, SQL Server akan mengijinkan nilai Null di dalam <i>field</i> .
Default Value	Digunakan untuk menentukan nilai <i>default</i> untuk kolom.

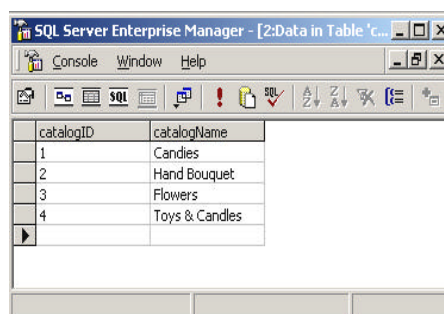
## Lampiran 1: Pembuatan Database dan Tabel dengan SQL Server 7.0 (sambungan)

Identify	Apabila kolom ini dipilih, maka akan menunjukkan bahwa <i>field</i> akan memiliki angka identitas yang secara otomatis dihasilkan.
Identify Seed	Menunjukkan nilai awal untuk <i>record</i> pertama.
Identify Increment	Menunjukkan nilai kenaikan untuk <i>record</i> baru.
IsRowGuid	Membuat isi eksklusif untuk kolom pada tingkat global. Sebuah tabel hanya dapat memiliki satu kolom dengan tipe ini dan harus dibuat dengan tipe data UniqueIdentifier.

Untuk *field* numerik, Length menunjukkan jumlah *byte* yang digunakan untuk menyimpan semua angka. Untuk *field* karakter atau Unicode, Length menunjukkan jumlah karakter, sedangkan untuk *field* Binary, Varbinary, dan Image, Length menunjukkan jumlah *byte*.

### 2. Melihat Data Tabel

Untuk melihat data dari tabel yang telah kita buat, kita dapat mengklik kanan pada tabel yang akan kita buka. Setelah itu kita dapat memilih menu Open Table| Return all rows untuk melihat semua baris dari tabel, atau Open Table| Return Top untuk menentukan banyak baris yang ingin dilihat. Sebuah *window* baru akan terbuka, yang akan menampilkan *record-record* pada tabel, seperti terlihat pada Gambar 9.



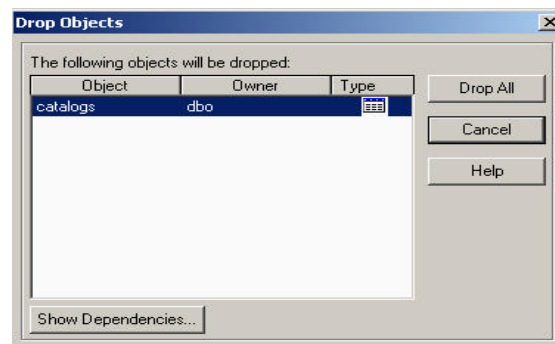
catalogID	catalogName
1	Candies
2	Hand Bouquet
3	Flowers
4	Toys & Candles

Gambar 9. Data pada Sebuah Tabel

## Lampiran 1: Pembuatan Database dan Tabel dengan SQL Server 7.0 (sambungan)

### 3. Menghapus Tabel

Untuk menghapus tabel, kita dapat mengklik kanan pada nama tabel yang akan kita hapus dan memilih menu Delete. Kotak dialog Drop Objects akan muncul, seperti pada Gambar 10. Untuk menghapus keseluruhan tabel, kita dapat menekan tombol Drop All. Untuk melihat semua ketergantungan yang dimiliki oleh tabel, kita dapat menekan tombol Show Dependencies. Jika terdapat ketergantungan, sebuah kotak dialog akan muncul. Untuk kembali ke layar sebelumnya, kita dapat menekan tombol Close.



Gambar 10. Drop Table

### 4. Manipulasi Tabel

Untuk melakukan perubahan-perubahan terhadap tabel yang telah kita buat, kita dapat memilih menu Design setelah kita mengklik kanan pada tabel yang akan kita hapus, yang berada pada panel sebelah kanan. Akan muncul *window* Design Table, seperti terlihat pada Gambar 8. di atas. Kita dapat mulai melakukan perubahan-perubahan yang diinginkan, termasuk penghapusan *field* yang tidak dibutuhkan, dan kemudian menekan tombol bergambar disket yang terletak pada pojok kanan atas dari *window* untuk menyimpan perubahan-perubahan yang kita lakukan.

## Lampiran 2 : Stored Procedures GiftHouse

### 1. sp\_ViewCategory

```
CREATE PROCEDURE sp_ViewCategory
AS
SELECT * FROM Catalogs
ORDER BY CatalogName
```

### 2. sp\_ViewProductsByCategory

```
CREATE PROCEDURE sp_ViewProductsByCategory
(
    @CatalogID_1 char(1)
)
AS
SELECT * FROM Products
WHERE CatalogID = @CatalogID_1
ORDER BY ProductName
```

### 3. sp\_ViewMembers

```
CREATE PROCEDURE sp_ViewMembers
AS
SELECT * FROM Member
ORDER BY UserName
```

### 4. sp\_CheckItemsOrdered

```
CREATE PROCEDURE sp_CheckItemsOrdered
(
    @orderId_1 int
)
AS
SELECT * FROM ItemsOrdered
WHERE orderId = @orderId_1
```

### 5. sp\_CheckLogers

```
CREATE PROCEDURE sp_CheckLogers
(
    @Username_1 varchar(50),
    @Password_1 varchar(50)
)
AS
SELECT * FROM Member
WHERE Username = @Username_1
AND Password = @Password_1
```

## Lampiran 2 : Stored Procedures GiftHouse (sambungan)

### 6. sp\_CheckOrders

```
CREATE PROCEDURE sp_CheckOrders
AS
SELECT * FROM Orders
ORDER BY OrderId
```

### 7. sp\_DeleteOrder

```
CREATE PROCEDURE sp_DeleteOrder
(
  @orderId_1 int
)
AS
DELETE FROM Orders
WHERE orderId = @orderId_1
```

### 8. sp\_DeleteItems

```
CREATE PROCEDURE sp_Deleteltems
(
  @orderId_1 int
)
AS
DELETE FROM ItemsOrdered
WHERE orderId = @orderId_1
```

### 9. sp\_GetCurrentOrder

```
CREATE PROCEDURE sp_GetCurrentOrder
AS
SELECT MAX(OrderID) As Maximum
FROM ORDERS
```

### 10. sp\_GetCurrPoints

```
CREATE PROCEDURE sp_GetCurrPoints
(
  @username_1 varchar(50)
)
AS
SELECT points FROM Member
WHERE username = @username_1
```

## Lampiran 2 : Stored Procedures GiftHouse (sambungan)

### 11. sp\_InsertItemsOrdered

```
CREATE PROCEDURE sp_ViewCategory
AS
SELECT * FROM Catalogs
ORDER BY CatalogName
```

### 12. sp\_InsertOrder

```
CREATE PROCEDURE sp_InsertOrder
(
    @orderID_1 int,
    @orderDate_1 smalldatetime,
    @fname_1 varchar(20),
    @lname_1 varchar(30),
    @address_1 varchar(150),
    @phone_1 varchar(15),
    @sfName_1 varchar(20),
    @slName_1 varchar(30),
    @sAddress_1 varchar(150),
    @Total_1 int
)
AS
INSERT INTO Orders
(
    orderID,
    orderDate,
    fname,
    lname,
    address,
    phone,
    sfName,
    slName,
    sAddress,
    Total
)
VALUES
(
    @orderID_1,
    @orderDate_1,
    @fname_1,
    @lname_1,
    @address_1,
    @phone_1,
    @sfName_1,
    @slName_1,
    @sAddress_1,
    @Total_1
)
```

**Lampiran 2 : Stored Procedures GiftHouse (sambungan)**

## 13. sp\_ProductDescription

```
CREATE PROCEDURE sp_ProductDescription
(
    @ProductID_1 int
)

AS
SELECT * FROM Products
WHERE ProductID = @ProductID_1
```

## 14. sp\_UpdateMember

```
CREATE Procedure sp_UpdateMember
(
    @username_1 varchar(50),
    @points_1 int
)

AS UPDATE member

SET
    points = @points_1

WHERE
(
    username = @username_1
)
```

### Lampiran 3 : Stored Procedures dari MoviePoll

#### 1. sp\_ViewCategory

```
CREATE PROCEDURE sp_ViewCategory
AS
SELECT * FROM Catalogs
ORDER BY CatalogName
```

#### 2. sp\_ViewProductsByCategory

```
CREATE PROCEDURE sp_ViewProductsByCategory
(
  @CatalogID_1 char(1)
)
AS
SELECT * FROM movies
WHERE CatalogID = @CatalogID_1
ORDER BY RateImg DESC
```

#### 3. sp\_ProductDescription

```
CREATE PROCEDURE sp_ProductDescription
(
  @productID_1 int
)
AS
SELECT * FROM movies
WHERE movieID = @productID_1
```

#### 4. sp\_RateOfTheMonth

```
CREATE PROCEDURE sp_RateOfTheMonth
(
  @maxRate_1 int
)
AS
SELECT *
FROM Movies
WHERE RateImg = @maxRate_1
```

### Lampiran 3 : Stored Procedures dari MoviePoll (sambungan)

#### 5. sp\_CheckLogers

```
CREATE PROCEDURE sp_CheckLogers
(
  @Username_1 varchar(50),
  @Password_1 varchar(50)
)

AS
SELECT * FROM Member

WHERE Username = @Username_1
      AND Password = @Password_1
```

#### 6. sp\_GetCurrPoller

```
CREATE PROCEDURE sp_GetCurrPoller
(
  @movieID_1 int,
  @value_1 int
)

AS
SELECT poller FROM PollResult

WHERE movieID = @movieID_1
      AND value = @value_1
```

#### 7. sp\_GetMaxRating

```
CREATE PROCEDURE sp_GetMaxRating

AS

SELECT Max(RateImg) As Maksimum

FROM Movies
```

#### 8. sp\_GetMovieRating

```
CREATE PROCEDURE sp_GetMovieRating
(
  @movieID_1 int
)

AS
SELECT * FROM PollResult

WHERE movieID = @movieID_1
```

### Lampiran 3 : Stored Procedures dari MoviePoll (sambungan)

#### 9. sp\_InsertNewMember

```

CREATE PROCEDURE sp_InsertNewMember
(
  @username_1 varchar(20),
  @password_1 varchar(20),
  @fName_1 varchar(15),
  @lName_1 varchar(25),
  @Address_1 varchar(100),
  @phone_1 varchar(10),
  @poll_1 char(1)
)

AS
INSERT INTO Member
(
  username,
  password,
  fName,
  lName,
  Address,
  phone,
  Poll
)
VALUES
(
  @username_1,
  @password_1,
  @fName_1,
  @lName_1,
  @Address_1,
  @phone_1,
  @poll_1
)

```

#### 10. sp\_InsertPollResult

```

CREATE PROCEDURE sp_InsertPollResult
(
  @movieID_1 int,
  @value_1 int,
  @poller_1 int
)

AS
INSERT INTO PollResult
(
  movieID,
  value,
  poller
)
VALUES
(
  @movieID_1,
  @value_1,
  @poller_1
)

```

### Lampiran 3 : Stored Procedures dari MoviePoll (sambungan)

#### 11. sp\_UpdateMember

```
CREATE PROCEDURE sp_UpdateMember
(
    @username_1 varchar(50)
)
AS UPDATE Member
SET
    poll = 'Y'
WHERE
(
    username = @username_1
)
```

#### 12. sp\_UpdateMovies

```
CREATE PROCEDURE sp_UpdateMovies
(
    @movieID_1 int,
    @movieRating_1 int,
    @Ratelmg_1 int
)
AS UPDATE movies
SET
    movieRating = @movieRating_1,
    Ratelmg = @Ratelmg_1
WHERE
(
    movieid = @movieid_1
)
```

#### 13. sp\_UpdatePollResult

```
CREATE PROCEDURE sp_UpdatePollResult
(
    @movieID_1 int,
    @value_1 int,
    @poller_1 int
)
AS UPDATE PollResult
SET    poller = @poller_1
WHERE
(
    movieid = @movieid_1
AND   value = @value_1
)
```

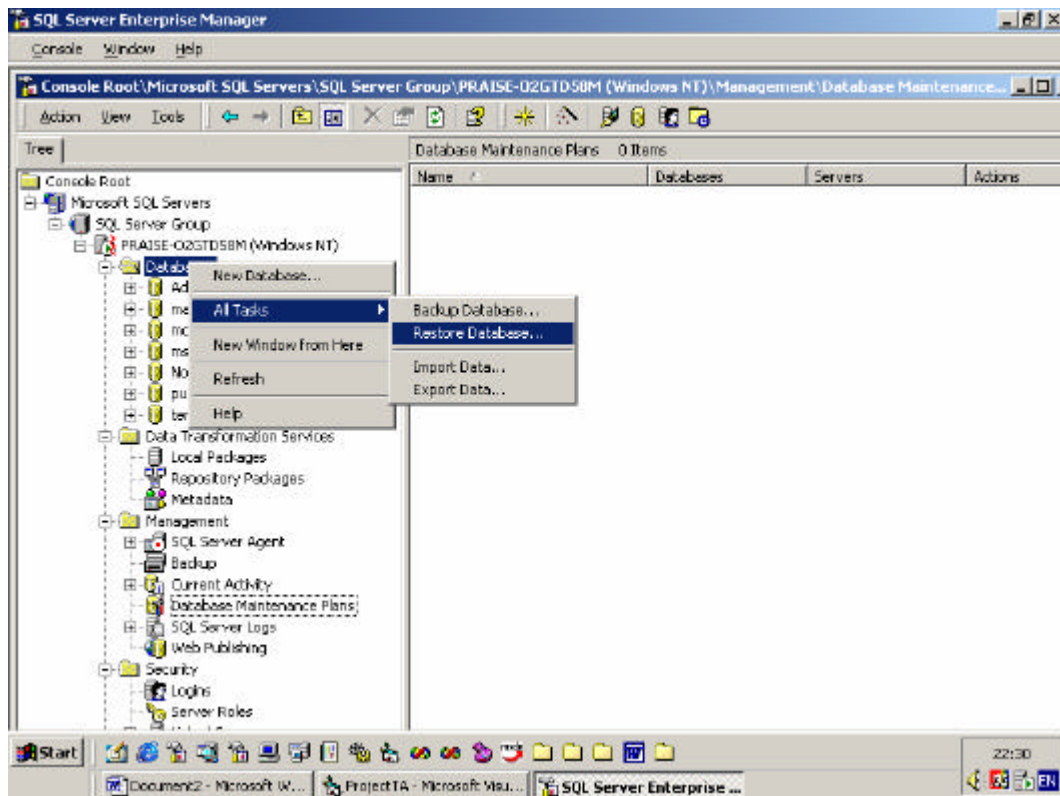
# MANUAL BOOK

Untuk dapat menjalankan keseluruhan aplikasi, perlu dilakukan beberapa konfigurasi. Adapun konfigurasi-konfigurasi yang perlu dilakukan meliputi:

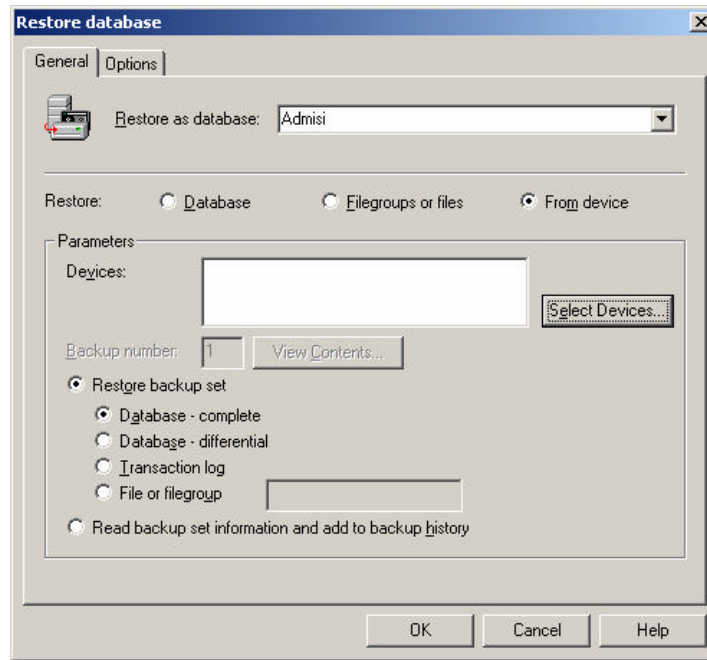
1. [Restore terhadap database-backup](#)
2. [Pengaturan Internet Information Services \(IIS\)](#)
3. [Instalasi middle-tier MTS Explorer \(MTX\)](#)
  - [Penciptaan package](#)
  - [Instalasi komponen](#)
  - [Pengeksporan package](#)
4. [Petunjuk penggunaan website](#)
  - [GiftHouse](#)
  - [MoviePoll](#)

# 1. RESTORE TERHADAP DATABASE-BACKUP

*Database* yang digunakan di dalam pembuatan kedua buah *database* ini adalah Microsoft SQL Server 7.0. Untuk menguji coba *database* yang telah di *backup* ini pada *server* Anda, gunakanlah *option* restore *database backup*. Cara penggunaannya cukup mudah yaitu dengan mengklik kanan pada bagian *databases* dan memilih all task restore *database*, seperti terlihat pada Gambar 1. Setelah itu akan muncul dialog untuk merestore *database* yang diinginkan, seperti pada Gambar 2. Pilihlah *option* Restore from Device dan tentukanlah *device* dan direktori yang diinginkan, yaitu *database-backup* yang terletak pada *folder* Database.



Gambar 1. Restore Database Backup

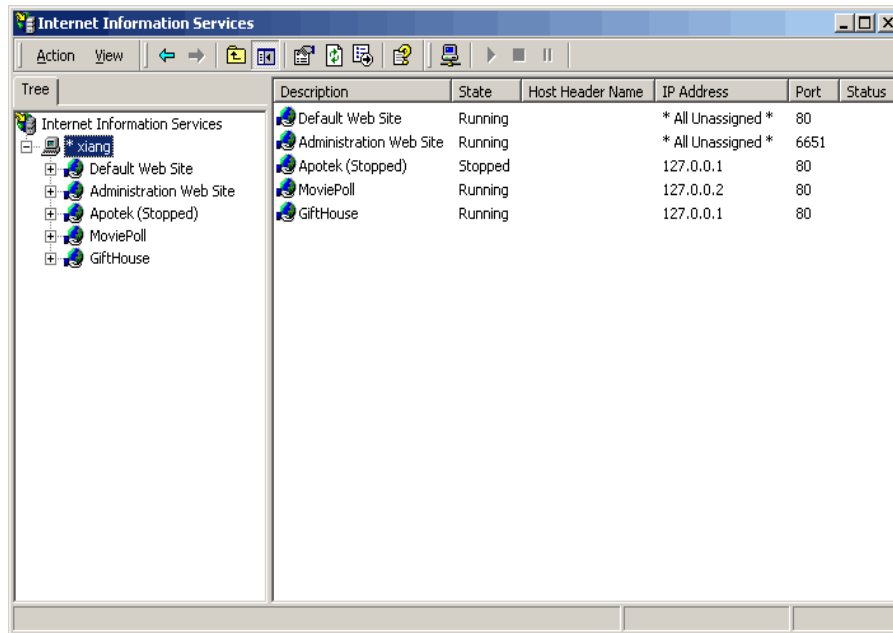


Gambar 2. Pilihan untuk Restore Database

## 2. PUBLIKASI WEBSITE

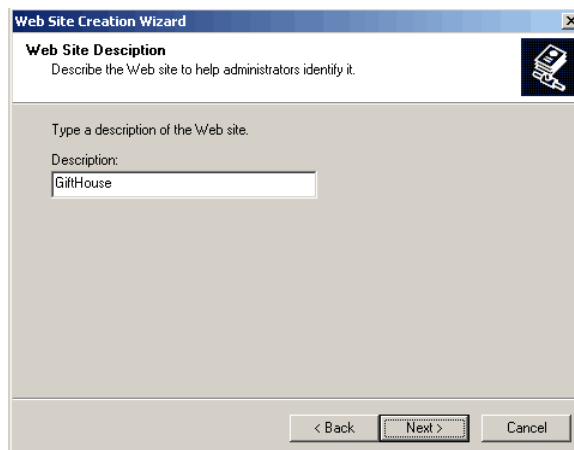
*Internet Information Services* (IIS) adalah merupakan sebuah komponen dari Windows yang dapat mempermudah kita untuk melakukan publikasi terhadap informasi dan membawa aplikasi-aplikasi bisnis ke dalam *website*. IIS akan mempermudah penciptaan sebuah dasar yang kuat bagi aplikasi-aplikasi beserta komunikasinya di dalam jaringan. Langkah-langkah yang harus dilakukan untuk melakukan publikasi *website* adalah sebagai berikut:

1. Pilihlah Start|Programs|Administrative Tools atau Start|Setting|Control Panel untuk kemudian masuk ke dalam *folder* Administrative Tools.
2. Pilih *Internet Service Manager* untuk mulai menggunakan *Internet Information Services* (IIS). Gambar 1. menunjukkan tampilan *window* dari *Internet Information Services* yang diakses melalui Internet Service Manager dari sebuah komputer bernama xiang. Secara *default*, Windows akan memiliki dua buah aplikasi, yaitu Default Web Site dan Administrative Web Site. Pada bagian kanan dari *windows* ini, akan ditampilkan deskripsi dari masing-masing aplikasi yang ada di dalam *server*.



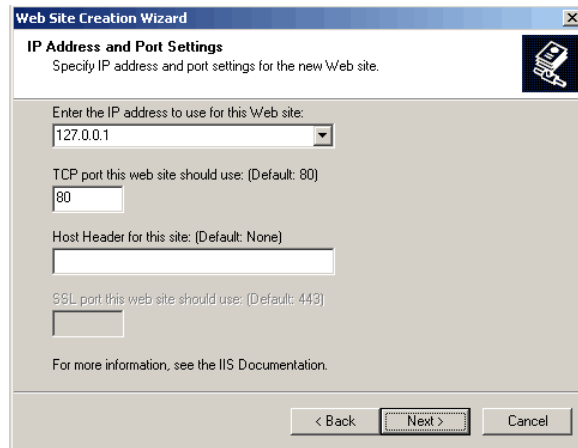
Gambar 3. Internet Information Services (IIS)

- Untuk mulai melakukan pengaturan publikasi dari *website* Anda, klik kanan pada nama komputer Anda (dalam contoh ini, xiang) dan pilihlah menu New|Web Site.
- Masukkan nama dari *website* baru yang ingin Anda bangun ke dalam *dialog* yang akan muncul, seperti pada Gambar 4.



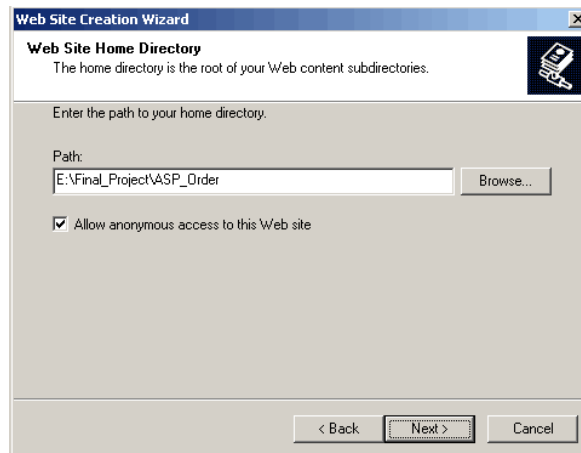
Gambar 4. Penentuan Nama Website

- Tekan tombol Next untuk melanjutkan.
- Lakukan konfigurasi *IP address*, *TCP port*, dan *host header* dari *website* yang ingin Anda bangun, seperti terlihat pada Gambar 5. Jika Anda bekerja di dalam *local server*, pilihlah *IP address* 127.0.0.[0-9].



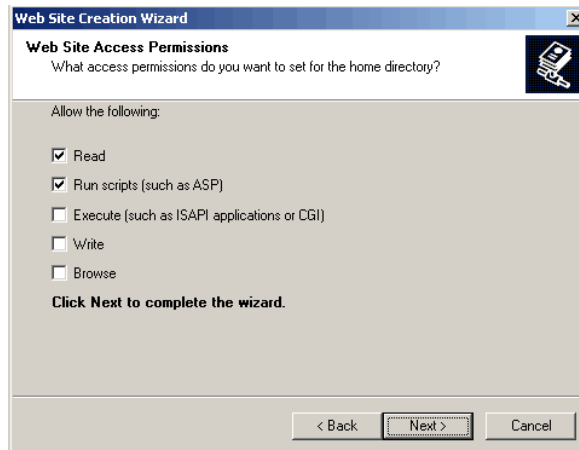
Gambar 5. Konfigurasi Website

7. Tekan tombol Next untuk melanjutkan.
8. Masukkan *path* lengkap dari keseluruhan *file* yang ingin Anda publikasikan.



Gambar 6. Penentuan Lokasi Website

9. Tekan tombol Next untuk melanjutkan.
10. Lakukan pengaturan terhadap ijin akses dari *website* Anda. Ada 5 pilihan yang dapat diaktifkan maupun dinon-aktifkan, yaitu *Read*, *Run Scripts (such as ASP)*, *Execute (such as ISAPI applications or CGI)*, *Write*, dan *Browse*.



Gambar 7. Website Access Permissions

11. Tekan tombol Next untuk menyelesaikan publikasi dari *website* Anda.
12. Perhatikan pada *field* State frame sebelah kanan dari *Internet Information Services* (IIS). Jika *state* dari *website* Anda memiliki status Running, maka *website* Anda siap untuk digunakan. Jika *state* bernilai Stopped, maka Anda harus mengaktifkan *website* Anda sebelum dapat mulai menjalankannya.
13. Cara mengaktifkan *website* yang berada dalam keadaan Stopped adalah dengan mengklik kanan pada nama *website* Anda dan memilih menu Start. Jika terdapat *website* lain yang memiliki *IP address* yang sama dengan *website* yang ingin Anda jalankan dan sedang dalam keadaan Running, maka Anda harus menonaktifkan *website* tersebut lebih dahulu dengan mengklik kanan pada nama *website* tersebut dan memilih menu Stop.
14. Untuk membuka halaman *website* Anda, bukalah sebuah *window* Internet Explorer dan ketikkan alamat berikut:  
`http://[IP address]/[nama file]`

### 3. INSTALASI KOMPONEN

Oleh karena *prototype* yang dibangun di sini bekerja bersama dengan MTS, maka dibutuhkan instalasi objek-objek yang telah dibangun ke dalam MTS. Hal ini akan melibatkan konfigurasi dari Microsoft Management Console, yaitu dengan menciptakan sebuah atau beberapa *package*. Packages dapat diasumsikan sebagai sebuah wadah dari komponen-komponen yang akan digunakan. Setiap package dapat berisi sekumpulan komponen yang dikelompokkan berdasarkan

fungsinya. Jadi adalah memungkinkan untuk meletakkan beberapa komponen yang berfungsi untuk melakukan fungsi pemrosesan ke dalam sebuah *package* dan komponen-komponen yang lain ke dalam *package* yang berbeda. Hal ini akan membuat keseluruhan aplikasi menjadi jauh lebih mudah didistribusikan, dikelola, dan memungkinkan untuk sebuah aplikasi menggunakan beberapa komponen.

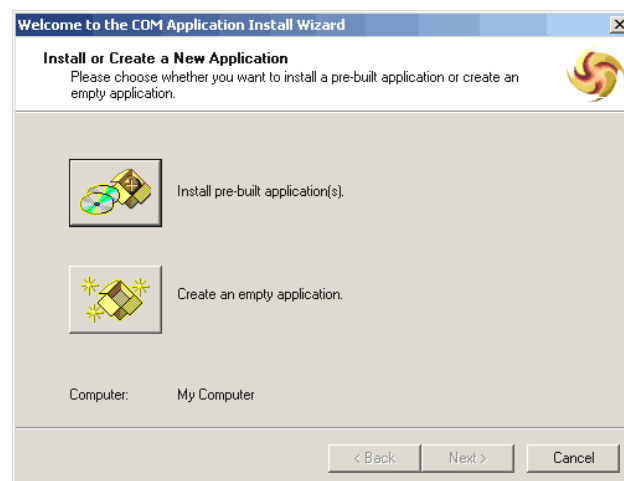
## • INSTALASI AWAL

Untuk memulai menggunakan *prototype*, maka dibutuhkan instalasi dari komponen-komponen. Berikut dijelaskan bagaimana cara melakukan instalasi komponen-komponen dengan menggunakan MTS Explorer (MTX).

### 1. Ciptakan sebuah *package* baru

Langkah-langkah yang harus dilakukan dalam menciptakan sebuah *package* baru adalah sebagai berikut:

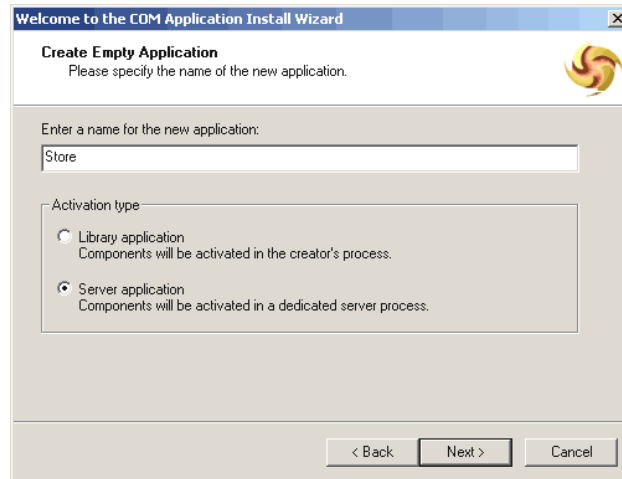
- a. Klik kanan pada *folder* Package Installed dan pilihlah menu New|Application.
- b. Jika kita telah menyimpan *package* ini sebelumnya, kita dapat melakukan penginstalan ulang padanya dengan memilih tombol pertama (Install pre-built packages) pada *dialog* pertama yang muncul. Untuk membangun sebuah *package* baru, tombol yang harus dipilih adalah Create an empty package.



Gambar 8. Create New Package

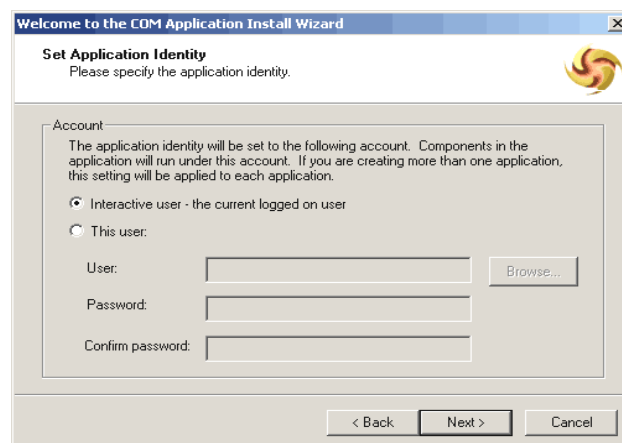
- c. Tekan tombol Next untuk melanjutkan.

- d. Tentukan nama dari aplikasi/*package* yang ingin Anda bangun. Tentukan pula jenis aktivasinya, dalam hal ini pilihlah Server application.



Gambar 9. Penentuan Nama Package

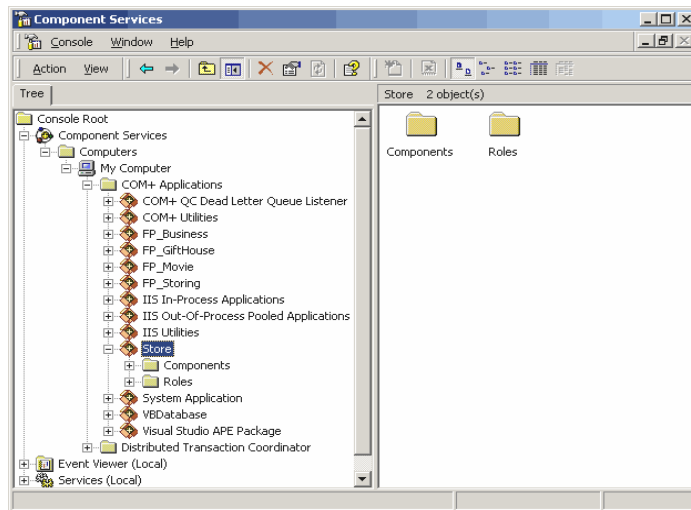
- e. Tekan tombol Next untuk melanjutkan.
- f. Tentukan hak akses bagi *package* yang ingin Anda bangun. Untuk memberikan hak akses untuk semua *user* yang sedang *login* pada saat itu, pilihlah Interactive user – the current logged on user. Untuk memberikan hak akses kepada *user* tertentu, pilihlah This user, dan masukkan *username* dan *password* pada *field-field* yang tersedia (jika Anda menggunakan Windows 9x, *dialog* ini tidak akan ditampilkan).



Gambar 10. Informasi User dari Package

- g. Tekan tombol Next untuk melanjutkan.

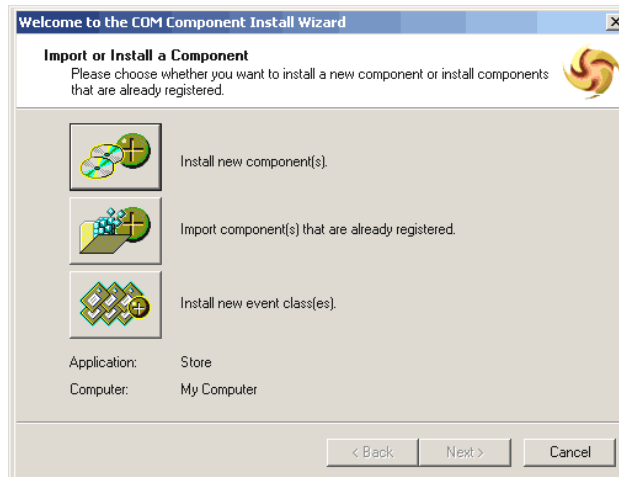
- h. *Package* telah selesai dibangun. Pada akhirnya akan terbentuk dua buah *folder* di dalam *package* Anda, yaitu *Components* dan *Roles*. *Folder Components* akan berisi *class-class* yang ada di dalam *file DLL* sedangkan *folder Roles* akan berisi *user-user* dari *group-group* yang diijinkan untuk menggunakan *package* tersebut.



Gambar 11. Package yang Telah Terbentuk

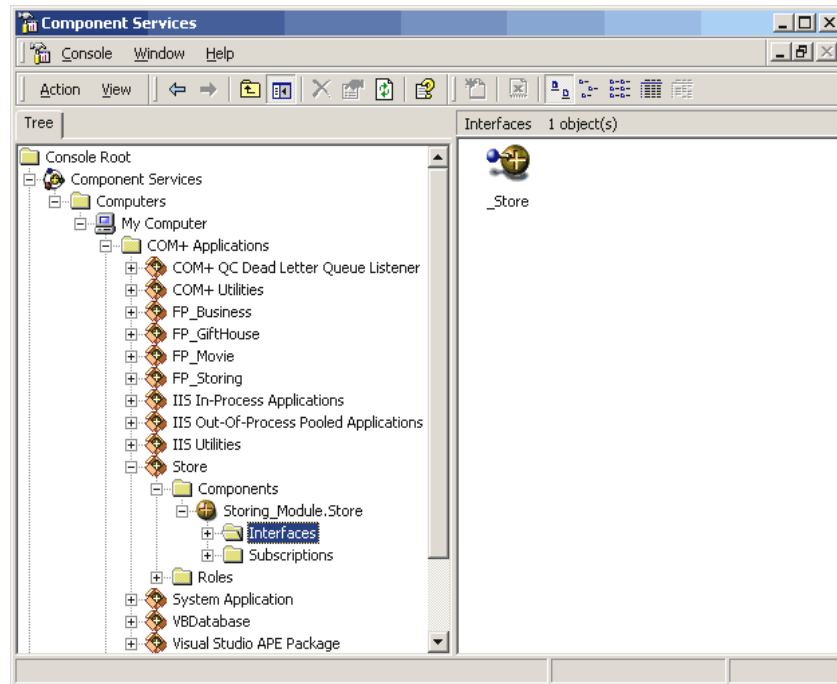
## 2. INSTALASI KOMPONEN KE DALAM PACKAGE

- a. Klik kanan pada *folder Components* dan pilihlah *New|Component*.
- b. Anda akan dihadapkan pada tiga buah pilihan, yaitu *Install new component(s)*, *Import component(s) that are already registered*, atau *Install new event class(es)*. Jika komponen telah diregistrasi, maka MTS tidak perlu meregistrasi ulang komponen tersebut sehingga komponen tersebut hanya perlu diimpor ke dalam *package*. Namun jika komponen tersebut belum diregistrasi (kemungkinan karena dipindahkan ke mesin lain sebagai sebuah *file*), maka dibutuhkan instalasi dari komponen tersebut sehingga MTS yang akan melakukan registrasi dari komponen tersebut dan kemudian mengimpornya ke dalam *package*. Dalam hal ini, asumsikan bahwa komponen-komponen yang Anda miliki belum teregistrasi sehingga diperlukan instalasi komponen.



Gambar 12. Instalasi Komponen ke Dalam Package

- c. Pilihlah komponen yang akan Anda masukkan ke dalam *package*. Perlu diingat bahwa komponen-komponen yang dipilih di sini adalah komponen-komponen dengan ekstensi *.dll*, yaitu komponen-komponen yang dibentuk dengan ActiveX DLL.
- d. Tekan tombol Finish untuk menyelesaikan proses instalasi komponen.
- e. Gambar berikut ini menunjukkan sebuah komponen dengan nama *Storing\_Module* yang memiliki satu buah *class* dengan nama *Store.cls*. Komponen ini diletakkan di dalam *package* yang bernama *Store*, yang telah kita buat pada langkah sebelumnya. Setiap komponen yang ada di dalam sebuah *package* ini akan memiliki dua buah *folder*, yaitu *Interfaces* dan *Subscriptions*. *Interfaces* akan berisi *interface-interface* yang dimiliki oleh komponen beserta dengan *method-method* yang dimiliki oleh setiap *interface*. Folder *Subscriptions* dapat digunakan untuk pengaturan komunikasi antar *interface* lebih lanjut.



Gambar 13. Komponen Storing\_Module dengan class Store

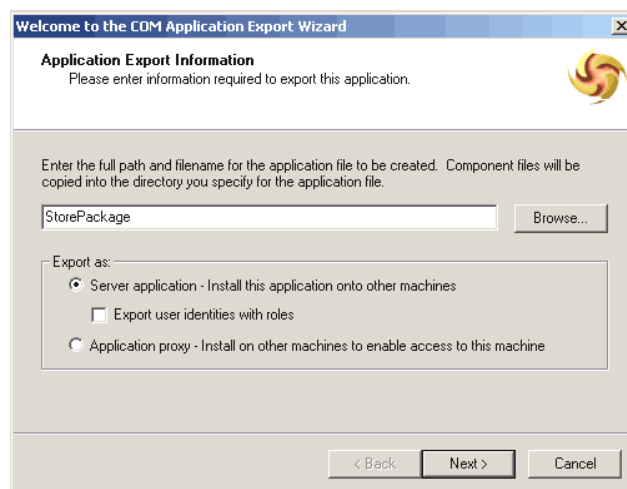
### 3. PENGEKSPORAN PACKAGE

Package bersifat sangat *portable* sehingga jika Anda menginginkan untuk memindahkan *package* dari satu *server* ke *server* yang lain, Anda dapat mengekspor *package* tersebut dari PC pertama dan kemudian menginstalnya pada PC yang baru. Pengeksporan *package* dilakukan untuk mengekspor *package* ke dalam komputer *server* yang lain, yang juga menjalankan MTS. Selain itu, kita juga dapat membuat sebuah aplikasi yang bersifat *executable* bagi komputer-komputer *client* yang terpisah yang berjalan di dalam Windows NT ataupun Windows 9x (dengan dukungan terhadap DCOM) agar dapat mengakses aplikasi *server*.

Registrasi dari setiap komputer *client* harus dikonfigurasi terlebih dahulu agar dapat dikenali dengan menggunakan DCOM di dalam mesin tertentu (di mana objek berada). Microsoft Transaction Server (MTS) memiliki sebuah *built-in utility* yang berfungsi untuk menciptakan sebuah *executable* yang dapat dijalankan di dalam mesin *client*. Pada saat *executable* ini dijalankan, ia akan melakukan modifikasi terhadap registrasi dari komputer *client*, yang akan membuat semua perubahan yang dibutuhkan terhadap nilai-nilai yang ada.

Langkah-langkah yang harus dilakukan di dalam melakukan ekspor terhadap sebuah *package* adalah sebagai berikut:

1. Klik kanan pada *package* yang ingin Anda ekspor dan pilihlah menu Export. Pemilihan menu ini akan menampilkan sebuah *dialog* seperti terlihat pada Gambar 14.
2. Isikan *path* dari *executable* lengkap dengan namanya, atau tekan tombol *Browse* dan tentukan lokasi di mana Anda akan menyimpan *file-file* yang akan diekspor.



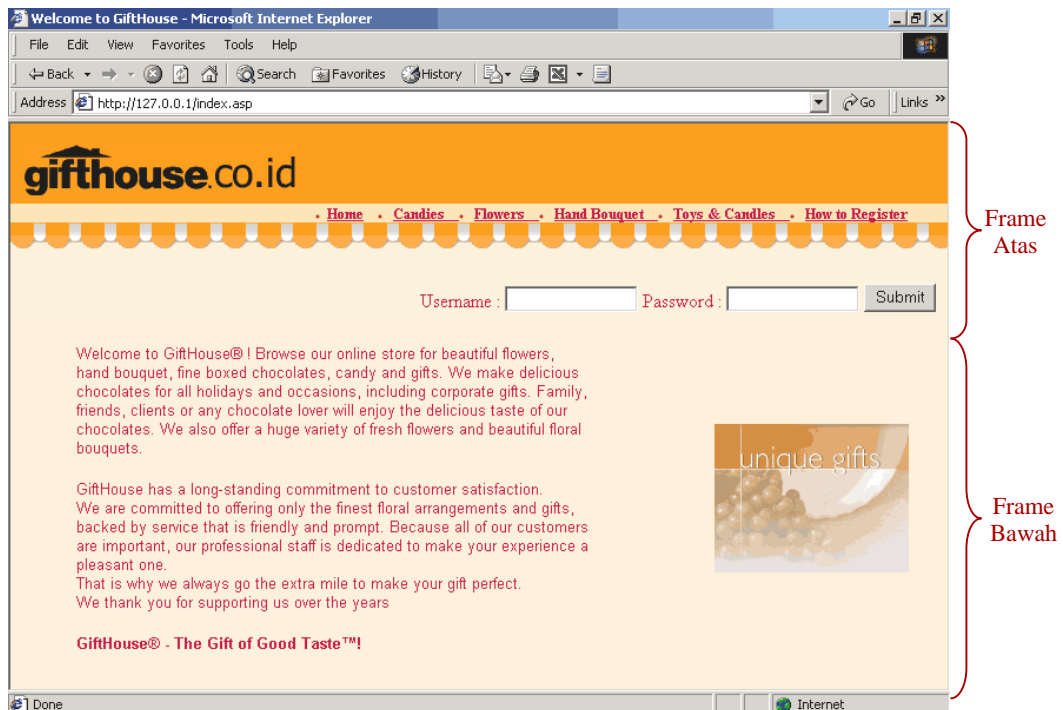
Gambar 14. Pengeksporan Package

3. Tekan tombol Next untuk melanjutkan.
4. MTS akan menciptakan sebuah *executable* dan sebuah *folder* dengan ekstensi .MSI di dalam *path* yang telah Anda tentukan sebelumnya.
5. Bukalah *path* ini dari komputer tempat Anda ingin mengekspor *package* ini.
6. Jalankan *executable* dari komputer tempat Anda ingin melakukan ekspor. Pada saat *executable* dijalankan, ia akan secara otomatis melakukan *unpacking* terhadap dirinya sendiri dan melakukan eksekusi di dalam *registry* serta memberikan masukan-masukan di dalam mesin di mana ia dijalankan sehingga ia dapat mengenali nama dari *server* di mana komponen-komponen tersebut diletakkan.

## 4. PETUNJUK PENGGUNAAN WEBSITE

### • WEBSITE GIFTHOUSE

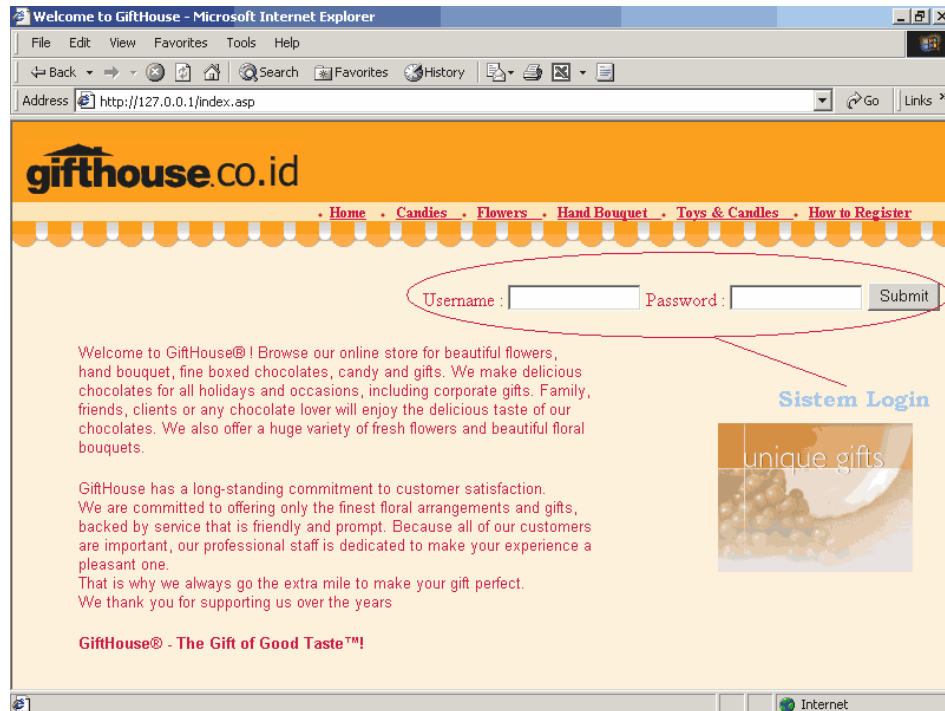
Website ini menggunakan 2 buah *frame* untuk memudahkan pengunjung berpindah-pindah halaman, seperti dapat dilihat pada Gambar 15. di bawah. *Frame* bagian atas berisi menu-menu yang dapat diakses oleh pengunjung, meliputi kategori-kategori produk dan *link* untuk mendapatkan informasi cara melakukan registrasi sedangkan *frame* bagian bawah akan menampilkan halaman *web* yang dipilih oleh pengunjung pada saat mengklik menu pada bagian *frame* atas.



Gambar 15. Halaman Utama GiftHouse

### 1. SISTEM LOGIN

Dalam sistem *login* yang terdapat pada halaman utama ini terdapat 2 buah *usergroup*, yaitu administrator dan *member*. Sistem *login* ini berfungsi untuk melakukan autentikasi dari pengakses sistem karena masing-masing *usergroup* memiliki akses terhadap fasilitas-fasilitas yang berbeda-beda.



Gambar 16. Sistem Login pada GiftHouse

Proses registrasi tidak dilakukan secara *online*. Menu How to Register menunjukkan bagaimana cara seseorang mendapatkan *username* dan *password* dan mendapatkan fasilitas-fasilitas yang hanya diberikan kepada *member*.

To become a member, you only have to contact our GiftHouse store everywhere around you and show your Rp.200.000,- purchasing bill of our store. Our staffs will personalize a username and password for you.

Gambar 17. Proses Registrasi pada GiftHouse

Pada saat melakukan *login*, maka pengunjung *web site* ini akan diminta memasukkan *username* dan *password*. Jika terjadi kesalahan dalam pemasukan *username* ataupun *password*, maka akan muncul pesan “*Invalid Username or Password*”, seperti terlihat pada Gambar 18. di bawah ini.

Invalid Username or Password  
[Re-Login](#)

Gambar 18. Kesalahan Login pada GiftHouse

Untuk melakukan *login* ulang ke *website*, pengunjung dapat memilih menu *Re-Login*. Menu ini akan mengembalikan pengunjung ke halaman utama dari *website* ini.

Jika *username* dan *password* yang diinputkan oleh pengunjung adalah benar, maka pengunjung akan mendapatkan statusnya sebagai *member* dan akan ditampilkan *point* terakhir yang dimilikinya, seperti terlihat pada Gambar 19. berikut:



Welcome erni, your points are 10

Gambar 19. Welcome Member GiftHouse

## 2. SISTEM PEMESANAN ONLINE

Sistem pemesanan *online* ini dapat diakses oleh semua pengunjung *website* GiftHouse. Sistem pemesanan *online* akan memungkinkan seluruh pengunjung untuk dapat melakukan *browsing* produk-produk yang ditawarkan oleh GiftHouse sampai kepada pemesanan produk-produk yang dipilih secara *online*. Ada beberapa tahap yang harus dilakukan oleh pengunjung agar transaksi pemesanan *online* dapat diproses, yaitu:

- Browsing Produk

Seorang pengunjung dapat melakukan *browsing* produk sesuai dengan kategori dari produk tersebut. Pada Gambar 20., ditampilkan produk-produk yang termasuk di dalam kategori Candies, setelah pengunjung meng-klik pada menu Candies di *frame* bagian atas. Tampilan dari daftar produk ini berupa tabel dengan jumlah kolom 6 buah yang dilengkapi dengan gambar produk dalam ukuran kecil untuk memudahkan pengunjung memilih produk yang diinginkan.



Gambar 20. Fasilitas Browse pada GiftHouse

- Tampilan Detail Produk

Seorang pengunjung dapat mengklik pada *link* Detail yang terdapat pada bagian paling bawah pada setiap kolom dari produk untuk dapat melihat rincian produk yang bersangkutan serta memasukkan produk tersebut ke dalam keranjang pembelian, yaitu dengan memasukkan jumlah produk yang akan dibeli ke dalam *field* yang telah disediakan. Gambar 21. menunjukkan detail dari produk bernama Jelly Belly.



Gambar 21. Detail Produk pada GiftHouse

- Pemasukan Produk ke Keranjang

Setiap sebuah produk dimasukkan ke dalam keranjang pembelian, akan ditampilkan daftar produk yang ada di dalam keranjang. Pada halaman ini, seorang pengunjung masih dapat melakukan *update* dari jumlah produk yang dipesan, ataupun membatalkan pemesanan dari produk tertentu. Peng-*update*-an dapat dilakukan dengan memasukkan jumlah pengurangan ataupun penambahan produk. Sebagai contoh, jika seorang pengunjung telah memasukkan sebuah produk sejumlah 3 buah, maka pada saat ia menginputkan 2 dan menekan tombol OK, jumlah produk akan menjadi 5. Sedangkan jika ia menginputkan -2 maka jumlah produk akan menjadi 1. Untuk menghapus produk tersebut dari keranjang, maka pengunjung harus menginputkan -3 dan menekan tombol OK.

Qty	Product Name	Price Per Item	Sub Total	Add/ Product
4	Jelly Belly	Rp. 37,500.00	Rp. 150,000.00	<input type="text" value="0"/> OK
2	Classic Camation	Rp. 320,000.00	Rp. 640,000.00	<input type="text" value="0"/> OK
Total:			Rp. 790,000.00	

[Check Out](#)

Gambar 22. Update Cart pada GiftHouse

- Pengisian Informasi Pelanggan dan Pengiriman Barang

Pada saat pengunjung telah selesai memilih produk yang akan dipesan, ia dapat mengklik Check Out. Pengunjung akan dibawa ke halaman yang berisi daftar terakhir dari produk-produk yang telah ia pesan, beserta dengan sebuah form untuk melakukan pengisian informasi pelanggan dan pengiriman barang, seperti dapat dilihat pada Gambar 23. Adapun informasi yang dibutuhkan meliputi data-data dari pemesan (nama lengkap pemesan, alamat, dan nomor telepon) serta data-data pengiriman barang (nama lengkap orang yang dikirim dan alamat pengiriman barang).

Cart Confirmation & Checkout

Jumlah	Nama Barang	Harga Satuan	Sub Total
4	Jelly Belly	Rp. 37,500.00	Rp.150,000.00
2	Classic Carnation	Rp. 320,000.00	Rp.640,000.00
Total:			Rp.790,000.00

**Informasi Pembayaran / Pengiriman Barang**

**Informasi Pelanggan**

Nama Lengkap :

Alamat:

Nomor Telepon:  Misalnya: 739632

---

**Informasi Pengiriman Barang (jika berbeda dari Informasi Pelanggan)**

Nama Lengkap:

Alamat:

Gambar 23. Form Informasi Pelanggan pada GiftHouse

- Validasi Pesanan

Tahap terakhir dari proses pemesanan *online* ini adalah melakukan *submit* pesanan. Hal ini dapat dilakukan dengan menekan tombol Submit Order pada halaman Pengisian Informasi Pelanggan yang dapat dilihat pada Gambar 24. di atas. Jika terjadi beberapa permasalahan di dalam pemrosesan pesanan, keseluruhan transaksi akan dibatalkan dan tidak ada satu pun data yang akan dimasukkan ke dalam *database* di dalam *server*. Pada halaman *website* akan muncul informasi sebagai berikut:

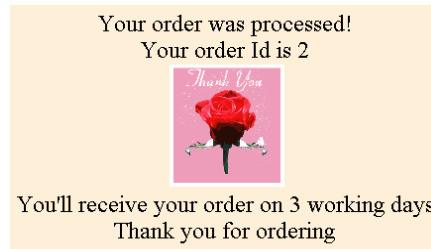
**We have problems with your order.  
Can't complete the transaction**

Please check back for your order  
or try again in a few moment

Gambar 24. Masalah Pemrosesan Pesanan pada GiftHouse

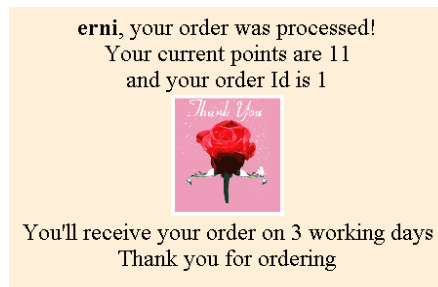
Namun jika tidak terdapat permasalahan pada pemrosesan pesanan dari pengunjung, maka pesanan akan tersimpan di dalam tabel Orders dan ItemsOrdered, serta akan dilakukan peng-*update*-an pada tabel Member

jika pengunjung yang melakukan pemesanan adalah *member*. Adapun untuk *browser* yang bukan *member*, akan diinformasikan bahwa pesanan telah berhasil diproses dan diberikan sebuah *orderId* yang akan digunakan untuk konfirmasi lebih lanjut.



Gambar 25. Informasi untuk Non-Member pada GiftHouse

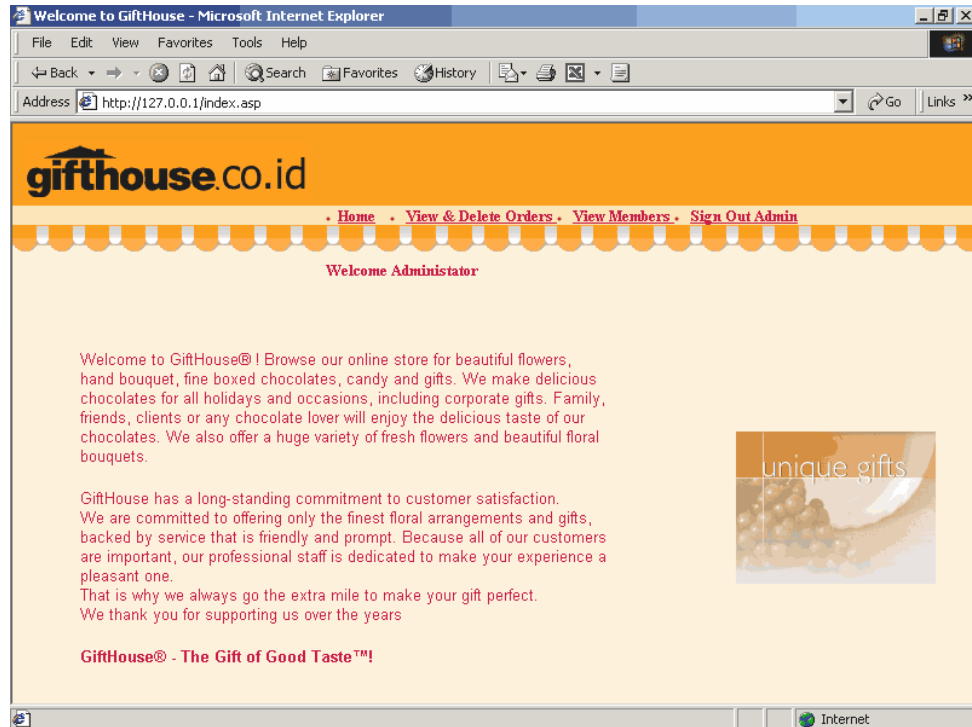
Sedangkan untuk *member*, selain diberikan *orderId* yang dapat digunakan untuk konfirmasi pesanan, juga akan ditampilkan pula informasi dari *point* terakhir yang berhasil dikumpulkannya agar mereka dapat memastikan berapa jumlah *point* yang berhasil mereka kumpulkan.



Gambar 26. Informasi untuk Member pada GiftHouse

### 3. SISTEM ADMINISTRATOR

Menu Administrator hanya dapat diakses oleh *user* yang memiliki *username* admin. Ada dua jenis menu yang diberikan kepada Administrator, yaitu View & Delete Orders untuk melihat dan menghapus pesanan, dan View Members untuk menampilkan data-data dari *member*. Kedua fasilitas ini diberikan kepada administrator untuk memudahkan mereka dalam mengorganisir pesanan-pesanan dari pengunjung serta melakukan pengecekan terhadap *member-member* dari *website* ini.



Gambar 27. Sistem Administrator


- View & Delete Order

Menu ini digunakan oleh administrator dari *website* GiftHouse untuk melihat daftar pesanan, serta untuk menghapus pesanan-pesanan yang sudah terpenuhi.



Gambar 28. View & Delete Order pada GiftHouse

Jika pesanan telah terpenuhi, maka administrator dapat menghapus pesanan tersebut dengan mengklik pada menu *Delete this order*. Jika penghapusan tidak berhasil dilakukan oleh *server*, maka akan muncul pesan sebagai berikut:



Can not delete order Id 3  
[Return to Order List](#)

Gambar 29. Proses Penghapusan Pesanan yang Gagal pada GiftHouse  
Namun jika penghapusan berhasil dilakukan, akan ditampilkan pesan:

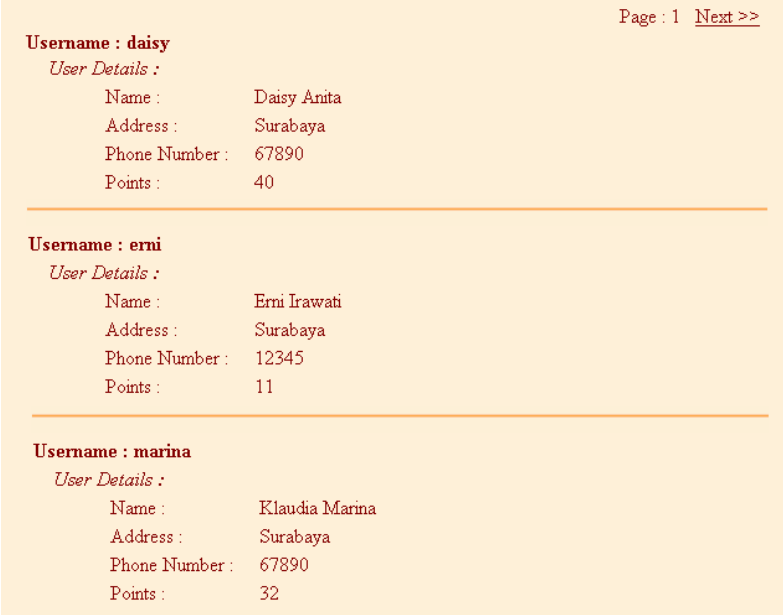


Order Id 4 has been deleted  
[Return to Order List](#)

Gambar 30. Proses Penghapusan Pesanan yang Berhasil pada GiftHouse

- View Members

Menu ini digunakan oleh administrator dari *website* GiftHouse untuk melihat data-data dari *member* mereka. Data-data tersebut akan ditampilkan sebanyak lima buah *record* untuk setiap halamannya. Tampilan menu ini dapat dilihat pada Gambar 31. berikut:



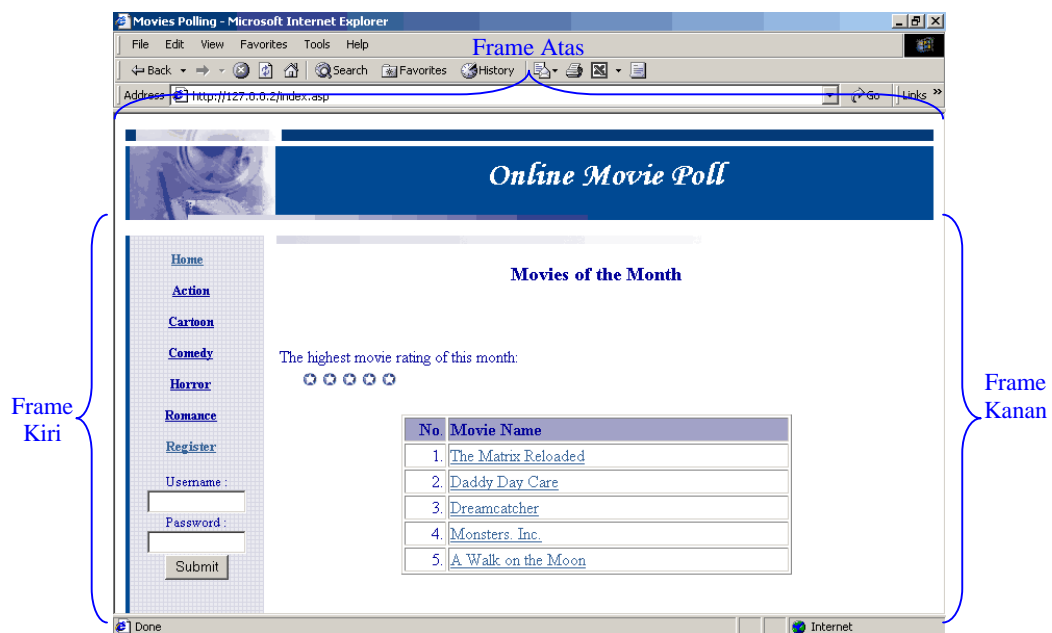
Page : 1 [Next >>](#)

<b>Username : daisy</b>	
<i>User Details :</i>	
Name :	Daisy Anita
Address :	Surabaya
Phone Number :	67890
Points :	40
<hr/>	
<b>Username : erni</b>	
<i>User Details :</i>	
Name :	Erni Irawati
Address :	Surabaya
Phone Number :	12345
Points :	11
<hr/>	
<b>Username : marina</b>	
<i>User Details :</i>	
Name :	Klaudia Marina
Address :	Surabaya
Phone Number :	67890
Points :	32

Gambar 31. View Members pada GiftHouse

## • WEBSITE MOVIEPOLL

Pada dasarnya *website* MoviePoll adalah sebuah *website* sederhana yang berfungsi untuk melakukan *polling* terhadap film-film yang sedang beredar selama satu bulan. *Website* ini menggunakan 3 buah *frame* untuk memudahkan pengunjung berpindah-pindah halaman, seperti dapat dilihat pada Gambar 4.28. di bawah. *Frame* bagian atas hanya merupakan *header* dari *website* dan bersifat statis. *Frame* sebelah kiri berisi menu-menu yang dapat diakses oleh pengunjung, serta sebuah *form* untuk melakukan *login* sebagai *member*. *Frame* di sebelah kanan akan menampilkan halaman *web* yang dipilih oleh pengunjung pada saat mengklik menu pada bagian *frame* bagian kiri. Pada halaman utama, akan ditampilkan daftar film-film yang memperoleh *rating* tertinggi.

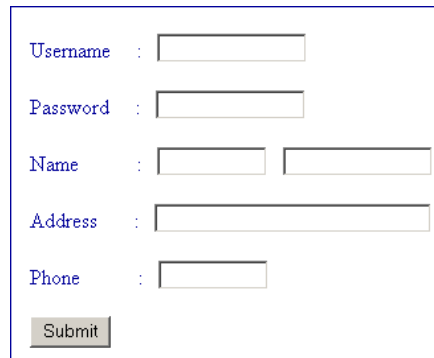


Gambar 32. Halaman Utama pada MoviePoll

### 1. SISTEM LOGIN DAN REGISTRASI

Dalam sistem *login* yang terdapat pada halaman utama ini hanya terdapat satu buah *usergroup*, yaitu *member*. Hal ini dikarenakan keseluruhan administrasi tidak dilakukan secara *online*. Sistem *login* ini berfungsi untuk melakukan autentikasi dari pengakses sistem karena hanya para pengunjung yang telah melakukan registrasi yang dapat mengikuti *polling*.

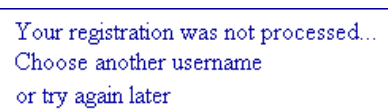
Adapun di dalam *website* ini, proses registrasi dilakukan secara *online*. Menu registrasi akan menampilkan form registrasi, seperti yang terlihat pada Gambar 33., yang harus dilengkapi oleh seorang pengunjung yang ingin mendaftarkan diri sebagai *member*.



A registration form with the following fields: Username, Password, Name (split into two input boxes), Address, and Phone. A Submit button is located at the bottom left of the form.

Gambar 33. Proses Registrasi pada MoviePoll

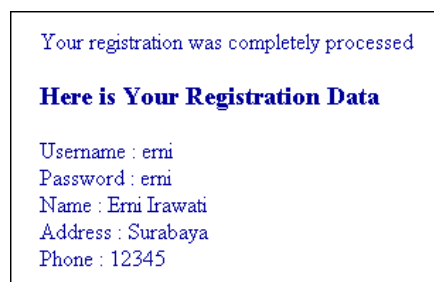
Registrasi ini akan diproses di dalam *server*, yang kemudian keseluruhan data ini akan disimpan di dalam *database*, yaitu di dalam tabel Member. Jika pada saat pemrosesan data terjadi beberapa permasalahan, maka keseluruhan data tidak akan tersimpan di dalam *database* dan akan ditampilkan informasi untuk pengunjung, sebagai berikut:



Your registration was not processed...  
Choose another username  
or try again later

Gambar 34. Kesalahan Registrasi pada MoviePoll

Namun jika tidak terjadi masalah dan *username* yang dipilih belum pernah terdaftar, maka akan ditampilkan pesan sebagai berikut:



Your registration was completely processed  
**Here is Your Registration Data**  
Username : erni  
Password : erni  
Name : Erni Irawati  
Address : Surabaya  
Phone : 12345

Gambar 35. Welcome Member GiftHouse

## 2. FASILITAS BROWSE

Fasilitas *browse* ini disediakan untuk para pengunjung yang hanya ingin melihat-lihat sinopsis dari film-film yang sedang beredar di pasaran. Gambar 36. menunjukkan daftar film yang ditampilkan berdasarkan kategori filmnya.

Movie Name	Current Rating
<a href="#">Monsters. Inc.</a>	★ ★ ★ ★ ★
<a href="#">Ice Age</a>	★ ★ ★
<a href="#">The Emperor's New Groove</a>	★ ★ ★
<a href="#">Hunchback of Notre Dame</a>	★ ★ ★
<a href="#">Shrek</a>	★ ★ ★
<a href="#">Lilo and Stitch</a>	★ ★ ★

Gambar 36. Browse Film berdasarkan Kategori pada MoviePoll

Pada saat seorang pengunjung memilih salah satu film, ia akan dibawa ke sebuah halaman yang berisi sinopsis dari film tersebut. Di sini ditampilkan sebuah informasi bahwa untuk melakukan *polling*, seorang pengunjung harus melakukan proses *login* sebagai *member*.



### The Emperor's New Groove

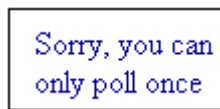
Kuzco (David Spade) is the young, arrogant emperor of a vast kingdom. For his entire life, he has always had everything his own way. When his adviser, Yzma (Eartha Kitt), decides to usurp his power for herself, she has her servant Kronk (Patrick Warburton) poison Kuzco--but something goes wrong, and instead of dying, Kuzco is transformed into a talking llama. The former emperor winds up in the company of Pacha (John Goodman), a simple peasant just evicted from his land by Kuzco's plans for a new summer home. Pacha and Kuzco grudgingly agree to work together to restore Kuzco to his throne--and back to human form--but a series of wacky misadventures and the schemes of Yzma ensure that their journey will be a treacherous one.

**You have to register first before you can poll**

Gambar 37. Fasilitas Browse untuk Non-Member pada MoviePoll

### 3. MENU MEMBER

Menu ini hanya dapat diakses oleh *member*. Untuk dapat menjadi *member*, seorang pengunjung harus melakukan registrasi agar mendapatkan *username* dan *password*. Seorang *member* hanya dapat melakukan *polling* satu kali dalam satu periode. Pada saat seorang *member* telah melakukan *polling*, status dari *member* tersebut akan berubah dan ia tidak dapat melakukan *polling* untuk kedua kalinya dan akan ditampilkan informasi bahwa ia hanya diijinkan melakukan *polling* satu kali dalam satu periode, seperti terlihat pada Gambar 38.



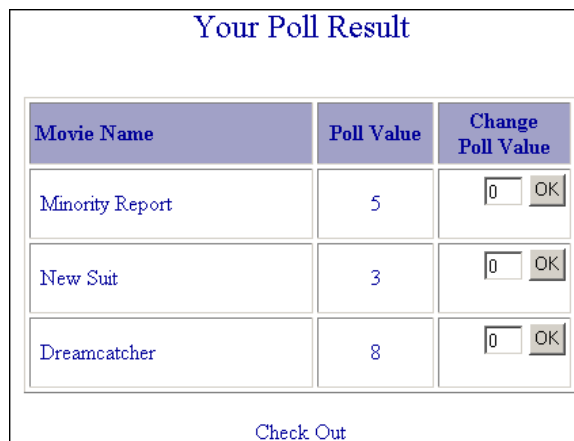
Gambar 38. Maksimal Polling

Pada saat seorang *member* memilih sebuah film dari daftar film-film yang ditampilkan berdasarkan kategorinya, ia akan dibawa ke sebuah halaman yang menampilkan sinopsis dari film tersebut serta sebuah *form* untuk menginputkan nilai yang ingin ia masukkan ke dalam *polling*. Tampilan dari halaman ini dapat dilihat pada Gambar 39. di bawah ini:



Gambar 39. Fasilitas Browse untuk Member pada MoviePoll

Setelah sebuah nilai dimasukkan ke dalam daftar *polling*, seorang *member* masih dapat melakukan *update* dari nilai yang telah ia masukkan, atau mengganti film yang ingin ia pilih. Peng-*update*-an dapat dilakukan dengan memasukkan jumlah pengurangan ataupun penambahan nilai. Sebagai contoh, jika seorang *member* telah memasukkan nilai 3 untuk sebuah film, maka pada saat ia menginputkan 2 dan menekan tombol OK, nilai yang ia masukkan untuk film tersebut akan menjadi 5. Sedangkan jika ia menginputkan -2 maka nilai yang ia inputkan akan menjadi 1. Untuk membatalkan pemilihan film, *member* harus menginputkan -3 dan menekan tombol OK



Movie Name	Poll Value	Change Poll Value
Minority Report	5	<input type="text" value="0"/> OK
New Suit	3	<input type="text" value="0"/> OK
Dreamcatcher	8	<input type="text" value="0"/> OK

[Check Out](#)

Gambar 40. Daftar Film yang Dipilih

Seorang *member* hanya dapat melakukan *polling* terhadap tiga buah film saja. Pada saat ia memasukkan pilihan film yang keempat, film tersebut tidak akan dimasukkan ke dalam daftar pilihan dan akan ditampilkan sebuah informasi bahwa jumlah film maksimal yang dapat dipilih adalah tiga buah, seperti dapat dilihat pada Gambar 41.:

Maksimal polling 3 buah film

### Your Poll Result

Movie Name	Poll Value	Change Poll Value
Minority Report	5	<input type="text" value="0"/> OK
New Suit	3	<input type="text" value="0"/> OK
Dreamcatcher	8	<input type="text" value="0"/> OK

[Check Out](#)

Gambar 41. Informasi Pilihan Film Maksimal

Tahap terakhir dari proses *polling* ini adalah melakukan *submit* dari pilihan-pilihan film. Hal ini dapat dilakukan dengan menekan tombol Submit pada halaman Poll Confirmation yang dapat dilihat pada Gambar 42. Jika terjadi beberapa permasalahan di dalam pemrosesan pilihan film, keseluruhan transaksi akan dibatalkan dan tidak ada satu pun data yang akan dimasukkan ke dalam *database* di dalam *server*. Pada halaman *website* akan muncul informasi seperti terlihat pada Gambar 43.

### Poll Confirmation

Movie Name	Poll Value
Minority Report	5
New Suit	3
Dreamcatcher	8

[Submit Your Poll](#)


[Edit Poll Value](#)

Gambar 42. Konfirmasi Pilihan Film

Your poll was not processed...  
Can't update movies rating

Gambar 43. Polling Tidak Diproses

Namun jika tidak terdapat permasalahan pada pemrosesan hasil *polling* dari *member*, maka hasil *poling* ini akan mempengaruhi tiga buah tabel, yaitu PollResult, Movies, dan Member.

A screenshot of a confirmation message displayed in a blue monospace font. The text is enclosed in a thin black rectangular border. The message reads: "erni, your poll was processed!" on the first line and "Thank you for polling" on the second line.

erni, your poll was processed!  
Thank you for polling

Gambar 44. Polling Diproses