

BAB IV

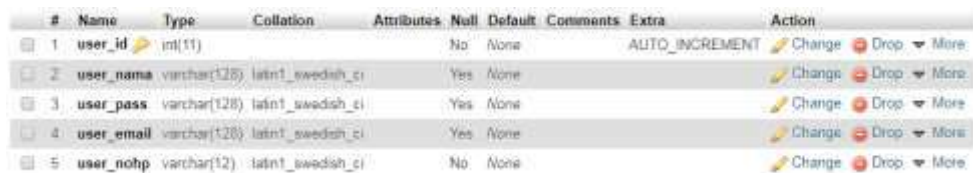
IMPLEMENTASI SISTEM

4.1 Implementasi Basis Data

Tahap implementasi merupakan tahap kelanjutan dari kegiatan perancangan sistem. Pembuatan basis data dilakukan dengan menggunakan *tools* MySQL. Sistem ini memiliki basis data yang terdiri dari 5 tabel. Berikut adalah tabel-tabel yang digunakan dalam basis data sistem aplikasi *booking room* karaoke.

4.1.1 Implementasi Tabel *User*

Tabel *user* terdiri dari 5 *fields* yaitu *user_id* sebagai *primary key*, *user_nama*, *user_pass*, *user_email* dan *user_nohp*. Semua data pada tabel *user* diisi oleh administrator.



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	user_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	user_nama	varchar(128)	latin1_swedish_ci		Yes	None			Change Drop More
3	user_pass	varchar(128)	latin1_swedish_ci		Yes	None			Change Drop More
4	user_email	varchar(128)	latin1_swedish_ci		Yes	None			Change Drop More
5	user_nohp	varchar(12)	latin1_swedish_ci		No	None			Change Drop More

Gambar 4.1 Implementasi Tabel *User*

4.1.2 Implementasi Tabel *Booking*

Tabel *booking* terdiri dari 14 *fields* yaitu *booking_id* sebagai *primary key*, *booking_nama*, *booking_email*, *booking_no_hp*, *booking_room_id*, *booking_harga*, *booking_jumlah_jam*, *booking_waktu*, *booking_tanggal*, *booking_ket*, *booking_total*, *booking_deposit*, *booking_Bukti_Dp* dan

booking_created. Semua data pada tabel *booking* merupakan data pengunjung yang melakukan *booking*.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	booking_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	booking_name	varchar(128)	latin1_swedish_ci		Yes	None			Change Drop More
3	booking_email	varchar(128)	latin1_swedish_ci		Yes	None			Change Drop More
4	booking_no_hp	varchar(12)	latin1_swedish_ci		Yes	None			Change Drop More
5	booking_room_id	int(11)			Yes	None			Change Drop More
6	booking_harga	int(11)			No	None			Change Drop More
7	booking_jumlah_jam	smallint(6)			Yes	None			Change Drop More
8	booking_waktu	time			Yes	None			Change Drop More
9	booking_tanggal	date			No	None			Change Drop More
10	booking_ket	varchar(255) Latin1_SF Bulet Keti_SF Bulet	latin1_swedish_ci		Yes	None			Change Drop More
11	booking_total	int(11)			No	None			Change Drop More
12	booking_deposit	int(11)			No	None			Change Drop More
13	booking_Bukti_Op	text	latin1_swedish_ci		No	None			Change Drop More
14	booking_created	timestamp			No	CURRENT_TIMESTAMP			Change Drop More

Gambar 4.2 Implementasi Tabel *Booking*

4.1.3 Implementasi Tabel *Room*

Tabel *room* terdiri dari 9 *fields* yaitu *Room_Id* sebagai *primary key*, *Room_Name*, *Room_Desc*, *Room_Img*, *Room_Capasitas*, *Room_Class*, *Room_Price*, *Room_SpecialPrice* dan *Room_BusinessPrice*. Semua data pada tabel *room* diisi oleh administrator.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Room_Id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	Room_Name	varchar(54)	latin1_swedish_ci		Yes	None			Change Drop More
3	Room_Desc	text	latin1_swedish_ci		Yes	None			Change Drop More
4	Room_Img	text	latin1_swedish_ci		Yes	None			Change Drop More
5	Room_Capasitas	int(11)			Yes	None			Change Drop More
6	Room_Class	enum('Room', 'Paket')	latin1_swedish_ci		No	None			Change Drop More
7	Room_Price	int(11)			Yes	None			Change Drop More
8	Room_SpecialPrice	int(11)			Yes	None			Change Drop More
9	Room_BusinessPrice	int(11)			No	None			Change Drop More

Gambar 4.3 Implementasi Tabel *Room*

4.1.4 Implementasi Tabel Menu

Tabel menu terdiri dari 7 *fields* yaitu Menu_id sebagai *primary key*, Menu_Category, Menu_Class, Menu_Desc, Menu_Name, Menu_tipe dan Menu_Price. Semua data pada tabel menu diisi oleh administrator.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Menu_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	Menu_Category	varchar(255)	latin1_swedish_ci		Yes	None			Change Drop More
3	Menu_Class	varchar(255)	latin1_swedish_ci		Yes	None			Change Drop More
4	Menu_Desc	varchar(255)	latin1_swedish_ci		Yes	None			Change Drop More
5	Menu_Name	varchar(255)	latin1_swedish_ci		Yes	None			Change Drop More
6	Menu_tipe	enum('Makanan', 'Minuman', 'Paket')	latin1_swedish_ci		No	None			Change Drop More
7	Menu_Price	int(11)			Yes	None			Change Drop More

Gambar 4.4 Implementasi Tabel Menu

4.1.5 Implementasi Tabel Event

Tabel *event* terdiri dari 5 *fields* yaitu Event_id sebagai *primary key*, Event_Name, Event_img, Event_Desc dan Event_Date. Semua data pada tabel *event* diisi oleh administrator.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Event_id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	Event_Name	text	latin1_swedish_ci		Yes	None			Change Drop More
3	Event_img	text	latin1_swedish_ci		Yes	None			Change Drop More
4	Event_Desc	text	latin1_swedish_ci		Yes	None			Change Drop More
5	Event_Date	date			Yes	None			Change Drop More

Gambar 4.5 Implementasi Tabel Event

4.2 Implementasi Program

Setelah selesai merancang sistem dan mengimplementasikan basis data, maka selanjutnya hasil rancangan tersebut diimplementasikan pada keadaan sesungguhnya dengan menggunakan bahasa pemrograman PHP. Adapun tampilan aplikasinya sebagai berikut :

4.2.1 Halaman Utama

Pada saat pengunjung menjalankan aplikasi *booking room karaoke*, pengunjung akan masuk pada halaman utama yakni beranda yang berisi daftar menu utama yang ada pada sistem. Di bawah merupakan tampilan halaman utama.



Gambar 4.6 Halaman Utama

Tampilan photo gallery seperti pada gambar diatas menggunakan `<a href="gallery.php">Photo Gallery</a>`, sehingga terhubung pada file `gallery.php` agar dapat menampilkan kumpulan foto. Halaman utama dibuat dengan *coding* sebagai berikut :

```
<h2>Tentang</h2>
  <p>
    Happy Puppy adalah Karaoke Keluarga pertama yang didirikan di Indonesia
    sejak tahun 1992. Semua outlet Happy Puppy didesain dengan konsep terang
    benderang yang cocok untuk hiburan keluarga. Semua ruangan dilengkapi dengan
    jendela besar tembus pandang.
  <div id="gallery">
    <h2><a href="gallery.php">Photo Gallery</a></h2></div>
  <div id="footer">
    <?php
      include './footer.php';?>
  </div>
```

4.2.2 Halaman Menu *Room* dan Paket

Pada halaman menu *room* dan paket berisi jenis *room*, kapasitas setiap *room*, deskripsi *room* serta harga masing-masing *room*. Untuk menampilkan *room* menggunakan `<?php ShowListRoom();?>` sehingga daftar *room* akan dimunculkan, sedangkan `<?php ShowListPaket();?>` digunakan untuk menampilkan seluruh daftar paket yang ada. Dibawah ini merupakan tampilan halaman menu *room* dan paket.



Gambar 4.7 Halaman Menu *Room* dan *Paket*

Halaman menu *room* dan *paket* ditampilkan dengan *coding* sebagai berikut:

```
<div class="body">
  <?php
    include './navigation-side.php';
    include './connect.php';

    ?>
    <div id="rooms">
      <h2>Rooms</h2>
      <?php
        ShowListRoom();
      ?>
    </div>
  </div>
```

4.2.3 Halaman Menu *Gallery*

Halaman menu *gallery* berisi kumpulan gambar yang berkaitan dengan karaoke Happy Puppy Sudirman Kupang sebagai media promosi kepada pengunjung untuk lebih mengetahui apa saja yang tersedia. Implementasi dari halaman menu *gallery* dapat dilihat pada gambar 4.8.



Gambar 4.8 Halaman Menu *Gallery*

Halaman menu *gallery* menggunakan tag `<div id="gallery">` untuk mengelompokkan gambar apa saja yang muncul pada halaman tersebut, sedangkan `<a href="images\INSIDE-3.jpeg">` digunakan untuk menampilkan gambar. Halaman ini dibuat dengan *coding* sebagai berikut :

```

<div class="body">
    <?php include './navigation-side.php';?>
    <div id="gallery">
        <h2>Photo Gallery</h2>
        <ul>
            <li><a href="images\INSIDE-3.jpeg"></a></li>
            <li><a href="images\INSIDE-3.jpeg"></a></li>
            <li><a href="images\INSIDE-3.jpeg"></a></li>
            <li><a href="images\INSIDE-3.jpeg"></a></li>
            <li><a href="images\PEOPLE-1.jpeg"></a></li>
            <li><a href="images\FRONT-1.jpeg"></a></li>
        </ul>
    </div>
</div>

```

4.2.4 Halaman Menu

Halaman menu berisi daftar menu makanan apa saja yang tersedia serta harga pada karaoke Happy Puppy Sudirman Kupang. Implementasi halaman menu dapat dilihat pada gambar 4.9.



Gambar 4.9 Halaman Menu

Untuk menampilkan menu menggunakan `<?php ShowListMenu("Makanan");?>` sehingga daftar menu dapat dimunculkan. Tampilan halaman menu dibuat dengan *coding* sebagai berikut :

```
<div class="body">
    <?php
        include './navigation-side.php';
        include './connect.php';
    ?>
    <div>
        <h2>Menu</h2>
        <?php ShowListMenu("Makanan"); ?>
    </table>
    </div>
</div>
```

4.2.5 Halaman *Drinks*

Halaman *drinks* berisi daftar minuman apa saja yang tersedia serta harga pada karaoke Happy Puppy Sudirman Kupang. Tampilan halaman *drinks* dapat dilihat pada gambar 4.10.



Gambar 4.10 Halaman *Drinks*

Untuk menampilkan *drinks* menggunakan `<?php ShowListMenu("Minum");?>` sehingga daftar minuman dapat dimunculkan. Tampilan halaman *drinks* dibuat dengan *coding* sebagai berikut :

```
<div class="body">
    <?php
        include './navigation-side.php';
        include './connect.php'
    ?>
    <div>
        <h2>Drinks</h2>
        <?php ShowListMenu("Minum"); ?>
    </div>
</div>
```

4.2.6 Halaman *Event*

Halaman ini berisi daftar *event* apa saja yang akan diselenggarakan serta deskripsi *event* tersebut. Tampilan halaman *event* dapat dilihat pada gambar 4.11. `<?php ShowListEvent();?>` digunakan untuk menampilkan daftar event yang sedang berjalan.



Gambar 4.11 Halaman *Event*

Tampilan halaman *event* dibuat dengan *coding* sebagai berikut :

```
<div class="body">
  <?php
    include './navigation-side.php';
    include './connect.php'
  ?>
  <div id="events">
    <h2>Events</h2>
    <?php ShowListEvent(); ?>
  </div>
```

4.2.7 Halaman *Booking*

Halaman *booking* berisi 3 sub menu yaitu *booking room*, *booking paket*, cari *booking*. Tampilan halaman *booking* dapat dilihat pada gambar 4.12.



Gambar 4.12 Halaman *Booking*

Tampilan halaman booking terdiri dari 3 tombol. Tombol untuk menampilkan *form booking room* menggunakan `onclick="document.getElementById('Room'), form booking` paket ditampilkan kembali menggunakan `onclick="document.getElementById('paket'),` sedangkan `onclick="document.getElementById('cari')` digunakan untuk menampilkan *form* pencarian bagi *user*. Halaman ini dibuat dengan *coding* sebagai berikut :

```
<center>
<button onclick="document.getElementById('Room').style.display='block'"
style="width:auto;">
    Booking Room</button>
<button onclick="document.getElementById('Paket').style.display='block'"
style="width:auto;">
    Booking Paket</button>
<button onclick="document.getElementById('Cari').style.display='block'"
style="width:auto;">
    Cari Booking</button>
</center>
```

4.2.8 Halaman *Booking Room*

Pada halaman ini tersedia *field-field* yang akan diisi oleh pengunjung agar dapat melakukan *booking* sesuai jenis *room* yang diinginkan. Maka, `$_POST` digunakan agar dapat menampung data hasil *inputan* dan di

simpan pada basis data. Tampilan halaman *booking room* dapat dilihat pada gambar 4.13.



Gambar 4.13 Halaman *Booking Room*

Tampilan halaman *booking room* dibuat dengan *coding* sebagai berikut :

```
foreach ($POST as $key => $value) {
    debug_to_console($key,$value); }
$RoomID = $_POST['Room_Booking'];
$RoomHr = $_POST['Jam_Booking'];
$RoomRow = getRoomByID($RoomID);
$Room_Name = $RoomRow["Room_Name"];
//Logika Harga
$Jam = $_POST['Waktu_Booking'];
$Tanggal = $_POST['Tanggal_Booking'];
$Harian = getWeekday($Tanggal);
debug_to_console("Jam",$Jam);
debug_to_console("Tanggal",$Tanggal);
debug_to_console("Harian",$Harian);
if ($Harian == 1) {
    debug_to_console("Hari","Senin");
    if(strtotime($Jam)<=strtotime('17:00:00')) {
        debug_to_console("Room Price","Normal");
        $Room_Price = $RoomRow["Room_Price"];
    } else {
        debug_to_console("Room_Price","Bussines");
        $Room Price = $RoomRow["Room Business Price"]; }
}
```

4.2.9 Halaman *Booking* Paket

Halaman *booking* paket tersedia *field-field* yang akan diisi oleh pengunjung agar dapat melakukan *booking* sesuai paket yang diinginkan.

Untuk itu, \$ POST digunakan agar dapat menampung data hasil *inputan*

dan di simpan pada basis data. Serta, `$Room_Price = $RoomRow["Room_Price"]` digunakan untuk menampilkan daftar harga setiap *room*. Tampilan halaman *booking* paket dapat dilihat pada gambar 4.14.



Gambar 4.14 Halaman *Booking* Paket

Tampilan Halaman *booking* paket dibuat dengan *coding* sebagai berikut :

```
<div id="header"> <a href="index.php" id="logo"></a>
    <?php
        include './connect.php';
        foreach ($_POST as $key => $value) {
            debug_to_console($key,$value); }
        $RoomID = $_POST['Room_Booking'];
        $RoomHr = $_POST['Jam_Booking'];
        $RoomRow = getRoomByID($RoomID);
        $Room_Name = $RoomRow['Room_Name'];
        $Room_Price = $RoomRow['Room_Price'];
        // echo "$RoomRow['Room_Price']";
    ?>
</div>
```

4.2.10 Halaman Cari *Booking*

Halaman ini tersedia *field* yang akan diisi oleh pengunjung untuk dapat melakukan pencarian *booking* yang telah dibuat agar dapat melakukan konfirmasi pembayaran. Halaman ini menggunakan `onclick = "document.getElementById('Cari')"` agar melakukan pencarian

sesuai kata kunci yang diinginkan *user*. Tampilan halaman ini dapat dilihat pada gambar 4.15.



Gambar 4.15 Halaman Cari *Booking*

Tampilan halaman cari *booking* dibuat dengan *coding* sebagai berikut :

```
<h1>Pencarian Booking </h1>
<p>Silahkan isi form ini dengan benar</p>
<hr>
<label for="email"><b>Email atau NomorHP</b></label>
<input type="text" placeholder="Email atau NomorHP" name="Cari"
value="" required>
<div class="clearfix">
  <button type="button"
onclick="document.getElementById('Cari').style.display='none'"
class="cancelbtn">Batal</button>
  <button type="submit" class="signupbtn">Cari Booking</button>
</div>
```

4.2.11 Halaman Pembatalan *Booking*

Halaman ini tersedia *field* yang akan diisi oleh pengunjung untuk dapat melakukan pembatalan *booking* yang telah dibuat. *Booking room* dan paket *buffet* hanya dapat dibatalkan maksimal 1x24 jam setelah pengunjung melakukan pembayaran uang muka. Pembayaran uang muka untuk *booking room* tidak dapat dikembalikan kepada pengunjung bila melakukan pembatalan. Sedangkan, uang muka untuk paket *buffet* hanya dapat dikembalikan 50% dari total uang muka yang dibayarkan. Tampilan halaman ini dapat dilihat pada gambar 4.16.



Gambar 4.16 Halaman Pembatalan *Booking*

Halaman batal *booking* menggunakan `onclick = "document.getElementById('id02')"` agar *user* dapat menemukan id *booking* yang sesuai agar dapat melakukan pembatalan. Tampilan halaman diatas dibuat dengan *coding* sebagai berikut :

```
<h1>Membatalkan Booking</h1>
    <p>Silahkan isi form ini dengan benar untuk membatalkan booking.</p>
    <hr>
    <input type="hidden" name="ID_Booking" value="<?php echo $_GET["id"]; ?>"
    required>
    <label for="email"><b>Email</b></label>
    <input type="text" placeholder="Email" name="Email_Booking" value=""
    required>
    <label for="email"><b>Nomor HP</b></label>
    <input type="text" placeholder="Nomor HP" name="NoHP_Booking" value=""
    required>
    <div class="clearfix">
        <button type="button"
        onclick="document.getElementById('id02').style.display='none'"
        class="cancelbtn">Kembali</button>
        <button type="submit" class="signupbtn">Membatalkan Booking</button>
    </div>
```

4.2.12 Halaman *Login Admin*

Pada halaman ini admin melakukan *login* ke sistem agar dapat meng-*input*, menyimpan, melihat, mengubah dan menghapus data *booking room* karaoke, paket, jenis *room*, harga sewa *room*, paket *buffet*, promo yang sedang berjalan dan jadwal *room* ter-*update*. `SetUpUserAuthorization()` digunakan agar sistem dapat melakukan konfirmasi terhadap data admin yang di-*input* dengan data yang

sudah ada pada basis data. Tampilan halaman ini dapat dilihat pada gambar 4.17.



Gambar 4.17 Halaman *Login* Admin

Tampilan halaman *login* admin dibuat dengan *coding* sebagai berikut :

```
function OnAfterLogin($userName, $connection, &$scanLogin, &$errorMessage)
{
    function OnBeforeLogout($userName, $connection){

        SetupUserAuthorization();

        $page = new LoginPage(
            'event.php',
            dirname(__FILE__),
            GetApplication()->GetUserAuthentication(),
            MySqlConnectionFactory::getInstance(),
            Captions::getInstance('UTF-8')
        );
    }
}
```