

BAB IV

IMPLEMENTASI SISTEM

4.1 Implementasi Database

Software pengolahan *database* yang digunakan dalam implementasi *database* yaitu *MySQL* dengan bahasa pemrograman *PHP*. Berikut tabel-tabel yang dibangun menjadi *database* pada sistem ini adalah :

1) Tabel *Admin*

Tabel ini memiliki 3 *field* diantaranya *id_admin* sebagai *primary key*, *username* dan *password*.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_admin	int(11)			No	None	AUTO_INCREMENT	Change Drop Primary
2	username	varchar(100)			No	None		Change Drop Primary
3	password	varchar(100)			No	None		Change Drop Primary

Gambar 4.1 Tabel *Admin*

2) Tabel *Berita*

Tabel ini memiliki 5 *field* diantaranya *id_berita* sebagai *primary key*, *judul_berita*, *isi_berita*, *waktu*, dan *gambar_berita*.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_berita	int(11)			No	None	AUTO_INCREMENT	Change Drop
2	judul_berita	varchar(100)			No	None		Change Drop
3	isi_berita	text			No	None		Change Drop
4	waktu	date			No	None		Change Drop
5	gambar_berita	varchar(100)			No	None		Change Drop

Gambar 4.2 Tabel *Berita*

3) Tabel Gejala

Tabel ini memiliki 2 *field* diantaranya kode_gejala sebagai *primary key* dan nama_gejala.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	kode_gejala	varchar(4)			No	None		Change Drop
2	nama_gejala	varchar(100)			No	None		Change Drop

Gambar 4.3 Tabel Gejala

4) Tabel Halaman

Tabel ini memiliki 4 *field* diantaranya id_halaman sebagai *primary key*, nama_halaman, judul_halaman dan detail_halaman.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_halaman	int(11)			No	None	AUTO_INCREMENT	Change Drop
2	nama_halaman	varchar(100)			No	None		Change Drop
3	judul_halaman	varchar(100)			No	None		Change Drop
4	detail_halaman	text			No	None		Change Drop

Gambar 4.4 Tabel Halaman

5) Tabel Kontak

Tabel ini memiliki 6 *field* diantaranya id_kontak sebagai *primary key*, nama_lengkap, email, isi_pesan, waktu, status_kontak.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_kontak	int(11)			No	None	AUTO_INCREMENT	Change Drop
2	nama_lengkap	varchar(100)			No	None		Change Drop
3	email	varchar(100)			No	None		Change Drop
4	isi_pesan	text			No	None		Change Drop
5	waktu	datetime			No	None		Change Drop
6	status_kontak	enum('baru', 'lama')			No	None		Change Drop

Gambar 4.5 Tabel Kontak

6) Tabel Penyakit

Tabel ini memiliki 4 *field* diantaranya kode_penyakit sebagai *primary key*, nama_penyakit, deskripsi dan solusi.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	kode_penyakit	varchar(4)			No	None		Change Drop
2	nama_penyakit	varchar(50)			No	None		Change Drop
3	deskripsi	text			No	None		Change Drop
4	solusi	text			No	None		Change Drop

Gambar 4.6 Tabel Penyakit

7) Tabel Penyakit dan Gejala

Tabel ini memiliki 4 *field* diantaranya id_penyakit_gejala sebagai *primary key*, kode_penyakit, kode_gejala, bobot dan *primary key* tabel ini adalah *id_penyakit_gejala*.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id_penyakit_gejala	int(11)			No	None	AUTO_INCREMENT	Change Drop
2	kode_penyakit	varchar(4)			No	None		Change Drop
3	kode_gejala	varchar(4)			No	None		Change Drop
4	bobot	int(11)			No	None		Change Drop

Gambar 4.7 Tabel Penyakit dan Gejala

8) Tabel Temp Gejala

Tabel ini memiliki 2 *field* diantaranya kode_gejala sebagai *primary key* dan bobot.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	kode_gejala	varchar(4)			No	None		Change Drop
2	bobot	int(11)			No	None		Change Drop

Gambar 4.8 Tabel Temp Gejala

9) Tabel Temp Hasil

Tabel ini memiliki 6 *field* diantaranya kode_penyakit, kode_gejala, x, x_y, x_y² dan *primary key* tabel ini adalah kode_penyakit.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	kode_penyakit	varchar(4)			No	None		Change Drop
2	kode_gejala	varchar(4)			No	None		Change Drop
3	x	int(11)			No	None		Change Drop
4	y	int(11)			No	None		Change Drop
5	x_y	int(11)			No	None		Change Drop
6	x_y2	int(11)			No	None		Change Drop

Gambar 4.9 Tabel Temp Hasil

10) Tabel Temp Hasil2

Tabel ini memiliki 2 *field* diantaranya *kode_penyakit* dan *nilai*.

Primary key tabel ini adalah *kode_penyakit*.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	kode_penyakit	varchar(4)			No	None		 Change  Drop
2	nilai	float			No	None		 Change  Drop

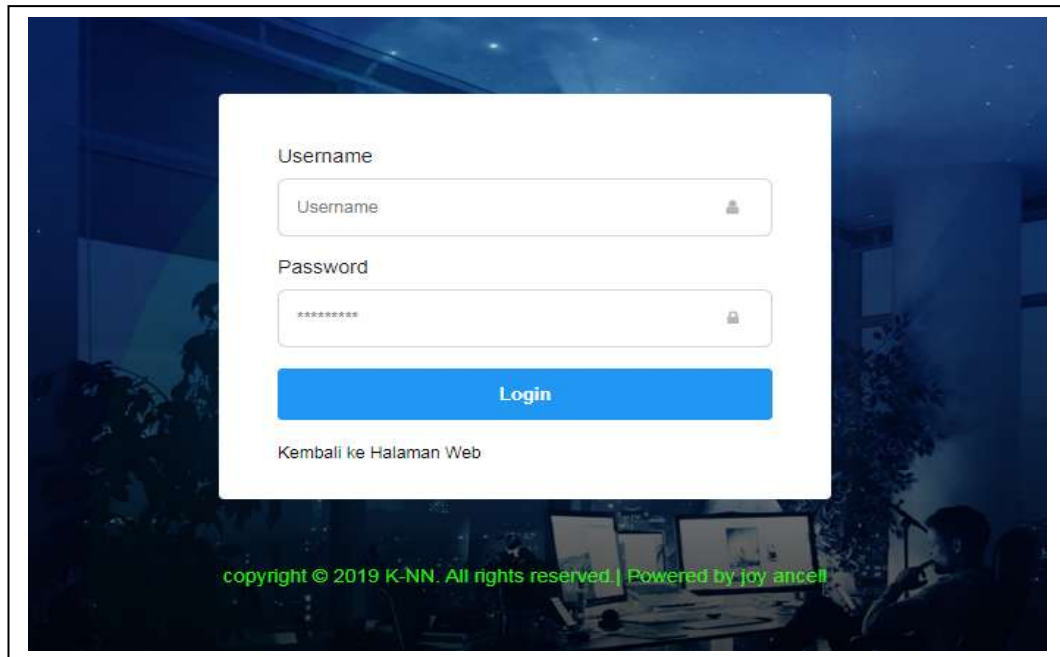
Gambar 4.10 Tabel Temp Hasil2

4.2 Implementasi Program

Untuk mengakses aplikasi, terlebih dahulu membuka *Mozilla firefox*, *Google chrome*, *Internet Explore*, kemudian masuk ke direktori aplikasi. Berikut adalah tampilan *Implementasi program*.

1) Tampilan Halaman *Login Administrator*

Halaman ini berisi *form login* admin yang akan ditemui pertama kali saat membuka *web* ini. Admin wajib memasukan *username* dan *password* untuk dapat masuk ke halaman ini. Berikut adalah tampilan halaman *login*.



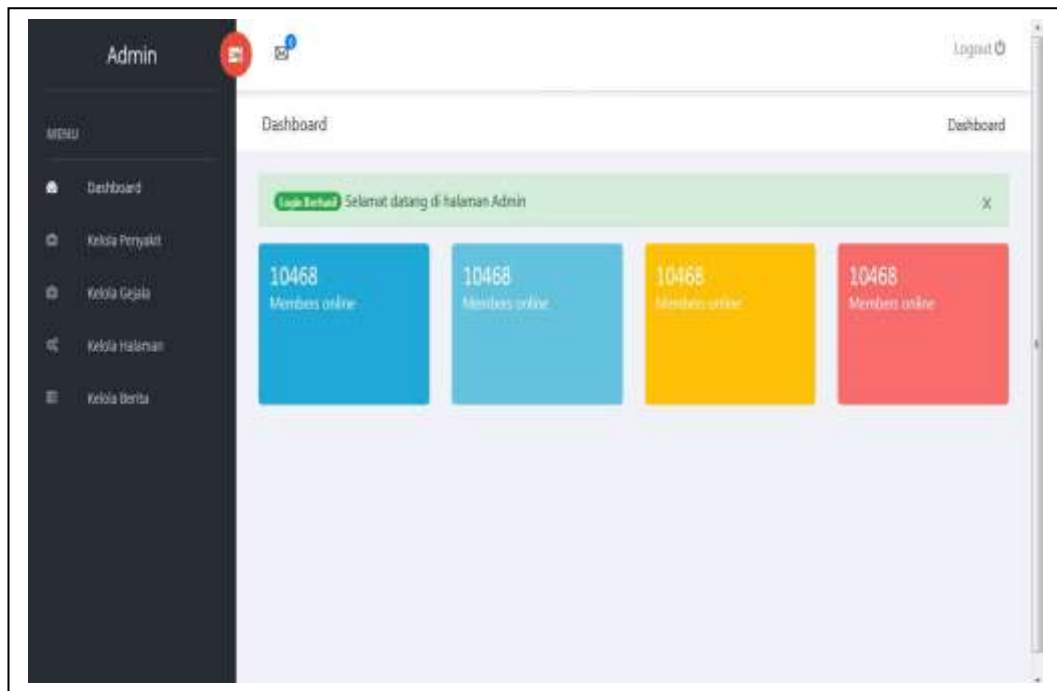
Gambar 4.11 Tampilan Halaman Login Admin

```
$username = $_POST['username'];
$password = md5($_POST['password']);
$ccek = mysqli_query($koneksi, " SELECT * FROM admin WHERE username =
'$username' AND password = '$password' ");
$hssl = mysqli_num_rows($ccek);
if ($hssl == 1) {
    $_SESSION['username_knn'] = $username
}
```

Script di atas berfungsi untuk menampilkan halaman *login* dengan menggunakan *method* *\$_POST* untuk mengirim data *username* (*user*) dan *password* (*pass*) saat menekan tombol *login*. Jika data yang dikirim admin aktif maka admin akan masuk pada halaman admin dan jika yang dikirim tidak terdaftar atau belum aktif maka admin akan mendapat pesan *login* gagal.

2) Tampilan Halaman *Dashboard* Admin

Halaman *dashboard* admin ini terdapat *link* agar admin dapat mengelola data-data mulai dari kelola penyakit, kelola gejala, kelola halaman dan kelola berita. Berikut adalah tampilan halaman utama pada admin.



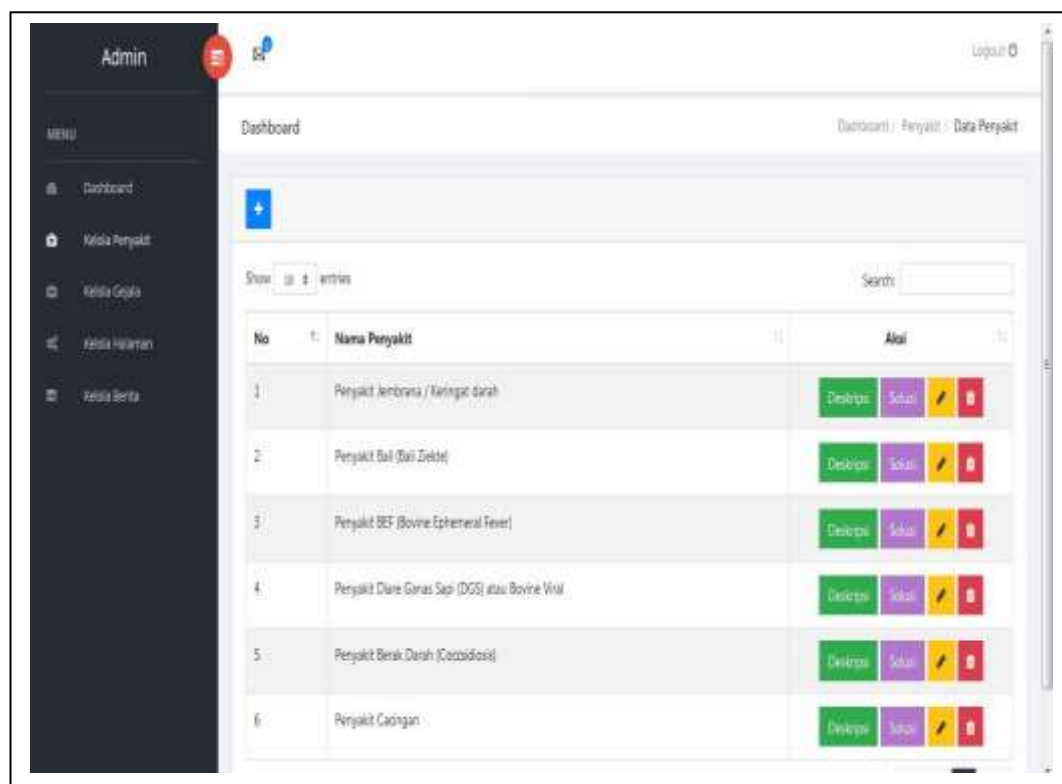
Gambar 4.12 Tampilan Halaman *Dashboard* Admin

```
if(@$_SESSION['admin']) {  
if(@$_GET['hal'] == "")  
{ $sql_admin = mysqli_query($koneksi, "SELECT * FROM tb_admin WHERE  
id_admin = '$_SESSION[admin]'");
```

Script di atas di deklarasikan awalnya dengan menggunakan *SESSION* yaitu dengan menggunakan *SESSION_START()* dan akan menset nilai variabelnya dengan menggunakan perintah *\$_SESSION*.

3) Tampilan Halaman Kelola Penyakit

Pada tampilan ini akan menampilkan seluruh data penyakit yang ada pada sistem yang sudah di *input* oleh admin. Di sini admin bisa menambahkan data penyakit, data deskripsi penyakit, data solusi penyakit, mengedit data penyakit dan juga menghapus data penyakit yang ada pada sistem. Berikut adalah tampilan halaman kelola penyakit.



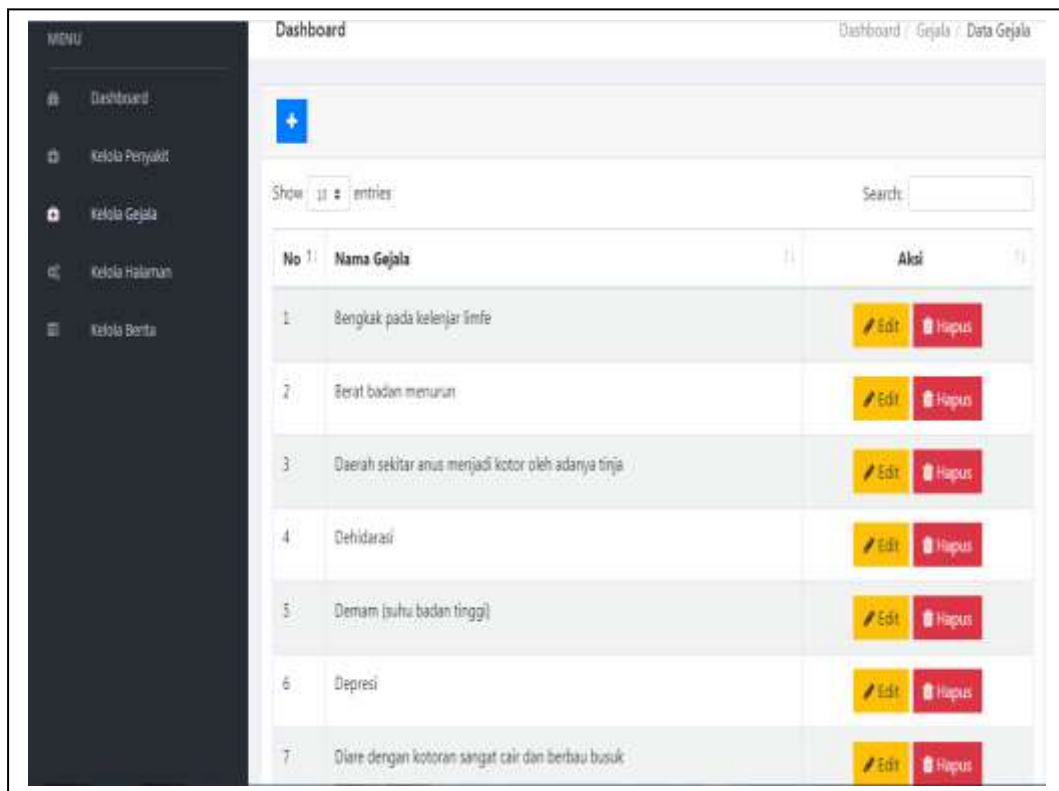
Gambar 4.13 Tampilan Halaman Kelola Penyakit

```
<?php
$no = 1;
$query = mysqli_query($koneksi, " SELECT * FROM penyakit ");
while ($hasil = mysqli_fetch_array($query)) {
?>
<tr>
<td><?php echo "$no"; ?></td>
<td><?php echo "$hasil[nama_penyakit]"; ?></td>
```


Script di atas menggunakan fungsi *\$query* yang berfungsi untuk mengambil data penyakit dari tabel penyakit pada *database* yang telah di *input* oleh admin dan menampilkan data penyakit pada tabel yang ada pada halaman kelola penyakit.

4) Tampilan Halaman Kelola Gejala

Pada tampilan ini akan menampilkan seluruh data gejala yang ada pada sistem yang sudah di *input* oleh admin. Di sini admin bisa menambahkan data gejala, mengedit data gejala dan menghapus data gejala yang ada pada sistem. Berikut adalah tampilan halaman kelola gejala.



No	Nama Gejala	Aksi
1	Bengkak pada kelenjar limfe	Edit Hapus
2	Berat badan menurun	Edit Hapus
3	Daerah sekitar anus menjadi kotor oleh adanya tinja	Edit Hapus
4	Dehidrasi	Edit Hapus
5	Demam (suhu badan tinggi)	Edit Hapus
6	Depresi	Edit Hapus
7	Diare dengan kotoran sangat cair dan berbau busuk	Edit Hapus

Gambar 4.14 Tampilan Halaman Kelola Gejala

```

<?php
$no = 1;
$query = mysqli_query($koneksi, " SELECT * FROM gejala ");
while ($hasil = mysqli_fetch_array($query)) {
?>
<tr>
<td><?php echo "$no"; ?></td>
<td><?php echo "$hasil[nama_gejala]"; ?></td>

```

Script di atas menggunakan fungsi *\$query* yang berfungsi untuk mengambil data dari tabel gejala pada *database* dan menampilkan data gejala pada tabel yang ada pada halaman kelola gejala.

5) Tampilan Halaman Tambah Data Gejala

Pada tampilan ini admin akan menambahkan data gejala bila terdapat gejala baru dengan memasukan nama gejala dan memilih penyakit yang mempunyai gejala tersebut. Berikut adalah tampilan halaman tambah data gejala.

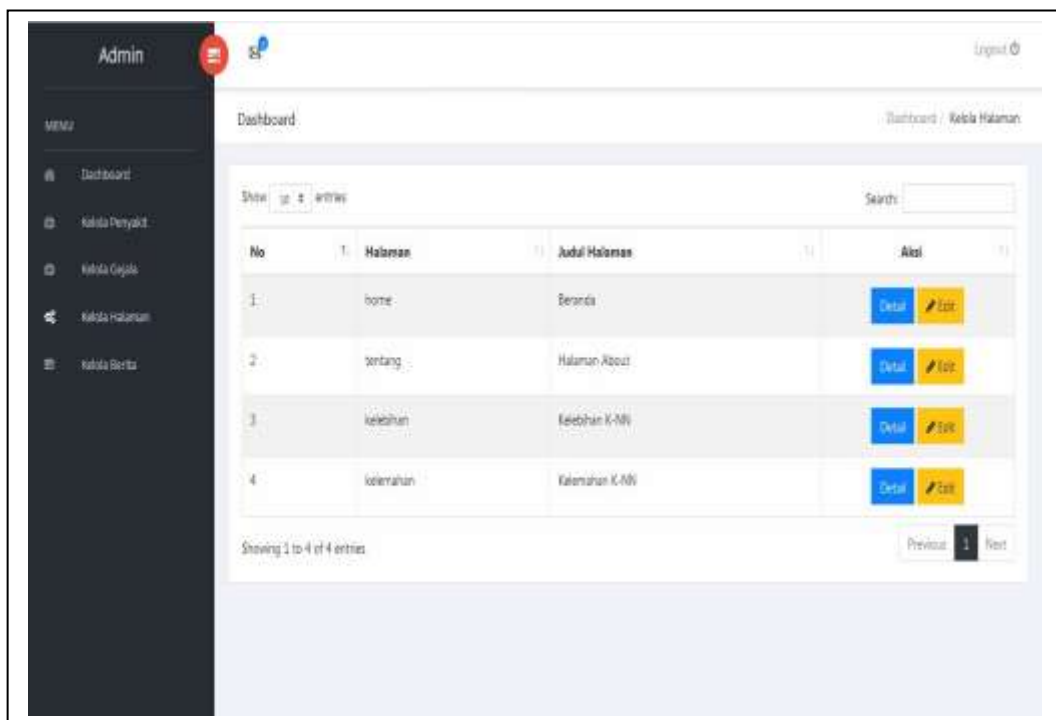
Gambar 4.15 Tampilan Halaman Tambah Data Gejala

```
$simpan_gejala=mysqli_query($koneksi,"INSERT INTO gejala(kode_gejala,
nama_gejala) VALUES ('$kode','$nama') ");
if($simpan_gejala){for($i=0;$i<$x;$i++){
$simpan=mysqli_query($koneksi,"INSERT INTO
penyakit_gejala(kode_penyakit,kode_gejala,bobot)VALUES
('$pilihan[$i]','$kode','1') ");}}
```

Script di atas berfungsi untuk menambah data gejala baru ke dalam tabel gejala pada *database* menggunakan pernyataan *INSERT INTO* dan fungsi *VALUES* akan memberikan bobot nilai = 1 pada tabel penyakit_gejala pada *database* sesuai penyakit pilihan.

6) Tampilan Halaman Kelola Halaman

Pada tampilan ini akan menampilkan beberapa *form* menu yang akan ditampilkan pada halaman *user*. Di sini admin bisa mengedit informasi dari masing-masing menu yang akan ditampilkan pada halaman *user*. Berikut adalah tampilan halaman kelola halaman.



Gambar 4.16 Tampilan Halaman Kelola Halaman

```
<?php
$no = 1;
$query = mysqli_query($koneksi, " SELECT * FROM halaman ");
while ($hasil = mysqli_fetch_array($query)) {
?>
<tr>
<td><?php echo "$no"; ?></td>
<td><?php echo "$hasil[nama_halaman]"; ?></td>
<td><?php echo "$hasil[judul_halaman]"; ?></td>
```

Script di atas menggunakan fungsi *mysqli_fetch_array* yang berfungsi untuk mengambil data dari tabel halaman pada *database* dan menampilkan data halaman pada tabel yang ada pada menu kelola halaman di halaman *user*.

7) Tampilan Halaman Kelola Berita

Pada tampilan ini akan menampilkan seluruh data berita yang ada pada sistem yang sudah di *input* oleh admin. Di sini admin bisa mengedit data berita dan menghapus data berita yang ada pada sistem. Berikut adalah tampilan halaman kelola berita.



Gambar 4.17 Tampilan Halaman Kelola Berita

```

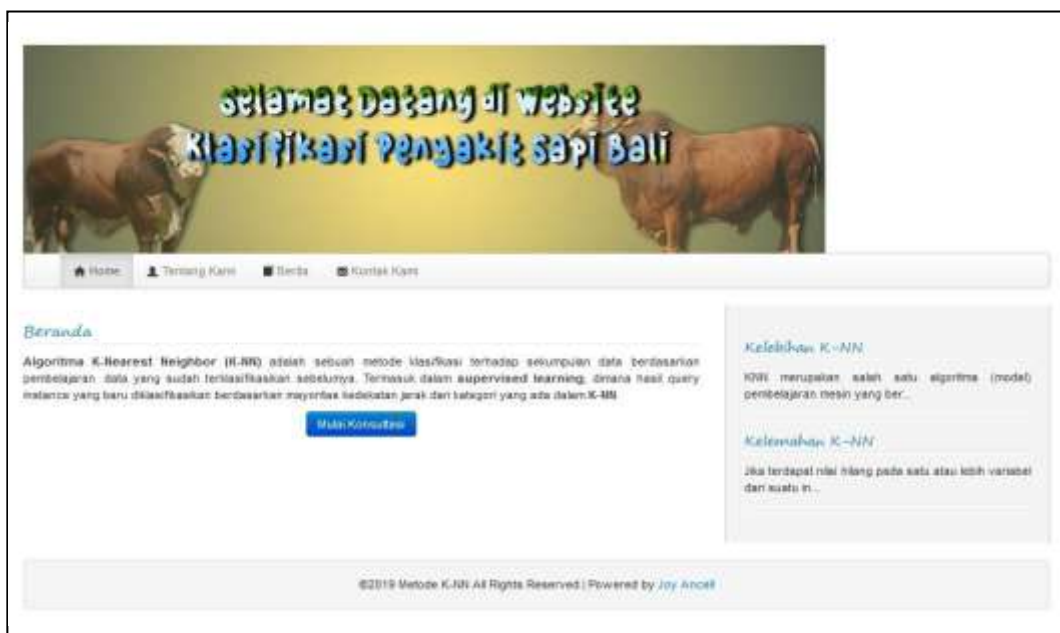
<?php
$no = 1;
$query = mysqli_query($koneksi, " SELECT * FROM berita order by id_berita desc
");
while ($hasil = mysqli_fetch_array($query)) {
?>
<tr>
<td><?php echo "$no"; ?></td>
<td><?php echo "$hasil[judul_berita]"; ?></td>

```

Script di atas menggunakan fungsi *\$query* yang berfungsi untuk mengambil data dari tabel berita pada *database* dan menampilkan data berita pada tabel yang ada pada halaman kelola berita.

8) Tampilan Halaman *Home*

Halaman *home* merupakan halaman tampilan awal dari isi *website* klasifikasi penyakit sapi bali, dengan tombol konsultasi *user* akan diarahkan untuk memilih gejala pada halaman konsultasi. Berikut adalah tampilan halaman *home*.



Gambar 4.18 Tampilan Halaman *Home*

```

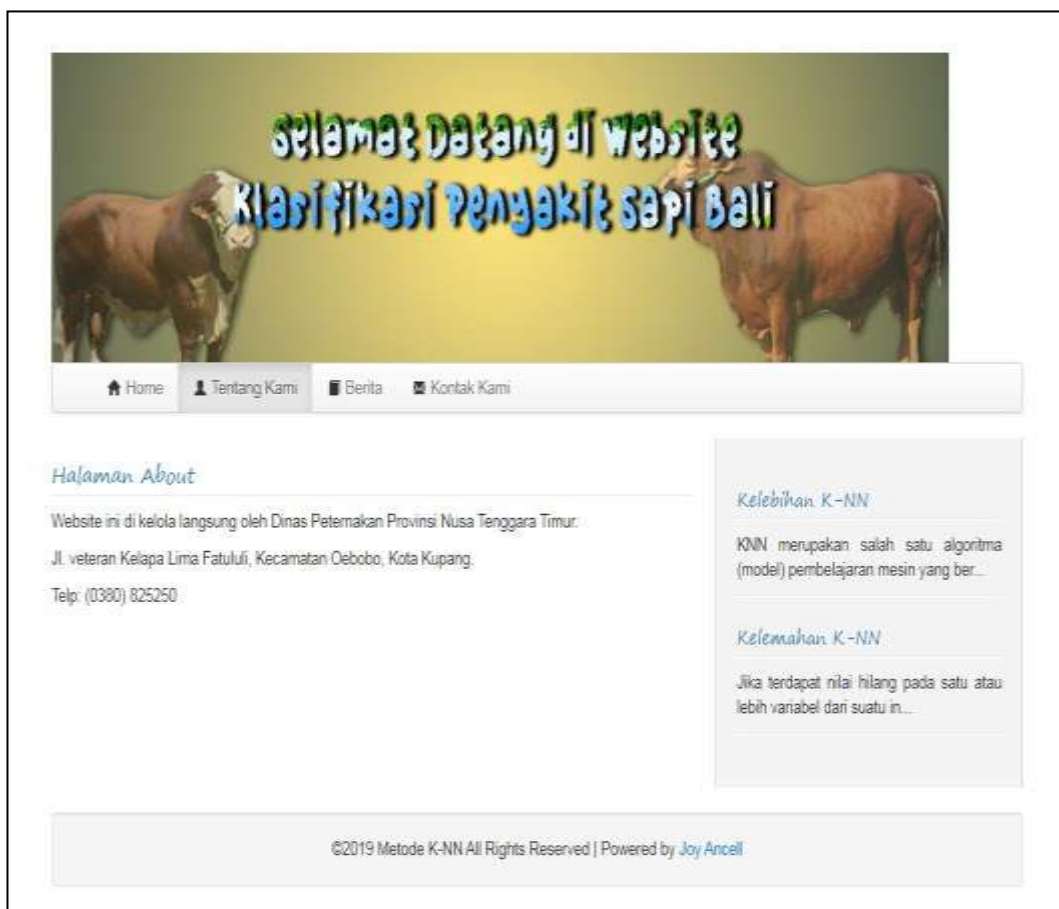
<span class='title'><?php echo "$hasil[judul_halaman]"; ?></span>
<p style="text-align: justify;"><?php echo "$hasil[detail_halaman]"; ?></p>
<div class="text-center">
    <a href="index.php?mod=konsultasi&page=konsultasi2">
        <button class="btn btn-primary">Mulai Konsultasi</button>
    </a></div>

```

Script di atas berfungsi untuk menampilkan menu-menu pada halaman *home* dan *tag button* berfungsi untuk membuat tombol untuk mengeksekusi perintah. Tombol mulai konsultasi yang pada gambar di atas akan mengarah ke halaman konsultasi pada sistem.

9) Tampilan Halaman Tentang Kami

Halaman ini akan menampilkan informasi yang sudah di masukan oleh admin tentang *website* ini. Berikut adalah tampilan halaman tentang kami.



Gambar 4.19 Tampilan Halaman Tentang Kami

```
<?php
$query = mysqli_query($koneksi, " SELECT * FROM halaman WHERE
nama_halaman = 'tentang' ");
$hasil = mysqli_fetch_array($query);
?>
<span class='title'><?php echo "$hasil[judul_halaman]"; ?></span><hr>
<p><?php echo "$hasil[detail_halaman]"; ?></p>
```

Script di atas menggunakan berfungsi untuk menampilkan data tentang aplikasi pada menu tentang kami. Fungsi *echo* digunakan untuk menampilkan hasil output detail halaman.

10) Tampilan Halaman Berita

Pada halaman ini akan menampilkan berita-berita yang diinput oleh admin. Berikut ini adalah tampilan halaman berita.



Gambar 4.20 Tampilan Halaman Berita

```

<?php $page = (isset($_GET['halaman']))? $_GET['halaman'] : 1;
$limit = 5; // Jumlah data per halamannya
$limit_start = ($page - 1) * $limit;
$sql=mysqli_query($koneksi, "select * from berita order by id_berita DESC LIMIT
$limit_start,$limit");
while($r=mysqli_fetch_array($sql)){
$ttl = tgl_indo($r['waktu']);
$isi_berita = nl2br($r['isi_berita']);

```

Berdasarkan pengkodean di atas, *method GET* digunakan untuk mulai mengambil data sedangkan *GET action* untuk menampilkan *form* berdasarkan *action* yang dipilih.

11) Tampilan Halaman Kontak kami

Halaman ini kami akan menampilkan *form* kontak untuk mengirim pesan ke admin. Pengunjung wajib memasukan nama lengkap dan email jika ingin mengirim pesan. Berikut adalah tampilan halaman kontak kami.

Gambar 4.21 Tampilan Halaman Kontak Kami


```
$simpan = mysqli_query($koneksi, " INSERT INTO kontak(nama_lengkap, email, isi_pesan,waktu,status_kontak)VALUES ('$nama_lengkap','$email','$pesan',now(),'baru') ");
```

Script di atas berfungsi untuk menampilkan tampilan halaman data nama, email dan isi pesan dengan menggunakan *INSERT INTO* untuk mengisi data dan menyimpannya di *database* dengan menggunakan *fungsi value* di dalamnya.

12) Tampilan Halaman Konsultasi

Halaman ini akan menampilkan gejala-gejala dari semua jenis penyakit yang sudah tersedia pada sistem dengan fitur *checkbox*. Berikut adalah tampilan halaman konsultasi.



Gambar 4.22 Tampilan Halaman Konsultasi

```

<?php
while ($hasil = mysqli_fetch_array($query)) { ?> <tr style="height: 50px">
<td>Apakah sapi anda mengalami <?php echo "$hasil[nama_gejala]"; ?></td>
<td><input type="checkbox" name="nama[]" value="<?php echo
$hasil['kode_gejala'] ?>" onclick="enable_text(this.checked)"></td>
<td style="width: 20px"></td> </tr>
<?php } ?>

```

Script di atas menggunakan fungsi *while* yang berfungsi untuk melakukan pengulangan pada teks yang *diinput* dan menggunakan *echo* untuk menampilkan teks ke layar dan ada juga fungsi *tag input checkbox* yang berfungsi untuk membuat *checkbox* atau kotak isian.

13) Tampilan Halaman Hasil Konsultasi

Halaman ini akan menampilkan hasil klasifikasi dari data gejala baru yang dipilih dengan data latih yang sudah ada pada sistem. Halaman ini juga dilengkapi dengan langkah-langkah kerja algoritma k-nn untuk lebih memperjelas hasil klasifikasi pada sistem ini. Sistem akan menampilkan data latih yang sudah tersedia dan data uji / data gejala baru yang di pilih, kemudian sistem akan menghitung kuadrat jarak *euclidean* antara data latih dan data uji yang diberikan menggunakan rumus k-nn seperti pada gambar di atas. Karena nilai parameter $K=1$, sistem akan menggunakan kategori tetangga terdekat dengan data uji sehingga dapat diprediksi kategori objek.


```
<?php
$out = mysqli_query($koneksi, " SELECT * FROM penyakit WHERE kode_penyakit
= '$hasil2[kode_penyakit]' ");
$hsl = mysqli_fetch_array($out);
?>
<b><?php echo "$hsl[nama_penyakit]"; ?></b>
<p style="text-align: justify;"><?php echo "$hsl[deskripsi]"; ?></p> <br>
<span class='title'>Solusi</span> <hr>
```

Script di atas menggunakan fungsi *mysqli_fetch_array* yang berfungsi untuk mengambil baris hasil sebagai *array* asosiatif, *array* numeric atau keduanya. *Mysqli_fetch_array* di atas digunakan untuk menampilkan data penyakit yang di derita sapi bali yang sesuai dengan gejala yang ada pada sistem.