

BAB IV

IMPLEMENTASI DAN PEMBAHASAN

4.1 Implementasi dan Uji Coba Sistem

4.1.1 Metode Inferensi

Metode inferensi adalah mekanisme berfikir dan pola-pola penalaran yang digunakan oleh sistem untuk mencapai suatu kesimpulan. Metode ini akan menganalisa masalah tertentu dan selanjutnya akan mencari jawaban atau kesimpulan yang terbaik. Penalaran dimulai dengan mencocokkan kaidah-kaidah dalam basis pengetahuan dengan fakta-fakta yang ada. Dalam metode Dempster Shafer dikenal *belief* yang merupakan ukuran kekuatan *Evidence* (bukti) dalam mendukung suatu himpunan proposisi. Untuk melakukan inferensi maka diperlukan proses konsultasi untuk mendapatkan suatu jawaban atau kesimpulan.

Untuk menentukan kode gejala dan nilai *belief*, dibuatkan beberapa program untuk menampilkan konsultasi serta menyimpan informasi dari tiap pertanyaan tersebut. Kemudian dibuat variabel untuk mendeklarasikan kode gejala dan nilai *belief* dengan nama variabel `$kode_gejala` dan `$nilai_belief`. Selanjutnya kode gejala disimpan pada tabel `temp_konsultasi` yang nantinya akan dipanggil untuk menampilkan nama gejala pada tabel `gejala`, berdasarkan kode gejala tersebut dan hasilnya disimpan pada variabel `$tampil_nama`. Yang ditunjukkan pada Gambar 4.1.

```

<?php
// Menentukan kode_gejala dan nilai_belief dari pakar
    $kode_gejala = "G012";
    $nilai_belief = 0.2;

// menambahkan kode gejala ke dalam tabel
temp_kode_gejala
include "../.../proses-login/koneksi.php";
$sql_login = "insert into
temp_konsultasi(temp_kode_gejala)
values('$kode_gejala')";
$hasil_login = mysqli_query($konek, $sql_login);
if(!$hasil_login) {
    echo "Gagal Simpan Kode Gejala,
    silahkan diulangi... <br />" ;
    echo mysqli_error($konek);
}

// query untuk menampilkan nama gejala berdasarkan
kode_gejala

    $sql_nama_gejala = "SELECT nama_gejala
FROM gejala WHERE kode_gejala ='$kode_gejala'";
$nama = mysqli_query($konek, $sql_nama_gejala);
if(!$nama) {
    echo "Gagal Simpan Kode Gejala,
    silahkan diulangi... <br />" ;
    echo mysqli_error($konek);
}
$stampil_nama = mysqli_fetch_assoc($nama);
?>

```

Gambar 4.1 Menentukan Kode Gejala dan Nilai Belief

Kemudian dicari penyakit apa saja yang mendukung gejala pada tabel temp_konsultasi, hasilnya akan disimpan pada variabel \$hasil_dugaan, kemudian ditampilkan pada *textarea* seperti yang ditunjukkan pada Gambar 4.2.

```

<!-- Menentukan penyakit -->
<?php
    include "../../proses-login/koneksi.php";
    $sql_dugaan = "SELECT
basis_pengetahuan.kode_penyakit
    AS 'kdPenyakit'
    FROM basis_pengetahuan
    WHERE basis_pengetahuan.kode_gejala='$kode_gejala'";
    $hasil_dugaan = mysqli_query($konek,$sql_dugaan);
?>

<textarea name="dugaan" style="display:none;">
<?php
    while ($dugaan = mysqli_fetch_assoc($hasil_dugaan)) {
        include "../../proses-login/koneksi.php";
        $lihat_dugaan = "SELECT nama_penyakit
        FROM penyakit WHERE kode_penyakit
        ='".$dugaan['kdPenyakit']."'";
        $output = mysqli_query($konek,$lihat_dugaan);
        $stampil = mysqli_fetch_assoc($output);
        echo $stampil['nama_penyakit'].", ";
    }
?></textarea>

```

Gambar 4.2 Menentukan Nama Penyakit

Dalam menampilkan pertanyaan dibuatlah form dengan action yang menuju program tambah_konsultasi.php untuk mengirimkan (*submit*) data. Dalam proses konsultasi ini ditampilkan pertanyaan apakah terdapat gejala tertentu yang dialami, dengan nama gejala yang telah dideklarasikan pada variabel \$stampil_nama. Selain itu juga terdapat data kode gejala dan nilai *belief* yang akan di proses oleh program tambah_konsultasi.php pada saat pengguna memilih tombol “Ya”. Namun jika user memilih tombol “Tidak”, maka akan diarahkan menuju program tidak.php. Seperti yang ditunjukkan pada Gambar 4.3.

```

<form action="tambah_konsultasi.php" method="post">
<!-- Menampilkan pertanyaan gejala -->

    <input type="hidden" name="kode" value=
    "<?php echo $kode_gejala; ?>" />
    <input type="hidden" name="belief" value=
    "<?php echo $nilai_belief; ?>" />
    <div class="form-group">
        <label for="text">Muncul Gejala :</label>
        <input type="text" name="gejala" class=
        "form-control" readonly value=
        "<?php echo $tampil_nama['nama_gejala']; ?>">
    </div>
    <button class="btn btn-info"
    type="submit">Ya</button>
    <a href=tidak.php class="btn btn-danger">Tidak</a>
</form>

```

Gambar 4.3 Menampilkan Pertanyaan

```

<?php
    $belief = $_POST['belief'];
    $kode = $_POST['kode'];
    $dugaan = $_POST['dugaan'];

    include "../proses-login/koneksi.php";
    $sql = "UPDATE temp_konsultasi
    SET temp_belief = $belief
    WHERE temp_kode_gejala = '$kode'";
    $hasil = mysqli_query($konek, $sql);
    if(!$hasil) {
        echo "Gagal Simpan Nilai Belief,
        silahkan diulangi... <br />" ;
        echo mysqli_error($konek);
    }
?>

```

Gambar 4.4 Memproses Data Konsultasi

Pada Gambar 4.4 merupakan *query* untuk menambahkan data nilai *belief* kedalam tabel `temp_konsultasi`, sedangkan untuk menambahkan dugaan penyakit ke dalam tabel `temp_konsultasi` ditunjukkan pada Gambar 4.5.

```

<?php
include "../../../proses-login/koneksi.php";
$sql = "UPDATE temp_konsultasi SET temp_dugaan =
'$dugaan' WHERE temp_kode_gejala = '$kode'";
$hasil = mysqli_query($konek, $sql);
if(!$hasil) {
    echo "Gagal Simpan Nilai Belief,
    silahkan diulangi... <br />" ;
    echo mysqli_error($konek);
}
echo '<META HTTP-EQUIV="Refresh"
Content="0; URL=../g013-2/g013-2.php ">';
?>

```

Gambar 4.5 Menambah Dugaan Penyakit

Setelah penyimpanan data nilai *belief* dan dugaan, dilanjutkan dengan pertanyaan gejala berikutnya dengan sintak `<META HTTP-EQUIV="Refresh" Content="0; URL=../g013-2/g013-2.php ">` yang juga ditunjukkan pada Gambar 4.5.

```

<?php
$kode = "G012";

include "../../../proses-login/koneksi.php";
$hasil = mysqli_query($konek, "DELETE FROM
temp_konsultasi WHERE temp_kode_gejala='$kode'");
if(!$hasil){
    die ("Gagal Query..".mysqli_error($konek));
}
echo '<META HTTP-EQUIV="Refresh" Content="0;
URL=../g014-3/g014-3.php">';
?>

```

Gambar 4.6 Proses Saat Memilih “Tidak”

Pada Gambar 4.6 saat pengguna memilih “Tidak”, maka data gejala dan penyakit yang sebelumnya telah disimpan akan dihapus kemudian dilanjutkan menuju pertanyaan berikutnya. Namun jika tidak ada pertanyaan lain yang mendukung maka user akan diarahkan ke halaman tidak terdiagnosa.

4.1.2 Menentukan Nilai Keyakinan Dari Hasil Diagnosa

Untuk menentukan diagnosa dengan metode *Dempster Shafer* terdapat pada program `diagnosa.php`. Untuk melakukan diagnosa maka dibutuhkan data dari proses inferensi yang tersimpan pada tabel `temp_konsultasi`, ditunjukkan pada Gambar 4.7.

```
//diagnosa.php
<?php
    include "../proses-login/koneksi.php";
    $sql_konsultasi = "SELECT * FROM temp_konsultasi";
    $hasil_konsultasi = mysqli_query($konek,
    $sql_konsultasi);
    if(!$hasil_konsultasi){
        die ("Gagal Query..".mysqli_error($konek));
    }
?>

<?php

    while($r = mysqli_fetch_array($hasil_konsultasi)){
        $data[] = array($r['temp_dugaan'],
        $r['temp_belief']);
    }
```

Gambar 4.7 Mengambil Informasi Hasil Inferensi

Kemudian data dari proses konsultasi disimpa kedalam *array* dengan variabel `$data`. Variabel tersebut merupakan variabel yang berisi *array* multi dimensi dengan 2 emelen. Variabel `$r['temp_dugaan']` merupakan elemen dimensi pertama sedangkan Variabel `$r['temp_belief']` adalah elemen dimensi kedua. Pada variabel `$data` tersimpan data *array* multi dimensi seperti yang ditunjukkan pada Gambar 4.8.

```
// page source variabel $data
Array
(
    [0] => Array
        (
            [0] => Pulorun, Ascites, Fowl Pox, Fowl
                Cholera,
                Infectious Bursall Disease,
                Inclusion Body Hepatitis
            [1] => 0.2
        )

    [1] => Array
        (
            [0] => Pulorun, Infectious Bursall Disease
            [1] => 0.8
        )

    [2] => Array
        (
            [0] => Mareks Disease,
                Infectious Bursall Disease, Inclusion
                Body Hepatitis, Avian Leukosis
            [1] => 0.6
        )
)
```

Gambar 4.8 Isi Dari Variabel Data

Proses selanjutnya adalah menentukan semesta pembicaraan yang terkait dengan gejala yang sudah dipilih. Namun sebelumnya data nama penyakit yang sama atau terduplikat akan dihilangkan terlebih dahulu dengan sintak `array_unique` yang ditunjukkan pada Gambar 4.9.

```
//diagnosa.php
$inferensi=array();
foreach($data as $d)
    $inferensi[]=$d[0];
$unik=array_unique(explode(', ',implode(', ', $inferensi)));
$jadi=implode(', ', $unik);
```

Gambar 4.9 Menghilangkan Duplikat Data

Dalam proses menentukan nilai densitas berdasarkan bukti (evidence), dari setiap gejala yang diberikan akan dihitung densitasnya satu per satu. Jika gejala yang pertama merupakan densitas ke-1 dan gejala kedua adalah densitas ke-2, maka hasil perhitungan dari kedua densitas tersebut akan diperoleh densitas ke-3. Kemudian jika terdapat gejala ketiga maka akan dijadikan sebagai densitas ke-4, yang akan dihitung dengan densitas ke-3 untuk memperoleh densitas baru yaitu densitas ke-5. Demikian seterusnya hingga semua gejala yang diberikan telah dihitung. Dengan kata lain jika terdapat N buah gejala yang diberikan, maka hasilnya adalah densitas ke N+1.

```

echo "<pre>";
$iterasi=array();
while(!empty($data)){
    $bukti=array();
    $penyakit[0]=array_shift($data);
    $penyakit[1]=array($jadi,1-$penyakit[0][1]);
}

```

Gambar 4.10 Menentukan Belief dan Plausability

Pada Gambar 4.10, data penyakit pada variabel `$data` akan diambil satu per satu dengan fungsi `array_shift()` dalam perulangan menggunakan `while()` selama data penyakit dalam variabel `$data` tersebut masih ada (`!empty($data)`). Data nilai *belief* yang muncul akan dimasukkan dalam variabel `$penyakit[0]`, sedangkan data *plausability* dimasukkan dalam variabel `$penyakit[1]`.

Pada iterasi pertama, jika belum ada hasil perhitungan densitas sebelumnya (`empty($iterasi)`) maka gejala berikutnya (yang kedua) dimasukkan ke dalam `$bukti[0]` sebagai densitas kedua. Sedangkan jika sudah ada maka semua hasil

perhitungan sebelumnya kecuali yang memiliki index "θ" dimasukkan ke variabel `$bukti=array()` sebagai item *array* baru yang merepresentasikan suatu densitas baru, yang ditunjukkan pada Gambar 4.11.

```
$bukti=array();
if(empty($iterasi))
    $bukti[0]=array_shift($data);
else
    foreach($iterasi as $k=>$r)
        if($k!="&theta;")
            $bukti[]=array($k,$r);
```

Gambar 4.11 Proses Iterasi Pertama

Selanjutnya menghitung nilai plausability dari densitas baru tersebut (θ); yaitu 1 dikurangi jumlah total nilai kemungkinan dari masing-masing elemen densitas baru. Caranya dengan menginisialisai variabel $\theta=1$; dan dilakukan pengurangan nilai kemungkinan dari masing-masing elemen densitas yang baru menggunakan sintak `foreach($bukti as $r) θ --$r[1];`. Kemudian hasilnya disimpan pada item *array* baru dari variabel array `$iterasi=array()`; yang ditunjukkan pada Gambar 4.12.

```
$theta=1;
foreach($bukti as $r) $theta--$r[1];
$bukti[]=array($jadi,$theta);
$iterasi=array();
```

Gambar 4.12 Menghitung Nilai Plausability Densitas Baru

Proses selanjutnya adalah menghitung jumlah iterasi, sesuai jumlah item *array* dari densitas yang baru dengan sintak `$m=count($bukti);`. Dalam perulangan `for()` dua dimensi dilakukan perhitungan untuk mendapatkan densitas yang baru. Karena hasil perkalian antara nilai kemungkinan dari plausibility, dari

densitas yang pertama dan kedua tidak diperhitungkan, maka dibatasi dengan sintak `if(!($y==$m-1 && $x==1))`. Tiap elemen dari `$penyakit` dan `$bukti` dikalikan nilai kemungkinannya, dan *array* indeksnya merupakan irisan dari penyakitnya (dalam hal ini memakai fungsi `array_intersect()` untuk menentukan irisan). Dan akhirnya dihitung masing-masing nilai kemungkinan dari tiap elemen, dari densitas yang baru tersebut ditunjukkan pada Gambar 4.13.

```

$m=count($bukti);
for($x=0;$x<$m;$x++){
    for($y=0;$y<2;$y++){

        if(!($x==$m-1 && $y==1)){
            $v=explode(',',$penyakit[$y][0]);
            $w=explode(',',$bukti[$x][0]);
            sort($v);sort($w);
            $vw=array_intersect($v,$w);

            if(empty($vw))
                $v="&theta;";
            else
                $v=implode(',',$vw);

            if(!isset($iterasi[$v]))
                $iterasi[$v]=$bukti[$x][1]*
                $penyakit[$y][1];
            else
                $iterasi[$v]+=$bukti[$x][1]*
                $penyakit[$y][1];
        }
    }
}

```

Gambar 4.13 Proses Menghitung Jumlah Iterasi

Kemudian hasil dari perhitungan iterasi akan diurutkan. Langkah pertama menghilangkan hasil sebelumnya yang mempunyai indeks `"θ"` dengan sintak `unset($iterasi["θ"]);`. Selanjutnya data tersebut diurutkan dengan tetap mempertahankan *associative keys*-nya dengan pengurutan secara terbalik dari besar ke kecil menggunakan fungsi `arsort`. Perintah

`print_r($iterasi);` digunakan untuk menampilkan hasil perhitungan terakhir yang sudah terurut dari yang terbesar hingga terkecil. Seperti yang ditunjukkan pada Gambar 4.14.

```
unset($iterasi["&theta;"]);
arsort($iterasi);
print_r($iterasi); // untuk melihat hasil iterasi
```

Gambar 4.14 Proses Mengurutkan Hasil Iterasi

Pada proses sebelumnya telah diurutkan hasil iterasi dari nilai terbesar sampai nilai terkecil, kemudian diambil nilai terbesar pada urutan paling awal yang akan disimpan pada tabel `temp_hasil`. Namun karena nilai tersebut masih berupa desimal maka sebelumnya harus dikalikan 100, yang ditunjukkan pada Gambar 4.15.

```
unset($iterasi["&theta;"]);
arsort($iterasi);
print_r($iterasi); // untuk melihat hasil iterasi

$key = $value = NULL;
foreach ($iterasi as $key => $value) {
    break;
}
$hitung = $value * 100;

$sql_simpan = "UPDATE temp_hasil SET nilai_tinggi =
$hitung WHERE id = 1";
$hasil_simpan = mysqli_query($konek, $sql_simpan);
if (!$hasil_simpan) {
    die ("Gagal Query.." . mysqli_error($konek));
} else {
    echo '<META HTTP-EQUIV="Refresh" Content="0;
URL=diagnosa-hasil.php">';
}
?>
```

Gambar 4.15 Proses Menyimpan Nilai Belief

4.1.3 Pengujian Sistem

Dalam pengujian program ini, dipilih tiga buah gejala yaitu :

1. Gejala Bulu Kusam & Mengkerut

Mendukung penyakit Pulatorun, Ascites, Fowl Pox, Fowl Cholera, Infectious Bursal Disease, dan Inclusion Body Hepatitis

2. Gejala Diare Keputihan

Mendukung penyakit Pulatorun, Infectious dan Bursal Disease,

3. Gejala Wajah Pucat

Mendukung penyakit Mareks Disease, Infectious Bursal Disease, Inclusion Body Hepatitis dan Avian Leukosis

```

Pulorun, Ascites, Fowl Pox, Fowl Cholera, Infectious Bursal
Disease, Inclusion Body Hepatitis, Mareks Disease, Avian
Leukosis
Array
(
  [Infectious Bursal Disease,Pulorun] => 0.8
  [Ascites, Fowl Cholera , Fowl Pox, Inclusion
  Body Hepatitis, Infectious Bursal Disease,
  Pulorun] => 0.04
)
Array
(
  [Infectious Bursal Disease] => 0.48
  [Infectious Bursal Disease,Pulorun] => 0.32
  [Inclusion Body Hepatitis,
  Infectious Bursal Disease] => 0.024
  [Ascites, Fowl Cholera , Fowl Pox,
  Inclusion Body Hepatitis, Infectious Bursal Disease,
  Pulorun] => 0.016
  [Avian Leukosis, Inclusion Body Hepatitis,
  Infectious Bursal Disease,Mareks Disease ] => 0.096
)
Array
(
  [Infectious Bursal Disease] => 0.48
  [Infectious Bursal Disease,Pulorun] => 0.32
  [Avian Leukosis, Inclusion Body Hepatitis,
  Infectious Bursal Disease,
  Mareks Disease ] => 0.096
  [Inclusion Body Hepatitis,
  Infectious Bursal Disease] => 0.024
  [Ascites, Fowl Cholera , Fowl Pox,
  Inclusion Body Hepatitis, Infectious Bursal Disease,
  Pulorun] => 0.016
)

```

Gambar 4.16 Proses Menentukan Penyakit

Hasil pengujian terkait pada Gambar 4.16 akan dibahas pada Sub Bab 4.1.3.

4.1.4 Menambah Kaidah

Pada bagian ini dibahas mengenai proses penambahan pengetahuan baru atau kaidah pada sistem pakar.

```
//kaidah.php
<!-- Modal Tambah-->

<h5 class="modal-title" id="exampleModalLabel">
Tambah Kaidah Penyakit
</h5>

<div class="modal-body">
<form action="../../../proses/add/tambah-kaidah.php"
        method="post"><div class="form-group">
<?php
    include "../../../proses-login/koneksi.php";
    $sql3 = "SELECT * FROM penyakit
    ORDER BY kode_penyakit";
    $nama2 = mysqli_query($konek, $sql3);
?>
<label for="sel1">Pilih Penyakit:</label>
<select class="form-control" name="sel1">
    <?php while ($row3 = mysqli_fetch_assoc($nama2)){ ?>
        <option><?php echo $row3['nama_penyakit']?></option>
</select>
</div>

<button class="btn btn-info "type="submit">
Lanjutkan</button>
<button class="btn btn-danger" type="button" data
dismiss="modal" aria-label="Close">Batal</button>
</form>
</div>
```

Gambar 4.17 Modal Tambah Kaidah

Untuk menambahkan kaidah dibuat modal pada program kaidah.php, yang ditunjukkan pada Gambar 4.17. Pada modal tersebut menampilkan data nama penyakit yang diambil dari tabel penyakit dalam bentuk dropdown dengan nama sel1. Pengguna diminta untuk memilih nama penyakit, kemudian data tersebut akan di proses pada program tambah-kaidah.php.

Pada program `tabah-kaidah.php` diproses data nama penyakit yang sebelumnya sudah dipilih, disini pengguna diminta untuk memilih nama gejala yang berkaitan dengan penyakit yang telah dipilih sebelumnya. Kemudian dilanjutkan menuju program `tambah-nilai.php` seperti yang ditunjukkan pada Gambar 4.18.

```
//tambah-kaidah.php

<form action="../../../add/tambah-nilai.php" method="post">

<label for="sel1">Pilih Gejala:</label>
<select class="form-control" name="sel2">
  <?php while ($row3 = mysqli_fetch_assoc($nama2)){ ?>
    <option><?php echo $row3['nama_gejala']?></option>
  <?php } ?>
</select>

<button class="btn btn-info" type="submit">Lanjut</button>
<a href="../../../page/kaidah.php" class="btn btn-danger"
role="button">Batal</a>

</form>
```

Gambar 4.18 Proses Tambah Kaidah

Pada program `tambah-nilai.php` pengguna diminta untuk menentukan nilai *belief* dari gejala yang telah dipilih sebelumnya seperti yang ditunjukkan pada Gambar 4.19. kemudian data nama penyakit, gejala, dan nilai *belief* yang sudah ditentukan akan di proses untuk disimpan pada program `simpan-kaidah.php`.

```
//tambah-nilai.php

<form action="../../save/simpan-kaidah.php" method="post">

<label for="text">Nilai:</label>
<input type="text" name="nilai"
class="form-control" value="">

<button class="btn btn-info" type="submit">Simpan</button>
<a href="../../page/kaidah.php" class="btn btn-danger"
role="button">Batal</a>

</form>
```

Gambar 4.19 Proses Tambah Nilai

Dalam program `simpan-kaidah.php`, data nama penyakit, gejala, dan nilai *belief* yang telah ditentukan oleh user akan disimpan pada tabel `basis_pengetahuan`. Jika penyimpanan data gagal akan diarahkan menuju program `peringatan.php` yang berisi informasi kesalahan pada saat proses penyimpanan data. Namun jika tidak ada kesalahan maka user akan diarahkan kembali menuju program `kaidah.php` yang ditunjukkan pada Gambar 4.20.

```

//simpan-kaidah.php
<?php
    include "../.../proses-login/session.php";
    $kode_penyakit = $_POST['kodep'];
    $kode_gejala = $_POST['kodeg'];
    $nilai = $_POST['nilai'];

    include "../.../proses-login/koneksi.php";
    $sql = "INSERT INTO basis_pengetahuan
            (id_pengetahuan, kode_penyakit, kode_gejala,
             nilai_belief)
    VALUES
    (NULL, '$kode_penyakit', '$kode_gejala', '$nilai')";
    $hasil = mysqli_query($konek, $sql);
    if(!$hasil) {
        include "../.../view/peringatan.php";
        exit;
    }
    echo '<META HTTP-EQUIV="Refresh" Content="0;
    URL=../.../page/kaidah.php">';
?>

```

Gambar 4.20 Proses Simpan Kaidah

4.2 Pembahasan

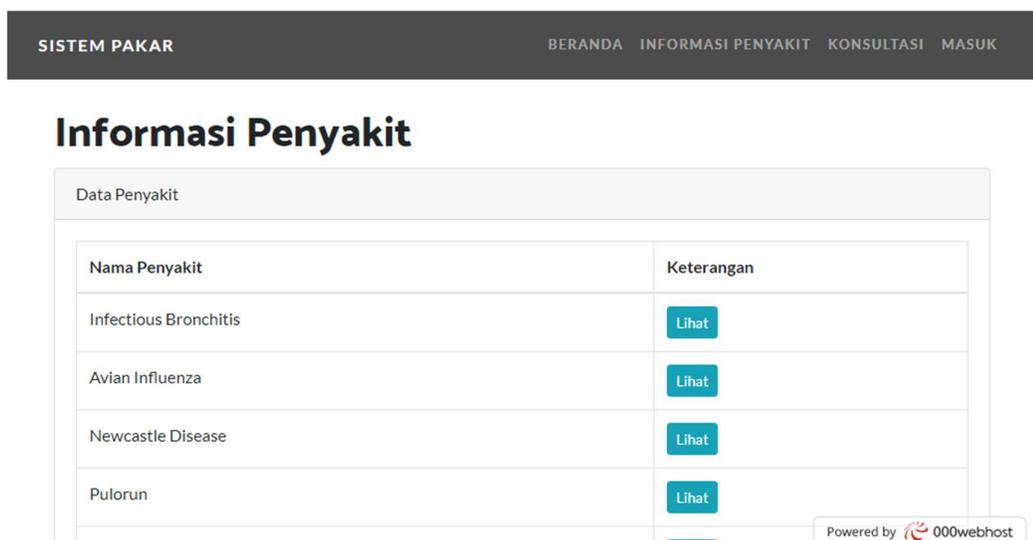
4.2.1 Sisi Peternak (Pengguna Umum)

Tampilan pada sisi peternak atau pengguna umum memiliki empat buah menu yaitu beranda, informasi penyakit, konsultasi, dan menu untuk masuk sebagai pakar. Pada tampilan utama atau beranda, menampilkan nama aplikasi yaitu sistem pakar untuk mendiagnosa penyakit ayam broiler seperti yang ditunjukkan pada Gambar 4.21.

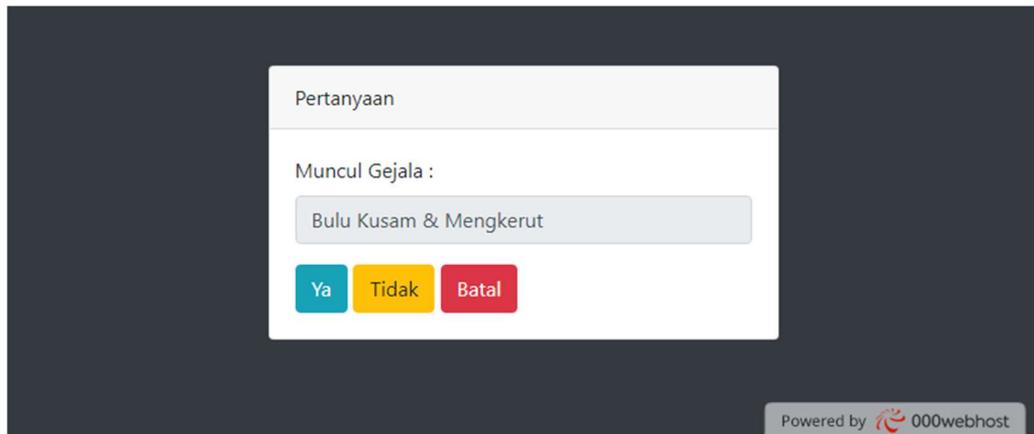


Gambar 4.21 Halama Beranda

Pada menu informasi penyakit terdapat daftar nama-nama penyakit serta keterangan yang berisi informasi mengenai penyakit tersebut, diantaranya penyebab penyakit serta solusi penanggulangan terhadap penyakit tersebut yang ditunjukkan pada Gambar 4.22.



Gambar 4.22 Halaman Informasi Penyakit



The image shows a web-based consultation form. At the top, there is a header labeled 'Pertanyaan'. Below this, the text 'Muncul Gejala :' is followed by a text input field containing the text 'Bulu Kusam & Mengkerut'. Underneath the input field are three buttons: a blue button labeled 'Ya', a yellow button labeled 'Tidak', and a red button labeled 'Batal'. In the bottom right corner of the form, there is a small logo and the text 'Powered by 000webhost'.

Gambar 4.23 Halaman Konsultasi

Dalam menu konsultasi terdapat beberapa pertanyaan mengenai gejala penyakit tertentu. Pengguna diminta untuk menjawab dengan pilihan “Ya” atau “Tidak” berdasarkan gejala yang muncul pada ayam seperti pada Gambar 4.23. Setelah proses konsultasi selesai maka pengguna akan di arahkan menuju halaman diagnosa yang ditunjukkan pada Gambar 4.24. Pada halaman diagnosa ditampilkan gejala yang dipilih, diagnosa penyakit, serta nilai persentase kemungkinan penyakit yang diderita. Kemudian terdapat tombol “Lihat Solusi” untuk menampilkan solusi dari penyakit tersebut seperti yang ditunjukkan pada Gambar 4.25.

Gejala Terpilih
SISTEM PAKAR

BERANDA INFORMASI PENYAKIT KONSULTASI MASUK

Nama Gejala	Dugaan Penyakit	Nilai
Bulu Kusam & Mengkerut	Puloron, Ascites, Fowl Pox, Fowl Cholera, Infectious Bursal Disease, Inclusion Body Hepatitis,	0.2
Diare Keputihan	Puloron, Infectious Bursal Disease,	0.8
Wajah Pucat	Mareks Disease, Infectious Bursal Disease, Inclusion Body Hepatitis, Avian Leukosis,	0.6

Hasil Diagnosa

Diagnosa Penyakit

Hasil diagnosa penyakit yang diderita adalah **Infectious Bursal Disease** dengan kemungkinan sebesar 48 %

Lihat Solusi

Powered by 000webhost

Gambar 4.24 Halaman Hasil Diagnosa

Solusi Penyakit Infectious Bursal Disease

Penyebab :

Infectious Bursal Disease atau gumboro disebabkan oleh virus yang menciri terhadap keluarga Birnaviridae. Didalam tubuh ayam, virus ini dapat hidup hingga lebih dari 3 bulan, kemudian akan

Solusi Penanggulangan :

Pengobatan terhadap penyakit ini tidak berhasil guna tetapi penyakit ikutannya harus diobati sesuai dengan jenis penyakitnya. Usaha pencegahan yang paling efektif adalah tidak vaksinasi, menjaga

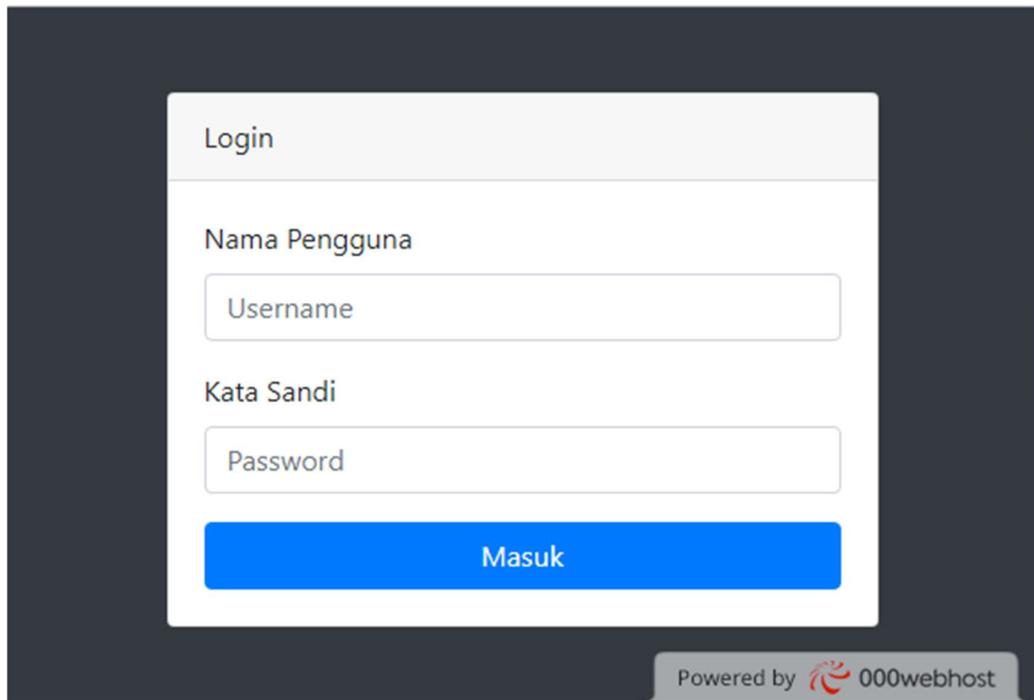
Kembali

Powered by 000webhost

Gambar 4.25 Solusi Penyakit

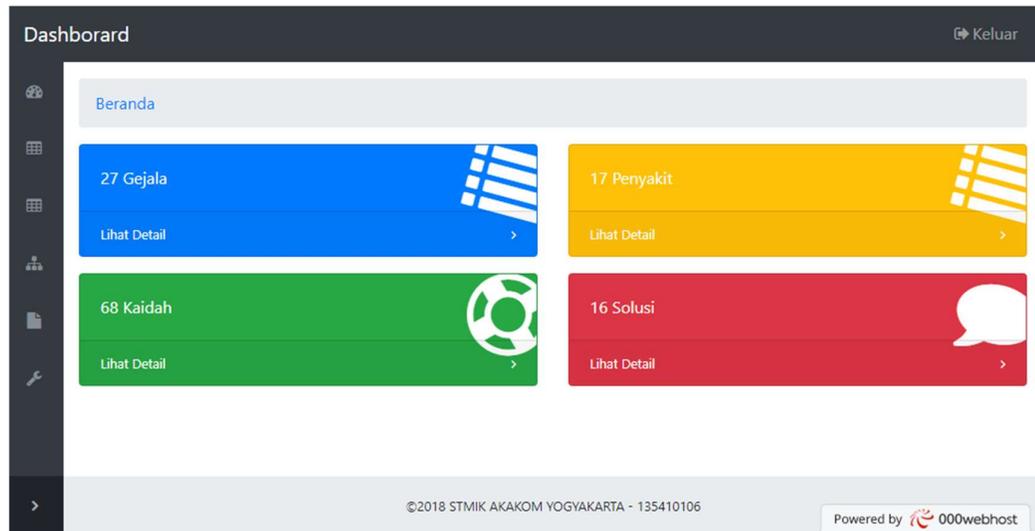
4.2.2 Sisi Pakar

Pada halaman pakar berisi beberapa menu untuk menentukan kaidah, solusi dan beberapa informasi lainnya yang diperlukan dalam proses konsultasi. Sebelum mengakses halaman tersebut, pakar diminta untuk login terlebih dahulu, seperti yang ditunjukkan pada Gambar 4.26.

The image shows a login form on a dark background. The form is white and has a title 'Login' at the top. Below the title, there are two input fields: 'Nama Pengguna' (Username) and 'Kata Sandi' (Password). The 'Username' field contains the text 'Username' and the 'Password' field contains the text 'Password'. Below these fields is a blue button with the text 'Masuk'. At the bottom right of the form, there is a small logo and the text 'Powered by 000webhost'.

Gambar 4.26 Halaman Login Pakar

Setelah login maka pakar akan diarahkan menuju halaman dashboard yang berisi informasi jumlah gejala, penyakit, serta kaidah dan solusi. Dalam menu ini pakar juga dapat mengubah atau menambahkan informasi terkait proses diagnosa penyakit, kaidah, serta solusi penanggulangan untuk penyakit yang ditunjukkan pada Gambar 4.27.



Gambar 4.27 Halaman Dashboard Pakar

4.2.3 Hasil Diagnosa

Poses diagnosa yang ditampilkan pada Gambar 4.24 dan Gambar 4.25 terlihat bahwa sistem menampilkan hasil akhir konsultasi. Pada tahap ini ditampilkan gejala yang dipilih, diagnosa penyakit, serta nilai persentase kemungkinan penyakit.

Dalam konsultasi yang ditunjukkan diperoleh hasil yang sesuai dengan kaidah dan proses inferensi sistem yang digunakan. Adapun kaidah yang menuju jenis penyakit Infectious Bursal Disease dengan gejala bulu kusam & mengkerut, diare keputihan, wajah pucat adalah sebagai berikut:

R13 : IF bulu kusam & mengkerut AND diare keputihan AND wajah pucat THEN Infectious Bursall Disease

Berdasarkan kasus tersebut menunjukkan bahwa hasil diagnosa yang ditampilkan sudah sesuai dengan kaidah yang ada. Untuk perhitungan Dempster Shafer sesuai dengan gejala yang dimaksud, selengkapnya disajikan pada lampiran.