

## LEMBAR PERSETUJUAN

### RANCANG BANGUN APLIKASI UNTUK MENGLASIFIKASI HEWAN MENGGUNAKAN METODE DECISION TREE DENGAN ALGORITMA ITERATIVE DICHOTOMIZER 3 BERBASIS VISUAL BASIC

#### SKRIPSI

Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh  
Gelar Sarjana Teknik Strata Satu (S-1)

Disusun Oleh:

Irfan Ari Krisanto  
NIM : 07.12.502


Mengetahui

Ketua Program Studi Teknik Elektro S-1

  
Ir. Yusuf Ismail Nakhoda, MT  
NIP. Y. 1018800189

Diperiksa dan Disetujui

Dosen Pembimbing I

  
Ir. Eko Nurcahyo, MT  
NIP.Y.1028700172

Dosen Pembimbing II

  
Ahmad Faisol, ST  
NIP.Y.1031000431

PROGRAM STUDI TEKNIK ELEKTRO S-1  
KONSENTRASI TEKNIK KOMPUTER & INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2012

## SURAT PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini :

Nama : Irfan Ary Krisanto  
NIM : 0712502  
Program Studi : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer & Informatika

Dengan ini menyatakan bahwa Skripsi yang saya buat adalah hasil karya sendiri, tidak merupakan plagiasi dari karya orang lain. Dalam Skripsi ini tidak memuat karya orang lain, kecuali dicantumkan sumbernya sesuai dengan ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat, dan apabila di kemudian hari ada pelanggaran atas surat pernyataan ini, saya bersedia menerima sanksinya.

Malang, 22 Oktober 2012



Irfan Ary Krisanto  
0712502

## ABSTRAK

### RANCANG BANGUN APLIKASI UNTUK MENGLASIFIKASI HEWAN MENGUNAKAN METODE *DECISION TREE* DENGAN ALGORITMA *ITERATIVE DICHOTOMIZER 3* BEBASIS *VISUAL BASIC*

Irfan Ary Krisanto  
0712502

Jurusan Teknik Elektro S-1, Konsentrasi Teknik Komputer & Informatika  
Fakultas Teknologi Industri, Institut Teknologi Nasional Malang  
Jl. Raya Karanglo Km 2 Malang  
E-mail : [Irfan\\_racer86@yahoo.com](mailto:Irfan_racer86@yahoo.com)

Dosen Pembimbing : I. Ir. Eko Nurcahyo, MT.  
II. Ahmad Faisol, ST.

Tuhan menciptakan makhluk hidup di muka bumi ini dalam bentuk yang beraneka ragam, masing-masing mempunyai ciri tersendiri yang membedakannya antara satu dengan yang lain. Makhluk hidup tersebut sangat beragam jenisnya, mulai dari tumbuh-tumbuhan, binatang, maupun bakteri. Setiap jenis hewan juga mempunyai ciri dan karakteristik yang berbeda-beda. Mengklasifikasi adalah suatu sistem untuk menyederhanakan suatu objek (makhluk hidup) dalam jumlah yang besar dan beragam menjadi kelompok-kelompok kecil. Di dalam mengklasifikasi hewan, cara yang dapat dilakukan adalah melakukan teknik data mining, dimana teknik data mining adalah suatu cara dalam menggali informasi yang terpendam dari database yang besar sehingga menjadi informasi yang berharga. Pada data mining terdapat tiga tahapan, yaitu pengumpulan data (*data collection*), transformasi data (*data transformation*), dan analisis data (*data analysis*). Pada penelitian ini menggunakan tipe pohon klasifikasi, karena mengklasifikasi hewan diklasifikasi menggunakan *rule* untuk masing – masing kelas dan algoritma yang digunakan adalah *iterative dichotomizer 3* menggunakan bahasa pemrograman *visual basic*. Dengan keberagaman hewan dimuka bumi ini yang dilihat dari ciri-ciri fisiknya maka, secara sederhana aplikasi mengklasifikasi hewan ini sedikit membantu para peneliti-peneliti biologi dalam mengelompokkan hewan berdasarkan typenya.

**Kata kunci:** Klasifikasi Hewan, Rancang Bangun Aplikasi Untuk Mengklasifikasi Hewan Menggunakan Metode *Decision Tree* Dengan Algoritma *Iterative Dichotomizer 3* Berbasis *Visual Basic*.

## KATA PENGANTAR

Dengan mengucap syukur kehadiran Allah SWT atas Rahmat dan Hidayah-Nya, yang telah memberikan kekuatan, kesabaran, bimbingan dan perlindungan sehingga penulis dapat menyelesaikan laporan skripsi dengan judul “Rancang Bangun Aplikasi Untuk Mengklasifikasi Hewan Menggunakan Metode *Decision Tree* Dengan Algoritma *Iterative Dichotomizer 3* Berbasis *Visual Basic*.” dengan lancar. Skripsi ini merupakan persyaratan kelulusan studi di jurusan Teknik Elektro S-1 konsentrasi Teknik Komputer & Informatika ITN Malang dan untuk mencapai gelar sarjana teknik.

Keberhasilan penyelesaian laporan skripsi ini tidak lepas dari dukungan dan bantuan berbagai pihak. Untuk itu penyusun menyampaikan terima kasih kepada:

1. Bapak Ir. Soeparno Djiwo, MT selaku Rektor ITN Malang.
2. Bapak Ir. Sidik Noertjahjono, MT selaku Dekan Fakultas Teknologi Industri.
3. Bapak Ir. Yusuf Ismail Nakhoda, MT selaku Ketua Program Studi Teknik Elektro S-1
4. Bapak Dr. Ir. Eng. Aryuanto Soetedjo, MT selaku Sekretaris Program Studi Teknik Elektro S-1.
5. Bapak Ir. Eko Nurcahyo, MI selaku Dosen Pembimbing I.
6. Bapak Ahmad Faisol, ST selaku Dosen Pembimbing II.
7. Orang tuaku Bapak, Ibu dan saudaraku tercinta, terima kasih atas doa, kasih sayang, perhatian, serta dukungan baik materi maupun non-materi yang tak ternilai yang telah diberikan
8. Semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang membangun sangat diharapkan.

Akhir kata, penulis mohon maaf kepada semua pihak bilamana selama penyusunan skripsi ini penyusun membuat kesalahan secara tidak sengaja dan semoga skripsi ini dapat bermanfaat bagi kita semua.

Malang, Oktober 2012

Penulis

Gambar 4.27. Memasukkan Parameter ke 2.....	65
Gambar 4.28. Memasukkan Range ke 2.....	66
Gambar 4.29. Hasil Memasukkan Nama Hewan dan Parameter ke 2 Beserta Rangnya.....	66
Gambar 4.30. Tombol Menyimpan Data.....	67
Gambar 4.31. Tombol Menambah Tabel.....	67
Gambar 4.32. Menghapus Tabel.....	68
Gambar 4.33. Hasil Keseluruhan Input Data Hewan.....	68
Gambar 4.34. Pembentukan Tree.....	69
Gambar 4.35. Pembentukan Tree Node 0.....	70
Gambar 4.36. Pembentukan Tree Node 1.....	70
Gambar 4.37. Lihat Kode.....	71
Gambar 4.38. Lihat Rule.....	71
Gambar 4.39. Display Data Uji.....	72
Gambar 4.40. Display Setelah Proses.....	73
Gambar 4.41. Display Salah Memasukkan Data.....	73
Gambar 4.42. Display Kurang Memasukkan Data.....	74
Gambar 4.43. Display seluruh Data Latih Diujikan.....	74
Gambar 4.44. Display Menguji Nilai Akurasi Data Latih.....	75
Gambar 4.45. Display Menguji Nilai Akurasi Data Latih Dengan 2 Nilai Salah.....	75
Gambar 4.46. Display Menguji Nilai Akurasi Data Latih Dengan 3 Nilai Salah.....	76
Gambar 4.47. Display Menguji Nilai Akurasi Data Latih Dengan 4 Nilai Salah.....	76
Gambar 4.48. Display Menguji Nilai Akurasi Data Latih Dengan 5 Nilai Salah.....	77
Gambar 4.49. Display Data Keseluruhan Diuji.....	77

## DAFTAR TABEL

Tabel 2.1 Dataset Auto Driving.....	14
Tabel 3.1 Data Latih.....	26
Tabel 3.2 Probabilitas Type.....	28
Tabel 3.3 Distribusi Kelas Level Organisasi 0.....	28
Tabel 3.4 Distribusi Kelas Simetri Tubuh Fase Dewasa 0.....	29
Tabel 3.5 Distribusi Kelas Pencernaan Makanan 0.....	30
Tabel 3.6 Distribusi Kelas Peredaran Darah 0.....	30
Tabel 3.7 Distribusi Kelas Sistem Saraf 0.....	31
Tabel 3.8 Distribusi Kelas Sistem Reproduksi 0.....	32
Tabel 3.9 Distribusi Kelas Rangka Tubuh 0.....	32
Tabel 3.10 Distribusi Kelas Jumlah Alat Gerak Tubuh 0.....	33
Tabel 3.11 Hasil Perhitungan Nilai Batas Jumlah Alat Gerak 0.....	35
Tabel 3.12 Distribusi Kelas Kelenjar Susu 0.....	36
Tabel 3.13 Distribusi Kelas Bulu 0.....	36
Tabel 3.14 Distribusi Kelas Sisik 0.....	37
Tabel 3.15 Distribusi Kelas Cangkang Literal 0.....	38
Tabel 3.16 Distribusi Kelas Carapak 0.....	38
Tabel 3.17 Nilai Information Gain Dari Masing-masing Kelas Atribut Iterasi 0.....	39





## BAB 1

### PENDAHULUAN

#### 1.1 Latar Belakang

Tuhan menciptakan makhluk hidup di muka bumi ini dalam bentuk yang beraneka ragam. Masing-masing mempunyai ciri tersendiri yang membedakannya antara satu dengan yang lain. Makhluk hidup tersebut sangat beragam jenisnya, mulai dari tumbuh-tumbuhan, hewan, maupun bakteri. Setiap jenis hewan juga mempunyai ciri dan karakteristik yang berbeda-beda. Untuk menggolongkan makhluk hidup tersebut, para ahli menggunakan sistem klasifikasi. Klasifikasi adalah sistem untuk mengenali makhluk hidup berdasarkan ciri – cirinya, kemudian ciri – ciri tersebut dikelompokkan berdasarkan jenisnya. Mengklasifikasi adalah suatu sistem untuk menyederhanakan suatu objek (makhluk hidup) dalam jumlah yang besar dan beragam menjadi kelompok-kelompok kecil.

Data mining adalah ekstraksi informasi atau pola yang penting atau menarik dari data yang ada di database yang besar menurut (Zalilia 2007). Teknik data mining merupakan penerapan dari sistem klasifikasi terkomputerisasi yang dapat menggali sebuah informasi, beberapa teknik yang sering digunakan adalah *association rule*, *clustering*, klasifikasi (*classification*), *neural network*, dan *genetic algorithm* (Kusnawi, 2007).

Pada proses mengklasifikasi hewan teknik data mining yang digunakan untuk menggali informasi pada data menggunakan teknik klasifikasi. Ada beberapa metode yang bisa digunakan untuk proses klasifikasi, seperti *decision tree*, *backpropagation*, dan *bayesian*. Metode yang digunakan dalam penelitian ini yaitu metode *decision tree*. Pada penelitian ini metode *decision tree* menggunakan algoritma *iterative dichotomiser 3* dan menggunakan bahasa pemrograman *microsoft visual basic*.

Berdasarkan latar belakang yang dikemukakan, maka judul yang diambil dalam skripsi ini adalah "RANCANG BANGUN APLIKASI UNTUK MENGLASIFIKASI HEWAN MENGGUNAKAN METODE *DECISION TREE* DENGAN ALGORITMA *ITERATIVE DICHOTOMIZER 3* BERBASIS *VISUAL BASIC*".



## 1.2 Rumusan Masalah

Berdasarkan pada latar belakang permasalahan yang ada, maka rumusan masalah dari skripsi ini adalah :

1. Bagaimana cara membangun sebuah aplikasi berbasis *visual basic* untuk mengklasifikasi data hewan menggunakan metode *decision tree* dengan algoritma *ID3*.
2. Berapa tingkat akurasi pada sistem yang dibuat dalam mengklasifikasi data hewan.

## 1.3 Tujuan

Tujuan pembuatan skripsi ini adalah :

1. Membuat sistem aplikasi berbasis *visual basic* untuk mengklasifikasi data hewan dengan menerapkan metode *decision tree* dan algoritma *ID 3*.
2. Pada sistem yang dibuat untuk mengklasifikasi data hewan dengan metode *decision tree* dan algoritma *ID 3* dan mengetahui tingkat keakurasiannya.

## 1.4 Batasan Masalah

Batasan masalah bertujuan agar penelitian tetap mengacu pada topik yang diteliti. Batasan masalahnya adalah :

1. Dalam mengklasifikasi data hewan tersebut akan diklasifikasi berdasarkan beberapa typenya. Diantaranya adalah type *aves*, *amphibi*, *mamalia*, *reptilia*, *osteichthyes*, *asteroidea*, *archnida*, *insecta*, *crustacea*, *cephalopoda*, *bivalvia*, *gastropoda*, dan type *scyphozoa*.
2. Data karakteristik hewan yang digunakan berbentuk catatan.
3. Tidak terdapat nilai hilang pada data hewan.
4. Aplikasi ini digunakan oleh peneliti-peneliti biologi.

## 1.5 Metodologi Penelitian

Adapun metode penelitian yang digunakan adalah sebagai berikut:

### 1. Studi literatur

Pengumpulan data yang dilakukan dengan mencari bahan-bahan kepustakaan dan referensi dari berbagai sumber sebagai landasan teori yang ada hubungannya dengan permasalahan yang dijadikan objek penelitian.

---

## 2. Analisa Kebutuhan Sistem

Data dan informasi yang telah diperoleh akan dianalisa agar didapatkan kerangka global yang bertujuan untuk mendefinisikan kebutuhan sistem di mana nantinya akan digunakan sebagai acuan perancangan sistem.

## 3. Perancangan dan Implementasi

Berdasarkan data dan informasi yang telah diperoleh serta analisa kebutuhan untuk membangun sistem ini, akan dibuat rancangan kerangka global yang menggambarkan mekanisme dari sistem yang akan dibuat dan diimplementasikan kedalam sistem.

## 4. Eksperimen dan Evaluasi

Pada tahap ini, sistem yang telah selesai dibuat akan diuji coba, yaitu pengujian berdasarkan fungsionalitas program, dan akan dilakukan koreksi dan penyempurnaan program jika diperlukan.

### 1.6 Sistematika Penulisan

Untuk mempermudah dan memahami pembahasan penulisan skripsi ini, maka sistematika penulisan disusun sebagai berikut :

#### Bab I : Pendahuluan

Berisi Latar Belakang, Rumusan Masalah, Tujuan Penelitian, Pembatasan Permasalahan, Metode Penelitian dan Sistematika Penulisan.

#### Bab II : Tinjauan Pustaka

Berisi tentang landasan teori mengenai permasalahan yang berhubungan dengan penelitian yang dilakukan.

#### Bab III : Perancangan Dan Analisa Aplikasi

Dalam bab ini berisi mengenai analisa kebutuhan Aplikasi baik software maupun hardware yang diperlukan untuk membuat kerangka global yang menggambarkan mekanisme dari sistem yang akan dibuat.

#### Bab IV : Pembuatan Dan Pengujian Sistem

Berisi tentang implementasi dari perancangan aplikasi yang telah dibuat serta pengujian terhadap aplikasi tersebut.

---

## Bab V : Penutup

Merupakan bab terakhir yang memuat intisari dari hasil pembahasan yang berisikan kesimpulan dan saran yang dapat digunakan sebagai pertimbangan untuk pengembangan penulisan selanjutnya.

---



## BAB II

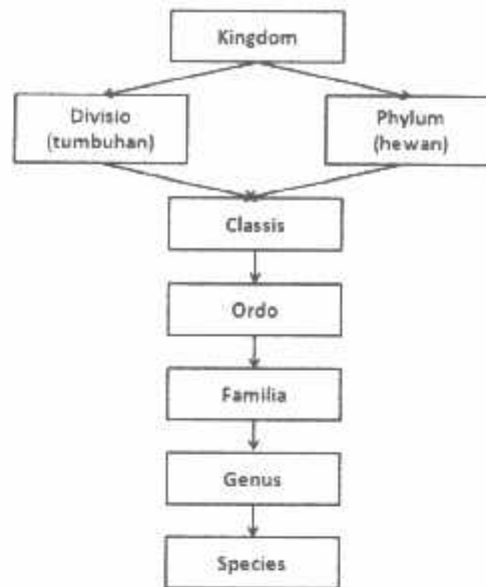
### LANDASAN TEORI

#### 2.1 Klasifikasi Hewan

Klasifikasi adalah suatu cara pengelompokan yang didasarkan pada ciri-ciri tertentu. Semua ahli biologi menggunakan suatu sistem klasifikasi untuk mengelompokkan tumbuhan ataupun hewan yang memiliki persamaan struktur. Kemudian setiap kelompok tumbuhan ataupun hewan tersebut dipasang-pasangkan dengan kelompok tumbuhan atau hewan lainnya yang memiliki persamaan dalam kategori lain. Hal itu pertama kali diusulkan oleh John Ray yang berasal dari Inggris. Namun ide itu disempurnakan oleh Carl Von Linne (1707-1778), seorang ahli botani berkebangsaan Swedia yang kini dikenal dengan nama Carolus Linnaeus (Chita 2011). Pada tahun 1753 Carolus Linnaeus mulai merintis klasifikasi secara modern. Di dalam klasifikasi, makhluk hidup dikelompokkan dalam kelompok besar hingga kecil. Kategori yang digunakan Linnaeus pada waktu itu adalah kingdom (kerajaan), filum (keluarga besar), kelas, ordo (bangsa), famili (suku), genus (marga), dan spesies (jenis).

Seiring dengan meningkatnya peradaban manusia terutama pengetahuan tentang manfaat makhluk hidup diberbagai sendi kehidupan, maka keperluan akan nama makhluk hidup semakin besar. Oleh karena itu mulai diperlukan suatu penggolongan makhluk hidup berdasarkan pemikiran yang rasional, misalnya penggolongan berdasarkan persamaan ciri, cara hidup, tempat hidup, dan sebagainya.

Klasifikasi makhluk hidup menggunakan dasar atau kriteria tertentu, yaitu persamaan ciri atau sifat morfologi, fisiologi, dan anatomi yang terdapat pada makhluk hidup. Sistem klasifikasi makhluk hidup terus berkembang sejalan dengan perkembangan ilmu pengetahuan, khususnya taksonomi (Chita, 2011). Tingkatan Takson dapat dilihat pada gambar 2.1.



Gambar 2.1 Tingkatan Makhluk Hidup (Takson)  
(Chita Najmi Nabila, 2011)

Tujuan dari mengklasifikasi makhluk hidup adalah untuk mendeskripsikan ciri-ciri makhluk hidup agar mudah dikenali, mengelompokkan makhluk hidup berdasarkan persamaan ciri, melihat hubungan kekerabatan antar anggota kelompok makhluk hidup dalam klasifikasi tersebut. Makin banyak persamaan satu golongan dengan golongan lain artinya kedua golongan tersebut memiliki hubungan kekerabatan makin dekat. Manfaat dari klasifikasi adalah menyederhanakan obyek studi, dimana dalam hal ini ialah apabila kita mempelajari sesuatu tidak perlu semua makhluk hidup yang ada di muka bumi diteliti satu persatu, tetapi cukup dengan sampel atau perwakilan dari objek tersebut yang dianggap sudah mewakili semua. Misalnya untuk mempelajari serangga atau lebah dengan karakteristik yang mewakili serangga tersebut.

## 2.2 Data Mining

### 2.2.1. Pengertian Data Mining

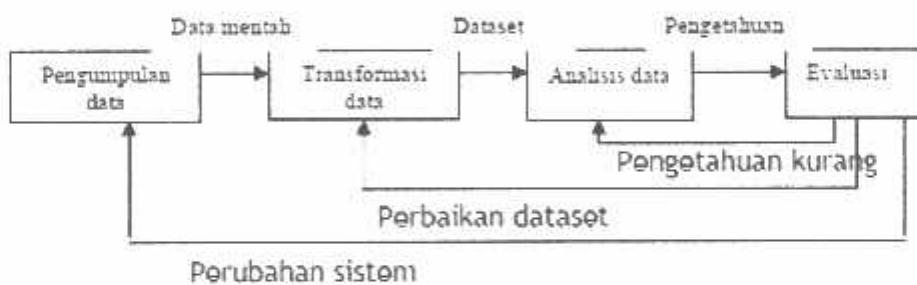
Data Mining adalah proses mencari informasi atau mengambil suatu keputusan dari database yang besar. Menurut Zalilia (2007), sebelum mengerti makna data mining, diperlukan untuk memahami makna kata mining. Dalam bahasa Inggris, to mining dapat diartikan sebagai kegiatan menambang. To mining dapat juga berarti mengambil ekstrak atau sari. Sehingga secara sederhana dapat dikatakan bahwa data mining adalah ekstraksi informasi atau pola yang penting atau menarik dari data yang ada di database yang besar.



Definisi sederhana dari data mining adalah ekstraksi informasi atau pola yang penting atau menarik dari data yang ada di database yang besar. Dalam jurnal ilmiah, data mining juga dikenal dengan nama *Knowledge Discovery in Databases* (KDD). (Yudho, 2003).

### 2.2.2. Proses data mining

Pada proses data mining terdapat tiga tahapan, yaitu pengumpulan data (*data collection*), transformasi data (*data transformation*), dan analisis data (*data analysis*). Proses tersebut diawali dengan *preprocessing* yang terdiri atas pengumpulan data untuk menghasilkan data mentah (*raw data*) yang dibutuhkan oleh *data mining*, yang kemudian dilanjutkan dengan transformasi data untuk mengubah data mentah menjadi format yang dapat diproses oleh *data mining*. Hasil transformasi data akan digunakan oleh analisis data untuk membangkitkan pengetahuan dengan menggunakan teknik seperti analisis statistik, *machine learning*, dan visualisasi informasi. Alur informasi pada data mining dapat dilihat pada gambar 2.2 dibawah ini :



Gambar 2.2 Aliran Informasi Dalam Data Mining

(Ayuh, 2007)

Pada gambar tersebut ditunjukkan diagram yang menggambarkan aliran informasi dalam proses data mining. Proses data mining tersebut ditunjukkan sebagai proses yang iteratif. Hasil evaluasi pengetahuan yang dihasilkan data mining dapat menimbulkan kebutuhan pengetahuan yang lebih lengkap, perbaikan kumpulan data (*data set*) atau perubahan oleh sistem (ayuh, 2007).

### 2.2.3. Teknik – Teknik Data Mining

Data mining adalah serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang belum diketahui secara manual. Arti dari kata *mining* itu sendiri berarti usaha mendapatkan sedikit data berharga dari sejumlah data

yang besar. Karena itu data mining sebenarnya memiliki akar yang panjang dari bidang ilmu seperti kecerdasan buatan (*artificial intellegent*), *machine learning*, statistik dan basis data. Beberapa teknik yang sering disebut – sebut dalam literatur data mining antara lain yaitu *association rule mining*, *clustering*, *klasifikasi (classification)*, *neural network*, *genetic algoritm*, dan lain – lain (Lia, 2007).

### 1. Association Rule Mining

*Association rule mining* adalah teknik data mining untuk menemukan aturan assosiatif antara suatu kombinasi item. Contoh dari aturan assosiatif dari analisa pembelian di suatu pasar swalayan adalah dapat diketahuinya berapa besar kemungkinan seorang pelanggan membeli roti bersamaan dengan susu. Dengan pengetahuan tersebut pemilik pasar swalayan dapat mengatur penempatan barangnya atau merancang kampanye pemasaran dengan memakai kupon diskon untuk kombinasi barang tertentu. Karena analisis asosiasi menjadi terkenal karena aplikasinya untuk menganalisa isi keranjang belanja di pasar swalayan. (Amirudin, 2007).

### 2. Klasifikasi (*classification*)

Klasifikasi adalah salah satu teknik dalam data mining yang bertujuan mengelompokkan data ke dalam kelas-kelas yang sudah ditentukan sebelumnya. Klasifikasi dapat juga didefinisikan sebagai proses untuk menemukan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk memperkirakan kelas dari suatu objek yang labelnya tidak diketahui (Pramudiono, 2003).

Teknik klasifikasi bekerja dengan mengelompokkan data berdasarkan data training dan nilai atribut klasifikasi. Aturan pengelompokan tersebut akan digunakan untuk klasifikasi data baru ke dalam kelompok yang ada. Klasifikasi dapat direpresentasikan dalam bentuk pohon keputusan (*decision tree*). Setiap node dalam pohon keputusan menyatakan suatu tes terhadap atribut dataset, sedangkan setiap cabang menyatakan hasil dari tes tersebut. Pohon keputusan yang terbentuk dapat diterjemahkan menjadi sekumpulan aturan dalam bentuk IF condition THEN outcome. (Ayub, 2007).

---

### 3. *Clustering*

Berbeda dengan association rule mining dan classification dimana kelas data telah ditentukan sebelumnya, clustering melakukan penge-lompokan data tanpa berdasarkan kelas data tertentu. Bahkan *clustering* dapat dipakai untuk memberikan label pada kelas data yang belum diketahui itu. Karena itu clustering sering digolongkan sebagai metode *unsupervised learning*.

Prinsip dari *clustering* adalah memaksimalkan kesamaan antar anggota satu kelas dan meminimumkan kesamaan antar kelas/cluster. *Clustering* dapat dilakukan pada data yang memiliki beberapa atribut yang dipetakan sebagai ruang multidimensi. (Pramudiono, 2003).

## 2.3. Pohon Keputusan (Decision Tree)

### 2.3.1. Pengertian Decision Tree

Pohon keputusan (*decision tree*) merupakan metode klasifikasi dan prediksi yang sangat kuat dan sangat populer. Metode pohon keputusan mengubah fakta yang sangat besar menjadi pohon keputusan yang mempresenstasikan aturan.

Sebuah pohon keputusan adalah sebuah struktur yang dapat digunakan untuk membagi kumpulan data yang besar menjadi himpunan-himpunan *record* yang lebih kecil dengan menerapkan serangkaian aturan-aturan keputusan (Berry dan Linnof, 2004). Pembagian data tersebut berupa struktur *flowchart* yang menyerupai *tree* (pohon), dimana setiap simpul internal menandakan suatu tes pada atribut, setiap cabang merepresentasikan hasil tes, dan simpul daun merepresentasikan kelas atau distribusi kelas (Han dan Kamber, 2001). Alur pada *decision tree* ditelusuri dari simpul akar ke simpul daun yang memegang prediksi kelas. *Decision tree* mudah untuk dikonversi ke aturan klasifikasi (classification) (Lia, 2007). Sebuah *decision tree* terdiri atas :

1. Root node

Node yang tidak memiliki edge yang mengarah pada dirinya.

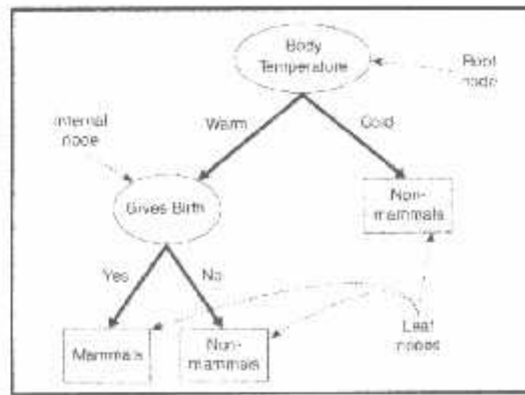
2. Internal node

Node yang memiliki sebuah *edge* yang mengarah pada dirinya dan dua atau lebih *edge* yang mengarah keluar.

3. Leaf node

Node yang memiliki satu *edge* yang menuju ke dirinya dan tidak memiliki *edge* menuju keluar.

---



Gambar 2.3 Struktur dari *Decision Tree*  
(Lia, 2007)

Pohon keputusan juga merupakan salah satu fungsional dari data mining yang menggunakan representasi tree untuk menentukan aturan – aturan klasifikasi. Ada dua tipe Pohon keputusan, yaitu *classification tree* dan *regression tree*.

*Classification tree* adalah memberi label dan memasukkan *record – record* ke dalam kelas – kelas yang telah disediakan, sedangkan *regression tree* adalah membuat estimasi nilai dari sebuah variabel target yang berdasarkan pada nilai numerik (Han dan Kamber, 2001). Pada penelitian ini menggunakan tipe *classification tree*, karena mengklasifikasi hewan diklasifikasi menggunakan *rule* untuk masing – masing kelas.

### 2.3.2 Metode Untuk Mengekspresikan Atribut

Algoritma induksi *decision tree* harus menyediakan sebuah metode untuk mengekspresikan tes atribut dan berhubungan dengan hasil untuk setiap atribut yang berbeda.

#### 1. Atribut biner

Kondisi tes untuk atribut biner menghasilkan 2 hasil keluaran (Tan dan Kumar, 2006).



Gambar 2.4 Kondisi Tes Atribut Biner.

## 2. Atribut nominal

Atribut nominal memiliki beberapa nilai, kondisi tes dapat diekspresikan ke *split* dua cara atau banyak cara. Untuk *split* banyak cara jumlah hasil keluaran berdasarkan perbedaan nilai untuk atribut yang berhubungan.



Gambar 2.5 Kondisi Tes untuk Atribut Nominal.

## 3. Atribut ordinal

Atribut ordinal juga dapat menghasilkan *split* dua cara atau banyak cara.

## 4. Atribut continuous

Untuk atribut *continuous* kondisi tes dapat diekspresikan sebagai sebuah tes perbandingan ( $A < v$ ) atau ( $A \geq v$ ) dengan hasil keluaran biner atau interval dengan hasil keluaran dalam bentuk



Gambar 2.6 Kondisi Tes Atribut *Continuous*

## 2.4. Algoritma Iterative Dichotomizer 3 (ID3)

### 2.4.1. Pengertian Algoritma Iterative Dichotomizer 3 (ID3)

*Iterative Dichotomizer 3 (ID3)* adalah algoritma *decision tree learning* (algoritma pembelajaran pohon keputusan) yang paling dasar. Algoritma ini melakukan pencarian secara rakus/ menyeluruh (greedy) pada semua kemungkinan pohon keputusan. Salah satu algoritma induksi pohon keputusan (*decision tree*) adalah *Iterative Dichotomizer 3 (ID3)* (wahyudin, 2010). Algoritma ID3 dikembangkan oleh J. Ross Quinlan.

Algoritma ID3 dapat diimplementasikan menggunakan fungsi rekursif (fungsi yang memanggil dirinya sendiri). Algoritma ID3 berusaha membangun *decision tree* (pohon keputusan) secara top – down (dari atas ke bawah), mulai dengan pertanyaan : “atribut mana yang permata kali dicek dan diletakkan di *root* ?” pertanyaan ini dijawab dengan mengevaluasi semua atribut yang ada dengan menggunakan suatu ukuran statistik (yang banyak digunakan adalah *information gain*) untuk mengukur efektivitas suatu atribut dalam mengklasifikasi kumpulan sampel data (wahyudin, 2010).

### 2.4.2. Algoritma Konstruksi Pohon

Algoritma ID3 secara rekursif membuat sebuah *decision tree* berdasarkan training data yang telah disiapkan. Algoritma ini mempunyai inputan berupa *training samples* dan *samples*. *Training samples* berupa data contoh yang akan digunakan untuk membangun sebuah tree yang telah diuji kebenarannya. Sedangkan *samples* merupakan *field – field* data yang nantinya akan digunakan sebagai parameter dalam melakukan klasifikasi data (Sunjana, 2010). Menurut M. Safii (2011) strategi pembentukan *decision tree* dengan algoritma ID3 adalah :

1. Pohon dimulai sebagai *node* tunggal (akar/*root*) yang merepresentasikan semua data.
  2. Sesudah *node root* dibentuk, maka data pada *node* akar akan diukur dengan *information gain* untuk dipilih atribut mana yang akan dijadikan atribut pembaginya.
  3. Sebuah cabang dibentuk dari atribut yang dipilih menjadi pembagi dan data akan didistribusikan ke dalam cabang masing-masing.
  4. Algoritma ini akan terus menggunakan proses yang sama atau bersifat rekursif untuk dapat membentuk sebuah *decision tree*. ketika sebuah atribut
-



Microsoft Visual Basic 6.0 menyediakan fasilitas yang memungkinkan kita menyusun sebuah program dengan memasang objek-objek grafis dalam sebuah form.

Visual Basic 6.0 berawal dari bahasa pemrograman BASIC (*Beginners All Purpose Symbolic Instruction Code*). Karena bahasa BASIC cukup mudah dipelajari dan populer, maka hampir setiap programmer menguasai bahasa ini.

Pada tahun 1980-an, sistem operasi DOS cukup populer di kalangan pengguna PC karena di dalamnya disertakan bahasa BASIC yang dikenal dengan QBASIC (*QuickBasic*). Sistem tersebut sekarang sudah jarang digunakan. Di era Windows, Microsoft menciptakan Visual Basic yang terus mengalami penyempurnaan hingga Visual Basic 6.0.

---



## BAB III

### ANALISA SISTEM DAN PERANCANGAN

#### 3.1 Deklarasi Umum

Pada subbab ini menjelaskan sistem dan perancangan yang akan dibuat yaitu metode yang digunakan adalah pohon keputusan dengan menggunakan algoritma *iterative dichotomizer 3* berbasis *visual basic* dalam mengklasifikasi hewan. Tentunya sangat bermanfaat bagi peneliti-peneliti khususnya dibidang biologi. Langkah – langkah yang dilakukan dalam penelitian ini adalah sebagai berikut :

1. Menentukan permasalahan dalam mengklasifikasi data hewan
2. Mengumpulkan data hewan dalam berbentuk catatan guna proses klasifikasi.
3. Merancang dan membangun perangkat lunak menggunakan bahasa pemrograman *microsoft visual basic* yang menerapkan metode pohon keputusan dengan algoritma *iterative dichotomizer 3*.
4. Menerapkan pengujian pada perangkat lunak tersebut.
5. Mengevaluasi dan menganalisa tingkat akurasi.

#### 3.2 Data

Data yang digunakan pada penelitian ini didapatkan dari catatan dan berbagai sumber informasi. Data yang didapatkan berupa nama binatang beserta ciri – cirinya. Terdapat 13 parameter antara lain level organisasi, simetri tubuh fase dewasa, pencernaan makanan, peredaran darah, sistem saraf, sistem reproduksi, rangka tubuh, jumlah alat gerak, kelenjar susu, bulu, sisik, cangkang literal dan carapace. Mengklasifikasi hewan digolongkan menjadi 13 type, yaitu *mamalia*, *reptil*, *aves*, *amphibi*, *osteichthyes*, *crustacea*, *chepalopoda*, *bivalvia*, *insecta*, *archnida*, *gastropoda*, *asteroidea*, dan *skyphozoa*.

#### 3.3 Sistem Perangkat Lunak

##### 3.3.1 Deskripsi Umum Sistem

Sistem ini menerapkan metode pohon keputusan dengan algoritma *iterative dichotomizer 3* dalam proses klasifikasi data hewan. Jika aturan yang sudah terbentuk didalam data latih tidak sesuai dengan ciri-ciri hewan yang diujikan, maka hewan tersebut tergolong jenis baru.

Dalam proses mengklasifikasi data hewan terdapat dua tahap. Tahap pertama adalah proses pembelajaran atau pelatihan terhadap sekumpulan data latih dan tahap berikutnya adalah proses mengklasifikasi hewan menggunakan data uji yang kemudian diklasifikasi menggunakan aturan yang sudah terbentuk dari proses pembuatan data latih.

Adapun proses yang terjadi ketika seorang user menggunakan perangkat lunak yang dibangun yaitu :

1. User memasukkan tabel data latih hewan yang kemudian disimpan dalam basisdata.
2. Sistem mengambil informasi dari data latih dan jumlah catatan.
3. Sistem menganalisa data dan melakukan pembangunan model klasifikasi dengan menggunakan algoritma *iterative dichotomizer 3*.
4. Sistem menghasilkan model klasifikasi berupa model pohon dan dikonversi ke dalam bentuk aturan.
5. Sistem melakukan proses klasifikasi terhadap data uji dengan aturan yang sudah terbentuk.
6. Sistem akan menampilkan hasil dari mengklasifikasi hewan.

### **3.4 Perancangan Sistem Perangkat Lunak**

#### **3.4.1 Perancangan proses**

Proses awal pada alur proses ini adalah sistem mendapatkan informasi dari user kemudian sistem akan melakukan proses membangun model klasifikasi kemudian sistem mengklasifikasi data tersebut. Tahap selanjutnya ialah sistem menampilkan hasil dari mengklasifikasi. Alur proses dapat dilihat pada gambar 3.1.

---



Gambar 3.1 Alur Proses

#### 3.4.1.1 Memasukkan Data

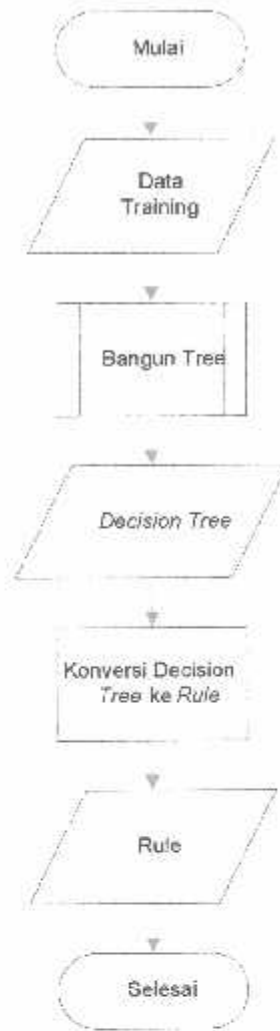
Langkah awal pada proses ini adalah memasukkan data, dimana seorang user memasukkan data kemudian sistem akan mengolah data tersebut menjadi sebuah data latih yang didapat dari informasi. Informasi berupa nama hewan, ciri-ciri hewan, nilai dari parameter hewan dan type hewan tersebut.

Data yang sudah dimasukkan oleh user akan diolah oleh sistem kedalam basis data. Proses akhir setelah sistem melakukan proses koneksi kedalam basis data ialah sistem menampilkan data latih kepada user.

#### 3.4.1.2 Proses Membangun Model Klasifikasi

Pada proses ini setelah sistem mengolah data maka akan terjadi proses membangun model klasifikasi, dimana metode yang digunakan adalah pohon keputusan. Proses yang terjadi dalam membentuk suatu pohon keputusan ialah pengujian terhadap data latih, sistem mengklasifikasi data uji terhadap data latih, pohon keputusan terbentuk dan sistem akan mengubah pohon keputusan menjadi sebuah aturan.

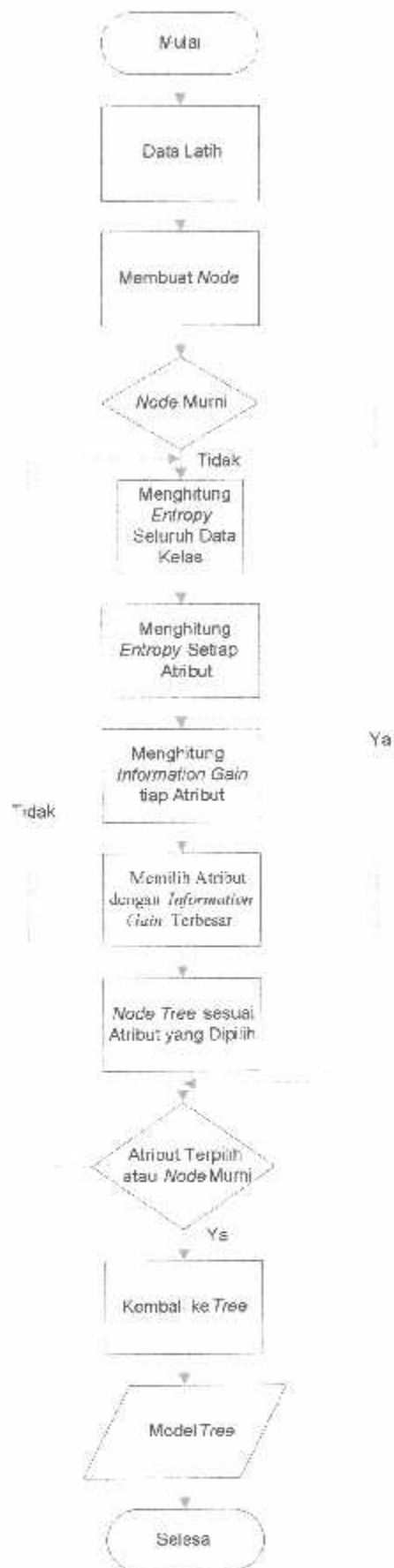
Flow chart proses membangun model klasifikasi dapat dilihat pada gambar 3.2



Gambar 3.2 Proses membangun model klasifikasi

Proses membangun pohon dengan menggunakan algoritma *iterative dichotomizer 3* dapat dilihat pada gambar 3.3

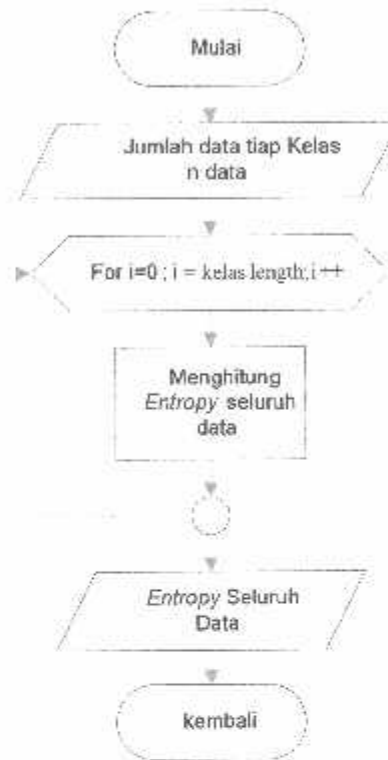




Gambar 3.3 Proses Pembangunan Pohon

### ➤ Proses Perhitungan *Entropy* dan *Information Gain*

Proses perhitungan *entropy* dan *information gain* digunakan untuk membentuk *node* pada *decision tree*. Pembentukan *node* dimulai dari *root node* dan *internal node*. Sebelum menghitung *information gain*, terlebih dahulu menghitung *entropy* seluruh data kelas, dan kemudian menghitung *entropy* setiap atribut. *Flowchart* dari perhitungan *entropy* seluruh data kelas dapat dilihat pada gambar 3.4 :

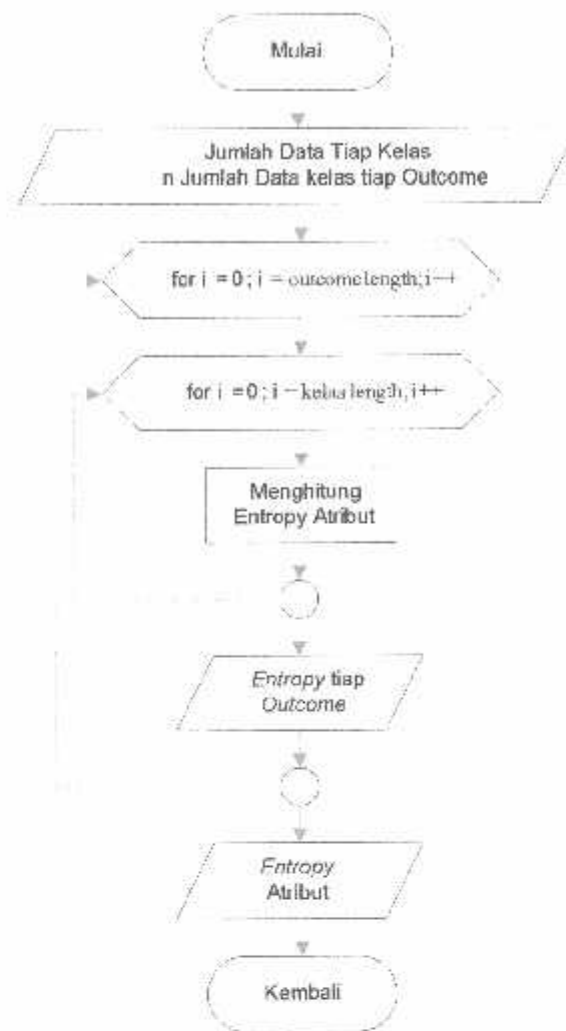


Gambar 3.4 Menghitung *entropy* seluruh data kelas

Langkah – langkah perhitungan *entropy* seluruh data kelas adalah sebagai berikut :

1. Memasukkan nilai probabilitas tiap kelas.
2. Menghitung *entropy* seluruh data kelas menggunakan persamaan 2.1.
3. Setelah melakukan perhitungan *entropy*, didapatkan nilai dari *entropy* kelas.

Setelah mendapatkan nilai *entropy* data kelas, perhitungan selanjutnya adalah *entropy* setiap atribut berdasarkan derajat keanggotaannya. *Flowchart* dari perhitungan *entropy* setiap atribut dapat dilihat pada gambar 3.5



Gambar 3.5 Menghitung *Entropy* setiap atribut

Langkah – langkah perhitungan *entropy* setiap atribut adalah sebagai berikut :

1. Memasukkan nilai probabilitas kelas dari tiap atribut.
2. Melakukan perhitungan *entropy* tiap atribut menggunakan persamaan 2.2.
3. Setelah melakukan perhitungan, didapatkan nilai dari *entropy* atribut.

Perhitungan selanjutnya adalah perhitungan *information gain* dari. *Flowchart* dari *information gain* dapat dilihat pada gambar 3.6



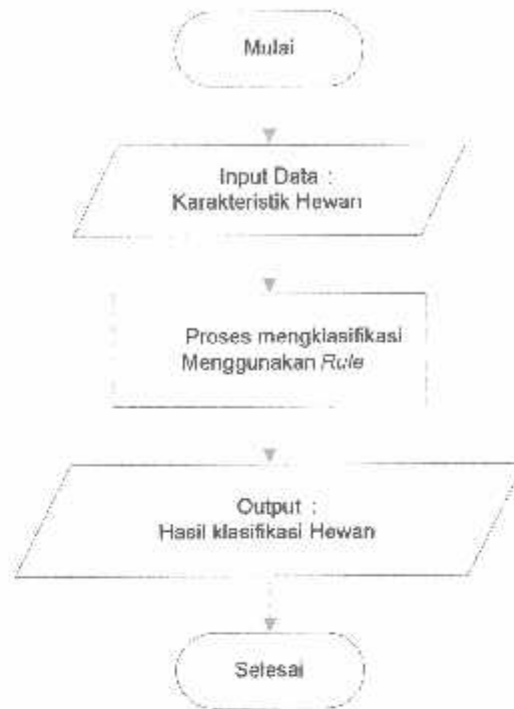
Gambar 3.6 Menghitung *Information Gain* tiap Atribut

Langkah – langkah perhitungan *entropy* setiap atribut adalah sebagai berikut :

1. Menginputkan nilai *entropy* dari data kelas dan *entropy* tiap atribut
2. Melakukan perhitungan *infomation gain* menggunakan persamaan 2.3
3. Setelah melakukan perhitungan, didapatkan nilai dari *information gain* dari tiap atribut.

#### 3.4.1.3 Proses Mengklasifikasi Hewan

Dalam proses Mengklasifikasi hewan, sistem menerima inputan data beberapa atribut, yaitu level organisasi, peredaran darah, sistem saraf, rangka tubuh, jumlah alat gerak, kelenjar susu, bulu, sisik, cangkang literal, carapace. *Flowchart* proses mengklasifikasi hewan dapat dilihat pada gambar 3.7



Gambar 3.7 Proses Mengklasifikasi Hewan

#### 3.4.1.4 Contoh perhitungan manual

Berikut ini adalah contoh perhitungan manual dalam mengklasifikasi hewan menggunakan metode *decision tree* dengan algoritma *iterative dichotomizer 3* menggunakan 26 data latih. Data latih dapat dilihat pada tabel 3.1

Tabel 3.1 Data Latih

Nama	Level organisasi	Simetri tubuh fase dewasa	Pencernaan makanan	Peredaran darah	Sistem saraf	Sistem reproduksi	Kompleks tubuh	Jumlah alat gerak	Kelajar susu	Bulu	Sisik	Cangkang Lateral	Carapak	Type
Harimau	0	0	0	0	0	0	0	4	0	1	1	1	1	1
Kuda	0	0	0	0	0	0	0	4	0	1	1	1	1	1
Ubur-abur	1	1	1	1	2	0	0	> 10	1	1	1	1	1	13
Buaya	0	0	0	0	0	0	0	4	1	1	0	1	1	3
Korokoto	0	0	0	0	0	0	0	4	1	1	0	1	1	3
Kura-kura	0	0	0	0	0	0	0	4	1	1	0	1	1	3
Laba-laba	0	0	0	1	2	0	1	8	1	1	1	1	1	10
Kanjengkung	0	0	0	1	2	0	1	8	1	1	1	1	1	10
Lelester	0	0	0	1	2	0	1	10	1	1	1	1	1	6
Kepiting	0	0	0	1	2	0	1	10	1	1	1	1	1	6
Gurila	0	0	0	0	2	0	0	8	1	1	1	1	1	7
Cumi-cumi	0	0	0	0	2	0	0	8	1	1	1	1	1	7
Siput	0	0	0	1	2	1	1	0	1	1	1	1	1	11
Bekicot	0	0	0	1	2	1	1	0	1	1	1	1	1	11
Belalang	0	0	0	1	2	0	1	6	1	1	1	1	1	9
Lebah	0	0	0	1	2	0	1	6	1	1	1	1	1	9
Nyamuk	0	0	0	1	2	0	1	6	1	1	1	1	1	9
Kerang	0	0	0	1	2	0	1	0	1	1	1	0	1	8
Keris	0	0	0	1	2	0	1	0	1	1	1	0	1	8
Ikan mas	0	0	0	0	0	0	0	8	1	1	0	1	1	5
Ikan gurami	0	0	0	0	0	0	0	8	1	1	0	1	1	5
Binisog seut	0	1	0	1	1	0	1	5	1	1	0	1	1	12
Merpati	0	0	0	0	0	0	0	4	1	0	1	1	1	2
Elang	0	0	0	0	0	0	0	4	1	0	1	1	1	2
Katak	0	0	0	0	0	0	0	4	1	1	1	1	1	4
Subramber	0	0	0	0	0	0	0	4	1	1	1	1	1	4



Keterangan :

Jumlah record = 26

Kelas hewan :

- Type 1 = kelas *Mamalia*
- Type 2 = kelas *Aves*
- Type 3 = kelas *Reptil*
- Type 4 = kelas *Amphibi*
- Type 5 = kelas *Osteichthyes*
- Type 6 = kelas *Crustacea*
- Type 7 = kelas *Cephalopoda*
- Type 8 = kelas *Bivalvia*
- Type 9 = kelas *Insecta*
- Type 10 = kelas *Archmida*
- Type 11 = kelas *Gastropoda*
- Type 12 = kelas *Asteroida*
- Type 13 = kelas *Skyphozoa*

Jumlah Data tiap Type :

- Type 1 = 2
  - Type 2 = 2
  - Type 3 = 3
  - Type 4 = 2
  - Type 5 = 2
  - Type 6 = 2
  - Type 7 = 2
  - Type 8 = 2
  - Type 9 = 3
  - Type 10 = 2
  - Type 11 = 2
  - Type 12 = 1
  - Type 13 = 1
-



Hitung *entropy outcome* 0

$$\begin{aligned}
 H_x(x) &= -2/25 * \log_2 \left( \frac{2}{25} \right) - 2/25 * \log_2 \left( \frac{2}{25} \right) - 3/25 * \log_2 \left( \frac{3}{25} \right) - 2/25 * \log_2 \left( \frac{2}{25} \right) - 2/25 \\
 &\quad * \log_2 \left( \frac{2}{25} \right) - 2/25 * \log_2 \left( \frac{2}{25} \right) - 2/25 * \log_2 \left( \frac{2}{25} \right) - 2/25 * \log_2 \left( \frac{2}{25} \right) - 3/25 * \\
 &\quad \log_2 \left( \frac{3}{25} \right) - 2/25 * \log_2 \left( \frac{2}{25} \right) - 2/25 * \log_2 \left( \frac{2}{25} \right) - 1/25 * \log_2 \left( \frac{1}{25} \right) \\
 &= 3.5434652
 \end{aligned}$$

Hitung *entropy outcome* = 1

$$\begin{aligned}
 H_x(x) &= -1/1 * \log_2 \left( \frac{1}{1} \right) \\
 &= 0
 \end{aligned}$$

$$\begin{aligned}
 G(S, A) &= H_s(s) - H_x(x) \\
 &= 3.6423714 - 25/26 (3.5647352) - 1/26(0) \\
 &= 0.2351934
 \end{aligned}$$

Tabel 3.4 distribusi kelas *simetri tubuh fase dewasa 0*

Out come	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13
0	2	2	3	2	2	2	2	2	3	2	2	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1	1

Hitung *entropy outcome* = 0

$$\begin{aligned}
 H_x(x) &= -2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 3/24 * \log_2 \left( \frac{3}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 \\
 &\quad * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 3/24 * \\
 &\quad \log_2 \left( \frac{3}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) \\
 &= -3.4387218
 \end{aligned}$$

Hitung *entropy outcome* = 1

$$\begin{aligned}
 H_x(x) &= -1/2 * \log_2 \left( \frac{1}{2} \right) - 1/2 * \log_2 \left( \frac{1}{2} \right) \\
 &= 1
 \end{aligned}$$

$$\begin{aligned}
 G(S, A) &= H_s(s) - H_x(x) \\
 &= 3.6423714 - 24/26 (3.4387218) - 2/26(1) \\
 &= 0.3912435
 \end{aligned}$$

Tabel 3.5 distribusi kelas *pencernaan makanan* 0

Out come	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13
0	2	2	3	2	2	2	2	2	3	2	2	1	0
1	0	0	0	0	0	0	0	0	0	0	0	0	1

Hitung *entropy outcome* = 0

$$\begin{aligned}
 H_x(x) &= -2/25 * \log_2\left(\frac{2}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 3/25 * \log_2\left(\frac{3}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 2/25 \\
 &\quad * \log_2\left(\frac{2}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 3/25 * \\
 &\quad \log_2\left(\frac{3}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 1/25 * \log_2\left(\frac{1}{25}\right) \\
 &= 3.5434652
 \end{aligned}$$

Hitung *entropy outcome* = 1

$$\begin{aligned}
 H_x(x) &= -1/1 * \log_2\left(\frac{1}{1}\right) \\
 &= 0
 \end{aligned}$$

$G(S, A) = H_s(s) - H_x(x)$

$$\begin{aligned}
 &= 3.6423714 - 25/26(3.5434652) - 1/26(0) \\
 &= -0.2351934
 \end{aligned}$$

Tabel 3.6 distribusi kelas *peredaran darah* 0

Out come	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13
0	2	2	3	2	2	0	2	0	0	0	0	0	0
1	0	0	0	0	0	2	0	2	3	2	2	1	1

Hitung *entropy outcome* = 0

$$\begin{aligned}
 H_x(x) &= -2/13 * \log_2\left(\frac{2}{13}\right) - 2/13 * \log_2\left(\frac{2}{13}\right) - 3/13 * \log_2\left(\frac{3}{13}\right) - 2/13 * \log_2\left(\frac{2}{13}\right) - 2/13 \\
 &\quad * \log_2\left(\frac{2}{13}\right) - 2/13 * \log_2\left(\frac{2}{13}\right) \\
 &= 2.5654483
 \end{aligned}$$

Perhitungan nilai batas ( $y <= 4$ ) or ( $y > 4$ )

$$\begin{aligned}
 H(x) &= 13/26 * (-2/13 * \log_2\left(\frac{2}{13}\right) - 2/13 * \log_2\left(\frac{2}{13}\right) - 3/13 * \log_2\left(\frac{3}{13}\right) - 2/13 * \\
 &\quad \log_2\left(\frac{2}{13}\right) - 2/13 * \log_2\left(\frac{2}{13}\right) - 2/13 * \log_2\left(\frac{2}{13}\right)) + 13/26 * (-2/13 * \log_2\left(\frac{2}{13}\right) - \\
 &\quad 2/13 * \log_2\left(\frac{2}{13}\right) - 2/13 * \log_2\left(\frac{2}{13}\right) - 3/13 * \log_2\left(\frac{3}{13}\right) - 2/13 * \log_2\left(\frac{2}{13}\right) - 1/13 * \\
 &\quad \log_2\left(\frac{1}{13}\right) - 1/13 * \log_2\left(\frac{1}{13}\right)) \\
 &= 1.282724186 + 1.35966472 \\
 &= 2.6423714
 \end{aligned}$$

$$\begin{aligned}
 G(S, A) &= H_s(s) - H_x(x) \\
 &= 3.6423714 - 26/26 (2.6423714) \\
 &= 1
 \end{aligned}$$

Perhitungan nilai batas ( $y <= 5$ ) or ( $y > 5$ )

$$\begin{aligned}
 H(x) &= 14/26 * (-2/14 * \log_2\left(\frac{2}{14}\right) - 2/14 * \log_2\left(\frac{2}{14}\right) - 3/14 * \log_2\left(\frac{3}{14}\right) - 2/14 * \\
 &\quad \log_2\left(\frac{2}{14}\right) - 2/14 * \log_2\left(\frac{2}{14}\right) - 2/14 * \log_2\left(\frac{2}{14}\right) - 1/14 * \log_2\left(\frac{1}{14}\right)) + 12/26 * (- \\
 &\quad 2/12 * \log_2\left(\frac{2}{12}\right) - 2/12 * \log_2\left(\frac{2}{12}\right) - 2/12 * \log_2\left(\frac{2}{12}\right) - 3/12 * \log_2\left(\frac{3}{12}\right) - 2/12 * \\
 &\quad \log_2\left(\frac{2}{12}\right) - 1/12 * \log_2\left(\frac{1}{12}\right)) \\
 &= 1.4826185 + 1.1640254 \\
 &= 2.6466439
 \end{aligned}$$

$$\begin{aligned}
 G(S, A) &= H_s(s) - H_x(x) \\
 &= 3.6423714 - 26/26(2.6466439) \\
 &= 0.9957274
 \end{aligned}$$

Perhitungan nilai batas ( $y <= 6$ ) or ( $y > 6$ )

$$\begin{aligned}
 H(x) &= 17/26 * (-2/17 * \log_2\left(\frac{2}{17}\right) - 2/17 * \log_2\left(\frac{2}{17}\right) - 3/17 * \log_2\left(\frac{3}{17}\right) - 2/17 * \\
 &\quad \log_2\left(\frac{2}{17}\right) - 2/17 * \log_2\left(\frac{2}{17}\right) - 3/17 * \log_2\left(\frac{3}{17}\right) - 2/17 * \log_2\left(\frac{2}{17}\right) - 1/17 * \\
 &\quad \log_2\left(\frac{1}{17}\right)) + 9/26 * (-2/9 * \log_2\left(\frac{2}{9}\right) - 2/9 * \log_2\left(\frac{2}{9}\right) - 2/9 * \log_2\left(\frac{2}{9}\right) - 2/9 * \\
 &\quad \log_2\left(\frac{2}{9}\right) - 1/9 * \log_2\left(\frac{1}{9}\right)) \\
 &= 1.922195 + 0.7895894 \\
 &= 2.7117853
 \end{aligned}$$

$$\begin{aligned}
 G(S, A) &= H_s(s) - H_x(x) \\
 &= 3.6423714 - 26/26(2.7117853) \\
 &= 0.9305861
 \end{aligned}$$

Perhitungan nilai batas ( $y \leq 8$ ) or ( $y > 8$ )

$$\begin{aligned}
 H(x) &= 23/26 * (-2/23 * \log_2\left(\frac{2}{23}\right) - 2/23 * \log_2\left(\frac{2}{23}\right) - 3/23 * \log_2\left(\frac{3}{23}\right) - 2/23 * \\
 &\quad \log_2\left(\frac{2}{23}\right) - 2/23 * \log_2\left(\frac{2}{23}\right) - 2/23 * \log_2\left(\frac{2}{23}\right) - 2/23 * \log_2\left(\frac{2}{23}\right) - 3/23 * \\
 &\quad \log_2\left(\frac{3}{23}\right) - 2/23 * \log_2\left(\frac{2}{23}\right) - 2/23 * \log_2\left(\frac{2}{23}\right) - 1/23 * \log_2\left(\frac{1}{23}\right)) + 3/26 * (-2/3 \\
 &\quad * \log_2\left(\frac{2}{3}\right) - 1/3 * \log_2\left(\frac{1}{3}\right)) \\
 &= 3.0204673 + 0.1059572 \\
 &= 3.1264245
 \end{aligned}$$

$$\begin{aligned}
 G(S, A) &= H_s(s) - H_x(x) \\
 &= 3.6423714 - 26/26(3.1264245) \\
 &= 0.515946
 \end{aligned}$$

Perhitungan nilai batas ( $y \leq 10$ ) or ( $y > 10$ )

$$\begin{aligned}
 H(x) &= 25/26 * (-2/25 * \log_2\left(\frac{2}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 3/25 * \log_2\left(\frac{3}{25}\right) - 2/25 * \\
 &\quad \log_2\left(\frac{2}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 2/25 * \\
 &\quad \log_2\left(\frac{2}{25}\right) - 3/25 * \log_2\left(\frac{3}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 2/25 * \log_2\left(\frac{2}{25}\right) - 1/25 * \\
 &\quad \log_2\left(\frac{1}{25}\right)) + 1/26 * (-1/1 * \log_2\left(\frac{1}{1}\right)) \\
 &= 3.4071780
 \end{aligned}$$

$$\begin{aligned}
 G(S, A) &= H_s(s) - H_x(x) \\
 &= 3.6423714 - 26/26(3.4071780) \\
 &= 0.2351933
 \end{aligned}$$

Tabel 3.11 Hasil perhitungan nilai batas jumlah alat gerak 0

Nilai Batas	Information Gain
( $y \leq 4$ ) or ( $y > 4$ )	<b>1</b>
( $y \leq 5$ ) or ( $y > 5$ )	0.9957274
( $y \leq 6$ ) or ( $y > 6$ )	0.9305861





Hitung entropy outcome = 0

$$\begin{aligned}
 H_x(x) &= - 2/24 * \log_2 \left( \frac{2}{24} \right) - 3/24 * \log_2 \left( \frac{3}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 \\
 &\quad * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 3/24 * \log_2 \left( \frac{3}{24} \right) - 2/24 * \\
 &\quad \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 1/24 * \log_2 \left( \frac{1}{24} \right) - 1/24 * \log_2 \left( \frac{1}{24} \right) \\
 &= 3.5220552
 \end{aligned}$$

Hitung entropy outcome = 1

$$\begin{aligned}
 H_x(x) &= - 2/2 * \log_2 \left( \frac{2}{2} \right) \\
 &= 0
 \end{aligned}$$

$G(S, A) = H_s(s) - H_x(x)$

$$\begin{aligned}
 &= 3.6423714 - 24/26(3.5220552) - 2/26 (0) \\
 &= 0.3912436
 \end{aligned}$$

Tabel 3.14 distribusi kelas *Sisik 0*

Out come	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13
0	2	2	0	2	0	2	2	2	3	2	2	0	1
1	0	0	3	0	2	0	0	0	0	0	0	1	0

Hitung entropy outcome = 0

$$\begin{aligned}
 H_x(x) &= - 2/20 * \log_2 \left( \frac{2}{20} \right) - 2/20 * \log_2 \left( \frac{2}{20} \right) - 2/20 * \log_2 \left( \frac{2}{20} \right) - 2/20 * \log_2 \left( \frac{2}{20} \right) - 2/20 \\
 &\quad * \log_2 \left( \frac{2}{20} \right) - 2/20 * \log_2 \left( \frac{2}{20} \right) - 3/20 * \log_2 \left( \frac{3}{20} \right) - 2/20 * \log_2 \left( \frac{2}{20} \right) - 2/20 * \\
 &\quad \log_2 \left( \frac{2}{20} \right) - 1/20 * \log_2 \left( \frac{1}{20} \right) \\
 &= 3.2841837
 \end{aligned}$$

Hitung entropy outcome = 1

$$\begin{aligned}
 H_x(x) &= - 3/6 * \log_2 \left( \frac{3}{6} \right) - 2/6 * \log_2 \left( \frac{2}{6} \right) - 1/6 * \log_2 \left( \frac{1}{6} \right) \\
 &= 1.4591479
 \end{aligned}$$

$G(S, A) = H_s(s) - H_x(x)$

$$\begin{aligned}
 &= 3.6423714 - 20/26(3.2841837) - 6/26 (1.4591479) \\
 &= 0.7793498
 \end{aligned}$$

Tabel 3.15 distribusi kelas *Cangkang literal 0*

Out come	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13
0	2	2	3	2	2	2	2	0	3	2	2	1	1
1	0	0	0	0	0	0	0	2	0	0	0	0	0

Hitung entropy outcome = 0

$$\begin{aligned}
 H_x(x) &= -2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 3/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 \\
 &\quad * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{3}{24} \right) - 3/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \\
 &\quad \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{1}{24} \right) - 1/24 * \log_2 \left( \frac{1}{24} \right) - 1/24 * \log_2 \left( \frac{1}{24} \right) \\
 &= 3,522055
 \end{aligned}$$

Hitung entropy outcome = 1

$$\begin{aligned}
 H_x(x) &= -2/2 * \log_2 \left( \frac{2}{2} \right) \\
 &= 0
 \end{aligned}$$

$G(S, A) = H_s(s) - H_x(x)$

$$\begin{aligned}
 &= 3.6423714 - 24/26(3,522055) - 6/26 (0) \\
 &= 0.7793498
 \end{aligned}$$

Tabel 3.16 distribusi kelas *Carapak 0*

Out come	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13
0	2	2	3	2	2	0	2	2	3	2	2	1	1
1	0	0	0	0	0	2	0	0	0	0	0	0	0

Hitung entropy outcome = 0

$$\begin{aligned}
 H_x(x) &= -2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 3/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 \\
 &\quad * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{3}{24} \right) - 3/24 * \log_2 \left( \frac{2}{24} \right) - 2/24 * \\
 &\quad \log_2 \left( \frac{2}{24} \right) - 2/24 * \log_2 \left( \frac{1}{24} \right) - 1/24 * \log_2 \left( \frac{1}{24} \right) - 1/24 * \log_2 \left( \frac{1}{24} \right) \\
 &= 3,522055
 \end{aligned}$$

Hitung entropy outcome = 1

$$H_x(x) = - 2/2 * \log_2 \left( \frac{2}{2} \right)$$

$$= 0$$

$$G(S, A) = H_s(s) - H_x(x)$$

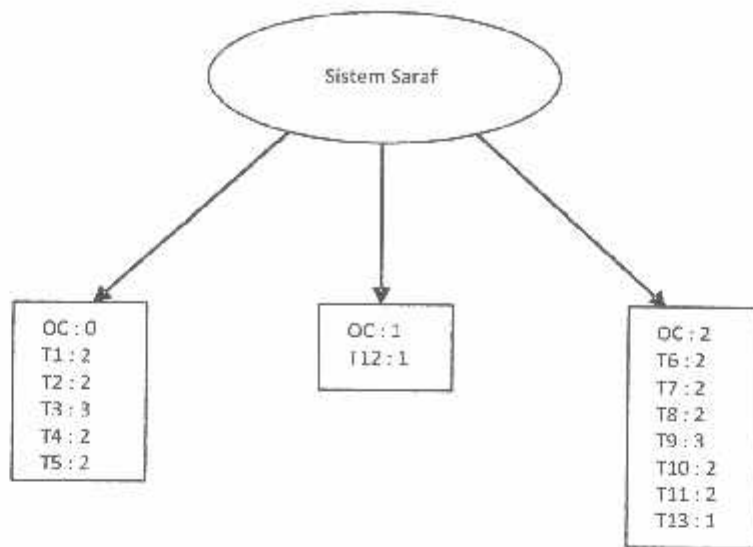
$$= 3.6423714 - 24/26(3,522055) - 6/26 (0)$$

$$= 0.7793498$$

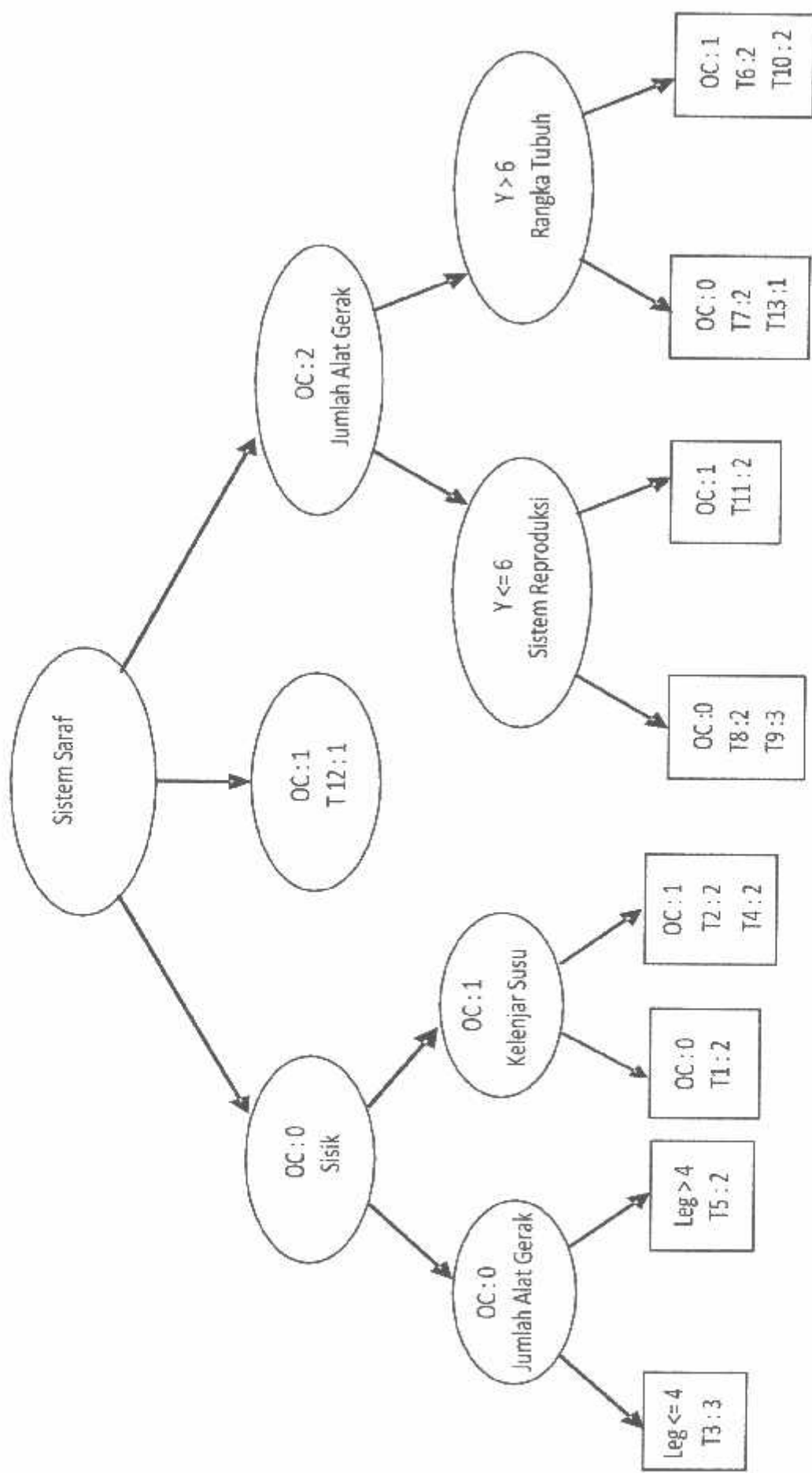
Tabel 3.17 nilai information gain dari masing – masing kelas atribut iterasi 0

Kelas/ atribut	Information gain
Level Organisasi	0.2351934
Simetri Tubuh Fase Dewasa	0.3912435
Pencernaan Makanan	0.2351934
Peredaran Darah	1
Sistem Saraf	<b>1.1867198</b>
Sistem Reproduksi	0.6219041
Rangka Tubuh	0.9957274
Jumlah Alat Gerak	1
Kelenjar Susu	0.391243
Bulu	0.3912435
Sisik	0.7793498
Cangkang Literal	0.7793498
Carapak	0.7793498

Karena kelas/ atribut *Sistem Saraf* mempunyai nilai *information gain* terbesar, maka atribut *Sistem Saraf* dipilih sebagai *root* dari klasifikasi hewan.



Gambar 3.8 *decision tree* iterasi - 0



Gambar 3.9 Model Decision Tree



### 3.4.1.5 Konversi Decision Tree Menjadi Rule

Setelah melakukan klasifikasi menggunakan algoritma *Iterative Dichotomizer 3 (ID3)* terhadap data uji, akan terbentuk *decision tree* seperti yang terlihat pada gambar diatas. Kemudian *decision tree* akan dikonversikan ke dalam bentuk *rule*.

If sistem saraf = 1

Then type = 12

Else if sistem saraf = 0 AND sisik = 0 AND jumlah alat gerak <= 4

Then type = 3

Else if sistem saraf = 0 AND sisik = 0 AND jumlah alat gerak > 4

Then type = 5

Else if sistem saraf = 0 AND sisik = 1 AND kelenjar susu = 0

Then type = 1

Else if sistem saraf = 0 AND sisik = 1 AND kelenjar susu = 1 AND bulu = 0

Then type = 2

Else if sistem saraf = 0 AND sisik = 1 AND kelenjar susu = 1 AND bulu = 1

Then type = 4

Else if sistem saraf = 2 AND jumlah alat gerak <= 6 AND sistem reproduksi = 0 AND cangkang literal = 0

Then type = 8

### 3.5 Analisa data

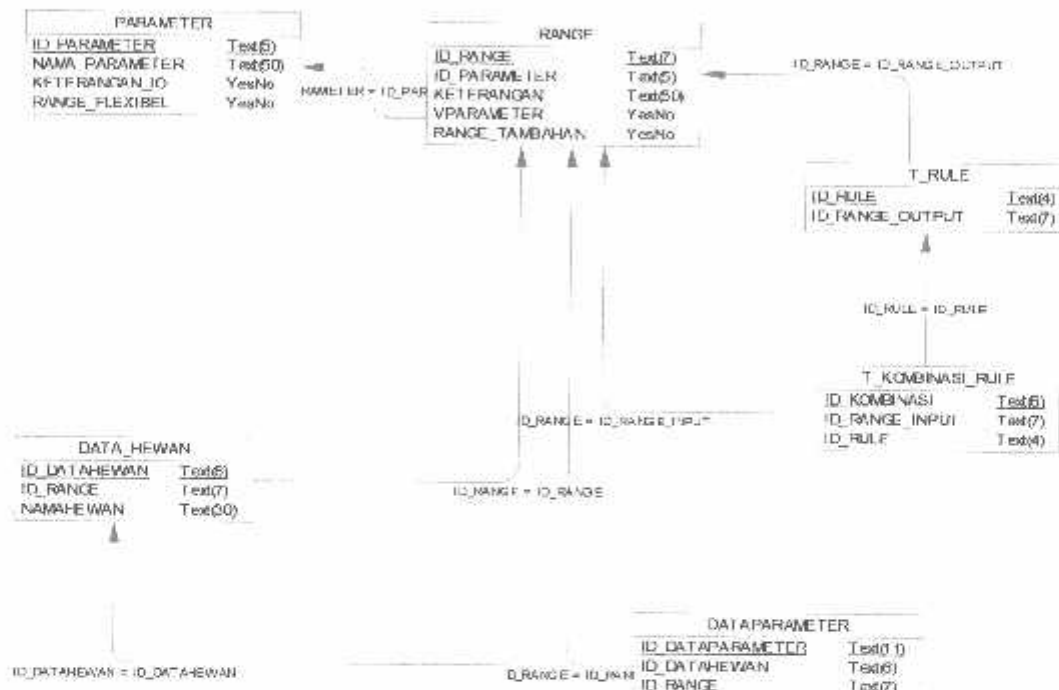
Mengklasifikasi data dilakukan pada saat mengklasifikasikan data latih (*training data*) dan data uji (*testing data*). Pada saat klasifikasi data latih, terjadi proses pembentukan aturan. Pada proses ini sistem mengklasifikasi data latih menggunakan algoritma *Iterative Dichotomizer 3 (ID3)*. Dari hasil klasifikasi terbentuk model *decision tree* dan kemudian dikonversikan kedalam *rule - rule* yang digunakan untuk mengklasifikasi data uji.

Pada saat mengklasifikasi data uji, proses yang terjadi adalah sistem menyimpan semua atribut data uji. Kemudian data uji diklasifikasikan menggunakan *rule* yang terbentuk dari model *decision tree* dan kemudian data dikelompokkan sesuai kelasnya. Setelah data uji diklasifikasi, hasil yang didapatkan adalah nama hewan digolongkan berdasarkan kelasnya. Kelas hewan yang ditampilkan sesuai dengan jenis hewan yang dimasukkan.

### 3.6 Perancangan Database

#### 3.6.1 Physical Data Model Klasifikasi Hewan

*Physical data model* merupakan model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Setiap tabel mempunyai sejumlah kolom di mana setiap kolom memiliki nama yang unik. Sebuah *physical data model* (desain database) adalah representasi dari desain data yang memperhitungkan fasilitas dan kendala sistem database yang diberikan manajemen. Dapat dilihat pada gambar 3.10



Gambar 3.10 *Physical Data Model* Klasifikasi Hewan

### 3.7 Perancangan Antarmuka

Pada bagian ini akan dijelaskan rancangan antarmuka yang akan dibuat. Pada gambar 3.11 adalah rancangan antarmuka untuk memasukkan data latihan.

Parameter input

Keterangan I/O

No Parameter

Nama Parameter

Range Flexibel

Save | Cancel | Delete

No Atribut	Nama Atribut	Keterangan

Gambar 3.11 Rancangan Antarmuka Parameter Input

Penjelasan rancangan antarmuka pengujian data latih dijelaskan sebagai berikut :

1. *User* memasukkan sejumlah nama parameter, nomor parameter, keterangan parameter binatang.
2. Tombol *save* digunakan untuk menyimpan nomor, nama, keterangan dari parameter binatang tersebut yg nantinya akan diproses kedalam pohon *Tree*.
3. Tombol *cancel* digunakan untuk membatalkan parameter yang berada pada tabel.
4. Tombol *Delete* digunakan untuk menghapus parameter yang berada pada tabel.

Range Input

Nama Parameter

Keterangan I/O

Keterangan

Parameter Bayangan

Save Cancel Delete

No Parameter	NamaParameter	Parameter Bayangan

Gambar 3.12 Rancangan Antarmuka Range Input

Pada tabel diatas dapat dijelaskan sebagai berikut:

1. *User* memilih parameter yang sudah terbentuk dari rancangan parameter input.
2. *User* memasukkan nomer atau outcome dari parameter.
3. *User* memberikan keterangan dari range parameter.
4. *User* dapat menyimpan, membatalkan, dan menghapus dari parameter tersebut.

No	Nama Hewan	Parameter 1	Parameter 2

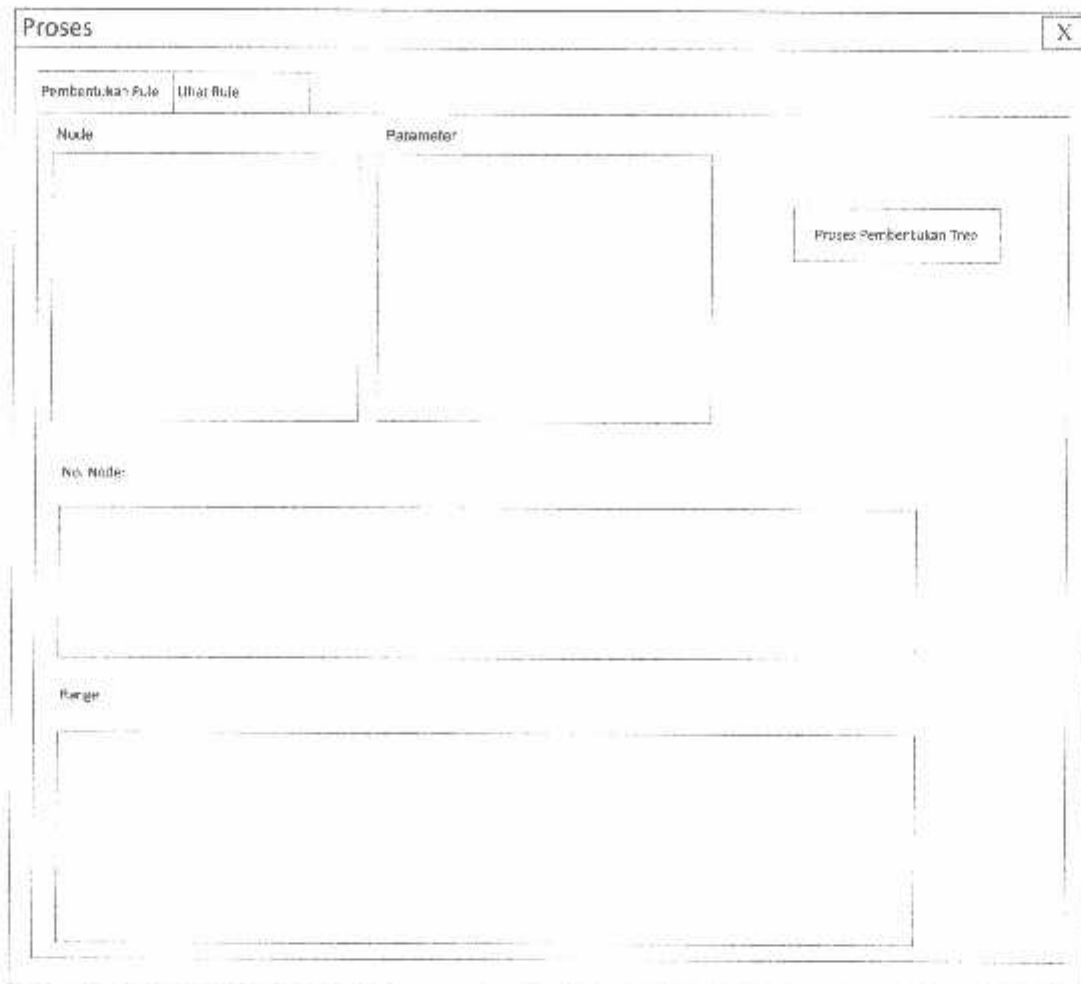
  

No.	<input type="text"/>	<input type="button" value="Insert Data"/>	<input type="button" value="Add Data"/>
Nama Hewan	<input type="text"/>	<input type="button" value="Delete label"/>	<input type="button" value="Save Data"/>
Parameter	<input type="text"/>	<input type="button" value="Delete label"/>	<input type="button" value="Save Data"/>
	<input type="text"/>	<input type="button" value="Delete label"/>	<input type="button" value="Save Data"/>

Gambar 3.13 Rancangan Antarmuka Data Hewan

Pada gambar tersebut dapat dijelaskan sebagai berikut:

1. *User* memasukkan nomer, nama hewan, atribut, dan juga parameternya.
2. Tombol *Insert* tabel digunakan untuk menyisipkan nomer, nama hewan, parameter, dan juga *range* kedalam tabel yang sudah terisi.
3. Tombol *Add tabel* digunakan untuk menambahkan sebuah tabel.
4. Tombol *Save data* digunakan untuk menyimpan data tabel.
5. Tombol *Delete* tabel digunakan untuk menghapus tabel.



Gambar 3.14 Rancangan Antarmuka Proses

Pada gambar tersebut dapat dijelaskan sebagai berikut:

1. Tombol proses pembentukan *tree* digunakan untuk mengetahui node pertama kali yang terpilih dan sampai *node* yang terakhir.
2. Tombol lihat aturan berfungsi untuk melihat aturan yang terbentuk dari proses *ID 3*.
3. Tombol lihat kode digunakan untuk menampilkan parameter, range, dan kelas hewan tersebut.



The screenshot shows a window titled "Pengujian" with a close button (X) in the top right corner. Below the title bar is a section labeled "Data" containing a table with four columns: "No", "Nama", "Parameter", and "Hasil Akurasi". The table is currently empty. Below the table, there are four input fields: "No" (a small text box), "Nama Hewan" (a larger text box), "Parameter" (a dropdown menu), and "Range" (a dropdown menu). To the right of these input fields are four buttons: "Insert Data", "Add Data", "Delete Tabel", and "Proses".

Gambar 3.15 Rancangan Antarmuka Pengujian

Pada tampilan interface tersebut diketahui penjelasan sebagai berikut:

1. User memasukkan nomer, nama hewan, parameter hewan, dan juga range hewan sebagai data uji.
2. Tombol proses digunakan untuk memproses data uji dan tingkat akurasi.
3. Pada kolom tabel akan menampilkan hasil proses dari data uji berupa kelas hewan.
4. Pada kolom hasil akurasi menampilkan tingkat akurasi dari data yang diujikan kedalam data latih.



## BAB IV

### IMPLEMENTASI DAN PENGUJIAN SISTEM

#### 4.1 Implementasi Sistem

Pada bab ini akan dijelaskan tahap pembuatan dari rancangan aplikasi untuk mengklasifikasi hewan, sehingga menjadi sebuah aplikasi yang utuh dan dapat digunakan oleh pengguna. Dalam sistem aplikasi ini, program yang digunakan dalam pengimplementasian adalah menggunakan *Visual Basic 6.0*. Sebagai media penyimpanan data (basisdata) aplikasi ini menggunakan basis data *Microsoft Access 2007*, yang di dalamnya terdapat beberapa tabel yang saling berhubungan.

##### 4.1.1. Koneksi Basis Data

*Activex Data Objects Connection* adalah program utama yang digunakan untuk koneksi ke *database* tabel hewan.

```
Global Mydb As ADODB.Connection
Global Myrs As ADODB.Recordset
Private Sub Open_Database()
    Dim Sqlcmd As String
    Set Mydb = New ADODB.Connection
    Set Myrs = New ADODB.Recordset
    Mydb.CursorLocation = adUseClient
    Mydb.ConnectionString = "Provider=Microsoft.Jet.OLEDB.4.0; " & _
    "Persist Security Info=False;Data Source=" & App.Path & _
    "\Hewan.mdb;Mode=readwrite"
    Mydb.Open
    Sqlcmd = "Select*from Parameter"
    Set Myrs = Mydb.Execute(Sqlcmd)
End Sub
```

Setiap unit program harus mengakses program di atas dengan membuat objek baru dari *ADODB.Connection* dan memanggil fungsi *ADODB.Connection* dan *ADODB.Recordset*.

##### 4.1.2. Pengoperasian Basis Data

*Class Module* merupakan proses utama dari pengoperasian basis data antara lain seperti pembacaan data, penyimpanan data, pengubahan data, dan penghapusan data.

Setiap unit program harus mendefinisikan objek baru dari kelas *Module* kemudian memanggil fungsi atau prosedur yang akan digunakan.

```
Private Type Parameter_Rec
  ID_Parameter As String
  Nama_Parameter As String
  Range_Flexibel As Boolean
  Keterangan_IO As Byte
End Type
Dim Parameter As Parameter_Rec
```

## 4.2. Setting Parameter

### 4.2.1 Setting Parameter I/O

#### 4.2.1.1 Input Parameter

Prosedur dalam mengisi masukan input parameter

1. Pilih label keterangan.
2. Isi label keterangan.
3. Tekan save untuk menyimpan.
4. Tekan cancel jika batal.
5. Pilih range dan tekan delete jika ingin menghapus range yang telah tersimpan.

Proses awal untuk memasukkan parameter keterangan input pada label dipilih seperti pada gambar 4.1 dan 4.3. Hasil menginputkan parameter Level Organisasi bisa dilihat pada gambar 4.2 dan 4.4.

---

MDIForm1

Master Data | Pelatihan | Pengujian

Setting Parameter

Keterangan I/O: 1

No Parameter: 0

Nama Parameter: 0

Range Flexibel:

Save | Cancel

ID Parameter	Nama Parameter	Keterangan I/O	Range Flexibel

Gambar 4.1 Memasukkan Keterangan Input pada I/O

MDIForm1

Master Data | Pelatihan | Pengujian

Setting Parameter

Keterangan I/O: 1

No Parameter: (0)

Nama Parameter: LEVEL ORGANISASI

Range Flexibel:

Save | Cancel

ID Parameter	Nama Parameter	Keterangan I/O	Range Flexibel

Gambar 4.2 Input Parameter Level Organisasi.

MDIForm1  
Master Data Pelatihan Pengujian

Setting Parameter

Keterangan I/O: I

No Parameter: 1

Nama Parameter: LEVEL ORGANISASI

Range Flexibel:

Save Cancel

ID Parameter	Nama Parameter	Keterangan I/O	Range Flexibel
1001	LEVEL ORGANISASI	INPUT	TETAP

Gambar 4.3 Hasil Input Parameter Level Organisasi

MDIForm1  
Master Data Pelatihan Pengujian

Setting Parameter

Keterangan I/O: I

No Parameter: 100

Nama Parameter: SIMETRI TUBUH FASE DEV

Range Flexibel:

Save Cancel

ID Parameter	Nama Parameter	Keterangan I/O	Range Flexibel
1001	LEVEL ORGANISASI	INPUT	TETAP

Gambar 4.4 Input Parameter Simetri Tubuh Fase Dewasa.

ID Parameter	Nama Parameter	Keterangan I/O	Range Fleksibel
1-001	LEVEL ORGANISASI	INPUT	TETAP
1-002	SIMETRI TUBUH FASE DEWASA	INPUT	TETAP

Gambar 4.5 Hasil Input Parameter Simetri Tubuh Fase Dewasa.

#### 4.2.1.2 Output Parameter

Untuk menginputkan keterangan output parameter maka keterangan I/O diubah menjadi O seperti pada gambar 4.6

ID Parameter	Nama Parameter	Keterangan I/O	Range Fleksibel
--------------	----------------	----------------	-----------------

Gambar 4.6 Memasukkan Keterangan Output pada I/O



MDIForm1

Master Data   Pelatihan   Pengujian

Setting Parameter

Keterangan I/O: [0]

No Parameter: [000]

Nama Parameter: TYPE

Range Fleksibel:

Update   Delete   Cancel

ID Parameter	Nama Parameter	Keterangan I/O	Range Fleksibel
▶ 000	TYPE	OUTPUT	TETAP

Gambar 4.7 Hasil Input Parameter 'Type pada Keterangan I/O.

Proses awal untuk menghapus parameter adalah menandai atau memilih parameter tersebut lalu menghapus seperti pada gambar 4.8.

MDIForm1

Master Data   Pelatihan   Pengujian

Setting Parameter

Keterangan I/O: [1]

No Parameter: [000]

Nama Parameter: SIMETRI TUBUH FASE DEV

Range Fleksibel:

Update   Delete   Cancel

ID Parameter	Nama Parameter	Keterangan I/O	Range Fleksibel
001	LEVEL DRUMHAGAS	INPUT	TETAP
▶ 002	SIMETRI TUBUH FASE DEWASA	INPUT	TETAP

Gambar 4.8 Proses Menghapus Parameter.

## 4.2.2 Setting Range

### 4.2.2.1 Input Range

Prosedur dalam mengisi masukan range parameter input

1. Pilih nama parameter.
2. Isi label keterangan.
3. Tekan save untuk menyimpan.
4. Tekan cancel jika batal.
5. Pilih range dan tekan delete jika ingin menghapus range yang telah tersimpan.

Langkah-langkah dalam memasukkan range parameter input seperti pada contoh Level Organisasi. Gambar 4.9 - gambar 4.13 menunjukkan langkah awal sampai langkah terakhir memasukkan range parameter Level Organisasi.

No Range	Keterangan	Parameter Bayangan

Gambar 4.9 Proses memilih nama parameter Level Organisasi.

MDIForm1

Master Data Pelabihan Pengujian

Form4

Nama Parameter: [LEVEL ORGANISASI]

Keterangan I/O: [I/O]

Keterangan: [SISTEM ORGAN]

Parameter Bayangan:

Save Cancel

No Range	Keterangan	Parameter Bayangan

Gambar 4.10 Proses Input Keterangan Sistem Organ pada Level Organisasi.

MDIForm1

Master Data Pelabihan Pengujian

Form4

Nama Parameter: [LEVEL ORGANISASI]

Keterangan I/O: [I/O]

Keterangan: [ ]

Parameter Bayangan:

Save Cancel

No Range	Keterangan	Parameter Bayangan
▶ 1-0011	SISTEM ORGAN	TIDAK

Gambar 4.11 Hasil Input Range 1 pada Level Organisasi.

MDIForm1

Master Data   Pelatihan   Pengujian

Form4

Nama Parameter: I-0011: DE-G/IN 522

Keterangan I/O: INPUT

Keterangan: JARINGAN

Parameter Bayangan:

Save   Cancel

No Range	Keterangan	Parameter Bayangan
I-0011	SISTEM ORGAN	TIDAK

Gambar 4.12 Proses Input Keterangan Jaringan pada Level Organisasi

MDIForm1

Master Data   Pelatihan   Pengujian

Form4

Nama Parameter: I-0011: ORGANISASI

Keterangan I/O: INPUT

Keterangan:

Parameter Bayangan:

Save   Cancel

No Range	Keterangan	Parameter Bayangan
I-0011	SISTEM ORGAN	TIDAK
I-0012	JARINGAN	TIDAK

Gambar 4.13 Hasil Input Range 1 dan 2 pada Level Organisasi

Langkah-langkah dalam memasukkan range parameter input seperti pada contoh Simetri Tubuh Fase Dewasa. Gambar 4.14 - gambar 4.18 menunjukkan langkah awal sampai langkah terakhir memasukkan range parameter Simetri Tubuh Fase Dewasa.

The screenshot shows a window titled 'MDIForm1' with a menu bar containing 'Master Data', 'Pelatihan', and 'Pengujian'. The main area is labeled 'Form4' and contains the following fields:

- Nama Parameter:** A dropdown menu with a list of options including 'LEVEL ORGANISASI', 'PERCEPATAN MAKANA', 'PEREDARAN DARAH', 'SISTEM SARAF', 'SISTEM REPRODUKSI', 'RANGKA TUBUH', and 'JUMLAH KLAT GERAK'.
- Keterangan I/O:** A text field containing 'LEVEL ORGANISASI'.
- Keterangan:** A text field containing 'PERCEPATAN MAKANA'.
- Parameter Bayangan:** A text field containing 'PEREDARAN DARAH'.

Below these fields is a table with three columns: 'No Range', 'Keterangan', and 'Parameter Bayangan'. The table is currently empty.

Gambar 4.14 Proses Memilih Nama Parameter Simetri Tubuh Fase Dewasa.

The screenshot shows the same 'MDIForm1' window. The fields are now populated with the following values:

- Nama Parameter:** 'SIMETRI TUBUH - FASE'.
- Keterangan I/O:** 'LEVEL'.
- Keterangan:** 'BILATERAL'.
- Parameter Bayangan:** An empty text field.

Below the fields are two buttons: 'Save' and 'Cancel'. The table below remains empty.

Gambar 4.15 Proses Input keterangan Bilateral pada Simetri Tubuh Fase Dewasa.

MDIForm1

Master Data Pelatihan Pengujian

Form4

Nama Parameter: SIMETRI TUBUH FASE

Keterangan I/O: INPUT

Keterangan:

Parameter Bayangan:

Save Cancel

No Range	Keterangan	Parameter Bayangan
▶ 1-0021	BILATERAL	TIDAK

Gambar 4.16 Hasil Input Range 1 pada Simetri Tubuh Fase Dewasa.

MDIForm1

Master Data Pelatihan Pengujian

Form4

Nama Parameter: SIMETRI TUBUH FASE

Keterangan I/O: INPUT

Keterangan: RADIAL

Parameter Bayangan:

Save Cancel

No Range	Keterangan	Parameter Bayangan
▶ 1-0021	BILATERAL	TIDAK

Gambar 4.17 Proses Input Keterangan Radial Pada Simetri Tubuh Fase Dewasa.

The screenshot shows a software window titled 'MDIForm1' with a menu bar containing 'Master Data', 'Pelatihan', and 'Pengujian'. Inside the window, there is a sub-window titled 'Form4'. The form contains the following fields:

- Nama Parameter:** A text box containing 'SIMPETRI TUBUH FASE'.
- Keterangan I/O:** A text box containing 'BPTI'.
- Keterangan:** An empty text box.
- Parameter Bayangan:** A checkbox that is currently unchecked.

Below the form are two buttons: 'Save' and 'Cancel'. At the bottom of the window is a table with the following data:

No Range	Keterangan	Parameter Bayangan
I-0021	BILATERAL	TIDAK
I-0022	RADIAL	TIDAK

Gambar 4.18 Hasil Input Range 1 dan 2 Pada Simetri Tubuh Fase Dewasa.

#### 4.2.2.2 Output Range

Prosedur dalam mengisi masukan range parameter output

1. Pilih nama parameter.
2. Isi label keterangan.
3. Tekan save untuk menyimpan.
4. Tekan cancel jika batal
5. Pilih range dan tekan delete jika ingin menghapus range yang telah tersimpan.

Langkah-langkah dalam memasukkan range parameter output pada type dengan mengisi nama-nama kelas hewan. Gambar 4.19 - gambar 4.22 menunjukkan langkah awal sampai langkah terakhir memasukkan range parameter Type

MDIForm1

Master Data Pelatihan Pengujian

Form4

Nama Parameter

Keterangan I/O

Keterangan

Parameter Bayangan

No Range	Keterangan	Parameter Bayangan

Gambar 4.19 Proses Memilih Nama Parameter Type.

MDIForm1

Master Data Pelatihan Pengujian

Form4

Nama Parameter

Keterangan I/O

Keterangan

Parameter Bayangan

Save Cancel

No Range	Keterangan	Parameter Bayangan

Gambar 4.20 Proses Input Mamalia pada Parameter Type.



MDIForm1

Master Data Pelatihan Pengujian

Form4

Nama Parameter: TYPE

Keterangan I/O: INPUT

Keterangan:

Parameter Bayangan:

Save Cancel

No Range	Keterangan	Parameter Bayangan
▶ 0-0011	MAMALIA	TIDAK

Gambar 4.21 Hasil Input Range 1 Pada Type.

MDIForm1

Master Data Pelatihan Pengujian

Form4

Nama Parameter: TYPE

Keterangan I/O: OUTPUT

Keterangan: GASTROPODA

Parameter Bayangan:

Save Cancel

No Range	Keterangan	Parameter Bayangan
▶ 0-0011	MAMALIA	TIDAK

Gambar 4.22 Proses Input Gastropoda pada Parameter Type.

No Range	Keterangan	Parameter Bayangan
0-0011	MAMALIA	TIDAK
0-0012	GASTROPODA	TIDAK

Gambar 4.23 Hasil Input Range 1 dan 2 Pada Parameter Type.

#### 4.2.3 Setting Data Hewan

Prosedur memasukkan data hewan

1. Masukkan nama hewan pada label nama hewan
2. Pilih parameter I/O
3. Pilih range parameter
4. Memasukkan data
5. Tekan *save* untuk menyimpan
6. *Add data* untuk menambah tabel hewan
7. Lakukan 1-6 untuk mengisi semua data hewan
8. Jika ingin menghapus tabel tekan *delete*

Langkah-langkah memasukkan data hewan seperti pada gambar 4.24 sampai gambar 4.31 dan gambar 4.32 yaitu untuk menghapus tabel, serta gambar 4.33 keseluruhan input data hewan.

The screenshot shows a software window titled 'MDIForm1' with a menu bar containing 'Materi Data', 'Pelatihan', and 'Pengujian'. Below the menu is a 'Form1' section containing a table with the following columns: 'No', 'Nama', 'LEVEL ORGANISASI', 'SIMETRI TUBUH FASE D', 'PENCERNAAN MAKANAN', 'PEREDARAN DARAH', and 'SIS'. The table has one row with '1' in the 'No' column. Below the table are input fields for 'No' (containing '1') and 'Nama Hewan' (containing 'HARIMAU'). A dropdown menu for 'LEVEL ORGANISASI' is open, showing options: 'LEVEL ORGANISASI', 'SIMETRI TUBUH FASE DEWASA', 'PENCERNAAN MAKANAN', 'PEREDARAN DARAH', 'SISTEM SARAF', and 'SISTEM REPRODUKSI'. The 'SISTEM ORGANISASI' option is selected. To the right of the input fields are buttons for 'Insert Data', 'Add Data', 'Delete Data', and 'Save Data'.

Gambar 4.24 Memasukkan Nama Hewan dan Parameter ke 1

The screenshot shows the same software window 'MDIForm1'. The table now has two rows: the first row with '1' in the 'No' column and 'HARIMAU' in the 'Nama' column; the second row with '1' in the 'No' column and 'RANGE' in the 'Nama' column. The 'LEVEL ORGANISASI' dropdown menu is open, showing options: 'LEVEL ORGANISASI', 'SIMETRI TUBUH FASE DEWASA', 'PENCERNAAN MAKANAN', 'PEREDARAN DARAH', 'SISTEM SARAF', and 'SISTEM REPRODUKSI'. The 'RANGE' option is selected. The 'Insert Data' button is circled in red. Other buttons 'Add Data', 'Delete Data', and 'Save Data' are also visible.

Gambar 4.25 Memasukkan Range ke 1

MODForm1

Master Data Pelatihan Pengujian

Form1

No	Nama	LEVEL ORGANISASI	SIMETRI TUBUH (ASLE)	PENYERAPAN MAKANAN	PEREDARAN DARAH
1	HARIMAU	SISTEM ORGAN			

No: 1

Nama Hewan: HARIMAU

LEVEL ORGANISASI: SISTEM ORGAN

Insert Data Add Data

Delete Data Save Data

Gambar 4.26 Hasil Memasukkan Nama Hewan dan Parameter ke 1 beserta Rangnya

MODForm1

Master Data Pelatihan Pengujian

Form1

No	Nama	LEVEL ORGANISASI	SIMETRI TUBUH (ASLE)	PENYERAPAN MAKANAN	PEREDARAN DARAH
1	HARIMAU	SISTEM ORGAN			
2	HARIMAU	SISTEM ORGAN			

No:

Nama Hewan: HARIMAU

LEVEL ORGANISASI: SISTEM ORGAN

Insert Data Add Data

Delete Data Save Data

- LEVEL ORGANISASI
- SISTEM ORGAN
- SISTEM SARAF
- SISTEM REPRODUKSI

Gambar 4.27 Memasukkan Parameter ke 2

MDIForm

Master Data Pelatihan Pengujian

Form1

No	Name	LEVEL ORGANISASI	SIMETRI TUBUH FASE 0	PENCERNAN MAKANAN	PEREDARAN DARAH	SIS
1	HARIMAU	SISTEM ORGAN	SIMETRI TUBUH FASE 0	PENCERNAN MAKANAN	PEREDARAN DARAH	SIS

No:

Nama Hewan:

SIMETRI TUBUH FASE DEWARA:

Insert Data Add Data

Delete Data Save Data

Gambar 4.28 Memasukkan Range ke 2

MDIForm

Master Data Pelatihan Pengujian

Form1

No	Name	LEVEL ORGANISASI	SIMETRI TUBUH FASE 0	PENCEPUSAN MAKANAN	PEREDARAN DARAH	SIS
1	HARIMAU	SISTEM ORGAN	BILATERAL	PENCEPUSAN MAKANAN	PEREDARAN DARAH	SIS

No:

Nama Hewan:

SIMETRI TUBUH FASE DEWARA:

Insert Data Add Data

Delete Data Save Data

Gambar 4.29 Hasil Memasukkan Nama Hewan dan Parameter ke 2 beserta Rangnya

MDCForm1  
Master Data Pelatihan Pengujian

M - Form1

No	BULU	SIK	GANGGANG LATERAL	CARAPACE	TYPE
1	TIDAK ADA	TIDAK ADA	TIDAK ADA	TIDAK ADA	MAMMALIA

No:

Nama Hewan:

TYPE:

Buttons: Insert Data, Add Data, Delete Data, **Save Data**

Gambar 4.30 Tombol Menyimpan Data

MDCForm1  
Master Data Pelatihan Pengujian

M - Form1

No	Nama	LEVEL ORGANISASI	SIMETRI TUBUH FASE D	PENCERNAAN MAMMANA	PEREDARAN DARAH	SP
1	HARIMAU	SISTEM ORGAN	BILATERAL	SALURAN PENCERNAAN TERBUKA		D.T
2						

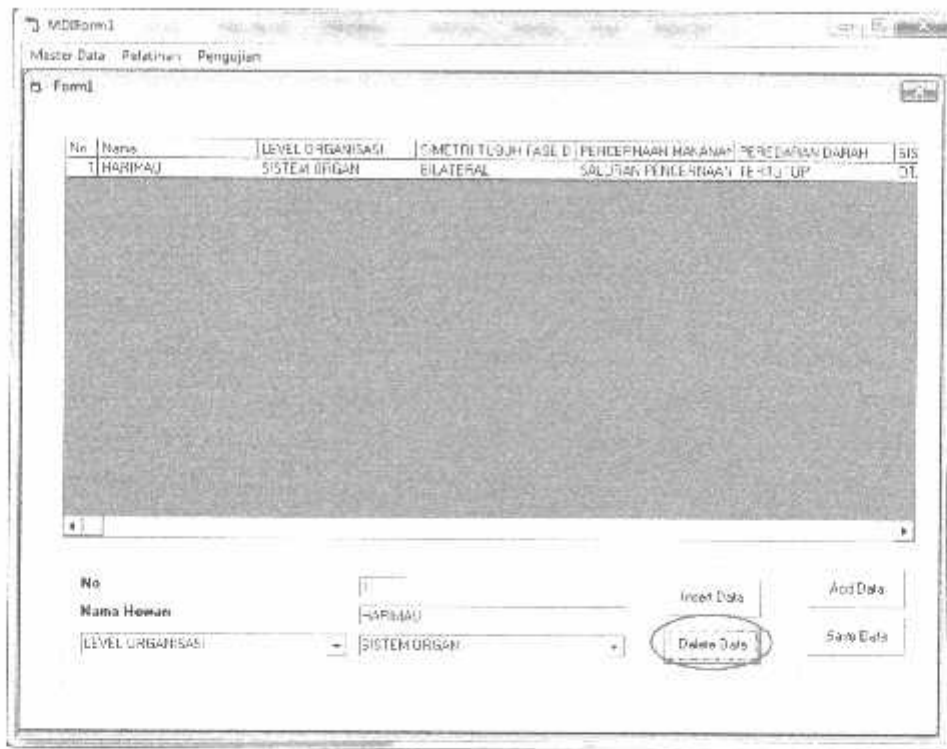
No:

Nama Hewan:

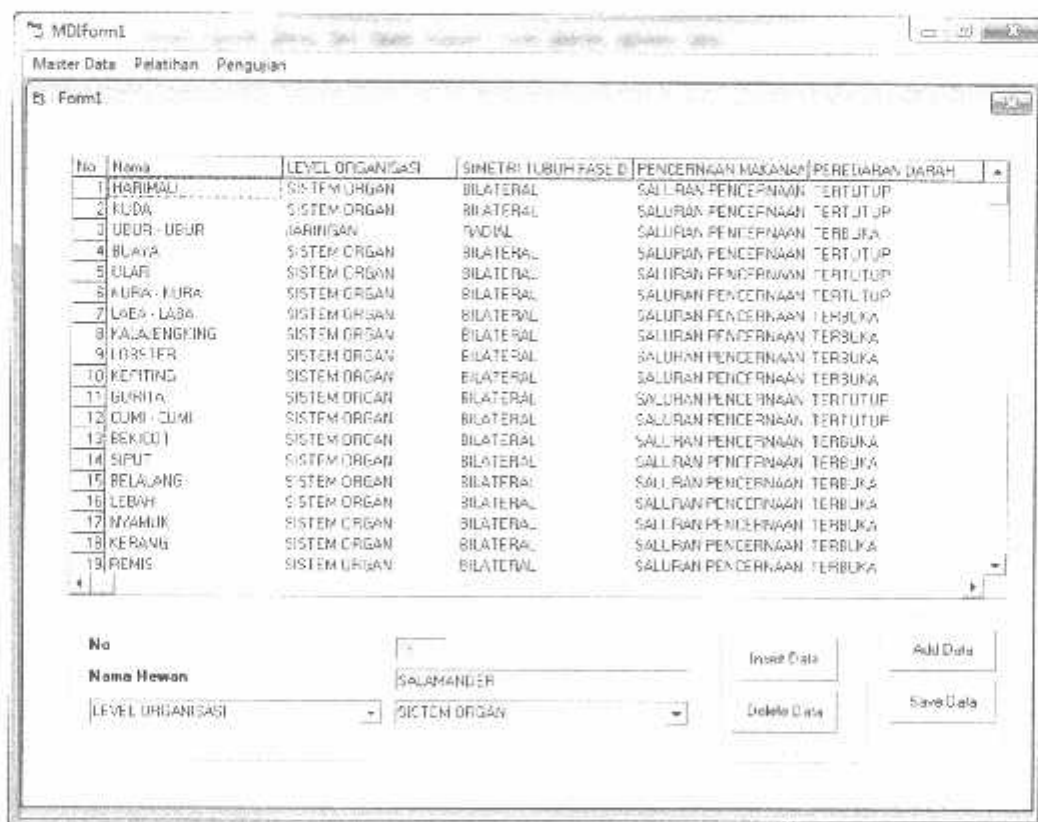
LEVEL ORGANISASI:

Buttons: Insert Data, **Add Data**, Delete Data, Save Data

Gambar 4.31 Tombol Menambah Tabel



Gambar 4.32 Tombol Menghapus Tabel

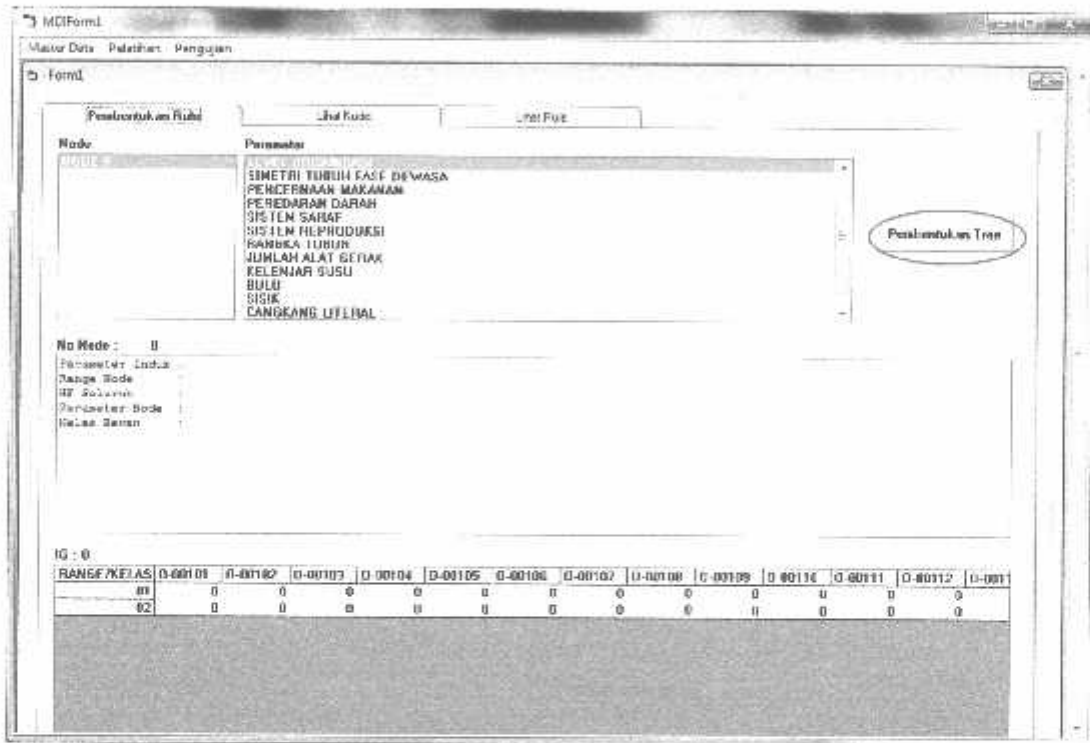


Gambar 4.33 Hasil Keseluruhan Input Data Hewan.

### 4.3 Setting Data Latih

#### 4.3.1 Pembentukan Rule

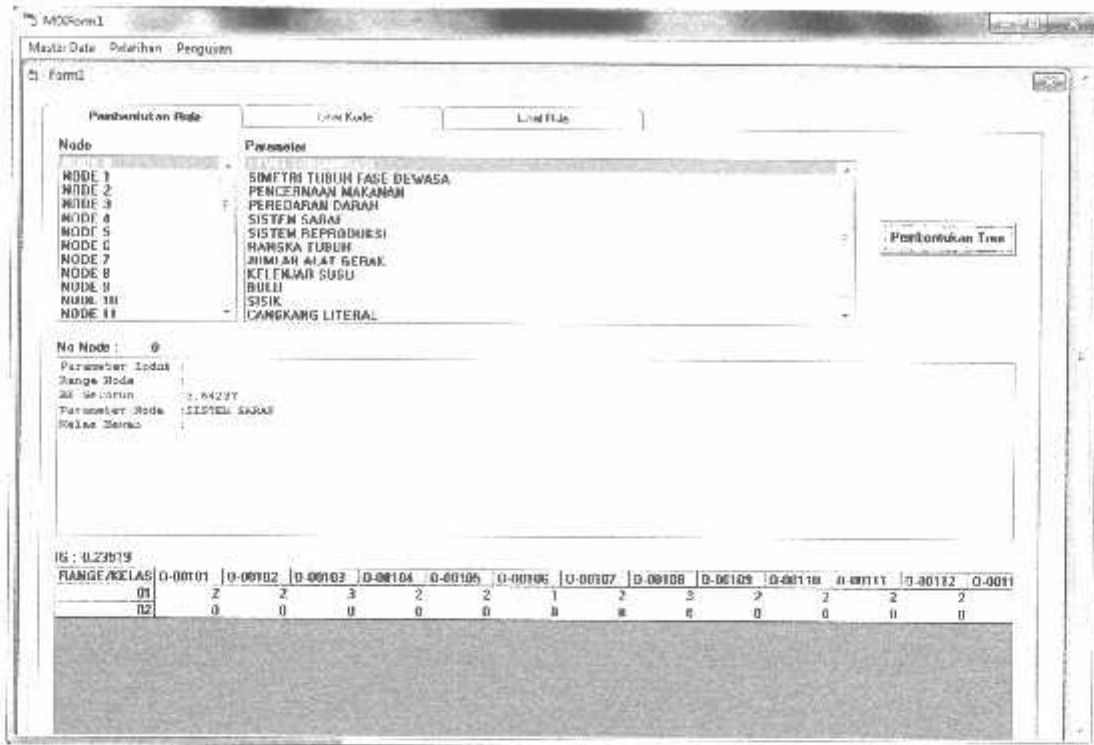
Pada pengaturan data latih ini berfungsi untuk membentuk suatu *decision tree* dengan algoritma *iterative dichotomizer 3* dan dengan menghitung *entropy* seluruh data kelas dan *entropy* setiap atribut maka proses selanjutnya mengambil *information gain* terbesar. Prosedur menjalankan data latih pilih/tekan pelatihan lalu tekan tombol pembentukan *tree* seperti pada gambar 4.34 Dan hasil dari pembentukan seperti gambar 4.35



Gambar 4.34 Pembentukan *Tree*

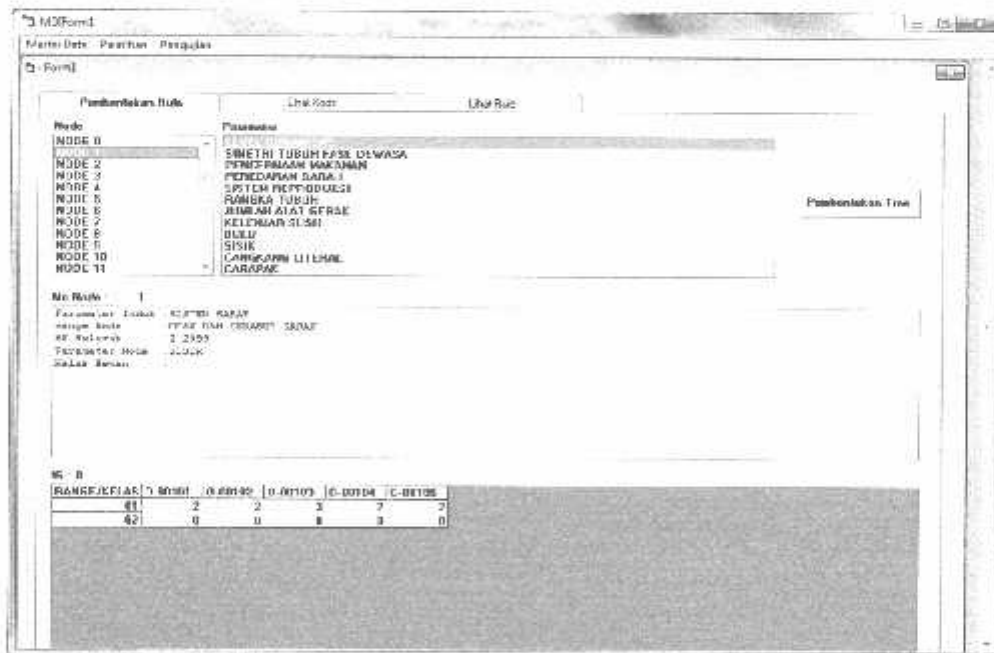


Pada gambar 4.35 Menunjukkan hasil dari pembentukan pohon pada node 0



Gambar 4.35 Pembentukan Tree Node 0

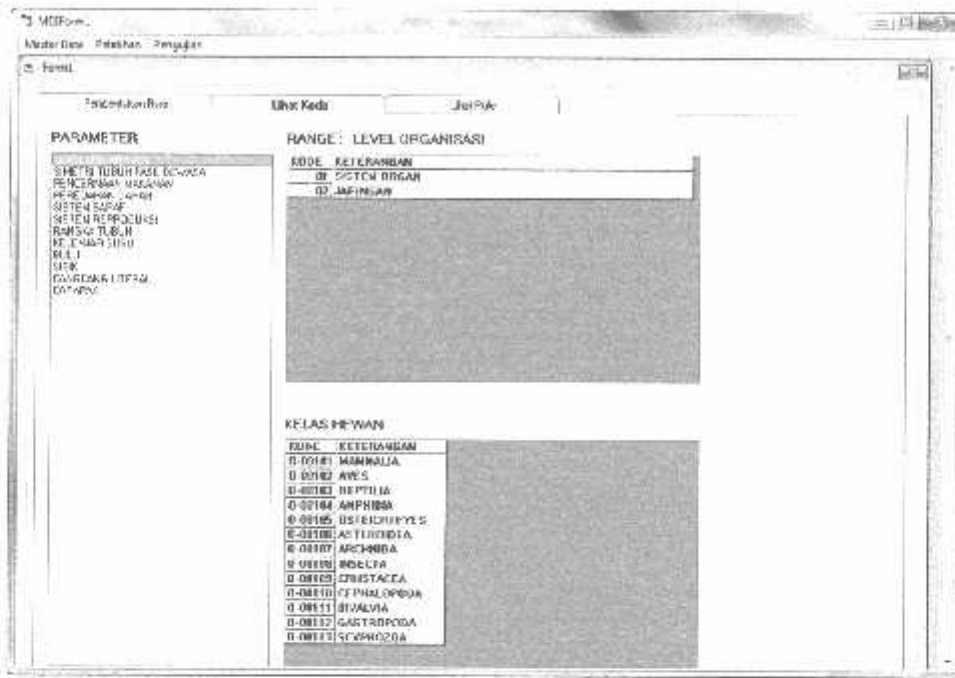
Pada gambar 4.36 Menunjukkan hasil dari pembentukan pohon pada node 1



Gambar 4.36 Pembentukan Tree Node 1

### 4.3.2 Lihat Kode

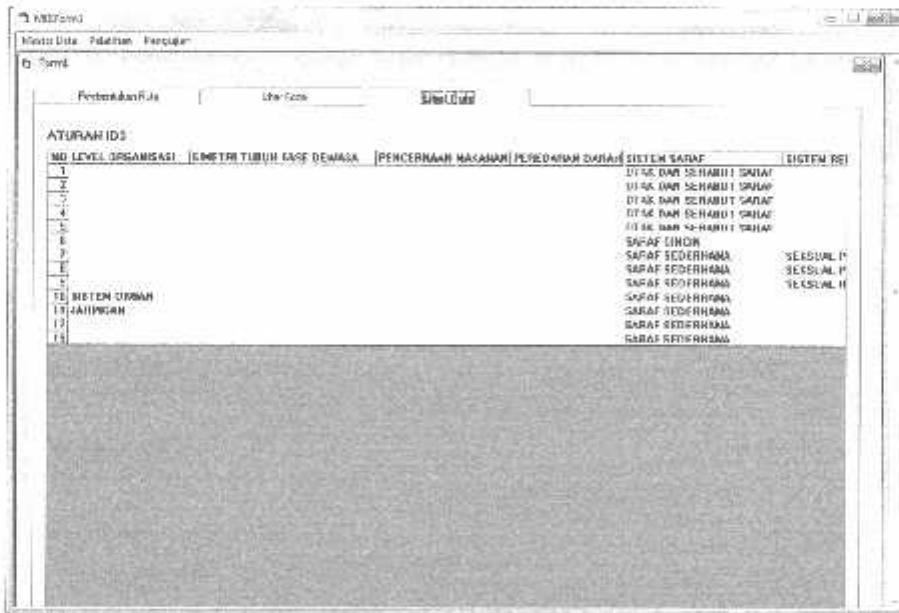
Pada gambar 4.37 menunjukkan kode range yang berfungsi untuk melihat range/kelas.



Gambar 4.37 Lihat Kode

### 4.3.3 Lihat Rule

Pada gambar 4.38 menunjukkan aturan yang terbentuk didalam pohon *tree*, dimana simetri tubuh, pencernaan makanan dan peredaran darah tidak terlalu diperhitungkan.



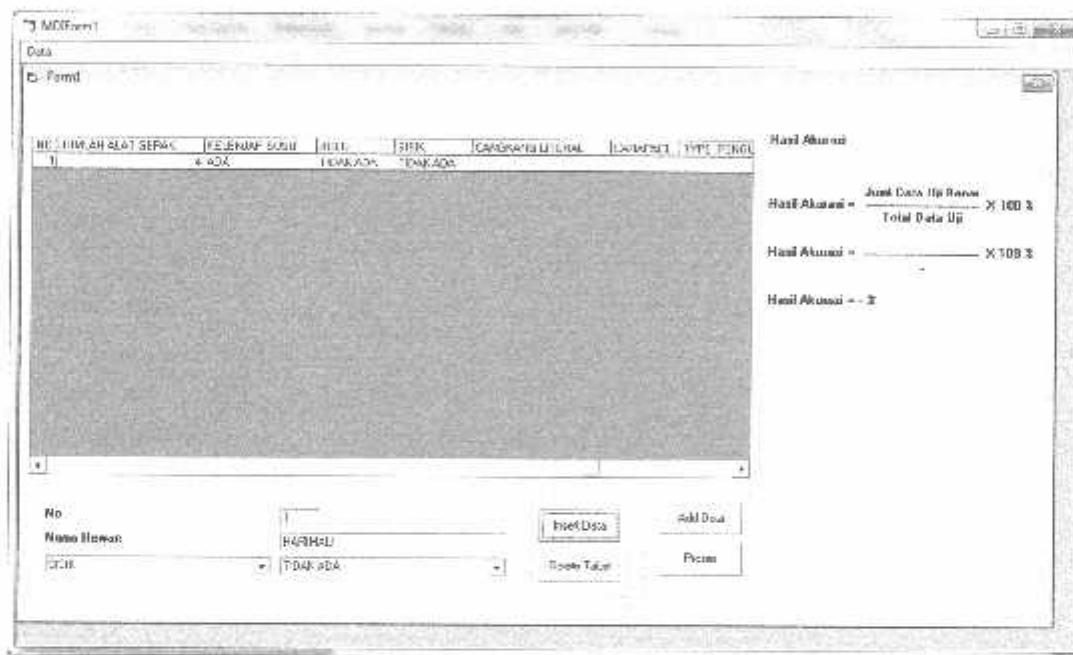
Gambar 4.38 Lihat Rule

#### 4.4 Pengujian Keseluruhan

Pengujian keseluruhan ini bertujuan untuk mencocokkan output data antara data latih dengan data uji memiliki kesamaan data. Jika memiliki outputan data yang sama antara data latih dan data uji maka pengujian tersebut berhasil.

Procedure pengujian keseluruhan:

1. Memasukkan nama hewan.
2. Memasukkan parameter I/O.
3. Memasukkan range.
4. Tekan tombol insert data untuk menambah data dalam parameter hewan.
5. Menekan tombol proses.



Gambar 4.39 Display Data Uji

Setelah tombol proses ditekan maka akan muncul persentase keberhasilan sebesar 100% sehingga bisa dikatakan berhasil.

The screenshot shows a software window titled "MDForm1" with a "Data" section. It contains a table with the following columns: NO, BULLI, S.SB, CANGKANG, TEPAL, CARAFACE, TYPE, PENGUNAN, and JRSI. The first row of data is: 1, TCAKADA, TCAKADA, TCAKADA, TCAKADA, MAMMAL, 0, and 0. To the right of the table, there are calculations for "Hasil Akurasi":

$$\text{Hasil Akurasi} = \frac{\text{Jumlah Data Uji Benar}}{\text{Total Data Uji}} \times 100 \%$$

$$\text{Hasil Akurasi} = \frac{1}{1} \times 100 \%$$

$$\text{Hasil Akurasi} = 100 \%$$

At the bottom of the window, there are input fields for "No" (value: 1), "Nama Hewan" (value: HARPAU), and "TYPE" (value: MAMMAL). There are also buttons for "Hapus Data", "Add Data", "Ukuran Tabel", and "Proses".

Gambar 4.40 Display Setelah Proses

Apabila salah dalam memasukkan nama dan data hewan maka sangat berpengaruh terhadap output data beserta persentase akurasi.

This screenshot shows the same software window "MDForm1" but with incorrect data entered. The table now shows: 1, TCAKADA, ADA, TCAKADA, TCAKADA, TCAKADA, REPTILIA, and 0. The calculation results are:

$$\text{Hasil Akurasi} = \frac{\text{Jumlah Data Uji Benar}}{\text{Total Data Uji}} \times 100 \%$$

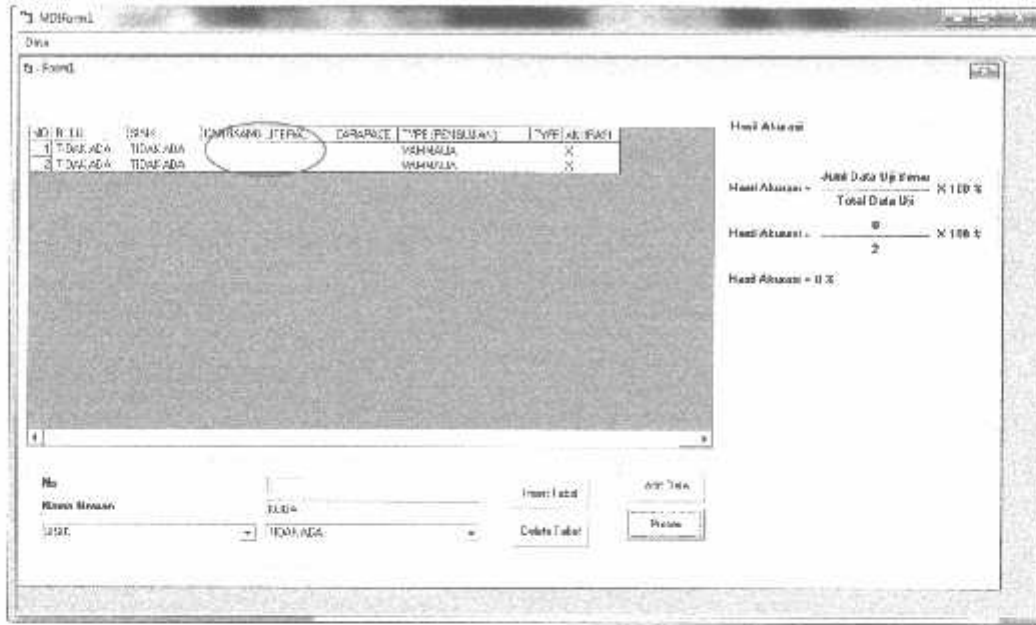
$$\text{Hasil Akurasi} = \frac{0}{1} \times 100 \%$$

$$\text{Hasil Akurasi} = 0 \%$$

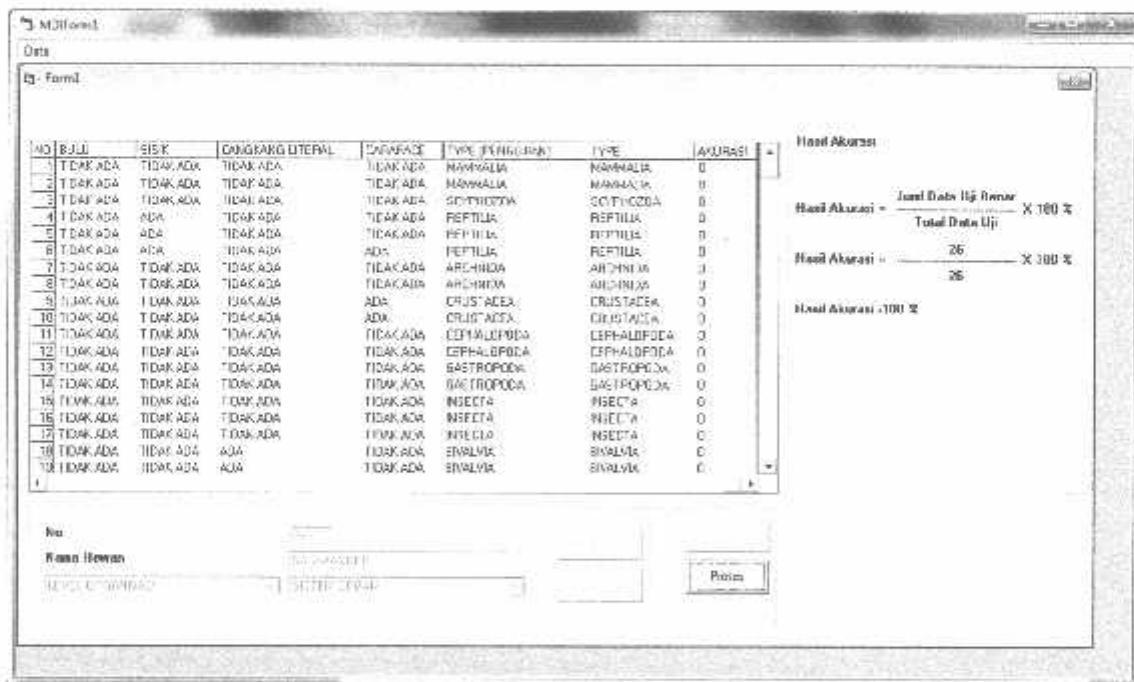
The input fields at the bottom now show "Nama Hewan" as HARPAU and "TYPE" as REPTILIA. The "Proses" button is highlighted.

Gambar 4.41 Display Salah Memasukkan Data

Pada gambar dibawah ini diketahui hasil dari proses kurang dalam memasukkan data hewan maka juga akan berpengaruh pada output data dan akurasi.

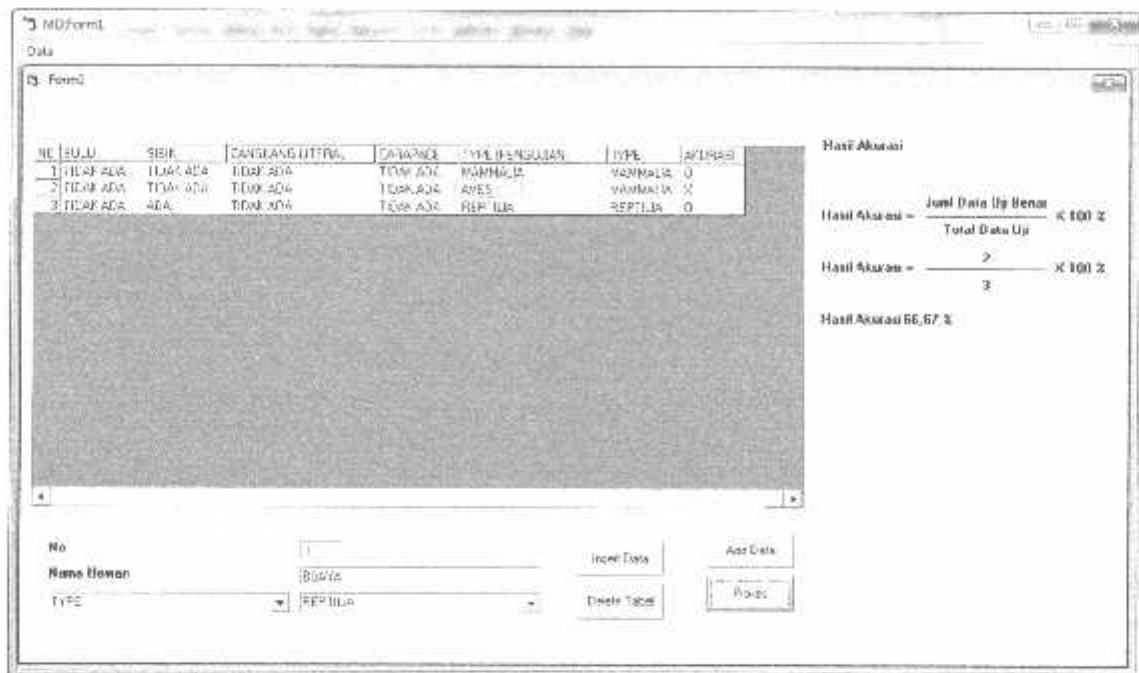


Gambar 4.42 Display Kurang Memasukkan Data



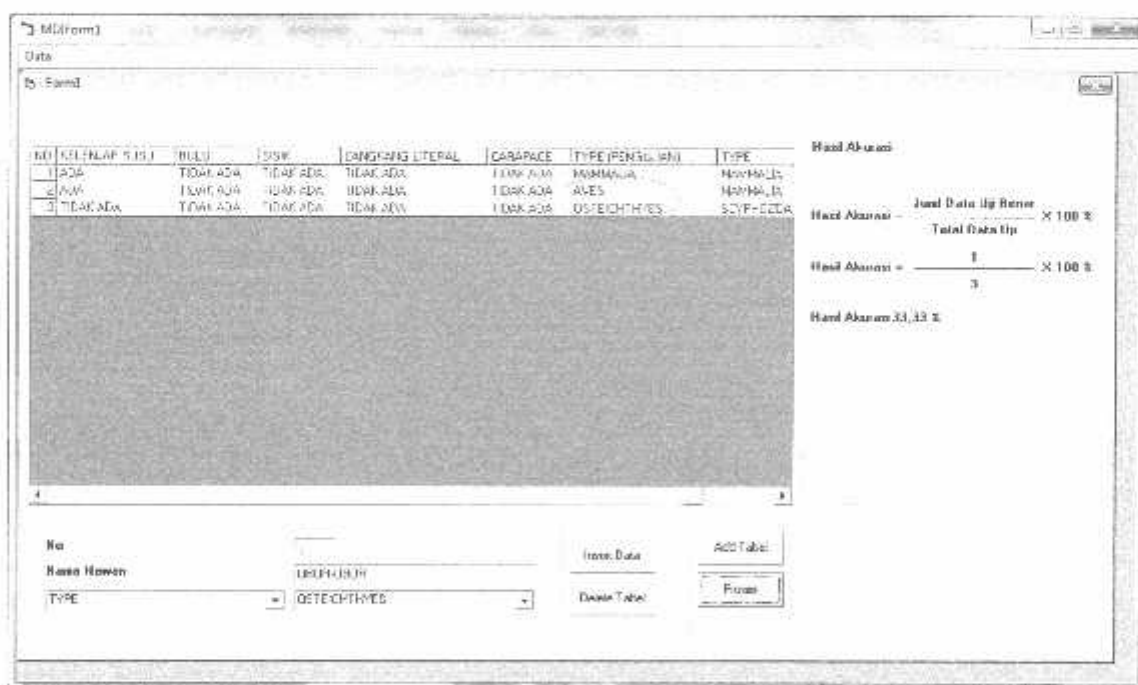
Gambar 4.43 Display Seluruh Data Latih Diujikan

Pada gambar dibawah data latih dimasukkan sebanyak 3 hewan dan user sengaja memasukkan nilai salah pada *type* kuda untuk mengetahui hasil akurasi.



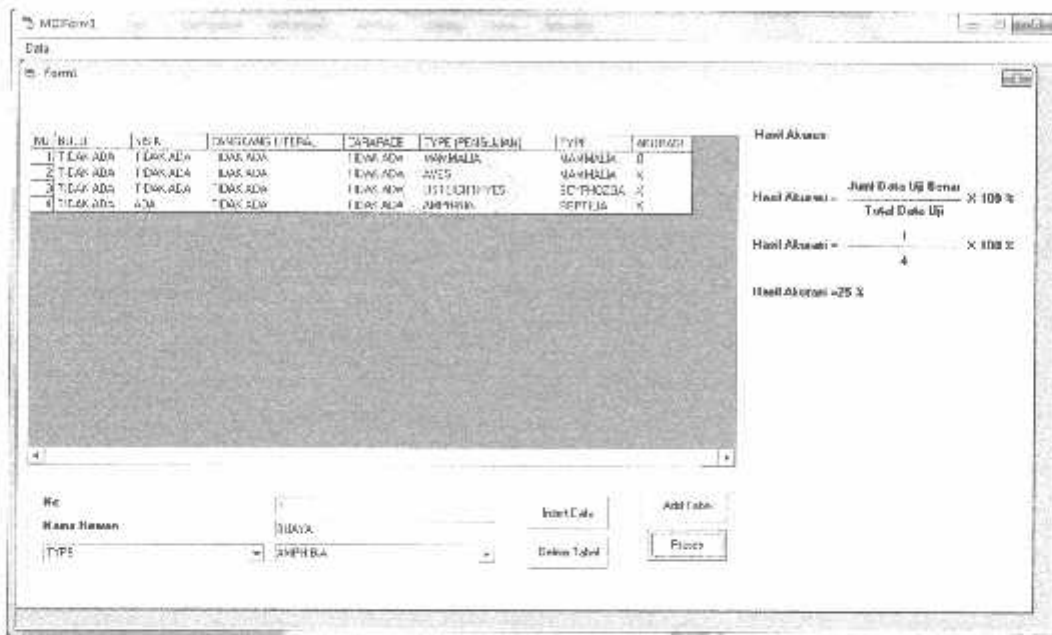
Gambar 4.44 Display Menguji Nilai Akurasi Data latih.

Pada gambar dibawah data latih dimasukkan sebanyak 3 hewan dan *user* sengaja memasukkan nilai salah pada *type* kuda dan ubur-ubur untuk mengetahui hasil akurasi.



Gambar 4.45 Display Menguji Nilai Akurasi Data latih Dengan 2 Nilai Salah.

Pada gambar dibawah data latih dimasukkan sebanyak 4 hewan dan *user* sengaja memasukkan nilai salah pada *type* kuda, ubur-ubur dan buaya untuk mengetahui hasil akurasi.



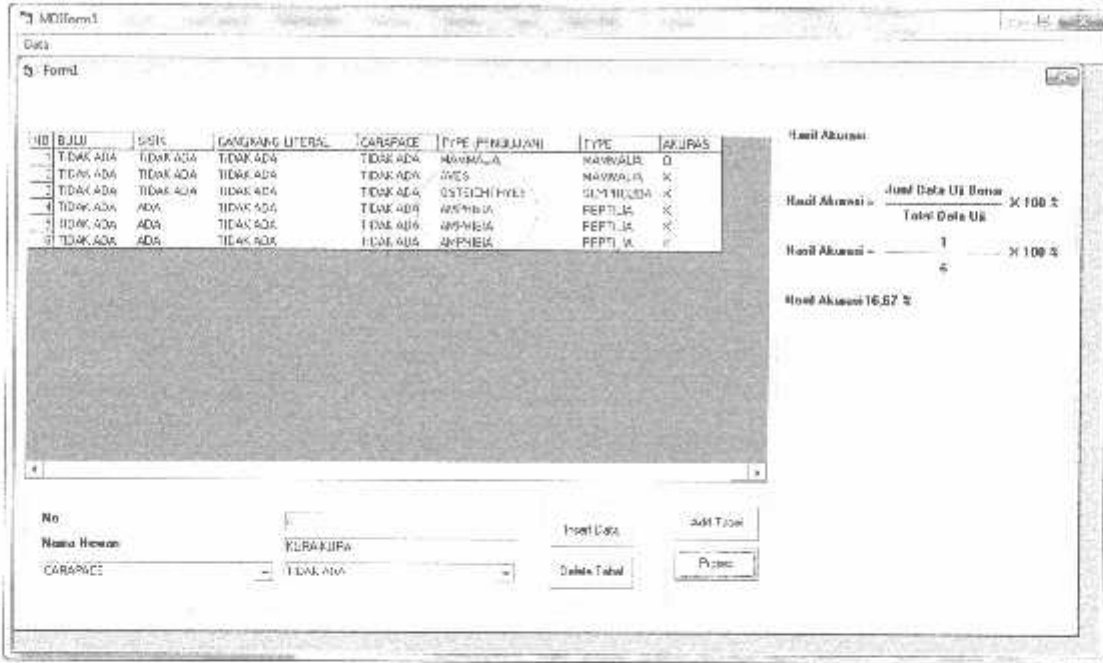
Gambar 4.46 Display Menguji Nilai Akurasi Data latih Dengan 3 Nilai Salah.

Pada gambar dibawah data latih dimasukkan sebanyak 5 hewan dan *user* sengaja memasukkan nilai salah pada *type* kuda, ubur-ubur, buaya dan komodo untuk mengetahui hasil akurasi.



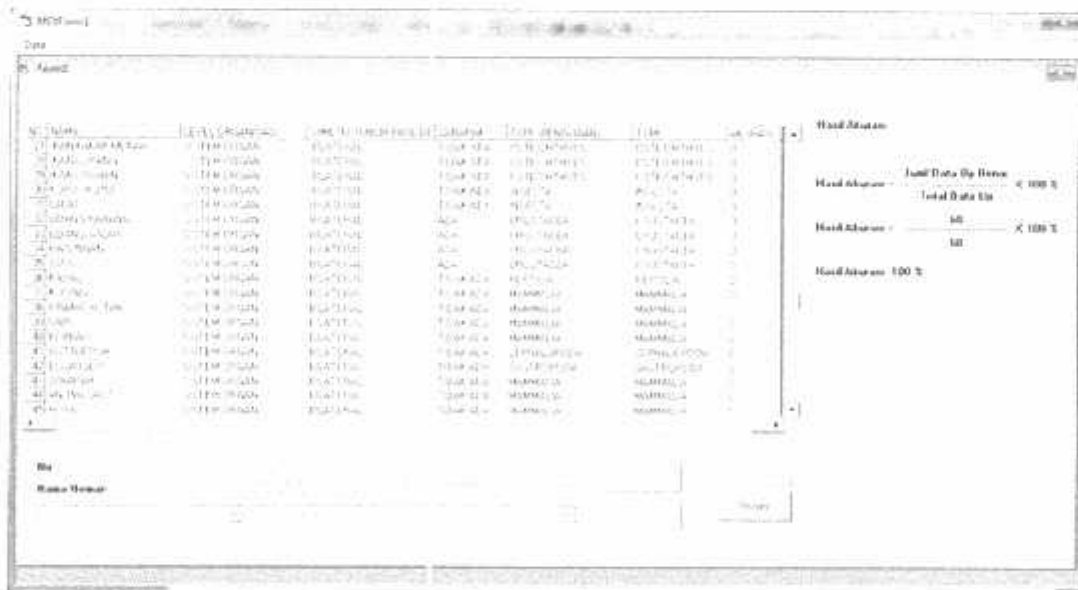
Gambar 4.47 Display Menguji Nilai Akurasi Data latih Dengan 4 Nilai Salah.

Pada gambar dibawah data latih dimasukkan sebanyak 6 hewan dan *user* sengaja memasukkan nilai salah pada *type* kuda, ubur-ubur, buaya, komodo, kura-kura untuk mengetahui hasil akurasi.



Gambar 4.48 Display Menguji Nilai Akurasi Data Latih Dengan 5 Nilai Salah.

Setelah memasukkan data keseluruhan sebanyak 60 nama hewan beserta parameternya, maka didalam gambar diketahui bahwa akurasinya 100% asalkan tidak mengalami kesalahan dalam menginputkan data hewan tersebut.



Gambar 4.49 Display Data Keseluruhan Diuji



INSTITUT TEKNOLOGI NASIONAL MALANG



MALANG

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Dari hasil analisis dan perancangan aplikasi dalam mengklasifikasi hewan serta implementasi dan pengujian yang telah dilakukan dapat diambil beberapa kesimpulan sebagai berikut :

1. Pada saat melakukan percobaan, jika salah memasukkan range maka akan mempengaruhi nilai keakurasiannya.
2. Pada percobaan pengujian memasukkan 3 data latih, user memasukkan nilai salah pada salah satu hewan, dari situ diketahui tingkat akurasi 66,66%.
3. Pada percobaan pengujian memasukkan 3 data latih, user memasukkan nilai salah pada 2 hewan, maka tingkat akurasi sebesar 33,33%.
4. Pada percobaan pengujian memasukkan 4 data latih, user memasukkan nilai salah pada 3 hewan, maka tingkat akurasi sebesar 25%.
5. Pada percobaan pengujian memasukkan 5 data latih, user memasukkan nilai salah pada 4 hewan, maka tingkat akurasi sebesar 20%.
6. Pada percobaan pengujian memasukkan 6 data latih, user memasukkan nilai salah pada 5 hewan, maka tingkat akurasi sebesar 16,67%.
7. Apabila kurang memasukkan data hewan maka tingkat akurasi akan salah dalam hasil mengklasifikasi hewan.
8. Jika keterangan memasukkan type pengujian benar sesuai dengan referensi maka tingkat akurasi 100%.

#### **5.2 Saran**

Aplikasi yang telah dibuat sudah bisa bekerja dengan baik, tetapi masih terdapat beberapa hal yang penulis sarankan untuk perbaikan dan kesempurnaan aplikasi yang telah dibuat. Adapun beberapa hal yang penulis sarankan adalah :

1. Dalam melakukan proses input data range/outcome, sebaiknya dilakukan dengan cermat agar tidak terjadi kesalahan pada proses – proses berikutnya.

2. Penyempurnaan dalam menambahkan beberapa parameter.
  3. Tampilan aplikasi sebaiknya lebih disempurnakan, dengan memberi gambar hewan agar lebih komunikatif dan menarik.
-

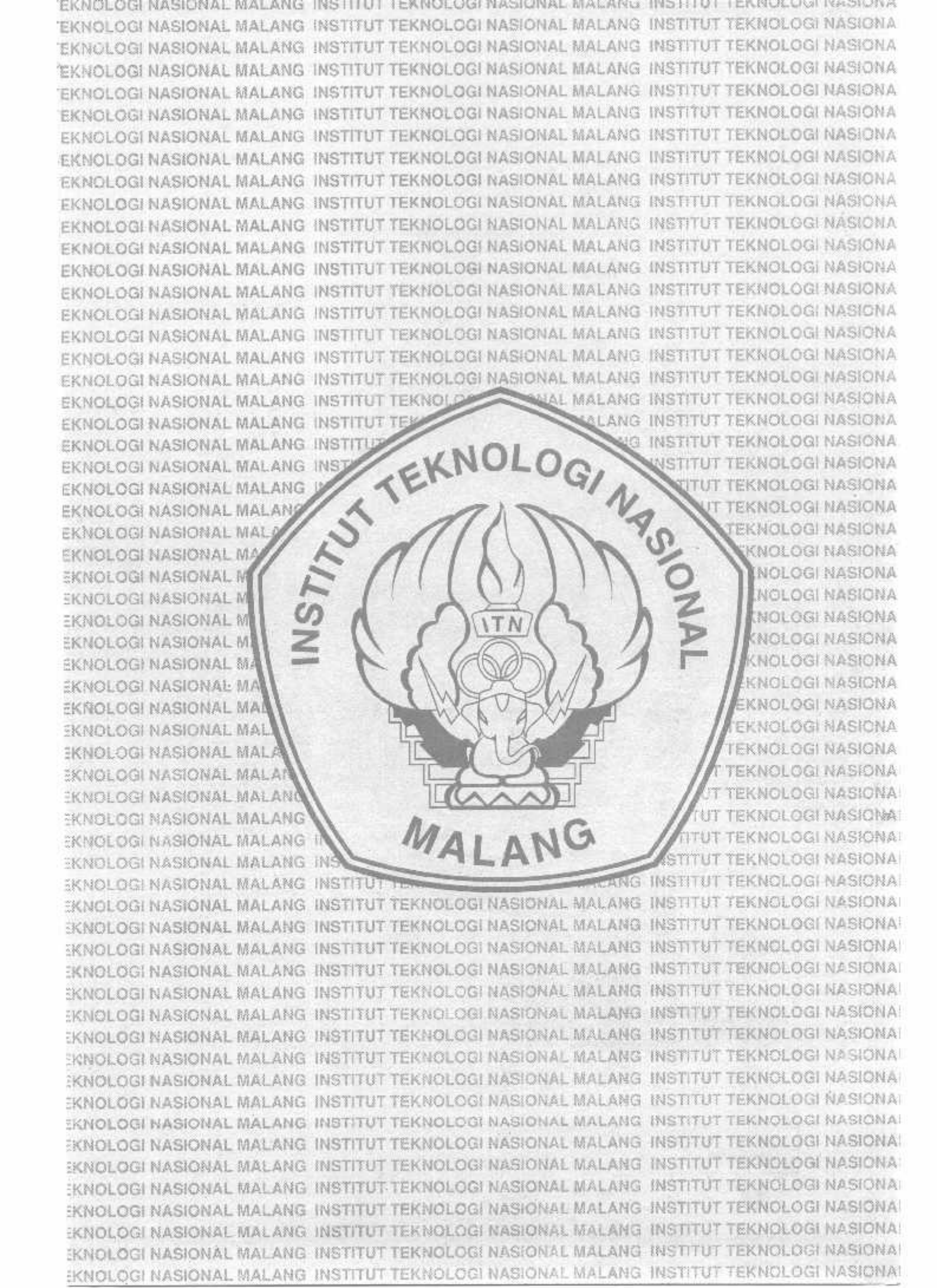




## DAFTAR PUSTAKA

- [1] Andi, 2006, Visual Basic 6.0, Andi dengan Madcoms.
- [2] Amirudin, 2007, Penerapan Association Rule mining pada Data Nomor Unik Pendidik dan Tenaga Kependidikan untuk menemukan pola Sertifikasi guru, Surabaya.
- [3]. Ayub, M. (2007). Proses Data Mining Dalam Sistem Pembelajaran Berbantuan Kompter. Bandung.
- [4] Dra. D.A. Pratiwi, Dra. Sri Maryati, Dra. Srikini, Drs. Soeharno, Drs Bambang. S, Biologi Untuk Sma Kelas X, 2006, Erlangga.
- [5] Linoff, M. J. (2004). *Data Mining Technuques For Marketing Sales*. Canada: Wiley Publishing.
- [6] Nabila, C. N. (2010). Mahkluk Hidup Dalam Pohon. Bandung: Institut Teknologi Bandung.
- [7] Pramudiono, I. 2007. *Pengantar Data Mining : Menambang Permata Pengetahuan di Gunung Data*. <http://www.ilmukomputer.org/wp-content/uploads/2006/08/iko-datamining.zip>, Diakses pada tanggal 6 April 2012.
- [8] Safii, M. (2011). Penetapan Strategi Pemasaran Dengan Pendekatan Model Aturan Pohon Keputusan Menggunakan Algoritma Iterative Dichotomizer 3 (ID3) Di Perguruan Tinggi Swasta. Medan: Unifersitas Sumatra Utara.
- [9]. Sunjana. (2010). Aplikasi Data Mining Mahasiswa Dengan Metode Klasifikasi *Decision Tree*. Yogyakarta: Universitas Widyatama.
- [10]. Wahyudin. (2009). *Metode Iterative Dichotomizer 3 (ID3) Untuk Penyeleksian Mahasiswa Baru*. Universitas Pendidikan Indonesia.
- [11]. Zalilia, L. (2007). Penerapan Data Mining untuk *IDS*. Bandung: Institut Teknologi Bandung.
- [12]. <http://www.scribd.com/doc/33807043/25/BPENGELOMPOKAN-HEWAN>, Diakses pada tanggal 2Mei 2012.
- [13]. [http://id.wikipedia.org/wiki/Klasifikasi\\_ilmiah](http://id.wikipedia.org/wiki/Klasifikasi_ilmiah), Diakses pada tanggal 3 April 2012.

- [14]. <http://www.scribd.com/doc/87437950/Bab-v-Pencernaan-Makanan-2>, diakses pada tanggal 2 Mei 2012.
  - [15]. [http://id.wikipedia.org/wiki/Sistem\\_organ](http://id.wikipedia.org/wiki/Sistem_organ), Diakses pada tanggal 5 April 2012.
  - [16]. <http://id.wikipedia.org/wiki/Jaringan>, Diakses pada tanggal 5 April 2012.
-



# Lampiran-Lampiran

---





**BERITA ACARA UJIAN SKRIPSI  
FAKULTAS TEKNOLOGI INDUSTRI**

NAMA : IRFAN ARI KRISANTO  
NIM : 07.12.502  
JURUSAN : Teknik Elektro S-1  
KONSENTRASI : Teknik Komputer & Informatika  
MASA BIMBINGAN: Semester Genap  
JUDUL : **RANCANG BANGUN APLIKASI UNTUK  
MENGKLASIFIKASI HEWAN MENGGUNAKAN METODE  
DECISION TREE DENGAN ALGORITMA ITERATIVE  
DICHOTOMIZER 3 BERBASIS VISUAL BASIC**

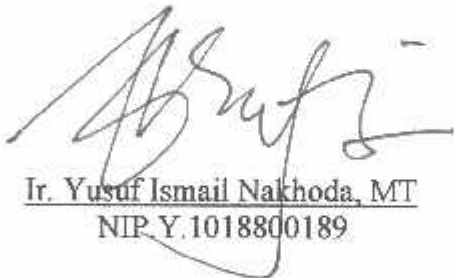
Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :

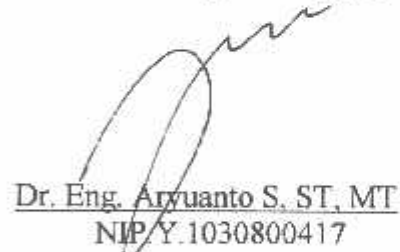
Hari : Selasa  
Tanggal : 7 Agustus 2012  
Dengan Nilai : (77,5) B+  $\checkmark$

**PANITIA UJIAN SKRIPSI**

Ketua Majelis Penguji,

Sekretaris Majelis Penguji,

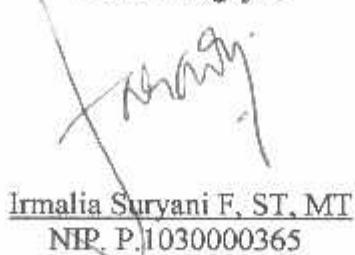
  
Ir. Yusuf Ismail Nakhoda, MT  
NIP. Y.1018800189

  
Dr. Eng. Arjuanto S., ST, MT  
NIP. Y.1030800417

**ANGGOTA PENGUJI**

Dosen Penguji I

Dosen Penguji II

  
Irmalia Suryani F., ST, MT  
NIP. P.1030000365

  
Bima Aulia Firmandani, ST  
1121



## FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan Ujian Skripsi Jenjang Strata I Jurusan Teknik Elektro Konsentrasi Teknik Komputer & Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : Irfan Ari Krisanto  
Nim : 07.12.502  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer & Informatika  
Masa Bimbingan : Semester Genap  
Judul Skripsi : **Rancang Bangun Aplikasi Untuk Mengklasifikasi Hewan Menggunakan Metode Decision Tree Dengan Algoritma Iterative Dichotomizer 3 Berbasis Visual Basic**

No	Penguji	Tanggal	Uraian	Paraf
1.	Penguji I	07 Agustus 2012	<ul style="list-style-type: none"><li>• Hasil Akurasi</li><li>• Kesimpulan</li></ul>	
2	Penguji II	07 Agustus 2012	<ul style="list-style-type: none"><li>• Hasil Akurasi</li><li>• Kesimpulan</li></ul>	

Disetujui:

Penguji I

Irmalia Suryani E, ST, MT  
NIP.P.1030000365

Penguji II

Bima Aulia Firmandani, ST  
1121

Mengetahui:

Dosen Pembimbing I

Ir. Eko Nurcahyo, MT  
NIP. Y. 1028700172

Dosen Pembimbing II

Ahmad Faisol, ST  
NIP. Y. 1031000431



## FORMULIR BIMBINGAN SKRIPSI

Nama : **IRFAN ARY KRISANTO**  
Nim : **07.12.502**  
Masa Bimbingan : **SEMESTER GENAP TAHUN AKADEMIK 2011-2012**  
Judul Skripsi : **RANCANG BANGUN APLIKASI UNTUK MENGLASIFIKASI HEWAN  
MENGUNAKAN METODE *DECISION TREE* DENGAN ALGORITMA  
*ITERATIVE DICHOTOMIZER 3* BERBASIS *VISUAL BASIC***

No	Tanggal	Uraian	Paraf Pembimbing
1	4/6 2012	Ace BAB I	
2	4/6 2012	BAB II : Tambahan Sinter.	
3	4/6 2012	Ace BAB II , BAB II : Flow Chart	
4	16/7 2012	Ace BAB III	
5	16/7 2012	Ace BAB IV , BAB V	
6		Ace Seminar hasil	
7		Revisi	
8		Ace mapre ujian Skripsi	
9			
10			

Malang ,  
Dosen Pembimbing I

**Ir. Eko Nurcahyo, MT**  
NIP. 1028700172



## FORMULIR BIMBINGAN SKRIPSI

Nama : **IRFAN ARY KRISANTO**  
Nim : **07.12.502**  
Masa Bimbingan : **SEMESTER GENAP TAHUN AKADEMIK 2011-2012**  
Judul Skripsi : **RANCANG BANGUN APLIKASI UNTUK MENGLASIFIKASI HEWAN  
MENGUNAKAN METODE *DECISION TREE* DENGAN ALGORITMA  
*ITERATIVE DICHOTOMIZER 3 BERRASIN VISUAL BASIC***

No	Tanggal	Uraian	Paraf Pembimbing
1	3/7 <sup>12</sup>	Acc Bab I	
2	3/7 <sup>12</sup>	Acc Bab II	
3	3/7 <sup>12</sup>	Acc Bab III	
4		Revisi Bab IV: Perbaikan implementasi dan pengujian	
5		Demo program	
6		Acc Bab IV	
7		Acc Bab V	
8			
9			
10			

Malang ,

Dosen Pembimbing II

**Ahmad Faisal/ST**  
**NIP.P.1010100358**

Form S-4B

```
FmenuUtama:  
Private Sub Master_Click(Index As Integer)  
    Select Case Index  
        Case 0: FParameter.Show  
        Case 1: FRange.Show  
        Case 2: FHewan.Show  
    End Select  
End Sub
```

```
Private Sub Menu_Click(Index As Integer)  
    Select Case Index  
        Case 1: FProses.Show  
        Case 2: FPengujian.Show  
    End Select  
End Sub
```

```
FHewan:  
Option Explicit  
Private Type RParameter  
    ID_Parameter As String  
    Nama_Parameter As String  
    Range_Flexibel As Boolean  
    Keterangan_IO As Byte
```

```
End Type  
Private Type RRange  
    Id_Range As String  
    ID_Parameter As String  
    Keterangan As String
```

```
End Type  
Private Type RHewan  
    ID_Hewan As String  
    Nama_Hewan As String  
    Id_Range As String
```

```
End Type  
Private Type RDataParameter  
    ID_Data_Parameter As String  
    ID_Hewan As String  
    ID_Parameter As String  
    Id_Range As String
```

```
End Type  
Dim Parameter As RParameter  
Dim Hewan As RHewan  
Dim DataParameter As RDataParameter  
Dim Range As RRange  
Dim No As Integer
```

```
Private Sub Atur_Tabel()  
    Dim Sqlcmd As String  
    Dim i As Integer, j As Integer  
    With Tabel_Data
```

```

.Rows = 2
.Cols = 2
.Clear
Sqlcmd = "Select Count(ID_Parameter) From Parameter "
Set Myrs = Mydb.Execute(Sqlcmd)
If Not (Myrs.BOF And Myrs.EOF) Then
    .Cols = .Cols + Myrs.Fields(0)
End If
.TextMatrix(0, 0) = "No"
.TextMatrix(0, 1) = "Nama"
.ColWidth(0) = 400
For i = 1 To .Cols - 1
    .ColWidth(i) = 2000
Next
i = 1
CParameter.Clear
Sqlcmd = "Select Top 1 * From Parameter order by ID_Parameter "
Set Myrs = Mydb.Execute(Sqlcmd)
Do While Not (Myrs.EOF)
    Parameter.ID_Parameter = Myrs!ID_Parameter
    i = i + 1
    Parameter>Nama_Parameter = Myrs!Nama_Parameter
    Tabel_Data.TextMatrix(0, i) = Parameter>Nama_Parameter
    CParameter.AddItem (Parameter>Nama_Parameter)
    Sqlcmd = "Select Top 1 * From Parameter Where ID_Parameter > " &
Parameter.ID_Parameter & " order by ID_Parameter "
    Set Myrs = Mydb.Execute(Sqlcmd)
Loop
If i > 1 Then
    CParameter.ListIndex = 0
    Parameter.ID_Parameter = Get_IDParameter(CParameter,Text)
    Call Isi_Combo_Range(Parameter.ID_Parameter)
    If CRange.ListCount > 0 Then
        CRange.ListIndex = 0
    End If
End If
End With
End Sub
Private Sub Setting_Awal()
    Atur_Tabel
    No = 1
    Tabel_Data.TextMatrix(No, 0) = No
    TxtNo.Text = No
    TxtNama.Text = ""
    TxtNo.Enabled = False
    Load_DataHewan
End Sub
Private Sub Isi_Combo_Range(ID_Parameter1 As String)
Dim Sqlcmd As String
    CRange.Clear

```

---

```

    Sqlcmd = "Select Top 1 * From Range Where ID_Parameter = " & ID_Parameter1
    & " And Range_Tambahan=false order by ID_Range "
    Set Myrs = Mydb.Execute(Sqlcmd)
    Do While Not (Myrs.EOF)
        Range.Id_Range = Myrs!Id_Range
        Range.Keterangan = Myrs!Keterangan
        CRange.AddItem (Range.Keterangan)
        Sqlcmd = "Select Top 1 * From Range Where ID_Parameter = " &
        ID_Parameter1 & " And Range_Tambahan=false And Id_Range > " &
        Range.Id_Range & " order by ID_Range "
        Set Myrs = Mydb.Execute(Sqlcmd)
    Loop
End Sub
Private Function Get_IDParameter>Nama_Parameter1 As String) As String
Dim Sqlcmd As String
    Sqlcmd = "Select * From Parameter Where Nama_Parameter= " &
    Nama_Parameter1 & " "
    Set Myrs = Mydb.Execute(Sqlcmd)
    If Not (Myrs.BOF And Myrs.EOF) Then
        Get_IDParameter = Myrs!ID_Parameter
    Else
        Get_IDParameter = ""
    End If
End Function
Private Function Get_IDRange>ID_Parameter1 As String, Nama_Range1 As String) As
String
Dim Sqlcmd As String
    Sqlcmd = "Select * From Range Where ID_Parameter= " & ID_Parameter1 & " " &
    "And Keterangan = " & Nama_Range1 & " "
    Set Myrs = Mydb.Execute(Sqlcmd)
    If Not (Myrs.BOF And Myrs.EOF) Then
        Get_IDRange = Myrs!Id_Range
    Else
        Get_IDRange = ""
    End If
End Function
Private Sub Save_Data_Hewan>ID_Hewan1 As String, Nama_Hewan1 As String,
ID_RangeOutput1 As String)
Dim Sqlcmd As String
    Sqlcmd = "Insert into DataHewan Values(" & ID_Hewan1 & "," & Nama_Hewan1
    & "," & ID_RangeOutput1 & ")"
    Mydb.BeginTrans
    Mydb.Execute (Sqlcmd)
    Mydb.CommitTrans
End Sub
Private Sub Save_Data_Parameter>ID_Data Parameter1 As String, ID_Hewan1 As
String, Id_Range1 As String)
Dim Sqlcmd As String

```

---

```

    Sqlcmd = "Insert into DataParameter Values('" & ID_Data_Parameter1 & "','" &
ID_Hewan1 & "','" & _
    "'" & Id_Range1 & "'" ) "
    Mydb.BeginTrans
    Mydb.Execute (Sqlcmd)
    Mydb.CommitTrans
End Sub
Private Function Get_Juml_Parameter_Output() As Integer
Dim Sqlcmd As String
    Sqlcmd = "Select Count(ID_Parameter) From Parameter Where Keterangan_IO=
True "
    Set Myrs = Mydb.Execute(Sqlcmd)
    If Not (Myrs.BOF And Myrs.EOF) Then
        Get_Juml_Parameter_Output = Myrs.Fields(0)
    Else
        Get_Juml_Parameter_Output = 0
    End If
End Function
Private Sub Load_DataHewan()
Dim Sqlcmd As String
Dim i As Integer
    No = 0
    Sqlcmd = "Select Top 1 * From DataHewan order by ID_DataHewan "
    Set Myrs = Mydb.Execute(Sqlcmd)
    Do While Not (Myrs.EOF)
        Hewan.ID_Hewan = Myrs!ID_DataHewan
        Hewan.Id_Range = Myrs!Id_Range
        Hewan>Nama_Hewan = Myrs>NamaHewan
        No = No + 1
        If No > (Tabel_Data.Rows - 1) Then
            Tabel_Data.Rows = Tabel_Data.Rows + 1
        End If
        Tabel_Data.TextMatrix(No, 0) = No
        Tabel_Data.TextMatrix(No, 1) = Hewan>Nama_Hewan
        If Find_ID_Range(Hewan.Id_Range) Then
            Tabel_Data.TextMatrix(No, Tabel_Data.Cols - 1) = Range.Keterangan
        End If
        Sqlcmd = "Select Top 1 * From DataParameter Where ID_DataHewan=" &
Hewan.ID_Hewan & "' order by ID_dataParameter "
        Set Myrs = Mydb.Execute(Sqlcmd)
        i = 1
        Do While Not (Myrs.EOF)
            DataParameter.ID_Data_Parameter = Myrs!ID_DataParameter
            DataParameter.Id_Range = Myrs!Id_Range
            i = i + 1
            If Find_ID_Range(DataParameter.Id_Range) Then
                Tabel_Data.TextMatrix(No, i) = Range.Keterangan
            End If
            Sqlcmd = "Select Top 1 * From DataParameter Where ID_DataHewan=" &
Hewan.ID_Hewan & "' " & _

```

---



```

        "And ID_DataParameter > " & DataParameter.ID_Data_Parameter & "
order by ID_dataParameter "
    Set Myrs = Mydb.Execute(Sqlcmd)
    Loop
    Sqlcmd = "Select Top 1 * From DataHewan Where ID_DataHewan > " &
Hewan.ID_Hewan & " order by ID_DataHewan "
    Set Myrs = Mydb.Execute(Sqlcmd)
    Loop
    If No > 0 Then
        TxtNo.Text = Tabel_Data.TextMatrix(No, 0)
        TxtNama.Text = Tabel_Data.TextMatrix(No, 1)
        If (Tabel_Data.Col - 2 >= 0) Then
            CParameter.ListIndex = Tabel_Data.Cols - 2
        End If
        CParameter_Click
    Else
        No = 1
    End If
End Sub
Private Function Find_ID_Range(Id_Range As String) As Boolean
Dim Sqlcmd As String
    Find_ID_Range = False
    Sqlcmd = "Select * From Range Where Id_Range=" & Id_Range & " "
    Set Myrs = Mydb.Execute(Sqlcmd)
    If Not (Myrs.BOF And Myrs.EOF) Then
        Find_ID_Range = True
        Range.ID_Parameter = Myrs!ID_Parameter
        Range.Keterangan = Myrs!Keterangan
    End If
End Function

    TxtNo.Text = No
    TxtNama.Text = Tabel_Data.TextMatrix(No, 1)
    CParameter.ListIndex = 0
    CRange.ListIndex = 0
    If Tabel_Data.TextMatrix(No, 2) <> "" Then
        CRange.Text = Tabel_Data.TextMatrix(No, 2)
    End If
End Sub

Private Sub CmdInsertTabel_Click()
Dim i As Integer
    i = CInt(TxtNo.Text)
    Tabel_Data.TextMatrix(i, 1) = TxtNama.Text
    Tabel_Data.TextMatrix(i, 2 + CParameter.ListIndex) = CRange.Text
End Sub

Private Sub CmdSaveData_Click()
Dim i As Integer, j As Integer, batas As Integer
Dim Sqlcmd As String

```

---

```

Dim Valid_data As Boolean
Valid_data = True
For i = 1 To Tabel_Data.Cols - 1
    For j = 1 To Tabel_Data.Rows - 1
        If Tabel_Data.TextMatrix(j, i) = "" Then
            Valid_data = False
            Exit For
        End If
    Next j
    If Not Valid_data Then Exit For
Next i
If Valid_data Then
    Mydb.BeginTrans
    For i = 1 To Tabel_Data.Rows - 1
        If Tabel_Data.Cols > 2 Then
            Hewan.ID_Hewan = "H-" + Format(i, "0000")
            Hewan>Nama_Hewan = Tabel_Data.TextMatrix(i, 1)
            If Get_Juml_Parameter_Output > 0 Then
                batas = Tabel_Data.Cols - 2
                Parameter.ID_Parameter = Get_IDParameter(Tabel_Data.TextMatrix(0,
Tabel_Data.Cols - 1))
                Hewan.Id_Range = Get_IDRange(Parameter.ID_Parameter,
Tabel_Data.TextMatrix(i, Tabel_Data.Cols - 1))
            Else
                Hewan.Id_Range = ""
                batas = Tabel_Data.Cols - 1
            End If
            Call Save_Data_Hewan(Hewan.ID_Hewan, Hewan>Nama_Hewan,
Hewan.Id_Range)
            For j = 2 To batas
                DataParameter.ID_Hewan = Hewan.ID_Hewan
                DataParameter.ID_Parameter
                DataParameter.Id_Range = Get_IDRange(DataParameter.ID_Parameter,
Tabel_Data.TextMatrix(i, j))
                Call Save_Data_Parameter(DataParameter.ID_Data_Parameter,
DataParameter.ID_Hewan, DataParameter.Id_Range)
            Next j
        End If
    Next i
    Setting_Awal
Else
    MsgBox "Terdapat kesalahan dalam pengisian data !!!", vbInformation
End If
End Sub
Private Sub CParameter_Click()
On Error Resume Next
Parameter.ID_Parameter = Get_IDParameter(CParameter.Text)
Call Isi_Combo_Range(Parameter.ID_Parameter)
If CRange.ListCount > 0 Then
    CRange.ListIndex = 0

```

---

```

End If
If IsNumeric(TxtNo.Text) Then
    If Tabel_Data.TextMatrix(CInt(TxtNo.Text), CParameter.ListIndex + 2) <> ""
Then
    CRange.Text = Tabel_Data.TextMatrix(CInt(TxtNo.Text),
CParameter.ListIndex + 2)
    End If
End If
End Sub

```

```

Private Sub Tabel_Data_DbClick()
    TxtNo.Text = Tabel_Data.TextMatrix(Tabel_Data.Row, 0)
    TxtNama.Text = Tabel_Data.TextMatrix(Tabel_Data.Row, 1)
    If (Tabel_Data.Col - 2 >= 0) Then
        CParameter.ListIndex = Tabel_Data.Col - 2
    End If
    CParameter_Click
End Sub

```

```

FParameter:
Option Explicit
Private Type Parameter_Rec
    ID_Parameter As String
    Nama_Parameter As String
    Range_Flexibel As Boolean
    Keterangan_IO As Byte
End Type
Dim Parameter As Parameter_Rec
Function Get_ID_Parameter(Status_IO As Boolean) As String
Dim Temp As String, Sqlcmd As String
    If Not Status_IO Then
        Temp = "I-"
    Else
        Temp = "O-"
    End If
    Sqlcmd = "Select Top 1 ID_Parameter From Parameter Where ID_Parameter Like '"
& Temp & "%' order by ID_Parameter desc "
    Set Myrs = Mydb.Execute(Sqlcmd)
    If Not (Myrs.EOF And Myrs.BOF) Then
        Parameter.ID_Parameter = Temp & Format(CLng(Right(Myrs!ID_Parameter, 3))
+ 1, "000")
    Else
        Parameter.ID_Parameter = Temp + "001"
    End If
    Get_ID_Parameter = Parameter.ID_Parameter
End Function
Function Find_ID_Parameter(ID_Parameter As String) As Boolean
Dim Sqlcmd As String
    Find_ID_Parameter = False
    Sqlcmd = "Select * From Parameter Where ID_Parameter=" & ID_Parameter & " "

```

---

```

Set Myrs = Mydb.Execute(Sqlcmd)
If Not (Myrs.EOF And Myrs.BOF) Then
    Find_ID_Parameter = True
    Parameter>Nama_Parameter = Myrs!Nama_Parameter
    If Myrs!Keterangan_IO Then
        Parameter.Keterangan_IO = 1
    Else
        Parameter.Keterangan_IO = 0
    End If
    Parameter.Range_Flexibel = Myrs!Range_Flexibel
End If
End Function
Private Sub Setting_Awal()
Dim i As Integer
Dim Temp As Boolean

CParameter.ListIndex = 0
For i = TxtParameter.LBound To TxtParameter.UBound
    TxtParameter(i).Text = ""
Next
If CParameter.ListIndex = 0 Then
    Temp = False
Else
    Temp = True
End If
TxtParameter(0).Text = Get_ID_Parameter(Temp)
TxtParameter(0).Enabled = False
CRange.Value = 0
Laporan_Parameter Temp
'Laporan_Parameter (If(CParameter.ListIndex = 0, False, True))
Btn(0).Caption = "&Save"
Btn(0).Enabled = True
Btn(1).Enabled = False
End Sub
Private Sub Laporan_Parameter(Status_IO As Boolean)
Dim Sqlcmd As String
Sqlcmd = "Select
Parameter.ID_Parameter,Parameter>Nama_Parameter,IIF(Parameter.Keterangan_IO,'O
UTPUT','INPUT'), " & _
    "IIF(Parameter.Range_Flexibel,'FLEXIBEL','TETAP') From Parameter " & _
    "Where Keterangan_IO = " & Status_IO & " order by Parameter.ID_Parameter "
Set Myrs = Mydb.Execute(Sqlcmd)
Set Tabel_Parameter.DataSource = Myrs
With Tabel_Parameter
    .Columns(0).Width = 10
    .Columns(0).Caption = "ID Parameter"
    .Columns(0).Alignment = dbLeft
    .Columns(1).Caption = "Nama Parameter"
    .Columns(2).Width = 10
    .Columns(3).Width = 10

```

---

```

        .Columns(3).Caption = "Range Flexibel"

    End With
End Sub
Private Sub Btn_Click(Index As Integer)
    Select Case Index
    End Select
End Sub
Private Sub Save_Data()
Dim Status_KetIO As Boolean
Dim Sqlcmd As String
    With Parameter
        .ID_Parameter = TxtParameter(0).Text
        .Nama_Parameter = TxtParameter(1).Text
        .Range_Flexibel = IIf(CRange.Value = 1, True, False)
        .Keterangan_IO = CParameter.ListIndex
    End With
    If .Keterangan_IO = 0 Then
        Status_KetIO = False
    Else
        Status_KetIO = True
    End If
Dim Sqlcmd As String
    Parameter.ID_Parameter = TxtParameter(0).Text
    Sqlcmd = "Delete * From Parameter Where ID_Parameter = " &
Parameter.ID_Parameter & " "
    Mydb.BeginTrans
    Mydb.Execute (Sqlcmd)
    Mydb.CommitTrans
    Setting_Awal
End Sub

Private Sub CParameter_Click()
Dim Temp As Boolean
    If CParameter.ListIndex = 0 Then
        Temp = False
    Else
        Temp = True
    End If
    TxtParameter(0).Text = Get_ID_Parameter(Temp)
    Laporan_Parameter Temp
End Sub

Private Sub Tabel_Parameter_DblClick()
    If Not (Tabel_Parameter.Bookmark = 0) Then
        Parameter.ID_Parameter =
Tabel_Parameter.Columns(0).CellText(Tabel_Parameter.Bookmark)
        TxtParameter(0).Text = Parameter.ID_Parameter
        If Find_ID_Parameter(Parameter.ID_Parameter) Then
            TxtParameter(1).Text = Parameter.Nama_Parameter
            CParameter.ListIndex = Parameter.Keterangan_IO
        End If
    End If
End Sub

```

---

```

        CRange.Value = IIf(Parameter.Range_Flexibel, 1, 0)
        Btn(0).Caption = "&Update"
        Btn(1).Enabled = True
        TxtParameter(1).SetFocus
    End If
End If
End Sub

FRange
Option Explicit
Private Type Parameter_Rec
    ID_Parameter As String
End Type
Dim Parameter As Parameter_Rec
Private Type Range_Rec
    Id_Range As String
    ID_Parameter As String
    Keterangan As String
    StatusVParameter As Boolean
    Range_Tambahan As Boolean
End Type
Dim Range As Range_Rec

Private Sub Setting_Awal()
    Dim i As Integer
    Dim Sqlcmd As String
    For i = TxtRange.LBound To TxtRange.LBound
        TxtRange(i).Text = ""
    Next
    Set DCRange.RowSource = Nothing
    DCRange.Text = ""
    Sqlcmd = "Select * From Parameter order by ID_Parameter"
    Set Myrs = Mydb.Execute(Sqlcmd)
    If Not (Myrs.BOF And Myrs.EOF) Then
        Set DCRange.RowSource = Myrs
        DCRange.ListField = "Nama_Parameter"
    End If
    CVParameter.Value = 0
    Call Laporan_Range(" ")
    DCRange.Enabled = True
    Btn(0).Caption = "&Save"
    TxtRange(0).Enabled = False
    TxtRange(1).Text = ""
    Btn(0).Enabled = False
    Btn(1).Enabled = False
End Sub
Private Sub Laporan_Range(ID_Parameter As String)
    Dim Sqlcmd As String

```

---

```
Sqlcmd = "Select ID_Range,Keterangan,IIF(VParameter,'YA','TIDAK') From Range  
Where ID_Parameter = "" & ID_Parameter & "" And Range_Tambahan= False ORDER  
BY ID_Range "
```

```
Set Myrs = Mydb.Execute(Sqlcmd)
```

```
Set Tabel_Range.DataSource = Myrs
```

```
With Tabel_Range
```

```
.Columns(0).Width = 10
```

```
.Columns(0).Caption = "No Range"
```

```
.Columns(0).Alignment = dbLeft
```

```
.Columns(1).Width = 25
```

```
.Columns(1).Caption = "Keterangan"
```

```
.Columns(2).Width = 20
```

```
.Columns(2).Caption = "Parameter Bayangan"
```

```
End With
```

```
End Sub
```

```
Private Function Get_ID_Parameter(Nama_Parameter As String) As String
```

```
Dim Sqlcmd As String
```

```
Get_ID_Parameter = ""
```

```
Sqlcmd = "Select ID_Parameter From Parameter Where Nama_Parameter="" &  
Nama_Parameter & "" "
```

```
Set Myrs = Mydb.Execute(Sqlcmd)
```

```
If Not (Myrs.EOF And Myrs.BOF) Then
```

```
Get_ID_Parameter = Myrs!ID_Parameter
```

```
End If
```

```
End Function
```

```
Private Function Get_ID_Range(ID_Parameter As String) As String
```

```
Dim Sqlcmd As String
```

```
Sqlcmd = "Select Top 1 ID_Range From Range Where ID_Parameter = "" &  
ID_Parameter & "" order by ID_Range desc "
```

```
Set Myrs = Mydb.Execute(Sqlcmd)
```

```
If Not (Myrs.EOF And Myrs.BOF) Then
```

```
Get_ID_Range = ID_Parameter & Format(CInt(Right(Myrs!Id_Range, 1)) + 1,  
"0")
```

```
Else
```

```
Get_ID_Range = ID_Parameter & "1"
```

```
End If
```

```
End Function
```

```
Function Find_ID_Parameter(ID_Parameter As String) As Boolean
```

```
Dim Sqlcmd As String
```

```
Find_ID_Parameter = False
```

```
Sqlcmd = "Select * From Parameter Where ID_Parameter="" & ID_Parameter & "" "
```

```
Set Myrs = Mydb.Execute(Sqlcmd)
```

```
If Not (Myrs.EOF And Myrs.BOF) Then
```

```
Find_ID_Parameter = True
```

```
Parameter.Nama_Parameter = Myrs!Nama_Parameter
```

```
If Myrs!Keterangan_IO Then
```

```
Parameter.Keterangan_IO = 1
```

```
Else
```

```
Parameter.Keterangan_IO = 0
```

```
End If
```