

**RANCANG BANGUN MEDIA PEMBELAJARAN
MENGENAL HEWAN BERBASIS JAVA DENGAN METODE
FORWARD CHAINING**

SKRIPSI

Oleh :

MAULANA MUKHTAR ZAKY

NIM. 06550030



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG**

2013

ABSTRAK

Teknologi telah membawa kehidupan manusia menjadi semakin maju. Komputer merupakan salah satu contoh kemajuan teknologi yang paling sering digunakan oleh masyarakat dan mengalami perkembangan. Perkembangan teknologi yang sangat pesat telah member dampak positif di berbagai bidang, salah satunya pada bidang pendidikan. Oleh karena itu penulis akan membuat sebuah aplikasi sederhana media pembelajaran mengenal hewan berbasis *Java* dengan metode *Forward Chaining*.

Aplikasi tersebut diperuntukkan bagi murid - murid yang masih duduk di sekolah dasar salah satunya kelas 1. Aplikasi akan berjalan secara berurutan sampai menuju akhir / hasil yang diharapkan. Rangkaian perjalanannya nanti adalah dari adanya gambar hewan – hewan yang bisa dipilih salah satu, kemudian dari hewan tersebut akan muncul sebuah table suara hewan dan juga dalam bahasa inggris serta terdapat penjelasan mengenai hewan tersebut. Setelah itu ada sebuah menu atau tabel yang digunakan sebagai tebak suara hewan atau dengan kata lain para murid secara satu persatu dibimbing untuk bisa menebak. Disini mereka bisa diajak bermain kuis tebak suara.

Di dalam penerapan di sekolah khususnya sekolah dasar kelas 1, aplikasi program pengenalan hewan berbasis *Java* yang sederhana tersebut sangat membantu untuk mempermudah siswa belajar lebih cepat dan mudah memahami serta mengerti tentang pelajaran pengenalan komputer, bahasa inggris dan informasi lain dari berbagai jenis hewan.

Kata kunci: Java, Gel, Forward Chaining.

**RANCANG BANGUN MEDIA PEMBELAJARAN
MENGENAL HEWAN BERBASIS JAVA DENGAN METODE
FORWARD CHAINING**

SKRIPSI

Diajukan Kepada :
Fakultas Sains dan Teknologi
Universitas Islam Negeri
Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana (S.Kom)

Oleh:

MAULANA MUKHTAR ZAKY

NIM. 06550030

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG**

2013

LEMBAR PERSETUJUAN

**RANCANG BANGUN MEDIA PEMBELAJARAN
MENGENAL HEWAN BERBASIS JAVA DENGAN METODE
FORWARD CHAINING**

SKRIPSI

Oleh :

MAULANA MUKHTAR ZAKY

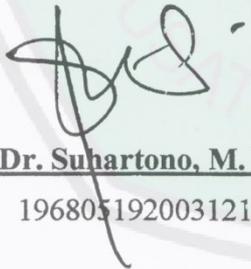
NIM. 06550030

Telah Diperiksa dan Disetujui untuk Diuji :

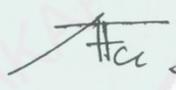
Tanggal : 12 Juli 2013

Pembimbing I,

Pembimbing II,


Dr. Suhartono, M. Kom

196805192003121001


Fatchurrochman, M. Kom

197007312005011002

Mengetahui,

Ketua Jurusan Teknik Informatika




Ririen Kusumawati, S.Si, M.Kom

197203092005012002

LEMBAR PENGESAHAN
RANCANG BANGUN MEDIA PEMBELAJARAN
MENGENAL HEWAN BERBASIS JAVA DENGAN METODE
FORWARD CHAINING

SKRIPSI

Oleh :
MAULANA MUKHTAR ZAKY
NIM. 06550030

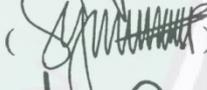
Telah Dipertahankan di Depan Dewan Penguji Skripsi
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik Informatika (S.Kom)
Tanggal : 17 Juli 2013

Susunan Dewan Penguji:

1. Penguji Utama: **Yunifa Miftachul Arif, MT**
19830616 201101 1 004
2. Ketua : **A'la Syauqi, M.Kom**
19771201 200801 1 007
3. Sekretaris : **Dr. Suhartono, M.Kom**
19680519 200312 1 001
4. Anggota : **Fatchurrochman, M.Kom**
19700731 2005011 002

Tanda Tangan :

()

()

()

()

Mengetahui,

Ketua Jurusan Teknik Informatika



Ririen Kusumawati, S.Si, M.Kom
197203092005012002

PERNYATAAN KEASLIAN TULISAN

Saya yang bertandatangan dibawah ini :

Nama : Maulana Mukhtar Zaky
NIM : 06550030
Jurusan : Teknik Informatika
Fakultas : Sains dan Teknologi

Menyatakan dengan sebenarnya bahwa tugas akhir/skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila dikemudian hari terbukti atau dibuktikan tugas akhir/skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 17 Juli 2013

Yang membuat pernyataan,

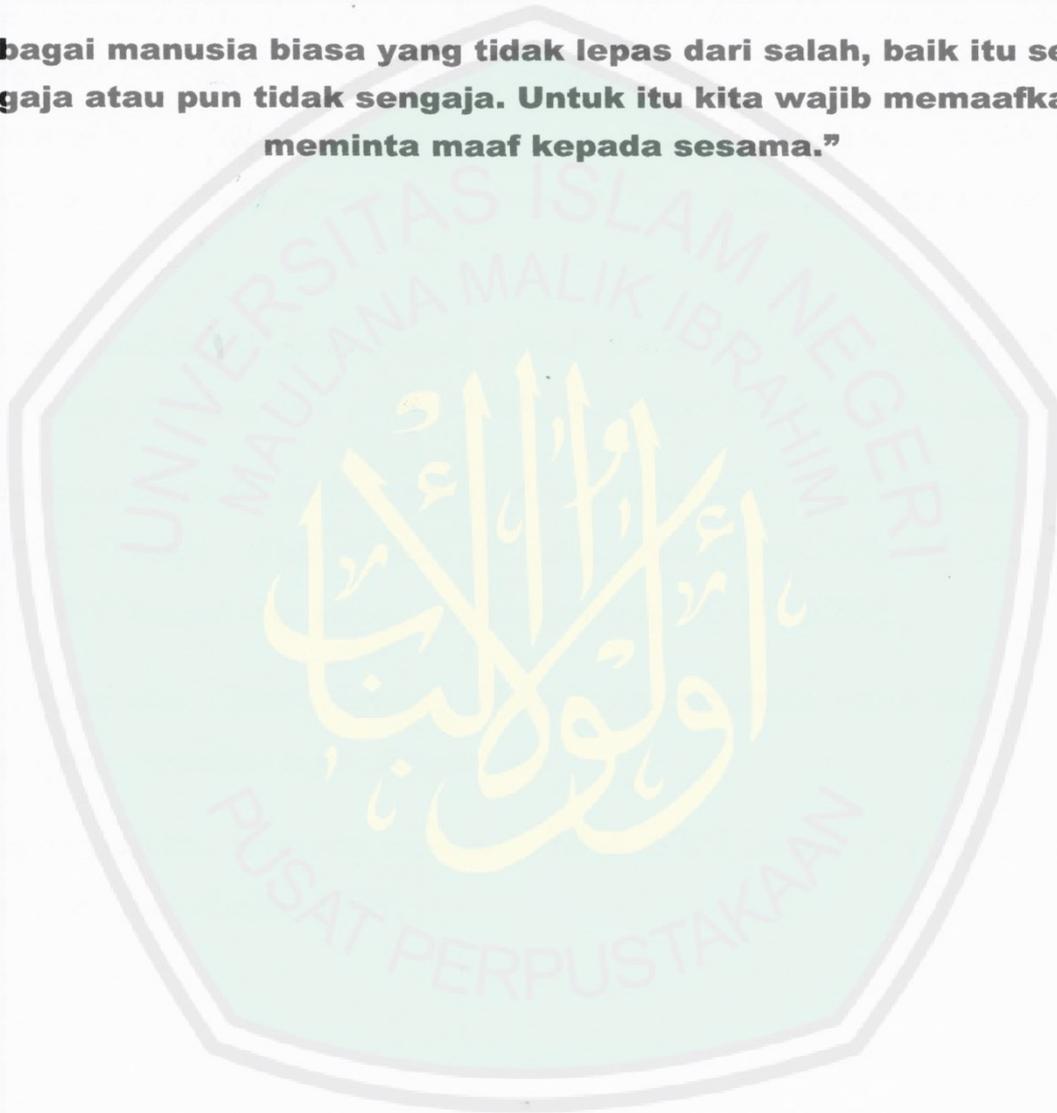

METERAI
TEMPEL
63D93ABF716033574
ENAM RIBU RUPIAH
6000 DJP
Maulana Mukhtar Zaky

NIM. 06550030

MOTTO

"Di dunia ini tiada yang sempurna, kecuali Allah SWT".

Sebagai manusia biasa yang tidak lepas dari salah, baik itu secara sengaja atau pun tidak sengaja. Untuk itu kita wajib memaafkan dan meminta maaf kepada sesama."



Alhamdulillah rasa syukur kepada Allah SWT, Semoga Skripsi ini Bermanfaat bagi Semuanya

Skripsi ini kupersembahkan untuk:

Kedua Orangtuaku Tercinta: Ayahanda H. Moh. Hasyim dan Ibunda Hj. Husnul Chotimah, yang telah mendo'akan, menyayangi, serta membiayaiiku... Hanya Allah SWT yang bisa membalas, semoga beliau berdua selalu diberi kesehatan, kebahagiaan, dan panjang umur. Amin YRA.

Saudara-saudaraku: Mas Tatok, Mbak Yani, Mas Gurit, Mbak Novi, serta semua keluargaku yang selalu memberi bantuan dan motivasi.

Dosen-dosen Teknik Informatika: Pak Subartono, Pak Fatchurrohman, Pak A'la Syauqi, Pak Yunifa M., dan Pak Badruddin M. Semoga Allah SWT membalas semua kebaikannya, Amin YRA.

Para Dosen dan Staff Jurusan Teknik Informatika yang telah mendukung terselesainya skripsi ini.

Teman-temanku tercinta, semoga Allah SWT membalas kebaikan mereka semua, Amin YRA.

KATA PENGANTAR

Kiranya tidak ada kata yang pantas terucap dari penulis selain Alhamdulillahirobbil'alamin kepada Allah SWT, atas segala petunjuk, kekuatan, dan kejernihan pikiran dalam menyusun tugas akhir ini, sehingga penulis dapat menyelesaikan tugas akhir ini dengan baik. Serta tidak lupa pula Sholawat dan salam tetap tercurahkan atas Nabi besar Rasullullah Muhammad SAW. Tugas akhir ini disusun untuk memenuhi salah satu persyaratan dalam memperoleh gelar sarjana Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Semoga penyusunan tugas akhir ini dapat memberikan pengetahuan tambahan dalam mengembangkan aplikasi-aplikasi pemrograman komputer dan bermanfaat bagi pengguna yang membutuhkannya.

Meskipun penulis sudah berupaya semaksimal mungkin untuk mempersembahkan yang terbaik, namun penulis menyadari bahwa tugas akhir ini masih jauh dari sempurna. Oleh karena itu segala saran dan kritik yang konstruktif sangatlah diharapkan.

Dengan segala keterbatasan, penulis menyadari pula bahwa laporan skripsi ini takkan terwujud tanpa bantuan, bimbingan, dan dorongan dari berbagai pihak. Untuk itu, dengan segala kerendahan hati, penulis menyampaikan ucapan terimakasih kepada :

1. Prof. Dr. H. Mudjia Rahardjo, M.Si, sebagai Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.

2. Drh. Bayyinatul Muchtaromah, M.Si, sebagai Dekan Fakultas Sains dan Teknologi.
3. Ririen Kusumawati, S.Si, M.Kom, sebagai Ketua Jurusan Teknik Informatika.
4. Suhartono, M.Kom sebagai dosen pembimbing pertama skripsi.
5. Fatchurrohman, M.Kom sebagai dosen pembimbing kedua skripsi sekaligus dosen wali.
6. Ustadz Badruddin Muhammad, sebagai salah satu dosen Fakultas Syariah Universitas Maulana Malik Ibrahim Malang yang telah memberi semangat dan meluangkan waktunya untuk membantu penulis dalam menyelesaikan skripsi ini.
7. Bibit Sudarmadji, S.Pd, sebagai Kepala SDN Kalirejo1 dan Wiwik Suparti, S.Ag, sebagai guru pendamping.
8. Seluruh dosen, staff, dan karyawan Fakultas Sains dan Teknologi Universitas Maulana Malik Ibrahim Malang.
9. Kedua orangtua Drs. H. Moh. Hasyim dan Hj. Husnul Chotimah yang telah membimbing, mendoakan, dan membiayai penulis.
10. Semua teman civitas akademika teknik informatika dan pihak yang telah membantu penulis dalam menyelesaikan tugas skripsi ini.

Semoga Allah SWT membalas kebaikan dan selalu mencurahkan hidayah serta taufik-Nya, Amin.

Malang, Juli 2013

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN	iv
MOTTO	v
KATA PENGANTAR	vii
ABSTRAKSI	ix
DAFTAR ISI	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xiv
BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah	5
1.3 Batasan Masalah	6
1.4 Tujuan Penelitian	6
1.5 Manfaat Penelitian	7
1.6 Sistematika Penulisan	7
BAB II TINJAUAN PUSTAKA	
2.1 Pengertian Media Pembelajaran	9
2.2 Tujuan dan Manfaat Media Pembelajaran	13
2.3 Bahasa Pemrograman <i>Java</i>	16
2.4 Elemen Bahasa	17
2.5 Aturan Penamaan	17
2.5.1 Penamaan Paket	18
2.5.2 Penamaan Kelas Dan Interface	18
2.5.3 Penamaan Method	18
2.5.4 Penamaan Variabel	19
2.5.5 Penamaan Konstanta	19
2.6 Objek	19
2.6.1 Menciptakan Objek	20

2.6.2	Memeriksa Tipe Objek	20
2.6.3	Menghapus Objek	20
2.7	Teknik Dasar Java	21
2.7.1	Deklarasi Konstanta	21
2.7.2	Evaluasi Ekspresi di Println	21
2.7.3	Operator Ternary	22
2.7.4	Komparasi Integer	22
2.8	Operasi Karakter	22
2.8.1	Memeriksa Jenis Karakter	23
2.8.2	Manipulasi Karakter	23
2.8.3	Komparasi Karakter	24
2.9	Konversi dan Casting	25
2.10	Membuat Window Swing	27
2.11	Menggunakan Button	29
2.12	Mengolah Data Gambar	29
2.13	Menampilkan Data Gambar	31
2.14	Menjalankan Aplikasi	32
2.15	Memainkan Audio	34
2.16	Sistem Pakar	36
2.16.1	Pendekatan Inferensi Dalam Sistem Pakar	37
2.16.2	Teknik Heuristik	38
2.16.3	Inferensi Berbasis Aturan	39
2.16.4	Pelacakan ke Belakang (<i>Backward Chaining</i>)	40
2.16.4	Pelacakan ke Depan (<i>Forward Chaining</i>)	40
BAB III ANALISIS DAN PERANCANGAN SISTEM		
3.1	Analisis Sistem	42
3.1.1	Alat dan Bahan Penelitian	43
3.1.2	Alat Pengumpul Data	43
3.2	Waktu dan Tempat Penelitian	44
3.3	Rancangan Sistem <i>Forward Chaining</i>	45
3.4	Alur Program	47

BAB IV HASIL DAN PEMBAHASAN

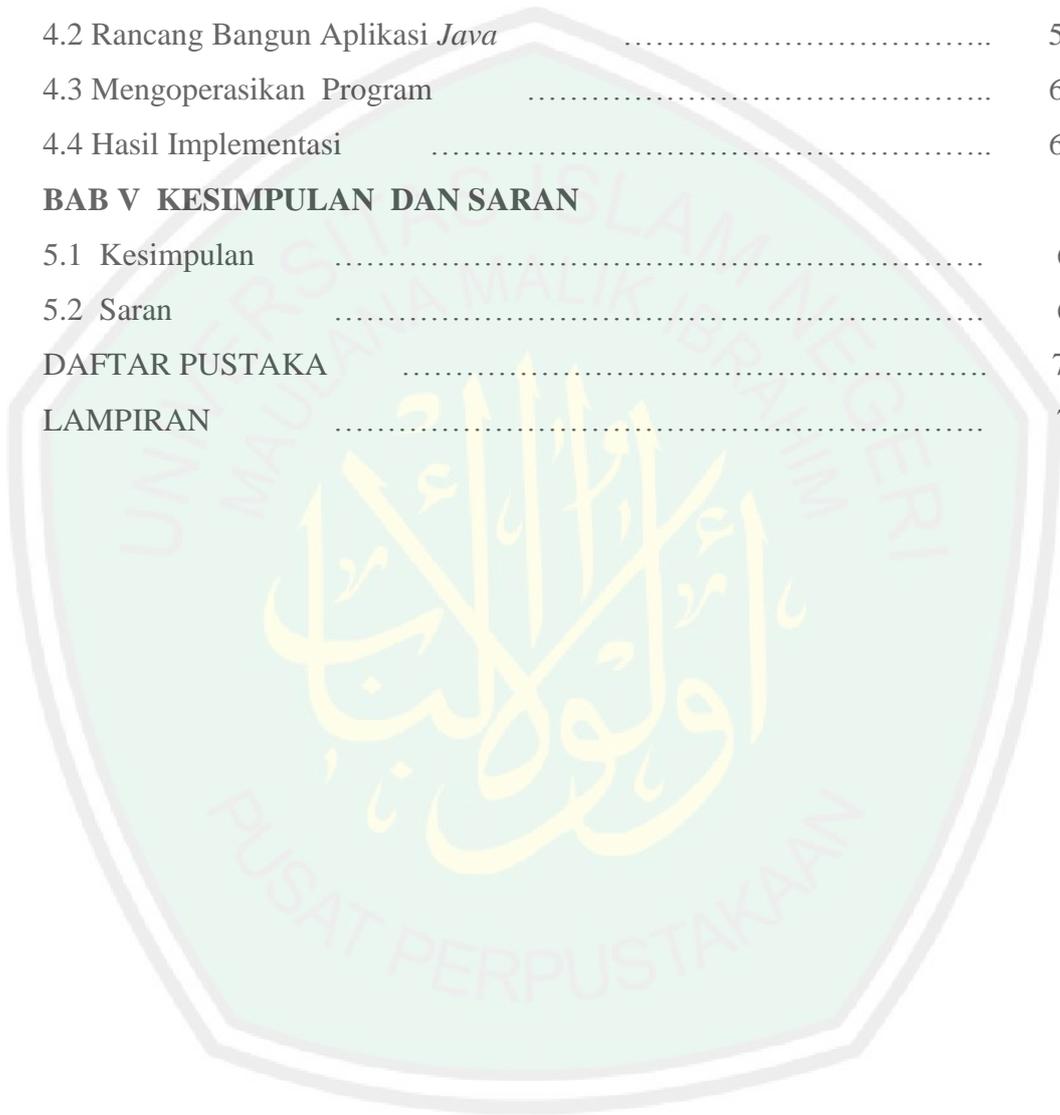
4.1 Rancang Bangun Berdasarkan Forward Chaining	48
4.2 Rancang Bangun Aplikasi <i>Java</i>	53
4.3 Mengoperasikan Program	60
4.4 Hasil Implementasi	65

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan	68
5.2 Saran	69

DAFTAR PUSTAKA	70
----------------------	----

LAMPIRAN	71
----------------	----

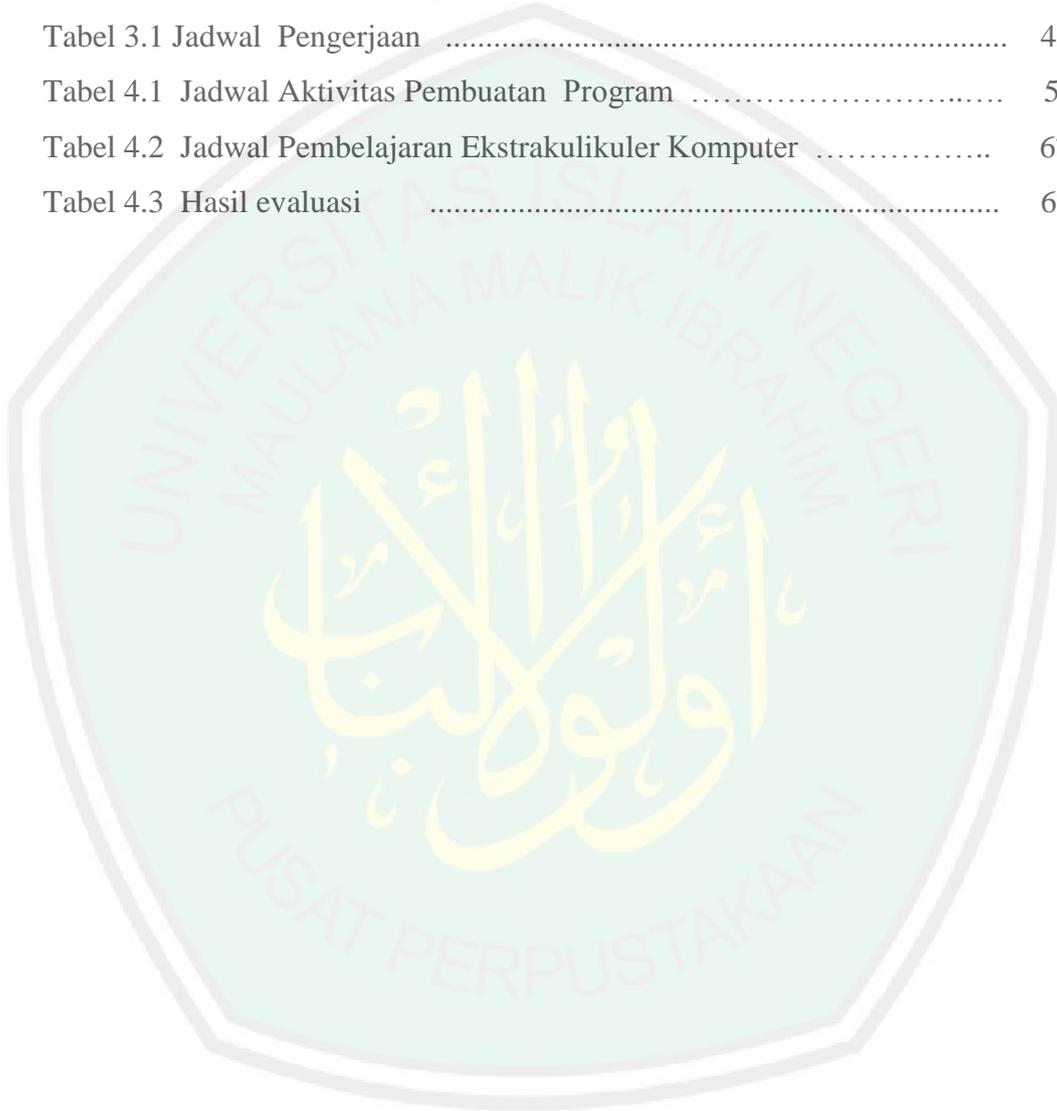


DAFTAR GAMBAR

Gambar 2.1 Perekayasa Pengetahuan Dengan Input Output Sistem Pakar	38
Gambar 2.2 <i>Forward Chaining</i>	41
Gambar 3.1 Proses <i>Forward Chaining</i>	45
Gambar 3.2 Alur Program	47
Gambar 4.1 Alur Program Menggunakan Metode <i>Forward Chaining</i>	50
Gambar4.2 Menu gambar dan pilihan hewan di kebun binatang	61
Gambar4.3 Petunjuk Operasional	62
Gambar4.2.a Menugambar dan pilihan hewan di kebun binatang	62
Gambar4.4 Pilihan hewan di kebun binatang	63
Gambar4.5 Menugambar dan pilihan hewan di peternakan	64
Gambar4.6 Pilihan hewan di peternakan.	65

DAFTAR TABEL

Tabel 2.1 Tabel konversi string	25
Tabel 3.1 Jadwal Pengerjaan	45
Tabel 4.1 Jadwal Aktivitas Pembuatan Program	52
Tabel 4.2 Jadwal Pembelajaran Ekstrakurikuler Komputer	67
Tabel 4.3 Hasil evaluasi	67



ABSTRAK

Teknologi telah membawa kehidupan manusia menjadi semakin maju. Komputer merupakan salah satu contoh kemajuan teknologi yang paling sering digunakan oleh masyarakat dan mengalami perkembangan. Perkembangan teknologi yang sangat pesat telah member dampak positif di berbagai bidang, salah satunya pada bidang pendidikan. Oleh karena itu penulis akan membuat sebuah aplikasi sederhana media pembelajaran mengenal hewan berbasis *Java* dengan metode *Forward Chaining*.

Aplikasi tersebut diperuntukkan bagi murid - murid yang masih duduk di sekolah dasar salah satunya kelas 1. Aplikasi akan berjalan secara berurutan sampai menuju akhir / hasil yang diharapkan. Rangkaian perjalanannya nanti adalah dari adanya gambar hewan – hewan yang bisa dipilih salah satu, kemudian dari hewan tersebut akan muncul sebuah table suara hewan dan juga dalam bahasa inggris serta terdapat penjelasan mengenai hewan tersebut. Setelah itu ada sebuah menu atau tabel yang digunakan sebagai tebak suara hewan atau dengan kata lain para murid secara satu persatu dibimbing untuk bisa menebak. Disini mereka bisa diajak bermain kuis tebak suara.

Di dalam pengetrapan di sekolah khususnya sekolah dasar kelas 1, aplikasi program pengenalan hewan berbasis *Java* yang sederhana tersebut sangat membantu untuk mempermudah siswa belajar lebih cepat dan mudah memahami serta mengerti tentang pelajaran pengenalan komputer, bahasa inggris dan informasi lain dari berbagai jenis hewan.

Kata kunci: *Java*, Gel, *Forward Chaining*.



BAB I

PENDAHULUAN

1.1 Latar Belakang

Di dalam jaman yang serba komputerisasi ini harus bisa mengikuti laju perkembangan dunia teknologi yang semakin hari semakin berkembang, karena dalam berbagai proses kegiatan selalu dijumpai keberadaanya. Selama ini komputer dapat dipakai untuk membantu orang-orang dalam memecahkan masalah. Semakin cerdas sistem itu dan semakin ditingkatkan level penanganan informasinya, maka semakin aktif peranan yang dipermainkan oleh komputer. Selama ini dalam perkembangannya, terjadi peningkatan minat dalam menggunakan komputer untuk mengasah kecerdasan manusia. Semakin cerdas manusia semakin tinggi tingkat perkembangan ilmu pengetahuan dan teknologi komputer yang diciptakan untuk membantu menyelesaikan permasalahan yang berkembang.

Ilmu pengetahuan dan teknologi merupakan suatu kesatuan yang tidak dapat terpisahkan didalam realita kehidupan manusia saat ini. Sebagaimana yang telah diketahui dalam Al-Qur'an tentang segala hal yang telah diciptakan oleh Allah SWT, melalui ciptaannya dan realitas kongkret yang terdapat di bumi dan di langit. Inilah sesungguhnya yang terdapat pada ilmu pengetahuan yang mana mengadakan observasi lalu menarik hukum-hukum alam berdasarkan observasi dan eksperimen. Sebagaimana telah dituliskan di dalam Al-Qur'an yang

menunjukkan bahwa manusia memiliki naluri untuk selalu haus akan ilmu pengetahuan, keinginan yang tidak akan pernah puas, keinginan menuntut ilmu, dan keinginan penuntut harta. Hal ini dapat menjadi pemicu bagi manusia untuk terus mengembangkan ilmu pengetahuan dan teknologi dengan memanfaatkan anugrah Allah SWT yang dilimpahkan kepadanya. Allah SWT berfirman di dalam Al-Qur'an surat Al-Mujadilah/58:11, yaitu:

يَأَيُّهَا الَّذِينَ ءَامَنُوا إِذَا قِيلَ لَكُمْ تَفَسَّحُوا فِي الْمَجَالِسِ فَافْسَحُوا
 يَفْسَحَ اللَّهُ لَكُمْ وَإِذَا قِيلَ أَنْشُرُوا فَأَنْشُرُوا يَرْفَعُ اللَّهُ الَّذِينَ ءَامَنُوا
 مِنْكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ ﴿١١﴾

Artinya: "Hai orang-orang yang beriman, apabila dikatakan kepadamu: "berlapang-lapanglah kamu dalam majelis", maka lapangkanlah. Niscaya Allah akan memberi kelapangan untukmu. Dan apabila dikatakan: "berdirilah kamu", maka berdirilah, niscaya Allah akan meninggikan orang-orang yang beriman diantaramu dan orang-orang yang diberi ilmu pengetahuan beberapa derajat. Dan Allah Maha Mengetahui apa yang kamu kerjakan" (Qs. Al-Mujadilah / 58:11).

Perkembangan ilmu pengetahuan dan teknologi yang salah satunya diwakili oleh komputer saat ini telah banyak diberikan atau diajarkan pada anak-anak sekolah dasar. Didalam aplikasi penerapan ilmu pengetahuan dan perkembangan teknologi yang salah satunya adalah pembelajaran komputer di lingkungan

anak-anak sekolah merupakan hal yang sangat penting untuk mengembangkan kecerdasan dan meningkatkan ilmu pengetahuan dan teknologi.

Dilihat dari perkembangan anak-anak sekolah dasar jaman sekarang berbeda dengan anak-anak sekolah dasar jaman dahulu, seperti fakta yang sudah ada anak-anak sekolah dasar jaman sekarang lebih memilih bermain *game* daripada bermain kelereng ataupun bola bekel. Hal ini tercermin pada kehidupan anak-anak sekolah dasar yang tinggal di daerah perkotaan yang sangat pesat akan perkembangan teknologi yang mana komputer untuk aplikasi permainan atau *game* bukan merupakan hal yang baru bagi mereka. Tetapi bagi anak-anak sekolah dasar yang tinggal di desa komputer masih merupakan hal yang awam bagi mereka dalam menunjang pembelajaran di sekolah.

Salah satu peluang untuk memberikan pemahaman kepada anak-anak sekolah dasar yang berada di desa, adalah dengan menggunakan media komputer dengan aplikasi yang disenangi, yang bisa menunjukkan kepada anak-anak tentang belajar yang lebih mudah dimengerti dan dipahami oleh anak-anak sekolah dasar dengan cepat. Cara ini diharapkan apa yang disampaikan oleh guru, dapat dengan mudah dan cepat dicerna serta meningkatkan pemahaman siswa, misalkan tentang pengetahuan berbagai nama dan suara hewan serta permainan melalui di media komputer.

قَدْ جَاءَكُمْ بَصَائِرٌ مِنْ رَبِّكُمْ فَمَنْ أَبْصَرَ فَلِنَفْسِهِ ۗ وَمَنْ عَمِيَ فَعَلَيْهَا
 وَمَا أَنَا عَلَيْكُمْ بِحَفِيظٍ ﴿١٠٤﴾

Artinya: “*Sesungguhnya telah datang dari Tuhanmu bukti-bukti yang terang; maka barangsiapa melihat (kebenaran itu), maka (manfaatnya) bagi dirinya sendiri; dan barangsiapa buta (tidak melihat kebenaran itu), maka kemudharatannya kembali kepadanya. Dan aku (Muhammad) sekali-kali bukanlah pemelihara(mu)*”. (QS. Al-Baqarah/6: 104)

Ayat ini mengatakan, sesuatu yang harus dinyatakan maka haruslah dikatakan sehingga jalan yang benar dapat dijelaskan. Harus menjelaskan sesuatu yang diperlukan untuk mengetahui hal-hal yang benar, begitu juga harus mengungkapkan dalil dan bukti untuk menerima hal itu. Dengan begitu ada orang yang berkeinginan lalu menerima. Hal itu akan bermanfaat bagi diri sendiri dan orang lain. Tetapi apabila seseorang tidak mau menerima, maka tiada hak memaksanya, karena Allah SWT telah membentangkan jalan kekufuran dan iman buat umat manusia dan setiap orang bebas menentukan pilihannya.

Dari ayat tadi terdapat dua pelajaran yang dapat dipetik:

- a. Tugas ulama / guru hanyalah membimbing umat manusia, sehingga hujjah dapat disempurnakan dikalangan manusia dan pintu alasan tidak tersisa sama sekali. Selain itu, tidak berhak memaksa manusia lainnya.
- b. Tanda-tanda kekufuran adalah tidak menganggap benar ajaran para nabi. Itu petanda butanya mata hati mereka.

Pelaksanaan kegiatan belajar mengajar merupakan interaksi antara guru dan siswa yang cukup dominan. Proses interaksi antara guru dan siswa tidak semata-mata hanya tergantung cara atau metode yang dipakai, tetapi komponen-komponen lain juga mempengaruhi keberhasilan dalam interaksi belajar mengajar. Komponen-komponen tersebut, antara lain: guru, siswa, metode, alat / teknologi, sarana dan tujuan. Atas dasar uraian diatas, maka penulis dalam hal ini ingin membahas mengenai bagaimana caranya membuat suatu produk sistem yang dapat dengan mudah melayani para user atau guru tanpa harus bersusah payah, yang memakan waktu, tenaga serta biaya. Oleh sebab itu, penulis akan membuat sebuah aplikasi sederhana tentang media pembelajaran mengenal hewan-hewan yang ditujukan untuk mempermudah anak-anak didik mempelajari bahkan mengenal lebih jauh hewan-hewan tersebut. Pada penelitian ini akan dibuat sebuah aplikasi pengenalan hewan yang ditujukan untuk murid – murid sekolah dasar yang tepatnya di salah satu kelas, yaitu kelas 1 Sekolah Dasar Negeri Kalirejo 1 Kecamatan Dringu Kabupaten Probolinggo yang berbasis *Java* dengan metode *Forward Chaining*. Aplikasi tersebut akan dibuat dengan menggunakan *Java* yang disertai editor *Gel* sebagai media penunjangnya. *Java* sudah cukup lama dan sangat familiar dikalangan para programmer, walaupun sebagian orang masih terasa asing mendengarnya.

Oleh karena itu pula penulis mewujudkannya dengan mencoba membuat suatu program aplikasi komputer yaitu **”Rancang Bangun Media Pembelajaran Mengenal Hewan Berbasis *Java* Dengan Metode *Forward Chaining*”**.

b.2 Rumusan Masalah

Untuk efektifitas maka perlu sekali dirumuskan permasalahan yang ada dengan tujuan agar proses mudah dilakukan dan terstruktur. Berdasarkan latar belakang di atas dirumuskan masalah dalam pembahasan ini sebagai berikut :

- a. Bagaimana membuat rancang bangun aplikasi media pembelajaran menggunakan sarana komputer dengan memakai aplikasi *Java* yang menggunakan metode *forward chaining* sebagai media pembuatannya, sehingga diharapkan programnya bisa berjalan atau bekerja membantu proses belajar mengajar sesuai yang diinginkan.
- b. Belum adanya sarana media pembelajaran yang dapat membantu siswa dalam belajar mengenai pengenalan berbagai jenis hewan beserta informasi pendukung lainnya khususnya untuk anak-anak kelas 1 Sekolah Dasar Negeri Kalirejo 1 Kecamatan Dringu Kabupaten Probolinggo.

b.3 Batasan Masalah

- a. Aplikasi yang dibangun menggunakan *Java*.
- b. Metode yang digunakan adalah metode *Forward Chaining*.
- c. Penelitian dilakukan di Sekolah Dasar Negeri Kalirejo 1 Dringu, Kabupaten Probolinggo.

c.4 Tujuan Penelitian

- a. Membangun suatu aplikasi yang dapat memberikan pemahaman kepada siswa dengan menggunakan media komputer yang bisa menunjukkan kepada siswa tentang belajar yang lebih mudah dimengerti dan dipahami oleh siswa dengan cepat serta sebagai sarana atau media belajar yang lebih menyenangkan.
- b. Memanfaatkan salah satu fungsi program *Java* sebagai *software* yang mudah dalam aplikasinya untuk membuat sebuah rancang bangun aplikasi pembelajaran mengenal hewan,
- c. Hasil dari penelitian ini kiranya dapat digunakan sebagai tambahan informasi dalam meningkatkan *output* pendidikan khususnya di perguruan tinggi, yakni Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.

c.5 Manfaat Penelitian

- a. Bagi SD Negeri Kalirejo 1 Dringu, Kab. Probolinggo diharapkan anak-anak didik secara tidak langsung belajar mengenal komputer dengan melihat gambaran secara jelas pengenalan berbagai jenis hewan serta dapat lebih mudah dan cepat dalam menerima atau mentransfer berbagai bentuk ilmu pengetahuan yang dilakukan di sekolah secara umum.

- b. Memberikan sarana baru bagi guru dan siswa dalam sistem belajar mengajar yang lebih menyenangkan dengan memanfaatkan sarana komputer.
- c. Bagi penulis dapat menambah ilmu dan pengalaman dibidang teknik informatika serta membantu memperkenalkan anak-anak atau para siswa belajar mengenal hewan lewat dunia maya.
- d. Hasil dari penelitian ini kiranya dapat digunakan sebagai tambahan informasi atau bahan referensi dalam meningkatkan *output* pendidikan khususnya di perguruan tinggi, yakni Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.

d.6 Sistematika Penulisan

Adapun sistematika penulisan skripsi ini adalah sebagai berikut :

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang, rumusan masalah, maksud dan tujuan, manfaat penelitian, batasan masalah, sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini menjelaskan tentang teori-teori yang terkait dengan masalah-masalah dan penyelesaian yang diambil.

BAB III METODE PENELITIAN

Bab ini menjelaskan tentang analisa yang dilakukan dalam merancang dan membuat aplikasi media pembelajaran.

BAB IV HASIL DAN PEMBAHASAN

Bab ini membahas tentang implementasi dari aplikasi yang telah dibuat secara keseluruhan. Serta melakukan mengujian terhadap aplikasi media pembelajaran tersebut untuk mengetahui bahwa aplikasi tersebut telah dapat dipakai menyelesaikan permasalahan yang dihadapi sesuai yang diharapkan.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan dan saran yang dapat diharapkan bermanfaat untuk mengembangkan pembuatan program aplikasi selanjutnya.

BAB II

TINJAUAN PUSTAKA

1.1 Pengertian Media Pembelajaran

Kemajuan dan perkembangan teknologi sudah demikian menonjol, sehingga penggunaan alat-alat bantu mengajar seperti alat-alat audio, visual serta perlengkapan sekolah disesuaikan dengan perkembangan jaman tersebut. Dan juga harus disesuaikan dengan tuntutan kurikulum sesuai dengan materi, metode, dan tingkat kemampuan belajar siswa agar dapat mencapai tujuan pembelajaran dengan baik disekolah. Untuk itu, para pengajar mulai berusaha membiasakan diri untuk menggunakan peralatan-peralatan seperti OHP, LCD, CD, VCD, video, computer dan internet dalam pembelajaran dikelas. dengan program pembelajarana yang dikembangkan ini patut dipelajari pengajar harus mempelajarinya agar mempermudah proses pembelajaran dan pendidikan, sehingga memudahkan pembelajaran untuk berjalan dengan baik dikelas.

Media adalah sebuah alat yang mempunyai fungsi menyampaikan pesan (Bovee, 1997). Media pembelajaran adalah sebuah alat yang berfungsi dan digunakan untuk menyampaikan pesan pembelajaran. Pembelajaran adalah proses komunikasi antara pembelajar, pengajar, dan bahan ajar. Banyak batasan atau pengertian yang dikemukakan para ahli tentang media, diantaranya adalah: Asosiasi Teknologi dan Komunikasi Pendidikan (*Asosociation of Education and*

Communication Technology (AECT). Secara umum dapat dikatakan bahwa substansi dari media pembelajaran adalah bentuk saluran, yang digunakan untuk menyalurkan pesan, informasi atau bahan pelajaran kepada penerima pesan atau pembelajar dapat pula dikatakan bahwa media pembelajaran adalah berbagai jenis komponen dalam lingkungan dalam lingkungan pembelajar yang dapat merangsang pembelajar untuk belajar.

Sebagaimana firman Allah SWT dalam Al-Qur'an surat Al-Isra' 84

قُلْ كُلُّ يَعْمَلُ عَلَىٰ شَاكِلَتِهِ، فَرَبُّكُمْ أَعْلَمُ بِمَنْ هُوَ أَهْدَىٰ سَبِيلًا ﴿٨٤﴾

Artinya : Katakanlah: “Tiap-tiap orang berbuat menurut keadaannya masing-masing”. Maka Tuhanmu lebih mengetahui siapa yang lebih benar jalannya”. (QS.Al-Isra' 84).

Ayat diatas mengatakan bahwa setiap orang yang melakukan suatu perbuatan, mereka akan melakukan sesuai keadaannya (termasuk di dalamnya keadaan alam sekitarnya) masing-masing. Hal ini menjelaskan bahwa dalam melakukan suatu perbuatan memerlukan media agar hal yang dimaksud dapat tercapai.

Dalam dunia pendidikan, seorang guru yang hendak mengajarkan suatu materi kepada muridnya dituntut menggunakan media sebagai pembantu sampainya materi tersebut. Media yang dipergunakan tidak harus berupa media yang mahal, melainkan media yang benar-benar efisien dan mampu manjadi alat penghubung antara seorang guru dengan murid agar materi yang diajarkan dapat diterima dan dipahami secara maksimal. Secara tersirat, kalimat diatas bermakna bahwa seorang guru hendaklah mendiskusikan dengan orang-orang yang lebih

mengetahui (dalam ayat tersebut Allah berperan sebagai Dzat yang maha mengetahui) tentang media apa yang akan digunakannya ketika ia mengajar.

Media sangat berperan penting dalam pencapaian hasil yang di harapkan. Ini terlihat secara tidak langsung dalam tafsirnya, yakni (Dia (Allah) akan memberi pahala kepada orang yang lebih benar jalannya). Dari penjelasan diatas penulis mengambil sebuah kesimpulan bahwa media yang baik dan benar akan mewakili sampainya materi yang di ajarkan, sedangkan media yang kurang tepat tidak akan mencapai hasil yang maksimal. Begitu juga diterangkan dalam surat An-Nahl 89

وَيَوْمَ نَبْعَثُ فِي كُلِّ أُمَّةٍ شَهِيدًا عَلَيْهِمْ مِّنْ أَنْفُسِهِمْ وَجِئْنَا بِكَ
 شَهِيدًا عَلَىٰ هَؤُلَاءِ وَنَزَّلْنَا عَلَيْكَ الْكِتَابَ تَبْيِينًا لِّكُلِّ شَيْءٍ
 وَهُدًى وَرَحْمَةً وَبُشْرَىٰ لِلْمُسْلِمِينَ

Artinya : “(Dan ingatlah) akan hari (ketika) Kami bangkitkan pada tiap-tiap umat seorang saksi atas mereka dari mereka sendiri dan Kami datangkan kamu (Muhammad) menjadi saksi atas seluruh umat manusia. Dan Kami turunkan kepadamu Al Kitab (Al Quran) untuk menjelaskan segala sesuatu dan petunjuk serta rahmat dan kabar gembira bagi orang-orang yang berserah diri” (QS.An-Nahl 89).

Dalam ayat ini secara tidak langsung Allah mengajarkan kepada manusia untuk menggunakan sebuah alat atau benda sebagai suatu media dalam menjelaskan segala sesuatu. Sebagaimana Allah SWT menurunkan Al Qur’an

kepada Nabi Muhammad Saw untuk menjelaskan segala sesuatu, maka sudah sepatutnya jika seorang menggunakan suatu media tertentu dalam menjelaskan segala hal.

Ayat diatas juga menjelaskan tentang bagaimana seharusnya syarat suatu media yang akan digunakan. Pada surat An Nahl ayat 89 tersebut dijelaskan bahwa Al Qur'an selain berperan untuk menjelaskan, juga merupakan sesuatu yang berfungsi sebagai petunjuk, rahmat, dan pemberi kabar gembira bagi orang yang menyerahkan diri.

Sebagaimana keterangan diatas, maka suatu media yang digunakan dalam pengajaran harus mampu menjelaskan kepada para siswa tentang materi yang sedang mereka pelajari. Syarat ini sejalan dengan esensitas sebuah media dalam pengajaran pada QS. Al Isra' : 84. Selain hal tersebut, sebuah media juga harus mampu menjadi petunjuk untuk melakukan sesuatu yang baik. Sedangkan mengenai Al Qur'an sebagai rahmat dan pemberi kabar gembira jika dikaitkan dengan masalah media dalam dunia pendidikan maka suatu media harus mampu menumbuhkan rasa gembira yang selanjutnya meningkatkan ketertarikan siswa dalam mempelajari materi-materi yang disampaikan. Hal tersebut karena tujuan pendidikan tidak hanya pada segi kognitif saja, melainkan juga harus mampu mempengaruhi sisi afektif dan psikomotor para siswa. Dalam hal ini maka media harus mampu meraih tujuan pendidikan tersebut.

Ayat 84 dalam surat Al Isra' bermakna bahwa segala perbuatan yang hendak dilakukan memerlukan adanya sebuah metode tertentu agar tujuan yang

diharapkan benar-benar dapat tercapai. Pada ayat ini juga dijelaskan bahwa media yang hendak digunakan tidak harus berupa sesuatu yang tak terjangkau oleh kemampuan orang yang bersangkutan, melainkan berupa segala sesuatu yang mudah didapatkan serta mudah dalam menggunakannya. Ayat diatas memberikan keterangan bahwa media juga bisa berasal dari lingkungan tempat seseorang berada.

2.2 Tujuan dan Manfaat Media Pembelajaran

يَهْدِي بِهِ اللَّهُ مَنِ اتَّبَعَ رِضْوَانَهُ سُبُلَ السَّلَامِ
وَيُخْرِجُهُم مِّنَ الظُّلُمَاتِ إِلَى النُّورِ بِإِذْنِهِ
وَيَهْدِيهِمْ إِلَى صِرَاطٍ مُسْتَقِيمٍ ﴿١٦﴾

Artinya : “Dengan kitab Itulah Allah menunjuki orang-orang yang mengikuti keredhaan-Nya ke jalan keselamatan, dan (dengan kitab itu pula) Allah mengeluarkan orang-orang itu dari gelap gulita kepada cahaya yang terang benderang dengan seizin-Nya, dan menunjuki mereka ke jalan yang lurus”. (QS. Al-Maidah 16).

Pada ayat diatas, Allah Swt menyebutkan tiga macam kegunaan dari Al Qur’an. Hal ini jika kita kaitkan dengan media dalam pendidikan maka kita akan mengetahui bahwa minimal ada tiga syarat yang harus dimiliki suatu media sehingga alat ataupun benda yang dimaksud dapat benar-benar digunakan sebagai media dalam pembelajaran. Tiga aspek itu adalah :

1. Bahwa media harus mampu memberikan petunjuk (pemahaman) kepada siapapun siswa yang memperhatikan penjelasan guru dan memahami medianya. Ringkasnya, media harus mampu mewakili setiap pikiran sang guru sehingga dapat lebih mudah memahami materi.
2. Dalam Tafsir Al Maraghi disebutkan bahwa Al Qur'an sebagai media yang digunakan oleh Allah akan mengeluarkan penganutnya dari kegelapan Aqidah berhala. Keterangan ini memiliki makna bahwa setiap media yang digunakan oleh seorang guru seharusnya dapat memudahkan siswa dalam memahami sesuatu.
3. Sebuah media harus mampu mengantarkan para siswanya menuju tujuan belajar mengajar serta tujuan pendidikan dalam arti lebih luas. Media yang digunakan minimal harus mencerminkan (menggambarkan) materi yang sedang diajarkan. Semisal dalam mengajarkan nama-nama benda bagi anak-anak, maka media yang digunakan harus mampu mewakili benda-benda yang dimaksud. Tidak mungkin dan tidak diperbolehkan mengajarkan kata "*Meja*" tetapi media yang digunakan adalah motor.

Selanjutnya pada ayat 16 dalam surat Al Maidah memberikan penjelasan tentang syarat suatu media. Yakni sebuah media yang digunakan oleh seorang guru harus mampu mempengaruhi berbagai aspek dalam diri siswa, yaitu kognitif, afektif dan psikomotor.

Ayat-ayat Al-Qur'an pada makalah ini memberikan gambaran tentang sebuah media yang baik. Media yang baik harus mampu mempengaruhi siswa sehingga

mereka memiliki kepribadian yang baik. Media yang digunakan seorang guru juga harus mewakili sebagian materi yang telah ia ajarkan sebelumnya serta harus mampu membangkitkan semangat para siswa sehingga mereka berkeinginan untuk memikirkan kembali pelajaran yang mereka bahas di kelas selama proses belajar.

Secara garis besar bahwa tujuan media pembelajaran sebagai alat bantu pembelajaran, adalah sebagai berikut :

- a. Mempermudah proses pembelajaran di kelas.
- b. Meningkatkan efisiensi proses pembelajaran.
- c. Menjaga relevansi antara materi pelajaran dengan tujuan belajar.
- d. Membantu konsentrasi pembelajar dalam proses pembelajaran.

Adapun manfaat media pembelajaran sebagai alat bantu dalam proses pembelajaran adalah sebagai berikut :

- a. Pengajaran lebih menarik perhatian pembelajar sehingga dapat menumbuhkan motivasi belajar.
- b. Bahan pengajaran akan lebih jelas maknanya, sehingga dapat lebih dipahami pembelajar, serta memungkinkan pembelajar menguasai tujuan pengajaran dengan baik
- c. Metode pembelajaran bervariasi, tidak semata-mata hanya komunikasi verbal melalui penuturan kata-kata lisan pengajar, pembelajar tidak bosan, dan pengajar tidak kehabisan tenaga.

- d. Pembelajar lebih banyak melakukan kegiatan belajar, sebab tidak hanya mendengarkan penjelasan dari pengajar saja, tetapi juga aktivitas lain yang dilakukan seperti mengamati, melakukan, mendemonstrasikan dan lain-lainnya (<http://der-traumer.blogspot.com/2012/09/>).

Ada beberapa manfaat media pembelajaran bagi pengajar, yaitu:

- a. Memberikan pedoman, arah untuk mencapai tujuan
- b. Menjelaskan struktur dan urutan pengajaran dengan baik
- c. Memberikan kerangka sistematis secara baik.
- d. Memudahkan kembali pengajar terhadap materi pembelajaran
- e. Membantu kecermatan, ketelitian dalam penyajian dalam pembelajaran.
- f. Membangkitkan rasa percaya diri seorang pengajar.
- g. Meningkatkan kualitas pembelajaran

Manfaat media pembelajaran bagi pembelajar, yaitu:

- a. Meningkatkan motivasi belajar pembelajar
- b. Memberikan dan meningkatkan variasi belajar pembelajar
- c. Memberikan struktur materi pelajaran
- d. Memberikan inti informasi pelajaran
- e. Merangsang pembelajar untuk berpikir dan beranalisis.
- f. Menciptakan kondisi dan situasi belajar tanpa tekanan.
- g. Pelajar dapat memahami materi pelajaran dengan sistematis yang disajikan pengajar .

3.2 Bahasa Pemrograman *Java*

Java merupakan sebuah bahasa pemrograman yang berorientasi obyek yang dikembangkan oleh *Sun Microsystem* yang digunakan untuk menghasilkan aplikasi-aplikasi *Java* yang mendefinisikan sintaks dan perilaku bahasa. Adapun pemrogramannya menggunakan kode yaitu dengan dikompilasi dan diinterpretasi.

Kompilasi kode program *Java* dilakukan menggunakan *tool command line* yang bernama *Javac*, atau biasa disebut kompiler *Java*. Tahap kompilasi ini bertujuan untuk mengkonversi kode sumber ke program biner yang berisi *bytecode*, yaitu instruksi-instruksi mesin. Kode program tidak dieksekusi di komputer secara langsung, tetapi berjalan diatas komputer hipotesis yang distandarisasikan, yang disebut *Java Virtual Machine*.

Dibandingkan dengan bahasa pemrograman yang lain, *Java* memiliki beberapa aspek keunggulan antara lain, *Java* bersifat sederhana dan relatif mudah, berorientasi pada objek, bersifat terdistribusi, *multiplatform* dan *multithred* (Benny Hermawan, 2004).

3.3 Elemen Bahasa

Elemen-elemen disetiap bahasa pemrograman sebenarnya hampir sama, meskipun demikian, ada elemen-elemen khusus yang membedakan dan sekaligus mencerminkan identitas suatu bahasa. Adapun elemen-elemen dasar bahasa pemrograman *Java* dapat diidentifikasi dengan melihat dari tipe data yang dikelompokkan menjadi dua jenis, yaitu tipe primitif dan *reference* (kelas).

Tipe primitif/dasar adalah tipe-tipe bawaan, meliputi *boolean*, *char*, *byte*, *short*, *int*, *long*, *float*, dan *double*. Sementara itu, tipe *reference* memiliki semantik seperti pointer.

3.4 Aturan Penamaan

Di dalam pemrograman, suatu nama digunakan untuk mengacu ke identitas yang dideklarasikan. Terkait hal itu, ada beberapa aturan dasar penamaan yang perlu sekali diperhatikan dalam upaya menghasilkan kode program yang *readable*. Adapun syarat penulisan identitas sebagai berikut : (Ir. Yuniar Supardi, 2005).

- a) Diawali huruf/abjad atau karakter \$ atau *underscore*/garis bawah ().
- b) Tidak boleh memakai symbol khusus, seperti @, !, *, #.
- c) Tidak boleh memakai kata kunci (*keyword/reserved word*).

c)4.1 Penamaan Paket

Nama awal paket sebaiknya terdiri atas dua atau tiga huruf kecil, dan biasanya menggunakan nama domain internet, seperti *com*, *org*, *net*, dan *edu*. Selain itu, diperkenankan memberi nama paket dengan kode-kode negara, seperti *id*, *uk*, atau *au*. Penggunaan nama domain ini bertujuan untuk mencegah terjadinya konflik paket, dengan asumsi bahwa tidak menggunakan nama domain orang lain. Sebagai contoh, nama paket berbasis domain <http://lana.indodesain.com> adalah `com.indodesain.lana`.

c)4.2 Penamaan Kelas Dan *Interface*

Nama kelas dan interface sebaiknya berupa kata atau ungkapan kata benda yang deskriptif dan tidak terlalu panjang. Penulisan nama mengacu pada sintaks pascal, dimana huruf pertama untuk setiap kata adalah huruf besar dan tidak ada spasi, misalnya Bangun, Segitiga, atau Koneksi Data.

c)4.3 Penamaan Method

Nama method seharusnya berupa kata kerja atau ungkapan kata kerja. Penulisan method mengacu pada sintaks camel, dimana huruf pertama untuk setiap kata pertama adalah huruf kecil dan huruf pertama kata selanjutnya adalah huruf besar. Method adalah prosedur atau fungsi yang dimiliki oleh suatu objek. Dalam beberapa referensi method dianalogikan juga dengan istilah prosedur, fungsi dan *behaviour*. Nama method umumnya juga mencerminkan operasi yang dilakukannya, contohnya seperti *setData*, *getData*, *isValidData*, atau *toStoring*.

c)4.4 Penamaan Variabel

Dalam beberapa beberapa referensi, variabel dianalogikan juga sebagai data, property, atau atribut. Penamaan variabel-variabel kelas (*fields*) mirip dengan penamaan method. Untuk penamaan variabel lokal dan parameter, seringkali menggunakan akronim, singkatan, atau istilah-istilah yang mudah diingat, contohnya seperti *sr* (*StreamReader*), *buf* (*buffer*), *d* (*double*), dan *s* (*String*).

c)4.5 Penamaan Konstanta

Konstanta adalah variabel yang diberi suatu nilai awal (inisialisasi) dan nilai ini selanjutnya bersifat tetap[, tidak dapat diubah oleh kelas maupun manapun. Seperti umumnya bahasa pemrograman, nama konstanta di *Java* harus berupa huruf besar semua. Apabila nama konstanta terdiri atas beberapa kata, sebaiknya pisahkan dengan tanda garis bawah “_”. Contoh penamaan konstanta misalnya MAX, MAX_DATA, atau MAX_LEN_DATA.

c).5 Objek

Objek adalah kesatuan entitas (benda), baik yang berwujud nyata ataupun hanya suatu sistem atau konsep yang memiliki sifat karakteristik dan fungsi (Benny Hermawan, 2004). Di pemrograman berorientasi objek, objek adalah entitas dasar saat runtime. Pada saat kode program dieksekusi, objek berinteraksi satu sama lain tanpa harus mengetahui detail data atau kodenya. Interaksi antara objek ini dilakukan dengan menggunakan suatu *message*. Objek memiliki suatu siklus hidup, yaitu diciptakan, dimanipulasi, dan dihancurkan.

c).5.1 Menciptakan Objek

Objek diciptakan dengan menggunakan operator *new*. Dari sisi kelas, langkah ini merupakan instantiasi kelas. Selanjutnya objek yang berhasil diciptakan tersebut akan diletakkan di memori heap. *ContohKelas ck = new ContohKelas() ;*

Dalam kasus-kasus tertentu, terkadang dapat menciptakan objek tanpa harus meng *assign* ke variabel. Langkah ini umumnya dilakukan apabila tidak

memerlukan referensi ke objek tersebut. Sebagai contoh jika memiliki method yang menerima argumen berupa objek `ContohKelas`, maka dapat dituliskan seperti berikut : `getData (new ContohKelas ())` ;

c).5.2 Memeriksa Tipe Objek

Memanfaatkan fungsionalitas operator `instanceof` untuk mengetahui tipe suatu objek pada saat runtime. Operator ini akan mengembalikan nilai `true` apabila tipe objek sesuai, sebaiknya mengembalikan nilai `false`.

Perlu diperhatikan, `instanceof` akan selalu mengembalikan nilai `false` jika variabel objek diinisialisasi dengan nilai `null`, karena nilai `null` tidak mencerminkan objek apapun. Operator `instanceof` hanya dapat digunakan pada tipe `reference` dan objek. Penggunaan operator ini pada tipe primitif akan mengakibatkan kesalahan saat kompilasi.

c).5.3 Menghapus Objek

`Java` menggunakan teknik yang dikenal sebagai `garbage collection` untuk menghapus objek-objek yang sudah tidak diperlukan. Dengan demikian, tidak perlu khawatir akan terjadinya kebocoran memori. Dalam praktiknya, `garbage collector` mampu mengidentifikasi kapan suatu objek dialokasikan dan kapan digunakan lagi. `Garbage collector` melakukan tugasnya secara tak sinkron berdasarkan ketersediaan sumber daya. Normalnya, jika suatu objek sudah tidak diacu (di-refer), maka ia akan segera dibersihkan. Terlepas dari mekanisme normal ini, dapat juga memanggil `garbage collector` secara eksplisit menggunakan `method statis gc`.

c).6 Teknik Dasar *Java*

Teknik-teknik dasar yang dimaksudkan disini adalah materi-materi dasar yang dipandang penting untuk diketahui dan frekuensi penggunaannya relatif tinggi. Pada kenyataannya banyak teknik yang terkait dengan operasi atau aktivitas tertentu yang akan menguraikan secara spesifik mencerminkan perilaku dan fitur *Java*.

c).6.1 Deklarasi Konstanta

Pada saat memerlukan suatu konstanta, deklarasi variabel dengan modifier *static* dan *final* serta konstanta bisa dipanggil langsung melalui namanya. Apabila konstanta ingin dapat diakses oleh class yang lain tanpa harus membuat objek baru, pakai modifier publik dan keyword *static*.

c).6.2 Komparasi Integer

Di bahasa C sering memperlakukan nilai integer sebagai *boolean* karena hal ini memang diperbolehkan. Artinya bisa mengindikasikan nilai 1 sebagai *true* dan 0 sebagai *false*. Namun tidak demikian di *Java* karena nilai integer tidak boleh diperbandingkan layaknya *boolean*.

c).6.3 Keluar dari Blok

Pernyataan *break* dapat digunakan untuk keluar dari blok, tidak hanya dari *switch*. Pernyataan ini terdiri atas dua jenis, yaitu *break* berlabel dan *break* tak berlabel. Dalam penerapannya, pernyataan *break* mengakibatkan *interpreter*

sesegera mungkin melompat ke akhir blok dimana pernyataan tersebut dinyatakan.

c).6.4 Menghentikan Program

Tidak seperti *return* yang menghentikan eksekusi method dan kembali ke pemanggil, untuk menghentikan program menggunakan *method exit*. Argumen yang diperlukan oleh method ini adalah int status, yang menyatakan kode kesalahan. Jika akan mencoba mengganti *exit* dengan *return*, kompiler akan menampilkan kesalahan dan menunjuk baris setelah *return*. Pesan yang ditampilkan tersebut mengindikasikan bahwa baris tidak tercapai atau tereksekusi.

c).7 Operasi Karakter

Ada dua jenis tipe data yang bisa digunakan karena bekerja dengan karakter, yaitu *char* (tipe primitive) dan *character* (tipe reference). Untuk memilih tipe yang sesuai, bisa didasarkan pada kebutuhan yang diperlukan. Apabil memerlukan suatu objek dan operasi-operasi terkait, gunakan objek *character*, dan sebaliknya manfaatkan tipe data *primitive char*. Aturan dasar inisialisasi nilai variable karekter adalah mengapit nilai karakter dengan tanda kutip tunggal (bukan tanda kutip ganda).

c).7.1 Memeriksa Jenis Karakter

Untuk mengidentifikasi jenis suatu karakter, bisa memanfaatkan method *isLetter*, *isDigit*, atau *isLetterOrDigit* yang didefinisikan di kelas *Character*. Ketiga method ini dapat membantu untuk mengetahui apakah suatu karakter merupakan huruf, angka, ataukah salah satu atau keduanya.

```
// Mengembalikan true jika karakter berupa huruf
system.out.println (Character.isLetter('a')) ;

// Mengembalikan true jika karakter berupa angka
system.out.println (Character.isDigit ('3')) ;

// Mengembalikan true jika karakter berupa huruf atau angka
system.out.println (Character.isLetterOrDigit ('a')) ;

// Mengembalikan true jika karakter berupa spasi
system.out.println (Character.isWhitespace (' ')) ;
```

c).7.2 Manipulasi Karakter

Apabila ingin mengetahui apakah suatu karakter merupakan karakter dengan huruf kecil atau besar, gunakan method *isLowerCase* dan *isUpperCase*. Selanjutnya, jika diperlukan, dapat mengonversi karakter dengan memanfaatkan method *toLowerCase* dan *toUpperCase*.

```
Char ch = 'a' ;

// Jika karakter adalah huruf kecil, ubah ke huruf besar
If (Character.isLowerCase (ch)) {

    System.out.println(Character.toUpperCase (ch)) ;

} else {
```

```

        System.out.println(Character.toLowerCase(ch) );
    }

```

c).7.3 Komparasi Karakter

Kelas *Character* mengidentifikasi *method compareTo* yang berfungsi untuk membandingkan dua objek *Character* dan mengembalikan nilai integer. Method ini melakukan komparasi secara numeric, dimana nilai 0 menyatakan nilai objek sama.

```

Character c = new Character ('c') ;
Character b = new Character ('b') ;
Int diff = c.compareTo (b) ;
If (diff == 0) {
    System.out.println("c = b") ;
} else if (diff < 0) {
    System.out.println("c < b") ;
} else if (diff > 0) {
    System.out.println("c > b") ;
}

```

Sebagai alternatif juga bisa memanfaatkan *method equals* untuk membandingkan nilai karakter dan mendapatkan nilai kembalian *Boolean*.

Dalam operasi data, sering kali perlu mengonversi suatu nilai ke nilai dengan tipe data berbeda.

c).8 Konversi dan Casting

Dalam operasi data, sering kali perlu mengonversi suatu nilai ke nilai dengan tipe data berbeda. Salah satu contoh yang sering ditemui adalah ketika ingin mengkalkulasi operasi bertipe *string* dan menyimpannya sebagai integer. Bagaimanapun juga, tipe integer tidak memperbolehkan kalkulasi dalam bentuk *string*, sehingga perlu mengonversinya.

Untuk lebih jelasnya, tabel berikut memperlihatkan konversi tipe data *primitive* ke dan dari *string*.

Tabel 2.1 Tabel konversi string

Tipe	Ke String	Dari String
<i>boolean</i>	<i>String.valueOf(boolean)</i>	<i>Boolean.parseBoolean(String)</i>
<i>byte</i>	<i>String.valueOf(byte)</i>	
<i>char</i>	<i>String.valueOf(char)</i>	
<i>double</i>	<i>String.valueOf(double)</i>	
<i>float</i>	<i>String.valueOf(float)</i>	
<i>int</i>	<i>String.valueOf(int)</i>	
<i>long</i>	<i>String.valueOf(long)</i>	
<i>short</i>	<i>String.valueOf(short)</i>	

Sementara itu, untuk mengembalikan representasi *string* dari tipe-tipe *primitive*, dapat memanfaatkan method *valueOf.Method* ini sebenarnya juga identik dengan method *toString* yang didefinisikan di setiap kelas pembungkus tipe data *primitive*.

```
double d1 = 3, 5;
```

```
Double d2 = d1;
```

```
// Mengembalikan representasi string
system.out.println(String.valueOf(d2)) ;
system.out.println(d2.toString()) ;
```

Java juga mendukung konversi otomatis, yakni konversi yang dilakukan tanpa intervensi pemrograman untuk menspesifikasikan tipe data. Konversi ini terjadi apabila tipe datanya kompatibel dan range dari tipe tujuan lebih besar dibanding tipe asal.

```
// Konversi otomatis byte ke int
Byte b1 = 13;
Int i1 = b1;
```

Jenis konversi lainnya adalah konversi casting, atau lebih sering disebut *casting* saja. *Casting* adalah konversi suatu tipe ke tipe yang dispesifikasikan secara eksplisit menggunakan operator `cast` “()”.

Contoh *casting* diperlihatkan seperti berikut :

```
// Casting int ke byte
int i2 = 32;
byte b2 = (byte) i2;
```

Data dari variable `i2` tidak diubah secara permanen. Dalam implementasinya, *Java* menciptakan salinan sementara untuk nilai `i2` dan menetapkan tipe datanya sebagai *byte*. Pada umumnya, *casting* dilakukan terhadap dua tipe yang tidak kompatibel.

Contoh kasus lain yang juga memerlukan keterlibatan *casting* adalah konversi bilangan desimal ke kode ASCII.

```
public static Character toAsciiChar (int i) {
    return new Character ((char)i) ;
}

System.out.println(toAsciiChar(65)) ;
// Output: A
```

Sebaliknya, jika ingin mendapatkan bilangan desimal yang merepresentasikan kode ASCII, cukup meng-*casting* nilai karakter ke tipe integer.

```
char ch = "A"
System.out.println((int)ch) ;
// Output: 65
```

c).9 Membuat *Window Swing*

Untuk menetapkan judul *window*, dapat dengan memanggil konstruktor *JFrame* dengan *argument string*. Pemanggilan ini dilakukan dengan memanfaatkan *keyword super*, selain itu juga diperkenankan memanfaatkan *method setTitle*. Adapun pembuatan *window* yang sederhana minimal adalah sebagai berikut :

```
Super ("Judul") ;

// atau dengan method setTitle

setTitle("Judul") ;
```

Ukuran awal yang ditetapkan pada suatu *window* adalah 123x34 (lebar x tinggi) dalam suatu *pixel* (*picture element*). Ukuran ini tentu tidak cukup untuk merepresentasikan sebuah *window*. Adapun untuk menetapkannya perlu menggunakan *method setSize*. Apabila menginginkan agar *window* menetapkan sendiri ukurannya ke ukuran minimum yang mampu menampung semua kompoen di dalamnya, gunakan *method pack*.

Apabila menggunakan *keyword* *super*, pastikan bahwa pemanggilan *keyword* ini dilakukan paling awal.

```

Import Javax.swing.JFrame;

public class MyFrame extends JFrame {

    public MyFrame () {

        // Menetapkan title window

        super ("Judul") ;

        // Menetapkan ukuran window (300x150 pixel)

        setSize(300, 150) ;

        // Menghentikan aplikasi ketika window ditutup

        setDefaultCloseOperation(EXIT_ON_CLOSE) ;

    }

```

```
public static void main(String[ ] arg) {  
  
    // Menciptakan objek MyFrame  
  
    MyFrame frm = new Myframe() ;  
  
    // Menetapkan visibilitas  
  
    frm.setVisible(true) ;  
  
    // Menetapkan background merah di container  
  
    getContentPane().setBackground(Color.RED) ;  
}
```

c).10 Menggunakan Button

Untuk menciptakan sebuah *button* (tombol), dapat dengan menginstantiasi kelas yang memperluas kelas *Javax.swing.AbstracButton*. Hal ini merupakan abstrak dasar bagi semua komponen button, seperti *JButton* dan *JtoggleButton*.

c).11 Mengolah Data Gambar

Apabila menginginkan masukan data BLOB, seperti gambar, dapat menggunakan fitur-fitur yang disediakan JDB. Adapun untuk efisiensi pengolahan data gambar, sebaiknya memanfaatkan objek *preparedStatement*.

```

import java.sql.*;

import java.io.*;

public class DataGambarPrepared {

public static void main(String[] args) {

connection conn = null;

preparedStatement pstmt = null;

try (

class.forName("com.mysql.jdbc.Driver");

conn = DriverManager.getConnection(

        "jdbc:mysql://localhost/myjdbc", "maulana", "maulana");

string prep = "INSERT INTO foto(id, foto) " +

        "VALUES(?, ?)";

// Misal file gambar yang akan ditambahkan

file fl = new File("D:/Java/foto/haris.png");

fileInputStream fis = new FileInputStream(fl);

// Menciptakan objek PreparedStatement

pstmt = conn.prepareStatement

// Menetapkan parameter id foto

```

```

pstmt.setInt(1, 1);

// Menetapkan parameter untuk objek Input stream

pstmt.setBinaryStream(2, fis, (int) fl.length());

// Eksekusi preped statement

Int n = pstmt.executeUpdate();

System.out.println (n+ "row affected");

Fis.close ();

} catch(SQLException ex) {

    System.out.println (ex.getMessage());

} catch (Exception ex) {

System.out.println(ex.getMessage());

} finally {

    Try {

        Pstmt.close();

        Conn.close();

    } catch (Exception ex) {

System.out.println(ex.getMessage());

    }

}

}

```

c).12 Menampilkan Data Gambar

Untuk bisa menampilkan gambar yang tersimpan di database melalui berbagai model, tahap pengambilan data gambar dapat menggunakan pendekatan yang sama, yakni melalui method *getBytes*. Method yang didefinisikan di kelas *ResultSet* ini akan mengembalikan *array byte* data gambar. Selanjutnya bisa memanfaatkan objek *ImageIcon* guna men-generate gambar kemudian menetapkannya sebagai *icon* label.

c).13 Menjalankan Aplikasi

Java memungkinkan untuk memanggil program (aplikasi) lain dan menjalankannya dengan memanfaatkan *method exec* yang didefinisikan di kelas *Java.lang.Runtime*. Penggunaan *method exec* yang paling sederhana adalah dengan mengirimkan argument berupa string perintah atau nama program. Apabila diperlukan, string perintah bisa berisikan path ke *file executable*. Adapun jika eksekusi berhasil dilaksanakan, *method exec* akan mengembalikan objek *Process*.

Tak hanya sekedar menjalankan program-program eksternal, program yang dibuat juga memungkinkan untuk menjalankan program *Java* yang lainnya. Sebenarnya tidak sulit untuk membuat program yang mampu menjalankan program lain meski lokasinya berbeda. Kuncinya adalah dengan menambahkan

path proram target ke *variable classpath*. Untuk lebih praktisnya bisa menggunakan opsi `-cp` atau `-classpath` pada saat menjalankan program.

```
import Java.io.File;

public class RunJava {

    /**
     *Menjalankan program Java dengan objek Runtime
     *Asumsi: File ditemukan
     *@param File fl Objek File (Path+Nama file)
     */

    private static void runJava (File fl) {
        string namaFile = s.contains(".") ?

        // Mendapatkan nama file tanpa ekstensi
        string namaFile = s.contains(".") ?

        s.substring(0, s.indexOf('.') : s;

        // Membentuk opsi -cp dan nama file .class-nya
        string arg = "-cp \" " + fl.getParent () +

        system.getProperty(path.separator") +

        system.getProperty("Java.class.path") +

        "\" \" " + namaFile ;

    try {

        // Simulasi interpreter Java

        runtime.getRuntime ().exec("Java " + arg) ;
```

```

    } catch (Java.io.IOException ex) {
system.out.println(ex.getMessage() );
    }
}
/**
 *Menjalankan program Java dengan objek processBuilder
 *Asumsi: File ditemukan
 *@param File fl Objek File (Path+Nama file)
 */
private static void runJavaPB (File fi) {
    String s = fl.getName() ;
    String namaFile = s.contains(".") ?
s.substring(0, s.indexOf(".")) : s;
    String arg = fl.getParent () +
System.getProperty("path.separator") +
System.getProperty("Java.class.path") ;
    try {
new ProcessBuilder("Java", "-cp", arg,
namaFile).start() ;
    } catch (Java.io.IOException x) {
System.out.println(ex.getMessage() );
    }
}

```

```
}
```

c).14 Memainkan Audio

Paket *Java.applet* mendefinisikan *interface audioclip* yang berfungsi untuk memainkan klip audio. Interface ini sebelumnya hanya dapat digunakan pada aplikasi-aplikasi applet. *Interface AudioClip* menyediakan *method play, loop, dan stop*, yang berfungsi untuk mengontrol klip audio. Sebelum dapat melakukan pengontrolan, terlebih dahulu ciptakan objek *audioclip* melalui method statis *newaudioclip* yang didefinisikan di kelas *Java.applet.applet*.

```
import Java.applet.* ;
import Java.awt.* ;
import Java.awt.event.* ;
import Java.swing.* ;
import Java.net.* ;

public class Audioplayer extends JFrame {

    private JButton btnPlay, btnLoop, btnStop ;

    private AudioClip audio ;

    public AudioPlayer (string sURL) {

        super ("AudioPlayer") ;

        try {

            URL ur1 = new URL (sURL) ;

            // Mendapatkan audio dari URL
```

```

        Audio = Applet.newAudioClip (url) ;
    } catch (MalformedURLException ex) {
        System.err.println(ex.getMessage() ) ;
        System.exit (0) ;
    }

    Jpanel pln = new Jpanel ( ) ;
    btnPlay = new Jbutton ("Play") ;
    btnLoop = new Jbutton ("Loop") ;
    btnStop = new Jbutton ("Stop") ;
    pn1.add(btnPlay) ;
    pn1.add(btnLoop) ;
    pn1.add(btnStop) ;
    ButtonHandler hnd = new ButtonHandler() ;
    btnPlay.addActionListener(hnd)
    btnLoop.addActionListener(hnd)
    btnStop.addActionListener(hnd)
    add(pln, "North") ;
    setDefaultCloseOperation(Exit_On_Close) ;
    pack() ;
    setVisible(true)
}

// Listener untuk mengontrol klip audio

```

Private class ButtonHandler implements ActionListener {

Public void actionPerformed(ActionEvent e) {

if (e.getSource() == btnPlay) {

// Memainkan klip audio sekali

Audio.play();

} else if (e.getSource() == btnLoop) {

// Memainkan klip terus-menerus

Audio.loop();

} else if (e.getSource() == btnStop) {

// Menghentikan klip

Audio.stop();

}

}

}

c).15 **Sistem Pakar**

Sistem pakar adalah bagian dari kecerdasan buatan yang berisi kombinasi pemahaman teoritis tentang suatu persoalan dan sekumpulan aturan pemecahan persoalan heuristik yang dikembangkan oleh manusia untuk dapat memecahkan problem pada suatu domain yang spesifik.

Sistem pakar merupakan suatu perangkat lunak yang digunakan untuk memecahkan masalah yang biasanya diselesaikan oleh seorang pakar. Aturan

aturan di dalamnya memberitahu program, bagaimana ia memberlakukan informasi-informasi yang tersimpan. Berdasarkan program itu memberikan solusi-solusi atau bantuan pengambil keputusan mengenai permasalahan tertentu, mirip dengan seorang pakar.

Karakteristik umum yang membedakan sistem pakar dengan perangkat lunak biasa adalah :

- a. Terdapat banyak kemungkinan jawaban
- b. Data kabur
- c. Heuristik
- d. Fasilitas Informasi

Sedangkan sistem pakar bisa disebut mempunyai sifat yang ideal bila mempunyai ciri-ciri:

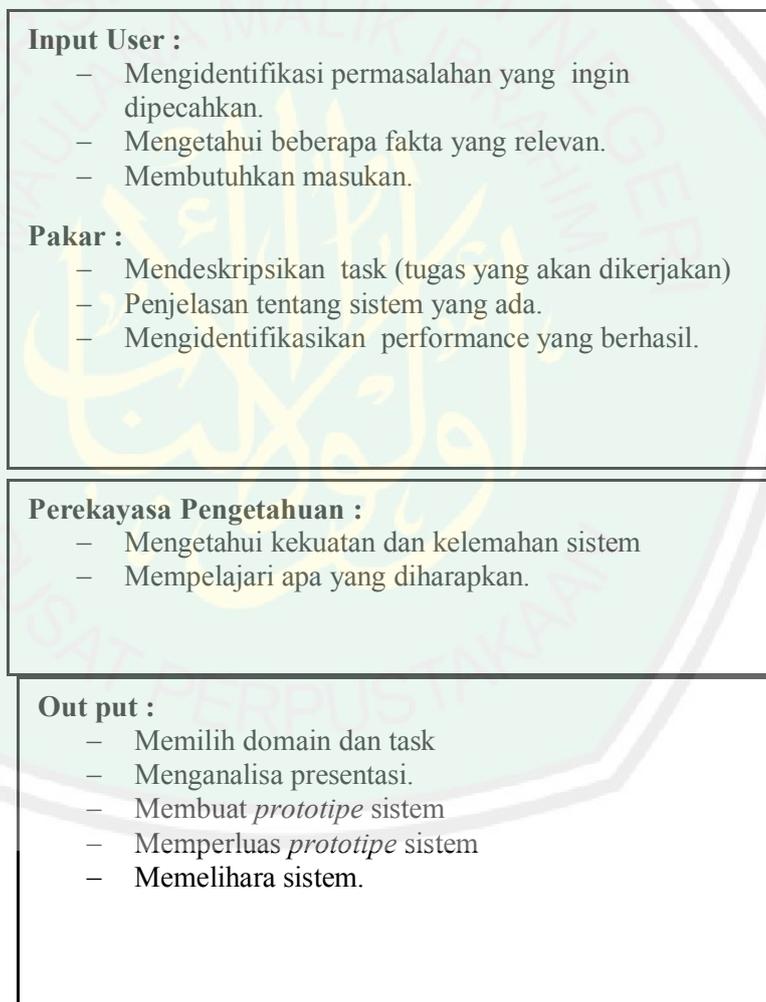
- a. Terbuka untuk diperiksa
- b. Mudah dimodifikasi
- c. Fasilitas penalaran/penjelasan.

c..15.1 Pendekatan Inferensi Dalam Sistem Pakar

Mekanisme inferensi dalam sistem pakar menerapkan pengetahuan untuk solusi problem, yang sebenarnya. Mekanisme inferensi merupakan pusat dari kemampuan untuk belajar dari pengalaman (proses belajar) sebab memungkinkan untuk men-generate fakta baru. Dan dapat juga melakukan verifikasi terhadap data-data yang ada, dengan melakukan deteksi terjadinya kesalahan dalam alur berpikir dan menuntun user untuk memodifikasi aturan-aturan yang digunakan

untuk mendapatkan goal. Penyusunan suatu mekanisme inferensi dengan pendekatan tertentu, melibatkan konversi dari representasi pengetahuan yang ada ke dalam bentuk pendekatan mekanisme inferensi yang dipilih. Misal konvensi pohon keputusan (decision tree) ke dalam bentuk aturan IF-THEN.

Ada beberapa pendekatan dalam menyusun struktur mekanisme inferensi



Gambar 2.1 Perekayasa Pengetahuan Dengan Input Output Sistem Pakar.

c.15.2 Teknik Heuristik

Agar dapat memecahkan problema yang sangat sulit, sering kali harus dilakukan kompromi terhadap kecepatan dan sistematika serta membuat struktur pengendalian yang tidak menjamin didapatkannya jawaban yang terbaik maupun optimal. Maka dipakailah teknik heuristik yaitu sebuah teknik yang mengembangkan efisiensi, namun dengan kemungkinan mengorbankan kelengkapan. Peran seorang pakar disini sangat diperlukan dalam memberikan suatu saran tentang sifat heuristik yang dipakai dalam pemecahan suatu problema.

Heuristik menggunakan pengetahuan tentang sifat problema untuk mendapatkan solusi secara efisien.

c..15.3 Inferensi Berbasis Aturan

Metode inferensi adalah program komputer yang memberikan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam *workplace*, dan untuk memformulasikan kesimpulan (Feri Fahrur Rohman; dan , Ami Fauziah, 1995).

Aturan diekspresikan dalam bentuk kondisi *IF-THEN*. *IF* adalah kondisi yang telah ada, *THEN* adalah aksi atau tanggapan lain yang akan timbul. Aturan *IF-THEN* lebih dekat dengan cara manusia memecahkan masalahnya sehari-hari, daripada program yang menyatakan pengetahuannya dalam kode komputer tingkat rendah (*low level*).

Kebanyakan sistem pakar berbasis aturan menggunakan strategi inferensi yang dinamakan modus ponens. Berdasarkan strategi ini, jika terdapat aturan “IF A THEN B”, dan jika diketahui bahwa A benar, maka dapat disimpulkan bahwa B

juga benar. Strategi inferensi modus ponens dinyatakan dalam bentuk: $[A \text{ And } (A \rightarrow B)] \rightarrow B$ dengan A dan $A \rightarrow B$ adalah proposisi-proposisi dalam basis pengetahuan.

Terdapat dua pendekatan dalam menyusun mekanisme inferensi berbasis aturan, yaitu *Backward Chaining* dan *Forward Chaining*.

2.16.4 Pelacakan ke Belakang (*Backward Chaining*)

Pelacakan ke belakang adalah pendekatan yang dimotori oleh tujuan (*goal-driven*). Dalam pendekatan ini pelacakan dimulai dari tujuan, selanjutnya dicari aturan yang memiliki tujuan tersebut untuk kesimpulannya. Selanjutnya proses pelacakan menggunakan premis untuk aturan tersebut sebagai tujuan baru dan mencari aturan lain dengan tujuan baru sebagai kesimpulannya. Proses berlanjut sampai semua kemungkinan ditemukan (Kusumadewi, 2003).

c..15.5 Pelacakan ke Depan (*Forward Chaining*)

Pelacakan kedepan adalah pendekatan yang dimotori data (*data-driven*). Dalam pendekatan ini pelacakan dimulai dari informasi masukan, dan selanjutnya mencoba menggambarkan kesimpulan. Pelacakan ke depan, mencari fakta yang sesuai dengan bagian *IF* dari aturan *IF-THEN*.

Forward chaining merupakan metode inferensi yang melakukan penalaran dari suatu masalah kepada solusinya. Jika klausa premis sesuai dengan situasi (bernilai *TRUE*), maka proses akan menyatakan konklusi. *Forward chaining* adalah data-driven karena inferensi dimulai dengan informasi yang tersedia dan baru

konklusi diperoleh. Jika suatu aplikasi menghasilkan *tree* yang lebar dan tidak dalam, maka gunakan *forward chaining*.

Contoh : Terdapat 10 aturan yang tersimpan dalam basis pengetahuan yaitu :

R1 : if A and B then C

R2 : if C then D

R3 : if A and E then F

R4 : if A then G

R5 : if F and G then D

R6 : if G and E then H

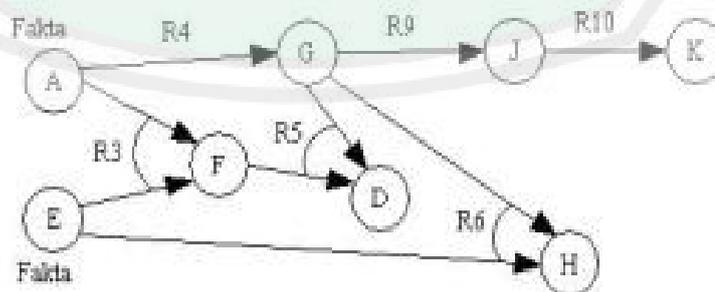
R7 : if C and H then I

R8 : if I and A then J

R9 : if G then J

R10 : if J then K

Fakta awal yang diberikan hanya A dan E, ingin membuktikan apakah K bernilai benar. Proses penalaran *forward chaining* terlihat pada gambar dibawah :



Gambar 2.2 *Forward Chaining*





BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Sebelum memulai dengan langkah-langkah instalasi *Java*, perlu diluruskan dulu bahwa *Java* sebenarnya terdiri dari atas 2 pengertian, yaitu :

- a. Teknologi *Java* sebagai suatu platform, yang biasa disebut juga sebagai *Java Runtime Environment* (JRE). Platform ini diperlukan sebagai lapisan atau *layer* untuk menjalankan program *Java* yang telah dikompilasi (*bytecode*) untuk melakukan kompilasi memerlukan *tools* tambahan yang ditambahkan ke dalam JRE dan dikenal sebagai *Java Development Kit* (JDK). Jadi di dalam JDK sendiri secara otomatis sudah terdapat platform JRE.
- b. Bahasa programming *Java* inilah yang mungkin dirujuk sebagai "*Java*" secara umum. *Java* merupakan bahasa *Object Oriented Programming* yang bisa diklaim telah diimplementasikan secara penuh . OOP di *Java* merupakan pengembangan lebih lanjut dari dari bahasa C++.

Adapun kelebihan atau keunggulan *Gel* antara lain, sebagai berikut :

- a. *Gel* penggunaanya lebih ringan dibandingkan editor lain misalnya netbean.
- b. Pada *Gel* pesan kesalahan saat pengetikan kode program (pengimplementasian dalam bahasa pemrograman) tidak akan ditampilkan

sebelum program dikompilasikan. Ini akan sangat membantu para programmer agar lebih teliti lagi dalam mengetikkan kode program.

- c. Saat mengetikkan kode program (misal System) maka secara otomatis dilayar akan muncul *list* (daftar) kode program sehingga tinggal memilih pilihan yang ada di *list* (daftar) tersebut (misal *Out*), dengan demikian kita tidak perlu repot – repot mengisi kode program secara manual.

3.1.1 Alat dan Bahan Penelitian

Bahan atau materi penelitian yang digunakan adalah data-data yang berkaitan dengan materi untuk menampilkan gambar dan suara dari berbagai jenis hewan.

3.1.2 Alat Pengumpul Data

Dalam upaya pengumpulan data, penulis menggunakan metode penelitian kualitatif, dan dalam melakukan penelitian, penulis menggunakan beberapa teknik pengumpulan data, yaitu :

- a. Studi Lapangan

Hal pertama yang dilakukan adalah melihat atau meninjau tempat yang akan dijadikan penelitian .

- b. Studi Pustaka / Literatur

Pengumpulan data yang dilakukan dengan membaca buku, *browsing* di internet maupun tulisan yang berkaitan dengan penulisan.

- c. Studi Wawancara

Mengadakan wawancara kepada orang lain atau anak-anak yang terlibat langsung dalam percobaan aplikasi tersebut.

d. Kebutuhan Hardware dan Software

Adapun kebutuhan hardware dan software mulai dari penelitian sampai dengan tahap implementasi adalah sebagai berikut :

Hardware :

- Sebuah laptop untuk pembuatan serta pengujian aplikasi dengan spesifikasi :
 - o Processor Core 2 duo atau spesifikasi di atasnya,
 - o Memory 1 GHz
 - o HDD 80 GB

Software :

- Aplikasi yang dibutuhkan antara lain :
 - o Windows 7 untuk aplikasi data
 - o Java untuk menyimpan data
 - o Gel untuk aplikasi mengelola dan editor data
 - o Adobe Photoshop 7.0 membuat tampilan gambar

3.2 Waktu dan Tempat Penelitian

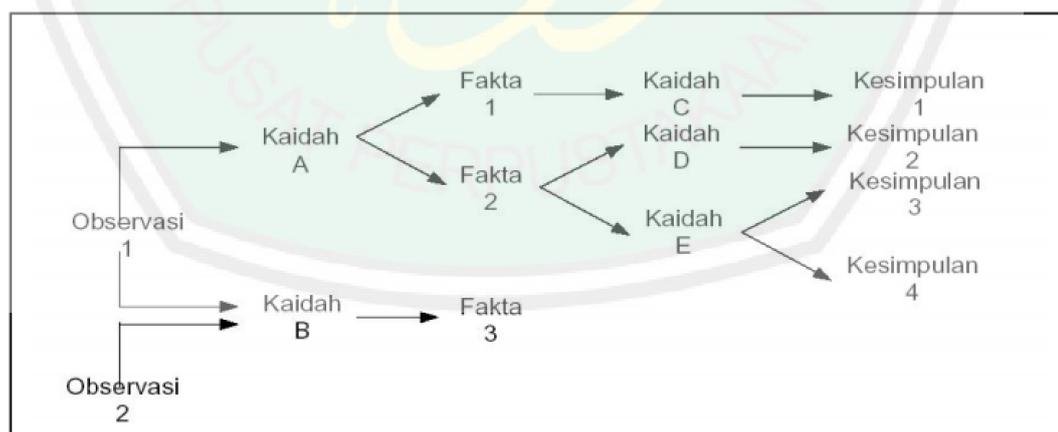
Sampai saat ini belum ada inovasi terbaru yang berkaitan dengan aplikasi metode pembelajaran yang sederhana yang dapat membantu di dalam proses belajar mengajar di sekolah dasar.

Dengan adanya implementasi sistem aplikasi metode pembelajaran mengenal hewan diharapkan proses belajar jadi menyenangkan, mudah dan cepat dimengerti, dan lebih bervariasi di Sekolah Dasar Negeri Kalirejo 1 Kecamatan Dringu Kabupaten Probolinggo.

3.3 Rancangan Sistem *Forward Chaining*

Metode yang dipakai pada penelitian ini adalah *forward chaining*. *Forward chaining* adalah strategi untuk memprediksi atau mencari solusi dari suatu masalah yang dimulai dengan sekumpulan fakta yang diketahui, kemudian menurunkan fakta baru berdasarkan aturan yang premisnya cocok dengan fakta yang diketahui. Proses dilanjutkan sampai dengan mencapai goal atau tidak ada lagi aturan yang premisnya cocok dengan fakta yang diketahui.

Berikut adalah diagram *Forward Chaining* secara umum untuk menghasilkan sebuah *goal* :



Gambar 3.1 Proses *Forward Chaining*

Langkah-langkah yang harus dilakukan dalam membuat sistem *forward chaining* berbasis aturan, yaitu:

1. Pendefinisian Masalah.

Tahap ini meliputi pemilihan domain masalah dan akuisisi pengetahuan.

2. Pendefinisian Data Input.

Sistem *forward chaining* memerlukan data awal untuk memulai inferensi.

3. Pendefinisian Struktur Pengendalian Data.

Aplikasi yang kompleks memerlukan premis tambahan untuk membantu mengendalikan pengaktifan suatu aturan.

4. Penulisan Kode Awal.

Tahap ini berguna untuk menentukan apakah sistem telah menangkap domain pengetahuan secara efektif dalam struktur aturan yang baik.

5. Pengujian Sistem.

Pengujian sistem dilakukan dengan beberapa aturan untuk menguji sejauh mana sistem berjalan dengan benar.

6. Perancangan Antarmuka.

Antarmuka adalah salah satu komponen penting dari suatu sistem.

Perancangan antarmuka dibuat bersama-sama dengan pembuatan basis pengetahuan.

7. Pengembangan Sistem.

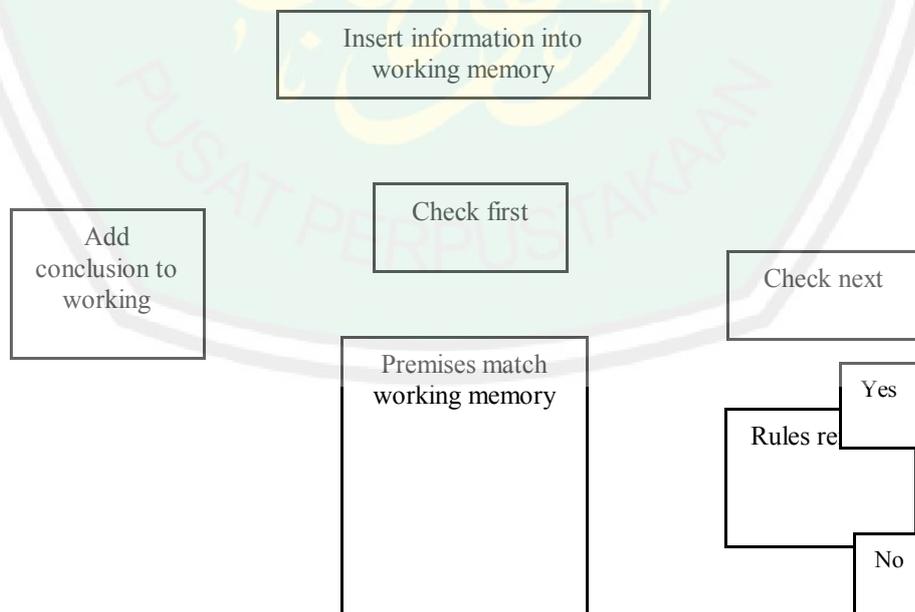
Pengembangan sistem meliputi penambahan antarmuka dan pengetahuan sesuai dengan *prototype* sistem.

8. Evaluasi Sistem.

Pada tahap ini dilakukan pengujian sistem dengan masalah yang sebenarnya. Jika sistem belum berjalan dengan baik maka akan dilakukan pengembangan kembali.

3.4 Alur Program

Dibawah ini akan digambarkan dari sebuah operasi metode *forward chaining*:



Gambar 3.2 Alur Program





BAB IV

HASIL DAN PEMBAHASAN

1.1 Rancang Bangun Berdasarkan *Forward Chaining*

Langkah-langkah yang harus dilakukan dalam membuat sistem *forward chaining* berbasis aturan, yaitu:

a. Pendefinisian Masalah.

Tahap ini meliputi pemilihan domain masalah dan akuisisi pengetahuan, antara lain adalah pemahaman siswa dalam menyerap ilmu pengetahuan dan cara penyampaian materi dengan menggunakan metode ceramah.

b. Pendefinisian Data *Input*.

Data awal untuk memulai inferensi adalah data-data yang nantinya dapat dijadikan referensi atau aplikasi yang dibuat melalui media pembelajaran yang lebih mudah dan menyenangkan, sehingga dapat menambah minat belajar.

c. Pendefinisian Struktur Pengendalian Data.

Aplikasi untuk pembuatan media pembelajaran tersebut dibuat dengan menggunakan *Java* yang disertai editor *GEL* sebagai media penunjangnya.

d. Penulisan Kode Awal.

Tahap ini berguna untuk menentukan apakah sistem telah menangkap domain pengetahuan secara efektif dalam struktur aturan mengenai hewan-hewan

yang mana penjelasannya meliputi ; nama hewan, suara hewan dan informasi lain tentang hewan-hewan tersebut.

e. Pengujian Sistem.

Aplikasi yang dijalankan dimulai dengan mencari nama atau jenis hewan yang dikehendaki dengan memakai *software java* yang menggunakan metode *forward chaining*, kemudian akan tampil berbagai informasi mengenai hewan tersebut.

f. Perancangan Antarmuka

Berdasarkan hasil pengamatan, antar muka yang diinginkan harus dengan tampilan sebaik mungkin, sehingga ramah bagi pengguna artinya pengguna dapat menggunakan perangkat lunak yang dibuat dengan senyaman mungkin dan mengurangi kesalahan dalam memasukkan data, proses maupun keluarannya.

g. Pengembangan Sistem.

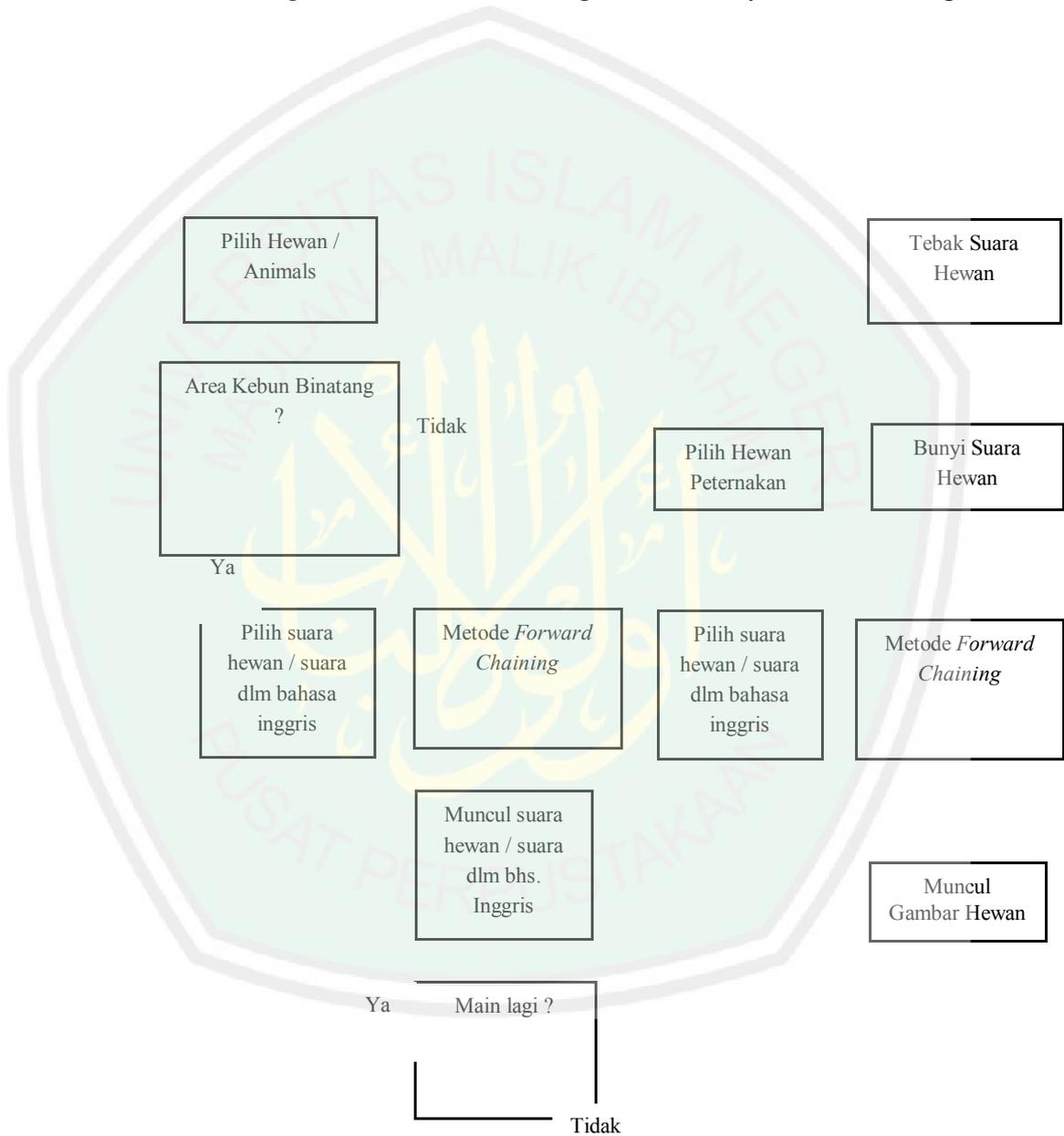
Pengembangan sistem menggunakan *Java* yang disertai editor *GEL* sebagai media penunjangnya yang dilengkapi dengan aplikasi audio untuk mengeluarkan suara.

h. Evaluasi Sistem.

Implementasi aplikasi hasil rancang bangun media pembelajaran dilakukan pada anak-anak di sekolah dasar guna mendapatkan hasil yang diharapkan.

i. Alur Program

Dibawah ini akan digambarkan dari sebuah operasi metode *forward chaining*:



Gambar 4.1 Alur Program Menggunakan Metode *Forward Chaining*

Hasil pembahasan ingin menemukan suara hewan yang sesuai dengan jenisnya. Adapun variabel-variabel yang digunakan:

A = Suara hewan

B = Nama hewan dalam bahasa Inggris

C = Hewan Peternakan atau hewan di kebun binatang

D = Area peternakan

E = Area kebun binatang

F = Suara hewan dan nama hewan dalam bahasa Inggris

G = Tebak suara

Setiap variabel dapat bernilai *TRUE* atau *FALSE*

Fakta yang ada :

§ Diasumsikan si user memiliki data:

o Jenis hewan peternakan (A TRUE)

o Jenis hewan yang ada di kebun binatang (B TRUE)

§ Ingin mencari suara dan nama hewan apakah yang tepat pada gambar yang tampak dalam komputer ?

RULES :

R1 : IF mencari suara hewan AND hewan peternakan atau hewan di kebun binatang THEN dapat mencari di area kebun binatang.

R2 : IF berada di area peternakan AND mencari hewan peternakan atau hewan di kebun binatang THEN dia dapat menemukan suara hewan dan nama hewan dalam bahasa inggris.

R3 : IF mencari nama hewan dalam bahasa inggris AND berada di area kebun binatang THEN menemukan suara hewan dan nama hewan dalam bahasa inggris

R4 : IF mencari nama hewan dalam bahasa inggris THEN dapat ditemukan di hewan peternakan atau hewan di kebun binatang.

R5 : IF suara hewan dan nama hewan dalam bahasa inggris THEN dapat ditemukan dalam tebak suara.

Rule simplification:

– R1: IF A and C, THEN E

– R2: IF D and C, THEN F

– R3: IF B and E, THEN F

– R4: IF B, THEN C

– R5: IF F, THEN G

Solusi dengan *Forward Chaining* :

Step I : IF A and C Then E = R1

Step II : IF B then C A,B,C -> True = R4

Step III : If A and C then E A,B,C -> True = R2

Step IV : If B ad E then F A,B,C,E,F -> true = R3

Step V : if F then G. G->True

j. Jadwal Pengerjaan

Tabel 4.1 Jadwal Aktivitas Pembuatan Program

No.	JENIS KEGIATAN	Minggu ke-							
		I	II	III	IV	V	VI	VII	VIII
1	Persiapan program	■							
2	Pengumpulan data		■						
3	Perancangan dan pembuatan			■					
4	Implementasi / Ujicoba				■				
5	Aplikasi program di kelas					■	■	■	■

j.2 Rancang Bangun Aplikasi Java

Untuk merealisasi apa yang telah diperoleh dalam pengumpulan data, telah dibuat rancang bangun aplikasi Java sederhana, sebagai berikut :

Source code Kelompok Kebun Binatang

```

import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
import java.applet.*;
import java.io.*;
import java.net.*;
import java.util.*;

class peternakan extends JFrame
{
    JLabel lbac=new JLabel(new ImageIcon("peternakan.jpg"));
    JLabel head=new JLabel(new ImageIcon("atasp.jpg"));
    JButton tutup=new JButton(new ImageIcon("cancel.png"));
    JLabel foot=new JLabel(new ImageIcon("bawah.jpg"));

    ImageIcon logo=new ImageIcon("logo.gif");
    JButton bkatak=new JButton("Katak");
    JButton bayam=new JButton("Ayam");
    JButton bbebek=new JButton("Bebek");
    JButton bdomba=new JButton("Domba");
    JButton bkucing=new JButton("Kucing");
    JButton banjing=new JButton("Anjing");
    JButton bsapi=new JButton("Sapi");
    JButton bbabi=new JButton("Babi");
    JButton bkuda=new JButton("Kuda");
    JButton bburunghantu=new JButton("Burung Hantu");
    ImageIcon ponbob=new ImageIcon("game2.gif");
    JButton game=new JButton("GAME",ponbob);
    ImageIcon yam=new ImageIcon("zebra.png");
    JButton lanjut=new JButton("Ke Kebun Binatang",yam);
    ImageIcon log=new ImageIcon("logo.gif");

    Container c=new Container();

    public peternakan()
    {
        setTitle("Peternakan");
        setSize(900,710);
        Dimension dim = Toolkit.getDefaultToolkit().getScreenSize();
        int w = getSize().width;
        int h = getSize().height;
        int x = (dim.width - w) / 2;
        int y = (dim.height - h) / 2;
        setIconImage(log.getImage());
    }
}

```

```

void nampil()
{
    c=getContentPane();
    c.setLayout(null);
    c.add(tutup);
    c.add(head);
    c.add(foot);
    c.add(lanjut);
    c.add(game);
    c.add(bkatak);
    c.add(bayam);
    c.add(bbebek);
    c.add(bdomba);
    c.add(bkucing);
    c.add(banjing);
    c.add(bbabi);
    c.add(bsapi);
    c.add(bkuda);
    c.add(bburunghantu);

    c.add(lbac);

    bkatak.setBounds(10,490,70,20);
    bkatak.setBackground(Color.white);
    bkatak.setFont(new Font("Juice ITC",Font.BOLD,15));
    bayam.setBounds(150,440,70,20);
    bayam.setBackground(Color.white);
    bayam.setFont(new Font("Juice ITC",Font.BOLD,15));
    bbebek.setBounds(130,590,70,20);
    bbebek.setBackground(Color.white);
    bbebek.setFont(new Font("Juice ITC",Font.BOLD,15));
    bdomba.setBounds(100,360,80,20);
    bdomba.setBackground(Color.white);
    bdomba.setFont(new Font("Juice ITC",Font.BOLD,15));
    bkucing.setBounds(320,360,70,20);
    bkucing.setBackground(Color.white);
    bkucing.setFont(new Font("Juice ITC",Font.BOLD,15));
    banjing.setBounds(390,390,70,20);
    banjing.setBackground(Color.white);
    banjing.setFont(new Font("Juice ITC",Font.BOLD,15));
    bbabi.setBounds(580,520,60,20);
    bbabi.setBackground(Color.white);
    bbabi.setFont(new Font("Juice ITC",Font.BOLD,15));
    bsapi.setBounds(620,370,60,20);
    bsapi.setBackground(Color.white);
    bsapi.setFont(new Font("Juice ITC",Font.BOLD,15));
    bkuda.setBounds(510,650,60,20);
    bkuda.setBackground(Color.white);
    bkuda.setFont(new Font("Juice ITC",Font.BOLD,15));
    bburunghantu.setBounds(780,450,110,20);
    bburunghantu.setBackground(Color.white);
    bburunghantu.setFont(new Font("Juice ITC",Font.BOLD,15));
}

```

```

game.setBounds(10,40,130,60);
game.setBackground(Color.black);
game.setForeground(Color.white);
game.setFont(new Font("Forte",Font.PLAIN,14));
head.setBounds(0,0,900,30);
tutup.setBounds(860,0,30,30);
foot.setBounds(0,675,900,30);
lanjut.setBounds(700,615,195,50);
lanjut.setBackground(Color.ORANGE);
lanjut.setFont(new Font("Monotype corsiva",Font.ITALIC,16));

lbac.setBounds(0,0,900,710);

setVisible(true);
}
void aksi()
{
tutup.addActionListener(new ActionListener()
{
public void actionPerformed(ActionEvent ee)
{
System.exit(1);
}
});

game.addActionListener(new ActionListener()
{
public void actionPerformed(ActionEvent e)
{
if(game==e.getSource())
{
gepuk gp=new gepuk();
gp.komponen();
gp.batas();
setVisible(false);
}
}
});

bkatak.addActionListener(new ActionListener()
{
public void actionPerformed(ActionEvent e)
{
if(bkatak==e.getSource())
{
SwingUtilities.invokeLater(new Runnable()
{
public void run()
{

```

```

        katak ne=new katak("file:suara binatang/katak.wav","file:inggris/katak.wav");
        ne.aksi();
        ne.tampil();
        setVisible(false);
    }
    });
}

}
});

bbabi.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bbabi==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    babi ne=new babi("file:suara binatang/babi.wav","file:inggris/babi.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

bayam.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bayam==e.getSource())
    {
        SwingUtilities.invokeLater(new Runnable()
        {
            public void run()
            {
                ayam ne=new ayam("file:suara binatang/ayam.wav","file:inggris/ayam.wav");
                ne.aksi();
                ne.tampil();
                setVisible(false);
            }
        });
    }
}

}
});

```

```
bdomba.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bdomba==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    domba ne=new domba("file:suara binatang/domba.wav","file:inggris/domba.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

b bebek.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bbebek==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    bebek ne=new bebek("file:suara binatang/bebek.wav","file:inggris/bebek.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});
```

```

bkucing.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bkucing==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    kucing ne=new kucing("file:suara binatang/kucing.wav","file:inggris/kucing.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

banjing.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(banjing==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    anjing ne=new anjing("file:suara binatang/anjing.wav","file:inggris/anjing.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

bsapi.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bsapi==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {

```

```

        sapi ne=new sapi("file:suara binatang/sapi.wav","file:inggris/sapi.wav");
        ne.aksi();
        ne.tampil();
        setVisible(false);
    }
    });
}

}
});
bkuda.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bkuda==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    kuda ne=new kuda("file:suara binatang/kuda.wav","file:inggris/kuda.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});
bburunghantu.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bburunghantu==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    burunghantu ne=new burunghantu("file:suara
binatang/burunghantu.wav","file:inggris/burunghantu.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});
}
});
}

```

```

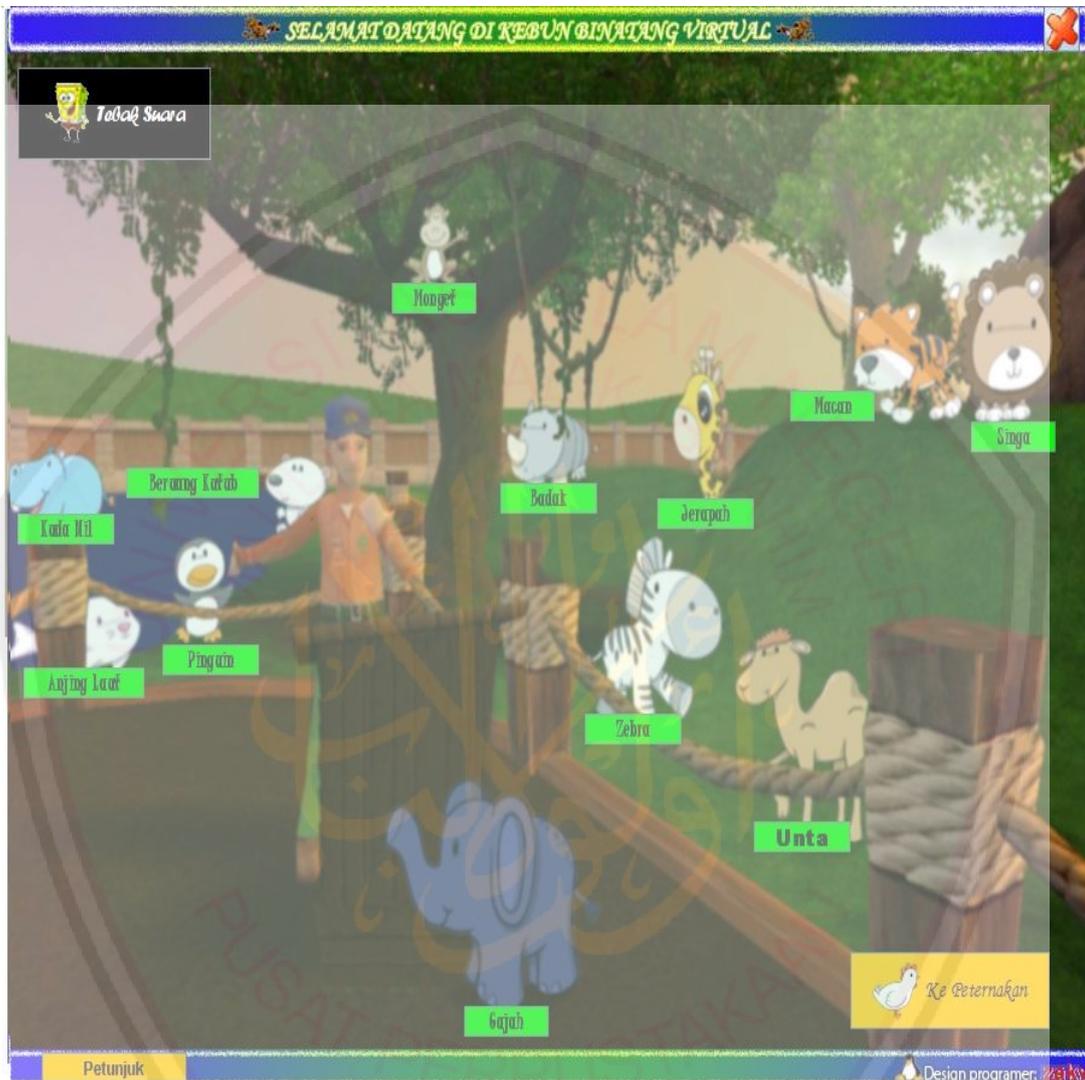
lanjut.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(lanjut==e.getSource())
        {
            kebunBinatang pt=new kebunBinatang();
            pt.aksi();
            pt.nampil();
            setVisible(false);
        }
    }
});
}
public static void main(String args[])
{
    peternakan pt=new peternakan();
    pt.nampil();
    pt.aksi();
}
}

```

j.3 Mengoperasikan Program

Dalam mengoperasikan hasil rancang bangun aplikasi media pembelajaran mengenal hewan berbasis *java* terlebih dahulu harus diinstall *software* program *java* dan disertai editor *gel* sebagai penunjangnya yang nantinya sebagai media eksekusi aplikasi rancang bangun yang diaktifkan didalam komputer.

Untuk memulai menjalankan aplikasi program pengenalan hewan, langkah pertama yang dilakukan adalah dengan operasikan program *java* terlebih dahulu atau klik menu mengenal hewan. Adapun bentuk gambar yang ditampilkan dalam komputer sebagai aplikasi dari program *java* antara lain sebagai berikut :



Gambar 4.2 Menu gambar dan pilihan hewan di kebun binatang.

Sebelum menjalankan aplikasi program selanjutnya, alangkah baiknya melihat petunjuk atau cara penggunaan program tersebut, yaitu dengan menekan klik petunjuk. Hal ini perlu dilakukan untuk lebih mengetahui atau memahami cara-cara mengoperasikan aplikasi program sesuai dengan yang diinginkan.

Cara menggunakan program ini sangat mudah
Bila kita ingin melihat keterangan tentang binatang beserta suaranya, kita tinggal mengklik nama binatang yang berada didekat binatang tersebut.
Untuk game menebak suara, Kita tinggal mengklik tombol jawaban untuk mengetahui jawaban dari suara yang telah diperdengarkan sebelumnya.
Sedangkan game yang satu lagi kita tinggal mengklik pinguin yang berjalan secara acak dengan menggunakan mouse untuk mendapatkan nilai. Nilai yang didapatkan maksimal 10 di setiap level. Terdapat 5 level yang dapat dimainkan.
Selamat memakai Program ini !

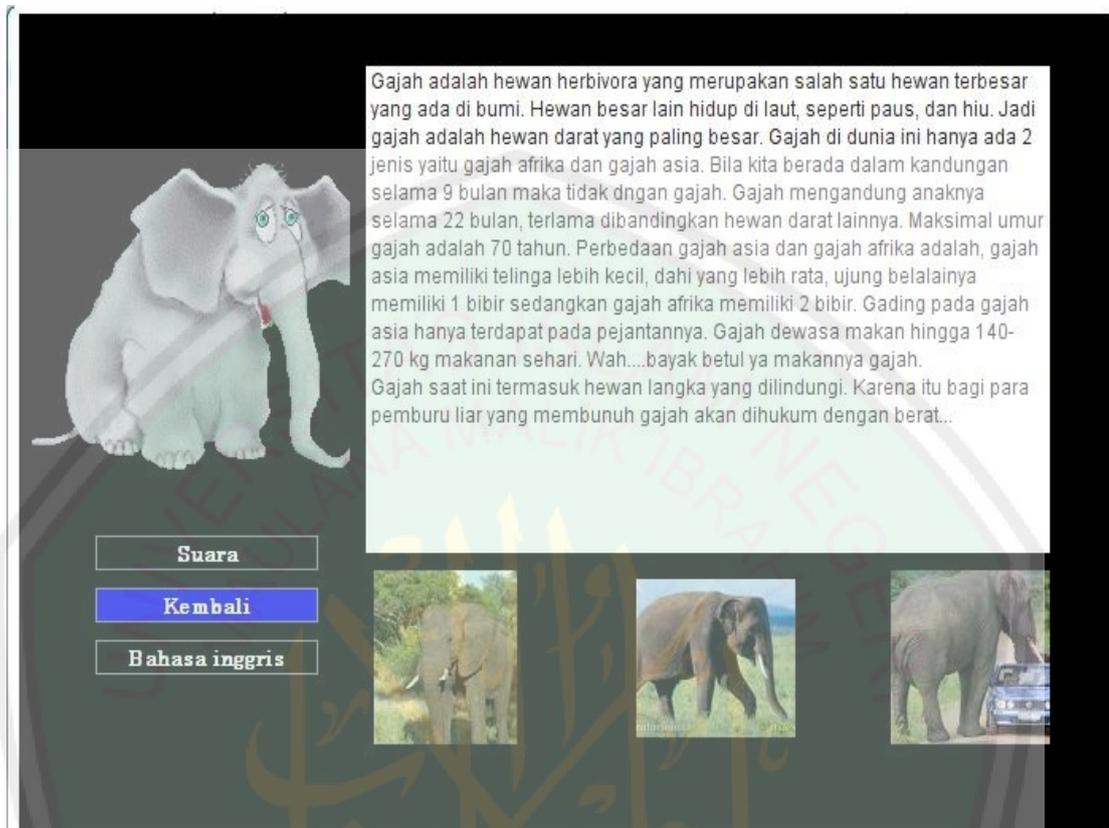
Creator... <maulanauin@gmail.com>

Setelah selesai membaca tentang cara operasional program tersebut, maka untuk dapat mengenal lebih dalam lagi tentang program pengenalan hewan-hewan yang ada di kebun binatang, langkah berikutnya adalah tinggal memilih atau klik



gambar yang diinginkan yang ada dalam menu di komputer tersebut.

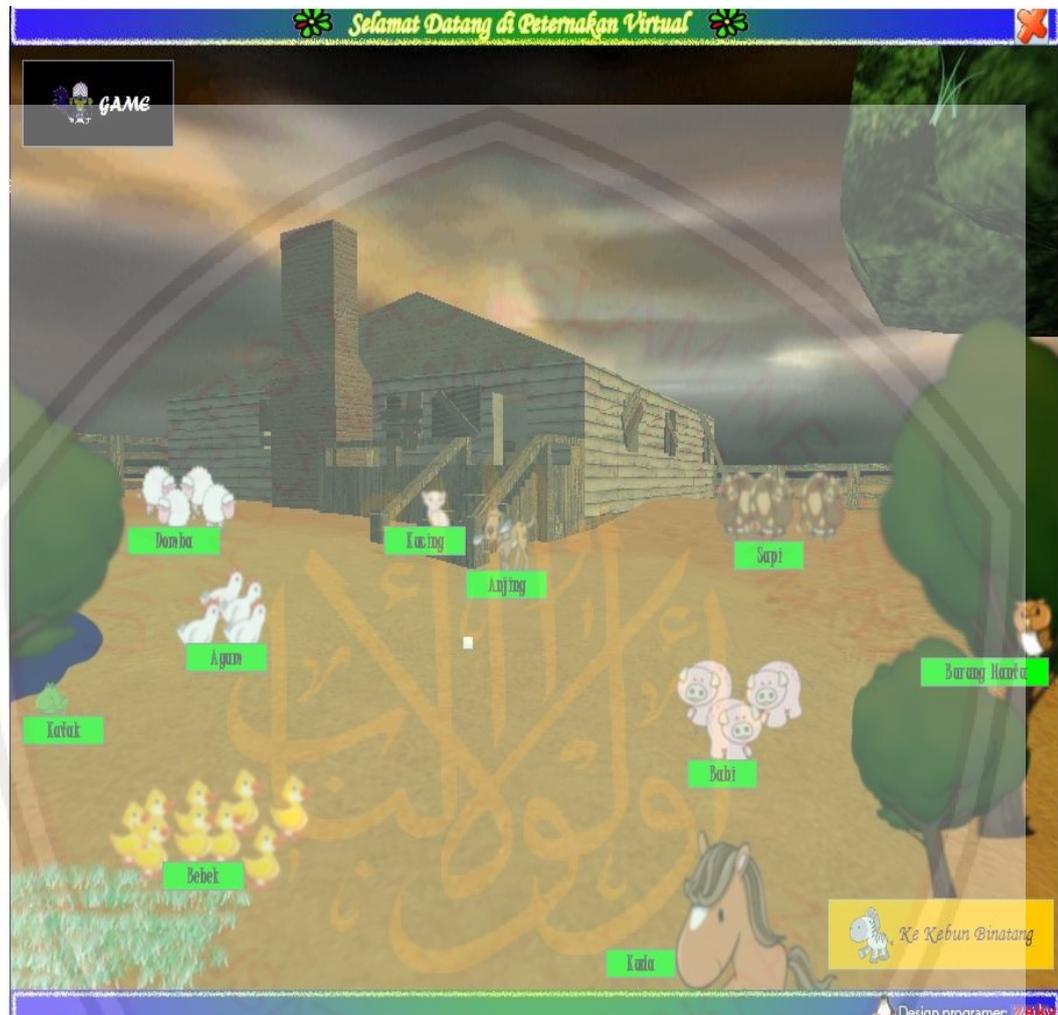
Gambar 4.2.a. Menu gambar dan pilihan hewan di kebun binatang.



Gambar 4.4 Pilihan hewan di kebun binatang.

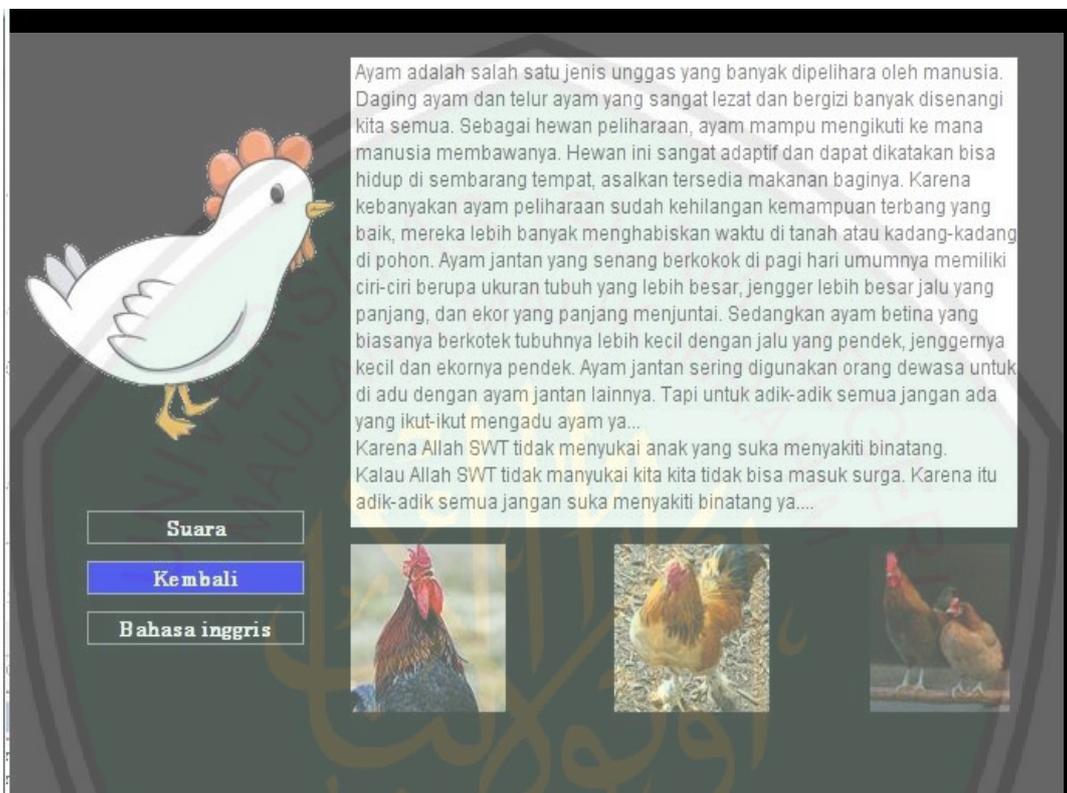
- Klik Suara untuk mengeluarkan suara hewan.
- Klik Kembali untuk keluar dari tampilan gambar tersebut dan kembali lagi ke tampilan menu awal.
- Klik Bahasa Inggris untuk mengeluarkan suara sebutan nama hewan tersebut dalam bahasa Inggris.

Setelah kembali ke menu gambar awal dan untuk mengetahui hewan-hewan peternakan, maka tekan klik ke peternakan, kemudian akan muncul gambar sebagaimana dibawah ini.



Gambar 4.5 Menu gambar dan pilihan hewan di peternakan

Jika ingin melihat tentang hewan yang ada di lingkungan menu hewan di peternakan, maka klik gambar salah satu contoh hewan tersebut. Adapun tampilan gambar hewan-hewan yang diinginkan akan ditampilkan sebagaimana dalam contoh di bawah ini.



Gambar 4.6 Pilihan hewan di peternakan.

Secara teknis untuk mengetahui atau mengeluarkan suara hewan peternakan adalah sama dengan cara pengoperasian hewan di kebun binatang.

j..4 Hasil Implementasi

Pada implementasi hasil rancang bangun yang diterapkan di kelas 1 Sekolah Dasar Negeri Kalirejo I Kecamatan Dringu Kabupaten Probolinggo yang berbasis *Java* yang disertai editor *Gel* sebagai media penunjangnya dengan metode *Forward Chaining*, adalah sebagai berikut :

a. Data Sekolah

- Sekolah Dasar Negeri Kalirejo I No. 20
- Jalan Kalimas No. 20 Kalirejo Kecamatan Dringu Kabupaten Probolinggo Jawa Timur Kode Pos 67271.
- Jumlah guru pengajar sebanyak tiga belas orang.
- Sarana yang dimiliki antara lain perpustakaan, komputer, drum band.
- Jumlah murid kelas 1 sebanyak 30 anak
- Jumlah murid kelas 2 sebanyak 25 anak
- Jumlah murid kelas 3 sebanyak 33 anak
- Jumlah murid kelas 4 sebanyak 25 anak
- Jumlah murid kelas 5 sebanyak 29 anak
- Jumlah murid kelas 6 sebanyak 36 anak
- Mata pelajaran yang wajib diikuti para siswa antara lain adalah Agama, Matematika, IPA, IPS, Bahasa Indonesia, Bahasa Inggris, PKN, Olah Raga, Kesenian dan muatan lokal (bahasa jawa).
- Mata pelajaran ekstrakurikuler antara lain adalah komputer, pramuka, drum band.

b. Data Siswa kelas 1 Sekolah Dasar Negeri Kalirejo I

- Jumlah siswa kelas satu sebanyak 30 murid dalam kelas.
- Salah satu ekstrakurikuler kelas 1 yang harus diikuti adalah antara lain pengenalan komputer.
- Jam belajar komputer dalam satu minggu selama dua jam per kelas.

c. Implementasi hasil rancang bangun aplikasi media pembelajaran mengenal hewan berbasis java yang diterapkan di kelas 1 sebagai berikut :

- Waktu pelaksanaan setiap hari sabtu selama 2 jam yang dimulai pada pukul 14.00 – 16.00 WIB.
- Metode pengajaran yang diberikan antara lain pengenalan komputer, aplikasi program dan evaluasi hasil pembelajaran.
- Pelaksanaan pembelajaran selama 8 jam pertemuan selama 1 bulan.

Tabel 4.2 Jadwal Pembelajaran Ekstrakurikuler Komputer

Materi	Waktu	Keterangan
1. Pengenalan komputer	2 jam	Waktu belajar pukul 14.00 – 16.00 WIB Guru pembimbing : Guru Pengawas :
2. Aplikasi program	4 jam	M. Mukhtar Zaky Dwi Agustini, S.Pd.
3. Evaluasi Siswa	2 jam	

- Adapun materi yang disampaikan kepada murid kelas 1 diantaranya adalah fungsi dan manfaat komputer.
- Penilaian hasil evaluasi dibuat berdasarkan hasil quisioner dan evaluasi pengenalan computer dan aplikasi program.
- Dari hasil evaluasi pelajaran ekstra kulikuler mata pelajaran komputer untuk para siswa kelas 1 sebagai berikut :

Tabel 4.3 Hasil evaluasi

Evaluasi	Hasil
Jumlah Murid	30 Siswa
Jumlah Skor Evaluasi	264
Jumlah Skor Maksimal	300
% Skor Tercapai	88 %
Ketuntasan Belajar Klasikal	100 %
Siswa yang memiliki komputer di rumahnya	2 Siswa (7%)
Siswa yang tidak memiliki komputer di rumahnya	28 Siswa (93%)

Siswa yang senang dengan komputer	29 Siswa (97%)
Siswa yang senang dengan aplikasi pengenalan hewan	29 Siswa (97%)

(Data : Lampiran 1)





BAB V

KESIMPULAN DAN SARAN

1..1Kesimpulan

Berdasarkan pembahasan penelitian penerapan rancang bangun aplikasi media pembelajaran mengenal hewan berbasis *java* dengan metode *forward chaining* maka dapat disimpulkan sebagai berikut :

1. Pembuatan aplikasi software rancang bangun media pembelajaran mengenal hewan berbasis *java* dengan menggunakan metode *forward chaining* sangat membantu dalam proses belajar mengajar bagi para guru pada khususnya dan bidang pendidikan dasar pada umumnya, karena setelah diterapkan pada anak-anak sekolah dasar dalam program ekstrakurikuler belajar komputer, para siswa lebih cepat memahami dan menghafal tentang jenis, kelompok, suara dan nama hewan dalam bahasa inggris beserta informasi lainnya dari berbagai macam jenis hewan diajarkan.
2. Proses belajar mengajar melalui media gambar dan suara yang ditampakkan dalam layar komputer, ternyata banyak disenangi oleh anak-anak sekolah dasar, karena bentuk nyata hewan melalui dunia maya yang disampaikan oleh pengajar dapat

secara langsung dilihat dan cepat dimengerti, hal ini juga dapat menambah antusias belajar para siswa.

2.2 Saran

- Perlu diperbanyak adanya program-program aplikasi lain yang sederhana yang dapat membantu untuk mempermudah dan mempercepat metode pembelajaran guna meningkatkan mutu pendidikan terutama untuk sekolah dasar.
 1. Guna meningkatkan mutu pendidikan di masa depan, di sekolah-sekolah dasar perlu diperbanyak sarana komputernya. Hal ini dikarenakan masih banyak siswa yang belum mempunyai komputer sendiri.
 2. Diharapkan lebih banyak lagi aplikasi-aplikasi *software* lain yang lebih mudah operasionalnya mengenai disiplin ilmu yang bermanfaat untuk tingkat sekolah dasar, karena masih banyak murid yang masih belum familier dengan penggunaan *mouse*, *monitor*, dan *cpu* dan lain-lain.

DAFTAR PUSTAKA

<http://stittattaqwa.blogspot.com/2011/07/tafsir-qs-al-mujadalah-58-11.html>
(diunduh pada tanggal 12 Juni 2013).

<http://www.mengujihasilinstalasiidegel.com> « Uqifumi's Blog.com. (diunduh pada tanggal 16 Juni 2013)

<http://www.java.sun.com>. (diunduh pada tanggal 16 Juni 2013)

Feri Fahrur Rohman dan Ami Fauzijah, 2008. *Rancang Bangun Aplikasi Sistem Pakar Untuk Menentukan Jenis Gangguan Perkembangan Pada Anak*. Yogyakarta, Media Informatika. (Jurnal).

<http://der-traumer.blogspot.com/2012/09/pengertian-tujuan-manfaat-dan-fungsi.html>(diunduh pada tanggal 19 Juni 2013).

Kusrini, M. Kom. 2008. *Aplikasi Sistem Pakar*. Yogyakarta: C.V Andi Offset

Dwi Prasetyo, Didik, “150 Rahasia Pemrograman Java”, PT Elex Media Komputindo, Jakarta, 2007.

Hermawan Beni, “Menguasai Java 2 & Object Oriented Programming”, Andi Offset, Yogyakarta, 2004.

Fatchurrochman, Achmad Basuki, “Pemrograman Visual Java”, Graha Ilmu, Yogyakarta, 2007.

Ir. Yuniar Supardi, “Pemrograman Java 2 SE SDK 1.4”, Gramedia, Jakarta, 2005.

Budi Susanto, “Pemrograman client/server dengan Java 2”, Gramedia, Jakarta, 2003.



Source Code Program

```
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
import java.applet.*;
import java.io.*;
import java.net.*;
import java.util.*;

class peternakan extends JFrame
{
    JLabel lbac=new JLabel(new ImageIcon("peternakan.jpg"));
    JLabel head=new JLabel(new ImageIcon("atasp.jpg"));
    JButton tutup=new JButton(new ImageIcon("cancel.png"));
    JLabel foot=new JLabel(new ImageIcon("bawah.jpg"));

    ImageIcon logo=new ImageIcon("logo.gif");
    JButton bkatak=new JButton("Katak");
    JButton bayam=new JButton("Ayam");
    JButton bbebek=new JButton("Bebek");
    JButton bdomba=new JButton("Domba");
    JButton bkucing=new JButton("Kucing");
    JButton banjing=new JButton("Anjing");
    JButton bsapi=new JButton("Sapi");
    JButton bbabi=new JButton("Babi");
    JButton bkuda=new JButton("Kuda");
    JButton bburunghantu=new JButton("Burung Hantu");
    ImageIcon ponbob=new ImageIcon("game2.gif");
    JButton game=new JButton("GAME",ponbob);
    ImageIcon yam=new ImageIcon("zebra.png");
    JButton lanjut=new JButton("Ke Kebun Binatang",yam);
    ImageIcon log=new ImageIcon("logo.gif");

    Container c=new Container();

    public peternakan()
    {
        setTitle("Peternakan");
        setSize(900,710);
        Dimension dim = Toolkit.getDefaultToolkit().getScreenSize();
        int w = getSize().width;
        int h = getSize().height;
        int x = (dim.width - w) / 2;
        int y = (dim.height - h) / 2;
        setIconImage(log.getImage());
    }
}
```



```

void tampil()
{
    c=getContentPane();
    c.setLayout(null);
    c.add(tutup);
    c.add(head);
    c.add(foot);
    c.add(lanjut);
    c.add(game);
    c.add(bkatak);
    c.add(bayam);
    c.add(bbebek);
    c.add(bdomba);
    c.add(bkucing);
    c.add(banjing);
    c.add(bbabi);
    c.add(bsapi);
    c.add(bkuda);
    c.add(bburunghantu);

    c.add(lbac);

    bkatak.setBounds(10,490,70,20);
    bkatak.setBackground(Color.white);
    bkatak.setFont(new Font("Juice ITC",Font.BOLD,15));
    bayam.setBounds(150,440,70,20);
    bayam.setBackground(Color.white);
    bayam.setFont(new Font("Juice ITC",Font.BOLD,15));
    bbebek.setBounds(130,590,70,20);
    bbebek.setBackground(Color.white);
    bbebek.setFont(new Font("Juice ITC",Font.BOLD,15));
    bdomba.setBounds(100,360,80,20);
    bdomba.setBackground(Color.white);
    bdomba.setFont(new Font("Juice ITC",Font.BOLD,15));
    bkucing.setBounds(320,360,70,20);
    bkucing.setBackground(Color.white);
    bkucing.setFont(new Font("Juice ITC",Font.BOLD,15));
    banjing.setBounds(390,390,70,20);
    banjing.setBackground(Color.white);
    banjing.setFont(new Font("Juice ITC",Font.BOLD,15));
    bbabi.setBounds(580,520,60,20);
    bbabi.setBackground(Color.white);
    bbabi.setFont(new Font("Juice ITC",Font.BOLD,15));
    bsapi.setBounds(620,370,60,20);
    bsapi.setBackground(Color.white);
    bsapi.setFont(new Font("Juice ITC",Font.BOLD,15));
    bkuda.setBounds(510,650,60,20);
    bkuda.setBackground(Color.white);
    bkuda.setFont(new Font("Juice ITC",Font.BOLD,15));
    bburunghantu.setBounds(780,450,110,20);
    bburunghantu.setBackground(Color.white);
    bburunghantu.setFont(new Font("Juice ITC",Font.BOLD,15));
}

```

```

game.setBounds(10,40,130,60);
game.setBackground(Color.black);
game.setForeground(Color.white);
game.setFont(new Font("Forte",Font.PLAIN,14));
head.setBounds(0,0,900,30);
tutup.setBounds(860,0,30,30);
foot.setBounds(0,675,900,30);
lanjut.setBounds(700,615,195,50);
lanjut.setBackground(Color.ORANGE);
lanjut.setFont(new Font("Monotype corsiva",Font.ITALIC,16));

lbac.setBounds(0,0,900,710);

setVisible(true);
}
void aksi()
{
tutup.addActionListener(new ActionListener()
{
public void actionPerformed(ActionEvent ee)
{
System.exit(1);
}
});

game.addActionListener(new ActionListener()
{
public void actionPerformed(ActionEvent e)
{
if(game==e.getSource())
{
gepuk gp=new gepuk();
gp.komponen();
gp.batas();
setVisible(false);
}
}
});

```

```

bkatak.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bkatak==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    katak ne=new katak("file:suara binatang/katak.wav","file:inggris/katak.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

bbabi.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bbabi==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    babi ne=new babi("file:suara binatang/babi.wav","file:inggris/babi.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

bayam.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bayam==e.getSource())

```

```

{
    SwingUtilities.invokeLater(new Runnable()
    {
        public void run()
        {
            ayam ne=new ayam("file:suara binatang/ayam.wav","file:inggris/ayam.wav");
            ne.aksi();
            ne.tampil();
            setVisible(false);
        }
    });
}

}

});

bdomba.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bdomba==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    domba ne=new domba("file:suara binatang/domba.wav","file:inggris/domba.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

b bebek.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bbebek==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {

```

```

        bebek ne=new bebek("file:suara binatang/bebek.wav","file:inggris/bebek.wav");
        ne.aksi();
        ne.tampil();
        setVisible(false);
    }
    });
}

}
});
bkucing.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bkucing==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    kucing ne=new kucing("file:suara binatang/kucing.wav","file:inggris/kucing.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

banjing.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(banjing==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    anjing ne=new anjing("file:suara binatang/anjing.wav","file:inggris/anjing.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

```

```

    });
    }

}
});
bsapi.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bsapi==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    sapi ne=new sapi("file:suara binatang/sapi.wav","file:inggris/sapi.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});
bkuda.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bkuda==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    kuda ne=new kuda("file:suara binatang/kuda.wav","file:inggris/kuda.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});
bburunghantu.addActionListener(new ActionListener()
{

```

```

public void actionPerformed(ActionEvent e)
{
    if(bburunghantu==e.getSource())
    {
        SwingUtilities.invokeLater(new Runnable()
        {
            public void run()
            {
                burunghantu ne=new burunghantu("file:suara
binatang/burunghantu.wav","file:inggris/burunghantu.wav");
                ne.aksi();
                ne.tampil();
                setVisible(false);
            }
        });
    }
}
});

```

```

lanjut.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(lanjut==e.getSource())
        {
            kebunBinatang pt=new kebunBinatang();
            pt.aksi();
            pt.nampil();
            setVisible(false);
        }
    }
});

```

```

}
public static void main(String args[])
{
    peternakan pt=new peternakan();
    pt.nampil();
    pt.aksi();
}

```

```

}

```

Source Code program2

```
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
import java.applet.*;
import java.io.*;
import java.net.*;
import java.util.*;

class peternakan extends JFrame
{
    JLabel lbac=new JLabel(new ImageIcon("peternakan.jpg"));
    JLabel head=new JLabel(new ImageIcon("atasp.jpg"));
    JButton tutup=new JButton(new ImageIcon("cancel.png"));
    JLabel foot=new JLabel(new ImageIcon("bawah.jpg"));

    ImageIcon logo=new ImageIcon("logo.gif");

    JButton bkatak=new JButton("Katak");
    JButton bayam=new JButton("Ayam");
    JButton bbebek=new JButton("Bebek");
    JButton bdomba=new JButton("Domba");
    JButton bkucing=new JButton("Kucing");
    JButton banjing=new JButton("Anjing");
    JButton bsapi=new JButton("Sapi");
    JButton bbabi=new JButton("Babi");
    JButton bkuda=new JButton("Kuda");
    JButton bburunghantu=new JButton("Burung Hantu");
    ImageIcon ponbob=new ImageIcon("game2.gif");
    JButton game=new JButton("GAME",ponbob);
    ImageIcon yam=new ImageIcon("zebra.png");
    JButton lanjut=new JButton("Ke Kebun Binatang",yam);
    ImageIcon log=new ImageIcon("logo.gif");

    Container c=new Container();

    public peternakan()
    {
        setTitle("Peternakan");
        setSize(900,710);
        Dimension dim = Toolkit.getDefaultToolkit().getScreenSize();
        int w = getSize().width;
        int h = getSize().height;
        int x = (dim.width - w) / 2;
        int y = (dim.height - h) / 2;
        setIconImage(log.getImage());
    }
}
```



```

setLocation(x,y);
setUndecorated(true);
}

void tampil()
{
c=getContentPane();
c.setLayout(null);
c.add(tutup);
c.add(head);
c.add(foot);
c.add(lanjut);
c.add(game);
c.add(bkatak);
c.add(bayam);
c.add(bbebek);
c.add(bdomba);
c.add(bkucing);
c.add(banjing);
c.add(bbabi);
c.add(bsapi);
c.add(bkuda);
c.add(bburunghantu);

c.add(lbac);

bkatak.setBounds(10,490,70,20);
bkatak.setBackground(Color.white);
bkatak.setFont(new Font("Juice ITC",Font.BOLD,15));
bayam.setBounds(150,440,70,20);
bayam.setBackground(Color.white);
bayam.setFont(new Font("Juice ITC",Font.BOLD,15));
bbebek.setBounds(130,590,70,20);
bbebek.setBackground(Color.white);
bbebek.setFont(new Font("Juice ITC",Font.BOLD,15));
bdomba.setBounds(100,360,80,20);
bdomba.setBackground(Color.white);
bdomba.setFont(new Font("Juice ITC",Font.BOLD,15));
bkucing.setBounds(320,360,70,20);
bkucing.setBackground(Color.white);
bkucing.setFont(new Font("Juice ITC",Font.BOLD,15));
banjing.setBounds(390,390,70,20);
banjing.setBackground(Color.white);
banjing.setFont(new Font("Juice ITC",Font.BOLD,15));
bbabi.setBounds(580,520,60,20);
bbabi.setBackground(Color.white);
bbabi.setFont(new Font("Juice ITC",Font.BOLD,15));
bsapi.setBounds(620,370,60,20);
bsapi.setBackground(Color.white);
bsapi.setFont(new Font("Juice ITC",Font.BOLD,15));
bkuda.setBounds(510,650,60,20);
bkuda.setBackground(Color.white);

```

```

bkuda.setFont(new Font("Juice ITC",Font.BOLD,15));
bburunghantu.setBounds(780,450,110,20);
bburunghantu.setBackground(Color.white);
bburunghantu.setFont(new Font("Juice ITC",Font.BOLD,15));

game.setBounds(10,40,130,60);
game.setBackground(Color.black);
game.setForeground(Color.white);
game.setFont(new Font("Forte",Font.PLAIN,14));
head.setBounds(0,0,900,30);
tutup.setBounds(860,0,30,30);
foot.setBounds(0,675,900,30);
lanjut.setBounds(700,615,195,50);
lanjut.setBackground(Color.ORANGE);
lanjut.setFont(new Font("Monotype corsiva",Font.ITALIC,16));

lbac.setBounds(0,0,900,710);

setVisible(true);
}
void aksi()
{
tutup.addActionListener(new ActionListener()
{
public void actionPerformed(ActionEvent ee)
{
System.exit(1);
}
});
game.addActionListener(new ActionListener()
{
public void actionPerformed(ActionEvent e)
{
if(game==e.getSource())
{
gepuk gp=new gepuk();
gp.komponen();
gp.batas();
setVisible(false);
}
}
});
}

```

```

bkatak.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bkatak==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    katak ne=new katak("file:suara binatang/katak.wav","file:inggris/katak.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

bbabi.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bbabi==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    babi ne=new babi("file:suara binatang/babi.wav","file:inggris/babi.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

bayam.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bayam==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {

```

```

    public void run()
    {
        ayam ne=new ayam("file:suara binatang/ayam.wav","file:inggris/ayam.wav");
        ne.aksi();
        ne.tampil();
        setVisible(false);
    }
    });
}

}
});

bdomba.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bdomba==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    domba ne=new domba("file:suara binatang/domba.wav","file:inggris/domba.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

bbebek.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bbebek==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {

```

```

        public void run()
        {
            bebek ne=new bebek("file:suara binatang/bebek.wav","file:inggris/bebek.wav");
            ne.aksi();
            ne.tampil();
            setVisible(false);
        }
    });
}

}
});
bkucing.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(bkucing==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    kucing ne=new kucing("file:suara binatang/kucing.wav","file:inggris/kucing.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});

banjing.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(banjing==e.getSource())
        {
            SwingUtilities.invokeLater(new Runnable()
            {
                public void run()
                {
                    anjing ne=new anjing("file:suara binatang/anjing.wav","file:inggris/anjing.wav");
                    ne.aksi();
                    ne.tampil();
                    setVisible(false);
                }
            });
        }
    }
});
}

```

```

    }
    });
    bsapi.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent e)
        {
            if(bsapi==e.getSource())
            {
                SwingUtilities.invokeLater(new Runnable()
                {
                    public void run()
                    {
                        sapi ne=new sapi("file:suara binatang/sapi.wav","file:inggris/sapi.wav");
                        ne.aksi();
                        ne.tampil();
                        setVisible(false);
                    }
                });
            }
        }
    });
    bkuda.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent e)
        {
            if(bkuda==e.getSource())
            {
                SwingUtilities.invokeLater(new Runnable()
                {
                    public void run()
                    {
                        kuda ne=new kuda("file:suara binatang/kuda.wav","file:inggris/kuda.wav");
                        ne.aksi();
                        ne.tampil();
                        setVisible(false);
                    }
                });
            }
        }
    });
    bburunghantu.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent e)
        {
            if(bburunghantu==e.getSource())
            {
                SwingUtilities.invokeLater(new Runnable()
                {
                    public void run()
                    {

```

```
        burunghantu ne=new burunghantu("file:suara
binatang/burunghantu.wav","file:inggris/burunghantu.wav");
        ne.aksi();
        ne.tampil();
        setVisible(false);
    }
    });
}
```

```
lanjut.addActionListener(new ActionListener()
{
    public void actionPerformed(ActionEvent e)
    {
        if(lanjut==e.getSource())
        {
            kebunBinatang pt=new kebunBinatang();
            pt.aksi();
            pt.nampil();
            setVisible(false);
        }
    }
});
```

```

}
public static void main(String args[])
{
    peternakan pt=new peternakan();
    pt.nampil();
    pt.aksi();
}
}
```



Source Code Informasi Hewan

```
import java.awt.event.*;
import java.awt.*;
import java.applet.*;
import javax.swing.*;
import java.net.*;

public class gajah extends JFrame
{
    ImagemIcon logo=new ImagemIcon("logo.gif");
    ImagemIcon a=new ImagemIcon("kartun/binatanggajah28829.gif");
    ImagemIcon a1=new ImagemIcon("gambar asli/gajah1.jpg");
    ImagemIcon a2=new ImagemIcon("gambar asli/gajah2.jpg");
    ImagemIcon a3=new ImagemIcon("gambar asli/gajah3.jpg");

    JLabel ayam=new JLabel(a);
    JLabel ayam1=new JLabel(a1);
    JLabel ayam2=new JLabel(a2);
    JLabel ayam3=new JLabel(a3);

    JButton btnPlay=new JButton("Suara");
    JButton kembali=new JButton("Kembali");
    JButton bahasa=new JButton("Bahasa inggris");

    JTextArea tarea=new JTextArea(" Gajah adalah hewan herbivora yang merupakan salah satu hewan
    terbesar \n yang ada di bumi. Hewan besar lain hidup di laut, seperti paus, dan hiu. Jadi \n gajah adalah
    hewan darat yang paling besar. Gajah di dunia ini hanya ada 2 \n jenis yaitu gajah afrika dan gajah asia.
    Bila kita berada dalam kandungan \n selama 9 bulan maka tidak dngan gajah. Gajah mengandung
    anaknya \n selama 22 bulan, terlama dibandingkan hewan darat lainnya. Maksimal umur \n gajah adalah
    70 tahun. Perbedaan gajah asia dan gajah afrika adalah, gajah \n asia memiliki telinga lebih kecil, dahi
    yang lebih rata, ujung belalainya \n memiliki 1 bibir sedangkan gajah afrika memiliki 2 bibir. Gading pada
    gajah \n asia hanya terdapat pada pejantannya. Gajah dewasa makan hingga 140-\n 270 kg makanan
    sehari. Wah....bayak betul ya makannya gajah. \n Gajah saat ini termasuk hewan langka yang dilindungi.
    Karena itu bagi para \n pemburu liar yang membunuh gajah akan dihukum dengan berat...");

    AudioClip audio;
    AudioClip engl;
    Container c=new Container();

    public gajah(String sURL, String bhsEngl)
    {

        super("GAJAH");
        setSize(690,470);
        Dimension dim = Toolkit.getDefaultToolkit().getScreenSize();
        int w = getSize().width;
        int h = getSize().height;
        int x = (dim.width - w) / 2;
        int y = (dim.height - h) / 2;
```

```

setLocation(x,y);
setUndecorated(true);
setIconImage(logo.getImage());
try
{
    URL ling = new URL(bhsEngl);
    engl = Applet.newAudioClip(ling);
    URL url = new URL(sURL);
    audio = Applet.newAudioClip(url);
}
catch (MalformedURLException ex)
{
    System.err.println(ex.getMessage());
    System.exit(0);
}

}

void tampil()
{
    c=getContentPane();
    c.setLayout(null);
    c.setBackground(Color.BLACK);
    c.add(btnPlay);
    c.add(kembali);
    c.add(bahasa);
    c.add(ayam);
    c.add(ayam1);
    c.add(ayam2);
    c.add(ayam3);
    c.add(tarea);

    ayam.setBounds(10,15,200,300);
    btnPlay.setBounds(50,300,140,20);
    btnPlay.setBackground(Color.black);
    btnPlay.setForeground(Color.WHITE);
    btnPlay.setFont(new Font("Century",Font.BOLD,12));
    kembali.setBounds(50,330,140,20);
    kembali.setBackground(Color.BLUE);
    kembali.setForeground(Color.WHITE);
    kembali.setFont(new Font("Century",Font.BOLD,12));
    bahasa.setBounds(50,360,140,20);
    bahasa.setBackground(Color.black);
    bahasa.setForeground(Color.white);
    bahasa.setFont(new Font("Century",Font.BOLD,12));
    ayam1.setBounds(220,320,100,100);
    ayam2.setBounds(390,320,100,100);
    ayam3.setBounds(550,320,100,100);
    tarea.setBounds(220,30,430,280);
    //tarea.setLineWrap(true);
    tarea.setEditable(false);

```

```

    setVisible(true);
}
void aksi()
{
    btnPlay.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent e)
        {
            if(btnPlay==e.getSource())
                audio.play();
            engl.stop();
        }
    });
    bahasa.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent e)
        {
            if(bahasa==e.getSource())
                engl.play();
            audio.stop();
        }
    });
    kembali.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent e)
        {
            if(kembali==e.getSource())
            {
                audio.stop();
                engl.stop();
                kebunBinatang kb=new kebunBinatang();
                kb.nampil();
                kb.aksi();
                dispose();
            }
        }
    });
}
public static void main(String args[])
{
    SwingUtilities.invokeLater(new Runnable()
    {
        public void run()
        {
            gajah ne=new gajah("file:suara binatang/gajah.au","file:inggris/gajah.wav");
            ne.aksi();
            ne.tampil();
        }
    });
}
}

```



Source Code Informasi Hewan

```
import java.awt.event.*;
import java.awt.*;
import java.applet.*;
import javax.swing.*;
import java.net.*;

public class ayam extends JFrame
{
    ImageIcon logo=new ImageIcon("logo.gif");
    ImageIcon a=new ImageIcon("kartun/ayamkartun.gif");
    ImageIcon a1=new ImageIcon("gambar asli/ayam1.jpg");
    ImageIcon a2=new ImageIcon("gambar asli/ayam2.jpg");
    ImageIcon a3=new ImageIcon("gambar asli/ayam3.jpg");

    JLabel ayam=new JLabel(a);
    JLabel ayam1=new JLabel(a1);
    JLabel ayam2=new JLabel(a2);
    JLabel ayam3=new JLabel(a3);

    JButton btnPlay=new JButton("Suara");
    JButton kembali=new JButton("Kembali");
    JButton bahasa=new JButton("Bahasa inggris");

    JTextArea tarea=new JTextArea(" Ayam adalah salah satu jenis unggas yang banyak dipelihara oleh manusia. \n Daging ayam dan telur ayam yang sangat lezat dan bergizi banyak disenangi \n kita semua. Sebagai hewan peliharaan, ayam mampu mengikuti ke mana \n manusia membawanya. Hewan ini sangat adaptif dan dapat dikatakan bisa \n hidup di sembarang tempat, asalkan tersedia makanan baginya. Karena \n kebanyakan ayam peliharaan sudah kehilangan kemampuan terbang yang \n baik, mereka lebih banyak menghabiskan waktu di tanah atau kadang-kadang \n di pohon. Ayam jantan yang senang berkokok di pagi hari umumnya memiliki \n ciri-ciri berupa ukuran tubuh yang lebih besar, jengger lebih besar jalu yang \n panjang, dan ekor yang panjang menjuntai. Sedangkan ayam betina yang \n biasanya berkotek tubuhnya lebih kecil dengan jalu yang pendek, jengger nya \n kecil dan ekornya pendek. Ayam jantan sering digunakan orang dewasa untuk \n di adu dengan ayam jantan lainnya. Tapi untuk adik-adik semua jangan ada \n yang ikut-ikutan mengadu ayam ya... \n Karena Allah SWT tidak menyukai anak yang suka menyakiti binatang. \n Kalau Allah SWT tidak menyukai kita kita tidak bisa masuk surga. Karena itu \n adik-adik semua jangan suka menyakiti binatang ya....");
    AudioClip audio;
    AudioClip engl;
    Container c=new Container();

    public ayam(String sURL, String bhsEngl)
    {
```

```

super("Ayam");
setSize(690,470);
Dimension dim = Toolkit.getDefaultToolkit().getScreenSize();
int w = getSize().width;
int h = getSize().height;
int x = (dim.width - w) / 2;
int y = (dim.height - h) / 2;

setLocation(x,y);
setUndecorated(true);
setIconImage(logo.getImage());
try
{
    URL ling = new URL(bhsEngl);
    engl = Applet.newAudioClip(ling);
    URL url = new URL(sURL);
    audio = Applet.newAudioClip(url);
}
catch (MalformedURLException ex)
{
    System.err.println(ex.getMessage());
    System.exit(0);
}

}
void tampil()
{
    c=getContentPane();
    c.setLayout(null);
    c.setBackground(Color.BLACK);
    c.add(btnPlay);
    c.add(kembali);
    c.add(bahasa);
    c.add(ayam);
    c.add(ayam1);
    c.add(ayam2);
    c.add(ayam3);
    c.add(tarea);
}

```

```

ayam.setBounds(10,15,200,300);
  btnPlay.setBounds(50,300,140,20);
  btnPlay.setBackground(Color.black);
  btnPlay.setForeground(Color.WHITE);
  btnPlay.setFont(new Font("Century",Font.BOLD,12));
  kembali.setBounds(50,330,140,20);
  kembali.setBackground(Color.BLUE);
  kembali.setForeground(Color.WHITE);
  kembali.setFont(new Font("Century",Font.BOLD,12));
  bahasa.setBounds(50,360,140,20);
  bahasa.setBackground(Color.black);
  bahasa.setForeground(Color.white);
  bahasa.setFont(new Font("Century",Font.BOLD,12));
  ayam1.setBounds(220,320,100,100);
  ayam2.setBounds(390,320,100,100);
  ayam3.setBounds(550,320,100,100);
  tarea.setBounds(220,30,430,280);
  //tarea.setLineWrap(true);
  tarea.setEditable(false);

  setVisible(true);
}
void aksi()
{
  btnPlay.addActionListener(new ActionListener()
  {
    public void actionPerformed(ActionEvent e)
    {
      if(btnPlay==e.getSource())
        audio.play();
        engl.stop();
    }
  });
  bahasa.addActionListener(new ActionListener()
  {
    public void actionPerformed(ActionEvent e)
    {
      if(bahasa==e.getSource())
        engl.play();
        audio.stop();
    }
  });
}

```

```

    }
    });
    kembali.addActionListener(new ActionListener()
    {
        public void actionPerformed(ActionEvent e)
        {
            if(kembali==e.getSource())
            {
                audio.stop();
                engl.stop();
                peternakan pt=new peternakan();
                pt.nampil();
                pt.aksi();
                dispose();
            }
        }
    });
}
public static void main(String args[])
{
    SwingUtilities.invokeLater(new Runnable()
    {
        public void run()
        {
            ayam ne=new ayam("file:suara binatang/ayam.wav","file:inggris/ayam.wav");
            ne.aksi();
            ne.tampil();
        }
    });
}
}
}

```





PEMERINTAH KABUPATEN PROBOLINGGO
DINAS PENDIDIKAN
CABANG DINAS PENDIDIKAN
SEKOLAH DASAR NEGERI KALIREJO I No. 20
Jl. Kalimas No.20 Kalirejo I Telp. (0335) 432773
Dringu – Probolinggo

Analisis Pelajaran Ekstra Kulikuler : Komputer Kelas 1 Semester 1 Tahun Pelajaran : 2012 / 2013, Evaluasi : I

No	NILAI MAKSIMAL NOMOR SOAL NAMA SISWA	1	1	1	1	1	1	1	1	1	10	% Keter capaian Belajar	Ketuntasan		
		1	2	3	4	5	6	7	8	9	10		Belajar		
		SKOR YANG DIPEROLEH											Skor	Belajar	Ya
1	ANDHIKA	X										9	90	V	
2	DWIKI ALFIANSYAH											10	100	V	
3	NURCAHYA		X									9	90	V	
4	AMELIA NURSYAHDA			X								9	90	V	
5	ABD. ROHIM			X								9	90	V	
6	AHMAD K. ROZIQI				X							9	90	V	
7	ADIT DWI S.			X								9	90	V	
8	BIMA PRAYOGA	X			X							8	80	V	
9	ELI KURNIAWATI					X						9	90	V	
10	ERLIANA EFFY P.							X				9	90	V	
11	INTAN DWI NUR A.				X	X						8	80	V	
12	INTAN HIDAYATI	X				X						8	80	V	
13	JOAN DAVISKA				X							9	90	V	
14	KHUSNUL KHOTIMAH				X							9	90	V	
15	MOCH. FIRMAN	X				X						8	80	V	
16	MOCH. SUNARI				X							9	90	V	
17	MOCH. DONY A.	X			X							8	80	V	
18	M. ZENDI M.					X						9	90	V	
19	MARATUS SOLEHA											10	100	V	
20	M. ILHAM	X		X								8	80	V	
21	M. FERUL H.				X							9	90	V	
22	PUTRI NUR A.				X							9	90	V	
23	PAEDI H. Z.		X									9	90	V	
24	RATIH PURWASIH				X							9	90	V	
25	RIVALDI EKO F.		X			X						8	80	V	
26	SEPTI SRI W.	X			X							8	80	V	
27	SYARIFAH AMELIA A.			X								9	90	V	
28	SINDY AULIA				X							9	90	V	
29	VIRGI DAVA M.			X								9	90	V	
30	MAHESA T. P.						X					9	90	V	
Jumlah Skor		23	27	24	18	24	29	29	30	30	30	264	2640		
Jumlah Skor Maksimal		30	30	30	30	30	30	30	30	30	30	300	3000		
% Skor Tercapai		77	90	80	60	80	97	97	100	100	100	88		100	0

Ketuntasan belajar perorangan siswa dari 30 siswa. Ketuntasan belajar secara klasikal 100 % Ketuntasan belajar secara klasikal 100 %. Perlu perbaikan siswa belum tuntas 0 Siswa. Perlu perbaikan untuk soal no. 4.

Hasil Questioner :

Yang memiliki komputer di rumahnya = 2 siswa 7 %
 Yang tidak memiliki komputer di rumahnya = 28 siswa 93 %
 Yang senang dengan komputer = 29 siswa 97 %
 Yang senang dengan aplikasi pengenalan hewan = 29 siswa 97 %

Dringu, Juni 2013

Batasan Nilai :

10 - 40
50 - 70
80 - 100

Tidak Tuntas (Remidi)
Tuntas (Perlu Perbaikan/Pemberian Tugas)
Tuntas

