

UNIVERSITAS GUNADARMA



PRAKTIKUM REKAYASA PERANGKAT LUNAK 2

MANUAL BOOK

**“Aplikasi Manajemen Peternakan Bibilintik
menggunakan Java Hibernate”**

Disusun oleh:

- | | |
|-------------------------------|------------|
| 1. Akmal Alfarisi | (50414710) |
| 2. Indra Arianggi Suryaatmaja | (55414290) |
| 3. Thomi Alghani | (5A414736) |
| 4. Triswanto | (5A414883) |

Kelas : 4IA01

Penanggung Jawab (PJ) : Adrian Faisal

**DIAJUKAN GUNA MELENGKAPI SEBAGIAN
SYARAT PRAKTIKUM REKAYASA PERANGKAT LUNAK 2**

2017

BAB I

PENDAHULUAN

1.1. Latar Belakang

Peternakan merupakan bagian dari pembangunan pertanian dalam arti yang luas. Sektor peternakan merupakan faktor yang sama sekali tidak bisa dipandang sebelah mata karena sektor peternakan memberikan kontribusi yang sangat besar bagi indeks peningkatan daya beli masyarakat dan perekonomian masyarakat di Indonesia. Untuk mengakselerasi percepatan target pemenuhan populasi sapi dalam negeri, Kementerian Pertanian meluncurkan program Upaya Khusus Percepatan Populasi Sapi dan Kerbau Bunting (Upsus Siwab). Memandang hal tersebut dalam bidang Teknologi Informasi dibutuhkan sistem yang dapat membantu dalam mengatur dan mencatat setiap perkembangan ternak di peternakan. Berdasarkan data dari Sekolah Tinggi Penyuluhan Pertanian (STPP) Bogor, terdapat 74 peternakan sapi perah, 43 sapi potong, dan 51 peternakan kerbau.

Selama ini pendataan dan rekapitulasi ternak di peternakan daerah bogor masih dalam bentuk manual. Peternak dan penyuluh harus melakukan pengecekan satu per satu dengan banyaknya ternak pada setiap peternakan. Penyuluh akan melakukan pencatatan dan menghitung setiap jadwal masing-masing sapi sehingga peternak dan penyuluh sulit dalam menentukan produktifitas dari masing-masing sapi.

Pada penulisan ini, penulis akan membuat dan mengembangkan aplikasi manajemen peternakan. Aplikasi ini diharapkan dapat membantu dalam mengefektifkan reproduksi sapi ternak sehingga semakin meningkatkan populasi sapi daerah.

1.2. Batasan Masalah

Batasan masalah pada penulisan ini adalah pembuatan aplikasi manajemen peternakan di Peternakan Bibilintik Kota Bogor.

1.3. Tujuan Penulisan

Tujuan penulisan ini adalah untuk melengkapi syarat kelulusan ‘Rekayasa Perangkat Lunak 2’ dengan membuat aplikasi manajemen peternakan di Peternakan Bibilintik Kota Bogor.

BAB II

PEMBAHASAN

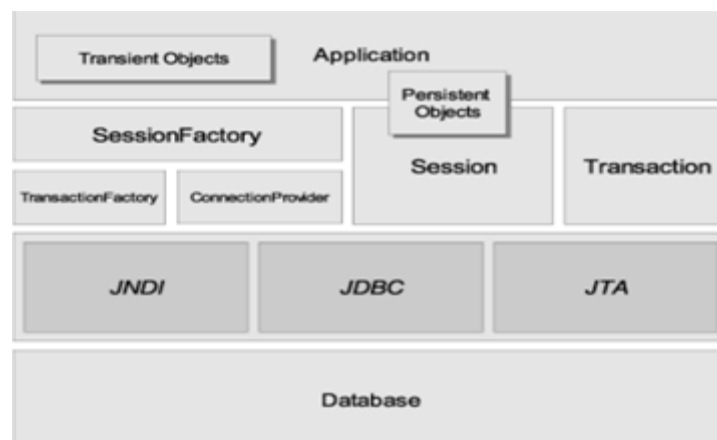
2.1. Java Hibernate

Hibernate adalah suatu alat yang digunakan untuk melakukan Object Relational Mapping (ORM) pada lingkungan java. Istilah ORM mengacu pada teknik untuk memetakan data yang terletak pada objek ke database relational. Hibernate tidak hanya mengatur pemetaan antara kelas kelas di java ke tabel di database, tapi juga memberikan mekanisme pengolahan data. Hal ini akan mempercepat proses pengembangan dibandingkan cara manual dengan melakukan penanganan data melalui SQL dan JDBC. Merupakan sebuah proyek Open Source profesional dan komponen terpenting dari Jboss Enterprises Middleware System.

Berikut ini merupakan fitur fitur yang terdapat pada Java Hibernate :

- Fitur pemetaan yang fleksibel
- Fitur pemetaan seperti inheritance dan polymorphism
- HQL yang bebas vendor database
- Assosiasi join secara otomatis
- Tidak akan mengupdate objek yang tidak dimodifikasi
- Mengurangi penulisan baris kode program

Berikut ini merupakan arsitektur Java Hibernate :



2.2. Netbeans

Netbeans merupakan sebuah aplikasi editor terpadu Integrated Development Environment (IDE) yang akan banyak mempermudah dalam membuat aplikasi karena menyediakan kontrol-kontrol visual yang penting dalam pemrograman desktop (atau lebih dikenal sebagai pemrograman visual). Netbeans berbasis Java dari Sun Microsystems yang berjalan di atas swing. Swing merupakan sebuah teknologi Java untuk pengembangan aplikasi desktop yang dapat berjalan pada berbagai macam platform seperti windows, linux, Mac OS X dan Solaris. Sebuah IDE merupakan lingkup pemrograman yang di integrasikan ke dalam suatu aplikasi perangkat lunak yang menyediakan Graphic User Interface (GUI), suatu kode editor atau text, suatu compiler dan suatu debugger.

2.3. MySQL

MySQL termasuk ke dalam RDBMS (Relational Database Management System), yang lebih populer dikalangan pemrograman web. MySQL dapat digunakan pada berbagai platform sistem operasi.

MySQL terdiri dari 2 (dua) lisensi, yaitu

- Lisensi Free (Free Software / Open Source GNU General Public License). Jenis lisensi ini bebas digunakan, dimodifikasi source programnya, dengan catatan harus dipublikasikan ke pemakai.
- Lisensi Komersial (Non – GPL). Pemakai harus membayar sejumlah biaya kepada MySQL AB sebagai pemegang hak cipta.

2.4. ERD (Entity Relationship Diagram)

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam database berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. Pada dasarnya terdapat tiga simbol yang digunakan pada ERD, yaitu ERD aplikasi manajemen peternakan.

2.5. UML

Unified Modelling Language (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. UML adalah himpunan struktur dan teknik untuk pemodelan desain object oriented programming (OOP) serta aplikasinya.

2.6. Struktur database

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	varchar(8)	latin1_swedish_ci		No	None			Change Drop Primary Unique Index Spatial More
2	berat	float			Yes	NULL			Change Drop Primary Unique Index Spatial More
3	jenis	varchar(25)	latin1_swedish_ci		Yes	NULL			Change Drop Primary Unique Index Spatial More
4	jenis_kelamin	varchar(11)	latin1_swedish_ci		Yes	NULL			Change Drop Primary Unique Index Spatial More
5	lingkar_dada	float			Yes	NULL			Change Drop Primary Unique Index Spatial More
6	panjang_badan	float			Yes	NULL			Change Drop Primary Unique Index Spatial More
7	umur	int(11)			Yes	NULL			Change Drop Primary Unique Index Spatial More

2.7. Rancangan Interface

ID

Lingkar Dada

Jenis Sapi

Panjang Badan

Jenis Kelamin

Jantan

Berat

Umur bulan

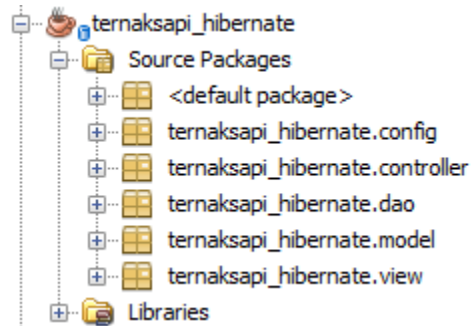
SIMPANUPDATEHAPUS

Title 1	Title 2	Title 3	Title 4

2.8. Pembuatan Program

2.8.1. Membuat Proyek Baru dan Menyiapkan Struktur Proyek

Berikut adalah struktur folder atau proyek program Aplikasi Manajemen Peternakan Bibilintik yang dibuat dengan menggunakan IDE Netbeans.



Terdapat beberapa package untuk mengelompokkan kode program sesuai dengan fungsinya masing-masing.

2.8.2. Kode Program dalam Default Package

Dalam default package terdapat file hibernate.cfg.xml, yang merupakan file konfigurasi untuk menghubungkan program dengan database mysql dan mapping struktur tabel yang didefinisikan oleh kode java ke dalam database mysql.

Isi kode programnya adalah sebagai berikut:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
  <session-factory>
    <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
    <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
    <property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/dbternaksapi?zeroDateTimeBehavior=conve
rtToNull</property>
    <property name="hibernate.connection.username">root</property>

    <property name="hibernate.hbm2ddl.auto">update</property>
    <property name="hibernate.show_sql">true</property>
    <property name="hibernate.format_sql">true</property>

    <mapping class="ternaksapi_hibernate.model.Sapi"></mapping>
  </session-factory>
</hibernate-configuration>
```

2.8.3. Kode Program dalam Package Config

Dalam package ini terdapat dua file, yaitu HibernateUtil.java yang menangani session dari hibernate terhadap mysql dan SapiTableModel.java yang memodelkan tampilan table pada interface program.

Berikut masing-masing isi kode programnya:

HibernateUtil.java

```
package ternaksapi_hibernate.config;

import org.hibernate.HibernateException;
import org.hibernate.SessionFactory;
import ternaksapi_hibernate.dao.SapiDao;
import ternaksapi_hibernate.dao.SapiDaoImpl;
import org.hibernate.cfg.Configuration;
import org.hibernate.HibernateException;

public class HibernateUtil {
    private static final SessionFactory SESSION_FACTORY;
    private static final SapiDao SAPI_DAO;

    static {
        try {
            SESSION_FACTORY = new Configuration().configure().buildSessionFactory();
            SAPI_DAO = new SapiDaoImpl(SESSION_FACTORY);
        } catch(HibernateException ex) {
            System.err.println("Initial SessionFactory creation failed." +ex);
            throw new ExceptionInInitializerError(ex);
        }
    }

    public static SessionFactory getSessionFactory(){
        return SESSION_FACTORY;
    }

    public static SapiDao getSapiDao() {
        return SAPI_DAO;
    }
}
```

SapiTableModel.java

```
package ternaksapi_hibernate.config;

import java.util.ArrayList;
import java.util.List;
import javax.swing.table.AbstractTableModel;
import ternaksapi_hibernate.model.Sapi;

public class SapiTableModel extends AbstractTableModel {

    private List<Sapi> listSapi = new ArrayList<>();
```

```

private final String HEADER[] = {"ID", "Jenis Sapi", "Jenis Kelamin", "Umur",
                                "Lingkar Dada", "Panjang Badan", "Berat"};

public SapiTableModel(List<Sapi> listSapi){
    this.listSapi = listSapi;
}

@Override
public int getRowCount() {
    return listSapi.size();
}

@Override
public int getColumnCount() {
    return HEADER.length;
}

@Override
public String getColumnName(int columnIndex) {
    return HEADER[columnIndex];
}

@Override
public Object getValueAt(int rowIndex, int columnIndex) {
    Sapi sapi = listSapi.get(rowIndex);

    switch(columnIndex) {
        case 0:
            return sapi.getId();
        case 1:
            return sapi.getJenis();
        case 2:
            return sapi.getJenis_kelamin();
        case 3:
            return sapi.getUmur();
        case 4:
            return sapi.getLingkar_dada();
        case 5:
            return sapi.getPanjang_badan();
        case 6:
            return sapi.getBerat();
        default:
            return null;
    }
}
}

```

2.8.4. Kode Program dalam Package Controller

Ada satu file kontroler bernama SapiController.java, yang merupakan logika transaksi data seperti simpan (save) data, modifikasi (update) data, dan hapus (delete) data.

```
package ternaksapi_hibernate.controller;
```



```

import java.util.List;
import javax.swing.JOptionPane;
import ternaksapi_hibernate.config.HibernateUtil;
import ternaksapi_hibernate.config.SapiTableModel;
import ternaksapi_hibernate.dao.SapiDao;
import ternaksapi_hibernate.model.Sapi;
import ternaksapi_hibernate.view.SapiView;

public class SapiController {

    private final SapiView sapiView;
    private SapiTableModel sapiTableModel;
    private List<Sapi> listSapi;
    private final SapiDao sapiDao = HibernateUtil.getSapiDao();

    public SapiController(SapiView bukuView) {
        this.sapiView = bukuView;
    }

    public void tampilData() {
        listSapi = sapiDao.getListSapi();
        sapiTableModel = new SapiTableModel(listSapi);
        this.sapiView.getTblSapi().setModel(sapiTableModel);
    }

    public void simpanSapi() {
        Sapi sapi = new Sapi();

        sapi.setId(this.sapiView.getId().getText());
        sapi.setJenis(this.sapiView.getJenis().getText());
        sapi.setJenis_kelamin(this.sapiView.getCbJenisKelamin().getSelectedItem().toString());
        sapi.setUmur(Integer.parseInt(this.sapiView.getUmur().getText()));
        sapi.setLingkar_dada(Float.parseFloat(this.sapiView.getLingkarDada().getText()));
        sapi.setPanjang_badan(Float.parseFloat(this.sapiView.getPanjangBadan().getText()));
        sapi.setBerat(Float.parseFloat(this.sapiView.getBerat().getText()));

        sapiDao.saveSapi(sapi);
        clearField();

        JOptionPane.showMessageDialog(null, "Data sapi berhasil disimpan", "Info",
            JOptionPane.INFORMATION_MESSAGE);
        tampilData();
    }

    public void updateSapi() {
        Sapi sapi = new Sapi();

        sapi.setId(this.sapiView.getId().getText());
        sapi.setJenis(this.sapiView.getJenis().getText());
        sapi.setJenis_kelamin(this.sapiView.getCbJenisKelamin().getSelectedItem().toString());
        sapi.setUmur(Integer.parseInt(this.sapiView.getUmur().getText()));
        sapi.setLingkar_dada(Float.parseFloat(this.sapiView.getLingkarDada().getText()));
        sapi.setPanjang_badan(Float.parseFloat(this.sapiView.getPanjangBadan().getText()));
        sapi.setBerat(Float.parseFloat(this.sapiView.getBerat().getText()));
    }
}

```

```

sapiDao.updateSapi(sapi);
clearField();

JOptionPane.showMessageDialog(null, "Data sapi berhasil diupdate", "Info",
    JOptionPane.INFORMATION_MESSAGE);
tampilData();
}

public void deleteSapi() {
    Sapi sapi = new Sapi();

    sapi.setId(this.sapiView.getId().getText());
    sapi.setJenis(this.sapiView.getJenis().getText());
    sapi.setJenis_kelamin(this.sapiView.getCbJenisKelamin().getSelectedItem().toString());
    sapi.setUmur(Integer.parseInt(this.sapiView.getTxtUmur().getText()));
    sapi.setLingkar_dada(Float.parseFloat(this.sapiView.getTxtLingkarDada().getText()));
    sapi.setPanjang_badan(Float.parseFloat(this.sapiView.getTxtPanjangBadan().getText()));
    sapi.setBerat(Float.parseFloat(this.sapiView.getTxtBerat().getText()));

    sapiDao.deleteSapi(sapi);
    clearField();

    JOptionPane.showMessageDialog(null, "Data sapi berhasil dihapus", "Info",
        JOptionPane.INFORMATION_MESSAGE);
    tampilData();
}

public void cekSapi() {
    int index = this.sapiView.getTblSapi().getSelectedRow();

    this.sapiView.getId().setText(
        String.valueOf(this.sapiView.getTblSapi()
            .getValueAt(index, 0))
    );
    this.sapiView.getId().setEditable(false);

    this.sapiView.getJenis().setText(
        String.valueOf(this.sapiView.getTblSapi()
            .getValueAt(index, 1))
    );

    this.sapiView.getCbJenisKelamin().setSelectedItem(
        String.valueOf(this.sapiView.getTblSapi()
            .getValueAt(index, 2))
    );

    this.sapiView.getTxtUmur().setText(
        String.valueOf(this.sapiView.getTblSapi()
            .getValueAt(index, 3))
    );

    this.sapiView.getTxtLingkarDada().setText(
        String.valueOf(this.sapiView.getTblSapi()
            .getValueAt(index, 4))
    );
}

```

```

        this.sapiView.getTxtPanjangBadan().setText(
            String.valueOf(this.sapiView.getTblSapi()
                .getValueAt(index, 5))
        );

        this.sapiView.getTxtBerat().setText(
            String.valueOf(this.sapiView.getTblSapi()
                .getValueAt(index, 6))
        );
    }

    private void clearField() {
        this.sapiView.getTxtId().setEditable(true);
        this.sapiView.getTxtId().setText(null);
        this.sapiView.getTxtJenis().setText(null);
        this.sapiView.getCbJenisKelamin().setSelectedIndex(0);
        this.sapiView.getTxtUmur().setText(null);
        this.sapiView.getTxtLingkarDada().setText(null);
        this.sapiView.getTxtPanjangBadan().setText(null);
        this.sapiView.getTxtBerat().setText(null);
    }
}

```

2.8.5. Kode Program dalam Package Dao

Terdapat 2 file, SapiDao.java merupakan interface yang akan dipanggil oleh kontroler, dan SapiDaoImpl.java yang menjalankan instruksi CRUD data.

SapiDao.java

```

package ternaksapi_hibernate.dao;

import java.util.List;
import ternaksapi_hibernate.model.Sapi;

public interface SapiDao {
    public void saveSapi(Sapi sapi);
    public void updateSapi(Sapi sapi);
    public void deleteSapi(Sapi sapi);
    public List<Sapi> getListSapi();
}

```

SapiDaoImpl.java

```

package ternaksapi_hibernate.dao;

import java.util.List;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import ternaksapi_hibernate.model.Sapi;

public class SapiDaoImpl implements SapiDao {

```

```

private final SessionFactory sessionFactory;

public SapiDaoImpl(SessionFactory sessionFactory) {
    this.sessionFactory = sessionFactory;
}

@Override
public void saveSapi(Sapi sapi) {
    Session session = sessionFactory.openSession();
    try {
        session.beginTransaction();
        session.save(sapi);
        session.getTransaction().commit();
    } catch (Exception e) {
        session.getTransaction().rollback();
    } finally {
        session.close();
    }
}

@Override
public void updateSapi(Sapi sapi) {
    Session session = sessionFactory.openSession();
    try {
        session.beginTransaction();
        session.update(sapi);
        session.getTransaction().commit();
    } catch (Exception e) {
        session.getTransaction().rollback();
    } finally {
        session.close();
    }
}

@Override
public void deleteSapi(Sapi sapi) {
    Session session = sessionFactory.openSession();
    try {
        session.beginTransaction();
        session.delete(sapi);
        session.getTransaction().commit();
    } catch (Exception e) {
        session.getTransaction().rollback();
    } finally {
        session.close();
    }
}

@Override
public List<Sapi> getListSapi() {
    Session session = sessionFactory.openSession();
    try {
        session.beginTransaction();
        List<Sapi> listSapi;
        listSapi = session.createCriteria(Sapi.class).list();
        session.getTransaction().commit();
    }
}

```

```

        return listSapi;
    } catch(Exception e) {
        session.getTransaction().rollback();
        return null;
    } finally {
        session.close();
    }
}
}

```

2.8.6. Kode Program dalam Package Model

Terdapat file Sapi.java, ditujukan untuk mendefinisikan struktur tabel dalam database.

Skema datanya terdiri dari id, jenis, jenis kelamin, umur, lingkar dada, panjang badan, dan berat.

```

package ternaksapi_hibernate.model;

import java.io.Serializable;
import javax.persistence.Entity;
import javax.persistence.Table;
import javax.persistence.Column;
import javax.persistence.Id;

@Entity
@Table(name = "tbl_sapi")
public class Sapi implements Serializable {

    @Id
    @Column(name = "id", length = 8)
    private String id;

    @Column(name = "jenis", length = 25)
    private String jenis;

    @Column(name = "jenis_kelamin", length = 11)
    private String jenis_kelamin;

    @Column(name = "umur")
    private int umur;

    @Column(name = "lingkar_dada")
    private float lingkar_dada;

    @Column(name = "panjang_badan")
    private float panjang_badan;

    @Column(name = "berat")
    private float berat;

    /**
     * @return the id

```

```
*/
public String getId() {
    return id;
}

/**
 * @param id the id to set
 */
public void setId(String id) {
    this.id = id;
}

/**
 * @return the jenis
 */
public String getJenis() {
    return jenis;
}

/**
 * @param jenis the jenis to set
 */
public void setJenis(String jenis) {
    this.jenis = jenis;
}

/**
 * @return the jenis_kelamin
 */
public String getJenis_kelamin() {
    return jenis_kelamin;
}

/**
 * @param jenis_kelamin the jenis_kelamin to set
 */
public void setJenis_kelamin(String jenis_kelamin) {
    this.jenis_kelamin = jenis_kelamin;
}

/**
 * @return the lingkaran_dada
 */
public float getLingkaran_dada() {
    return lingkaran_dada;
}

/**
 * @param lingkaran_dada the lingkaran_dada to set
 */
public void setLingkaran_dada(float lingkaran_dada) {
    this.lingkaran_dada = lingkaran_dada;
}

/**
 * @return the panjang_badan
```

```

    */
    public float getPanjang_badan() {
        return panjang_badan;
    }

    /**
     * @param panjang_badan the panjang_badan to set
     */
    public void setPanjang_badan(float panjang_badan) {
        this.panjang_badan = panjang_badan;
    }

    /**
     * @return the umur
     */
    public int getUmur() {
        return umur;
    }

    /**
     * @param umur the umur to set
     */
    public void setUmur(int umur) {
        this.umur = umur;
    }

    /**
     * @return the berat
     */
    public float getBerat() {
        return berat;
    }

    /**
     * @param berat the berat to set
     */
    public void setBerat(float berat) {
        this.berat = berat;
    }
}

```

2.8.7. Kode Program dalam Package View

Terdapat file SapiView.java, yang mendefinisikan interface program. Tampilan interfacenya sebagai berikut:

ID		Lingkar Dada	
Jenis Sapi		Panjang Badan	
Jenis Kelamin	Jantan	Berat	
Umur			

SIMPAN UPDATE HAPUS

Title 1	Title 2	Title 3	Title 4

Dalam interface tersebut terdapat 6 text field dan 1 combo box untuk menginputkan maupun modifikasi data, serta sebuah tabel untuk menampilkan data-data yang tersimpan dalam database.

Sedangkan isi kode program SapiView.java adalah sebagai berikut:

```
package ternaksapi_hibernate.view;

import ternaksapi_hibernate.controller.SapiController;

public class SapiView extends javax.swing.JFrame {

    private final SapiController sapiController = new SapiController(this);

    /**
     * Creates new form SapiView
     */
    public SapiView() {
        initComponents();
    }
}
```



```

}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    txtId = new javax.swing.JTextField();
    txtJenis = new javax.swing.JTextField();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    txtUmur = new javax.swing.JTextField();
    jLabel6 = new javax.swing.JLabel();
    cbJenisKelamin = new javax.swing.JComboBox();
    jLabel7 = new javax.swing.JLabel();
    jLabel8 = new javax.swing.JLabel();
    jLabel9 = new javax.swing.JLabel();
    txtLingkarDada = new javax.swing.JTextField();
    txtPanjangBadan = new javax.swing.JTextField();
    txtBerat = new javax.swing.JTextField();
    jScrollPane1 = new javax.swing.JScrollPane();
    tblSapi = new javax.swing.JTable();
    btnSimpan = new javax.swing.JButton();
    btnUpdate = new javax.swing.JButton();
    btnHapus = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jLabel1.setText("ID");

    jLabel2.setText("Jenis Sapi");

    jLabel3.setText("Jenis Kelamin");

    txtJenis.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            txtJenisActionPerformed(evt);
        }
    });

    jLabel5.setText("Umur");

    jLabel6.setText("bulan");

    cbJenisKelamin.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "Jantan",
"Betina" }));

    jLabel7.setText("Lingkar Dada");

```

```
jLabel8.setText("Panjang Badan");

jLabel9.setText("Berat");

tblSapi.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "Title 1", "Title 2", "Title 3", "Title 4"
    }
));
tblSapi.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        tblSapiMouseClicked(evt);
    }
});
jScrollPane1.setViewportViewView(tblSapi);

btnSimpan.setText("SIMPAN");
btnSimpan.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnSimpanActionPerformed(evt);
    }
});

btnUpdate.setText("UPDATE");
btnUpdate.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnUpdateActionPerformed(evt);
    }
});

btnHapus.setText("HAPUS");
btnHapus.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnHapusActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .add(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .add(jLabel1)
            .add(jLabel2)
            .add(jLabel3)
            .add(jLabel4)
            .add(jLabel5)
            .add(jLabel6)
            .add(jLabel7)
            .add(jLabel8)
            .add(jLabel9)
            .add(btnKirim)
            .add(btnBatal)
            .add(btnHapus)
            .add(btnSimpan)
            .add(btnUpdate)
            .add(btnTambah)
            .add(btnEdit)
            .add(btnDelete)
            .add(btnCancel)
            .add(btnOK)
            .add(btnClose)
            .add(btnPrint)
            .add(btnExport)
            .add(btnImport)
            .add(btnRefresh)
            .add(btnSearch)
            .add(btnFilter)
            .add(btnSort)
            .add(btnGroupBy)
            .add(btnPivotTable)
            .add(btnChart)
            .add(btnForm)
            .add(btnTable)
            .add(btnQuery)
            .add(btnReport)
            .add(btnDashboard)
            .add(btnMenu)
            .add(btnHelp)
            .add(btnAbout)
            .add(btnSettings)
            .add(btnLanguage)
            .add(btnTheme)
            .add(btnFont)
            .add(btnColor)
            .add(btnSize)
            .add(btnWeight)
            .add(btnStyle)
            .add(btnAlign)
            .add(btnText)
            .add(btnImage)
            .add(btnVideo)
            .add(btnAudio)
            .add(btnLink)
            .add(btnEmail)
            .add(btnPhone)
            .add(btnAddress)
            .add(btnDate)
            .add(btnTime)
            .add(btnClock)
            .add(btnCalendar)
            .add(btnWeather)
            .add(btnNews)
            .add(btnSports)
            .add(btnEntertainment)
            .add(btnScience)
            .add(btnTechnology)
            .add(btnHealth)
            .add(btnFinance)
            .add(btnLaw)
            .add(btnEducation)
            .add(btnHistory)
            .add(btnCulture)
            .add(btnArt)
            .add(btnMusic)
            .add(btnDance)
            .add(btnTheater)
            .add(btnFilm)
            .add(btnTV)
            .add(btnRadio)
            .add(btnNewspaper)
            .add(btnMagazine)
            .add(btnBook)
            .add(btnComic)
            .add(btnManga)
            .add(btnAnime)
            .add(btnGame)
            .add(btnApp)
            .add(btnSoftware)
            .add(btnHardware)
            .add(btnNetwork)
            .add(btnSecurity)
            .add(btnPrivacy)
            .add(btnEncryption)
            .add(btnAuthentication)
            .add(btnAuthorization)
            .add(btnAccessControl)
            .add(btnDataProtection)
            .add(btnBackup)
            .add(btnRecovery)
            .add(btnDisasterRecovery)
            .add(btnBusinessContinuity)
            .add(btnRiskManagement)
            .add(btnCompliance)
            .add(btnGovernance)
            .add(btnPolicy)
            .add(btnProcedure)
            .add(btnStandard)
            .add(btnSpecification)
            .add(btnRequirement)
            .add(btnConstraint)
            .add(btnAssessment)
            .add(btnEvaluation)
            .add(btnVerification)
            .add(btnValidation)
            .add(btnTesting)
            .add(btnDeployment)
            .add(btnMonitoring)
            .add(btnMaintenance)
            .add(btnSupport)
            .add(btnTraining)
            .add(btnDocumentation)
            .add(btnCommunication)
            .add(btnCollaboration)
            .add(btnTeamwork)
            .add(btnLeadership)
            .add(btnManagement)
            .add(btnStrategy)
            .add(btnPlanning)
            .add(btnExecution)
            .add(btnReview)
            .add(btnImprovement)
            .add(btnInnovation)
            .add(btnCreativity)
            .add(btnProblemSolving)
            .add(btnDecisionMaking)
            .add(btnConflictResolution)
            .add(btnNegotiation)
            .add(btnMediation)
            .add(btnArbitration)
            .add(btnLitigation)
            .add(btnJudicialProcess)
            .add(btnLegalSystem)
            .add(btnGovernment)
            .add(btnPolitics)
            .add(btnSocialIssues)
            .add(btnHumanRights)
            .add(btnEnvironment)
            .add(btnClimateChange)
            .add(btnPollution)
            .add(btnConservation)
            .add(btnWildlife)
            .add(btnMarineLife)
            .add(btnTerrestrialLife)
            .add(btnBotany)
            .add(btnZoology)
            .add(btnAnatomy)
            .add(btnPhysiology)
            .add(btnBiochemistry)
            .add(btnGenetics)
            .add(btnEvolution)
            .add(btnEcology)
            .add(btnEnvironmentalScience)
            .add(btnEarthScience)
            .add(btnGeology)
            .add(btnMeteorology)
            .add(btnAstronomy)
            .add(btnCosmology)
            .add(btnPhysics)
            .add(btnChemistry)
            .add(btnBiology)
            .add(btnMedicine)
            .add(btnNursing)
            .add(btnPharmacy)
            .add(btnVeterinary)
            .add(btnAgriculture)
            .add(btnForestry)
            .add(btnFishing)
            .add(btnHunting)
            .add(btnGardening)
            .add(btnLandscaping)
            .add(btnArchitecture)
            .add(btnInteriorDesign)
            .add(btnFashion)
            .add(btnBeauty)
            .add(btnHair)
            .add(btnSkincare)
            .add(btnCosmetics)
            .add(btnJewelry)
            .add(btnWatches)
            .add(btnShoes)
            .add(btnClothing)
            .add(btnAccessories)
            .add(btnBags)
            .add(btnLuggage)
            .add(btnTravel)
            .add(btnHotels)
            .add(btnRestaurants)
            .add(btnCafes)
            .add(btnBars)
            .add(btnClubs)
            .add(btnNightlife)
            .add(btnEvents)
            .add(btnConferences)
            .add(btnWorkshops)
            .add(btnCourses)
            .add(btnClasses)
            .add(btnTutorials)
            .add(btnWebinars)
            .add(btnPodcasts)
            .add(btnYouTube)
            .add(btnInstagram)
            .add(btnFacebook)
            .add(btnTwitter)
            .add(btnLinkedIn)
            .add(btnPinterest)
            .add(btnSnapchat)
            .add(btnWhatsApp)
            .add(btnTelegram)
            .add(btnSignal)
            .add(btnZoo)
            .add(btnAquarium)
            .add(btnPlanetarium)
            .add(btnObservatory)
            .add(btnMuseum)
            .add(btnLibrary)
            .add(btnArchive)
            .add(btnRepository)
            .add(btnDatabase)
            .add(btnCloudStorage)
            .add(btnFileSharing)
            .add(btnCollaborativeTools)
            .add(btnProductivityTools)
            .add(btnProjectManagement)
            .add(btnTaskManagement)
            .add(btnTimeManagement)
            .add(btnResourceManagement)
            .add(btnInventoryManagement)
            .add(btnSupplyChainManagement)
            .add(btnLogisticsManagement)
            .add(btnTransportationManagement)
            .add(btnWarehousingManagement)
            .add(btnManufacturingManagement)
            .add(btnQualityManagement)
            .add(btnSafetyManagement)
            .add(btnSecurityManagement)
            .add(btnRiskManagement)
            .add(btnComplianceManagement)
            .add(btnGovernanceManagement)
            .add(btnPolicyManagement)
            .add(btnProcedureManagement)
            .add(btnStandardManagement)
            .add(btnSpecificationManagement)
            .add(btnRequirementManagement)
            .add(btnConstraintManagement)
            .add(btnAssessmentManagement)
            .add(btnEvaluationManagement)
            .add(btnVerificationManagement)
            .add(btnValidationManagement)
            .add(btnTestingManagement)
            .add(btnDeploymentManagement)
            .add(btnMonitoringManagement)
            .add(btnMaintenanceManagement)
            .add(btnSupportManagement)
            .add(btnTrainingManagement)
            .add(btnDocumentationManagement)
            .add(btnCommunicationManagement)
            .add(btnCollaborationManagement)
            .add(btnTeamworkManagement)
            .add(btnLeadershipManagement)
            .add(btnManagementManagement)
            .add(btnStrategyManagement)
            .add(btnPlanningManagement)
            .add(btnExecutionManagement)
            .add(btnReviewManagement)
            .add(btnImprovementManagement)
            .add(btnInnovationManagement)
            .add(btnCreativityManagement)
            .add(btnProblemSolvingManagement)
            .add(btnDecisionMakingManagement)
            .add(btnConflictResolutionManagement)
            .add(btnNegotiationManagement)
            .add(btnMediationManagement)
            .add(btnArbitrationManagement)
            .add(btnLitigationManagement)
            .add(btnJudicialProcessManagement)
            .add(btnLegalSystemManagement)
            .add(btnGovernmentManagement)
            .add(btnPoliticsManagement)
            .add(btnSocialIssuesManagement)
            .add(btnHumanRightsManagement)
            .add(btnEnvironmentManagement)
            .add(btnClimateChangeManagement)
            .add(btnPollutionManagement)
            .add(btnConservationManagement)
            .add(btnWildlifeManagement)
            .add(btnMarineLifeManagement)
            .add(btnTerrestrialLifeManagement)
            .add(btnBotanyManagement)
            .add(btnZoologyManagement)
            .add(btnAnatomyManagement)
            .add(btnPhysiologyManagement)
            .add(btnBiochemistryManagement)
            .add(btnGeneticsManagement)
            .add(btnEvolutionManagement)
            .add(btnEcologyManagement)
            .add(btnEnvironmentalScienceManagement)
            .add(btnEarthScienceManagement)
            .add(btnGeologyManagement)
            .add(btnMeteorologyManagement)
            .add(btnAstronomyManagement)
            .add(btnCosmologyManagement)
            .add(btnPhysicsManagement)
            .add(btnChemistryManagement)
            .add(btnBiologyManagement)
            .add(btnMedicineManagement)
            .add(btnNursingManagement)
            .add(btnPharmacyManagement)
            .add(btnVeterinaryManagement)
            .add(btnAgricultureManagement)
            .add(btnForestryManagement)
            .add(btnFishingManagement)
            .add(btnHuntingManagement)
            .add(btnGardeningManagement)
            .add(btnLandscapingManagement)
            .add(btnArchitectureManagement)
            .add(btnInteriorDesignManagement)
            .add(btnFashionManagement)
            .add(btnBeautyManagement)
            .add(btnHairManagement)
            .add(btnSkincareManagement)
            .add(btnCosmeticsManagement)
            .add(btnJewelryManagement)
            .add(btnWatchesManagement)
            .add(btnShoesManagement)
            .add(btnClothingManagement)
            .add(btnAccessoriesManagement)
            .add(btnBagsManagement)
            .add(btnLuggageManagement)
            .add(btnTravelManagement)
            .add(btnHotelsManagement)
            .add(btnRestaurantsManagement)
            .add(btnCafesManagement)
            .add(btnBarsManagement)
            .add(btnClubsManagement)
            .add(btnNightlifeManagement)
            .add(btnEventsManagement)
            .add(btnConferencesManagement)
            .add(btnWorkshopsManagement)
            .add(btnCoursesManagement)
            .add(btnClassesManagement)
            .add(btnTutorialsManagement)
            .add(btnWebinarsManagement)
            .add(btnPodcastsManagement)
            .add(btnYouTubeManagement)
            .add(btnInstagramManagement)
            .add(btnFacebookManagement)
            .add(btnTwitterManagement)
            .add(btnLinkedInManagement)
            .add(btnPinterestManagement)
            .add(btnSnapchatManagement)
            .add(btnWhatsAppManagement)
            .add(btnTelegramManagement)
            .add(btnSignalManagement)
            .add(btnZooManagement)
            .add(btnAquariumManagement)
            .add(btnPlanetariumManagement)
            .add(btnObservatoryManagement)
            .add(btnMuseumManagement)
            .add(btnLibraryManagement)
            .add(btnArchiveManagement)
            .add(btnRepositoryManagement)
            .add(btnDatabaseManagement)
            .add(btnCloudStorageManagement)
            .add(btnFileSharingManagement)
            .add(btnCollaborativeToolsManagement)
            .add(btnProductivityToolsManagement)
            .add(btnProjectManagementManagement)
            .add(btnTaskManagementManagement)
            .add(btnTimeManagementManagement)
            .add(btnResourceManagementManagement)
            .add(btnInventoryManagementManagement)
            .add(btnSupplyChainManagementManagement)
            .add(btnLogisticsManagementManagement)
            .add(btnTransportationManagementManagement)
            .add(btnWarehousingManagementManagement)
            .add(btnManufacturingManagementManagement)
            .add(btnQualityManagementManagement)
            .add(btnSafetyManagementManagement)
            .add(btnSecurityManagementManagement)
            .add(btnRiskManagementManagement)
            .add(btnComplianceManagementManagement)
            .add(btnGovernanceManagementManagement)
            .add(btnPolicyManagementManagement)
            .add(btnProcedureManagementManagement)
            .add(btnStandardManagementManagement)
            .add(btnSpecificationManagementManagement)
            .add(btnRequirementManagementManagement)
            .add(btnConstraintManagementManagement)
            .add(btnAssessmentManagementManagement)
            .add(btnEvaluationManagementManagement)
            .add(btnVerificationManagementManagement)
            .add(btnValidationManagementManagement)
            .add(btnTestingManagementManagement)
            .add(btnDeploymentManagementManagement)
            .add(btnMonitoringManagementManagement)
            .add(btnMaintenanceManagementManagement)
            .add(btnSupportManagementManagement)
            .add(btnTrainingManagementManagement)
            .add(btnDocumentationManagementManagement)
            .add(btnCommunicationManagementManagement)
            .add(btnCollaborationManagementManagement)
            .add(btnTeamworkManagementManagement)
            .add(btnLeadershipManagementManagement)
            .add(btnManagementManagementManagement)
            .add(btnStrategyManagementManagement)
            .add(btnPlanningManagementManagement)
            .add(btnExecutionManagementManagement)
            .add(btnReviewManagementManagement)
            .add(btnImprovementManagementManagement)
            .add(btnInnovationManagementManagement)
            .add(btnCreativityManagementManagement)
            .add(btnProblemSolvingManagementManagement)
            .add(btnDecisionMakingManagementManagement)
            .add(btnConflictResolutionManagementManagement)
            .add(btnNegotiationManagementManagement)
            .add(btnMediationManagementManagement)
            .add(btnArbitrationManagementManagement)
            .add(btnLitigationManagementManagement)
            .add(btnJudicialProcessManagementManagement)
            .add(btnLegalSystemManagementManagement)
            .add(btnGovernmentManagementManagement)
            .add(btnPoliticsManagementManagement)
            .add(btnSocialIssuesManagementManagement)
            .add(btnHumanRightsManagementManagement)
            .add(btnEnvironmentManagementManagement)
            .add(btnClimateChangeManagementManagement)
            .add(btnPollutionManagementManagement)
            .add(btnConservationManagementManagement)
            .add(btnWildlifeManagementManagement)
            .add(btnMarineLifeManagementManagement)
            .add(btnTerrestrialLifeManagementManagement)
            .add(btnBotanyManagementManagement)
            .add(btnZoologyManagementManagement)
            .add(btnAnatomyManagementManagement)
            .add(btnPhysiologyManagementManagement)
            .add(btnBiochemistryManagementManagement)
            .add(btnGeneticsManagementManagement)
            .add(btnEvolutionManagementManagement)
            .add(btnEcologyManagementManagement)
            .add(btnEnvironmentalScienceManagementManagement)
            .add(btnEarthScienceManagementManagement)
            .add(btnGeologyManagementManagement)
            .add(btnMeteorology
```

```

        .addComponent(jLabel2)
        .addComponent(jLabel3)
        .addComponent(jLabel5))
    .addGap(35, 35, 35)
    .addGroup(layout.createParallelGroup(GroupLayout.Alignment.LEADING,
false)
        .addComponent(txtId)
        .addComponent(txtJenis)
        .addGroup(layout.createSequentialGroup()
            .addComponent(txtUmur,
                GroupLayout.PREFERRED_SIZE, 70,
                GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(GroupLayout.ComponentPlacement.RELATED)
            .addComponent(jLabel6))
        .addComponent(cbJenisKelamin, 0, 120, Short.MAX_VALUE))
    .addGap(119, 119, 119)
    .addGroup(layout.createParallelGroup(GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addComponent(jLabel7)
            .addGap(71, 71, 71)
            .addComponent(jLabel4))
        .addComponent(jLabel8)
        .addComponent(jLabel9))
    .addPreferredGap(GroupLayout.ComponentPlacement.RELATED)
    .addGroup(layout.createParallelGroup(GroupLayout.Alignment.LEADING,
false)
        .addComponent(txtLingkarDada,
            GroupLayout.DEFAULT_SIZE, 120,
            Short.MAX_VALUE)
        .addComponent(txtPanjangBadan)
        .addComponent(txtBerat))
    .addGap(0, 0, Short.MAX_VALUE))
    .addGroup(layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(btnSimpan,
            GroupLayout.PREFERRED_SIZE, 216,
            GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(GroupLayout.ComponentPlacement.UNRELATED)
        .addComponent(btnUpdate,
            GroupLayout.DEFAULT_SIZE, 216,
            Short.MAX_VALUE)
        .addPreferredGap(GroupLayout.ComponentPlacement.UNRELATED)
        .addComponent(btnHapus,
            GroupLayout.PREFERRED_SIZE, 216,
            GroupLayout.PREFERRED_SIZE)))
    .addContainerGap()
);
layout.setVerticalGroup(
    layout.createParallelGroup(GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGap(30, 30, 30)
            .addGroup(layout.createParallelGroup(GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel1)
                .addComponent(txtId,
                    GroupLayout.PREFERRED_SIZE,
                    GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel4)
                .addComponent(jLabel7)
                .addComponent(txtLingkarDada,
                    GroupLayout.PREFERRED_SIZE,
                    GroupLayout.DEFAULT_SIZE, GroupLayout.PREFERRED_SIZE))
            .addGap(18, 18, 18)
            .addGroup(layout.createParallelGroup(GroupLayout.Alignment.BASELINE)

```

```

        .addComponent(jLabel2)
        .addComponent(txtJenis, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel8)
        .addComponent(txtPanjangBadan, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel3)
        .addComponent(cbJenisKelamin, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel9)
        .addComponent(txtBerat, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(25, 25, 25)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel5)
        .addComponent(txtUmur, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jLabel6))
        .addGap(18, 18, 18)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(btnSimpan)
        .addComponent(btnUpdate)
        .addComponent(btnHapus))
        .addGap(14, 14, 14)
        .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE, 293,
Short.MAX_VALUE)
        .addContainerGap())
    );

    pack();
} // </editor-fold>

private void txtJenisActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

private void tblSapiMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    sapiController.cekSapi();
}

private void btnSimpanActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    sapiController.simpanSapi();
}

private void btnUpdateActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    sapiController.updateSapi();
}

private void btnHapusActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    sapiController.deleteSapi();
}

```

```

}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
     * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
     */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(SapiView.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(SapiView.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(SapiView.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(SapiView.class.getName()).log(java.util.logging.Level.SEVERE, null,
ex);
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new SapiView().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton btnHapus;
private javax.swing.JButton btnSimpan;
private javax.swing.JButton btnUpdate;
private javax.swing.JComboBox cbJenisKelamin;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;

```

```
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable tblSapi;
private javax.swing.JTextField txtBerat;
private javax.swing.JTextField txtId;
private javax.swing.JTextField txtJenis;
private javax.swing.JTextField txtLingkarDada;
private javax.swing.JTextField txtPanjangBadan;
private javax.swing.JTextField txtUmur;
// End of variables declaration

/**
 * @return the tblSapi
 */
public javax.swing.JTable getTblSapi() {
    return tblSapi;
}

/**
 * @return the txtBerat
 */
public javax.swing.JTextField getTxtBerat() {
    return txtBerat;
}

/**
 * @return the txtId
 */
public javax.swing.JTextField getTxtId() {
    return txtId;
}

/**
 * @return the txtJenis
 */
public javax.swing.JTextField getTxtJenis() {
    return txtJenis;
}

/**
 * @return the txtLingkarDada
 */
public javax.swing.JTextField getTxtLingkarDada() {
    return txtLingkarDada;
}

/**
 * @return the txtPanjangBadan
 */
public javax.swing.JTextField getTxtPanjangBadan() {
    return txtPanjangBadan;
}
```

```

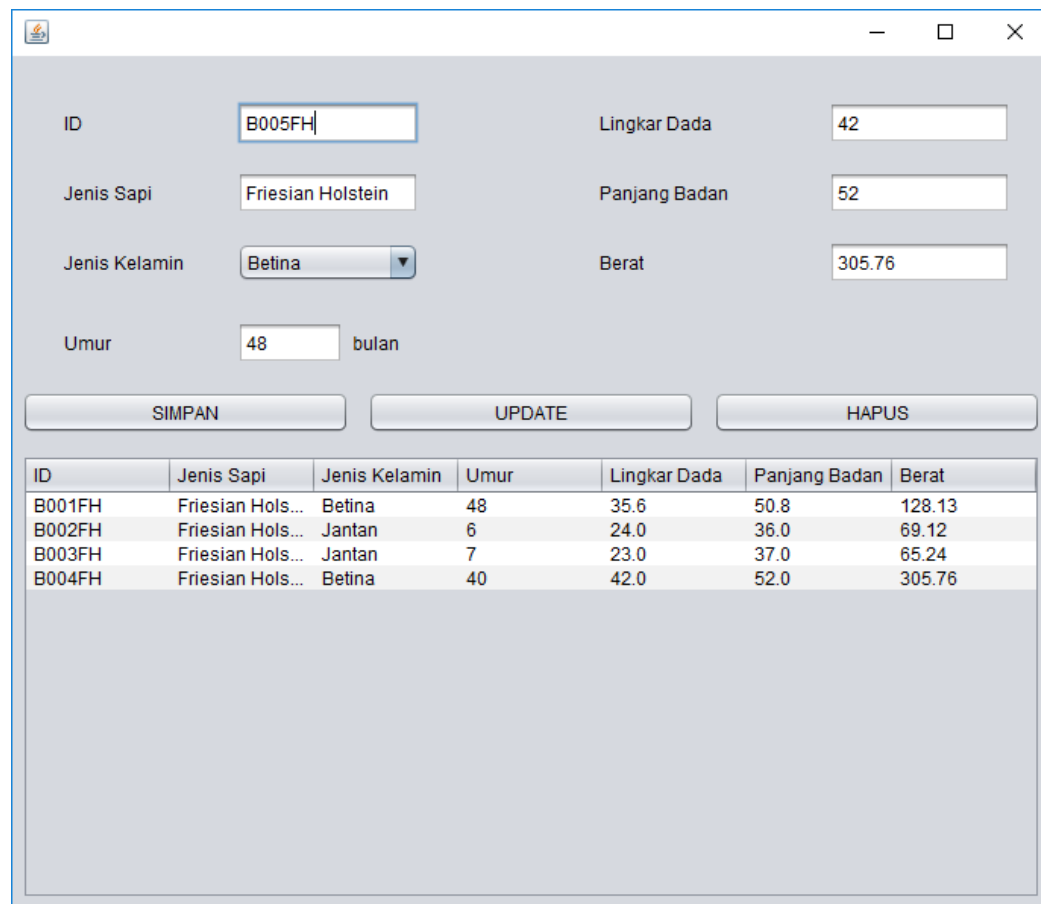
/**
 * @return the txtUmur
 */
public javax.swing.JTextField getTxtUmur() {
    return txtUmur;
}

/**
 * @return the cbJenisKelamin
 */
public javax.swing.JComboBox getCbJenisKelamin() {
    return cbJenisKelamin;
}
}

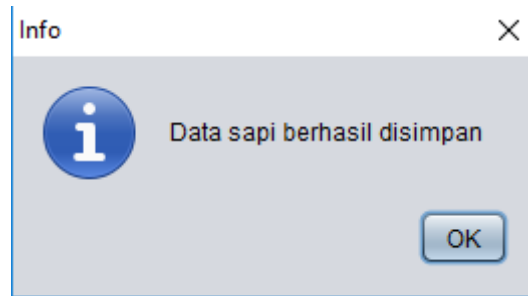
```

2.9. Output Aplikasi

2.9.1. Insert Ternak



ID	Jenis Sapi	Jenis Kelamin	Umur	Lingkar Dada	Panjang Badan	Berat
B001FH	Friesian Hols...	Betina	48	35.6	50.8	128.13
B002FH	Friesian Hols...	Jantan	6	24.0	36.0	69.12
B003FH	Friesian Hols...	Jantan	7	23.0	37.0	65.24
B004FH	Friesian Hols...	Betina	40	42.0	52.0	305.76



2.9.2. Update Ternak

Update Ternak

ID: B003FH

Jenis Sapi: Friesian Holstein

Jenis Kelamin: Jantan

Umur: 30 bulan

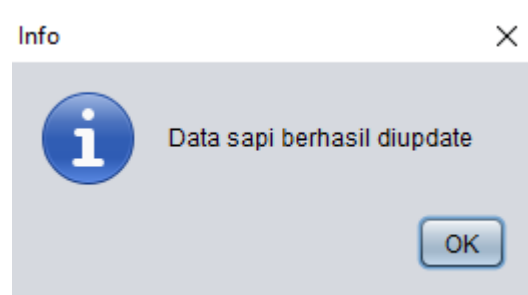
Lingkar Dada: 40.58

Panjang Badan: 51.65

Berat: 298.57

SIMPAN UPDATE HAPUS

ID	Jenis Sapi	Jenis Kelamin	Umur	Lingkar Dada	Panjang Badan	Berat
B001FH	Friesian Hols...	Betina	48	35.6	50.8	128.13
B002FH	Friesian Hols...	Jantan	6	24.0	36.0	69.12
B003FH	Friesian Hols...	Jantan	30	40.58	51.65	298.57
B004FH	Friesian Hols...	Betina	40	42.0	52.0	305.76
B005FH	Friesian Hols...	Betina	48	42.0	52.0	305.76



2.9.3. Delete Ternak

ID

B005FH

Lingkar Dada

42.0

Jenis Sapi

Friesian Holstein

Panjang Badan

52.0

Jenis Kelamin

Betina

Berat

305.76

Umur

48

bulan

SIMPAN

UPDATE

HAPUS

ID	Jenis Sapi	Jenis Kelamin	Umur	Lingkar Dada	Panjang Badan	Berat
B001FH	Friesian Hols...	Betina	48	35.6	50.8	128.13
B002FH	Friesian Hols...	Jantan	6	24.0	36.0	69.12
B003FH	Friesian Hols...	Jantan	30	40.58	51.65	298.57
B004FH	Friesian Hols...	Betina	40	42.0	52.0	305.76
B005FH	Friesian Hols...	Betina	48	42.0	52.0	305.76

Info

i

Data sapi berhasil dihapus

OK

ID

Jenis Sapi

Jenis Kelamin

Umur

Lingkar Dada

Panjang Badan

Berat

bulan

SIMPAN

UPDATE

HAPUS

ID	Jenis Sapi	Jenis Kelamin	Umur	Lingkar Dada	Panjang Badan	Berat
B001FH	Friesian Hols...	Betina	48	35.6	50.8	128.13
B002FH	Friesian Hols...	Jantan	6	24.0	36.0	69.12
B003FH	Friesian Hols...	Jantan	30	40.58	51.65	298.57
B004FH	Friesian Hols...	Betina	40	42.0	52.0	305.76

BAB III

PENUTUP

3.1. Kesimpulan

Aplikasi manajemen peternakan Bibilintik di Kota Bogor dibuat dengan Java Hibernate, Netbeans dan MySQL untuk menyimpan database. Aplikasi ini dapat menyimpan dan mengolah data ternak, membuat laporan data ternak.

3.2. Saran

Dalam aplikasi ini masih dibutuhkan beberapa pengembangan dan modifikasi agar terlihat lebih menarik dan interaktif. Oleh karena itu, penulis menyarankan agar Aplikasi ini dapat lebih dikembangkan lagi, agar dapat bermanfaat bagi para pegawai marketing sehingga mempermudah pekerjaan mereka.