

The background features a large, light gray watermark of the Universitas Muhammadiyah Purwokerto logo. The logo is a circular emblem with a scalloped border. Inside the circle, the text "UNIVERSITAS MUHAMMADIYAH" is written along the top arc, and "PURWOKERTO" is written along the bottom arc. The center of the emblem contains a sunburst design with rays emanating from a central point, flanked by two stylized floral or leaf-like motifs. Two small asterisks are positioned on the left and right sides of the emblem.

LAMPIRAN 1

Lampiran 1.

Data Kesesuaian Lahan Pohon Acacia Mangium

Karakteristik Lahan	Kelas Kesesuaian Lahan				
	S1	S2	S3	N1	N2
Ketinggian (m dpl)	300 – 700				
Kemiringan (Topografi)	Datar (<8)	Landai (8-15)	Curam (>15-30)	Agak Curam (>30-50)	Sangat Curam (>50)
Tekstur Tanah	SL, L, SCL, SiL, Si, CL, SiCL	S, LS, SC, SiC	liat masif, StrC	-	Kerikil
Curah Hujan (mm)/Tahun	1300-2500	>2500-4000 1000<1300	-	-	>4000 <1000
Solum	Dalam (>100)		Sedang(50 - <75)		Dangkal (<50)

Data Kesesuaian Lahan Pohon Damar

Karakteristik Lahan	Kelas Kesesuaian Lahan				
	S1	S2	S3	N1	N2
Ketinggian (m dpl)	300 –1200				
Kemiringan (Topografi)	Datar (<8)	Landai (8 - 15)	Curam (>15 - 30)	Agak Curam (>30 - 50)	Sangat Curam (>50)
Tekstur Tanah	SL, L, SCL, SiL, Si, CL, SiCL	SC, SiC,Strc	Liat masif	Tidak ada	Kerikil, pasir
Curah Hujan (mm)/Tahun	2500-3000	>3000-4000 2000- <2500	Tidak Ada	Tidak Ada	>4000 <2000
Solum	Dalam(≥150)		Sedang (75 - <100)		Dangkal (<50)

Data Kesesuaian Lahan Pohon Gaharu

Karakteristik Lahan	Kelas Kesesuaian Lahan				
	S1	S2	S3	N1	N2
Ketinggian (m dpl)	0-750				
Kemiringan (Topografi)	Landai(<15)	Curam (16-30)	Agak Curam (31-45)	Datar (<8)	Sangat Curam (>65)
Tekstur Tanah	Liat Berpasir,liat berdebu,liat	Lempung berpasir	Lempungliat berpasir		berpasir
Curah Hujan (mm)/Tahun	2000-3000	1000-2000 3000-4000	500-1000 4000-5000	Tidak Ada	>5000
Solum	Sedang(>75)		Dangkal (25-50)	Tidak ada	Dangkal (<25)

Data Kesesuaian Lahan Pohon Jati

Karakteristik Lahan	Kelas Kesesuaian Lahan				
	S1	S2	S3	N1	N2
Ketinggian (m dpl)	0-700				
Kemiringan (Topografi)	Datar (<8)	Landai (8-15)	Curam (>15-30)	Agak Curam (>30-50)	Sangat Curam (>50)
Tekstur Tanah	L, CL, SiCL, SiL, SC, SiC	SL, StrC	LS.Liat Masif	Tidak ada	Kerikil, pasir
Curah Hujan (mm)/Tahun	1500-<2000	2000-<2250	2250-<2500	Tidak Ada	>2500<1000
Solum	Dalam (=150)		Sedang (75<100)		Dangkal (50)

Data Kesesuaian Lahan Pohon Mahoni

Karakteristik Lahan	Kelas Kesesuaian Lahan				
	S1	S2	S3	N1	N2
Ketinggian (m dpl)	0-1000				
Kemiringan (Topografi)	Datar (<8)	Landai (8-15)	Curam (>15-30)	Agak Curam (>30-50)	Sangat Curam (>50)
Tekstur Tanah	L, SCL, SiL, Si, CL, SiCL	SL, SC,SiC,	LS, StrC, liat masif	Tidak ada	Kerikil,pasir
Curah Hujan (mm)/Tahun	2000-3000	3000-3500 1750<2000	>3500-4000 1500-<1750	Tidak Ada	>4000 <1500
Solum	Dalam (>150)		Sedang (75-<100)		Dangkal (<50)

Data Kesesuaian Lahan Pohon Pinus

Karakteristik Lahan	Kelas Kesesuaian Lahan				
	S1	S2	S3	N1	N2
Ketinggian (m dpl)	200 –1 700				
Kemiringan (Topografi)	Datar (<8)	Landai (8 -15)	Curam (>15 - 30)	Agak Curam (>30 - 50)	Sangat Curam (>50)
Tekstur Tanah	SL,L,SCL, SiL,Si,CL, SiCL	LS, SC, SiC	-	-	Kerikil,pasir
Curah Hujan (mm)/Tahun	2500-3000	>3000-4000 2000-<2500			>4000 <2000
Solum	Dalam (>100)		Sedang (50-<70)		Dangkal (>50)

Data Kesesuaian Lahan Pohon Rasamala

Karakteristik Lahan	Kelas Kesesuaian Lahan				
	S1	S2	S3	N1	N2
Ketinggian (m dpl)	500-1.500				
Kemiringan (Topografi)	Datar (<8)	Landai (8-15)	Curam (>15-30)	Agak Curam (>30-50)	Sangat Curam (>50)
Tekstur Tanah	L.,SCL, SiL,Si,CL,SiCL	SL,SC,SiC,Str C	LS, Liat masif	Tidak ada	Kerikil,pasir
Curah Hujan (mm)/Tahun	2000-2500	>2500-3000 1500-<2000	<1500-1000	Tidak Ada	>3000 <1000
Solum	Dalam(>150)		Sedang(75<100)		Dangkal (<50)

Data Kesesuaian Lahan Pohon Sengon

Karakteristik Lahan	Kelas Kesesuaian Lahan				
	S1	S2	S3	N1	N2
Ketinggian (m dpl)	0-1500				
Kemiringan (Topografi)	Datar (<8)	Landai (8-15)	Curam (16-30)	Agak Curam (>30)	Sangat Curam (>50)
Tekstur Tanah	L, SCL, SiL, Si, CL, SC, SiCL	S, LS, SL, SiC	liat masif, StrC		Kerikil
Curah Hujan (mm)/Tahun	2500-3000	>3000-4000 2000-<2500	Tidak ada	Tidak Ada	>4000 <2000
Solum		Dalam(>100)	Sedang (75-<100)		Dangkal(<50)

The background of the page features a large, light gray watermark of the logo of Universitas Muhammadiyah Purwokerto. The logo is circular with a scalloped border. It contains a central emblem with Arabic calligraphy, surrounded by a wreath of leaves and flowers. The text "UNIVERSITAS MUHAMMADIYAH" is written along the top inner edge, and "PURWOKERTO" is written along the bottom inner edge. Two small asterisks are positioned on the left and right sides of the inner circle.

LAMPIRAN 2

Lampiran 3. Kode Program

- Kode Program Koneksi

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data;
using System.Data.SqlClient;

namespace SPK_WP.Koneksi
{
    class koneksi
    {
        public SqlConnection koneksi_db = new SqlConnection();
        string konfigurasi = "Data Source=DIMAS-PC; Initial Catalog =
spk_db; Integrated Security=true";
        public void buka_koneksi()
        {
            koneksi_db = new SqlConnection(konfigurasi);
            koneksi_db.Open();
        }
        public void tutup_koneksi()
        {
            koneksi_db = new SqlConnection(konfigurasi);
            koneksi_db.Close();
        }
    }
}

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace SPK_WP
{
    public partial class home : Form
    {
        private profil_kph _profil_kph;
        private bagian_hutan _bagian_hutan;
        private pohon _pohon;
        private bobot_kriteria _bobot_kriteria;
        private penilaian _penilaian;
        private tambah_user _tambah_user;
        private perhitungan _perhitungan;
        public home()
        {
            InitializeComponent();
        }
        private void TutupMDIform()
    }
}
```

```

    {
        foreach (Form f in this.MdiChildren)
        {
            f.Close();
        }
    }

    private void jgkToolStripMenuItem_Click(object sender, EventArgs e)
    {
        foreach (Form f in this.MdiChildren)
        {
            if (f is bagian_hutan)
            {
                return;
            }
        }

        TutupMDIform();
        _bagian_hutan = new bagian_hutan();
        _bagian_hutan.MdiParent = this;
        _bagian_hutan.Show();
    }

    private void jenisJenisPohonToolStripMenuItem_Click(object sender,
    EventArgs e)
    {
        foreach (Form f in this.MdiChildren)
        {
            if (f is pohon)
            {
                return;
            }
        }

        TutupMDIform();
        _pohon = new pohon();
        _pohon.MdiParent = this;
        _pohon.Show();
    }

    private void bobotKriteriaToolStripMenuItem_Click(object sender,
    EventArgs e)
    {
        foreach (Form f in this.MdiChildren)
        {
            if (f is bobot_kriteria)
            {
                return;
            }
        }

        TutupMDIform();
        _bobot_kriteria = new bobot_kriteria();
        _bobot_kriteria.MdiParent = this;
        _bobot_kriteria.Show();
    }

    private void kToolStripMenuItem_Click(object sender, EventArgs e)
    {

```



```

        foreach (Form f in this.MdiChildren)
        {
            if (f is penilaian)
            {
                return;
            }
        }
        TutupMDIform();
        _penilaian = new penilaian();
        _penilaian.MdiParent = this;
        _penilaian.Show();
    }
    private void userToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        foreach (Form f in this.MdiChildren)
        {
            if (f is tambah_user)
            {
                return;
            }
        }
        TutupMDIform();
        _tambah_user = new tambah_user();
        _tambah_user.MdiParent = this;
        _tambah_user.Show();
    }
    private void kPHBanyumasTimurToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        foreach (Form f in this.MdiChildren)
        {
            if (f is profil_kph)
            {
                return;
            }
        }
        TutupMDIform();
        _profil_kph = new profil_kph();
        _profil_kph.MdiParent = this;
        _profil_kph.Show();
    }
    private void home_Load(object sender, EventArgs e)
    {
        timer1.Interval = 1000;
        timer1.Enabled = true;
        label6.Text = DateTime.Now.ToLongTimeString();
        label7.Text = DateTime.Now.ToLongDateString();
    }

    private void pERHITUNGANToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        foreach (Form f in this.MdiChildren)
        {
            if (f is perhitungan)
            {

```

```

        return;
    }
}

TutupMDIform();
_perhitungan = new perhitungan();
_perhitungan.MdiParent = this;
_perhitungan.Show();
}
}
}

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace SPK_WP
{
    public partial class bagian_hutan : Form
    {
        DataTable dt = new DataTable();
        string bagianhutan_id;

        public bagian_hutan()
        {
            InitializeComponent();
        }
        public void tampil_data()
        {
            Koneksi.koneksi k = new Koneksi.koneksi();
            k.buka_koneksi();
            dt = new DataTable();
            SqlDataAdapter sda = new SqlDataAdapter("SELECT ROW_NUMBER
            ()over(order by bagianhutan_id asc) as 'No',
            bagianhutan_id,bagianhutan_nama FROM tabel_bagianhutan",k.koneksi_db);
            sda.Fill(dt);
            gv_bagianhutan.DataSource = dt;
            k.tutup_koneksi();
            gv_bagianhutan.Columns[1].Visible = false;
            gv_bagianhutan.Columns[0].Width = 50;
            gv_bagianhutan.Columns[2].Width = 340;
        }
        private void bagian_hutan_Load(object sender, EventArgs e)
        {
            tampil_data();
            btn_edit.Enabled = false;
            btn_hapus.Enabled = false;
        }
        public void tambah_data()
        {

```

```

        if (tx_namahutan.Text == "")
        {
            MessageBox.Show("Nama Hutan Tidak boleh kosong",
"Peringatan");
        }
        else
        {
            string bagianhutan_nama = tx_namahutan.Text;
            Koneksi.koneksi k = new Koneksi.koneksi();
            k.buka_koneksi();
            SqlCommand com = new SqlCommand();
            com.Connection = k.koneksi_db;
            com.CommandType = CommandType.Text;
            com.CommandText = "INSERT INTO tabel_bagianhutan
(bagianhutan_nama) values (@bagian_hutan_nama)";
            com.Parameters.Add("@bagian_hutan_nama",
SqlDbType.VarChar).Value = bagianhutan_nama;
            com.ExecuteNonQuery();
            k.tutup_koneksi();
            MessageBox.Show("Data Bagian Hutan Berhasil di Tambah",
"informasi");
            refresh();
        }
    }
    public void edit_data()
    {
        if (tx_namahutan.Text == "")
        {
            MessageBox.Show("Nama Hutan Tidak boleh kosong",
"Peringatan");
        }
        else
        {
            string bagianhutan_nama = tx_namahutan.Text;
            Koneksi.koneksi k = new Koneksi.koneksi();
            k.buka_koneksi();
            SqlCommand com = new SqlCommand();
            com.Connection = k.koneksi_db;
            com.CommandType = CommandType.Text;
            com.CommandText = "UPDATE tabel_bagianhutan SET
bagianhutan_nama='" + bagianhutan_nama + "' WHERE bagianhutan_id='" +
bagianhutan_id + "'";
            com.ExecuteNonQuery();
            k.tutup_koneksi();
            MessageBox.Show("Data Berhasil di Edit", "Informasi");
            refresh();
        }
    }
    public void hapus_data()
    {
        if (tx_namahutan.Text == "")
        {
            MessageBox.Show("Nama Hutan Tidak boleh kosong",
"Peringatan");
        }
        else
        {

```

```

        string bagianhutan_nama = tx_namahutan.Text;
        Koneksi.koneksi k = new Koneksi.koneksi();
        k.buka_koneksi();
        SqlCommand com = new SqlCommand();
        com.Connection = k.koneksi_db;
        com.CommandType = CommandType.Text;
        com.CommandText = "DELETE FROM tabel_bagianhutan WHERE
bagianhutan_id='" + bagianhutan_id + "'";
        com.ExecuteNonQuery();
        k.tutup_koneksi();
        MessageBox.Show("Data Berhasil di Hapus", "Informasi");

        refresh();
    }
}

public void tabel_klik()
{
    if (gv_bagianhutan.Rows.Count > 0)
    {
        tx_namahutan.Text =
gv_bagianhutan.SelectedRows[0].Cells[2].Value.ToString();
        bagianhutan_id =
gv_bagianhutan.SelectedRows[0].Cells[1].Value.ToString();
        btn_edit.Enabled = true;
        btn_hapus.Enabled = true;
        btn_tambah.Enabled = false;
    }
    else
    {
        gv_bagianhutan.Refresh();
    }
}

public void refresh()
{
    btn_edit.Enabled = false;
    btn_hapus.Enabled = false;
    btn_tambah.Enabled = true;
    tx_namahutan.Text = "";
}

private bool validasi()
{
    Boolean cek = true;

    if (String.IsNullOrEmpty(tx_namahutan.Text))
    {
        tx_namahutan.Focus();
        errorProvider1.SetError(tx_namahutan, "Isi Nama Bagian
Hutan dulu");
        cek = false;
    }
    return cek;
}

private void btn_tambah_Click(object sender, EventArgs e)
{
    tambah_data();
}

```

```

        tampil_data();
    }
    private void gv_bagianhutan_MouseClick(object sender,
MouseEventArgs e)
    {
        tabel_klik();
    }
    private void bt_edit_Click(object sender, EventArgs e)
    {
        edit_data();
        tampil_data();
    }
    private void btn_hapus_Click(object sender, EventArgs e)
    {
        hapus_data();
        tampil_data();
    }
    private void btn_cancel_Click(object sender, EventArgs e)
    {
        btn_edit.Enabled = false;
        btn_hapus.Enabled = false;
        btn_tambah.Enabled = true;
        tx_namahutan.Text = "";
    }

    private void tx_namahutan_KeyPress(object sender, KeyPressEventArgs
e)
    {
        string karakter =
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ \b";
        if (karakter.IndexOf(e.KeyChar) < 0)
        {
            tx_namahutan.Focus();
            errorProvider1.SetError(tx_namahutan, "Hanya Bisa Dimasuki
Karakter Huruf");
            e.Handled = true;
        }
        else
        {
            errorProvider1.Clear();
            e.Handled = false;
        }
    }
}

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace SPK_WP

```

```

{
    public partial class bobot_kriteria : Form
    {
        public bobot_kriteria()
        {
            InitializeComponent();
        }

        public void edit()
        {
            float ketinggian = float.Parse(tx_ketinggian.Text);
            float teksturtanah = float.Parse(tx_teksturtanah.Text);
            float solum = float.Parse(tx_solum.Text);
            float curahhujan = float.Parse(tx_curahhujan.Text);
            float topografi = float.Parse(tx_topografi.Text);
            float jumlah = topografi + teksturtanah + solum + curahhujan +
ketinggian;

            if (tx_topografi.Text == "")
            {
                MessageBox.Show("Nilai Topografi Tidak boleh kosong",
"Peringatan");
            }
            else if(tx_teksturtanah.Text == "")
            {
                MessageBox.Show("Nilai Tekstur Tanah Tidak Boleh
Kosong", "Peringatan");
            }
            else if (tx_curahhujan.Text == "")
            {
                MessageBox.Show("Nilai Curah Hujan Tidak Boleh
Kosong", "Peringatan");
            }
            else if(tx_solum.Text== "")
            {
                MessageBox.Show("Nilai Solum Tidak Boleh
Kosong", "Peringatan");
            }
            else if(tx_ketinggian.Text == "")
            {
                MessageBox.Show("Nilai Ketinggian Tidak Boleh
Kosong", "Peringatan");
            }
            else if (topografi >= 100)
            {
                MessageBox.Show("Nilai Topografi tidak boleh lebih dari 100
%", "Peringatan");
            }
            else if (teksturtanah >= 100)
            {
                MessageBox.Show("Nilai Tekstur Tanah Tidak Boleh Lebih dari
100 %", "Peringatan");
            }
            else if (solum >= 100)
            {

```



```

        MessageBox.Show("Nilai solum Tidak Boleh Lebih dari 100 %",
"Peringatan");
    }
    else if (curahhujan >= 100)
    {
        MessageBox.Show("Nilai Curah Hujan Tidak Boleh Lebih dari
100 %", "Peringatan");
    }
    else if (ketinggian >= 100)
    {
        MessageBox.Show("Nilai Ketinggian Tidak Boleh Lebih dari
100 %", "Peringatan");
    }
    else if (jumlah > 100)
    {
        MessageBox.Show("Jumlah Nilai Lebih Dari 100 %, Semua Bobot
Harus 100 %", "Peringatan");
    }
    else if (jumlah < 100)
    {
        MessageBox.Show("Jumlah Nilai Kurang Dari 100 %. Semua
Bobot Harus 100 %", "Peringatan");
    }
    else
    {
        Koneksi.koneksi k = new Koneksi.koneksi();
        k.buka_koneksi();
        SqlCommand com = new SqlCommand();
        com.Connection = k.koneksi_db;
        com.CommandType = CommandType.Text;
        com.CommandText = "update tabel_bobot set
bobot_topografi='" + topografi + "', bobot_tekstur tanah='" + tekstur tanah +
"', bobot_curahhujan='" + curahhujan + "', bobot_solum='" + solum + "',
bobot_ketinggian=' ' + ketinggian + "' where bobot_id=1";
        com.ExecuteNonQuery();
        k.tutup_koneksi();
        MessageBox.Show("Bobot Berhasil Di Ubah", "informasi");
    }
}
private void btn_bobot_Click(object sender, EventArgs e)
{
    edit();
    //tampil_data();
}
public void tampil_data()
{
    Koneksi.koneksi k = new Koneksi.koneksi();
    k.buka_koneksi();
    SqlCommand com = new SqlCommand();
    com.Connection = k.koneksi_db;
    com.CommandType = CommandType.Text;
    com.CommandText = "select bobot_topografi, bobot_tekstur tanah,
bobot_curahhujan, bobot_solum, bobot_ketinggian from tabel_bobot";
    SqlDataReader reader = null;
    reader = com.ExecuteReader();
}

```

```

        if (reader.Read())
        {
            tx_topografi.Text = reader["bobot_topografi"].ToString();
            tx_teksturtanah.Text =
reader["bobot_teksturtanah"].ToString();
            tx_solum.Text = reader["bobot_solum"].ToString();
            tx_curahhujan.Text = reader["bobot_curahhujan"].ToString();
            tx_ketinggian.Text = reader["bobot_ketinggian"].ToString();
        }

        k.tutup_koneksi();
    }

    private void bobot_kriteria_Load(object sender, EventArgs e)
    {
        tampil_data();
    }

    private void tx_topografi_KeyPress(object sender, KeyPressEventArgs
e)
    {
        string karakter = "1234567890\b";
        if (karakter.IndexOf(e.KeyChar) < 0)
        {
            tx_topografi.Focus();
            errorProvider1.SetError(tx_topografi, "Hanya Bisa Dimasuki
Angka");
            e.Handled = true;
        }
        else
        {
            errorProvider1.Clear();
            e.Handled = false;
        }
    }

    private void tx_teksturtanah_KeyPress(object sender,
KeyPressEventArgs e)
    {
        string karakter = "1234567890\b";
        if (karakter.IndexOf(e.KeyChar) < 0)
        {
            tx_teksturtanah.Focus();
            errorProvider1.SetError(tx_teksturtanah, "Hanya Bisa
Dimasuki Angka");
            e.Handled = true;
        }
        else
        {
            errorProvider1.Clear();
            e.Handled = false;
        }
    }

    private void tx_curahhujan_KeyPress(object sender,
KeyPressEventArgs e)

```



```

    {
        string karakter = "1234567890\b";
        if (karakter.IndexOf(e.KeyChar) < 0)
        {
            tx_curahhujan.Focus();
            errorProvider1.SetError(tx_curahhujan, "Hanya Bisa Dimasuki
Angka");
            e.Handled = true;
        }
        else
        {
            errorProvider1.Clear();
            e.Handled = false;
        }
    }
    private void tx_solum_KeyPress(object sender, KeyPressEventArgs e)
    {
        string karakter = "1234567890\b";
        if (karakter.IndexOf(e.KeyChar) < 0)
        {
            tx_solum.Focus();
            errorProvider1.SetError(tx_solum, "Hanya Bisa Dimasuki
Angka");
            e.Handled = true;
        }
        else
        {
            errorProvider1.Clear();
            e.Handled = false;
        }
    }
    private void tx_ketinggian_KeyPress(object sender,
KeyPressEventArgs e)
    {
        string karakter = "1234567890\b";
        if (karakter.IndexOf(e.KeyChar) < 0)
        {
            tx_ketinggian.Focus();
            errorProvider1.SetError(tx_ketinggian, "Hanya Bisa Dimasuki
Angka ");
            e.Handled = true;
        }
        else
        {
            errorProvider1.Clear();
            e.Handled = false;
        }
    }
}
}

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;

```

```

using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace SPK_WP
{
    public partial class penilaian : Form
    {
        DataTable dt = new DataTable();
        string penilaian_id;

        public penilaian()
        {
            InitializeComponent();
        }
        public void tambah_data()
        {
            int bagianhutan_id =
Convert.ToInt32(cmb_bagianhutan.SelectedValue);
            string jenis_pohon = cmb_jenistanaman.Text;
            string topografi = cmb_topografi.Text;
            string tekstur_tanah = cmb_teksturtanah.Text;
            string solum = cmb_solum.Text;
            double curahhujan = Convert.ToDouble(tx_curahhujan.Text);
            double ketinggian = Convert.ToDouble(tx_ketinggian.Text);

            if (jenis_pohon == "Acacia Mangium")
            {
                if (ketinggian >= 300 && ketinggian <= 700)
                {
                    txt_ketinggian.Text = "2";
                }
                if (ketinggian < 300 || ketinggian > 700)
                {
                    txt_ketinggian.Text = "1";
                }
                if (topografi == "DATAR")
                {
                    txt_topografi.Text = "5";
                }
                if (topografi == "LANDAI")
                {
                    txt_topografi.Text = "4";
                }
                if (topografi == "CURAM")
                {
                    txt_topografi.Text = "3";
                }
                if (topografi == "AGAK CURAM")
                {
                    txt_topografi.Text = "2";
                }
                if (topografi == "SANGAT CURAM")
                {
                    txt_topografi.Text = "1";
                }
            }
        }
    }
}

```

```

        if (tekstur_tanah == "LEMPUNG BERPASIR(SL)" ||
tekstur_tanah == "LEMPUNG(L)" || tekstur_tanah == "LEMPUNG LIAT
BERPASIR(SCL)" || tekstur_tanah == "LEMPUNG BERDEBU(SiL)" || tekstur_tanah
== "DEBU(Si)" || tekstur_tanah == "LIAT(CL)" || tekstur_tanah == "LIAT
BERPASIR(SiCl)")
        {
            txt_tekstur.Text = "6";
        }
        if (tekstur_tanah == "PASIR(S)" || tekstur_tanah == "PASIR
BERLUMPUR(LS)" || tekstur_tanah == "LEMPUNG PASIRAN(SC)" || tekstur_tanah
== "LEMPUNG DEBUAN(SiC)")
        {
            txt_tekstur.Text = "5";
        }
        if (tekstur_tanah == "LIAT BERSTRUKTUR(Strc)" ||
tekstur_tanah == "LIAT MASIF")
        {
            txt_tekstur.Text = "4";
        }
        if (tekstur_tanah == "KERIKIL")
        {
            txt_tekstur.Text = "2";
        }
        if (tekstur_tanah == "LIAT BERDEBU(SiCl)" || tekstur_tanah
== "LEMPUNG LIAT BERDEBU(SiClL)" || tekstur_tanah == "LEMPUNG
BERLIAT(CiL)")
        {
            txt_tekstur.Text = "1";
        }

        if (solum == "DALAM")
        {
            txt_solum.Text = "5";
        }
        if (solum == "SEDANG")
        {
            txt_solum.Text = "3";
        }
        if (solum == "DANGKAL")
        {
            txt_solum.Text = "1";
        }
        if (curahhujan > 1300 && curahhujan <= 2500)
        {
            txt_curah_hujan.Text = "6";
        }
        if (curahhujan > 2500 && curahhujan <= 4000)
        {
            txt_curah_hujan.Text = "5";
        }
        if (curahhujan >= 1000 && curahhujan <= 1300)
        {
            txt_curah_hujan.Text = "5";
        }
        if (curahhujan > 4000 || curahhujan < 1000)
        {
            txt_curah_hujan.Text = "2";
        }

```

```

    }
}
if (jenis_pohon == "Damar")
{
    if (ketinggian >= 300 && ketinggian <= 1200)
    {
        txt_ketinggian.Text = "2";
    }
    if (ketinggian < 300 || ketinggian > 1200)
    {
        txt_ketinggian.Text = "1";
    }
    if (topografi == "DATAR")
    {
        txt_topografi.Text = "5";
    }
    if (topografi == "LANDAI")
    {
        txt_topografi.Text = "4";
    }
    if (topografi == "CURAM")
    {
        txt_topografi.Text = "3";
    }
    if (topografi == "AGAK CURAM")
    {
        txt_topografi.Text = "2";
    }
    if (topografi == "SANGAT CURAM")
    {
        txt_topografi.Text = "1";
    }
    if (tekstur_tanah == "LEMPUNG BERPASIR(SL)" ||
tekstur_tanah == "LEMPUNG(L)" || tekstur_tanah == "LEMPUNG LIAT
BERPASIR(SCL)" || tekstur_tanah == "LEMPUNG BERDEBU(SiL)" || tekstur_tanah
== "DEBU(Si)" || tekstur_tanah == "LIAT(CL)" || tekstur_tanah == "LIAT
BERPASIR(SiCl)" )
    {
        txt_tekstur.Text = "6";
    }
    if (tekstur_tanah == "LEMPUNG DEBUAN(SiC)" || tekstur_tanah
== "LIAT BERSTRUKTUR(Strc)" || tekstur_tanah == "LEMPUNG PASIRAN(SC)")
    {
        txt_tekstur.Text = "5";
    }
    if (tekstur_tanah == "LIAT MASIF")
    {
        txt_tekstur.Text = "4";
    }
    if (tekstur_tanah == "KERIKIL")
    {
        txt_tekstur.Text = "2";
    }
    if (tekstur_tanah == "LIAT BERDEBU(SiCl)" || tekstur_tanah
== "LEMPUNG LIAT BERDEBU(SiCl)" || tekstur_tanah == "LEMPUNG BERLIAT(CiL)"
|| tekstur_tanah == "PASIR BERLUMPUR(LS)" || tekstur_tanah == "PASIR(S)")

```

```

        {
            txt_tekstur.Text = "1";
        }

        if (solum == "DALAM")
        {
            txt_solum.Text = "5";
        }
        if (solum == "SEDANG")
        {
            txt_solum.Text = "3";
        }
        if (solum == "DANGKAL")
        {
            txt_solum.Text = "1";
        }
        if (curahhujan > 2500 && curahhujan <= 3000)
        {
            txt_curah_hujan.Text = "6";
        }
        if (curahhujan > 3000 && curahhujan <= 4000)
        {
            txt_curah_hujan.Text = "5";
        }
        if (curahhujan >= 2000 && curahhujan < 2500)
        {
            txt_curah_hujan.Text = "3";
        }
        if (curahhujan > 4000 || curahhujan < 2000)
        {
            txt_curah_hujan.Text = "2";
        }
    }
    Koneksi.koneksi k = new Koneksi.koneksi();
    k.buka_koneksi();
    SqlCommand com = new SqlCommand();
    com.Connection = k.koneksi_db;
    com.CommandType = CommandType.Text;
    com.Parameters.Add("@bagian_hutan_id", SqlDbType.Int).Value =
bagianhutan_id;
    com.Parameters.Add("@pohon_nama", SqlDbType.VarChar).Value =
jenis_pohon;
    com.Parameters.Add("@topografi", SqlDbType.VarChar).Value =
topografi;
    com.Parameters.Add("@nilai_topografi", SqlDbType.Float).Value =
txt_topografi.Text;
    com.Parameters.Add("@teksturtanah", SqlDbType.VarChar).Value =
tekstur_tanah;
    com.Parameters.Add("@nilai_teksturtanah",
SqlDbType.Float).Value = txt_tekstur.Text;
    com.Parameters.Add("@curahhujan", SqlDbType.VarChar).Value =
curahhujan;
    com.Parameters.Add("@nilai_curahhujan", SqlDbType.Float).Value
= txt_curah_hujan.Text;
    com.Parameters.Add("@solum", SqlDbType.VarChar).Value = solum;

```

```

        com.Parameters.Add("@nilai_solum", SqlDbType.Float).Value =
txt_solum.Text;
        com.Parameters.Add("@ketinggian", SqlDbType.VarChar).Value =
ketinggian;
        com.Parameters.Add("@nilai_ketinggian", SqlDbType.Float).Value
= txt_ketinggian.Text;
        com.CommandText = "INSERT INTO tabel_penilaian (bagianhutan_id,
pohon_nama, topografi, nilai_topografi, teksturtanah, nilai_teksturtanah,
curahhujan, nilai_curahhujan, solum, nilai_solum, ketinggian,
nilai_ketinggian) values (@bagian_hutan_id, @pohon_nama,
@topografi,@nilai_topografi, @teksturtanah, @nilai_teksturtanah,
@curahhujan, @nilai_curahhujan, @solum,
@nilai_solum,@ketinggian,@nilai_ketinggian)";
        com.ExecuteNonQuery();
        MessageBox.Show("Data Penilaian Berhasil Di Simpan");
        k.tutup_koneksi();
    }

    public void tampil_bagian_hutan()
    {
        Koneksi.koneksi k = new Koneksi.koneksi();
        k.buka_koneksi();
        dt = new DataTable();
        SqlCommand com = new SqlCommand();
        com.Connection = k.koneksi_db;
        com.CommandType = CommandType.Text;
        com.CommandText = "SELECT bagianhutan_id, bagianhutan_nama FROM
tabel_bagianhutan";
        SqlDataReader reader;
        reader = com.ExecuteReader();
        dt.Columns.Add("bagianhutan_id", typeof(string));
        dt.Columns.Add("bagianhutan_nama", typeof(string));
        dt.Load(reader);
        cmb_bagianhutan.ValueMember = "bagianhutan_id";
        cmb_bagianhutan.DisplayMember = "bagianhutan_nama";
        cmb_bagianhutan.DataSource = dt;
        k.tutup_koneksi();
    }

    public void tabel_klik()
    {
        if (gv_penilaian.Rows.Count > 0)
        {
            cmb_bagianhutan.Text =
gv_penilaian.SelectedRows[0].Cells[1].Value.ToString();
            penilaian_id =
gv_penilaian.SelectedRows[0].Cells[0].Value.ToString();
            cmb_jenistanaman.SelectedItem =
gv_penilaian.SelectedRows[0].Cells[2].Value.ToString();
            cmb_topografi.SelectedItem =
gv_penilaian.SelectedRows[0].Cells[3].Value.ToString();
            cmb_teksturtanah.SelectedItem =
gv_penilaian.SelectedRows[0].Cells[4].Value.ToString();
            tx_curahhujan.Text =
gv_penilaian.SelectedRows[0].Cells[5].Value.ToString();
            cmb_solum.SelectedItem =
gv_penilaian.SelectedRows[0].Cells[6].Value.ToString();

```



```

        tx_ketinggian.Text =
gv_penilaian.SelectedRows[0].Cells[7].Value.ToString();
        txt_topografi.Text =
gv_penilaian.SelectedRows[0].Cells[8].Value.ToString();
        txt_tekstur.Text =
gv_penilaian.SelectedRows[0].Cells[9].Value.ToString();
        txt_curah_hujan.Text =
gv_penilaian.SelectedRows[0].Cells[10].Value.ToString();
        txt_solum.Text =
gv_penilaian.SelectedRows[0].Cells[11].Value.ToString();
        txt_ketinggian.Text =
gv_penilaian.SelectedRows[0].Cells[12].Value.ToString();
        //btn_tambah.Enabled = false;
    }
    else
    {
        gv_penilaian.Refresh();
    }
}
private void penilaian_Load(object sender, EventArgs e)
{
    //btn_edit.Visible = false;
    tampil_bagian_hutan();
    tampil_data();
    txt_topografi.Visible = false;
    txt_tekstur.Visible = false;
    txt_curah_hujan.Visible = false;
    txt_solum.Visible = false;
    txt_ketinggian.Visible = false;
}
public void tampil_data()
{
    Koneksi.koneksi k = new Koneksi.koneksi();
    k.buka_koneksi();
    dt = new DataTable();
    SqlDataAdapter sda = new SqlDataAdapter("SELECT * from
vw_penilaian", k.koneksi_db);
    sda.Fill(dt);
    gv_penilaian.DataSource = dt;
    k.tutup_koneksi();
    gv_penilaian.Columns[1].Width = 170;
    gv_penilaian.Columns[0].Visible = false;
    gv_penilaian.Columns[2].Width = 170;
    gv_penilaian.Columns[3].Width = 150;
    gv_penilaian.Columns[4].Width = 150;
    gv_penilaian.Columns[5].Width = 150;
    gv_penilaian.Columns[6].Width = 150;
    gv_penilaian.Columns[7].Width = 150;
    gv_penilaian.Columns[8].Visible = false;
    gv_penilaian.Columns[9].Visible = false;
    gv_penilaian.Columns[10].Visible = false;
    gv_penilaian.Columns[11].Visible = false;
    gv_penilaian.Columns[12].Visible = false;
}
private void btn_tambah_Click(object sender, EventArgs e)
{
    tambah_data();
}

```

```

    tampil_data();
}

public void clear()
{
    txt_topografi.Text = "";
    txt_tekstur.Text = "";
    txt_solum.Text = "";
    txt_ketinggian.Text = "";
    txt_curah_hujan.Text = "";
    cmb_bagianhutan.SelectedValue = 1;
    cmb_jenistanaman.SelectedIndex = 0;
    cmb_topografi.SelectedIndex = 0;
    cmb_teksturtanah.SelectedIndex = 0;
    tx_curahhujan.Text = "";
    cmb_solum.SelectedIndex = 0;
    tx_ketinggian.Text = "";
}

public void hapus_data()
{
    int bagianhutan_id =
Convert.ToInt32(cmb_bagianhutan.SelectedValue);
    string jenis_pohon = cmb_jenistanaman.Text;
    string topografi = cmb_topografi.Text;
    string tekstur_tanah = cmb_teksturtanah.Text;
    string solum = cmb_solum.Text;
    double curahhujan = Convert.ToDouble(tx_curahhujan.Text);
    double ketinggian = Convert.ToDouble(tx_ketinggian.Text);

    Koneksi.koneksi k = new Koneksi.koneksi();
    k.buka_koneksi();
    SqlCommand com = new SqlCommand();
    com.Connection = k.koneksi_db;
    com.CommandType = CommandType.Text;
    com.Parameters.Add("@bagian_hutan_id", SqlDbType.Int).Value =
bagianhutan_id;
    com.Parameters.Add("@pohon_nama", SqlDbType.VarChar).Value =
jenis_pohon;
    com.Parameters.Add("@topografi", SqlDbType.VarChar).Value =
topografi;
    com.Parameters.Add("@nilai_topografi", SqlDbType.Float).Value =
txt_topografi.Text;
    com.Parameters.Add("@teksturtanah", SqlDbType.VarChar).Value =
tekstur_tanah;
    com.Parameters.Add("@nilai_teksturtanah",
SqlDbType.Float).Value = txt_tekstur.Text;
    com.Parameters.Add("@curahhujan", SqlDbType.VarChar).Value =
curahhujan;
    com.Parameters.Add("@nilai_curahhujan", SqlDbType.Float).Value =
txt_curah_hujan.Text;
    com.Parameters.Add("@solum", SqlDbType.VarChar).Value = solum;
    com.Parameters.Add("@nilai_solum", SqlDbType.Float).Value =
txt_solum.Text;
    com.Parameters.Add("@ketinggian", SqlDbType.VarChar).Value =
ketinggian;
    com.Parameters.Add("@nilai_ketinggian", SqlDbType.Float).Value =
txt_ketinggian.Text;
}

```



```

        com.CommandText = "DELETE FROM tabel_penilaian WHERE
penilaian_id='" + penilaian_id + "'";
        com.ExecuteNonQuery();
        MessageBox.Show("Data Penilaian Berhasil Di Hapus");
        k.tutup_koneksi();

    }
    public void edit_data()
    {
        int bagianhutan_id =
Convert.ToInt32(cmb_bagianhutan.SelectedValue);
        string jenis_pohon = cmb_jenistanaman.Text;
        string topografi = cmb_topografi.Text;
        string tekstur_tanah = cmb_teksturtanah.Text;
        string solum = cmb_solum.Text;
        double curahhujan = Convert.ToDouble(tx_curahhujan.Text);
        double ketinggian = Convert.ToDouble(tx_ketinggian.Text);

        Koneksi.koneksi k = new Koneksi.koneksi();
        k.buka_koneksi();
        SqlCommand com = new SqlCommand();
        com.Connection = k.koneksi_db;
        com.CommandType = CommandType.Text;
        com.Parameters.Add("@penilaian_id", SqlDbType.Int).Value =
penilaian_id;
        com.Parameters.Add("@bagianhutan_id", SqlDbType.Int).Value =
bagianhutan_id;
        com.Parameters.Add("@pohon_nama", SqlDbType.VarChar).Value =
jenis_pohon;
        com.Parameters.Add("@topografi", SqlDbType.VarChar).Value =
topografi;
        com.Parameters.Add("@nilai_topografi", SqlDbType.Float).Value =
txt_topografi.Text;
        com.Parameters.Add("@teksturtanah", SqlDbType.VarChar).Value =
tekstur_tanah;
        com.Parameters.Add("@nilai_teksturtanah",
SqlDbType.Float).Value = txt_tekstur.Text;
        com.Parameters.Add("@curahhujan", SqlDbType.VarChar).Value =
curahhujan;
        com.Parameters.Add("@nilai_curahhujan", SqlDbType.Float).Value
= txt_curah_hujan.Text;
        com.Parameters.Add("@solum", SqlDbType.VarChar).Value = solum;
        com.Parameters.Add("@nilai_solum", SqlDbType.Float).Value =
txt_solum.Text;
        com.Parameters.Add("@ketinggian", SqlDbType.VarChar).Value =
ketinggian;
        com.Parameters.Add("@nilai_ketinggian", SqlDbType.Float).Value
= txt_ketinggian.Text;
        com.CommandText = "UPDATE tabel_penilaian SET
bagianhutan_id=@bagianhutan_id,pohon_nama=@pohon_nama,topografi=@topografi,
nilai_topografi=@nilai_topografi, teksturtanah=@teksturtanah,
nilai_teksturtanah=@nilai_teksturtanah, curahhujan=@curahhujan,
nilai_curahhujan=@nilai_curahhujan, solum=@solum, nilai_solum=@nilai_solum,
ketinggian=@ketinggian,nilai_ketinggian=@nilai_ketinggian WHERE
penilaian_id=@penilaian_id";
        com.ExecuteNonQuery();
        MessageBox.Show("Data Penilaian Berhasil Di Edit");
    }
}

```

```

        k.tutup_koneksi();
    }

    private void btn_clear_Click(object sender, EventArgs e)
    {
        clear();
    }

    private void gv_penilaian_MouseClick(object sender, MouseEventArgs
e)
    {
        tabel_klik();
    }

    private void btn_hapus_Click(object sender, EventArgs e)
    {
        hapus_data();
        tampil_data();
    }

    private void btn_edit_Click(object sender, EventArgs e)
    {
        edit_data();
        tampil_data();
    }

    private void cmb_bagianhutan_SelectedIndexChanged(object sender,
EventArgs e)
    {
    }
}

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace SPK_WP
{
    public partial class perhitungan : Form
    {
        DataTable dt = new DataTable();
        string bagianhutan_nama;
        public perhitungan()
        {
            InitializeComponent();
        }
        public void tampil_bagian_hutan()
        {

```

```

Koneksi.koneksi k = new Koneksi.koneksi();
k.buka_koneksi();
dt = new DataTable();
SqlCommand com = new SqlCommand();
com.Connection = k.koneksi_db;
com.CommandType = CommandType.Text;
com.CommandText = "SELECT bagianhutan_id, bagianhutan_nama FROM
tabel_bagianhutan";
SqlDataReader reader;
reader = com.ExecuteReader();
dt.Columns.Add("bagianhutan_id", typeof(string));
dt.Columns.Add("bagianhutan_nama", typeof(string));
dt.Load(reader);
cmb_bagianhutan.ValueMember = "bagianhutan_id";
cmb_bagianhutan.DisplayMember = "bagianhutan_nama";
cmb_bagianhutan.DataSource = dt;
k.tutup_koneksi();
}
private void perhitungan_Load(object sender, EventArgs e)
{
    tampil_bagian_hutan();
    tampil_data();
    tampil_data_perbaikanbobot();
    tampil_data_vektor_s();
    tampil_data_vektor_v();
}
public void tampil_data()
{
    Koneksi.koneksi k = new Koneksi.koneksi();
    k.buka_koneksi();
    dt = new DataTable();
    SqlDataAdapter sda = new SqlDataAdapter("SELECT * from
vw_matrix where bagianhutan_nama='"+bagianhutan_nama+'"', k.koneksi_db);
    sda.Fill(dt);
    gv_matrix.DataSource = dt;
    k.tutup_koneksi();
    gv_matrix.Columns[1].Width = 150;
    gv_matrix.Columns[0].Visible = false;
    gv_matrix.Columns[2].Width = 150;
    gv_matrix.Columns[3].Width = 110;
    gv_matrix.Columns[4].Width = 120;
    gv_matrix.Columns[5].Width = 120;
    gv_matrix.Columns[6].Width = 110;
}
public void tampil_data_perbaikanbobot()
{
    Koneksi.koneksi k = new Koneksi.koneksi();
    k.buka_koneksi();
    dt = new DataTable();
    SqlDataAdapter sda = new SqlDataAdapter("SELECT * FROM
vw_perbaikan_bobot", k.koneksi_db);
    sda.Fill(dt);
    gv_perbaikan_bobot.DataSource = dt;
    k.tutup_koneksi();
    gv_perbaikan_bobot.Columns[0].Width = 170;
    gv_perbaikan_bobot.Columns[1].Width = 170;
}

```

```

gv_perbaikan_bobot.Columns[2].Width = 170;
gv_perbaikan_bobot.Columns[3].Width = 170;
gv_perbaikan_bobot.Columns[4].Width = 170;

}
public void tampil_data_vektor_s()
{
    Koneksi.koneksi k = new Koneksi.koneksi();
    k.buka_koneksi();
    dt = new DataTable();
    SqlDataAdapter sda = new SqlDataAdapter("SELECT * FROM
vw_vektor_s where bagianhutan_nama='" + bagianhutan_nama + "'",
k.koneksi_db);
    sda.Fill(dt);
    gv_vektor_s.DataSource = dt;
    k.tutup_koneksi();
    gv_vektor_s.Columns[0].Width = 200;
    gv_vektor_s.Columns[1].Width = 200;
    gv_vektor_s.Columns[2].Width = 250;
}
public void tampil_data_vektor_v()
{
    Koneksi.koneksi k = new Koneksi.koneksi();
    k.buka_koneksi();
    dt = new DataTable();
    SqlDataAdapter sda = new SqlDataAdapter("SELECT * FROM
vw_vektor_v where bagianhutan_nama='" + bagianhutan_nama + "' order by
vektor_v desc", k.koneksi_db);
    sda.Fill(dt);
    gv_vektor_v.DataSource = dt;
    k.tutup_koneksi();
    gv_vektor_v.Columns[0].Width = 200;
    gv_vektor_v.Columns[1].Width = 200;
    gv_vektor_v.Columns[2].Width = 250;
}
private void btn_tampil_Click(object sender, EventArgs e)
{
    tampil_data_bagianhutan();
    tampil_data();
    tampil_data_vektor_s();
    tampil_data_vektor_v();
}
public void tampil_data_bagianhutan()
{
    bagianhutan_nama = cmb_bagianhutan.Text;
    //MessageBox.Show(bagianhutan_nama);
}
}
}
}

```