BAB 6

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan oleh penulis berdasarkan masalah yang ada dan tujuan penelitian yang dilakukan pada salah satu situs *online* yaitu situs x maka dapat ditarik kesimpulan sebagai berikut :

a. Berdasarkan tujuan penelitian, *tools* atau program dapat dibuat dan dijalankan dengan baik untuk mengklasifikasikan komentar pelanggan ke dalam dimensi *E-Service quality.*
b. Berdasarkan hasil olah data *text mining* berupa komentar yang sangat acak dapat diklasifikasikan sesuai dengan dimensi *online service quality (E-Service Quality)*.
c. Berdasarkan kuesioner pra penelitian yang telah dilakukan didapatkan prioritas tingkat kepentingan dimensi *e-servqual.*

6.2 Saran

Saran untuk perusahaan agar lebih meningkatkan pelayanan dari segi dimensi *e-servqual* yang memiliki presentase sentimen negatif lebih besar yaitu dalam penelitian ini adalah dimensi *access*..

DAFTAR PUSTAKA

Armadillo, W. J., & Procaccino, J. D. (2016). Competitive Analysis of Online Reviews Using Exploratory Text Mining. *Tourism and Hospitality Management Vol. 22 No. 2* , 193-210.

Buttle, F. (2004). *Customer Relationship Management (Manajemen Hubungan Pelanggan).* Malang: Banyumedia.

Christo, B., & Terblanche, N. (1997). Measuring retail service quality: a replication study. *South African Journal of Business Management Vol. 28 No. 4* , 123- 128.

Dabholkar, P., D.I.Thorpe, & Rentz, J. (1996). A measure of service quality for retail stores: scale development and validation. *Journal of the Academy of Marketing Science Vol. 24 No. 1* , 3-16.

Gagliano, K. B., & Hathcote, J. (1994). Customer Exoectations and Perceptions of Service Quality in Retail Apparel Specialty Stores. *Journal of Services Marketing Vol. 8 No. 1* , 60 - 69.

Griffin, J. (2003). *Customer Royalty.* Jakarta: Erlangga.

Ha, S., & Stoel, L. (2012). Online apparel retailing: roles of e-shopping quality and experiential e-shopping motives. *Journal of Service Management Vol. 23 No. 2* , 197-215.

Kalia, P., Arora, D. R., & Kumalo, S. (2016). E-service quality, consumer satisfaction and future purchase intentions in e-retail. *E-Service Journal Vol. 10 No. 1* , 24-41.

Kandulapati, S., & Bellamkonda, R. S. (2014). E-service quality: a study of online shoppers in India. *American Journal of Business Vol. 29 No. 2* , 178-188.

Kholil, M., & Hanifah, S. (2014). Analisis Kualitas Pelayanan pada Bagian Klaim Berdasarkan Tingkat Kepuasan Nasabah Di PT.AJC Menggunakan Metode Service Quality. *Jurnal Ilmiah Teknik Industri Vol. 2 No. 2* , 75 - 82.

Kim, S., & Jin, B. (2002). Validating the retail service quality scale for US and Korean customers of discount stores: an exploratory study. *Journal of Services Marketing Vol. 16 No. 3* , 223-237.

Mooney, R. J. (2006). *Machine Learning Text Catergorozation.* CS.

Parasuraman, A. Z. (1994). Alternative scales for measuring service quality: a comparative assessment based on psychometric and diagnostic criteria. *Journal of Retailing Vol. 70 No. 3* , 201-230.

Parasuraman, A., A. Zeithaml, V., & Malhotra, A. (2005). Electronic Service Quality : A Multiple-Item Scale for Assessing Electronic Service Quality. *Journal of Service Research* , (3) : 213-233.

Salomon, L. L., Srwana, I. K., & Delila, N. (2014). Analisis Pengukuran Kualitas Pelayanan Bank X dengan Menggunakan Metode Servqual. *Jurnal Ilmiah Teknik Industri Vol.2 No.1* , 28 - 36.

Shingarwade, A. K., & Mulkalwar, D. P. (2017). Study of Text Content Mining for E-Commerce Web Sites. *International Journal of Advanced Research in Computer Science Vol. 8 No.5* , 1269-1274.

Sugiyono. (2008). *Metode Penelitian Kuantitatif dan Kualitatif dan R & D.* Bandung: Alfabeta.

Wijaya, T. (2001). *Manajemen Kualitas Jasa.* Jakarta: Indeks.

Yuliarmi, N. N., & Riyasa, P. (2007). Analisa Faktor- Faktor yang Mempengaruhi Kepuasan Pelanggan Terhadap Pelayanan PDAM Kota Denpasar. *Jurnal Buletin Studi Ekonomi Vol 12 No 1* , 9-28.

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Data.OleDb;
using System.IO;



namespace Skripsi_Puthut
{
    public partial class Form1 : Form
    {

        public Form1()
        {
            InitializeComponent();
        }
        OleDbConnection con = new
OleDbConnection(@"Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Skripsi
Puthut\text minning\textMinning.accdb");

        int i = 0, listmax=0;
        private void Form1_Load(object sender, EventArgs e)
        {
            LoadDimensi();
            loadKataKuncitoList();
            CountPosDimensi();
            CountNegDimensi();

        }
        private void CountPosDimensi()
        {
            CountReliability();
            CountResponsiveness();
            CountAccess();
            CountFletyxibili();
            CountEasyOfNavigation();
            CountEfficiency();
            CountAssurance();
            CountSecurity();
            CountPriveKnowledge();
            CountSiteAesthetics();
            CountCustomization();
        }
        private void CountNegDimensi()
        {
            NegCountReliability();
            NegCountResponsiveness();
            NegCountAccess();
            NegCountFletyxibili();
            NegCountEasyOfNavigation();
            NegCountEfficiency();
            NegCountAssurance();
            NegCountSecurity();
            NegCountPriveKnowledge();
            NegCountSiteAesthetics();
            NegCountCustomization();
```

```
        }

        List<keyword> list= new List<keyword>();

        private void loadKataKuncitoList()
        {
            string strSql = "Select kata_kunci,id_dimensi,status from
tbl_kata_kunci";
            OleDbDataAdapter adapter = new OleDbDataAdapter(new
OleDbCommand(strSql, con));
            DataSet ds = new DataSet();
            adapter.Fill(ds);
            con.Close();
            i = 0;
            foreach (DataRow dr in ds.Tables[0].Rows )
            {
                keyword key = new keyword();
                key.Katakunci = ds.Tables[0].Rows[i]["kata_kunci"].ToString();
                key.Id_dimensi =  Convert.ToInt32(ds.Tables[0].Rows[i]
["id_dimensi"]);
                key.Status = ds.Tables[0].Rows[i]["status"].ToString();
                i++;
                listmax++;
                list.Add(key);

            }
        }
        private void btnCari_Click(object sender, EventArgs e)
        {
            list.Clear();
            Form1_Load(sender, null);
        }
        #region keyword positif
        private void CountReliability()
        {
            try
            {
                int result = 0;

                //  foreach (var key in list)
                for (int x = 0; x < list.Count(); x++)
                {

                    if (list.ElementAt(x).Id_dimensi == 1 &&
list.ElementAt(x).Status == "pos")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" +
list.ElementAt(x).Katakunci + "%' ", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {
                                DB_Reader.Read();
                                result = result + DB_Reader.GetInt32(0);
                            }
                            con.Close();
                        }
                    }

                }
                lblRea.Text = result.ToString();
            } catch (Exception x)
```

```
            {
                MessageBox.Show("Error Reability Count, Close the program and
open again, application will show result count after that");
            }
        }
        private void CountResponsiveness()
        {
            int hasil = 0;
            try
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 2 && id.Status == "pos")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {

                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);

                            }
                            con.Close();
                        }

                    }
                }
                lblRes.Text = hasil.ToString();
            }catch (Exception x)
            {
                MessageBox.Show("Error Reability Count, Close the program and
open again, application will show result count after that");
            }
        }

        private void CountAccess()
        {
            int hasil = 0;
            try
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 3 && id.Status == "pos")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {

                                    DB_Reader.Read();
                                    hasil = hasil + DB_Reader.GetInt32(0);

                            }
                            con.Close();
                        }
```

```
                    }
                }
                lblAcc.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Access Count, Close the program and open
again, application will show result count after that");
            }
        }

        private void CountFletyxibili()
        {
            int hasil = 0;
            try
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 4 && id.Status == "pos")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {
                                while (DB_Reader.Read()) // <- You say here
while reader returns a data do the stuff in statement block.
                                {
                                    hasil = hasil + DB_Reader.GetInt32(0);
                                }
                            }
                            con.Close();
                        }

                    }
                }
                lblFle.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Flexibility Count, Close the program and
open again, application will show result count after that");
            }
        }

        private void CountEasyOfNavigation()
        {
            int hasil = 0;
            try
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 5 && id.Status == "pos")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
```

```
                            if (DB_Reader.HasRows)
                            {
                                while (DB_Reader.Read()) // <- You say here
while reader returns a data do the stuff in statement block.
                                {
                                    hasil = hasil + DB_Reader.GetInt32(0);
                                }
                            }
                            con.Close();
                        }

                    }
                }
                lblEas.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Easy of Navigation Count, Close the
program and open again, application will show result count after that");
            }
        }

        private void CountEfficiency()
        {
            int hasil = 0;
            try
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 6 && id.Status == "pos")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {
                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);
                            }
                            con.Close();
                        }

                    }
                }
                lblEff.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Efficiency Count, Close the program and
open again, application will show result count after that");
            }
        }

        private void CountAssurance()
        {
            int hasil = 0;
            try
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 7 && id.Status == "pos")
```

```
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {
                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);
                            }
                            con.Close();
                        }

                    }
                }
                lblAss.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Assurance Count, Close the program and
open again, application will show result count after that");
            }
        }

        private void CountSecurity()
        {
            int hasil = 0;
            try
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 8 && id.Status == "pos")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {
                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);
                            }
                            con.Close();
                        }

                    }
                }
                lblSec.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Security Count, Close the program and
open again, application will show result count after that");
            }
        }

        private void CountPriveKnowledge()
        {
            int hasil = 0;
            try
```

```
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 9 && id.Status == "pos")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {
                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);
                            }
                            con.Close();
                        }

                    }
                }
                lblPKno.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Price Knowlogde Count, Close the program
and open again, application will show result count after that");
            }
        }

        private void CountSiteAesthetics()
        {
            int hasil = 0;
            try
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 10 && id.Status == "pos")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {
                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);
                            }
                            con.Close();
                        }

                    }
                }
                lblAes.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Site Aesthetic Count, Close the program
and open again, application will show result count after that");
            }
        }
```

```
        private void CountCustomization()
        {
            try
            {
                int hasil = 0;
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 11 && id.Status == "pos")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {
                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);
                            }
                            con.Close();
                        }

                    }
                }
                lblCus.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Customization Count, Close the program
and open again, application will show result count after that");
            }
        }
        #endregion
        #region keyword negatif
        private void NegCountReliability()
        {
            try
            {
                int result = 0;

                foreach (var key in list)
                {

                    if (key.Id_dimensi == 1 && key.Status == "neg")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + key.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {
                                DB_Reader.Read();
                                result = result + DB_Reader.GetInt32(0);
                            }
                            con.Close();
                        }
                    }

                }
                lblReaNeg.Text = result.ToString();
            }
```

```
            catch (Exception x)
            {
                MessageBox.Show("Error Reability Count, Close the program and
open again, application will show result count after that");
            }
        }
        private void NegCountResponsiveness()
        {
            int hasil = 0;
            try
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 2 && id.Status == "neg")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {

                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);

                            }
                            con.Close();
                        }

                    }
                }
                lblResNeg.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Reability Count, Close the program and
open again, application will show result count after that");
            }
        }

        private void NegCountAccess()
        {
            int hasil = 0;
            try
            {
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 3 && id.Status == "neg")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {

                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);

                            }
```

```
                        con.Close();
                    }

                }
            }
            lblAccNeg.Text = hasil.ToString();
        }
        catch (Exception x)
        {
            MessageBox.Show("Error Access Count, Close the program and open
again, application will show result count after that");
        }
    }

    private void NegCountFletyxibili()
    {
        int hasil = 0;
        try
        {
            foreach (var id in list)
            {
                if (id.Id_dimensi == 4 && id.Status == "neg")
                {
                    using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                    {
                        con.Open();
                        OleDbDataReader DB_Reader = Command.ExecuteReader();
                        if (DB_Reader.HasRows)
                        {
                            DB_Reader.Read();
                            hasil = hasil + DB_Reader.GetInt32(0);
                        }
                        con.Close();
                    }

                }
            }
            lblFleNeg.Text = hasil.ToString();
        }
        catch (Exception x)
        {
            MessageBox.Show("Error Flexibility Count, Close the program and
open again, application will show result count after that");
        }
    }

    private void NegCountEasyOfNavigation()
    {
        int hasil = 0;
        try
        {
            foreach (var id in list)
            {
                if (id.Id_dimensi == 5 && id.Status == "neg")
                {
                    using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                    {
                        con.Open();
                        OleDbDataReader DB_Reader = Command.ExecuteReader();
                        if (DB_Reader.HasRows)
```

```
                        {
                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);
                        }
                        con.Close();
                    }

                }
            }
            lblEasNeg.Text = hasil.ToString();
        }
        catch (Exception x)
        {
            MessageBox.Show("Error Easy of Navigation Count, Close the
program and open again, application will show result count after that");
        }
    }

    private void NegCountEfficiency()
    {
        int hasil = 0;
        try
        {
            foreach (var id in list)
            {
                if (id.Id_dimensi == 6 && id.Status == "neg")
                {
                    using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                    {
                        con.Open();
                        OleDbDataReader DB_Reader = Command.ExecuteReader();
                        if (DB_Reader.HasRows)
                        {
                            DB_Reader.Read();
                            hasil = hasil + DB_Reader.GetInt32(0);
                        }
                        con.Close();
                    }

                }
            }
            lblEffNeg.Text = hasil.ToString();
        }
        catch (Exception x)
        {
            MessageBox.Show("Error Efficiency Count, Close the program and
open again, application will show result count after that");
        }
    }

    private void NegCountAssurance()
    {
        int hasil = 0;
        try
        {
            foreach (var id in list)
            {
                if (id.Id_dimensi == 7 && id.Status == "neg")
                {
                    using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
```

```
                    {
                        con.Open();
                        OleDbDataReader DB_Reader = Command.ExecuteReader();
                        if (DB_Reader.HasRows)
                        {
                            DB_Reader.Read();
                            hasil = hasil + DB_Reader.GetInt32(0);
                        }
                        con.Close();
                    }

                }
            }
            lblAssNeg.Text = hasil.ToString();
        }
        catch (Exception x)
        {
            MessageBox.Show("Error Assurance Count, Close the program and
open again, application will show result count after that");
        }
    }

    private void NegCountSecurity()
    {
        int hasil = 0;
        try
        {
            foreach (var id in list)
            {
                if (id.Id_dimensi == 8 && id.Status == "neg")
                {
                    using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                    {
                        con.Open();
                        OleDbDataReader DB_Reader = Command.ExecuteReader();
                        if (DB_Reader.HasRows)
                        {
                            DB_Reader.Read();
                            hasil = hasil + DB_Reader.GetInt32(0);
                        }
                        con.Close();
                    }

                }
            }
            lblSecNeg.Text = hasil.ToString();
        }
        catch (Exception x)
        {
            MessageBox.Show("Error Negative Security Count, Close the
program and open again, application will show result count after that");
        }
    }

    private void NegCountPriveKnowledge()
    {
        int hasil = 0;
        try
        {
            foreach (var id in list)
            {
                if (id.Id_dimensi == 9 && id.Status == "neg")
```

```
                {
                    using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                    {
                        con.Open();
                        OleDbDataReader DB_Reader = Command.ExecuteReader();
                        if (DB_Reader.HasRows)
                        {
                            DB_Reader.Read();
                            hasil = hasil + DB_Reader.GetInt32(0);
                        }
                        con.Close();
                    }

                }
            }
            lblPKnoNeg.Text = hasil.ToString();
        }
        catch (Exception x)
        {
            MessageBox.Show("Error Negative Price Knowlogde Count, Close the
program and open again, application will show result count after that");
        }
    }

    private void NegCountSiteAesthetics()
    {
        int hasil = 0;
        try
        {
            foreach (var id in list)
            {
                if (id.Id_dimensi == 10 && id.Status == "neg")
                {
                    using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                    {
                        con.Open();
                        OleDbDataReader DB_Reader = Command.ExecuteReader();
                        if (DB_Reader.HasRows)
                        {
                            DB_Reader.Read();
                            hasil = hasil + DB_Reader.GetInt32(0);
                        }
                        con.Close();
                    }
                }
            }
            llblAesNeg.Text = hasil.ToString();
        }
        catch (Exception x)
        {
            MessageBox.Show("Error Negative Site Aesthetic Count, Close the
program and open again, application will show result count after that");
        }
    }

    private void NegCountCustomization()
    {
        try
        {
            int hasil = 0;
```

```
                foreach (var id in list)
                {
                    if (id.Id_dimensi == 11 && id.Status == "neg")
                    {
                        using (OleDbCommand Command = new OleDbCommand(" SELECT
count(komentar) from tbl_komentar where komentar like '%" + id.Katakunci + "%'
", con))
                        {
                            con.Open();
                            OleDbDataReader DB_Reader = Command.ExecuteReader();
                            if (DB_Reader.HasRows)
                            {
                                DB_Reader.Read();
                                hasil = hasil + DB_Reader.GetInt32(0);
                            }
                            con.Close();
                        }
                    }
                }
                lblCusNeg.Text = hasil.ToString();
            }
            catch (Exception x)
            {
                MessageBox.Show("Error Customization Count, Close the program
and open again, application will show result count after that");
            }
        }
        #endregion
        private void LoadDimensi()
        {
            try {
                    string strSql = "Select dimensi from tbl_dimensi";
                    OleDbDataAdapter adapter = new OleDbDataAdapter(new
OleDbCommand(strSql, con));
                    DataSet ds = new DataSet();
                    adapter.Fill(ds);
                    cmbDimensi.DataSource = ds.Tables[0];
                    cmbDimensi.DisplayMember = "Pilih Dimensi";
                    cmbDimensi.ValueMember = "dimensi";

                    cmbNegDimensi.DataSource = ds.Tables[0];
                     cmbNegDimensi.DisplayMember = "Pilih Dimensi";
                    cmbNegDimensi.ValueMember = "dimensi";
                con.Close();
            } catch (Exception x) {
                MessageBox.Show(x.ToString());
                con.Close();
            }
        }
        private void showKataKunci(int selectedIndex)
        {
            try
            {
                con.Open();
                selectedIndex = selectedIndex + 1;
                string strSql = "Select id_kata_kunci, kata_kunci from
tbl_kata_kunci where id_dimensi="+ selectedIndex + " and status='pos'";
                OleDbDataAdapter adapter = new OleDbDataAdapter(new
OleDbCommand(strSql, con));
                DataSet ds = new DataSet();
                adapter.Fill(ds);
                con.Close();
                dataGridView1.DataSource = ds.Tables[0];
                dataGridView1.Columns[0].Visible = false;
```

```
          dataGridView1.Show();
            }
            catch (Exception x)
            {
                MessageBox.Show(x.ToString());
                con.Close();
            }
        }
        private void showKataKunciNegative(int selectedIndex)
        {
            try
            {
                con.Open();
                selectedIndex = selectedIndex + 1;
                string strSql = "Select id_kata_kunci, kata_kunci from
tbl_kata_kunci where id_dimensi=" + selectedIndex + " and status='neg'";
                OleDbDataAdapter adapter = new OleDbDataAdapter(new
OleDbCommand(strSql, con));
                DataSet ds = new DataSet();
                adapter.Fill(ds);
                con.Close();
                dataGridView2.DataSource = ds.Tables[0];
                dataGridView2.Columns[0].Visible = false;
                dataGridView2.Show();

            }
            catch (Exception x)
            {
                MessageBox.Show(x.ToString());
                con.Close();
            }
        }
        private void cmbDimensi_SelectedIndexChanged(object sender, EventArgs e)
        {
            try
            {
                showKataKunci(cmbDimensi.SelectedIndex);
            } catch (Exception x)
            {
                MessageBox.Show(x.ToString());
                con.Close();
            }
        }
        private void button1_Click(object sender, EventArgs e)
        {
            if(button1.Text=="Add")
            {
                button1.Text = "Save";
                cmbDimensi.Enabled = false;
       textBox1.Enabled = true;
            }
            else
            {
                try
                {
                    OleDbCommand cmd = new OleDbCommand();
                    cmd.Connection = con;
                    string strSql = "insert into tbl_kata_kunci(kata_kunci,
id_dimensi, status) values('" + textBox1.Text + "', " +
(cmbDimensi.SelectedIndex + 1) + ", 'pos')";
                    con.Open();
                    cmd.CommandText = strSql;
                    cmd.ExecuteNonQuery();
                    con.Close();
```

```
                    showKataKunci(cmbDimensi.SelectedIndex);
                    textBox1.Text = "";
                    button1.Text = "Add";
                    MessageBox.Show("Succesfully Add Keyword");
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString());
                }
            }
        }
        private void dataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
        {
            dataGridView1_KeyUp(sender, null);
        }
        private void dataGridView1_KeyUp(object sender, KeyEventArgs e)
        {
            lbl_id_kata_kunci.Text = getKolom(dataGridView1, 0);
            textBox1.Text = getKolom(dataGridView1, 1);
        }
        private string getKolom(DataGridView dg, int i)
        {
            try
            {
                return dg[dg.Columns[i].Index,
dg.CurrentRow.Index].Value.ToString();
            }
            catch (Exception ex)
            {
                return null;
            }
        }
        private void button2_Click(object sender, EventArgs e)
        {
            try
     {
                OleDbCommand cmd = new OleDbCommand();
                cmd.Connection = con;
                if (textBox1.Text != "")
                {
                    string strSql = "delete from tbl_kata_kunci where
id_kata_kunci = " + int.Parse(lbl_id_kata_kunci.Text) + "";
                    DialogResult dialogResult = MessageBox.Show("Are you sure to
delete keyword " + textBox1.Text + " ???", "Confirmation",
MessageBoxButtons.YesNo);
                    if (dialogResult == DialogResult.Yes)
                    {
                        con.Open();
                        cmd.CommandText = strSql;
                        cmd.ExecuteNonQuery();
                        con.Close();
                        showKataKunci(cmbDimensi.SelectedIndex);
                        textBox1.Text = "";
                        MessageBox.Show("Succesfully Delete Keyword");
                    }
                    else if (dialogResult == DialogResult.No)
                    { }
                }
                else {
                    MessageBox.Show("Please click keyword in the table which you
want to delete");
                }
            }
```

```
            catch (Exception ex)
            {
                MessageBox.Show("Error in delete method, call ur developer");
            }
        }
        private void cmbNegDimensi_SelectedIndexChanged(object sender, EventArgs
e)
        {
            try
            {
                showKataKunciNegative(cmbNegDimensi.SelectedIndex);
            }
            catch (Exception x)
            {
                MessageBox.Show(x.ToString());
                con.Close();
            }
        }
        private void dataGridView2_KeyUp(object sender, KeyEventArgs e)
        {
            lbl_id_kata_kunci_neg.Text = getKolom(dataGridView2, 0);
            textBox2.Text = getKolom(dataGridView2, 1);
        }
        private void dataGridView2_CellContentClick(object sender,
DataGridViewCellEventArgs e)
        {
            dataGridView2_KeyUp(sender, null);
        }
        private void button5_Click(object sender, EventArgs e)
        {
            if (button5.Text == "Add")
            {
                button5.Text = "Save";
                cmbNegDimensi.Enabled = false;
        textBox2.Enabled = true;
            }
            else
            {
                try
                {
                    OleDbCommand cmd = new OleDbCommand();
                    cmd.Connection = con;

                    string strSql = "insert into tbl_kata_kunci(kata_kunci,
id_dimensi, status) values('" + textBox2.Text + "', " +
(cmbNegDimensi.SelectedIndex + 1) + ", 'neg')";

                    con.Open();
                    cmd.CommandText = strSql;
                    cmd.ExecuteNonQuery();
                    con.Close();
                    showKataKunciNegative(cmbNegDimensi.SelectedIndex);
                    textBox2.Text = "";
                    button5.Text = "Add";
                    MessageBox.Show("Succesfully Add Keyword");
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString());
                }
            }
        }
        private void button3_Click(object sender, EventArgs e)
        {
```

```
            button5.Text = "Add";
            cmbNegDimensi.Enabled = true;
            cmbNegDimensi.SelectedIndex = 0;
            textBox2.Enabled = false;
            textBox2.Text = "";
            lbl_id_kata_kunci_neg.Text = "0";
        }
        private void button4_Click(object sender, EventArgs e)
        {
            try
            {
                if (textBox2.Text!="") {
                    OleDbCommand cmd = new OleDbCommand();
                    cmd.Connection = con;

                    string strSql = "delete from tbl_kata_kunci where
id_kata_kunci = " + int.Parse(lbl_id_kata_kunci_neg.Text) + "";
                    DialogResult dialogResult = MessageBox.Show("Are you sure to
delete keyword " + textBox2.Text + " ???", "Confirmation",
MessageBoxButtons.YesNo);
                    if (dialogResult == DialogResult.Yes)
                    {
                        con.Open();
                        cmd.CommandText = strSql;
                        cmd.ExecuteNonQuery();
                        con.Close();
                        showKataKunciNegative(cmbNegDimensi.SelectedIndex);
                        textBox2.Text = "";
                        MessageBox.Show("Succesfully Delete Keyword");
                    }
                    else if (dialogResult == DialogResult.No)
                    { }
                } else {
                    MessageBox.Show("Please click keyword in the table which you
want to delete");
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show("Error in delete method, call ur developer");
            }
        }
        private void dataGridView2_CellClick(object sender,
DataGridViewCellEventArgs e)
        {
            dataGridView2_KeyUp(sender, null);
        }
        private void dataGridView1_CellClick(object sender,
DataGridViewCellEventArgs e)
        {
            dataGridView1_KeyUp(sender, null);
        }
        private void btnClear_Click(object sender, EventArgs e)
        {
            button1.Text = "Add";
            cmbDimensi.Enabled = true;
            cmbDimensi.SelectedIndex = 0;
            textBox1.Enabled = false;
            textBox1.Text = "";
            lbl_id_kata_kunci.Text = "0";
        }
    }
}
```