

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Саратовский государственный технический университет имени Гагарина Ю. А.»

Кафедра
«Прикладные информационные технологии»

ОТЧЕТ
по лабораторной работе №5

Студента группы 62-ПИНФ21
Нефёдова Д. В.

Саратов, 2020

```
using DAL;
using Entities;
using System;
using System.Collections.Generic;
using System.Text.RegularExpressions;

namespace BLL
{
    public enum OperationResult
    {
        OK,
        UserExists,
        UserNotFound,
        BadWords
    }

    public static class Bll
    {
        public static string GetPassword(string user)
        {
            return Dal.GetPassword(user);
        }

        public static List<Message> GetMessegges(string user)
        {
            return Dal.GetMessegges(user);
        }

        public static OperationResult AddUser(string user, string password, string fullName)
        {
            if (GetPassword(user) == "")
            {
                Dal.AddUser(user, password, fullName);
                return OperationResult.OK;
            }
            else
            {
                return OperationResult.UserExists;
            }
        }

        public static OperationResult AddMessege(string from, string to, string sub, string body)
        {
            Regex r = new Regex(@"\b(fuck)|(idiot)\b");

            if (Dal.GetPassword(to) == "")
            {
                return OperationResult.UserNotFound;
            }
        }
    }
}
```

```
    }
    else if (r.IsMatch(body))
    {
        return OperationResult.BadWords;
    }
    else
    {
        Dal.AddMessege(from, to, sub, body, DateTime.Now);
        return OperationResult.OK;
    }
}
}
```

```
using Entities;
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.SqlClient;

namespace DAL
{
    public static class Dal
    {
        public static string GetPassword(string user)
        {
            string conStr =
System.Configuration.ConfigurationManager.ConnectionStrings["DB1"].ConnectionString;
            string pass = "";

            using (SqlConnection con = new SqlConnection(conStr))
            {
                SqlCommand com = new SqlCommand("[GetPassword]", con)
                {
                    CommandType = CommandType.StoredProcedure
                };

                com.Parameters.Add(new SqlParameter("@user", user));
                con.Open();
                object obj = com.ExecuteScalar();
                con.Close();

                if (obj != null)
                    pass = obj.ToString();
            }

            return pass;
        }

        public static void AddUser(string user, string password, string fullName)
        {
            string conStr =
System.Configuration.ConfigurationManager.ConnectionStrings["DB1"].ConnectionString;

            using (SqlConnection con = new SqlConnection(conStr))
            {
                SqlCommand com = new SqlCommand("[AddUser]", con);
                com.CommandType = CommandType.StoredProcedure;
                com.Parameters.Add(new SqlParameter("@user", user));
                com.Parameters.Add(new SqlParameter("@password", password));
                com.Parameters.Add(new SqlParameter("@fullName", fullName));
                con.Open();
            }
        }
    }
}
```

```
        com.ExecuteNonQuery();
        con.Close();
    }
}

public static void AddMessege(string from, string to, string sub, string body, DateTime date)
{
    string conStr =
System.Configuration.ConfigurationManager.ConnectionStrings["DB1"].ConnectionString;

    using (SqlConnection con = new SqlConnection(conStr))
    {
        SqlCommand com = new SqlCommand("[AddMessege]", con);
        com.CommandType = CommandType.StoredProcedure;
        com.Parameters.Add(new SqlParameter("@from", from));
        com.Parameters.Add(new SqlParameter("@to", to));
        com.Parameters.Add(new SqlParameter("@sub", sub));
        com.Parameters.Add(new SqlParameter("@body", body));
        com.Parameters.Add(new SqlParameter("@date", date));
        con.Open();
        com.ExecuteNonQuery();
        con.Close();
    }
}

public static List<Message> GetMessegas(string user)
{
    List<Message> messegas = new List<Message>();
    string conStr =
System.Configuration.ConfigurationManager.ConnectionStrings["DB1"].ConnectionString;

    using (SqlConnection con = new SqlConnection(conStr))
    {
        SqlCommand com = new SqlCommand("[GetMessegas]", con);
        com.CommandType = CommandType.StoredProcedure;
        com.Parameters.Add(new SqlParameter("@user", user));
        con.Open();
        SqlDataReader reader = com.ExecuteReader();

        while (reader.Read())
        {
            Message m = new Message();
            m.Id = (int)reader["MsgId"];
            m.To = reader["MsgTo"].ToString();
            m.From = reader["MsgFrom"].ToString();
            m.Sub = reader["MsgSub"].ToString();
            m.Body = reader["MsgBody"].ToString();
            m.Date = (DateTime)reader["MsgDate"];
        }
    }
}
```

```
        messeges.Add(m);
    }

    con.Close();
}

return messeges;
}
}
```

```
using BLL;
using System;
using System.Windows.Forms;

namespace Lab_5
{
    public partial class ComposeForm : Form
    {
        public ComposeForm()
        {
            InitializeComponent();
        }

        private void CancelBtn_Click(object sender, EventArgs e)
        {
            this.Close();
        }

        private void SendBtn_Click(object sender, EventArgs e)
        {
            Messenger m = (Messenger) MdiParent;

            if (Bll.AddMessege(m.currUser, toTxt.Text, subTxt.Text, bodyRTxt.Text) ==
                OperationResult.BadWords)
                errorLbl1.Text = "There are bad words in the message.";
            else
                this.Close();
        }
    }
}
```

```
using BLL;
using System;
using System.Windows.Forms;

namespace Lab_5
{
    public partial class LoginForm : Form
    {
        public LoginForm()
        {
            InitializeComponent();

            public event Action Login;

            private void CancelBtn_Click(object sender, EventArgs e)
            {
                Application.Exit();
            }

            private void LoginBtn_Click(object sender, EventArgs e)
            {
                string password = Bll.GetPassword(userNameTxt.Text);

                if (password.Equals(passwordTxt.Text))
                {
                    Messenger m = (Messenger)MdiParent;
                    m.currUser = userNameTxt.Text;
                    Login?.Invoke();
                    Close();
                }
            }

            private void SungUpBtn_Click(object sender, EventArgs e)
            {
                RegistrationForm rf = new RegistrationForm();
                rf.MdiParent = MdiParent;
                rf.Show();
                Close();
            }
        }
    }
}
```



```
using BLL;
using System;
using System.Windows.Forms;

namespace Lab_5
{
    public partial class RegistrationForm : Form
    {
        public RegistrationForm()
        {
            InitializeComponent();
        }

        private void RegisterBtn_Click(object sender, EventArgs e)
        {
            OperationResult or = Bll.AddUser(userNameTxt.Text, passwordTxt.Text, fullNameTxt.Text);
            if (or == OperationResult.UserExists)
            {
                errorLbl.Text = "User with this user name is exist";
            }
            if (or == OperationResult.OK)
            {
                LoginForm lf = new LoginForm();
                lf.MdiParent = this.MdiParent;
                Messenger m = (Messenger)MdiParent;
                lf.Login += m.LegitUserChangeHandler;
                lf.Login += m.OpenMessegesForm;
                lf.Show();
                Close();
            }
        }
    }
}
```

```

using System;
using System.Windows.Forms;

namespace Lab_5
{
    public partial class MessengesForm : Form
    {
        public MessengesForm()
        {
            InitializeComponent();

            private void FillToolStripButton_Click(object sender, EventArgs e)
            {
                try
                {
                    this.getMessegasTableAdapter.Fill(this.messengerDBDataSet.GetMessegas,
userToolStripTextBox.Text);
                }
                catch (System.Exception ex)
                {
                    System.Windows.Forms.MessageBox.Show(ex.Message);
                }
            }

            private void MessengesForm_Load(object sender, EventArgs e)
            {
                Messenger m = (Messenger)MdiParent;
                try
                {
                    this.getMessegasTableAdapter.Fill(this.messengerDBDataSet.GetMessegas, m.currUser);
                }
                catch (System.Exception ex)
                {
                    System.Windows.Forms.MessageBox.Show(ex.Message);
                }
            }

            private void GetMessegasDataGridView_RowHeaderMouseClick(object sender,
DataGridViewCellMouseEventArgs e)
            {
                if (getMessegasDataGridView.SelectedRows.Count > 0)
                    richTextBox1.Text = getMessegasDataGridView.SelectedRows[0].Cells[3].Value.ToString();
            }
        }
    }
}

```

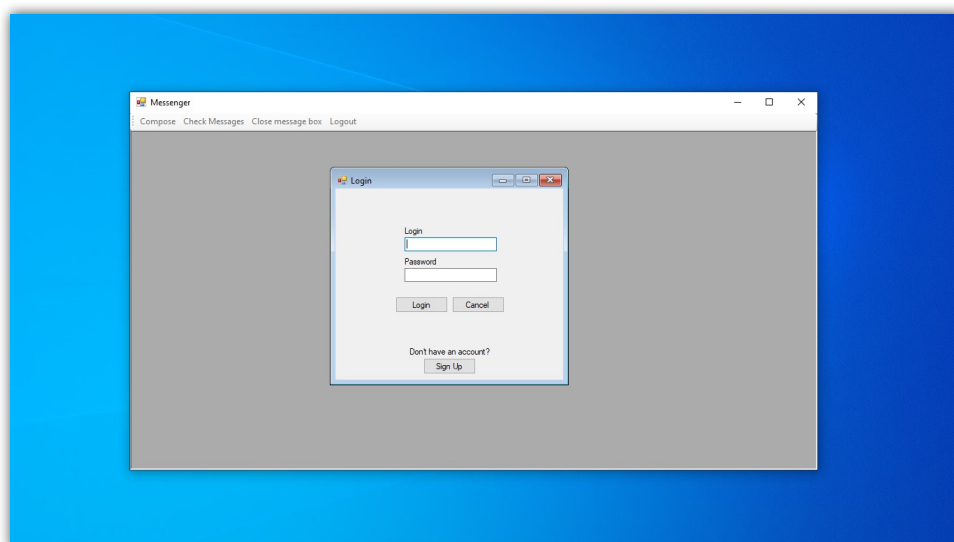


Рис. 1. Форма входа.

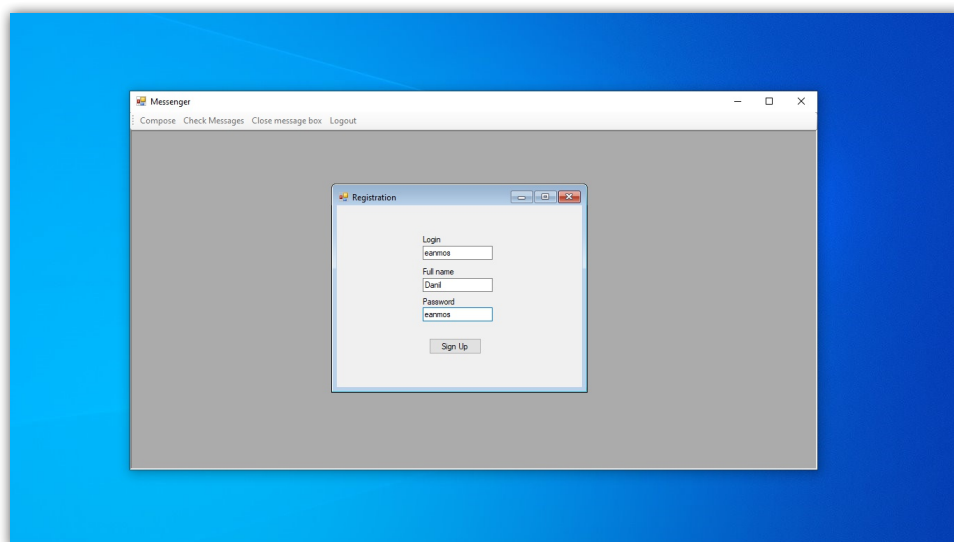


Рис. 2. Форма регистрации.

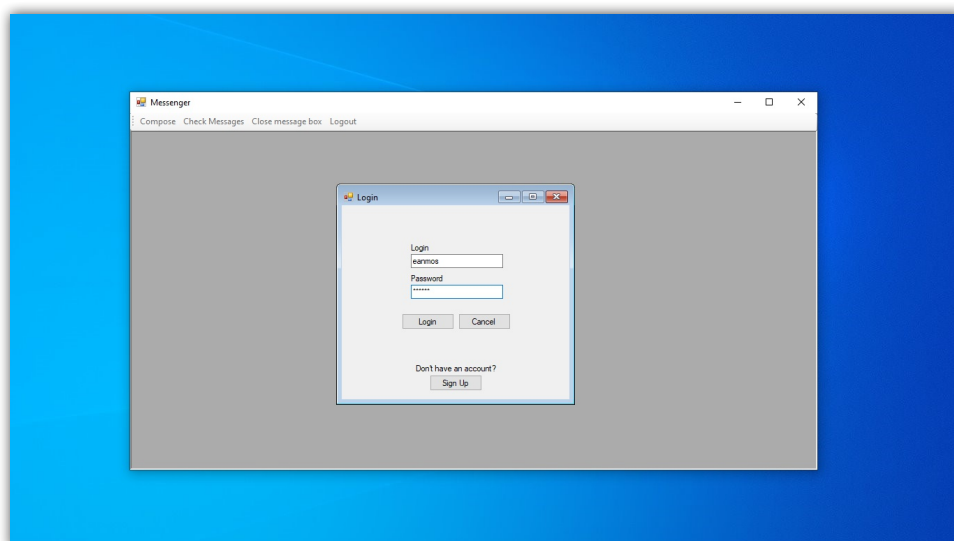


Рис. 3. Форма входа с заполненными полями.

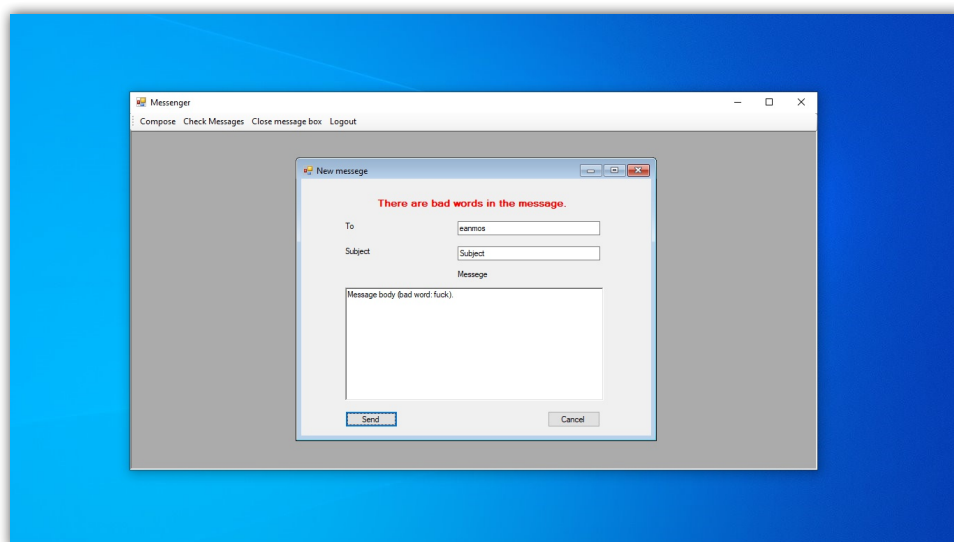


Рис. 4. Проверка текста сообщения на наличие нецензурных слов.

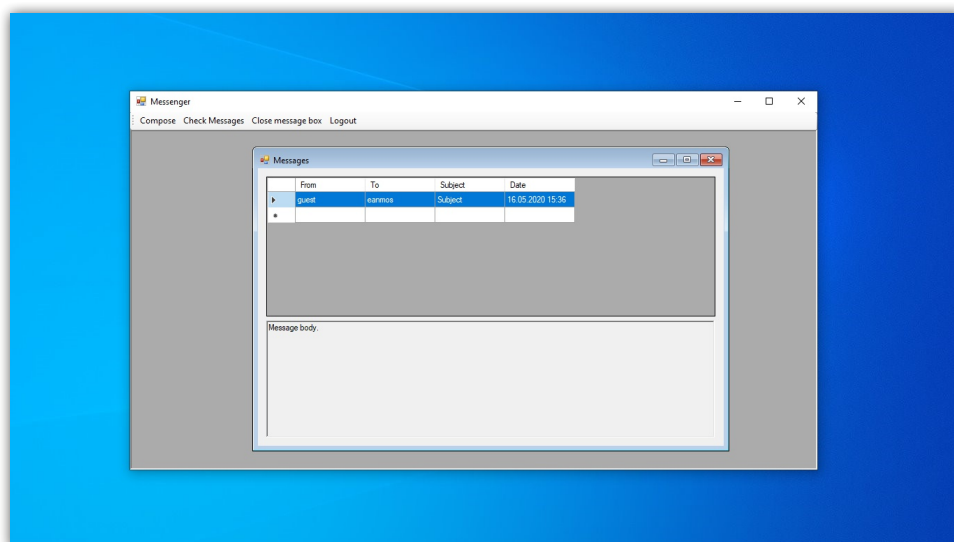


Рис. 5. Просмотр полученных сообщений.