

WE'FRIENDS



Submitted by:
Hussnain Ahmad (2021-CS-121)

Supervised by:
Mam Maida Shahid

Department of Computer Science.
University of Engineering and Technology
Lahore, Pakistan.

Table of content

Contents

1	Short description	3
2	Project Features	3
3	Use cases	3
4	Class Diagram	4
5	Wireframes of Projects:	5
6	Technology and Resources	10
6.1	Front end	10
6.2	Back end	10
7	Full Code	10

1 Short description

We’FRIENDS is a friendly desktop app.It is implemented using graphs.one can be talk with Their friends.user can add friends and see their location and user can send friend request With others.user can see how many people send requests to them also can see his Friends .user can see the active users of app.File handling used for data store .one can see His profile information and user can search id to add friend.Win forms used for Frontend development and backend implement in c-sharp.

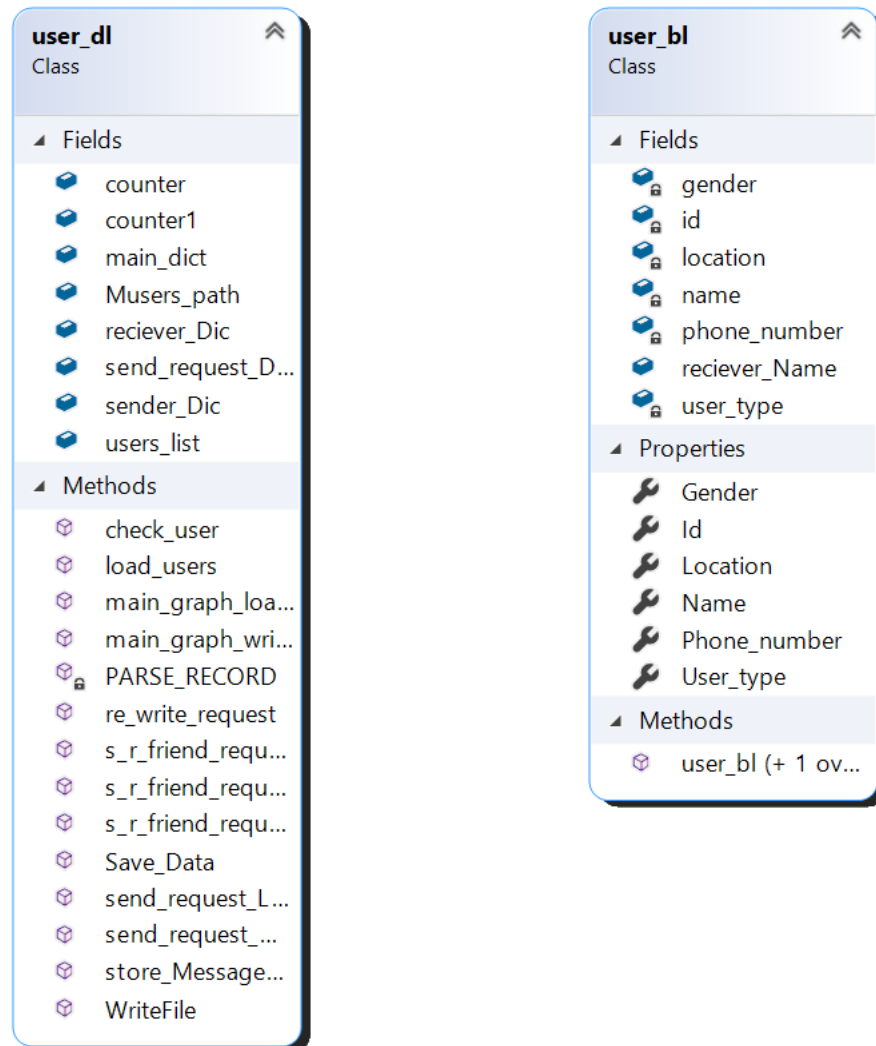
2 Project Features

- User can made friends
- User can chat with friends
- User can send friends requests
- User can see his friends
- User can see his profile
- User can check his friend location
- User can search his friends

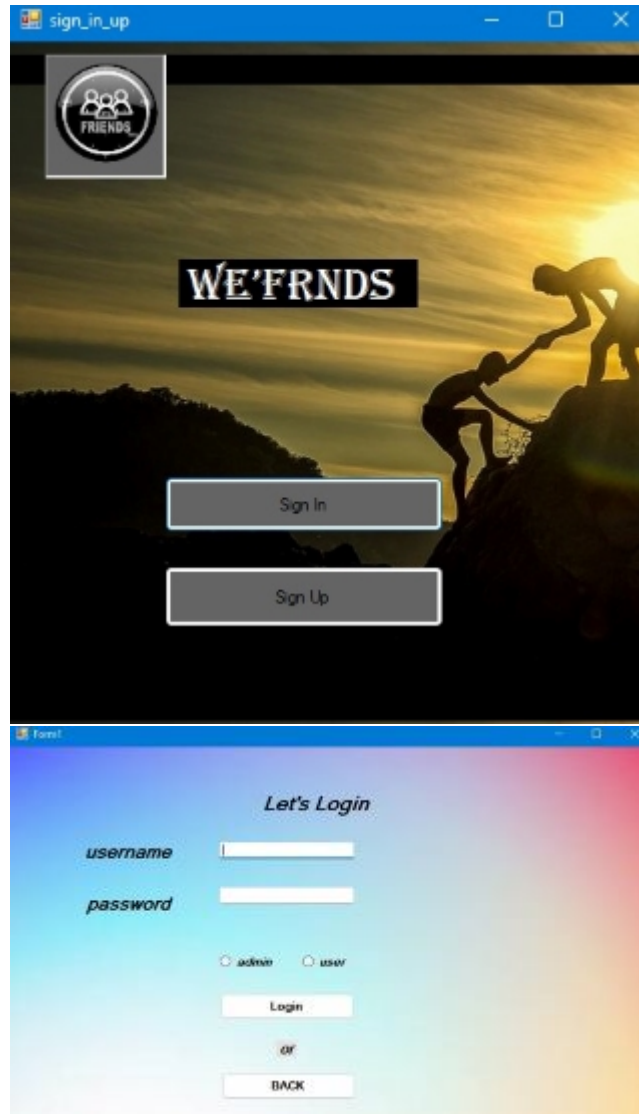
3 Use cases

- It’s a friendly app that use for entertaining
- Users can use this app for chatting
- It’s use for search friend and send and add friends requests
- It’s use for see his friend location
- Multiple user can add and use this app
- User can see his friends
- User can check his personal information and profile

4 Class Diagram



5 Wireframes of Projects:



The screenshot shows a window titled "sign_up_new_user" with a blue header bar. The background is a gradient of blue, purple, and pink. On the left, there is a large black silhouette of a person's head and shoulders, with a white plus sign inside a circle to its right. To the right of the silhouette, there is a dashed rectangular box containing the text "Let's Sign Up !!!". Below this, the form contains the following fields and options:

- Name**: A text input field.
- ID**: A text input field.
- Gender**: Two radio buttons labeled "male" and "female".
- phone #**: A text input field.
- Location**: A dropdown menu.
- user_type**: Two radio buttons labeled "admin" and "user".

At the bottom of the form, there are two buttons: "Back" and "Next".



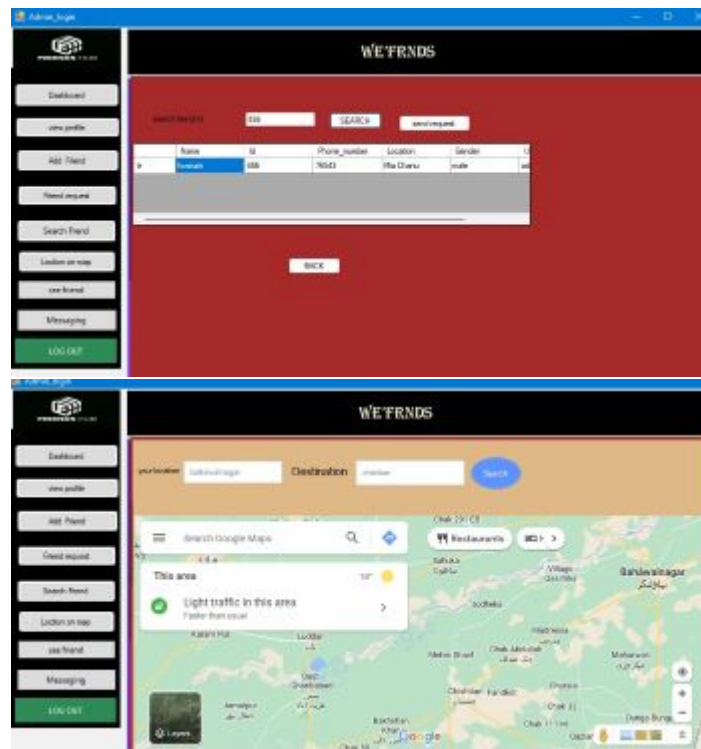
The image displays two screenshots of a web application titled "WE FRIENDS".

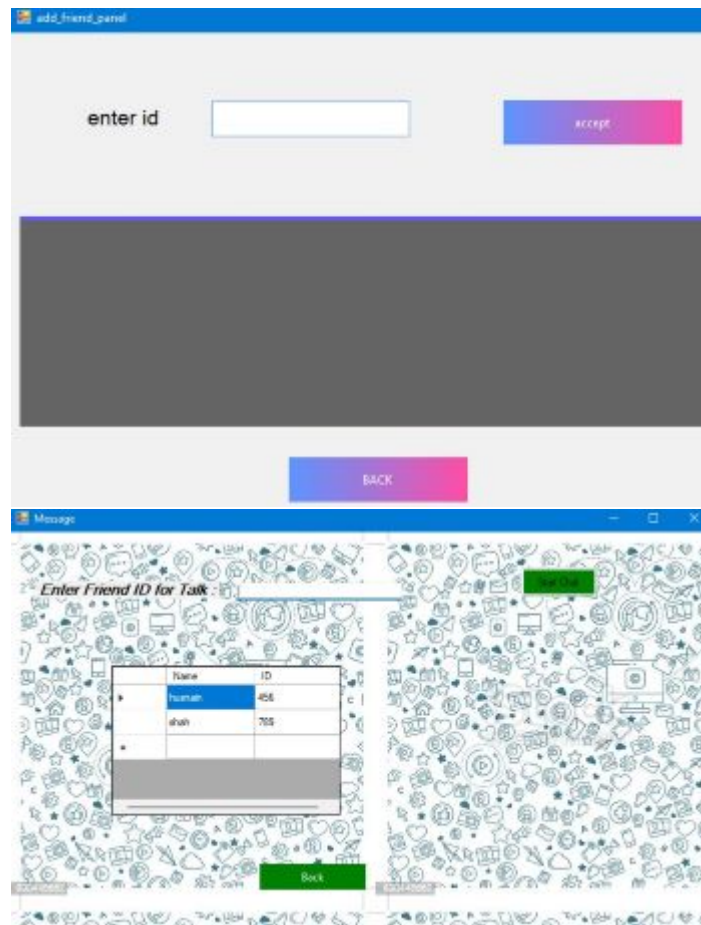
The top screenshot shows a registration form with the following fields and buttons:

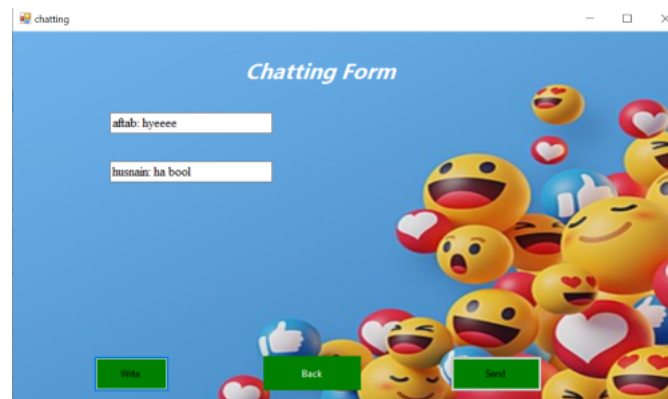
- your name:
- phone#:
- location:
- BACK button

The bottom screenshot shows a search interface with the following elements:

- A large grey rectangular placeholder for a profile picture or map.
- A search input field with a "Search" button.
- A "back" button.







6 Technology and Resources

6.1 Front end

Front end is developed in windows forms using Gunna-extension in C-sharp.NET.

6.2 Back end

Back end is developed in C-sharp. NET. For data permanent store file handling is used. Graphs is implemented Using dictionary .Location is show using google API. Lists are used for total user storage.

7 Full Code

Form 1

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }

    private void login_btn_Click(object sender, EventArgs e)
    {
        //bool flag = false;
        string user_name, password, user_type;
        user_name = username_input.Text;
        password = password_input.Text;
    }
}
```

```
user_type = admin_rb.Text;
if (username_input.Text == "" || password_input.Text == "" || admin_rb.Text == ' ')
{
    MessageBox.Show("Please! Fill All Credentials  :)");
}
else
{
    user_bl new_user = new user_bl();
    new_user = user_dl.check_user(user_name, password, user_type);
    if (new_user == null)
    {
        MessageBox.Show("WE do not have such User :) ");
    }
    else
    {
        if (new_user.User_type == "admin")
        {
            this.Hide();
            Admin_login foam = new Admin_login(new_user);
            foam.Show();
        }
        else
        {
            this.Hide();
            Admin_login foam = new Admin_login(new_user);
            foam.Show();
        }
    }
}
}
```

Sign-IN and Sign-up Form

```
public partial class sign_in_up : Form
{
    public sign_in_up()
    {
        InitializeComponent();
    }

    private void label1_Click(object sender, EventArgs e)
```

```
{  
  
}  
  
private void SignUp_btn_Click(object sender, EventArgs e)  
{  
    this.Hide();  
    sign_up_new_user foam = new sign_up_new_user();  
    foam.Show();  
}  
  
private void signIn_btn_Click(object sender, EventArgs e)  
{  
    this.Hide();  
    Form1 foam = new Form1();  
    foam.Show();  
}  
  
private void panel1_Paint(object sender, PaintEventArgs e)  
{  
  
}  
  
private void sign_in_up_Load(object sender, EventArgs e)  
{  
    user_dl.load_users();  
    user_dl.s_r_friend_request_load_file();  
    user_dl.main_graph_load_file();  
}  
  
private void label2_Click(object sender, EventArgs e)  
{  
  
}  
}
```

SignUpNewUser Form

```
public partial class sign_up_new_user : Form  
{  
    string path = "sem3Final.txt";  
    public sign_up_new_user()  
    {
```

```
InitializeComponent();
comboBox1.Items.Add("Lahore");
comboBox1.Items.Add("Multan");
comboBox1.Items.Add("Karachi");
comboBox1.Items.Add("Peshawar");
comboBox1.Items.Add("Islamabad");
comboBox1.Items.Add("Mia Chanu");
comboBox1.Items.Add("Faisalabad");
comboBox1.Items.Add("Miawali");
comboBox1.Items.Add("Rahim Yar Khan");
comboBox1.Items.Add("Sahiwal");
comboBox1.Items.Add("Abbatabad");
}

private void panel1_Paint(object sender, PaintEventArgs e)
{

}

private void next_btn_Click(object sender, EventArgs e)
{
    string name=name_textbox.Text;
    string id=id_textbox.Text;
    string phone_number=phone_textbox.Text;
    string location = comboBox1.Text;
    string gender;
    if (name_textbox.Text == "" || id_textbox.Text == "" || phone_textbox.Text == "")
    {

        MessageBox.Show("Please! Fill All Credentials :)");
    }
    else
    {

        if (female_rb.Checked == true)
        {
            gender = female_rb.Text;
        }
        else
        {
            gender = male_rb.Text;
        }
        string user_type;
        if (admin_rad_btn.Checked == true)
```

```

        {
            user_type = admin_rad_btn.Text;
        }
        else
        {
            user_type = user_rad_btn.Text;
        }
        user_bl A = new user_bl(name, id, phone_number, location, gender, user_type);
        user_dl.users_list.Add(A);
        MessageBox.Show("successfully added ");
        user_dl.WriteFile(A, path);
        this.Hide();
        sign_in_up foam = new sign_in_up();
        foam.Show();
    }
}

private void back_btn_Click(object sender, EventArgs e)
{
    this.Hide();
    sign_in_up n = new sign_in_up();
    n.Show();
}

```

Admin login Form

```

public partial class Admin_login : Form
{
    int c_box=0;
    user_bl new_user;
    user_bl send_friends_request;
    List<user_bl> local_id_list = new List<user_bl>();
    public Admin_login(user_bl new_user)
    {
        InitializeComponent();
        this.new_user = new_user;
        webBrowser1.ScriptErrorsSuppressed = true;
    }
}

```

```
private void title_panel_Paint(object sender, PaintEventArgs e)
{

}

private void panel4_Paint(object sender, PaintEventArgs e)
{

}

private void view_profile_Click(object sender, EventArgs e)
{
    hide_all_panels();

    dash_board_panel_1.Show();
    view_profile_panel_2.BringToFront();
    view_profile_panel_2.Show();

}

private void dashboard_btn_Click(object sender, EventArgs e)
{

    hide_all_panels();
    dash_board_panel_1.Show();

}

private void active_user_text_TextChanged(object sender, EventArgs e)
{

}

private void dash_board_panel_Paint(object sender, PaintEventArgs e)
{
    active_user_text.Text=" "+ user_dl.users_list.Count;
}

private void view_profile_panel_Paint(object sender, PaintEventArgs e)
{
    pro_name_txtbox.Text = new_user.Name;
    pro_phone_txtbox.Text = new_user.Phone_number;
    pro_locat_txtbox.Text = new_user.Location;
    // MessageBox.Show("i am inside the view profile panel");
}
```

```
    }

    private void profile_back_Click(object sender, EventArgs e)
    {
        hide_all_panels();
        dash_board_panel_1.Show();
    }

    private void Admin_login_Load(object sender, EventArgs e)
    {

        hide_all_panels();
        dash_board_panel_1.Show();
    }
    private void hide_all_panels()
    {
        update_user_panel_3.Hide();
        view_profile_panel_2.Hide();
        dash_board_panel_1.Hide();
    }

    private void update_user_Click(object sender, EventArgs e)
    {
        hide_all_panels();
        update_user_panel_3.Show();
    }
}

private void add_friend_btn_Click(object sender, EventArgs e)
{
    add_friend_panel f = new add_friend_panel(new_user);
    f.Show();
    this.Close();
}

private void searchButton_Click(object sender, EventArgs e)
{

    string name = id_Box.Text;
    MessageBox.Show(user_dl.users_list.Count.ToString());
    foreach(user_bl user in user_dl.users_list)
```

```
{
    if(name == user.Name)
    {
        local_id_list.Add(user);
    }
}
dataGV.DataSource = local_id_list;

}

private void search_panel_4_Paint(object sender, PaintEventArgs e)
{

}

private void update_user_panel_3_Paint(object sender, PaintEventArgs e)
{

}

private void search_btn_Click(object sender, EventArgs e)
{
    dash_board_panel_1.Visible = true;
    view_profile_panel_2.Visible = true;
    update_user_panel_3.Visible = true;
    search_panel_4.Visible = true;
    frien_resuestsss_panel_5.Visible = false;
    map_panel_6.Visible = false;
}

private void add_friend_panel_Paint(object sender, PaintEventArgs e)
{

}

private void send_reuest_btn_Click(object sender, EventArgs e)
{

    if (send_friends_request != null)
    {
        if (!user_dl.send_request_Dict.ContainsKey(send_friends_request))
        {
            user_dl.send_request_Dict.Add(send_friends_request, new List<user_bl>())
        }
        if (!user_dl.send_request_Dict.ContainsKey(new_user))
    }
}
```

```

        {
            user_dl.send_request_Dict.Add(new_user, new List<user_bl>());
        }
        user_dl.send_request_Dict[send_friends_request].Add(new_user);
        user_dl.s_r_friend_request_write_file(new_user, send_friends_request);
        local_id_list.Clear();
        dataGridView1.DataSource = null;

    }
    else
    {
        MessageBox.Show("please select friend you wanted to send request");
    }

}

private void ADD_FRIEND_SEARCH_BTN_Click(object sender, EventArgs e)
{
    string id = add_friend_tbox.Text;
    foreach (user_bl user in user_dl.users_list)
    {
        if (id == user.Id)
        {
            local_id_list.Add(user);
            send_friends_request = user;
            dataGridView1.DataSource = local_id_list;
            break;
        }
    }
}

private void add_friend_back_btn_Click(object sender, EventArgs e)
{
    hide_all_panels();
    dash_board_panel_1.Show();
}

private void friend_request_btn_Click(object sender, EventArgs e)
{
    dash_board_panel_1.Visible = true;
    view_profile_panel_2.Visible = true;
    update_user_panel_3.Visible = true;
    search_panel_4.Visible = true;
    frien_resuestsss_panel_5.Visible = true;
    // search_panel_4.Visible = false;
    map_panel_6.Visible = false;
}

```

```
    }

    private void dataGridView1_CellContentClick(object sender, DataGridViewCellEventArgs e)
    {

    }

    private void dataGV_CellContentClick(object sender, DataGridViewCellEventArgs e)
    {

    }

    private void button1_Click(object sender, EventArgs e)
    {
        hide_all_panels();
        sign_in_up n =new sign_in_up();
        n.Show();
    }

    private void button2_Click(object sender, EventArgs e)
    {
        hide_all_panels();
        dash_board_panel_1.Show();
    }

    private void webBrowser1_DocumentCompleted(object sender, WebBrowserDocumentCompletedEventArgs e)
    {

    }

    private void guna2find_Click(object sender, EventArgs e)
    {
        string current_location = guna2Location.Text;
        string distination = guna2Destination.Text;
        StringBuilder queryAddress = new StringBuilder();
        //this.map_panel_6.MaximumSize;
        queryAddress.Append("https://www.google.com/maps/dir/");
        if (current_location != string.Empty)
        {
            queryAddress.Append(current_location + "/");
        }

        if (distination != string.Empty)
        {
```

```
        queryAddress.Append(distination + "/");
    }
    webBrowser1.Navigate(queryAddress.ToString());
}

private void loc_map_btn_Click(object sender, EventArgs e)
{
    hide_all_panels();
    dash_board_panel_1.Visible = true;
    view_profile_panel_2.Visible = true;
    update_user_panel_3.Visible = true;
    search_panel_4.Visible = true;
    frien_resuestssss_panel_5.Visible = true;
    map_panel_6.Visible = true;
}

private void map_panel_6_Paint(object sender, PaintEventArgs e)
{
}

private void guna2ImageButton1_Click(object sender, EventArgs e)
{
    // PictureBox pic = new PictureBox();
    // Panel panel = new Panel();
    // this.Controls.Add(panel);
    //// pic.Image= sem3Final.re
}

private void fnd_btn_Click(object sender, EventArgs e)
{
    this.Hide();
    SeeMyFriend form = new SeeMyFriend(send_friends_request);
    form.Show();
}

private void label8_Click(object sender, EventArgs e)
{
}

private void guna2GradientButton1_Click(object sender, EventArgs e)
{
    this.Hide();
}
```

```

        SeeMyFriend f = new SeeMyFriend(new_user);
        f.Show();

    }

    private void guna2GradientButton2_Click(object sender, EventArgs e)
    {
        sign_in_up f = new sign_in_up();
        f.Show();
        this.Close();
    }

    private void msg_btn_Click(object sender, EventArgs e)
    {
        Message f = new Message(new_user);
        f.Show();
        this.Close();
    }

    private void pictureBox1_Click(object sender, EventArgs e)
    {
        PictureBox pb = new PictureBox();
        pb.BackgroundImage = sem3Final.Properties.Resources.friends_logo_share;
        pb.Size = new Size(100, 100);
        this.Controls.Add(pb);
    }
}

```

Messaging Form

```

{
    public partial class Message : Form
    {
        public string user = "";
        user_bl new_user;
        public Message(user_bl new_user)
        {
            InitializeComponent();
            this.new_user = new_user;
        }

        private void Message_Load(object sender, EventArgs e)
        {

```

```

        DataTable table = new DataTable();
        table.Columns.Add("Name", typeof(string));
        table.Columns.Add("ID", typeof(string));
        foreach(KeyValuePair<user_bl,List<user_bl>> o in user_dl.main_dict)
        {
            if (o.Key.Name == new_user.Name)
            {
                for (int x = 0; x < o.Value.Count; x++)
                {
                    table.Rows.Add(o.Value[x].Name,o.Value[x].Id);

                }
            }

            DGW.DataSource = table;

        }
    }
    public static void check()
    {

    }

    private void guna2TextBox1_TextChanged(object sender, EventArgs e)
    {

    }

    private void textBox1_TextChanged(object sender, EventArgs e)
    {

    }

    private void button1_Click(object sender, EventArgs e)
    {

        if(text_id.Text == "" )
        {
            MessageBox.Show("Please Enter ID :");
        }
        else
        {
            foreach(KeyValuePair<user_bl,List<user_bl>> v in user_dl.main_dict)
            {

```

```

        if(v.Key.Id == text_id.Text)
        {
            user = v.Key.Name;
        }
    }
    MessageBox.Show(user);
    this.Hide();
    chatting chat = new chatting(new_user.Name,user);
    chat.Show();

}

}

private void guna2GradientButton1_Click(object sender, EventArgs e)
{
    this.Close();
    Admin_login v = new Admin_login(new_user);
    v.Show();
}

```

AddFriendForm

```

public partial class add_friend_panel : Form
{
    user_bl new_user;
    public add_friend_panel(user_bl new_user)
    {
        InitializeComponent();
        this.new_user = new_user;
    }

    private void guna2DataGridView1_CellContentClick(object sender, DataGridViewCellEventArgs e)
    {
    }

    private void guna2Panel1_Paint(object sender, PaintEventArgs e)
    {
        foreach(KeyValuePair<user_bl,List<user_bl>> user in user_dl.send_request_Dict)
        {

```

```

        if(user.Key.Id == new_user.Id)
        {
            accccccep_gv.DataSource = user.Value;
        }
    }
}

private void accccccep_gv_CellContentClick(object sender, DataGridViewCellEventArgs e)
{

}

private void add_btn_Click(object sender, EventArgs e)
{
    string id = text_box_add.Text;
    foreach(user_bl user in user_dl.users_list)
    {
        if(id == user.Id)
        {
            if (!user_dl.main_dict.ContainsKey(user))
            {
                user_dl.main_dict.Add(user, new List<user_bl>());
            }
            if(!user_dl.main_dict.ContainsKey(new_user))
            {
                user_dl.main_dict.Add(new_user, new List<user_bl>());
            }
            user_dl.main_dict[new_user].Add(user);
            user_dl.main_dict[user].Add(new_user);
            user_dl.main_graph_write_data(new_user, user);
            user_dl.s_r_friend_request_delete(new_user, user);
            user_dl.re_write_request("send_receive_friends_data.txt");
            MessageBox.Show(user_dl.counter);
            MessageBox.Show(" Request Accept Successfully ");

            break;
        }
    }
}

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

```



```
private void guna2GradientButton1_Click(object sender, EventArgs e)
{
    Admin_login f = new Admin_login(new_user);
    f.Show();
    this.Close();
}
```

Chatting Form

```
public partial class chatting : Form
{
    public static int X = 120;
    public static int Y = 100;
    public static TextBox mybox1;
    public string send;
    public string reciever;
    public string path="";
    public string path1="";
    public static string current_user="";
    public static bool flag = false;
    public List<string> my_list = new List<string>();

    public chatting(string send, string reciever)
    {
        InitializeComponent();

        this.send = send;
        current_user = send;
        this.reciever = reciever;
        mybox1 = new TextBox();
        path = send + reciever + ".txt";
        path1 = reciever + send + ".txt";
        if (loadMessages(path1) == true)
        {
            MessageBox.Show("Load");
        }
        else
        {
            loadMessages(path);
        }
    }
}
```

```
}
public chatting()
{

}

private bool loadMessages(string path2)
{
    chatting f = new chatting();
    string line = "";
    if (File.Exists(path2))
    {
        StreamReader mess = new StreamReader(path2);
        while ((line = mess.ReadLine()) != null)
        {
            my_list.Add(line);
        }
        mess.Close();
        return true;
    }
    return false;
}

private void load_Messsage_In_Textbox()
{
    for (int i = 0; i < my_list.Count; i++)
    {
        TextBox mybox = new TextBox();
        mybox.Text = my_list[i];
        mybox.Location = new Point(X, Y);
        mybox.BackColor = Color.White;
        mybox.ForeColor = Color.Black ;
        mybox.Font = new Font("Times New Roman",12);
        mybox.Size = new Size(200, 150);
        mybox1 = mybox;
        this.Controls.Add(mybox1);
        Y = Y + 60;
    }

}

private void btn_write_Click(object sender, EventArgs e)
{
    if (!user_dl.sender_Dic.ContainsKey(send))
    {
```

```

        user_dl.sender_Dic.Add(send, new List<string>());
    }
    TextBox box = new TextBox();
    box.Location = new Point(X, Y);
    box.BackColor = Color.White;
    box.ForeColor = Color.Black;
    box.Font = new Font("Times New Roman", 12);
    box.Size = new Size(200, 150);
    mybox1 = box;
    this.Controls.Add(mybox1);
    flag = true;
    Y = Y + 60;

}

private void btn_send_Click(object sender, EventArgs e)
{
    if(flag == true)
    {
        if (mybox1.Text == "" )
        {
            MessageBox.Show("Please Enter some text inside :)");
        }
        else
        {
            storeMessagesintoFile(mybox1.Text, path1);
            storeMessagesintoFile(mybox1.Text, path);
            MessageBox.Show("Message Box Sent Suuccessfully");
        }
        flag = false;
    }
    else
    {
        MessageBox.Show("Please Enter some text inside :)");
    }
}

}

public static void storeMessagesintoFile(string text,string path2)
{
    StreamWriter message = new StreamWriter(path2,true);
    message.WriteLine(text+";"+current_user);
    message.Flush();
}

```

```
        message.Close();
    }

    private void chatting_Load(object sender, EventArgs e)
    {
        load_Messsage_In_Textbox();
    }

    private void guna2Button1_Click(object sender, EventArgs e)
    {
        this.Close();
    }
```

UserBL class

```
public class user_bl
{
    private string user_type;
    private string name;
    private string id;
    private string phone_number;
    private string location;
    private string gender;
    public string reciever_Name;

    public string Name { get => name; set => name = value; }
    public string Id { get => id; set => id = value; }
    public string Phone_number { get => phone_number; set => phone_number = value; }
    public string Location { get => location; set => location = value; }
    public string Gender { get => gender; set => gender = value; }
    public string User_type { get => user_type; set => user_type = value; }

    public user_bl()
    {
    }

    public user_bl(string name,string id,string phone_number, string location,string gender)
    {
        this.user_type = user_type;
        this.name = name;
        this.id = id;
    }
}
```

```

        this.gender = gender;
        this.location = location;
        this.phone_number = phone_number;
    }

```

UserDL class

```

public class user_dl
{
    public static string counter = "";
    public static string counter1 = "";
    public static string Musers_path = "sem3Final.txt";
    public static Dictionary<user_bl, List<user_bl>> main_dict = new Dictionary<user_bl, List<user_bl>>();
    public static Dictionary<user_bl, List<user_bl>> send_request_Dict = new Dictionary<user_bl, List<user_bl>>();
    public static List<user_bl> users_list = new List<user_bl>();

    public static Dictionary<string, List<string>> sender_Dic = new Dictionary<string, List<string>>();
    public static Dictionary<string, List<string>> reciever_Dic = new Dictionary<string, List<string>>();

    public static void re_write_request(string path)
    {
        string path1 = path;
        foreach (KeyValuePair<user_bl, List<user_bl>> friend in send_request_Dict)
        {
            user_bl temp1 = new user_bl();
            temp1 = friend.Key;
            counter = friend.Key.Id;
            foreach (user_bl user in friend.Value)
            {
                user_bl temp2 = new user_bl();
                temp2 = user;
                s_r_friend_request_write_file(temp1, temp2);
            }
        }
    }

    public static void s_r_friend_request_write_file(user_bl s_user, user_bl r_user)
    {
        string path = "";
        path = "send_receive_friends_data.txt";

        StreamWriter file = new StreamWriter(path, true);
    }
}

```

```

        file.WriteLine(s_user.Name + "," + s_user.Id + "," + s_user.Phone_number + "," + s_user.Location + ",");
        file.Flush();
        file.Close();
    }

    public static void main_graph_write_data(user_bl s_user, user_bl r_user)
    {
        string path = "";
        path = "main_graph.txt";
        StreamWriter file = new StreamWriter(path, true);
        file.WriteLine(s_user.Name + "," + s_user.Id + "," + s_user.Phone_number + "," + s_user.Location + ",");
        file.Flush();
        file.Close();
    }

    public static void s_r_friend_request_load_file()
    {
        StreamReader account_read = new StreamReader("send_receive_friends_data.txt");

        string record;
        string name;
        string id;
        string gender;
        string phone_number;
        string location;
        string user_type;
        while ((record = account_read.ReadLine()) != null)
        {
            name = PARSE_RECORD(record, 0);
            //PARSE_FUNCTION(ref record,accountant_name,father_name,cnic,source_income,receiver)
            id = PARSE_RECORD(record, 1);
            phone_number = PARSE_RECORD(record, 2);
            location = PARSE_RECORD(record, 3);
            gender = PARSE_RECORD(record, 4);
            user_type = PARSE_RECORD(record, 5);
            user_bl sender = new user_bl(name, id, phone_number, location, gender, user_type);
            name = PARSE_RECORD(record, 6);
            //PARSE_FUNCTION(ref record,accountant_name,father_name,cnic,source_income,receiver)
            id = PARSE_RECORD(record, 7);
            phone_number = PARSE_RECORD(record, 8);
            location = PARSE_RECORD(record, 9);
            gender = PARSE_RECORD(record, 10);
            user_type = PARSE_RECORD(record, 11);
            user_bl receiver = new user_bl(name, id, phone_number, location, gender, user_type);

            if (!user_dl.send_request_Dict.ContainsKey(receiver))

```

```

        {
            user_dl.send_request_Dict.Add(receiver, new List<user_bl>());

        }
        if (!user_dl.send_request_Dict.ContainsKey(sender))
        {
            user_dl.send_request_Dict.Add(sender, new List<user_bl>());
        }
        user_dl.send_request_Dict[receiver].Add(sender);
    }
}

public static void main_graph_load_file()
{
    StreamReader account_read = new StreamReader("main_graph.txt");

    string record;
    string name;
    string id;
    string gender;
    string phone_number;
    string location;
    string user_type;
    while ((record = account_read.ReadLine()) != null)
    {
        name = PARSE_RECORD(record, 0);

        id = PARSE_RECORD(record, 1);
        phone_number = PARSE_RECORD(record, 2);
        location = PARSE_RECORD(record, 3);
        gender = PARSE_RECORD(record, 4);
        user_type = PARSE_RECORD(record, 5);
        user_bl sender = new user_bl(name, id, phone_number, location, gender, user_type);
        name = PARSE_RECORD(record, 6);

        id = PARSE_RECORD(record, 7);
        phone_number = PARSE_RECORD(record, 8);
        location = PARSE_RECORD(record, 9);
        gender = PARSE_RECORD(record, 10);
        user_type = PARSE_RECORD(record, 11);
        user_bl receiver = new user_bl(name, id, phone_number, location, gender, user_type);

        if (!main_dict.ContainsKey(sender))

```

```

        {
            main_dict.Add(sender, new List<user_bl>());
        }
        if (!main_dict.ContainsKey(receiver))
        {
            main_dict.Add(receiver, new List<user_bl>());
        }
        counter = receiver.Name;
        counter1 = sender.Name;
        main_dict[receiver].Add(sender);
        main_dict[sender].Add(receiver);
    }
}
public static void s_r_friend_request_delete(user_bl sender, user_bl receiver)
{
    foreach (KeyValuePair<user_bl, List<user_bl>> friend in send_request_Dict)
    {
        if (friend.Key.Id == sender.Id)
        {
            for (int i = 0; i < friend.Value.Count; i++)
            {
                if (friend.Value[i].Id == receiver.Id)
                {
                    friend.Value.RemoveAt(i);
                }
            }
        }
    }
    foreach (KeyValuePair<user_bl, List<user_bl>> friend in send_request_Dict)
    {
        if (friend.Key.Id == receiver.Id)
        {
            for (int i = 0; i < friend.Value.Count; i++)
            {
                if (friend.Value[i].Id == sender.Id)
                {
                    friend.Value.RemoveAt(i);
                }
            }
        }
    }
}
public static void Save_Data(string path, user_bl n, user_bl j)
{
    StreamWriter file = new StreamWriter(path, true);

```

```

        file.WriteLine(j + "," + n.Name + "," + n.Id);
        file.Flush();
        file.Close();
    }
    public static void WriteFile(user_bl user, string path)
    {
        StreamWriter file = new StreamWriter(path, true);
        file.WriteLine(user.Name + "," + user.Id + "," + user.Phone_number + "," + user);
        file.Flush();
        file.Close();
    }
    public static void send_request_write_data(user_bl user, string path)
    {
        StreamWriter file = new StreamWriter(path, true);
        file.WriteLine(user.Name + "," + user.Id + "," + user.Phone_number + "," + user);
        file.Flush();
        file.Close();
    }
    public static void send_request_Load_data()
    {
        StreamReader a = new StreamReader("send_request_file.txt");
        int count = 0;
        string record;
        string name;
        string id;
        string gender;
        string phone_number;
        string location;
        string user_type;
        user_bl key = null;
        while ((record = a.ReadLine()) != null)
        {
            name = PARSE_RECORD(record, 0);

            id = PARSE_RECORD(record, 1);
            phone_number = PARSE_RECORD(record, 2);
            location = PARSE_RECORD(record, 3);
            gender = PARSE_RECORD(record, 4);
            user_type = PARSE_RECORD(record, 5);

            user_bl A = new user_bl(name, id, phone_number, location, gender, user_type);
            if (count == 0)
            {
                key = A;
                count++;
            }
        }
    }

```

```

        else
        {
            send_request_Dict[key].Add(A);
        }
    }

    a.Close();
}

static string PARSE_RECORD(string record, int field)
{
    int comma = 0;
    string item = "";
    for (int a = 0; a < record.Length; a++)
    {
        if (record[a] == ',')
        {
            comma++;
        }
        else if (comma == field)
        {
            item = item + record[a];
        }
    }
    return item;
}

public static user_bl check_user(string user_name, string password, string user_type)
{
    for (int x = 0; x < users_list.Count; x++)
    {
        if (user_name == users_list[x].Name && password == users_list[x].Id && user_type == users_list[x].Type)
        {
            return users_list[x];
        }
    }
    return null;
}

public static void load_users()
{
    users_list.Clear();
    StreamReader account_read = new StreamReader(Musers_path);

    string record;
    string name;
    string id;

```

```
        string gender;
        string phone_number;
        string location;
        string user_type;
        while ((record = account_read.ReadLine()) != null)
        {
            name = PARSE_RECORD(record, 0);
            //PARSE_FUNCTION(ref record,accountant_name,father_name,cnic,source_income,
            id = PARSE_RECORD(record, 1);
            phone_number = PARSE_RECORD(record, 2);
            location = PARSE_RECORD(record, 3);
            gender = PARSE_RECORD(record, 4);
            user_type = PARSE_RECORD(record, 5);

            user_bl A = new user_bl(name, id, phone_number, location, gender, user_type);
            users_list.Add(A);

        }

        account_read.Close();
    }
    public static void store_Messages_IN_file(string path,string mess)
    {
        StreamWriter file = new StreamWriter(path, true);
        file.WriteLine(mess);
        file.Flush();
        file.Close();
    }
}
```