

# Proyecto de Programación Avanzada

Carlos Andrés Cuéllar Velásquez 2497117

Walter Alexander Osoy Veliz 1126017



# Class Form1

- Se inicializa el botón de Log in, en donde se capturan los datos por medio de un boolean.

```
private void button2_Click(object sender, EventArgs e)
{
    string user = textBox1.Text;
    string password = textBox2.Text;
    bool permitir = Goathemala.find(user, password);
    if (permitir == true)
    {
        this.Hide();
        Principal obj = new Principal();
        obj.Show();
    }
    else
    {
        textBox3.Text = "datos invalidos";
    }
}
```

- Se crea el boton de Sign up para registrarse.

```
private void button1_Click(object sender, EventArgs e)
{
    this.Hide();
    Singup sign = new Singup();
    sign.Show();
}
```

# Class Goathemala

- Se crean los atributos de la clase.

```
private static Usuario[] congreso = new Usuario[100];  
private static int cont = -1;
```

- Se crea la función para agregar un nuevo usuario.

```
public static void addPolitico(Usuario politicos)  
{  
    cont++;  
    congreso[cont] = politicos;  
}
```

- Se crea la función para iniciar sesión si ya se tiene usuario.

```
public static bool find(string name, string password)  
{  
    bool permitido = false;  
    for (int i = 0; i < cont; i++)  
    {  
        if (name == congreso[i].getUser())  
        {  
            if (password.Equals(congreso[i].getPassword()))  
            {  
                permitido = true;  
                return permitido;  
            }  
        }  
    }  
    return permitido;  
}
```



# Class Sign up

- Boton de salida del sign up.

```
private void button2_Click(object sender, EventArgs e)
{
    this.Close();
    Form1 obj = new Form1();
    obj.Show();
}
```

- Se crea el botón de aceptar para el registro del nuevo usuario, si falta información sale una alerta de llenar todos los datos y si todo se encuentra bien manda a llamar al arreglo para almacenar el nuevo usuario.

```
private void button1_Click(object sender, EventArgs e)
{
    //1 diputado, 2 asesor
    int job = 0;
    int allright = 0;
    if (radioButton1.Checked == true)
    {
        job = 1;
        allright++;
    }
    else
    {
        if (radioButton2.Checked == true)
        {
            job = 2;
            allright++;
        }
        else
        {
            textBox6.Text = "Debe de llenar todos los campos";
        }
    }
    if (textBox4.Text.Equals(textBox5.Text))
    {
        allright++;
    }
}
```

```

    if (allright == 2)
    {
        allright = 0;
        obj = new Usuario (textBox1.Text, textBox2.Text, textBox3.Text, job, textBox4.Text);
        Goathemala.addPolitico(obj);
        Principal main = new Principal();
        main.Show();
        this.Hide();
    }
    else
    {
        textBox6.Text = "revise que todos los datos esten bien";
    }
}

private void Singup_Load(object sender, EventArgs e)
{
}
}
```

# Class Usuario

- Se crean los atributos de la clase usuario.

```
class Usuario
{
    //atributos
    private string name;
    private string lastname;
    private string user;
    public int job;//1 diputado, 2 asesor
    private string password;
```

- Se crea el constructor de la clase Usuario.

```
public Usuario(string name, string lastname, string user, int job, string password)
{
    this.name = name;
    this.lastname = lastname;
    this.user = user;
    this.job = job;
    this.password = password;
}
```

- Se escriben los get y set de los atributos de class.

```
public void setName(string name)
{
    this.name = name;
}
public string getName()
{
    return name;
}
public void setLastname(string last)
{
    this.lastname = last;
}
public string getLastname()
{
    return this.lastname;
}
public void setUser(string user)
{
    this.user = user;
}
public string getUser()
{
    return this.user;
}
public void setJob(int job)
{
    this.job = job;
}
public int getJob()
{
    return job;
}
public void setPassword(string password)
{
    this.password = password;
}
```

# Class Leyes

- Captura los datos de la nueva ley y los manda a la clase ley.

```
//captura datos para la nueva ley t los manda a la clase Ley
private void button1_Click(object sender, EventArgs e)
{
    leys[numleyes] = new Ley();
    leys[numleyes].addLey(textBox1.Text, textBox2.Text);
    numleyes++;
    textBox1.Clear();
    textBox2.Clear();
    MessageBox.Show("Ley creada con exito");
}
```

- Botón de mostrar las leyes; graba las leyes y sus descripción en una listbox.

```
private void button7_Click(object sender, EventArgs e)
{
    listBox1.Items.Clear();
    showLeyes();
    tabControl1.SelectTab(5);
}
//boton de mostrar leyes
//graba las leyes y su descripcion a un listBox
public void showLeyes()
{
    int cont = 1;
    string cadena = "";
    for (int i = 0; i < numleyes; i++)
    {
        listBox1.Items.Add("Ley No. " + cont);
        listBox1.Items.Add("Titulo: " + leys[cont - 1].getTitulo());
        listBox1.Items.Add("Descripción: " + leys[cont - 1].getInfo());
        cont++;
    }
}
```



- Agrega a un combo las leyes existentes y botón de modificar.

```
//agrega a un combo box los leyes existentes
public void addPosiblesCambios()
{
    for (int i = 0; i < numleyes; i++)
    {
        comboBox1.Items.Add(leys[i].getTitulo());
    }
}

//boton de modificar
private void button4_Click(object sender, EventArgs e)
{
    int op = comboBox1.SelectedIndex;
    if (textBox3.Text != "") {
        leys[op].setTitulo(textBox3.Text);
    }
    if (textBox4.Text != "") {
        leys[op].setInfo(textBox4.Text);
    }
    textBox3.Clear();
    textBox4.Clear();
    MessageBox.Show("Cambios realizados con exito");
}
```

- Menú de las posibles opciones en el combo box.

```
//agrega a un combo box los leyes existentes
public void addPosiblesCambios()
{
    for (int i = 0; i < numleyes; i++)
    {
        comboBox1.Items.Add(leys[i].getTitulo());
    }
}

//boton de modificar
private void button4_Click(object sender, EventArgs e)
{
    int op = comboBox1.SelectedIndex;
    if (textBox3.Text != "") {
        leys[op].setTitulo(textBox3.Text);
    }
    if (textBox4.Text != "") {
        leys[op].setInfo(textBox4.Text);
    }
    textBox3.Clear();
    textBox4.Clear();
    MessageBox.Show("Cambios realizados con exito");
}
```

- Se agrega a un combo box las leyes que se desean eliminar y abajo en el código se elimina la ley.

```
//agrega a un combo box los leyes para ver cual se decea eliminar
public void addPosibleseliminados()
{
    for (int i = 0; i < numleyes; i++)
    {
        comboBox4.Items.Add(leys[i].getTitulo());
    }
}

private void button9_Click(object sender, EventArgs e)
{
    tabControl1.SelectTab(6);
}

//Elimina una ley
private void button10_Click(object sender, EventArgs e)
{
    int op = comboBox1.SelectedIndex;
    leys[op].setTitulo(null);
    leys[op].setInfo(null);
    textBox3.Clear();
    textBox4.Clear();
    MessageBox.Show("ley eliminada con exito");
    comboBox4.SelectedIndex = -1;
    numleyes--;
}
```