

**Instituto Tecnológico Nacional de
México**

**Instituto Tecnológico del Sur de
Nayarit**

27-3-2020

**Programación Orientada a Objetos
Tema IV “Herencia y Polimorfismo”**



Alumno: Meza Rosales Luis Antonio Ismael

No. De control:191140008

Docente: Cinthia Anahí Mata Bravo

Segundo semestre

**Documentación JuegoUnidad4. “Piedra,
Papel y Tijeras”**

Índice

Descripción del Juego.....	3
Solución Propuesta.....	4
Solución Propuesta.....	4
Instrucciones del juego	7
Diagrama de Interfaces	13

Descripción del Juego

hoy en día existen una gran cantidad de cosas de la vida diaria que tienen un algoritmo que nos ayuda a poder desarrollarlas o simplemente cosas que tiene una delimitación para para poder realizarlas o no.

Se pretende crear el juego piedra papel y tijeras.

Se pretende que realice lo que normalmente se hace cuando se juega dicho juego, pero en este caso se pretende que la persona juegue contra la computadora, la manera de ganar será fácil de identificar:

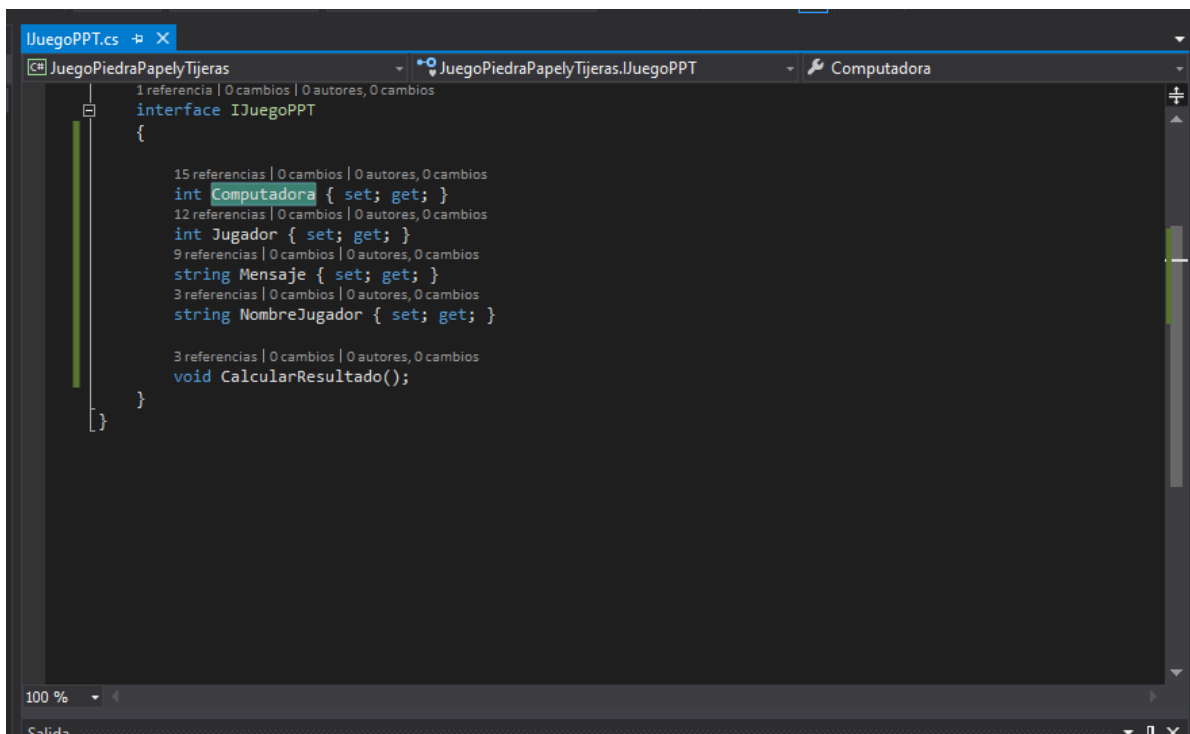
- Si alguna de los dos involucrados, es decir, el jugador y elige como opción piedra y el otro elige tijera, automáticamente gana el que haya elegido piedra
- Si alguna de los dos involucrados, es decir, el jugador y elige como opción papel y el otro elige piedra, automáticamente gana el que haya elegido papel
- Si alguna de los dos involucrados, es decir, el jugador y elige como opción tijera y el otro elige papel, automáticamente gana el que haya elegido tijera

Solución Propuesta

Solución Propuesta

1. La solución para el programa del juego Piedra, papel y tijeras se utilizó la interface IJuegoPPT

Donde crearon las propiedades de tipo int que son Computadora y Jugador, también se crearon propiedades tipo string que son Mensaje y NombreDelJugador, además se creó el método CalcularResultado, para posteriormente se utilizarlos en las clases que hereden dicha interface



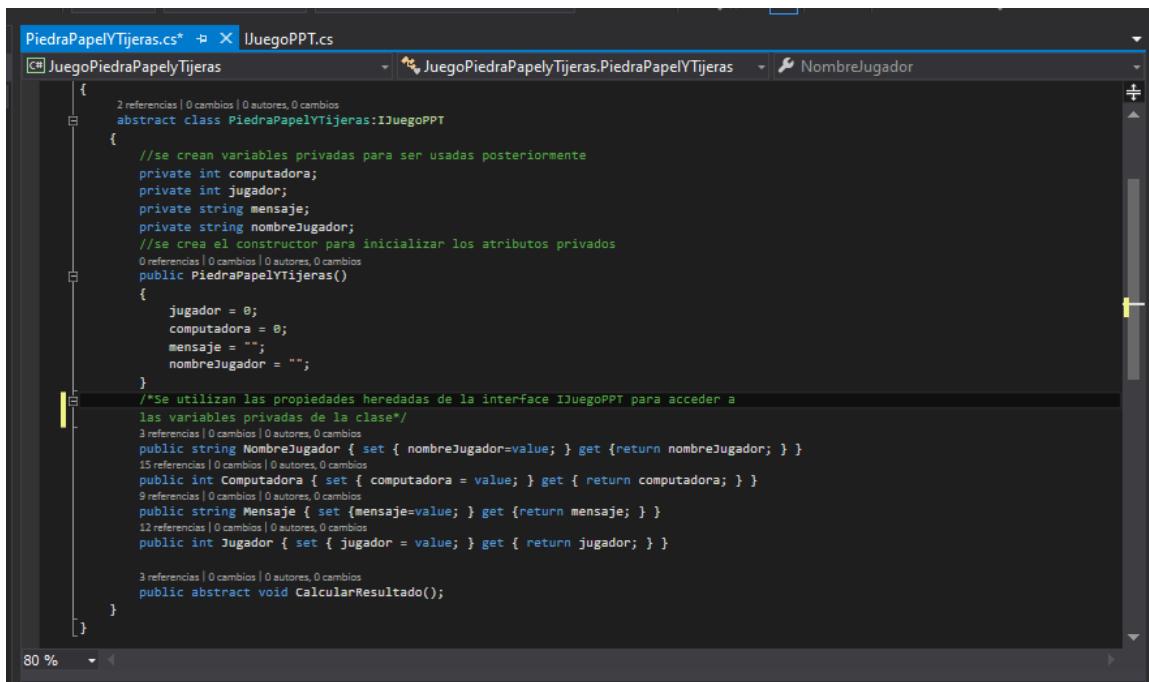
```
JuegoPPT.cs
JuegoPiedraPapelyTijeras
    JuegoPiedraPapelyTijeras.IJuegoPPT
    Computadora

1 referencia | 0 cambios | 0 autores, 0 cambios
interface IJuegoPPT
{
    15 referencias | 0 cambios | 0 autores, 0 cambios
    int Computadora { set; get; }
    12 referencias | 0 cambios | 0 autores, 0 cambios
    int Jugador { set; get; }
    9 referencias | 0 cambios | 0 autores, 0 cambios
    string Mensaje { set; get; }
    3 referencias | 0 cambios | 0 autores, 0 cambios
    string NombreJugador { set; get; }

    3 referencias | 0 cambios | 0 autores, 0 cambios
    void CalcularResultado();
}
}
```

2. Posteriormente se creó la clase Abstracta PiedraPapelYTijeras que hereda las propiedades y métodos de la interface IJuegoPPT, además la clase utiliza las propiedades heredadas para acceder a los atributos privados creados en dicha clase que son las variables de tipo int que son computadora y jugador, también se crearon variables tipo string que son mensaje y nombreDelJugador

Así mismo se convirtió al método CalcularResultado a abstracto para posteriormente poder utilizarlo en las clases hijas.



```
PiedraPapelYTijeras.cs* - X IJuegoPPT.cs
JuegoPiedraPapelyTijeras  JuegoPiedraPapelyTijeras.PiedraPapelYTijeras  NombreJugador

{
    2 referencias | 0 cambios | 0 autores, 0 cambios
    abstract class PiedraPapelYTijeras:IJuegoPPT
    {
        //se crean variables privadas para ser usadas posteriormente
        private int computadora;
        private int jugador;
        private string mensaje;
        private string nombreJugador;
        //se crea el constructor para inicializar los atributos privados
        0 referencias | 0 cambios | 0 autores, 0 cambios
        public PiedraPapelYTijeras()
        {
            jugador = 0;
            computadora = 0;
            mensaje = "";
            nombreJugador = "";
        }

        /*Se utilizan las propiedades heredadas de la interface IJuegoPPT para acceder a
        las variables privadas de la clase*/
        3 referencias | 0 cambios | 0 autores, 0 cambios
        public string NombreJugador { set { nombreJugador=value; } get {return nombreJugador; } }
        15 referencias | 0 cambios | 0 autores, 0 cambios
        public int Computadora { set { computadora = value; } get { return computadora; } }
        9 referencias | 0 cambios | 0 autores, 0 cambios
        public string Mensaje { set {mensaje=value; } get {return mensaje; } }
        12 referencias | 0 cambios | 0 autores, 0 cambios
        public int Jugador { set { jugador = value; } get { return jugador; } }

        3 referencias | 0 cambios | 0 autores, 0 cambios
        public abstract void CalcularResultado();
    }
}
```

3. Se crea una clase que heredara todos los métodos y atributos de PiedraPapelyTijeras y use el método y las propiedades para realizar las operaciones necesarias para evaluar el resultado y saber quien gana

```

namespace JuegoPiedraPapelyTijeras
{
    5 referencias | 0 cambios | 0 autores, 0 cambios
    class JuegoPPT:PiedraPapelyTijeras
    {
        //utiliza el metodo Calcular resultado para decidir que mensaje enviar al usuario
        3 referencias | 0 cambios | 0 autores, 0 cambios
        public override void CalcularResultado()
        {
            //Evalua todas las formas de ganar
            if (Jugador > Computadora & Computadora == 2)
            {
                Mensaje = "¡FELICIDADES USTED GANO! :)";
            }
            else if ((Jugador == 2) & (Jugador > Computadora))
            {
                Mensaje = "¡FELICIDADES USTED GANO! :)";
            }
            else if (Computadora == 3 & Jugador == 1)
            {
                Mensaje = "¡FELICIDADES USTED GANO! :)";
            }
            //Evalua todas las formas de perder
            else if ((Computadora == 2) & (Computadora > Jugador))
            {
                Mensaje = "LO SENTIMOS USTED PERDIO :C";
            }
            else if (Jugador < Computadora & Computadora == 2)
            {
                Mensaje = "LO SENTIMOS USTED PERDIO :C";
            }
        }
    }
}

```

```

        else if (Computadora == 3 & Jugador == 1)
        {
            Mensaje = "¡FELICIDADES USTED GANO! :)";
        }
        //Evalua todas las formas de perder
        else if ((Computadora == 2) & (Computadora > Jugador))
        {
            Mensaje = "LO SENTIMOS USTED PERDIO :C";
        }
        else if (Jugador < Computadora & Computadora == 2)
        {
            Mensaje = "LO SENTIMOS USTED PERDIO :C";
        }

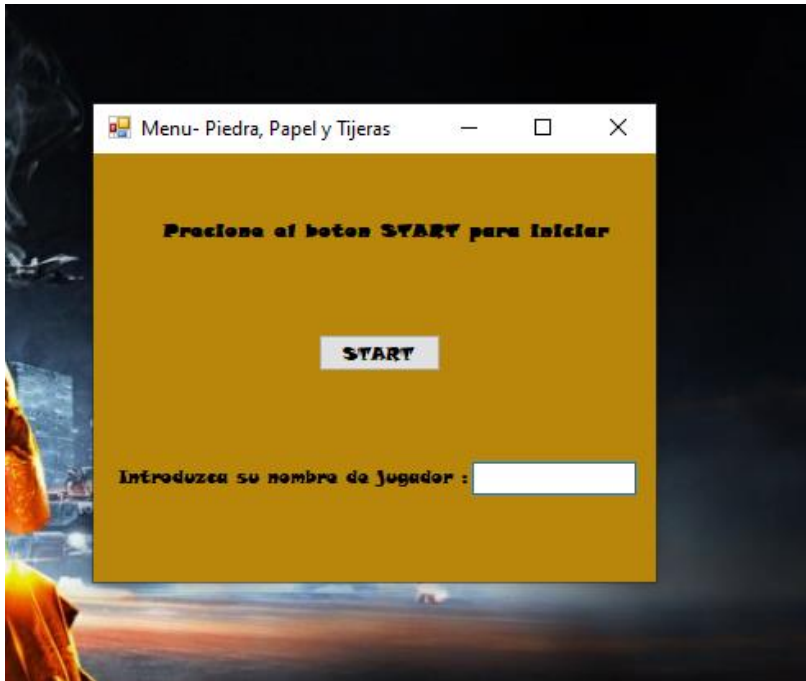
        else if (Computadora == 1 & Jugador == 3)
        {
            Mensaje = "LO SENTIMOS USTED PERDIO :C";
        }
        //Evalua cuanto es un empate
        else if (Jugador == Computadora)
        {
            Mensaje = "¡VALLA FUE UN EMPATE! :)";
        }
    }
}

```

Instrucciones del juego

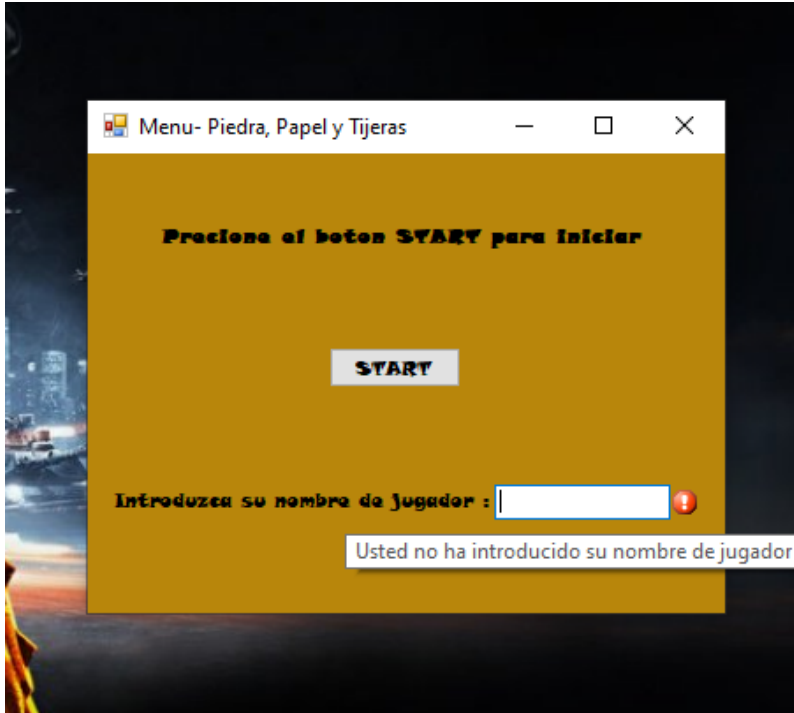
1. Al iniciar el juego aparece el menú donde da instrucciones de debe de realizar el usuario

- Indica que debes presionar el botón "START" para iniciar el juego
- Indica que debes de colocar tu nombre de jugador

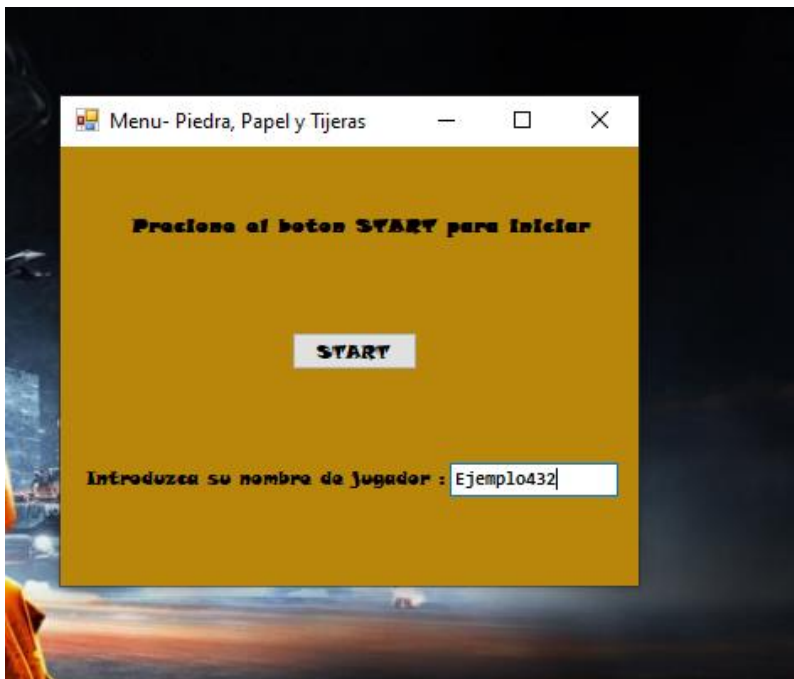


2. Si no coloca el nombre del jugador no podrá acceder a la pantalla del juego

- Aparece un errorProvider que muestra el mensaje que debe introducir el nombre y el icono de advertencia con un fondo rojo

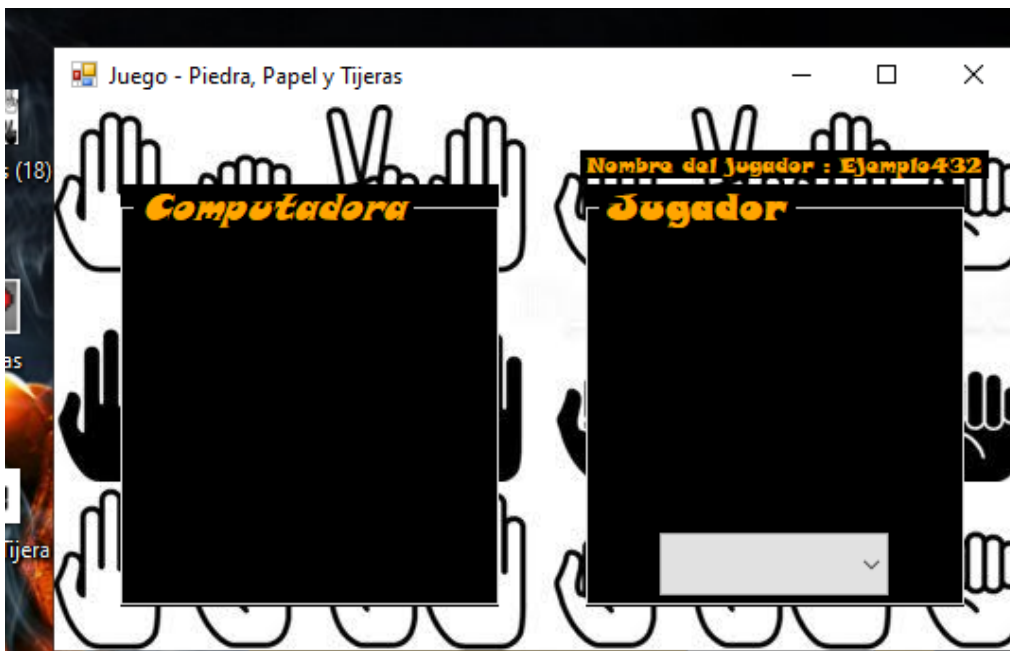


3. ya introducido el nombre del jugador presiona el botón "START" para iniciar el juego



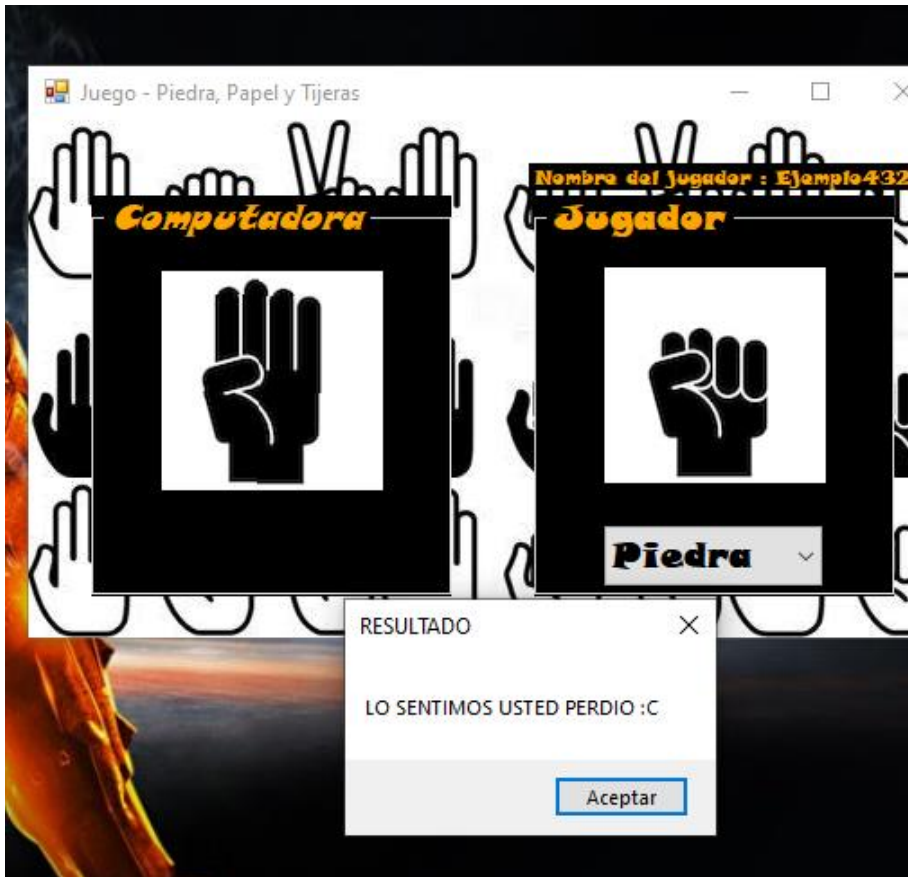
4. Al iniciar la pantalla del juego se coloca el nombre que introduciste con anterioridad en un label

Se muestran dos groupBox el primero dice “Computadora” e indica que ahí se visualizaran las opciones elegidas por la computadora y el segundo groupBox que dice “Jugador” que indica que ahí se visualizaran las opciones elegidas por el usuario, además cuenta con un comboBox donde el usuario colocara las opciones que el elija



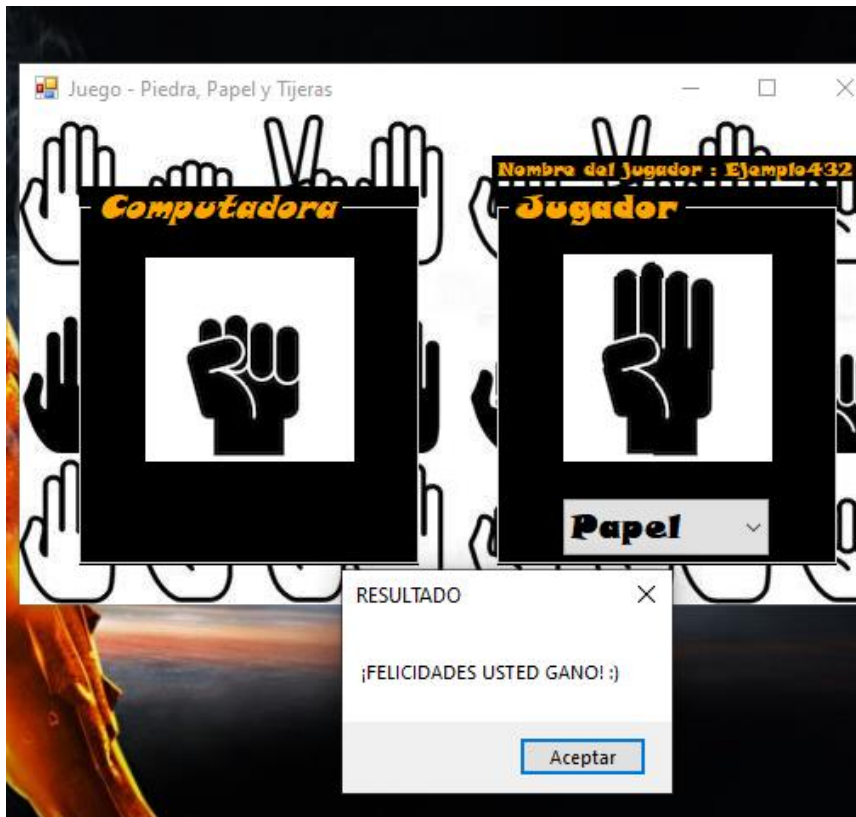
5. si el Jugador elige una opción y la computadora elige la opción que le gana a la del usuario se mostrara un mensaje que indicara el resultado y un texto que dice:

“LO SENTIMOS USTED PERDIO :C”



6. si la computadora elige una opción y el Jugador elige la opción que le gana a la de la computadora se mostrara un mensaje que indicara el resultado y un texto que dice:

“¡FELICIDADES USTED GANO! 😊”



7. si la computadora elige una opción y el Jugador elige la misma opción que la de la computadora se mostrara un mensaje que indicara el resultado y un texto que dice:

“¡VALLA FUE UN EMPATE! 😊”

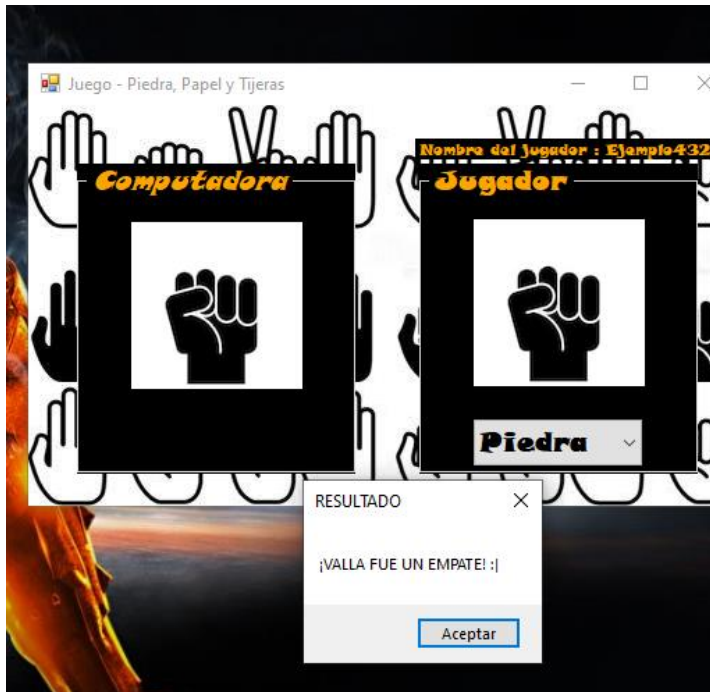


Diagrama de Interfaces

