



La Liga Torneo

PRO CS:GO

Liga online de Counter-Strike Global Offensive

Luis Andaur - 2E

Trabajo práctico N°3 (Primera entrega)

----- Temas vistos entre la clase 10 y 15 -----

Laboratorio II



Descripción del proyecto

Breve resumen

El proyecto simula un torneo online de Counter-Strike. La metodología del torneo es una competencia individual , todos contra todos, en 9 (nueve) mapas con modalidades diferentes, dónde cada jugador tiene la posibilidad de sumar puntos de la siguiente manera:

- Kills (*matar a otros jugadores*) x1 punto por kill
- Bombas (*bombas plantadas*) x3 puntos por bomba
- Rehenes (*rescate de rehenes*) x3 puntos por rehen
- Headshot (*disparo a la cabeza*) x2 puntos por headshot

Pero se les resta por cada vez que mueren o los matan

- Muertes -x2 puntos por muerte.

El proyecto ya toma datos de jugadores registrados desde un archivo json, pero tambien cuenta con la posibilidad de agregar, editar y eliminar el jugador que desee.

Se simula el torneo y luego puede descargar en el formato que deseé los reportes que se obtuvieron del mismo.

Home

Menu principal del sistema



Administrar Jugadores

Se visualiza a todos los jugadores, y también se agrega, edita o elimina a los mismos.

Simular Torneo

Simula el torneo completando las estadísticas de los jugadores con números random y sumando los puntos obtenidos.

Reportes

Reportes del torneo que se pueden guardar.

Sair

Sale y cierra la aplicación.

Administrar jugadores

AMB de la aplicación

Administrador de jugadores						
Nro. Jugador	Nombre	Edad	Genero	Nacionalidad	Especialidad	Primer Torneo
4567	Shir	36	Female	China	M	<input checked="" type="checkbox"/>
6354	Nils	38	Female	Brazil	S	<input checked="" type="checkbox"/>
594	Timmie	28	Female	Japan	XL	<input type="checkbox"/>
1937	Kare	20	Male	Czech Republic	L	<input checked="" type="checkbox"/>
726	Antone	47	Female	Brazil	XS	<input type="checkbox"/>
2456	Kevina	12	Male	Japan	XXL	<input checked="" type="checkbox"/>
4322	Eustacia	52	Male	Peru	XL	<input type="checkbox"/>
3345	Hillary	43	Male	Colombia	M	<input type="checkbox"/>
1111	Gordan	14	Female	China	S	<input checked="" type="checkbox"/>
2154	Kamilah	21	Female	France	XS	<input checked="" type="checkbox"/>
5847	Kristo	15	Female	Brazil	XXL	<input checked="" type="checkbox"/>
3847	Sancho	29	Female	China	L	<input type="checkbox"/>
3722	Nerissa	54	Male	Russia	L	<input checked="" type="checkbox"/>
736	Hetty	25	Female	Czech Republic	XXL	<input checked="" type="checkbox"/>
343	Ebba	17	Male	Panama	XXL	<input type="checkbox"/>
4234	Nonie	58	Female	Poland	XL	<input checked="" type="checkbox"/>
5522	Kessia	28	Male	Russia	XS	<input type="checkbox"/>
2091	Julietta	37	Male	United States	S	<input checked="" type="checkbox"/>
1112	Kinny	53	Male	China	S	<input type="checkbox"/>

Lista de todos los jugadores con sus atributos como nombre, edad, genero y nacionalidad y otros detallados a continuación:

Nro. Jugador: Nro único por jugador.

Especialidad: Especifica la modalidad donde el jugador es más hábil

"XS" = Cuerpo a cuerpo.

"S" = Corta distancia.

"M" = Media distancia.

"L" = Larga distancia.

"XL" = ExtraL.

"XXL" = Francotirador.

Primer Torneo: Se marca cuando es el primer torneo que juega.

Agregar

Abre el menu para completar los datos con un nuevo jugador

Editar

Selecciona un jugador y abre un menu con sus datos para que pueda modificarlos, menos el Nro de jugador.

Eliminar

Para eliminar sólo debe seleccionar un jugador y al presionar eliminar este será eliminado

Volver

Vuelve al menu principal/Home.



Simular Torneo

Simulador de todo el torneo

Simular Torneo



Simula la partida haciendo que los jugadores jueguen en los 9 mapas, asignandoles de manera random las estadísticas para kills, muertes, bombas, rehenes, headshots y calculando la cantidad de puntos y sumandolos a cada jugador.

Reportes

Análisis de datos del torneo

Reportes

Reportes del torneo

Campeon del torneo

Estadisticas de Jugadores

Los más Killers

Los más mancos

Primer Torneo

Bombardeiros

Rescatistas

XXL Headshots

El menor

Volver



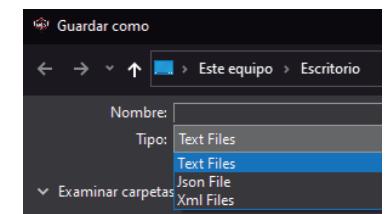
TorneoPRO : CSGO

EASport

U\$D1000000



Reportes resultantes del torneo que se pueden guardar en la extensión que deseé el usuario y también puede elegir la ubicación donde desee guardar sus reportes generados.





Temas utilizados

----- *entre clase 10 y 15* -----

Excepciones
Pruebas Unitarias
Tipos Genéricos
Interfaces
Archivos
Serialización

Excepciones

Ejemplo de implementación

Implementado en varias partes del sistema para controlar posibles incidentes y evitar que el programa deje de funcionar, por ejemplo una de las implementadas es:

```
8 referencias
public class Exception_NroJugadorYaExiste : Exception
{
    /// <summary>
    /// Crea una excepcion con un mensaje
    /// </summary>
    /// <param name="message">Mensaje de la excepcion</param>
    1 referencia
    public Exception_NroJugadorYaExiste(string message) : this(message, null)
    {

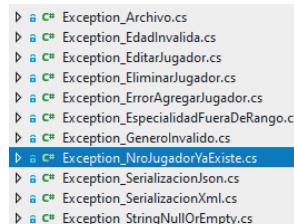
    }

    /// <summary>
    /// Crea una excepcion con un mensaje y un innerException
    /// </summary>
    /// <param name="message">Mensaje de la excepcion</param>
    /// <param name="innerException">innerException de la excepcion</param>
    1 referencia
    public Exception_NroJugadorYaExiste(string message, Exception innerException) : base(message, innerException)
    {
    }
}
```

Dicha excepcion se lanzará cuando el nro. de jugador ingresado coincida con uno existente. Por ejemplo se implementa en el metodo AgregarJugador()

```
1 referencia
public static void AgregarJugador(string nroJugador, string nombre, string edad, string ge
{
    if (!string.IsNullOrEmpty(nroJugador) && !string.IsNullOrEmpty(nombre) && !string.IsNullOrEmpty(edad))
    {
        int.TryParse(nroJugador, out int auxNroJugador);
        if (ExisteJugador(auxNroJugador))
        {
            throw new Exception_NroJugadorYaExiste("Error el nro de jugador ya existe");
        }
        int.TryParse(edad, out int auxEdad);

        listaJugadores.Add(new Jugador(auxNroJugador, nombre, auxEdad, genero, nacionalida
    }
}
```

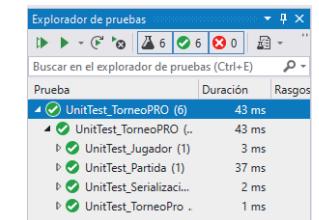
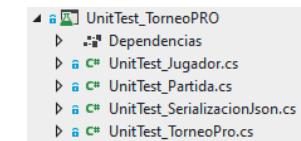


Pruebas unitarias

Ejemplo de implementación

En el proyecto 'UnitTest_TorneoPRO' hay test unitarios, donde se testean métodos que hagan lo correspondiente.

Se puede observar que el 100% de los test pasan correctamente las pruebas.



Un ejemplo de implementación es en la deserialización de un archivo json, dónde si no encuentra el archivo enviado por parámetro, lanza una excepcion del tipo Exception_SerializacionJson.

```
[TestClass]
0 referencias
public class UnitTest_SerializacionJson
{
    [TestMethod]
    [ExpectedException(typeof(Exception_SerializacionJson))]
    0 referencias
    public void TestLeer_RecibeNombreArchivo_ReturnaExceptionSerializacionJson()
    {
        string archivo = "noExiste.json";
        SerializacionJson<List<Jugador>>.Leer(archivo);
    }
}
```

Tipos genéricos

Ejemplo de implementación

Implementados en archivos y serialización por un tema de practicidad.



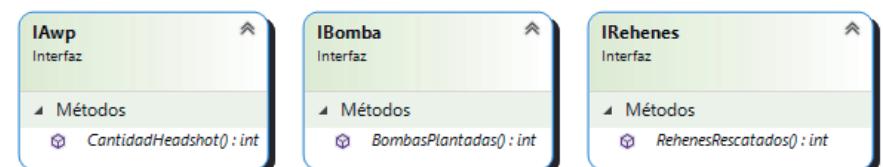
Se puede apreciar su uso en el método GuardarComo():

```
2 referencias
private void GuardarComo(Jugador tipo)
{
    this.saveFile.ShowDialog();
    if (this.saveFile.FileName != String.Empty)
    {
        if (Path.GetExtension(this.saveFile.FileName) == ".txt")
        {
            try
            {
                Archivo<Jugador>.Escribir(tipo, this.saveFile.FileName);
            }
        }
        -----
        try
        {
            SerializacionJson<Jugador>.Escribir(tipo, this.saveFile.FileName);
        }
        -----
        try
        {
            SerializacionXml<Jugador>.Escribir(tipo, this.saveFile.FileName);
        }
    }
}
```

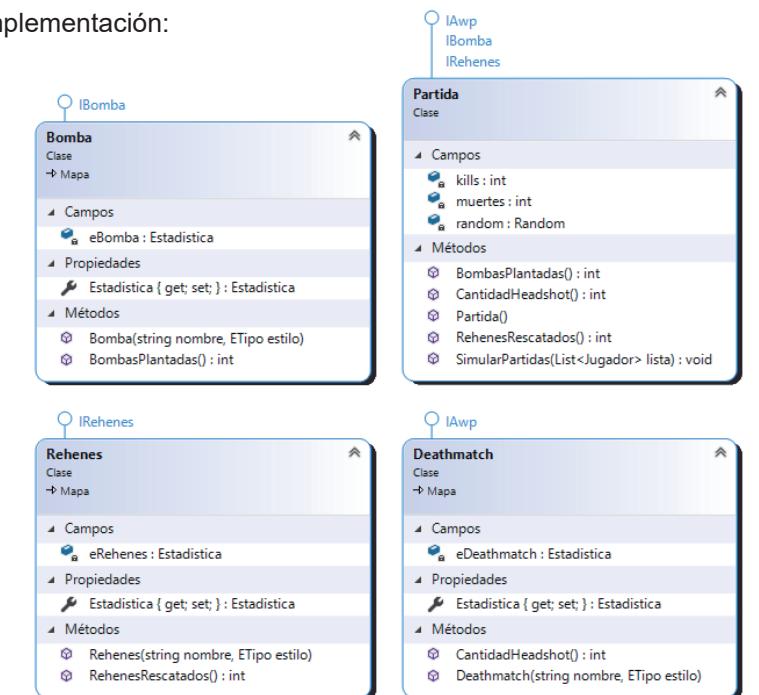
Interfaces

Ejemplo de implementación

Se crearon 3 interfaces con métodos que permiten implementarlos en clases heredadas, por ejemplo Bomba, Deathshot y Rehenes que heredan de la clase Mapa, y otra clase que nada tiene que ver con estas como lo es Partida, pudiendo darles funcionalidades particulares dependiendo la necesidad.



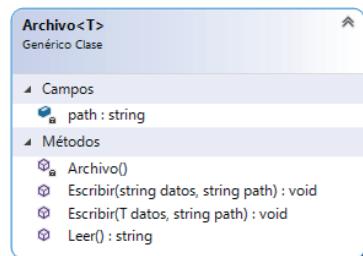
Implementación:



Archivos

Ejemplo de implementación

Se utilizan para escribir en txt los reportes generados en el torneo.



Se puede apreciar su uso en el método GuardarComo():

```

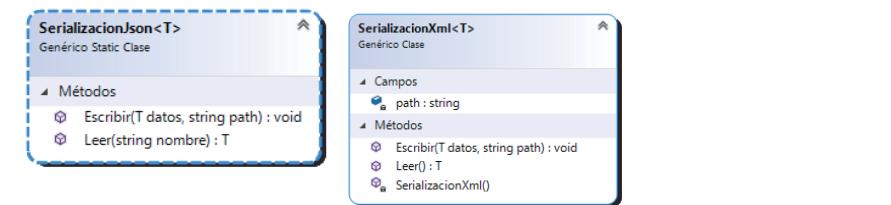
2 referencias
private void GuardarComo(Jugador tipo)
{
    this.saveFile.ShowDialog();
    if (this.saveFile.FileName != String.Empty)
    {
        if (Path.GetExtension(this.saveFile.FileName) == ".txt")
        {
            try
            {
                Archivo<Jugador>.Escribir(tipo, this.saveFile.FileName);
            }
        }
    }
}

```

Serialización

Ejemplo de implementación

Se utilizan para serializar y deserializar archivos y listas de los reportes generados en el torneo.



SerializacionXml se puede apreciar su uso en el método GuardarComo():

```

try
{
    SerializacionXml<Jugador>.Escribir(tipo, this.sendFile.FileName);
}

```

SerializacionJson se puede apreciar su uso deserializando la lista inicial con los jugadores del torneo en el método static CargarJugadores() de la clase static TorneoPro.

```

private static void CargarJugadores()
{
    listaJugadores = SerializacionJson<List<Jugador>>.Leer("listaJugadores.json");
    if (listaJugadores==null)
    {
        throw new Exception_SerializacionJson("No se deserializo la lista inicial");
    }
}

```