

## Trabajo Práctico numero 4

### Funcionamiento:

Este trabajo tiene la funcionalidad de analizar los datos según los jugadores de Valorant, sobre de que localidad son, que rango es cada uno, que edad tienen y cuál es el agente que eligen. Permitiendo ver en cantidad de jugadores que eligen a cada agente, los porcentajes de cada localidad, porcentajes de cada rango y edad promedio

### Temas utilizados:

#### Excepciones:

Fueron implementadas para controlar:

- La inexistencia de las rutas, de archivos, fallas en la lectura o escritura.
- También cuando la extensión no es correcta.
- Si la conexión con la base de datos ya está abierta.
- Y cuando se excede del tiempo intentando conectar con una base de datos inexistente.

#### Test Unitarios:

Los test ejecutados son:

- Validar que un jugador se instancie correctamente y que no sea nulo
- Que funcione correctamente la excepción de leer un archivo inexistente
- Que se lean correctamente archivos XML guardados en una ruta específica
- Conectar a la base de datos y pueda traer los datos correctamente

#### Generics:

Este tema fue implementado junto a las interfaces de archivos para que los métodos reciban un tipo genérico al escribir o lo devuelva al leer un archivo

#### Interfaces:

Las interfaces fueron generadas para guardar o leer archivos. Para guardar se recibirá una ruta y un dato del tipo genérico; y para leerlo se recibirá una ruta y retornara un tipo genérico

#### Archivos:

Para guardar y leer archivos se implementaran las interfaces, generando un método para guardar que escribirá un archivo a través de un StreamWriter, y también un método para leer que retornara el tipo genérico seteandolo como string, utilizando StreamReader

#### Serializacion:

Solo serializara y deserializara para XML, debido a que al deserializar en JSON había un error el cual no se pudo solucionar junto a los profesores, dejándolo descartado para este TP.

#### Base de datos:

La implementación fue generando una consulta haciendo un INSERT para agregar un jugador pasándole la edad, localidad, rango y agente elegido. También para obtener los jugadores que hay en la base de datos. Teniendo en cuenta que la base de datos exista, sino lanzara una excepción.

#### Hilos:

La descarga desde la base de datos se ejecuta en un hilo secundario por si es una descarga pesada, en este caso fue simulado con un sleep de 5 segundos. Pretendiendo que tarde y muestre correctamente el uso del hilo, también mediante eventos y delegados incrementa una barra marcando el progreso de la descarga

#### Eventos y Delegados:

La comunicación entre el formulario principal y el formulario de mostrar jugadores y análisis se ejecuta a través de eventos y delegados, ya que tiene que mostrar en tiempo real si hay una actualización en la lista de los jugadores

#### Métodos de Extensión:

Fue usado para obtener la cantidad de jugadores que hay en la lista y poder mostrarlo tanto en el formulario principal como en el de mostrar los archivos guardados