

## **Escape Room Virtual: La Sala de los Enigmas**

Nos han encargado desarrollar una aplicación para administrar un Escape Room virtual, donde los usuarios puedan disfrutar de emocionantes aventuras y resolver desafiantes enigmas en una sala temática llamada "La Sala de los Enigmas".

### **Requisitos:**

1. **Salas Temáticas y Niveles de Dificultad:** Cada sala tiene un identificador único, nombre, nivel de dificultad y un valor total (Euros) que engloba el conjunto de pistas y objetos de decoración que la conforman.
2. **Pistas Intrigantes:** Las pistas tienen un identificador único, precio, nombre, tiempo estimado que se espera que los jugadores dediquen a resolver la pista y finalmente una temática específica que permitirá guiar a los jugadores en la resolución de enigmas.
3. **Objetos de Decoración Únicos:** Los objetos de decoración contribuyen a la atmósfera inmersiva de cada sala. Tienen un identificador único, precio, descripción/nombre y tipo de material.
4. **Precio Asociado:** Cada elemento del inventario tiene un precio asociado, que incluye el cálculo del impuesto sobre el valor añadido (IVA). Las pistas tienen un 10% de IVA y los objetos de decoración un 21%.
5. **Gestión del Inventario:** Mantener un registro actualizado de todas las salas, pistas y objetos de decoración disponibles.

### **Funcionalidades Mínimas:**

1. Crear una nueva sala.
2. Añadir pistas a una sala específica.
3. Introducir objetos de decoración para ambientar las salas.
4. Mostrar el inventario actualizado, incluyendo cantidades disponibles y valor total.
5. Permitir la retirada de elementos del inventario.

### **Requisitos Adicionales:**

1. Uso de herencia / polimorfismo.
2. Tratamiento de excepciones para situaciones como por ejemplo la creación de salas duplicadas
3. Uso de lambdas y Java Stream Filter.
4. Empleo de variables estáticas y constantes.
5. Creación de enums para parametrizar valores.

6. Control de las entradas por teclado, gestionando excepciones según corresponda. Apoyarse en la Tasca S1.02. Exceptions - Nivel 2
7. Evitar el código spaghetti, siguiendo el principio de responsabilidad única para que las clases y métodos tengan una única función clara y concisa.