

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**  
**FACULTAD DE CIENCIAS E INGENIERÍA**

**LENGUAJE DE PROGRAMACIÓN 2**

**5ta. práctica (tipo b)**  
**(Segundo Semestre 2022)**

**Indicaciones Generales:**

- Tiempo estimado: 1h 50 minutos
- Se les recuerda que, de acuerdo al reglamento disciplinario de nuestra institución, constituye una falta grave copiar del trabajo realizado por otro estudiante o cometer plagio para el desarrollo de esta práctica.
- Debe hacer uso del entorno de desarrollo integrado de **Visual Studio 2022 (.NET Framework 4.8)**.
- Está permitido el uso de apuntes de clase, diapositivas, videos de clase y ejercicios de clase.
- Está permitido el uso de Internet (páginas de documentación oficial de Microsoft y Oracle).
- Está prohibida toda forma de comunicación con terceros.

**PARTE PRÁCTICA (20 puntos)**

PUEDE UTILIZAR MATERIAL DE CONSULTA.

Antes de comenzar el laboratorio, descargue todos los proyectos, apuntes, diapositivas que utilizará.

Se considerará en la calificación el uso de buenas prácticas de programación (aquellas vistas en clase).

**PREGUNTA 1:**

**NaciónPlay** es una empresa que comercializa videojuegos para las plataformas de Playstation 5, Nintendo Switch, Xbox One. En la actualidad esta empresa desea incursionar en el mercado peruano, y para ello ha aperturado algunas sucursales en Lima. Sin embargo, debido a la gran cantidad de productos que maneja, requiere de un producto de software que le permita gestionar toda la información de aquellos videojuegos que ofrecen al mercado.

Bajo este contexto se le solicita implementar en C# el **registro, listado o búsqueda, selección y visualización** de los datos de videojuegos. Para alcanzar este propósito se ha diseñado la interfaz gráfica que se muestra en la Figura 01.

Figura 01. Interfaz gráfica del formulario a desarrollar

Asimismo, se ha elaborado el diseño de la base de datos el cual se presenta en la Figura 02. Se le solicita descargar el archivo "**ScriptSQL\_Laboratorio05.sql**" y ejecutar el script en su motor de base de datos MySQL alojado en AWS. El script generará las tablas, sus relaciones y los procedimientos almacenados (*stored procedures*) necesarios para la programación de la funcionalidad solicitada. Es necesario indicar que, para realizar el registro de la información de un videojuego, se requiere de la implementación de las funcionalidades de "listar géneros" y "búsqueda/selección de desarrolladora".

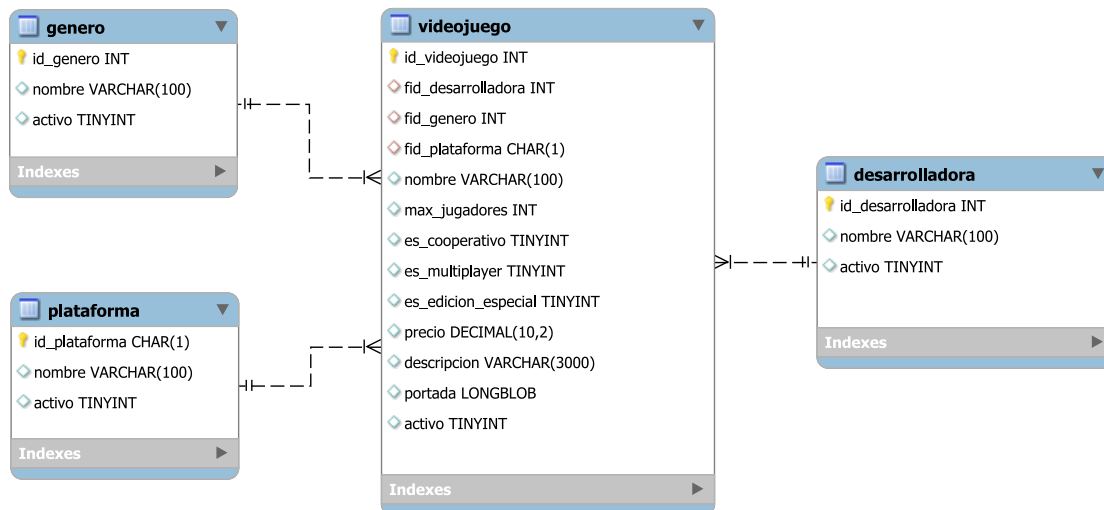


Figura 02. Diagrama EER de la base de datos

Con respecto a la lógica de negocio, un videojuego posee un nombre, ha sido elaborado por una desarrolladora, es de un determinado género, funciona para una determinada plataforma, tiene un número máximo de jugadores que va desde 1 al 10, puede ser o no cooperativo, puede ser o no multijugador (*multiplayer*), puede ser o no una edición especial, tiene un precio de venta al mercado, una descripción y una portada (o imagen).

Para la programación de la funcionalidad solicitada debe utilizar el proyecto en C# que se proporciona en PAIDEIA. No debe programar el proyecto desde cero. Asimismo, **NO se está solicitando implementar ni el modificar ni el eliminar videojuegos. NO son requeridas validaciones.**

Suba su propuesta de solución en un .zip a PAIDEIA. (Realice un *clean* antes de empaquetar).

#### NOTAS SOBRE LA CALIFICACIÓN Y DESARROLLO DE LA PRÁCTICA:

- Es necesario seguir las instrucciones establecidas en el enunciado.
- Se descontarán significativamente puntos si existen errores de compilación.
- Los programas no deben presentar errores en tiempo de ejecución.
- Los proyectos no deben tener referencias innecesarias.
- El *driver* de conexión a base de datos debe adjuntarse al proyecto.
- Que el *driver* de conexión a base de datos no se encuentre referenciado con ruta relativa en el proyecto.
- Es necesario seguir las prácticas vistas en clase: patrón DAO, desarrollo por componentes en capas, etc.
- **No se considerará puntaje alguno si es que no ha trabajado de manera progresiva las funcionalidades**, es decir, primero deberá implementar el **listado de géneros**. Luego las **búsqueda y selección de desarrolladora**. Finalmente implementará **el registro, la búsqueda, la selección y visualización de los datos de videojuegos**.
- No se considerará puntaje alguno si es que, para implementar cada una de las funcionalidades, no ha trabajado de manera progresiva, es decir, primero el *back* y luego el *front*. No se considerará puntaje por programación de *front* si es que no se encuentra correctamente programado el *back*. Es por este motivo que se le solicita trabajar primero, para cada una de las funcionalidades solicitadas: el proyecto DBManager, la capa de acceso a base de datos (las clases DAO y MySQL) y finalmente la capa de presentación (funcionamiento de la interfaz gráfica).

**Rúbrica de Calificación: (El puntaje se alcanza de manera progresiva [para alcanzar el puntaje del siguiente punto es necesario implementar el anterior])**

- Programación del **proyecto DBManager**: 1 punto.
- Programación de la clase GeneroDAO y GeneroMySQL: 2 puntos.
- Programación del combobox para visualizar los géneros registrados en BD: 1 punto.
- Programación de la clase DesarrolladoraDAO y DesarrolladoraMySQL: 2 puntos.
- Se listan las desarrolladoras por nombre a nivel de interfaz y es posible seleccionar la desarrolladora: 3 puntos.
- Programación de la clase VideojuegoDAO y VideojuegoMySQL: 4 puntos.
- Se registran todos los datos de un videojuego ingresados por la interfaz gráfica: 4 puntos.
- Se listan los videojuegos por nombre a nivel de interfaz y es posible seleccionar un videojuego y ver sus datos: 3 puntos.

**Profesor del Curso:**  
**Dr. Freddy Paz**

**22 de septiembre del 2022**

### **ANOTACIONES DE PROGRAMACIÓN:**

**Para asignar la foto al objeto utilice:**

```
FileStream fs = new FileStream(_rutaFotoPortada, FileMode.Open, FileAccess.Read);  
BinaryReader br = new BinaryReader(fs);  
_videojuego.Portada = br.ReadBytes((int)fs.Length);  
fs.Close();
```

**Para enviar la foto al procedimiento almacenado puede utilizar la siguiente sentencia:**

```
comando.Parameters.AddWithValue("_portada", videojuego.Portada);
```

**Para leer la foto de la BD puede utilizar la siguiente sentencia:**

```
videojuego.Portada = (byte[]) reader["portada"];
```

**Para mostrar la imagen desde el arreglo de bytes[] extraído de la BD utilice:**

```
MemoryStream ms = new MemoryStream(_videojuego.Portada);  
pbPortada.Image = new Bitmap(ms);
```