

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJE DE PROGRAMACIÓN 2

6ta. práctica (tipo b)
(Segundo Semestre 2023)

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.
- Se le solicita hacer uso del entorno de desarrollo integrado de **Visual Studio 2022** y **.NET Framework 4.8**.
- Está permitido el uso de apuntes de clase, diapositivas, ejercicios de clase y código fuente.
(Debe descargarlos antes de iniciar con la solución del enunciado).
- Está permitido el uso de Internet (únicamente para consultar páginas oficiales de Microsoft, Oracle y PAIDEIA). No obstante, está prohibida toda forma de comunicación con otros estudiantes o 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: (20 puntos)

Una reconocida cadena de cines peruana desea contar con un producto de software que le permita **gestionar la información relacionada a todas las películas cinematográficas** que adquiere para la proyección en sus salas. Es por este motivo que se le ha contactado a Ud. para que realice la programación del formulario de gestión de películas. Para este laboratorio se le está solicitando programar las **funcionalidades de registro y listado de películas**. Para alcanzar este propósito se ha diseñado la interfaz gráfica que se muestra en la Figura 01. **Es necesario mencionar que, de programar correctamente la funcionalidad de registro y listado de películas, la funcionalidad de selección de películas al momento de listar debería funcionar también de forma correcta. Puede utilizar esta implementación para verificar que ha realizado correctamente lo solicitado.**

Formulario de Gestión de Películas

Nuevo Guardar Buscar Cancelar

Datos Generales de la Película

ID de la Película:

Nombre:

Género:

Productora:

Fecha de Estreno:

Formatos Disponibles: ☒ Doblada ☒ Subtitulada ☐ X-treme ☐ 3D

Costo de Producción (\$):

Sinopsis:

Dom Cobb (Leonardo DiCaprio) es un ladrón con una extraña habilidad para entrar a los sueños de la gente y robarle los secretos de sus subconscientes. Su habilidad lo ha convertido en un atractivo en el mundo del espionaje corporativo, pero ha tenido un gran costo en la gente que ama.

Fig. 01. Interfaz gráfica del formulario a implementar

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 "**Script SQL Laboratorio 06.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 las funcionalidades solicitadas.

Asimismo, es necesario indicar que, para realizar el registro de los datos de una película, se requiere previamente la implementación de la funcionalidad de listar géneros y listar productoras. Cuando se seleccione una productora en específico dentro del ComboBox, el panel superior (panelSuperior) deberá cambiar de color dependiendo del color asociado a la productora seleccionada. Así también, al seleccionar una productora deberá mostrarse el logo asociado a la productora en el componente pbLogoProductora. Utilice el evento "SelectedIndexChanged" del ComboBox para este propósito. Finalmente, utilice "Color.FromArgb" para formar el color a partir de los valores R, G, B y la propiedad BackColor del panel. Use lo descrito en la siguiente sección para mostrar la imagen:

```
MemoryStream ms = new MemoryStream(productora.Logo);
pbLogoProductora.Image = new Bitmap(ms);
```

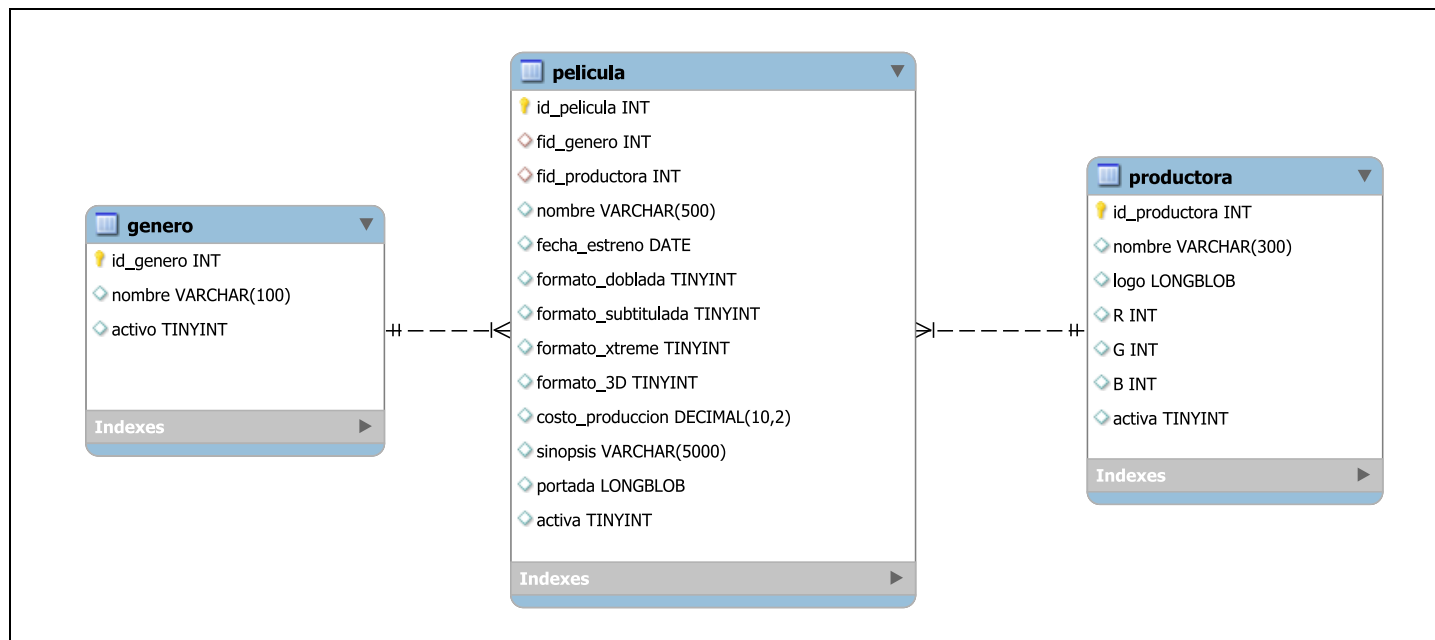


Fig. 02. Diagrama EER de la base de datos

Con respecto a la lógica del negocio, se requiere registrar de cualquier película los siguientes datos: el nombre, la fecha de estreno, si es que viene en formato doblada, si es que viene en formato subtitulada, si es que viene en formato xtreme, si es que viene en formato 3D, el costo de producción, la sinopsis y la portada. Asimismo, la película es de un género y ha sido elaborada y financiada por una productora.

Para la programación de la funcionalidad solicitada debe utilizar el proyecto en C# que se proporciona en PAIDEIA. No se está solicitando implementar ni el modificar ni el eliminar películas. NO son requeridas validaciones por lo que debe asumir que el usuario ingresa correctamente la información al formulario. A continuación, se adjunta un video que muestra la forma en que debería funcionar el formulario: <https://youtu.be/OeJuJliMqGw>

Se le solicita colocar su nombre completo y su código a modo de comentario en las líneas 15-21 de la clase llamada: "frmGestionPelículas". No colocar conllevará a una disminución de 2 puntos en la nota final obtenida.

Está prohibido modificar el proyecto "MovieSoftModel" a nivel de programación y también está prohibido modificar la estructura de tablas y los procedimientos almacenados.

Suba su propuesta de solución en un único .zip a PAIDEIA. (Realice un proceso *clean* antes de empaquetar todos los proyectos [solución] en el archivo .zip). El formato de nombre del archivo es: codigoPUCP_Lab06.zip Verifique que haya subido correctamente su archivo a PAIDEIA. Es responsabilidad de cada estudiante asegurarse de haber subido correctamente a PAIDEIA su propuesta de solución a los ejercicios propuestos. No se aceptarán reclamos por equivocaciones respecto a los archivos entregados. Tampoco se aceptarán archivos entregados por otros medios.

Anotaciones de Programación:

Para asignar la foto de portada de la película a la variable de tipo byte[]

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

Para enviar un arreglo de bytes como parámetro mediante un comando a un procedimiento almacenado

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

En caso se desee leer un LONGBLOB a una variable de tipo byte[]

```
if (!lector.IsDBNull(lector.GetOrdinal("logo"))) productora.Logo = (byte[]) lector["logo"];  
  
if (!lector.IsDBNull(lector.GetOrdinal("portada"))) pelicula.Portada = (byte[])lector["portada"];
```

Para mostrar una imagen en un PictureBox desde una variable de tipo byte[]

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

Rúbrica de calificación:

- No se considerará puntaje alguno si es que el aspecto a evaluar no funciona a nivel de interfaz gráfica, por lo que se solicita que, a medida que va avanzando el *back-end*, realice la programación del *front-end*. Asimismo, la corrección de los apartados será progresiva. Para obtener el puntaje de un apartado en específico, todos los anteriores aspectos a evaluar deben estar correctamente desarrollados.

A continuación, se citan algunos ejemplos:

- “no se considerará puntaje por la programación de la funcionalidad de registro, si es que para ese registro son necesarias algunas búsquedas y selecciones en el formulario y estas no se encuentran correctamente implementadas.”
 - “no se considerará puntaje por programar la funcionalidad de búsqueda si es que el registro no se encuentra correctamente implementado”.
 - “no se considerará puntaje por la programación de un formulario si es que los métodos de acceso a base de datos que son requeridos para su funcionamiento no se encuentran correctamente implementados”.
- El driver de conexión a base de datos debe estar referenciado con ruta relativa (no absoluta).
 - Se descontarán puntos significativamente si es que no se siguen las indicaciones del enunciado.
 - Se descontarán puntos significativamente si es que existen errores de compilación.
 - Se descontarán puntos por declaración de variables que admiten cualquier tipo de dato como "var" (C#).
 - Se descontarán puntos si el código no está optimizado, por ejemplo, repetición innecesaria de código.
 - Se descontarán puntos si es que no sigue las instrucciones en referencia a los métodos que deberían implementarse en cada clase.
 - Se descontarán puntos por referencias innecesarias entre proyectos.
 - Debe utilizar la estructura vista en clase, proyectos relacionados a: DBManager, a la capa de modelo, a la capa de acceso a base de datos y a la vista.

A. Desarrollo del proyecto "MovieSoftDBManager" y sus clases asociadas. (1.0 punto).

B. Implementación de la funcionalidad que permite listar los géneros (*back-end*): (2.0 puntos).

C. Implementación de la funcionalidad que permite listar los géneros (*front-end*): (1.0 punto).

D. Implementación de la funcionalidad que permite listar las productoras (*back-end*): (2.0 puntos).

E. Implementación de la funcionalidad que permite listar las productoras (*front-end*): (1.0 punto).

F. Implementación de los colores y logo asociado a la productora seleccionada (*front-end*): (2.0 puntos).

G. Implementación de la funcionalidad que permite registrar las películas (*back-end*): (3.0 puntos).

H. Implementación de la funcionalidad que permite registrar las películas (*front-end*): (2.0 puntos).

I. Implementación de la funcionalidad que permite listar las películas (*back-end*): (2.0 puntos).

J. Implementación de la funcionalidad que permite listar las películas (*front-end*): (2.0 puntos).

K. Implementación de la selección y visualización de los datos de una película (*front-end*): (2.0 puntos).

Profesor del Curso:

Dr. Freddy Paz

21 de setiembre del 2023