

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

**LENGUAJE DE PROGRAMACIÓN 2**

**12va. práctica (tipo b)**  
**(Primer 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 Apache Netbeans y JDK 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)**

**Nintendo Company Ltd.** es una empresa dedicada al desarrollo de consolas y videojuegos. Actualmente se encuentra trabajando en una nueva versión de **Super Smash Bros**, el cual es un videojuego de lucha donde participan personajes reconocidos de las principales franquicias de Nintendo y de otras compañías del mismo rubro. Nintendo requiere de un programa que le permita gestionar a los personajes que aparecerán en el videojuego. Para este propósito, lo ha contratado a Ud. para que desarrolle la funcionalidad de registro, listado o búsqueda, selección y visualización de personajes. Con este objetivo se ha diseñado la interfaz gráfica que se muestra en la Figura 01.

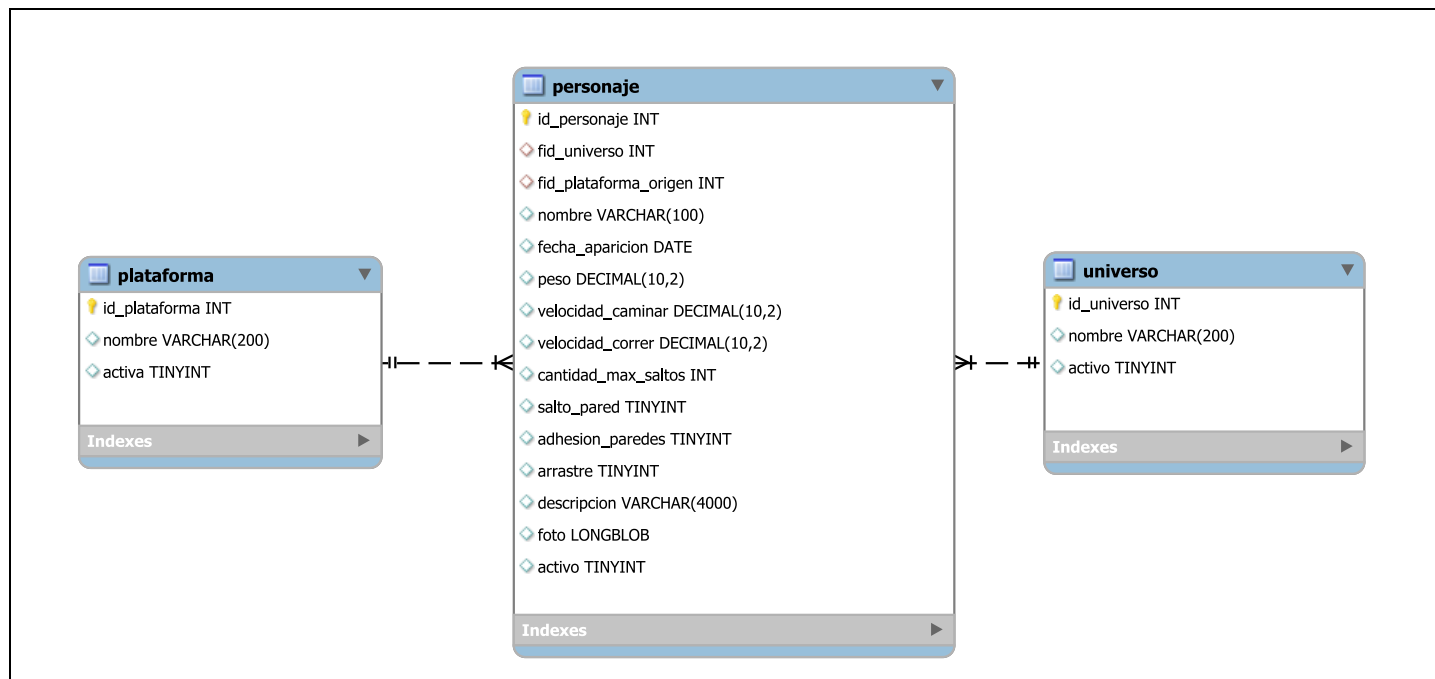
**Fig. 01. Interfaz gráfica del formulario a implementar**

Se le solicita realizar la programación que permita la funcionalidad solicitada. Para esto deberá descargar los proyectos que se encuentran en PAIDEIA. El back-end se ha solicitado en JAVA, mientras que para el front-end debe utilizar C#. La tecnología que debe conectar el back-end con el front-end por disposición de Nintendo debe ser servicios web de tipo SOAP utilizando la tecnología JAX-WS bajo el JDK 1.8 y el Glassfish 5.0 (de acuerdo a lo empleado en clase). Puede utilizar un único servicio llamado "ServicioWS" y trabajar en localhost.

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 "**Laboratorio12-ScriptSQL.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*) que deberá utilizar para la programación de las funcionalidades solicitadas.

Es necesario indicar que, para realizar el registro de los datos de un personaje, se requiere previamente la implementación de la funcionalidad de listar plataformas y listar/buscar/seleccionar universos.



**Fig. 02. Diagrama EER de la base de datos**

Con respecto a la lógica del negocio, se puede mencionar que un personaje debe tener un id que lo genera el motor de base de datos, pertenece a un universo Nintendo, tiene un nombre, es de una determinada plataforma de origen, tiene una fecha de aparición en el universo Nintendo, tiene un peso (número decimal que determina hasta qué punto se le puede enviar volando al personaje por los aires y cuál es su fuerza física), tiene una velocidad al caminar (número decimal que determina lo rápido que puede el personaje deslizarse caminando de un punto a otro), tiene una velocidad al correr (número decimal que determina lo rápido que puede el personaje deslizarse corriendo de un punto a otro), tiene una cantidad máxima de saltos (número entero que determina la cantidad máxima de saltos que puede dar el personaje en el aire), una descripción, una foto e indicadores como si puede o no dar una patada en una pared apoyándose en ella para así saltar y llegar más alto, si puede o no adherirse a las paredes, si puede o no arrastrarse para evitar ataques.

No se está solicitando implementar ni el modificar ni el eliminar personajes.

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/jTAcWL0k6GM>

**Se le solicita colocar su nombre completo y su código a modo de comentario en las líneas 13-18 de la clase llamada: "frmGestionPersonajes". No colocar conllevará a una disminución de 2 puntos en la nota final obtenida.**

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

**Los proyectos SmashSoftDBManager, SmashSoftController y SmashSoftServices deben ser creados desde cero.**

Se procederá a revisar la fecha de creación de las carpetas y en caso verificar que no se ha seguido la indicación (haber trabajado sobre un proyecto ya creado/o sobre el proyecto de clase), no se considerará puntaje alguno.

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\_Lab12.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 y fuera de los plazos establecidos para la evaluación.

## Anotaciones de Programación:

### **Para asignar la foto a la variable de tipo byte[]**

```
FileStream fs = new FileStream(_rutaFotoLocal, FileMode.Open, FileAccess.Read);  
BinaryReader br = new BinaryReader(fs);  
_personaje.foto = br.ReadBytes((int)fs.Length);  
fs.Close();
```

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

```
cs.setBytes("_foto", personaje.getFoto());
```

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

```
personaje.setFoto(rs.getBytes("foto"));
```

### **Para mostrar una imagen en el PictureBox desde una variable de tipo byte[]**

```
MemoryStream ms = new MemoryStream(_personaje.foto);  
pbFoto.Image = new Bitmap(ms);
```

### **En caso de ocurrir algún desbordamiento de memoria en caso de trabajar con un cliente C#, modificar las siguientes líneas relacionadas al servicio en el App.config:**

```
<basicHttpBinding>  
  <binding name="ServicioWSPortBinding" receiveTimeout="00:1:00" sendTimeout="00:1:00"  
    maxBufferSize="2147483647" maxReceivedMessageSize="2147483647"/>  
</basicHttpBinding>
```

### 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”.
- Se ha trabajado con una versión de JDK y Glassfish distinta a la solicitada.
  - 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 "Object" (JAVA) o "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. Datos de conexión en la clase DBManager. (0.5 puntos).

B. Implementación del método "listarTodas()" en las clases PlataformaDAO y PlataformaMySQL. (1.5 puntos).

C. Implementación del método "listarTodasPlataformas()" en el servicio web. (0.5 puntos).

D. Implementación del listado de plataformas en el front-end (combobox). (1 punto).

E. Implementación del método "listarPorNombre" en las clases UniversoDAO y UniversoMySQL. (1.5 puntos).

F. Implementación del método "listarUniversosPorNombre" en el servicio web. (0.5 puntos).

G. Implementación de la búsqueda, selección y asignación de universos al personaje en el front-end. (2.5 puntos).

H. Implementación del método "insertar" en las clases PersonajeDAO y PersonajeMySQL. (2.5 puntos).

I. Implementación del método "insertarPersonaje" en el servicio web. (0.5 puntos).

J. Implementación del registro de un personaje en el front-end. (3 puntos).

K. Implementación del método "listarPorNombre" en las clases PersonajeDAO y PersonajeMySQL. (2.5 puntos).

L. Implementación del método "listarPersonajesPorNombre" en el servicio web. (0.5 puntos).

M. Implementación de la búsqueda y selección de un personaje en el front-end. (3 puntos).

Profesor del Curso:

Dr. Freddy Paz

15 de junio del 2023