



## **Ejercicio de nivel 5**

### **Liga de Bolos**

### **Consideraciones adicionales de diseño**

#### **Consejos prácticos para la construcción de la interfaz gráfica de la aplicación**

1. Recomendación para la distribución de los paneles en la ventana principal.



Cada cuadro en rojo representa un panel como se describe a continuación:

- A. Panel con la imagen del banner.
- B. Panel con la información de un jugador.
- C. Panel con las opciones de búsqueda y de extensión.

2. El segundo panel presentado en el numeral anterior incluye a su vez tres diferentes paneles internos: uno para la imagen o fotografía del jugador, otro para sus datos y para la opción de agregar un nuevo puntaje, y finalmente un tercer panel que incluye las opciones de navegación entre jugadores.

3. Para incluir una imagen en un panel o una ventana, se puede utilizar un `JLabel` de la siguiente manera:

```
JLabel lblImagen = new JLabel( );  
ImageIcon icono = new ImageIcon( rutaImagen );  
lblImagen.setIcon( icono );
```

Donde `rutaImagen` es la ruta donde se encuentra ubicada la imagen. Por ejemplo, la imagen del banner se encuentra en `"/data/imagenes/banner.png"`.

4. Para el manejo de distintos tipos de fuentes en el `JLabel` se usa la clase `Font` de `java.awt`. Para crear una fuente personalizada y asignarla a un `JLabel` se pueden seguir los siguientes pasos:

```
JLabel lblEjemplo = new JLabel( textoLabel );  
Font fuenteNueva = new Font("Arial", Font.BOLD, 20);  
lblEjemplo.setFont(fuenteNueva);
```

En este ejemplo, se asignó una fuente Arial, de tamaño 20, en negrilla a la etiqueta con nombre `lblEjemplo`. El constructor de la clase `Font` recibe 3 parámetros: el tipo de la fuente, el estilo de la fuente y el tamaño de la fuente.

A continuación se muestran dos ejemplos de cómo crear una fuente personalizada:

a) Una etiqueta creada con los siguientes comandos:

```
JLabel lblEjemplo = new JLabel( "Texto ejemplo" );  
Font fuenteNueva = new Font("Arial", Font.BOLD, 15);  
lblEjemplo.setFont(fuenteNueva);
```

Se verá así:

**Texto ejemplo**

b) Los siguientes comandos corresponden a la misma etiqueta pero con una fuente distinta:

```
JLabel lblEjemplo = new JLabel( "Texto ejemplo" );  
Font fuenteNueva = new Font("Comic Sans MS", Font.ITALIC, 30);  
lblEjemplo.setFont(fuenteNueva);
```

Se verá así:

*Texto ejemplo*



Si desea consultar la documentación de la clase Font, para conocer los estilos y tipos de letra disponibles, lo puede hacer en el siguiente enlace:  
<http://docs.oracle.com/javase/6/docs/api/java/awt/Font.html>

5. En Mac se debe tener en cuenta que para poder pintar un botón, se deben activar las siguientes propiedades del mismo:
  - ✓ `botonX.setOpaque( true );`
  - ✓ `botonX.setBorderPainted( true );`
  - ✓ `botonX.setBorder( BorderFactory.createLineBorder( Color.white ) );`

### Explicación de algunos aspectos del mundo

1. El constructor de la clase principal, **LigaBolos**, se encarga de crear todos los elementos cargando la información de un archivo.
2. La clase **LigaBolos** maneja el método **darJugadorActual**, que devuelve el jugador que está actualmente seleccionado.
3. La clase **LigaBolos** maneja los métodos **darSiguienteJugador**, **darAnteriorJugador**, **darPrimerJugador** y **darUltimoJugador**, que devuelven el jugador requerido en cada caso
4. La clase **LigaBolos** maneja los métodos **darJugadorPorNombre**, y **darJugadorMejorPromedio**, que se encargan de realizar búsquedas de un jugador por diferentes criterios. Estos métodos devuelven el jugador buscado.

