



Ejercicio de nivel 5 **Criaturas Mágicas** **Consideraciones adicionales** **de diseño**

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

1. La ventana principal debe tener 780 píxeles de ancho y 750 píxeles de alto.
2. Recomendación para la distribución de los componentes en la ventana principal.



Cada cuadro de color es un panel como se describe a continuación:

- ✓ Etiqueta o panel con la imagen del banner.
- ✓ Panel con la información del juego.
- ✓ Panel con el mapa.
- ✓ Panel con la enciclopedia de criaturas.
- ✓ Panel con las consultas.
- ✓ Panel con las opciones.

3. Para incluir una imagen en un label, se utiliza un ImageIcon 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. En este nivel, cuando se carga el mapa de juego, lo único que hará la aplicación será mostrar una imagen del mismo. Para esto, utilice la imagen que se encuentra en `"/data/mapa.png"`.
5. Para definir las dimensiones que tiene una imagen usted puede utilizar la siguiente instrucción:

```
ImageIcon icono = new ImageIcon( new ImageIcon(  
pCriatura.darRutaImagen( ) ).getImage( ).getScaledInstance( 300, 300,  
Image.SCALE_DEFAULT ) );
```

En esta instrucción se está definiendo el alto y el ancho de la imagen como 300 y 300. Esto le será útil ya que las imágenes de las criaturas tienen distintas dimensiones dependiendo del contexto en el que se estén utilizando, en el panel de la enciclopedia la imagen tiene 200 píxeles de alto y de ancho.

6. Para modificar el color de fondo de la ventana principal, utilice la siguiente instrucción en el constructor:

```
getContentPane( ).setBackground( new Color(48, 41, 84) );
```

Para que un panel u otro componente gráfico no tenga un color de fondo (por defecto es gris) se debe usar la siguiente instrucción:

```
setOpaque( false );
```

De esta forma, el fondo del componente pasa de ser opaco a transparente, y el color de atrás se vuelve visible. Esta instrucción puede ser utilizada en cualquier componente gráfico, de la siguiente manera:

```
txtGustos = new JTextField();  
txtGustos.setOpaque( false );
```

7. Para modificar la apariencia de un label existen distintas instrucciones:

- ✓ Para modificar la fuente:

```
lblPuntos = new JLabel( );  
lblPuntos.setFont( new Font("Helvetica", Font.BOLD, 18) );
```

En el ejemplo se está asignando la fuente Helvética en negrita, de tamaño 18. Todos los labels de la aplicación tienen esta fuente, excepto por algunos labels del panel de la enciclopedia, que su tamaño de fuente es 22.

- ✓ Para modificar el color del label,

```
lblPuntos.setForeground( Color.WHITE );
```

Algunos textos son de color blanco, como en el ejemplo, otros utilizan el color `new Color(255,252,222)`.

- ✓ Para cambiar la alineación del texto,

```
lblPuntos.setHorizontalAlignment( JLabel.CENTER );
```