

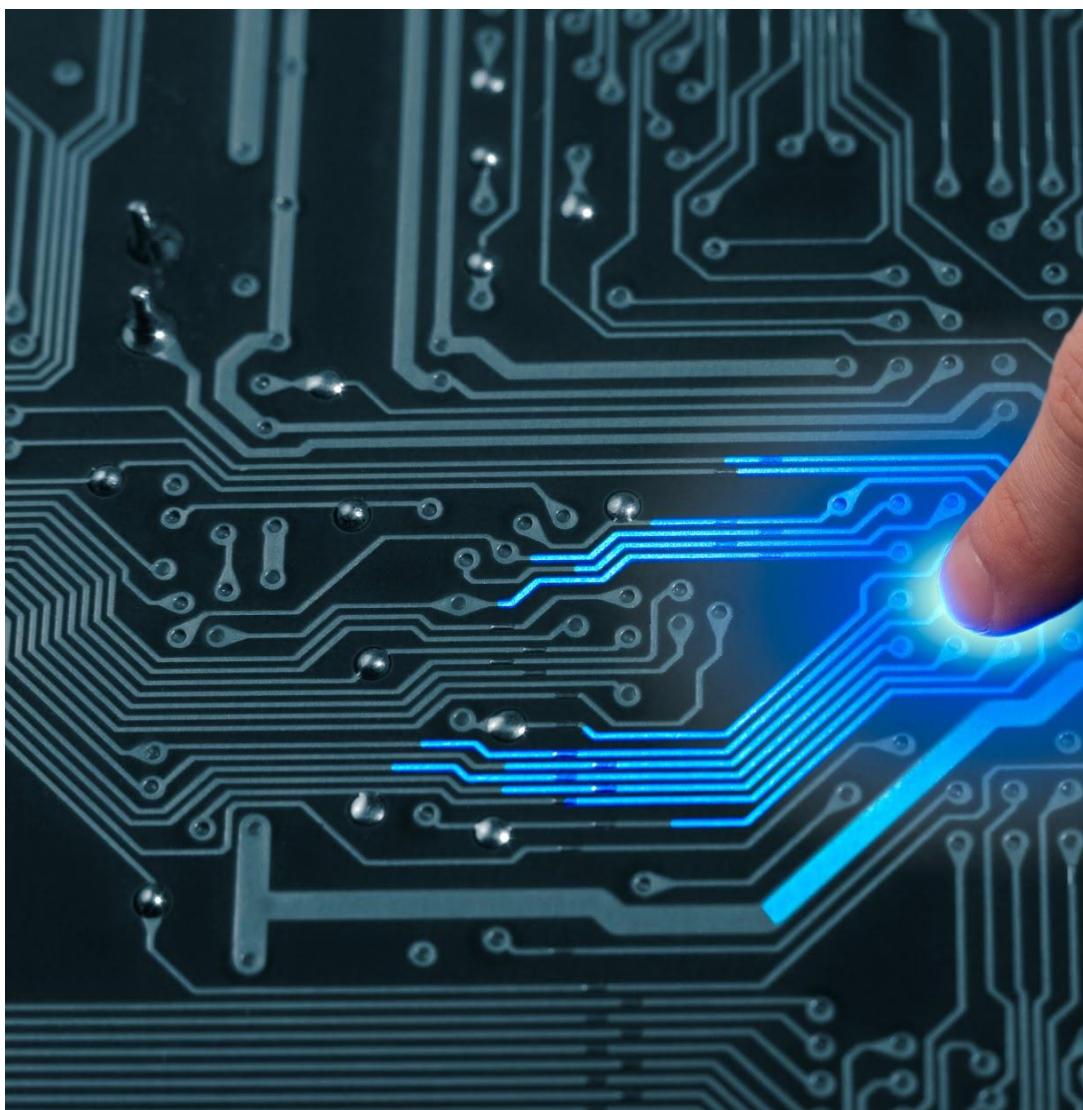
CFGS

DESARROLLO DE APLICACIONES MULTIPLATAFORMA

PROGRAMACIÓN B

PROGRAMACIÓN ORIENTADA A OBJETOS. LIBRERÍAS DE CLASES
FUNDAMENTALES

PAC 1: Interfaces gráficos de usuario



PAC 1: Interfaces gráficos de usuario

INFORMACIÓN

Para responder a las siguientes cuestiones deberás ayudarte del material didáctico y consultar internet.

Requisitos varios que deben cumplirse en vuestros trabajos:

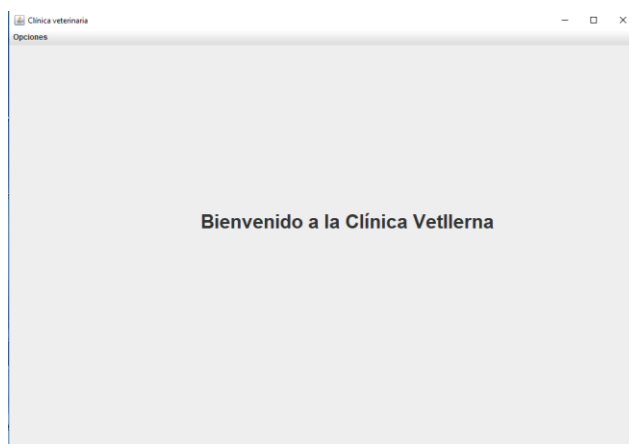
- En los ejercicios, si se requieren de cálculos, estos deben aparecer en la respuesta que planteéis.
- Siempre que utilicéis información de Internet para responder / resolver alguna pregunta, tenéis que citar la fuente (la página web) de dónde habéis sacado aquella información.
- Siempre que utilicéis información del libro digital para responder / resolver alguna pregunta, tenéis que citar el tema y la página de dónde habéis sacado aquella información.
- No se aceptarán respuestas sacadas de Internet utilizando la metodología de copiar y pegar. Podéis utilizar Internet para localizar información, pero el redactado de las respuestas ha de ser vuestro.
- Las respuestas a las preguntas deben estar bien argumentadas, no se admiten respuestas escuetas o monosílabas.
- Si el proyecto no compila, la puntuación máxima será un 4.
- Se valorará la presentación, ortografía y gramática de vuestro trabajo hasta con un punto de la nota final.

Entrega de la PAC:

- Se debe entregar junto con el proyecto comprimido, una memoria en el que se expliquen los pasos realizados junto con los puntos más importantes del código explicados.

1. Vamos a realizar la interfaz gráfica de la clínica Vetllerna.

La aplicación de escritorio tendrá 4 ventanas diferentes. La primera ventana, que es la que aparece al abrir la aplicación, se forma con una única etiqueta:



Tendrá un menú de opciones, que nos permite ir a las demás ventanas de la aplicación.

- **Registro.** Permitirá registrar un cliente con la mascota que se va a atender.

The screenshot shows a window titled 'Clínica veterinaria' with a menu bar containing 'Opciones'. The main area is titled 'Registro Cliente' and contains the following fields and controls:

- Registro Cliente:**
 - Nombre Cliente (text input)
 - DNI Cliente (text input)
 - Dirección Cliente (text input)
- Datos mascota:**
 - Nombre Mascota (text input)
- Tipo:**
 - Radio buttons for Gato, Perro, and Roedor.
 - color (text input)
 - raza (text input)
 - Especialización de roedor (text input)
 - Radio buttons for macho and hembra.
 - Radio buttons for Conejo and Ratón.
- Insertar** (button)

- **NuevaMascota.** Permitirá añadir una nueva mascota a un cliente que ya está registrado.

The screenshot shows a window titled 'Clínica veterinaria' with a menu bar containing 'Opciones'. The main area is titled 'NuevaMascota' and contains the following fields and controls:

- DNI cliente (text input)
- Buscar** (button)
- Mascotas (dropdown menu)
- Datos mascota:**
 - Tratamiento:
 - Fecha (text input)
 - Tratamiento (text input)
 - Añadir** (button)

- **Tratamiento.** Permitirá añadir el tratamiento que se ha realizado a la mascota y mostrar el historial clínico de la misma.

The screenshot shows a Java Swing window titled "Clínica veterinaria". The window has a title bar with standard minimize, maximize, and close buttons. Below the title bar, the text "Opciones" is displayed. The main area of the window contains a form with the following elements:

- A text input field labeled "DNI cliente".
- A "Buscar" button.
- A dropdown menu labeled "Mascotas".
- A section header "Datos mascota".
- A section header "Tratamiento".
- A text input field labeled "Fecha".
- A text input field labeled "Tratamiento".
- An "Añadir" button.
- A large, empty rectangular area below the input fields, likely intended for displaying a list or history of treatments.

Para realizar esta PAC se debe utilizar el software Eclipse, sin el plugin de WindowsBuilder. Para trabajar con la parte gráfica, no es obligatorio utilizar Layouts. Por ello, para todos los que sea vuestra primera aplicación gráfica podéis utilizar:

```
setLayout(null);
```

Para colocar los elementos, podéis entonces utilizar el método `setBounds()`, en el siguiente enlace podéis consultar todo lo necesario para su uso:

[https://docs.oracle.com/javase/7/docs/api/java/awt/Component.html#setBounds\(int,%20int,%20int,%20int\)](https://docs.oracle.com/javase/7/docs/api/java/awt/Component.html#setBounds(int,%20int,%20int,%20int))

No hay que dar funcionalidad al código, únicamente hay que trabajar la parte gráfica. La única parte de funcionalidad es indicar algunos elementos que estén deshabilitados al crear la pantalla.

¡Buen trabajo!

