

Fecha Inicial: 19/05/25

Grupo De Trabajo: Luis Tinoco y Matías Benavides

Fecha:	18/05/25
Nombre:	Matias Benavides
¿Qué he hecho?	Hacer la función de mostrar los detalles e iniciar con la creación del xml
¿Qué haré a continuación?	Terminar el xml
¿En qué ocupo ayuda?	En como se planea hacer las matrices para integrar los detalles, y nada mas

Evidencias:

```

def cargarPokemon(idPokemon, nombre, entradas, imagen):
    """
    Carga los datos del Pokémon desde la API y actualiza la interfaz.

    Args:
        idPokemon: ID del Pokémon a cargar
        nombreLabel: Label para mostrar el nombre
        entradas: Diccionario de campos de entrada para los datos
        imagenLabel: Label para mostrar la imagen
    """
    try:
        datosPokemon = obtenerInfoPokemon(idPokemon)
        if datosPokemon:
            for id, datos in datosPokemon.items():
                nombre.config(text=datos[0].capitalize())
                # Llenar campos
                esShiny, peso, altura = datos[1]
                totalEstad, stats = datos[2]
                tipos = datos[3]
                urlImagen = datos[4]
                entradas["Peso:"].insert(0, f"{peso/10:.1f} kg")
                entradas["Altura:"].insert(0, f"{altura/10:.1f} m")
                # Lista original de tipos
                tiposCap = []
                for t in tipos:
                    tipoCap = t.capitalize()
                    # Agregar el resultado a la nueva lista
                    tiposCap.append(tipoCap)
                # Unir todos los elementos capitalizados usando "/" como separador
                tipos = "/".join(tiposCap)
                entradas["Tipos:"].insert(0, tipos)
                # Llenar estadísticas
                camposStats = ["PS:", "Ataque:", "Defensa:", "A Esp:", "D Esp:", "Veloc:"]
                for i, campoStat in enumerate(camposStats):
                    entradas[campoStat].insert(0, stats[i])
                entradas["Estad T:"].insert(0, totalEstad)
                # Cargar la imagen
                cargarImagen(imagen, urlImagen)
    except Exception as e:
        print(f"Error al cargar el Pokémon: {e}")

def mostrarDetallePokemon(idPokemon):
    """
    Muestra una ventana con los detalles del Pokémon.

    Args:
        idPokemon: ID del Pokémon a mostrar (por defecto: 16, Pidgy)
    """
    # Crear la ventana
    ventana = tk.Toplevel()
    ventana.title("Detalle del Pokémon")
    ventana.resizable(False, False)
    # Marco principal para el contenido
    marcoContenido = tk.Frame(ventana, bg="white")
    marcoContenido.pack(fill=tk.BOTH, expand=True, padx=5, pady=5)
    # Marco para el nombre y la imagen

```