

HISTORIA INTERACTIVA

GRUPO CAPICÚA

START

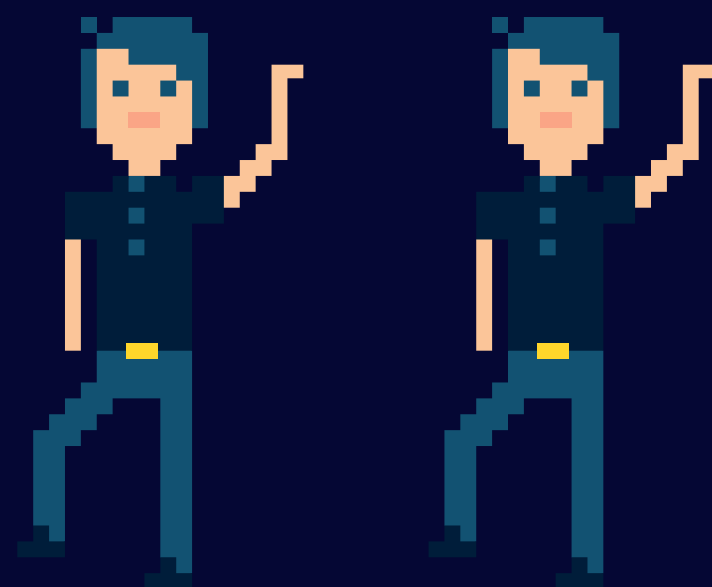




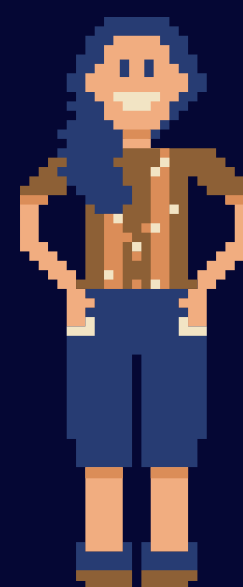
NUESTRO EQUIPO



SERGIO Y
SERGGIO ARTURO



CAMILA NÚÑEZ



EDWIN
FIGUEROA





DESCRIPCIÓN DEL PROYECTO

"NO DEJES DE CREER
EN LA MAGIA DEL
MOMENTO"



Nuestro proyecto consiste en crear un videojuego para PC. La temática del juego consiste en el viaje de un marino español llamado Arturo Figueroa de Núñez y Díaz que intenta llegar a la Fuente de la Juventud y durante su travesía se enfrentará a diferentes problemas que el jugador deberá resolver. Al ser un juego de género Historia interactiva, el jugador podrá elegir el rumbo del personaje principal a través de diferentes opciones y por ende diferentes finales de la historia. El reto que deberá resolver el jugador para pasar de nivel serán operaciones de razonamiento matemático y verbal.



OBJETIVOS



GENERAL

Crear un juego pedagógico

01

04

SECUNDARIO

Que el usuario se divierta cambiando la historia a su antojo

SECUNDARIO

Usar las herramientas que este curso nos ha proporcionado

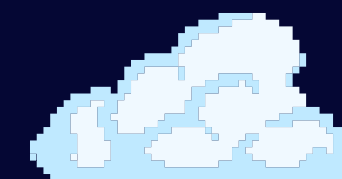
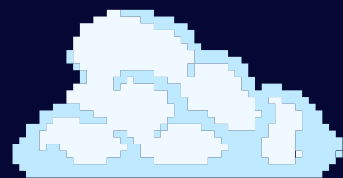
02

05

SECUNDARIO

Que el usuario ponga a prueba sus habilidades de resolución de problemas matemáticos y de lenguaje

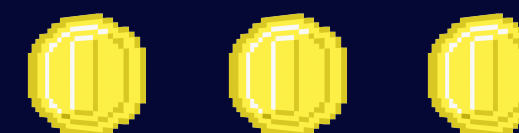


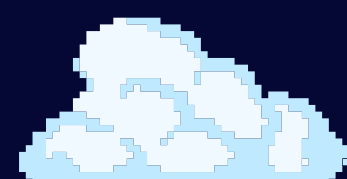
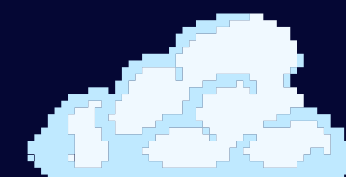


01

RESULTADOS ESPERADOS

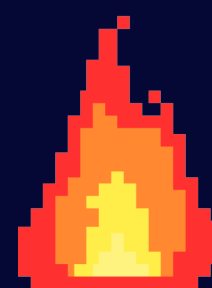
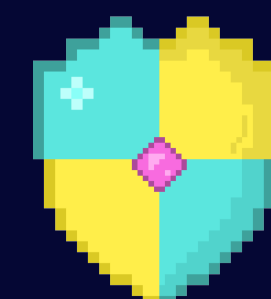
CREAR UNA EXPERIENCIA QUE LE PERMITA
AL SUJETO APRENDER A TRAVÉS DEL
JUEGO SOBRE TEMAS TANTO DE HISTORIA
UNIVERSAL COMO DE MATEMÁTICA, LENGUA
Y CULTURA POPULAR.





02

CÓDIGO DEL PROYECTO





Importa el módulo Tkinter. Tkinter es una biblioteca estándar de Python para crear interfaces gráficas.

Importa la clase "simpdialog" del módulo tkinter. Esta clase se utiliza para crear cuadros de diálogo simples para la entrada del usuario.

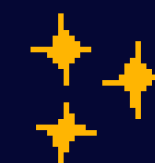
```
1 import tkinter as tk
2 from tkinter import simpdialog # Nueva importación para el cuadro de diálogo de entrada
3 from PIL import Image, ImageTk
4 import requests
5 from io import BytesIO
6
7 import pygame
8 import time
9 import threading
10
11
```

Importa el módulo pygame, que se utiliza para trabajar con audio y video en juegos y aplicaciones multimedia.

Importa el módulo threading, que se utiliza para trabajar con hilos (threads) en Python.

Importa el módulo requests, que se utiliza para realizar solicitudes HTTP.

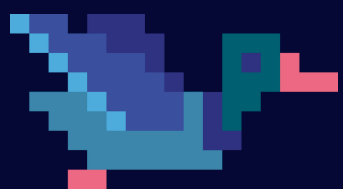
En esta primera parte, se importan las bibliotecas necesarias para la interfaz gráfica, manipulación de imágenes, solicitudes HTTP, manipulación de audio y trabajo con hilos.





```
12 # Esta función se encargará de reproducir el sonido en un hilo separado
13 def reproducir_audio(archivo_wav):
14     def reproducir():
15         pygame.mixer.init()
16         pygame.mixer.music.load(archivo_wav)
17         pygame.mixer.music.play()
18         while pygame.mixer.music.get_busy():
19             time.sleep(1)
20
```





```
24  ▼ def cargar_imagen(url):  
25      ▼ try:  
26          imagen_respuesta = requests.get(url)  
27          imagen_respuesta.raise_for_status()  
28  
29          imagen_pil = Image.open(BytesIO(imagen_respuesta.content))  
30          imagen_tk = ImageTk.PhotoImage(imagen_pil)  
31          label_imagen.config(image=imagen_tk)  
32          label_imagen.image = imagen_tk  
33      ▼ except Exception as e:  
34          print(f"Error al cargar la imagen: {e}")
```





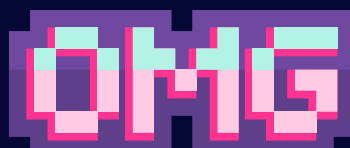
```
def mostrar_historia(texto_historia, opciones):
    texto.delete(1.0, tk.END)
    texto.insert(tk.END, texto_historia)
    crear_botones_opciones(opciones)

def crear_botones_opciones(opciones):
    for boton in botones_opciones:
        boton.destroy()
    botones_opciones.clear()
    for opcion_texto, opcion_accion, *resto_opcion in opciones:
        if resto_opcion:
            url_imagen = resto_opcion[0]
            boton = tk.Button(root, text=opcion_texto, command=lambda url=url_imagen: opcion_accion(url), fg="white", bg="black", font=("Arial", 16))
        else:
            boton = tk.Button(root, text=opcion_texto, command=opcion_accion, fg="white", bg="black", font=("Arial", 16))
        boton.pack(side="top", pady=5)
        botones_opciones.append(boton)
```





```
54 def iniciar_historia():
55     imagen_inicial_url = "https://i.ibb.co/qMDYKxZ/Imagen-Juan-Sanchez-1.jpg" # Reemplaza esto con la URL de tu imagen inicial
56     cargar_imagen(imagen_inicial_url)
57     historia_inicial = "Érase una vez, en el año 1512, el explorador Juan Sánchez se encontraba reflexionando sobre los relatos de"
58
59     opciones = [
60         ("La corona le da financiamiento", Corona_da),
61         ("La corona no le da financiamiento", Corona_no_da)
62     ]
63
64     mostrar_historia(historia_inicial, opciones)
65
66
67 def Corona_da(url_imagen=None):
68     nueva_historia = "Juan Sánchez inicia su expedición con 4 barcos y 160 hombres. Luego de 5 noches, se cruzan con un inmenso k"
69     imagen1 = "https://media.istockphoto.com/id/1306702763/es/foto/buque-de-querra-navegando-por-el-mar-durante-una-tormenta.jpg"
70     opciones = [
71         ("Enfrentarse al kraken", Enfrenta_Kraken),
72         ("Desviarse de la ruta original para evitar al kraken", Desvío_Kraken)
73     ]
74     cargar_imagen(imagen1)
75     mostrar_historia(nueva_historia, opciones)
76     # Llamar a la función para reproducir música de fondo
77     reproducir_audio('waves.wav')
78     mostrar_historia(nueva_historia, opciones)
79
```

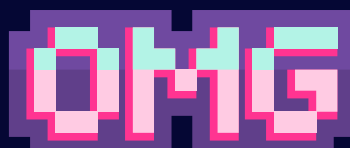




simplifiedialog.askinteger
es una opción parecida
al input pero que se usa
para aplicaciones
gráficas

Condicionales para
verificar respuesta

```
117 # Añade un ejercicio matemático
118 respuesta_correcta = 4
119 respuesta_usuario = simplifiedialog.askinteger("Ejercicio Matemático", "Resuelve: 2 + 2")
120
121 # Verifica la respuesta
122 if respuesta_usuario == respuesta_correcta:
123     opciones = [
124         ("Decide atacar porque piensa que los indígenas son violentos", Ataca_fin1),
125         ("Intenta conversar con la tribu indígena", Conversa_fin2)
126     ]
127     cargar_imagen(url_imagen) if url_imagen else None
128     mostrar_historia(nueva_historia, opciones)
129 else:
130     # Si la respuesta es incorrecta, puedes mostrar un mensaje y volver a solicitar la respuesta o manejarlo según tus necesidades.
131     print("Respuesta incorrecta. Inténtalo de nuevo.")
132
```





Se crea una instancia llamada "Tk" que lo que hace es crear la ventana principal del juego

Luego se crean "widgets" a través de "label" para controlar las imágenes y el texto tanto de las descripciones como las de los botones (fuente, tamaño, grosor, etc.)

```
157 # Crear la ventana principal
158 root = tk.Tk()
159 root.title("Historia Interactiva")
160 root.geometry("800x600")
161
162 # Crear widgets
163 label_imagen = tk.Label(root)
164 label_imagen.pack()
165
166 texto = tk.Text(root, wrap="word", width=80, height=10, font=("Arial", 14), padx=10, pady=10)
167 texto.pack()
168
169 botones_opciones = []
170
171 # Iniciar la historia
172 iniciar_historia()
173
174 # Bucle principal de la interfaz gráfica
175 root.mainloop()
176
```

Luego se crea un bucle con la función "root.mainloop()" el cual mantiene abierta la ventana del juego.



GAME OVER

GRACIAS