

INFORME PYGAME ARCADE

Emilio Schiesser V.

1. Informacion:

Repositorio: https://github.com/Mamorut/Melty_Blood_Arcade_Test

Para ejecutar el proyecto basta con descargar la carpeta " Melty_Blood_Test ", luego ejecutar el archivo " Melty_Blood_Prueba_Terminada "

2. Descripción del proyecto:

Mi proyecto consiste en animar un personaje del juego "Melty Blood", del cual obtuve los sprites necesarios, en pygame junto con probar diferentes elementos que se disponen de la librería pygame.

El proyecto al correr despliega una ventana de 640 x 480 en pantalla completa, al momento de iniciar, se introduce un tema musical, "The end of 1000 years" del mismo juego. Dentro de la ventana se dispone del personaje a animar, Akiha para ser exactos, el cual nos permite interactuar con el programa mediante la creación de eventos para interactuar con el teclado.

El programa reconoce las siguientes teclas junto con sus respectivas acciones:

Key_up --> acciones de salto hacia arriba y salto en diagonal.

Key_down --> acciones de agacharse.

Key_left --> acciones de caminar a la izquierda.

Key_right --> acciones de caminar a la derecha.

Key_a --> golpe 1

Key_s --> golpe 2

Key_d --> patada 1

Key_g --> burla

Key_esc --> cierre y salida del programa.

Nota: Es importante explicar que este programa se basa en los conocimientos básicos de los fighters junto con sus respectivas leyes de prioridad de efectos.

Gracias al reconocimiento de las teclas dadas, nuestro personaje es capaz ese tipo de acciones que son en sí, las principales en todo fighter clásico.

Los diferentes comandos utilizados en pygame para la elaboración de este proyecto fueron principalmente *:

pygame.mixer --> permite la integración de sonido al juego.

pygame.image.load --> permite cargar imágenes al programa.

pygame.event.get() --> reconocedor de eventos en pygame.

pygame.key.get_pressed()[] --> reconoce la tecla presionada.

pygame.quit() --> cierre del programa.

screen.blit --> despliegue de la imagen o fondo en la ventana.

pygame.time.wait --> detiene la acción por un determinado tiempo.

* Esto hay que agregarles los comandos vistos en python.

3. Tiempo en Pygame:

Estimo que inicie con pygame la penúltima semana de agosto, en ese entoces comencé con la búsqueda de tutoriales de pygame y me abastecí de un libro que introduce y enseña pygame: “Beginning-game-development-with-python-and-pygame-from-novice-to-professional” by Will McGugan.

El tiempo dedicado a pygame, estimo que seran unas 6 – 7 horas a la semana aprox.

4. Entendimiento en Pygame, Python y Github:

Con respecto al conocimiento en Pygame, diría que aprendí mucho con respecto a animación de personajes, la creación de eventos y sonido.

En python, conozco lo aprendido en las clases de intro a la programación más algunos comandos extras vistos en el libreo guía ocupado, lo referente a listas y librerías junto con un poco de clases.

En github no he podido aprender lo suficiente, me ha costado mucho entender el funcionamiento de este. Eso sí, ya se manejarlo un poco mejor pero todavía me siento inexperto en el tema.

5. Dificultades en el proceso:

Dentro de todo el desarrollo del proyecto, también existieron complicaciones:

- 1- Ocurrieron problemas con respecto a la creación de las animaciones en el momento de interrelacionarlas con los eventos teclas. Esto se solucionó gracias a la separación de ambos temas y en la creación de estados que marcaran el inicio y fin de alguna de las animaciones.
- 2- Se requerían buscar una forma de hacer fluido las animaciones por los sprites. Para ello, se requirió el comando `pygame.time.wait`, el cual permitió detener la animación por un tiempo leve para así poder hacer la animación más fluida.

En general, no pasaron mayores inconvenientes que esos durante la realización del proyecto. Actualmente, se está buscando la forma de poder arreglar ciertas animaciones para el despliegue de un segundo personaje, por lo cual, aun no se ha resuelto, pero eso queda como otro desafío.

6. Próximos desafíos:

En general, el desarrollo de animaciones todavía está en proceso de perfeccionamiento, todavía se requiere entender y aplicar:

- 1- Colisión de personajes.
- 2- Despliegue simultaneo de personajes.
- 3- Aplicación de ventanas para interactuar (menú, ventana comandos, etc)