Trabajo Final – Programación Declarativa

Fecha: *25/11/2022*
Profesor: *Daniele, Valeria*

Integrantes del grupo:
Mora, Jonathan
Lugani, Santiago
Ventura, Gino

DESCRIPCIÓN DEL PROYECTO

El proyecto realizado como trabajo final de la materia Programación Declarativa consiste en un videojuego de plataformas. El juego describe las aventuras de un pirata llamado Pascal el cual naufragó en una isla y debe recuperar el tesoro que llevaba, ahora en manos de unos monstruos.

A través de seis diferentes niveles, el jugador podrá controlar al personaje y enfrentarse a los temibles monstruos que habitan en la isla, con el fin de recuperar su tesoro.

PERSONAJES

PASCAL

Es el personaje principal y el que el jugador podrá controlar a través de la aventura para recuperar su tesoro perdido.

MONSTRUOS

Son los enemigos de Pascal, cuando naufragó se apoderaron de su tesoro, el jugador deberá derrotarlos para poder recuperar sus monedas de oro y plata.

JUGABILIDAD

El jugador toma el rol de Pascal. El objetivo es recorrer la isla para derrotar a las fuerzas del rey de la isla y recuperar su tesoro. Si recibe contacto enemigo, pierde un porcentaje de vida, por ello, Pascal tiene un ataque que consiste en simplemente saltar sobre el enemigo – presionando la tecla “espacio”.

El juego consta de 6 niveles, cada nivel es diferente. Al final de los niveles hay un sombrero pirata el cual Pascal debe atrapar para poder acceder al siguiente nivel.

OBJETOS

BLOQUES

Los bloques donde el personaje puede avanzar son sobre cajas desparramadas en la isla las cuales era la que llevaba Pascal en su barco o sobre la superficie de la isla propiamente dicha.

SOMBRERO PIRATA

Es la meta de cada nivel, para poder pasar de nivel el jugador debe saltar sobre él para agararrlo.

BARRA DE VIDA

Indica la salud actual del personaje, cuando esta se acaba, se reinicia el juego desde el nivel 1.

MONEDAS

Son las monedas que llevaba el tesoro de Pascal, luego del naufragio se desparramaron por toda la isla. Recolectarlas ayudará a obtener una puntuacion mayor.

Las monedas de oro valen 5 puntos y las de plata valen 1 punto.

AGUA

Pascal no sabe nadar, asique, ¡NO te caigas al agua!

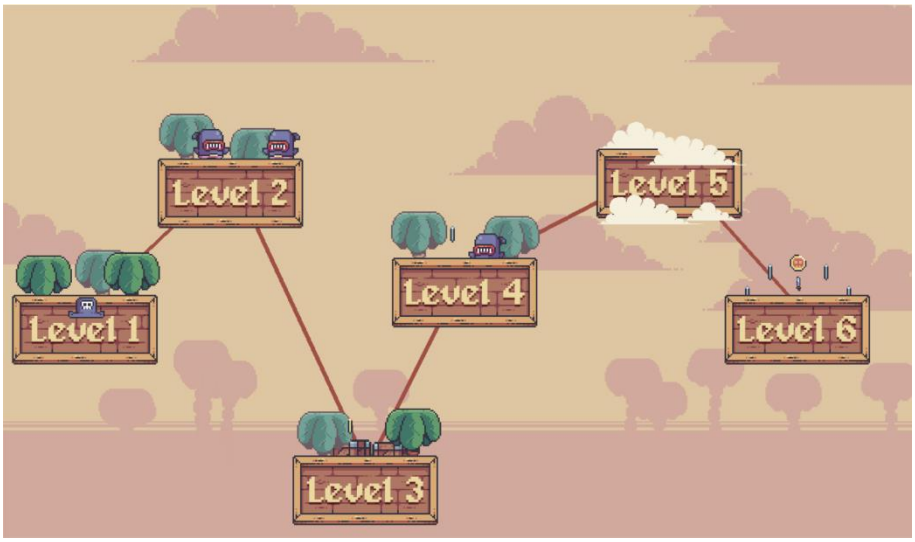
CAPTURAS DEL JUEGO

PANTALLA PRINCIPAL

INSTRUCCIONES DE JUEGO

CRÉDITOS

NIVELES



DENTRO DEL JUEGO

ESPECIFICACIONES TÉCNICAS DEL PROYECTO

El proyecto fue desarrollado con el lenguaje de programación Python, junto con la librería PyGame, que es un conjunto de módulos que permiten la creación de videojuegos en dos dimensiones.

Pygame está basada en SDL, que es una librería que nos provee acceso de bajo nivel al audio, teclado, ratón y al hardware gráfico de nuestro ordenador. Es un producto que funciona en cualquier sistema: Mac OS, Windows o Linux.

El SDL son bibliotecas en lenguaje C para gestión de gráficos 2D (manipulación de las imágenes como objetos de 2D en el lienzo, es decir, la ventana), imágenes (ficheros de tipo .jpg o .png), audio y periféricos a bajo nivel (teclado, ratón).

ESTRUCTURA DE UN VIDEOJUEGO CON PYGAME

La estructura básica de un videojuego con programación entre PyGame y Python orientada a objetos se define por:

- ***Preparación del entorno:*** se debe importar PyGame a nuestro programa en Python en nuestro entorno virtual, es una librería que no forma parte del *startup* de Python. Para ello, se debe instalar desde la consola de comandos con el comando: *pip install pygame*.
- ***Bucle principal de evento-actualización-repintado:*** aquí se encuentra el constructor y la función de lanzamiento del videojuego. Este último creará el bucle con funciones como *start, mainloop y handleEvent.*
- ***Finalizacion del juego:*** cuando se finaliza PyGame, es decir, el juego.

ESQUELETO DE UN PROGRAMA CON PYGAME

Los elementos principales de un programa con PyGame son:

- ***Función main() o clase Game():*** contenedor del videojuego.
- ***Control de eventos:*** la lista de eventos a procesar.
- ***Sprites:*** rectángulos que representan los objetos móviles o fijos del juego. Estos pueden animarse y también detectar colisiones entre ellos.
- ***Sonidos:*** se pueden incluir sonidos a las distintas instancias del videojuego.
- ***Textos:*** se pueden incluir textos.

ASPECTOS DE UN VIDEOJUEGO

Un videojuego tiene 4 aspectos básicos:

- ***Mecánica:*** responde a qué y cómo lo debe hacer el juego.
- ***Historia:*** es una secuencia de eventos lineal o ramificada.
- ***Estética:*** es el aspecto del juego, es decir, gráficos, audios o características visuales.
- ***Tecnología:*** forma o metodo en la que se representa el videojuego. En este caso, PyGame y Python.

MODELADO DE NIVELES

Para modelar los niveles se utiliza el programa Tiled.

Tiled es un editor de niveles 2D que ayuda a desarrollar el contenido del juego. Su característica principal es editar mapas de mosaicos de varias formas, pero también admite la colocación de imágenes. Se enfoca en la flexibilidad general mientras intenta mantenerse intuitivo.

En términos de mapas de mosaicos, admite capas de mosaicos rectangulares rectos, pero también capas isométricas proyectadas, isométricas escalonadas y hexagonales escalonadas.

Las capas de objetos ofrecen mucha flexibilidad para agregar casi cualquier información a tu nivel que tu juego necesite.

Otras cosas que vale la pena mencionar son el soporte para agregar mapas personalizados o formatos de mosaicos a través de complementos, extender Tiled con JavaScript, la memoria de sellos de mosaicos, el soporte de animación de mosaicos y el editor de colisión de mosaicos.

ARCHIVOS

- ***Decoration.py:*** se encarga de la decoración del entorno, tanto del fondo de pantalla principal como el fondo donde se selecciona los niveles y dentro de cada nivel. Le da movimiento al agua y permite que el fondo del nivel no sea estático.
- ***Enemy.py*** encargado de darle vida a cada enemigo que aparece en los niveles.
- ***Game_data.py:*** se encarga de contener todos los archivos necesarios para que se genere el nivel, haciendo referencia a los archivos .csv.
- ***Intro.py:*** se encarga de manejar la pantalla principal y comenzar el juego al pulsar la opcion “Jugar”.

- ***Level.py:*** encargado de darle “vida” al nivel, de acuerdo a los datos en game_data.py y los del archivo .csv, crea los bloques con su correspondiente gráfica.
- ***Main.py:*** clase principal. Se encarga de inicializar todas las variables, como los atributos del juego, musica, opciones, ventana, etc.
- ***Overworld.py:*** contiene toda la lógica para mostrar los niveles disponibles, como también para inicializarlos según el seleccionado.
- ***Particles.py:*** es la encargada de generar las particulas al saltar, correr o matar un enemigo.
- ***Player.py:*** se encarga de controlar al jugador, los movimientos, tiempo de invencibilidad al colisionar con un enemigo, sonidos del jugador, etc.
- ***Settings.py:*** contiene la configuracion del tamaño de la ventana del juego.
- ***Support.py:*** contiene todas las importaciones de carpetas para los archivos .csv y gráficos.
- ***Tile.py:*** carga los gráficos depende la clase de baldosa que esté especificada en el archivo .csv.
- ***UI.py:*** muestra la salud del personaje y las monedas recogidas en total en todos los niveles.

BIBLIOGRAFÍA Y DOCUMENTACIÓN

https://www.pygame.org/docs/

https://docs.python.org/3/

https://es.wikipedia.org/wiki/Python

https://es.wikipedia.org/wiki/Pygame

https://doc.mapeditor.org/en/stable