



Herramientas computacionales: el arte de la programación (Gpo 102)

Evidencia de proyecto

Profesora:

Dra. Verónica Rodríguez Rodríguez

Integrantes

~

Manuel Villalpando Linares	A01352033
Oliver Vázquez Lima	A01738020

Fecha de entrega:

22 de marzo del 2024

Documento de Documentación del Proyecto: Cosmic Rush

1. Introducción

Cosmic Rush es un juego simple desarrollado en Python que desafía a los jugadores a controlar una nave espacial y esquivar asteroides para obtener la mayor puntuación posible. Este documento proporciona una visión general del proyecto, incluyendo su diseño, implementación, funcionalidades principales y futuras mejoras.

2. Tecnologías Utilizadas

El proyecto se desarrolló utilizando las siguientes tecnologías:

- Python: Lenguaje de programación principal utilizado para el desarrollo del juego.
- Biblioteca Turtle: Utilizada para la visualización gráfica del juego.
- Freegames: Proporcionó funciones y utilidades adicionales para el desarrollo del juego.

3. Diseño del Juego

El diseño del juego se centró en crear una experiencia de juego simple pero desafiante. Se implementaron las siguientes características principales:

- Movimiento de la nave espacial controlada por el jugador.
- Generación aleatoria de asteroides para aumentar la dificultad.
- Detección de colisiones entre la nave y los asteroides.
- Puntuación que aumenta a medida que se esquivan los asteroides.

4. Implementación

La implementación del juego se llevó a cabo utilizando Python y la biblioteca Turtle. Se diseñaron y desarrollaron las funciones clave para controlar el movimiento de la nave, generar asteroides y detectar colisiones. Se prestaron especial atención a la optimización del rendimiento y la legibilidad del código.

5. Funcionalidades Principales

- Las funcionalidades principales del juego incluyen:
- Movimiento suave y responsivo de la nave espacial.
- Generación aleatoria y progresiva de asteroides.
- Detección precisa de colisiones entre la nave y los asteroides.
- Puntuación que se incrementa con el tiempo y la dificultad.

6. Pruebas y Depuración

Se llevaron a cabo pruebas para garantizar el funcionamiento correcto del juego. Se identificaron y corrigieron varios errores y problemas durante el proceso de desarrollo como el de que conforme más rapido avanzabas se veían menos asteroides. Se utilizaron técnicas de depuración y pruebas de regresion e implementacion para resolver problemas de rendimiento y comportamiento.

7. Resultados y Conclusiones

El juego resultante proporciona una experiencia divertida y desafiante para los jugadores. Se logró el objetivo principal de crear un juego simple pero adictivo. Se extrajeron lecciones importantes sobre el diseño y desarrollo de juegos en Python.

8. Futuras Mejoras

- Se identificaron varias áreas para posibles mejoras futuras, incluyendo:
- Incorporación de sonidos y efectos de sonido.
- Adición de niveles de dificultad ajustables.
- Implementación de opciones de personalización de la nave.

9. Referencias

Se utilizaron varios recursos durante el desarrollo del proyecto, incluyendo tutoriales en línea, documentación oficial de Python y la biblioteca Turtle, y foros de desarrollo de juegos.

10. Anexos

<https://github.com/A01352033/Python-Ships-Game.git>

