

El Juego

El juego toma inspiración de un clásico juego de mesa originario de Nápoles, Italia. El juego original está diseñado para 2 y 4 jugadores, dependiendo del estilo, y se basa en un sistema de puntos diferente al de los típicos juegos de mesa con cartas. De hecho, en este sistema se reciben puntos en base a la cantidad mayor de cartas por tipo, sin tener en cuenta el valor numérico de las cartas en sí. Además de esto, se pueden ganar puntos haciendo “scopa”, de donde proviene el nombre del juego.

Antes de explicar qué es “scopa”, es importante mencionar la mecánica principal. Para simplificar, vamos a explicar las reglas como si estuvieran jugando 2 jugadores. El juego comienza con 3 cartas por jugador y colocando las siguientes 4 cartas en la mesa. En cada turno, cada jugador debe elegir una carta que coincida en valor numérico con las cartas de la mesa. Si, por ejemplo, en la mesa hay un 4, 5, 3 y 8, y el jugador A tiene un 1, 2 y 8, el jugador podrá tomar la carta de valor 8 o bien una combinación de 5 y 3 (la prioridad la tiene la carta de valor exacto). Si no hay combinaciones disponibles, el jugador está obligado a colocar una de sus cartas en la mesa. El turno continúa hasta que ambos jugadores se quedan sin cartas y comienza un nuevo giro, hasta que se acaben las cartas en la baraja, compuesta por un total de 40 cartas, de 1 a 10 por cada uno de los 4 tipos. Para hacer “scopa”, el objetivo principal del juego, el jugador debe recoger todas las cartas de la mesa con una sola carta. El nombre “scopa”, o “escoba” en español, hace referencia al hecho de limpiar la mesa de todas las cartas.

En la versión realizada, al jugar en solitario, tuve que modificar las reglas originales del juego, aunque la mecánica de combinar las cartas con el mismo valor numérico se mantiene. El puntaje se basa en la suma de los valores numéricos de todas las cartas recogidas. Se debe alcanzar un mínimo de 60 puntos (al obtener todas las cartas se superan los 200 puntos) para ganar.

Para seleccionar las cartas de la mano (las 3 que aparecen en la parte inferior), se pueden usar las teclas ‘A’, ‘S’ y ‘D’. Después de seleccionar una de estas cartas, se pueden seleccionar las cartas de la mesa con ‘Q’, ‘W’, ‘E’ y ‘R’.

Presionando ‘C’ en la pantalla principal se puede iniciar el juego. Al terminar el juego, se pueden presionar ‘N’ y ‘M’ para salir o reiniciar.

Preguntas

1. ¿Cuál era tu objetivo al empezar la prueba?

Mi objetivo al empezar la prueba era tener al menos un sistema de menú y la posibilidad de elegir cartas en una ronda del juego. Nunca había hecho retos de este tipo, pero estoy feliz y orgullosa de lo que he logrado, aunque hay muchas cosas que se deben mejorar y optimizar.

2. ¿Has tenido que recortar algo que tenías pensado hacer?

Desafortunadamente, he tenido que eliminar toda la parte de la IA y el sistema de puntos originales. El juego original está pensado para 2 y 4 jugadores y un sistema de puntos basado en el número y tipo de cartas que cada jugador tiene al final.

3. ¿Qué dificultades te has encontrado y cómo las has resuelto?

Una de las mayores dificultades que he encontrado fueron los problemas con las colisiones y el texto en la pantalla, sin contar el tiempo de trabajo. Como no tuve la posibilidad de contar con una base para construir mis clases para este proyecto, lo solucioné utilizando imágenes y texturas para simular un pequeño sistema de menú. En cuanto al tema de las colisiones, inicialmente pensaba en permitir que se eligieran las cartas con el ratón; ahora las cartas se pueden seleccionar directamente con el teclado, sin necesidad de manejar colisiones.

4. Si tuvieras más tiempo, ¿qué partes de tu código mejorarías y cómo?

La parte que maneja la creación y carga de las barajas de cartas y sus texturas, ya que actualmente tarda exactamente 6 segundos en prepararse todo. Además, limpiaría un poco el código; sé que hay variables que no se utilizan y solo ocupan espacio, o que no están inicializadas, y también me aseguraría de realizar un

`delete` al final de la ejecución del ciclo del juego. También reorganizaría las clases; ahora la estructura es algo confusa, ya que el main llama a `World`, que a su vez llama a otras clases (`Sprite`, `Mazzo`, `Carta` y `Vec2`), y luego hay `Mazzo` que llama a `Carta`, `Sprite` y `Vec2`, y así sucesivamente hasta llegar a `Vec2`. Se pueden organizar para evitar incluir todo en cada archivo y, además, cambiar los atributos necesarios a `private` y agregar `get` y `set` para hacerlo más seguro. Finalmente, sé que hay partes del código que se repiten, por lo que sería mejor crear funciones para esas tareas y mejorar el código para hacerlo escalable a largo plazo.