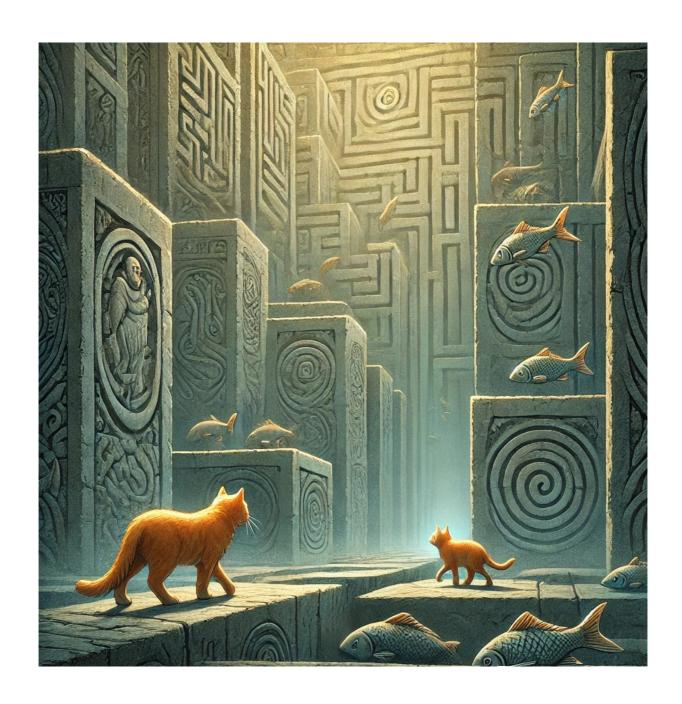
Meow Mania



Lianet Tamarit Tejas C111

INFORME TÉCNICO: MEOW MANIA

1. Introducción

Meow Mania es un videojuego por turnos desarrollado en C# que implementa un sistema de

laberinto generado automáticamente. Se integran los conocimientos de C# y Programación

Orientada a Objetos (POO). El objetivo es que dos jugadores compitan por recolectar la mayor

cantidad de pescados mientras sortean trampas y aprovechan habilidades especiales.

2. Requisitos de Instalación

Para ejecutar correctamente Meow Mania, se deben instalar los siguientes componentes:

- .NET SDK (versión 6.0 o superior): Necesario para compilar y ejecutar programas en C#.

- Editor de Código (opcional): Se recomienda Visual Studio Code o Visual Studio Community para

facilitar la edición del código.

- Consola de comandos: Para ejecutar el programa desde la terminal.

- Biblioteca Spectre. Console: Se usa para mejorar la representación visual en consola. Se puede

instalar con el siguiente comando:

dotnet add package Spectre.Console

3. Funcionalidad del Juego

- Juego para dos jugadores con mecánica de turnos.

- Movimiento con teclas WASD en un entorno de laberinto.

Recolección de pescados para acumular puntos.

- Trampas estratégicamente distribuidas que afectan el rendimiento de los jugadores.

- Sistema de fichas con habilidades únicas, activables durante la partida.

- Condición de victoria basada en la cantidad de puntos acumulados tras recolectar todos los

pescados.

4. Arquitectura del Código

4.1. Juego.cs

- Orquesta la lógica principal del juego.
- Coordina los turnos y define las condiciones de finalización.

4.2. Jugador.cs

- Representa a cada jugador, gestionando su estado y progreso en la partida.

4.3. Laberinto.cs

- Implementa la generación del laberinto mediante el algoritmo de Prim.
- Contiene el método DibujarEnBuffer(), que renderiza el laberinto mediante caracteres ASCII.

4.4. Celda.cs

- Define las propiedades de cada unidad del laberinto, incluyendo su tipo (camino, pared, trampa, pescado).

4.5. Fichas.cs

- Administra la selección de fichas y sus características.

4.6. Trampa.cs

- Modela las trampas y sus efectos adversos:
 - Quitar Puntos.
 - Aumentar el tiempo de reutilización de habilidades.
 - Reducir la velocidad de desplazamiento.

4.7. Habilidad.cs

- Define las capacidades especiales que modifican la estrategia del juego:
 - Duplicar Puntos.
 - Supervelocidad.
 - Atravesar Pared.

- Inmunidad a Trampas.
- Teletransportación.

4.8. ScreenBuffer.cs

- Gestiona la representación visual del juego en consola.

4.9. Program.cs

- Punto de entrada del programa, inicializa la instancia de Juego y ejecuta su ciclo principal.

Fin del Informe