

## Documentación del proyecto 2: Juego Escapa/Cazador

Centro Académico de Alajuela

Joselyn Tatiana Nuñez Miranda - 2025121993

Mariana González Céspedes - 2025077428

### **1. Atributo de Análisis de Problema:**

El proyecto consiste en crear un sistema de navegación inteligente para agentes dentro de un laberinto generado de manera aleatoria. Los agentes toman decisiones de movimiento en tiempo real utilizando cálculos simples y reglas claras, con el objetivo de mantener un equilibrio entre eficiencia, comportamiento lógico y una jugabilidad justa. Como el entorno cambia en cada partida, tanto jugadores como enemigos deben adaptarse continuamente.

Las reglas de movimiento son asimétricas: los cazadores pueden usar lianas y “las presas” pueden usar túneles, lo que genera estrategias distintas y evita comportamientos repetitivos. La lógica de escape se basa en cálculos de distancia, dando mayor peso a la cercanía del peligro mediante un factor multiplicador de 2.5. Esto permite que los agentes reaccionen de manera coherente y no se dirijan hacia zonas riesgosas. La generación del mapa se realiza mediante funciones recursivas que producen un laberinto nuevo en cada juego, y el sistema de puntaje asigna valores según capturas, escapes, trampas activadas y tiempo empleado.

El diseño se desarrolló mediante programación orientada a objetos, utilizando clases como Jugador, Enemigo y Mapa para mantener una estructura modular, clara y fácil de ampliar. Entre las ventajas del sistema se encuentran su eficiencia computacional, la posibilidad de ajustar comportamientos modificando parámetros, la alta rejugabilidad gracias a los laberintos dinámicos y la variedad estratégica entre roles. Las limitaciones incluyen la posibilidad de que algunos agentes queden atrapados, la falta de planificación a largo plazo porque solo se evalúan casillas cercanas y la posibilidad de bloqueos entre múltiples enemigos.

### **2. Atributo de Herramientas de Ingeniería:**

Para implementar el sistema se utilizaron varias herramientas de ingeniería. Pygame se empleó para la visualización, manejo del bucle principal del juego y gestión de eventos. La Programación Orientada a Objetos permitió organizar las entidades de manera clara, encapsulando comportamiento y atributos. Además, se aplicaron operaciones vectoriales y conceptos matemáticos como distancia euclidiana para determinar movimientos óptimos. La generación de laberintos se logró mediante técnicas recursivas, que permiten variedad sin aumentar la complejidad.

La técnica de distancia ponderada se usa en el método que selecciona la dirección del enemigo. La fórmula multiplica la distancia a la amenaza por 2.5 y le resta la distancia a la salida. La casilla con el valor más alto es la elegida, priorizando la seguridad. Durante las pruebas iniciales, un peso menor generaba comportamientos incorrectos: el agente escapaba hacia la salida ignorando amenazas cercanas. El ajuste resolvió el problema sin aplicar algoritmos pesados.

En cuanto al sistema de puntaje, en modo Escapa se parte de 10 000 puntos y se resta tiempo; se suman 500 puntos por enemigo vivo y 50 puntos por enemigo atrapado con una trampa. Si el jugador es atrapado, el puntaje final se convierte en cero automáticamente.

En modo Cazador, cada captura vale 100 puntos y cada enemigo que llega a la salida resta 50 puntos, sin permitir que el puntaje baje de cero. Al finalizar el tiempo, el puntaje se conserva tal cual. Luego se aplica un multiplicador según la dificultad: 0.8 en fácil, 1.0 en normal y 1.2 en difícil. Solo se guardan los cinco puntajes más altos para cada modo, incluyendo nombre, puntos, hora y dificultad.

Este sistema refleja adecuadamente el desempeño del jugador y ajusta el puntaje en función de acciones, tiempo y dificultad, manteniendo coherencia entre los objetivos de cada modo.

### Diagrama de clases:

