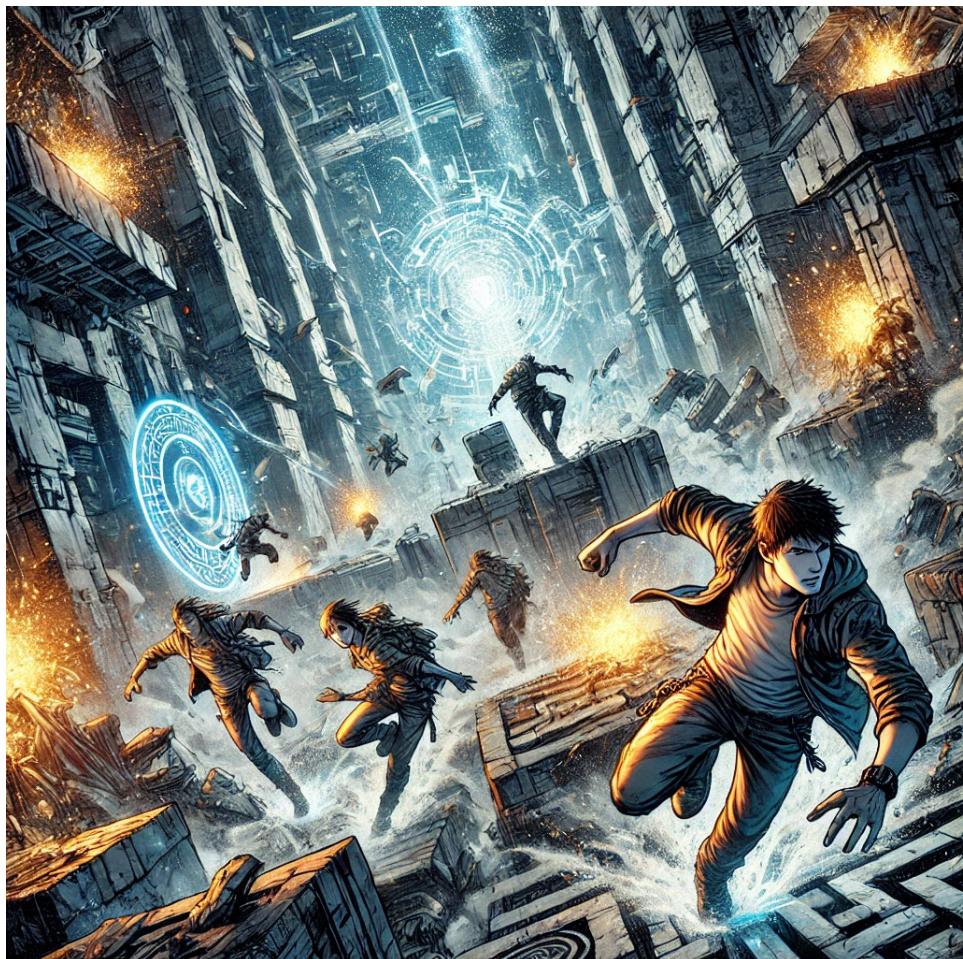


1er proyecto de Programación
Maze Runners
MATCOM - 2024-2025



Introducción

A partir de aquí comienza lo interesante. Su tarea, si decide aceptarla, consiste en crear un juego multijugador con temática de escape de laberinto. Se le darán varias tareas esenciales que definirán el límite mínimo de lo que usted debe implementar. A partir de ahí, puede dejar volar su imaginación para crear un juego único y personal del cual pueda sentirse orgulloso.

Reglas

El juego debe admitir dos o más jugadores por partida. Cada jugador posee una cantidad x de fichas esparcidas por el laberinto.

Tablero

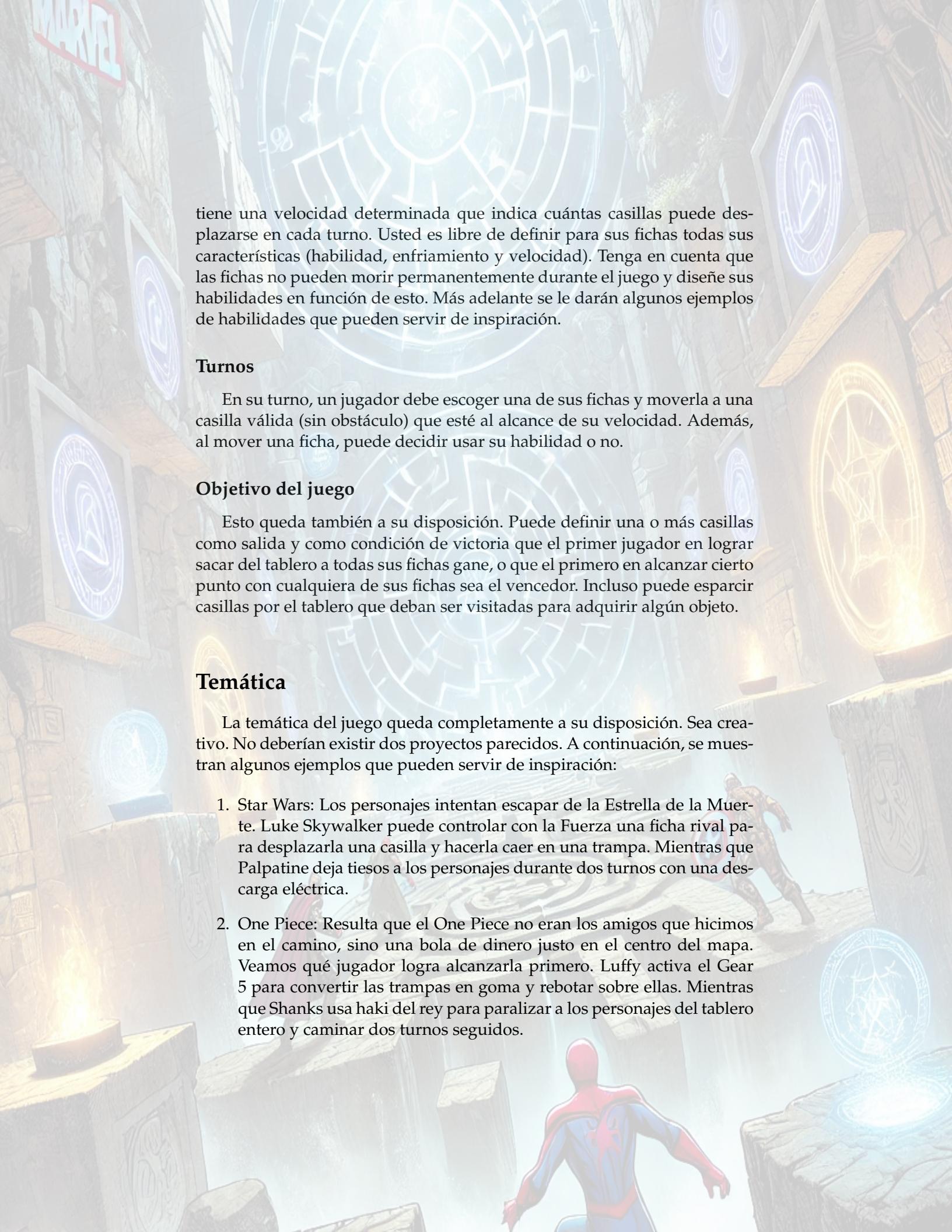
En este juego los tableros se generan dinámicamente. Debe implementar un algoritmo tal que, al iniciar una partida, genere un tablero cuadrado de $n \times n$ casillas (usted elige el tamaño) con objetos esparcidos aleatoriamente. A continuación, se muestran los posibles objetos básicos que deben implementarse; usted es libre de añadir más:

1. Obstáculos: Impiden el paso de las fichas.
2. Trampas: Aplican alguna penalización a la primera ficha que se mueva a esa casilla.

Tenga en cuenta que el tablero debe ser generado de manera que ninguna casilla sea inalcanzable desde ninguna posición. Además, su juego debe tener al menos tres trampas distintas implementadas.

Preparación

Debe contar con un conjunto de al menos 5 fichas disponibles para que los jugadores las escojan. Puede decidir si cada jugador puede escoger una o más fichas para sus partidas, pero todos los jugadores deben tener la misma cantidad de fichas. Cada ficha disponible debe tener una habilidad definida y un tiempo de enfriamiento. El tiempo de enfriamiento impide el uso de esa habilidad durante los siguientes t turnos. Además, cada ficha



tiene una velocidad determinada que indica cuántas casillas puede desplazarse en cada turno. Usted es libre de definir para sus fichas todas sus características (habilidad, enfriamiento y velocidad). Tenga en cuenta que las fichas no pueden morir permanentemente durante el juego y diseñe sus habilidades en función de esto. Más adelante se le darán algunos ejemplos de habilidades que pueden servir de inspiración.

Turnos

En su turno, un jugador debe escoger una de sus fichas y moverla a una casilla válida (sin obstáculo) que esté al alcance de su velocidad. Además, al mover una ficha, puede decidir usar su habilidad o no.

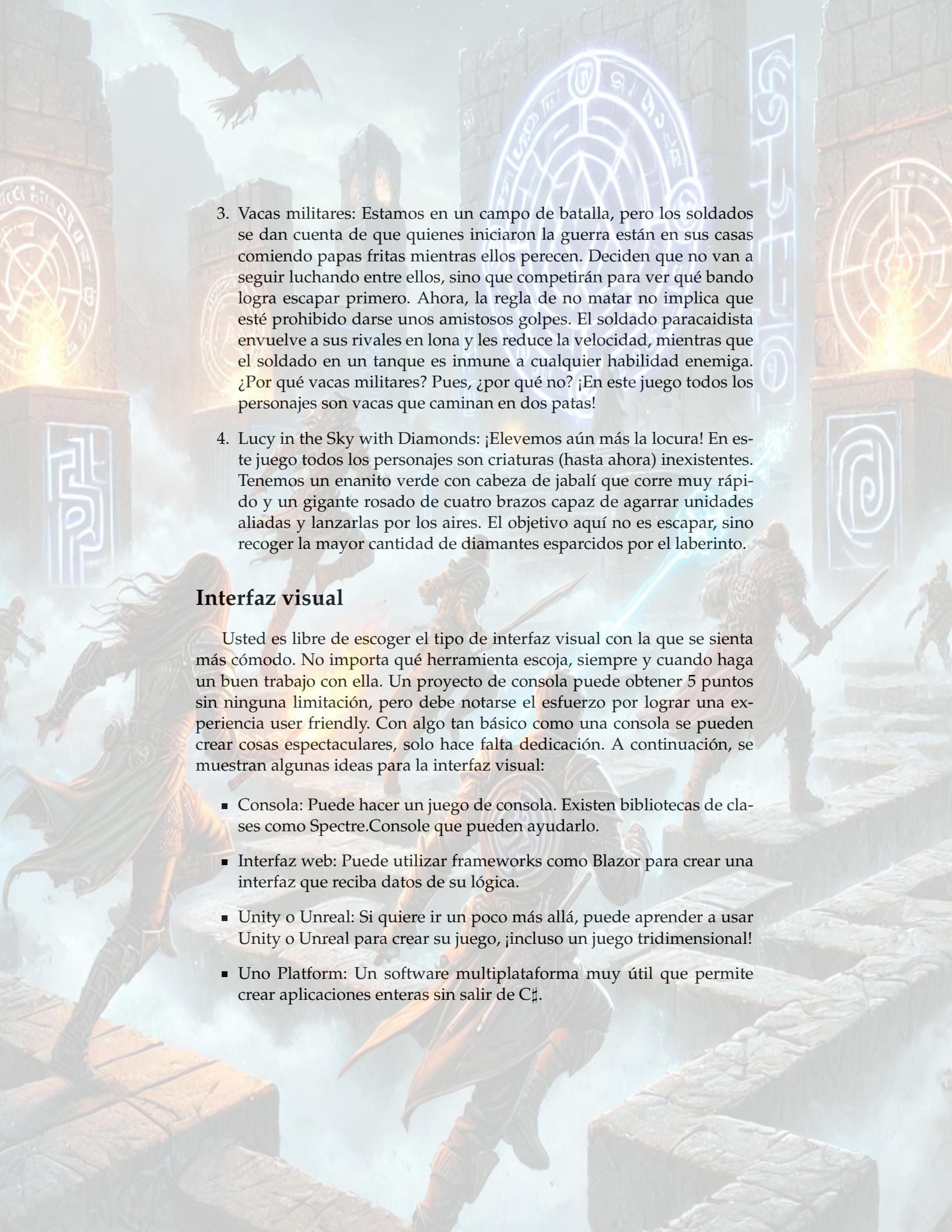
Objetivo del juego

Esto queda también a su disposición. Puede definir una o más casillas como salida y como condición de victoria que el primer jugador en lograr sacar del tablero a todas sus fichas gane, o que el primero en alcanzar cierto punto con cualquiera de sus fichas sea el vencedor. Incluso puede esparcir casillas por el tablero que deban ser visitadas para adquirir algún objeto.

Temática

La temática del juego queda completamente a su disposición. Sea creativo. No deberían existir dos proyectos parecidos. A continuación, se muestran algunos ejemplos que pueden servir de inspiración:

1. Star Wars: Los personajes intentan escapar de la Estrella de la Muerte. Luke Skywalker puede controlar con la Fuerza una ficha rival para desplazarla una casilla y hacerla caer en una trampa. Mientras que Palpatine deja tisos a los personajes durante dos turnos con una descarga eléctrica.
2. One Piece: Resulta que el One Piece no eran los amigos que hicimos en el camino, sino una bola de dinero justo en el centro del mapa. Veamos qué jugador logra alcanzarla primero. Luffy activa el Gear 5 para convertir las trampas en goma y rebotar sobre ellas. Mientras que Shanks usa haki del rey para paralizar a los personajes del tablero entero y caminar dos turnos seguidos.

- 
3. Vacas militares: Estamos en un campo de batalla, pero los soldados se dan cuenta de que quienes iniciaron la guerra están en sus casas comiendo papas fritas mientras ellos perecen. Deciden que no van a seguir luchando entre ellos, sino que competirán para ver qué bando logra escapar primero. Ahora, la regla de no matar no implica que esté prohibido darse unos amistosos golpes. El soldado paracaidista envuelve a sus rivales en lona y les reduce la velocidad, mientras que el soldado en un tanque es inmune a cualquier habilidad enemiga. ¿Por qué vacas militares? Pues, ¿por qué no? ¡En este juego todos los personajes son vacas que caminan en dos patas!
 4. Lucy in the Sky with Diamonds: ¡Elevemos aún más la locura! En este juego todos los personajes son criaturas (hasta ahora) inexistentes. Tenemos un enanito verde con cabeza de jabalí que corre muy rápido y un gigante rosado de cuatro brazos capaz de agarrar unidades aliadas y lanzarlas por los aires. El objetivo aquí no es escapar, sino recoger la mayor cantidad de diamantes esparcidos por el laberinto.

Interfaz visual

Usted es libre de escoger el tipo de interfaz visual con la que se sienta más cómodo. No importa qué herramienta escoja, siempre y cuando haga un buen trabajo con ella. Un proyecto de consola puede obtener 5 puntos sin ninguna limitación, pero debe notarse el esfuerzo por lograr una experiencia user friendly. Con algo tan básico como una consola se pueden crear cosas espectaculares, solo hace falta dedicación. A continuación, se muestran algunas ideas para la interfaz visual:

- Consola: Puede hacer un juego de consola. Existen bibliotecas de clases como Spectre.Console que pueden ayudarlo.
- Interfaz web: Puede utilizar frameworks como Blazor para crear una interfaz que reciba datos de su lógica.
- Unity o Unreal: Si quiere ir un poco más allá, puede aprender a usar Unity o Unreal para crear su juego, ¡incluso un juego tridimensional!
- Uno Platform: Un software multiplataforma muy útil que permite crear aplicaciones enteras sin salir de C#.

Evaluación

Los requisitos mínimos para aprobar el proyecto ya se mencionaron en este documento. A continuación, se ofrece un resumen:

1. Tablero generado aleatoriamente.
2. Obstáculos.
3. Al menos tres trampas distintas implementadas.
4. Todas las casillas deben ser alcanzables desde cualquier posición.
5. Más de 5 fichas para escoger.
6. Todas las fichas deben tener una habilidad, tiempo de enfriamiento y velocidad.
7. Debe existir una condición de victoria.
8. Debe existir una interfaz visual que permita jugar.
9. El juego debe iniciar, ejecutarse, y detenerse cuando algún jugador cumpla con la condición de victoria.

Una vez implementados los requisitos mínimos, es posible obtener la nota mínima. Si desea ir más allá, puede añadir a su proyecto cualquier cosa que se le ocurra, siempre y cuando no entre en conflicto con los requisitos básicos. A continuación, se muestran algunas ideas:

1. Interfaz visual bien trabajada y agradable de jugar.
2. Jugador virtual: puede implementar una inteligencia artificial que juegue por su cuenta.
3. Variedad de fichas y trampas.
4. Algoritmo inteligente de generación de tablero, con coherencia. Trampas inevitables en pasillos, laberintos grandes y difíciles de descifrar, posiciones iniciales estratégicas de las fichas.
5. Casillas con beneficios y ayudas que las fichas pueden utilizar.
6. Armas que las fichas puedan recoger.
7. Portales de teletransportación.
8. Fichas NPC que obstaculicen a los jugadores.

Entrega y Git

Un requisito para este proyecto es tenerlo guardado en un repositorio en la plataforma GitHub. Tenga en cuenta que tener un buen historial de commits será beneficioso en su evaluación. Además, en caso de sufrir algún accidente durante el desarrollo del proyecto (un virus, se daña su computadora), su proyecto estará a salvo en una plataforma y su profesor podrá ver el progreso realizado.

El día de la entrega, se pedirá exclusivamente el enlace de su repositorio. No se aceptarán otros tipos de entrega fuera de casos muy específicos. Se recomienda no dejar el aprendizaje de Git y la creación del repositorio para la noche antes de la entrega.