# Spellbound Forest: The legacy of the Stone

1. Descripción:

Queremos implementar un videojuego sobre una **búsqueda del tesoro**.

El jugador es un hechicero que debe recorrer un bosque embrujado para llegar al tesoro, que es la piedra de la resurrección. Tenemos que nuestro hechicero tiene 3 posibles hechizos con los que defenderse de los peligrosos monstruos que aparecen por el bosque:

* Incendio. Lanza fuego al contrincante. (-10 de vida al monstruo)
* Petrificus Totalus. Paraliza al monstruo y le congela durante cierto tiempo (-5 vida al monstruo)
* Avada Kedavra. Mata, que es un hechizo que solo puede realizar cada cierto tiempo (20 segundos x ejemplo, y mata de manera definitiva al monstruo)

Y los monstruos que viven en el bosque son:

* Dwarf: lanza maldiciones y quita la vida (-5 vida del hechicero)
* Werewolf: solo ataca mordiendo (-10 vida del hechicero)
* Spirit: intenta robarle el alma al hechicero (-15 vida del hechicero)

El hechicero tiene a su servicio un mapa del tesoro que le indica su posición actual y la posición del tesoro, pero no le indica dónde están los monstruos, por lo que esto causa expectación. Este mapa se genera de forma aleatoria cada vez que comienza el juego, junto al paisaje que contiene elementos decorativos como árboles, arbustos y un lago.

1. Requisitos:

* El menú principal permitirá elegir entre jugador/multijugador (Tal vez también entre nivel de dificultad. Podríamos generar más monstruos más frecuentemente 🡪Si da tiempo.)
* Loop del juego: el hechicero o muere o llega al tesoro. En cualquier caso, se le muestra un menú al usuario para que decida si quiere volver a empezar o salir.

Play: el jugador se encuentra recorriendo el mapa y en búsqueda activa del tesoro

*Win state*: encuentra el tesoro

*Fail state*: le matan

*Play again*: opción en el menú para volver a jugar

* Sistema de puntuación: vida del hechicero. Se va a implementar también un reloj que controle el tiempo que ha tardado el jugador el encontrar el tesoro. (De cara al futuro, podemos ver si ponemos un tiempo específico para encontrar el tesoro, si no llega a tiempo muere y vuelve a empezar)
* Oponentes de IA programada: diseña el comportamiento de cada monstruo.
* Se destruyen y crean los componentes de los monstruos, el mapa, el mundo y el hechicero.
* El hechicero tiene animaciones como estar en reposo o en guardia con su varita. Los monstruos en reposo o atacando en función de su ataque. (Por ejemplo, el monstruo werewolf puede estar en reposo o mordiendo)
* En el caso multijugador tendríamos una división de pantalla. El objetivo del juego no se ve modificado, con el mismo tesoro y mundo. Deben ver quien llega antes al tesoro.

1. Patrones:

* Facade para inicializar el mundo, administrar jugadores y la creación y destrucción de elementos.
* Singleton para un único mapa.
* Factoría para los monstruos y los hechizos.
* Máquina de estados para el hechicero por ejemplo (reposo, defendiendo, muerto y ganador) y para los monstruos (reposo, atacando, congelado, muerto).
* Bridge para construir los extras (árboles, lagos…) de forma que tengamos un par de diseños y podamos variar su color y posición al inicializar el juego.
* Command para gestionar los inputs del jugador a través del teclado y redireccionarlos a efectos en el juego.

1. Historias de usuario:

1) Como hechicero quiero encontrar el tesoro para ganar.

2) Como hechicero quiero poder lanzar mis hechizos para defenderme de los monstruos.

3) Como hechicero, quiero tener un sistema de puntuación basado en mi vida (o en el tiempo) para ver como estoy progresando en la partida.

4) Como hechicero, quiero que el juego me permita lanzar mis hechizos con intervalos de tiempo específicos, especialmente para hechizos poderosos como *Avada Kedavra*.

5) Como monstruo quiero atacar para matar al hechicero.

6) Como monstruo quiero encontrar al hechicero para perseguirle.

7) Como jugador, quiero elegir entre jugar en modo individual o multijugador desde el menú principal para adaptar la experiencia a mi preferencia.

8) Como jugador, quiero que el juego me dé la opción de reiniciar la partida o salir tras morir o ganar, para decidir si quiero volver a intentarlo.

9) Como jugador, quiero que el mapa del tesoro y el mundo se generen aleatoriamente en cada partida para que la experiencia sea única cada vez.

10) Como jugador, quiero ver animaciones en mi hechicero y en los monstruos (reposo, guardia, ataque) para que el juego sea más inmersivo.

11) Como jugador en modo multijugador, quiero una pantalla dividida para competir con otro hechicero y ver quién alcanza primero el tesoro.

1. Análisis del objetivo, extracción de requisitos y extracción de tareas a realizar:
2. **Objetivo**: ganar

**Requisitos**: encontrar el tesoro

**Tareas**: ver el mapa del tesoro, detectar las pulsaciones de tecla para traducirlo en movimientos del hechicero. Detectar la colisión con el tesoro.

1. **Objetivo**: defenderse de los monstruos/no morir

**Requisitos**: poder lanzar hechizos

**Tareas**: ver el monstruo, ver las animaciones de los hechizos y de los ataques del monstruo. Estar en el rango (dentro del área colisión que detecta el monstruo) para poder lanzarlo.

Detectar pulsaciones de tecla para lanzar los hechizos.

Ver como se modifica en tiempo real mi vida y la de los monstruos (una barra de vida que disminuye su tamaño).

1. **Objetivo**: ver la vida

**Requisitos**: estar vivo

**Tareas**: diseño de la barra de vida, la actualización de la barra de vida según reciba ataques (está por decidir si la vida se puede ir recuperando si no recibe daño durante algún tiempo)

Mostrar al final de la partida la vida.

1. **Objetivo**: lanzar hechizos poderosos con “cooldown”

**Requisitos**: que haya pasado el tiempo determinado para poder volver a lanzar el hechizo

**Tarea**: diseño del control del tiempo, mostrar los segundos que le quedan restantes para lanzar el hechizo en algún lugar de la interfaz (está por diseñar).

Saber que hechizo se lanza en cada momento para saber si tiene “cooldown” o no. (2 no tienen y 1 sí)

Que se van las animaciones/efectos de los hechizos.

1. **Objetivo**: atacar para matar al hechicero

**Requisitos**: vivir y encontrar al hechicero

**Tareas**: con el agente de IA el monstruo se va a poder mover por el espacio. Con un objeto colisión el monstruo puede saber si se ha encontrado al jugador o no y atacarle hasta matarle o morir.

Animaciones de los ataques del monstruo.

El monstruo persigue al jugador siempre que esté en la zona de colisión. Si el jugador consigue huir de esa zona el monstruo no será capaz de detectar al jugador. (decidir las velocidades de los monstruos, probablemente serán menores a la del jugador para que sea fácil escapar)

1. **Objetivo**: perseguir al hechicero

**Requisitos**: detectar al hechicero y conocer su posición

**Tareas**: si el monstruo se detecta, que se mueva hacia su posición. Si no le detecta se mueve aleatoriamente según el agente de IA. A partir de cierta distancia lanza su ataque cada cierto tiempo para dañar la vida del hechicero. (Diseñar dos objetos colisión para ver si la distancia es suficiente para atacar y otra para detectar al hechicero)

1. **Objetivo**: adaptar la experiencia a la preferencia del jugador/mejorar UX

**Requisitos**: dar la opción a elegir entre modo individual y jugador

**Tareas**: si se elige individual representar en la pantalla todo el juego, si es multijugador hacer las divisiones de pantallas necesarias (de momento 2 al hacerlo en local), y un diseño de un menú interactivo.

1. **Objetivo**: poder salir en cualquier momento (¿deberíamos guardar el proceso?) o reiniciar la partida tras morir o ganar.

**Requisitos**: dar las opciones a elegir en cada momento

**Tareas**: otro menú interactivo con estas opciones y estar disponible durante la partida, en el caso de querer salir.

1. **Objetivo**: experiencia única

**Requisitos**: tener un diseño para generar el mundo y el mapa de forma aleatoria

**Tareas**: generar aleatoriamente el mundo y el mapa del tesoro en cada partida

1. **Objetivo**: mejorar la UX

**Requisitos**: diseño de distintas animaciones tanto para los monstruos como para el hechicero

**Tareas**: coordinar los eventos y efectos con las animaciones visuales en cada momento.

1. **Objetivo**: definir el “win state” en caso de multijugador

**Requisitos**: estar en modo multijugador

**Tareas**: alterar el comportamiento del juego para que ahora se trate de comparar quién llega antes al tesoro, como una competición.