**Requerimientos funcionales**

Con el objetivo de contribuir a las fuentes de entretenimiento de la comunidad, los estudiantes del grupo 3 del curso Algoritmos y Programación II, de la Universidad Icesi, han decidido crear un juego que emule un juego de mesa.

Tal juego trata de una aventura entre Rick y Morty, en donde están en busca de recolectar las súper semillas que crecen en los súper árboles y que ayudarán a Rick a seguir con su investigación acerca de la bomba de neutrinos.

Estas semillas serán esparcidas, aleatoriamente, por un tablero conectado por portales que Rick ha creado con el fin de acelerar la recolección. Así pues, el juego funcionará por turnos (para Rick y para Morty).

Con este contexto, el programa debe estar en la capacidad de:

R1. Crear el tablero de juego.

R1.1 Recibir unas medidas digitadas por el jugador.

R1.2 Evitar la creación de un tablero con 0 casillas (si el jugador digita 0 en una de las medidas).

R1.3 Evitar la creación de un tablero con una sola casilla.

R1.4 Crear el tablero en una estructura circular doblemente enlazada.

R2. Mostrar el tablero de juego.

R2.1 Representar el tablero de juego de una manera visual.

R2.2 Imprimir el tablero en forma de culebrita.

R2.3 Mostrar las ubicaciones de los portales si el jugador así lo desea.

R2.4 Mostrar las casillas con su posición en caso de que no tengan un jugador o una semilla.

R2.5 Si hay un jugador en una casilla, esta deberá imprimirse con la letra que represente al jugador en lugar de su posición.

R2.6 Si hay una semilla en una casilla, esta deberá imprimir una representación de la semilla (\*) en lugar de su posición.

R3. Posicionar a los jugadores al azar, en el tablero.

R4. Crear las semillas.

R4.1 Recibir la cantidad de semillas, digitada por el jugador, y crearlas.

R4.2 Evitar que se creen más semillas que casillas.

R4.3 Distribuir las semillas, al azar, dentro del tablero.

R4.4 Evitar que haya más de una semilla en una casilla.

R5. Crear los portales.

R5.1 Recibir la cantidad de portales, digitada por el jugador, y crearlos.

R5.2 Evitar que la cantidad de portales creados sea igual, o mayor, a la mitad de las casillas existentes.

R5.3 Distribuir los portales, al azar, dentro del tablero.

R5.4 Conectar pares casillas, para ser los portales, de manera aleatoria.

R5.5 Evitar que una casilla esté conectada a más de una casilla.

R6. Mover a los jugadores.

R6.1 Determinar qué jugador tiene el turno actual y dejar que tire un dado.

R6.2 Mover al jugador un número de casillas igual al resultado del dado.

R6.3 Mover al jugador hacia adelante o hacia atrás según este lo elija.

R6.4 Si hay un portal en la casilla en la que cae el jugador que se ha movido, este deberá ser teletransportado a la casilla a la que lleve tal portal.

R7. Recolectar las semillas.

R7.1 Si hay una semilla en la casilla a la que se ha movido un jugador, este deberá recolectarla.

R7.2 Llevar la cuenta de las semillas recolectadas por los jugadores de cada partida.

R7.3 Detener la partida cuando se hayan recolectado todas las semillas que se habían creado para esta.

R8. Calcular el puntaje.

R8.1 Calcular el puntaje del jugador que haya recolectado más semillas en la partida con la fórmula:

R8.2 Almacenar los puntajes de los ganadores.

R8.3 Almacenar los puntajes de los ganadores de manera persistente.

R8.4 Si el jugador ganador ya tenía un puntaje registrado, el nuevo puntaje deberá sumarse al anterior.

R8.5 Si el jugador ganador no tenía un puntaje registrado, deberá registrarse con el puntaje obtenido.

R8.6 Mostrar el top 5 de los mejores puntajes al finalizar la partida.

R9. Registrar jugadores.

R9.1 Registrar jugadores con un username.

R9.2 Asignar a los jugadores si serán Rick o Morty.

R10. Rotar turnos.

R10.1 Otorgar el primer turno al jugador que sea Rick.

R10.2 Una vez que el jugador se mueva, el turno deberá pasar al otro.

R11. Mostrar el marcador de la partida cuando el jugador lo quiera.