

DISEÑO

Diagrama de flujo del funcionamiento programa:

Inicio del Juego:

1. El juego comienza con una pantalla de inicio que muestra el título y las opciones para empezar a jugar o salir del juego.

Selección de Nivel:

2. Si el jugador elige comenzar el juego, se le presenta una pantalla de selección de nivel, donde puede elegir entre los tres niveles: la casa encantada, la nave espacial y el laboratorio en el espacio

Carga del Nivel:

3. Una vez que el jugador selecciona un nivel, se carga el nivel correspondiente, con la configuración de enemigos, objetos y la ubicación inicial de Rick.

Jugando el Nivel:

4. El jugador controla a Rick y puede moverlo dentro del nivel para avanzar, esquivar enemigos y recoger objetos como vidas adicionales o el suero necesario.

Colisiones y Eventos:

5. El juego detecta colisiones entre Rick y los enemigos o con los rayos de las armas, así como la recolección de objetos.
6. Si Rick choca con un enemigo, pierde un poco de vida (media vida de las 5 posibles). Si se queda sin vidas, el juego termina.
7. Si Rick recoge vidas del piso, y perdió algún corazón, este se llena. en caso de que esté lleno no pasa nada (ya que no es posible cargar vidas extras a demas de las 5 establecidas).

Niveles:

Nivel 1:

8. En el Nivel 1 (Esquivar los somnbulos), Rick debe esquivar los obstaculos y no dejarse alcanzar por los somnambuleros que lo persiguen por la casa, el nivel acaba cuando rick llega a la puerta principal.

Nivel 2:

9. En el Nivel 2 (la nave espacial), Rick debe esquivar asteroides y recoger vidas adicionales para mantenerse con vida, esto presentara un gran desafio ya que los asteroides tiene movimientos completamente distintos a los demas.

Nivel 3:

10. En el Nivel 3 (El Laboratorio Espacial del Suero), Rick se enfrenta al jefe Jerry nocturno, que posee poderes espaciales y armas de super ataques; Rick no se debe dejar alcanzar por algun ataque del enemigo ya que esto representara una perdida de vida importante, ya que el jefe Jerry es mucho mas fuerte que rick (puede quitar $1 + \frac{1}{2}$ de vida!) y debe derrotarlo para obtener el suero (quitarle todas la vidas a Jerry).

Victoria:

11. Si Rick derrota al jefe en el Nivel 3 y recoge el suero, avanza a la pantalla de victoria, donde Morty ya está despierto.

Derrota:

12. Si Rick se queda sin vidas en cualquier nivel, o sin tiempo, ya que para evitar que los jugadores se queden estancados en un nivel se cuenta con este para determinar un limite de jugabilidad en cada nivel; pasado este tiempo o que el jugador ya no tenga vidas se muestra la pantalla de derrota y el jugador puede optar por reiniciar el nivel o salir del juego.

Fin del Juego:

13. El juego permite al jugador volver al menú principal desde cualquier pantalla de victoria o derrota.

14. El jugador puede optar por volver a jugar otro nivel o salir del juego por completo.

Estructura de datos:

Niveles:

Utiliza una lista (o vector) para mantener un registro de los niveles del juego. Cada elemento de la lista representa un nivel y contiene información como la configuración del nivel, la posición inicial de los objetos/enemigos y los criterios de finalización. Por ejemplo, podrías tener una lista de objetos de tipo Nivel.

Jugador (Rick):

Podemos utilizar una estructura o clase que almacene la información del jugador, como la posición actual, la cantidad de vidas, la puntuación y otros datos relacionados. Esta estructura se actualiza a medida que el jugador progresa a través del juego.

Enemigos y Objetos Recolectables:

Crear listas separadas (o vectores) para mantener un registro de los enemigos y objetos recolectables presentes en el nivel actual. Cada elemento en estas listas puede ser una instancia de una clase que represente a un enemigo u objeto específico. Por ejemplo, podríamos tener una lista de objetos de tipo Enemigo y una lista de objetos de tipo ObjetoRecolectable.

Puntuación y Vidas:

Utiliza variables simples para llevar un registro de la puntuación y la cantidad de vidas del jugador. No es necesario una estructura de datos compleja para estos valores.

Tiempo:

Utilizar una variable para llevar un registro del tiempo transcurrido en el nivel.

Clases que podríamos implementar:

1. Clase Juego (Game): Esta clase controla el flujo general del juego, como la transición entre niveles, la gestión de eventos y la detección de victoria o derrota.
2. Clase Personaje (Personaje): La clase personaje es el conjunto de dos entes importantes en nuestro juego, es una clase que tiene ciertos atributos y funciones que se les pueden dar estos dos objetos (Player y Enemys):
 - a. Objeto Jugador (Player): Representa al personaje controlado por el jugador, Rick. Contiene información como la posición del jugador, la cantidad de vidas, la puntuación y métodos para mover al jugador y responder a eventos.
 - b. Objeto Enemigo (Enemy): Esta clase representa a los enemigos en el juego, como los sonámbulos en el Nivel 1 y el jefe Jerry nocturno en el Nivel 3. Debe incluir atributos como la posición del enemigo, la vida, la velocidad y métodos para mover y atacar.
3. Clase objeto (Object): Esta clase representa los diferentes objetos que se pueden encontrar en los diferentes niveles, como los obstáculos del primer nivel, los asteroides del segundo nivel y los items recolectables como las vidas, y el suero del nivel 3:
 - a. Objeto Recolectable (CollectibleObject): Representa objetos recolectables como vidas adicionales y el suero en el Nivel 3. Incluye atributos como la posición del objeto y métodos para recogerlo.
 - b. Objeto Obstáculo (ObstacleObject): Representa los obstáculos del primer nivel (cosas que se pueden encontrar en la casa) y los asteroides del segundo nivel, incluye atributos como la posición del objeto, si son estáticos o si tiene algún movimiento/desplazamiento por la interfaz del nivel.

4. Clase Controlador de Eventos (EventController): Administra los eventos del juego, como las entradas del jugador y las colisiones, y conecta estas señales a las ranuras (métodos) en las clases correspondientes.
5. Clase Interfaz de Usuario (UserInterface): Si tu juego tiene elementos de interfaz de usuario, como barras de vida, puntajes, y botones, una clase de interfaz de usuario puede ayudar a gestionar y mostrar estos elementos de manera eficiente.
6. Clase Temporizador (Timer): Podemos usar una clase de temporizador para controlar el tiempo en el juego, como el tiempo restante en cada nivel o el tiempo entre eventos específicos.
7. Clase nivel (Level): Una instancia de esta clase representa un nivel específico del juego. Contendría la configuración del nivel, la información sobre enemigos y objetos, y los criterios de finalización del nivel.

Interacción

Nivel 1 - La Casa Encantada:

El entorno en este nivel consta de una serie de habitaciones, pasillos y objetos dentro de la casa encantada. Rick se mueve utilizando controles de teclado (awsd), y la detección de colisiones se utiliza para asegurarse de que no atraviese las paredes ni los obstáculos.

Al entrar en contacto con objetos o elementos en el entorno, como puertas o interruptores, se activan eventos. Por ejemplo, abrir una puerta podría desencadenar la generación de un enemigo o revelar un objeto recolectable. Rick puede recolectar objetos como "Vidas Extra" que están dispersas en el nivel. Cuando colisiona con estos objetos, se actualiza su cantidad de vidas.

Los sonámbulos (enemigos) se generan en ubicaciones predefinidas y se mueven de manera autónoma a lo largo de un camino establecido, atacando a Rick si lo ven. La detección de colisiones se utiliza para verificar si Rick y un sonámbulo colisionan.

Nivel 2 - La Nave Espacial en Peligro:

El entorno en este nivel representa una nave espacial en medio de un campo de asteroides.

Rick controla la nave espacial, y la interacción con el entorno se enfoca en el movimiento y la colisión con asteroides.

El movimiento de la nave espacial se controla mediante las teclas de dirección o la pantalla táctil. La física del juego puede aplicarse para simular la aceleración y la gravedad en el espacio.

Los asteroides se generan de manera aleatoria y se mueven hacia la nave espacial. La detección de colisiones se utiliza para determinar cuándo un asteroide colisiona con la nave espacial y cuánto daño causa, cada colisión reduce la vida de Rick.

Nivel 3 - El Laboratorio Espacial del Suero:

El entorno en este nivel representa un laboratorio espacial con el jefe final, Jerry nocturno, como enemigo principal.

La interacción se centra en el enfrentamiento con Jerry nocturno y la recolección del suero.

Rick puede moverse alrededor del laboratorio para esquivar los ataques de Jerry nocturno y dispararle. La detección de colisiones se utiliza para determinar si los disparos de Rick impactan en Jerry.

Jerry nocturno tiene una inteligencia artificial que le permite perseguir y atacar a Rick. La detección de colisiones se utiliza para verificar si Jerry colisiona con Rick y cuánto daño inflige.

La recolección del suero es una interacción clave. Rick debe recoger el suero al acercarse a él, y una vez que lo hace, se desencadena un evento de victoria y se avanza al siguiente nivel o se muestra la pantalla de victoria.

Interacción de clases:

Juego (Game):

Controla el flujo general del juego y coordina la interacción entre los diferentes elementos.

Administra la transición entre niveles, la detección de victoria o derrota y muestra las pantallas de victoria o derrota.

Utiliza instancias de las clases Nivel y Jugador para iniciar y gestionar los niveles y al jugador.

Nivel (Level):

Representa un nivel específico del juego y contiene información sobre la configuración del nivel, como la disposición de enemigos y objetos recolectables.

Inicia el nivel y controla la lógica específica de ese nivel, incluyendo la generación de enemigos y objetos.

Puede acceder al jugador (Jugador) para verificar la interacción con elementos del nivel y las condiciones de finalización del nivel.

Personaje: Representa a los distintos entes que tiene enfrentamientos entre si:

Jugador (Player):

Representa al personaje controlado por el jugador, como Rick.

Responde a las entradas del jugador para moverse y ejecutar acciones específicas.

Administra la puntuación, las vidas y otras estadísticas del jugador.

Interactúa con objetos recolectables y enemigos, y responde a eventos, como perder vidas en caso de colisión con enemigos.

Enemigo (Enemy) y Objeto Recolectable (CollectibleObject):

Representan enemigos y objetos recolectables en el nivel.

Responden a las acciones del jugador, como colisiones o disparos.

En el caso de los enemigos, tienen lógica de movimiento y ataque.

En el caso de los objetos recolectables, al recogerlos, el jugador obtiene beneficios, como vidas adicionales o el suero en el Nivel 3.