**Concepto:** Es un juego que te sumerge en las hilarantes y caóticas experiencias de Rick y Morty y sus aventuras Interdimensionales, tomando como base el episodio 7 de la temporada 1, titulado "Raising Gazorpazorp ". Los jugadores asumen los roles de Rick y Morty mientras exploran diferentes dimensiones, resuelven acertijos y enfrentan desafíos únicos.

**Descripción:** La jugabilidad del juego se desarrolla en 2D con elementos de plataformas y resolución de acertijos. Los jugadores pueden elegir entre Rick y Morty para aprovechar sus habilidades únicas en tres escenarios emocionantes basados en el episodio, el juego en el que nos basamos para la vista y la jugabilidad es fireboy and water girl, y el enfoque sera algo similar:

* Escenario 1: El Mercado Interdimensional

En el primer escenario, los jugadores aterrizan en un mercado interdimensional alienígena que presenta un caos surrealista. Rick y Morty deben recopilar los ingredientes necesarios para una poción que les permitirá avanzar. La tarea es encontrar todos los ingredientes que en este escenario estarán distribuidas en un laberinto tipo pac-man, antes de que un contador de tiempo llegue a cero, lo que resultaría en la captura de los personajes por los habitantes del mercado.

* Escenario 2: La Aventura en Gazorpazorp

Una vez que Rick y Morty obtienen la poción, son transportados a Gazorpazorp, una sociedad dominada por mujeres. En este escenario, deben navegar por la ciudad, evitando situaciones cómicas y desafíos culturales mientras buscan una manera de volver a casa. La misión consiste en escapar sorteando a través de obstáculos en el entorno y enemigos que quieren impedir llegar al portal para escapar de las mujeres de Gazorpazorp y permitir que Rick y Morty regresen a su dimensión.

* Escenario 3: Regreso a Casa

Después de una serie de situaciones hilarantes y eventos inesperados en Gazorpazorp, Rick y Morty finalmente encuentran una forma de regresar a casa. Sin embargo, deben enfrentar un último desafío: luchar contra un jefe final, una versión gigante y enojada de una Gazorpian, que se interpone en su camino de vuelta a casa. Los jugadores deben usar las habilidades y objetos qu estran presentes en el escenario del juego para derrotar al jefe y salvar su dimension de la destrucción a manos de el bebe de morty ya crecido y enojado.

# **MODELAMIENTO Y CLASES**

* Modelamiento:

El juego "Rick and Morty: Aventuras Interdimensionales" tendrá una estructura de desarrollo que consta de la creación de un menú inicial para la selección de personajes y la elección de escenario. Después de esto, se generará un mapa en 2D que representará las dimensiones interdimensionales que los jugadores explorarán. El mapa estará compuesto por rectángulos que servirán como paredes y obstáculos que los personajes deberán sortear.

Cada personaje (Rick y Morty) tendrá atributos como velocidad, ataque, vida y una posición inicial en la parte inferior izquierda del mapa. Además, se les asignarán diferentes acciones en el juego, como moverse hacia la izquierda y la derecha, realizar ataques con proyectiles y saltar, que se regirá por las leyes de la gravedad.

Para la generación de ataques dirigidos hacia los personajes, se implementarán modelos físicos que simularán el lanzamiento de proyectiles con trayectorias parabólicas y el lanzamiento de bolas que seguirán trayectorias lineales, y a lo mejor otros objetos que seguirán trayectorias elásticas.

* Clases:
* Clase Jugador: Representa al personaje jugable (Rick o Morty). Contiene atributos de velocidad, vida y posición. Tiene métodos para moverse y saltar.
* Clase Obstáculo: Usada para representar obstáculos y paredes en los escenarios 1 y 2. Los objetos de esta clase actúan como barreras.
* Clase Objeto de Recolección: Utilizada en el escenario 1 para representar los ingredientes necesarios para la poción. Los jugadores deben recolectar estos objetos.
* Clase Enemigo: En el escenario 2, esta clase representa a los enemigos que intentan impedir que Rick y Morty lleguen al portal. Contiene atributos de velocidad y métodos de movimiento.
* Clase Portal: En el escenario 2, este objeto representa el portal que Rick y Morty deben alcanzar para completar la misión.
* Clase Jefe Final: Utilizada en el escenario 3 para representar al jefe final (bebé de Morty crecido y enojado). Esta clase contendrá atributos de vida y métodos de ataque.
* Clase Tiempo: Se utiliza en el escenario 1 para llevar un seguimiento del tiempo restante antes de que los personajes sean capturados. Esta clase gestiona la mecánica de tiempo del juego.
* Clase Menú Inicial: Para mostrar el menú de inicio del juego, permitiendo a los jugadores seleccionar sus personajes antes de comenzar la aventura.
* Clase iniciar Sección: registrar usuarios en documento

**Clase Personaje:**

Atributos:

Velocidad: velocidad de movimiento en la pantalla.

Salto: capacidad de alcance al realizar un salto.

Vida: cantidad de resistencia a ataques.

PosX: posición en el eje X.

PosY: posición en el eje Y.

Tamaño: longitud de ancho y alto al manejar su imagen en un rectángulo.

Tipo de Raza: un entero que indica la raza del personaje, con atributos y apariencia específicas.

Métodos:

moverPersonaje(dirX, dirY): permite moverse en las direcciones arriba, abajo, izquierda, derecha y realizar saltos.

disparar(): realizar disparos.

incinerar(): este método será específico para el enemigo final, y lanzará fuego con un movimiento parabólico.

Constructor: inicializa las características del personaje.

Destructor: libera recursos y memoria al destruir el personaje.

SetPosX(newPosX): establece la posición en el eje X.

SetPosY(newPosY): establece la posición en el eje Y.

GetPosX(): obtiene la posición en el eje X.

GetPosY(): obtiene la posición en el eje Y.

Funciones gráficas de Qt:

El constructor también debe implementar funciones gráficas para darle apariencia al personaje. Cada apariencia tendrá atributos y apariencia distintos.

Este diseño de clase de "Personaje" es flexible y puede ser utilizado tanto para Rick como para Morty, ya que ambos comparten atributos similares, como velocidad, vida y posición.

**Clase Obstáculo:**

Atributos:

Tamaño: tamaño del obstáculo en términos de longitud y altura.

PosX: posición en el eje X.

PosY: posición en el eje Y.

Métodos:

Constructor: inicializa el obstáculo con su tamaño y posición.

Destructor: libera los recursos y memoria asociados al obstáculo.

SetPosX(newPosX): establece la posición en el eje X.

SetPosY(newPosY): establece la posición en el eje Y.

GetPosX(): obtiene la posición en el eje X.

GetPosY(): obtiene la posición en el eje Y.

Clase Objeto de Recolección:

Atributos:

Tipo: el tipo de ingrediente necesario para la poción.

Tamaño: tamaño del objeto de recolección.

PosX: posición en el eje X.

PosY: posición en el eje Y.

Métodos:

Constructor: inicializa el objeto de recolección con su tipo, tamaño y posición.

Destructor: libera los recursos y memoria asociados al objeto de recolección.

SetPosX(newPosX): establece la posición en el eje X.

SetPosY(newPosY): establece la posición en el eje Y.

GetPosX(): obtiene la posición en el eje X.

GetPosY(): obtiene la posición en el eje Y.

**Clase Enemigo:**

Atributos:

Velocidad: velocidad de movimiento del enemigo.

PosX: posición en el eje X.

PosY: posición en el eje Y.

Métodos:

moverEnemigo(dirX, dirY): permite al enemigo moverse en las direcciones específicas.

Constructor: inicializa al enemigo con su velocidad y posición.

Destructor: libera los recursos y memoria asociados al enemigo.

SetPosX(newPosX): establece la posición en el eje X.

SetPosY(newPosY): establece la posición en el eje Y.

GetPosX(): obtiene la posición en el eje X.

GetPosY(): obtiene la posición en el eje Y.

**Clase portal:**

Atributos:

Tamaño: tamaño del portal.

PosX: posición en el eje X.

PosY: posición en el eje Y.

Métodos:

Constructor: inicializa el portal con su tamaño y posición.

Destructor: libera los recursos y memoria asociados al portal.

SetPosX(newPosX): establece la posición en el eje X.

SetPosY(newPosY): establece la posición en el eje Y.

GetPosX(): obtiene la posición en el eje X.

GetPosY(): obtiene la posición en el eje Y.

**Clase jefe final:**

Atributos:

Vida: cantidad de resistencia del jefe final.

PosX: posición en el eje X.

PosY: posición en el eje Y.

Métodos:

atacar(): permite al jefe final realizar ataques especiales.

Constructor: inicializa al jefe final con su vida y posición.

Destructor: libera los recursos y memoria asociados al jefe final.

SetPosX(newPosX): establece la posición en el eje X.

SetPosY(newPosY): establece la posición en el eje Y.

GetPosX(): obtiene la posición en el eje X.

GetPosY(): obtiene la posición en el eje Y.

**Clase Tiempo:**

Atributos:

Duración: tiempo restante antes de que los personajes sean capturados.

Métodos:

iniciarTemporizador(): comienza el temporizador.

detenerTemporizador(): detiene el temporizador.

Constructor: inicializa el tiempo con la duración.

Destructor: libera los recursos y memoria asociados al tiempo.

**Clase Menú Inicial:**

Métodos:

mostrarMenu(): muestra el menú de inicio del juego y permite a los jugadores seleccionar personajes y configuraciones antes de comenzar la aventura.