



PONTIFICIA UNIVERSIDAD CATÓLICA MADRE Y MAESTRA – PUCMM
FACULTAD DE CIENCIAS DE LAS INGENIERÍAS
Departamento de Ingeniería en Sistemas y Computación – ISC

Asignatura:

ISC-210-001 Programación Aplicada

Trabajo de Curso:

Ninja Runner

Autores:

2014-1811 Johanna Rodríguez

2014-1336 Karvin Jimenez

Tutor:

Ing. Miguel Moronta

Entregado en Fecha:

Jueves, 12 de marzo de 2020



RESUMEN / ABSTRACTO DEL PROBLEMA

Esta versión de Bob's World a la que llamamos Ninja Runner consiste en un personaje Ninja que debe atravesar los caminos entre montañas hasta llegar a su aldea antes de que sea invadida, venciendo los enemigos malvados que se atraviesan en el camino a la aldea. El jugador posee 1 vida, si un enemigo lo choca se debilita perdiendo una vida, al perder la vida se pierde la partida, además al caer al precipicio se pierde la partida. Posee un límite de tiempo y si no termina en este tiempo también pierde la partida. Va recolectando y acumulando monedas, también va registrando un puntaje acorde a su recolección de monedas y vencimiento de enemigos. Cada moneda recolectada suma 50 puntos al score final.

This version of Bob's World called Ninja Runner consists on a Ninja character who must cross the paths between mountains until reaching his village before it is invaded, defeating the evil enemies that are crossed on the way to the village. The player has 1 live, if an enemy hits him, he weakens, losing one life, losing the live loses the game, and falling to the precipice, the game is lost. It has a time limit and if it does not end in this time it also loses the game. It is collecting and accumulating coins, it is also registering a score according to its collection of coins and expiration of enemies. Each collected coin adds 50 points to the final score.



TABLA DE CONTENIDO

Resumen / Abstracto del Problema	1
Análisis	1
Cuerpo del Informe	2
Código Fuente	3
Condiciones de Error	4
Notas al Profesor	5
Conclusiones	6
Recomendaciones	7
Referencias Bibliográficas	8

ANÁLISIS

Bob's World es un juego de plataformas que consiste en ayudar a un personaje, similar a Super Mario, a recorrer cada uno de los niveles sin impactar contra los obstáculos. Para ello, se debe prestar mucha atención al entorno y esquivar cada trampa mientras se recolectan recompensas. Además del escenario, el sistema de control se asemeja mucho al de los clásicos juegos protagonizados por el famoso fontanero de Nintendo. Es por esto que basta con pulsar sobre las flechas de dirección para moverse y sobre los botones de acción para saltar o golpear a los enemigos. En la parte superior de la interfaz se puede observar el tiempo que se tarda en completar cada nivel, las recompensas cosechadas y la puntuación obtenida.

Bob's World da la oportunidad de retroceder en el tiempo a la infancia con una misión legendaria: rescatar a la Princesa. El universo de este juego contiene niveles bien diseñados, varios enemigos, superjefes, jugabilidad, gráficos atractivos, y música y sonidos relajantes. El mundo de Bob (Bob's World) ha estado vacío desde que la princesa fue secuestrada en la selva. Entonces, así comienza la aventura, la tarea es ayudar a Bob a atravesar corriendo la misteriosa jungla y a saltar sobre los obstáculos y monstruos supermalvados para salvar a la hermosa Princesa en el destino final de la aventura.

El funcionamiento del juego está definido por:

- Usar los botones para saltar, moverte y disparar
- Comer setas y objetos para hacerte más fuerte y derrotar a todos los monstruos.
- Recoger todas las monedas y objetos de bonus para conseguir más puntos y poder comprar artículos adicionales en la tienda.

Los objetivos principales de la práctica consisten en desarrollar un videojuego sencillo utilizando la plataforma de desarrollo Unity en el cual se debe dibujar objetos sencillos y animados y programar detección de colisiones. Además la incorporación de por los menos cuatro funciones físicas en el videojuego.

Para esta práctica se desarrollará un videojuego basado en Bob's World llamado Ninja Runner, este videojuego será similar pero con una reducción de sus funcionalidades, sólo se abarcará el primer nivel del juego y no se presentarán todas las opciones del menú del juego original. Además, el escenario no será exactamente el mismo, a diferencia del juego original que el personaje toma pociones para aumentar su tamaño el Ninja comerá frutas para este mismo efecto.

Esta versión de Bob's World envuelve la trama de la aldea de los ninjas que será invadida por los enemigos y todos los ninjas deben reunirse para protegerla, el ninja personaje principal debe atravesar los caminos entre montañas hasta llegar a su aldea evitando los enemigos malvados que se atraviesan en el camino. El Ninja posee 1 vida, si un enemigo lo choca va perdiendo una vida, al perder la vida se pierde la partida, al caer al precipicio se pierde la partida. Posee un límite de tiempo y si no termina en este tiempo también pierde la partida. Va recolectando y acumulando monedas, también va registrando un puntaje acorde a su recolección de monedas y vencimiento de enemigos. Cada moneda recolectada suma 50 puntos al score final. Al recolectar la fruta gana una vida adicional.

El escenario será una representación de un camino entre montañas (se verán detrás) hasta llegar a una aldea, tendrá obstáculos y varios enemigos en el trayecto que deberá evitar o vencer, saltando y corriendo, aparecerán frutos para que el ninja los coma y se haga más fuerte. Tendrá música de fondo y sonidos de la recolección de monedas, crecimiento, pérdida de vida y pérdida de la partida.

CUERPO DEL INFORME

El cuerpo del informe se estructura en dos capítulos:

- Capítulo 1 – Descripción del Negocio

En Ninja Runner, una versión de Bob's World, el jugador deberá ir acumulando puntos en su trayecto hasta llegar a la aldea según va recolectando monedas, frutas y venciendo los enemigos. El juego posee las siguientes características:

- El juego inicia con un total de 1 vida.
- El juego debe iniciar con un tiempo de 240 segundos.
- El jugador podrá correr y saltar para recolectar monedas, frutas y atacar los enemigos.
- Cada moneda recolectada suma 50 puntos al puntaje.
- Cada fruta atrapada suma 1 vida al ninja.
- Se gana la partida al llegar a la aldea.
- El juego se pierde al terminar el límite de tiempo, al caer al precipicio o al agua o al perder la vida.
- El juego posee 1 nivel con una música de fondo.
- El juego permite volver a jugar luego de ganar o perder la partida.

- Capítulo 2 – Solución Propuesta

Diagrama de clases

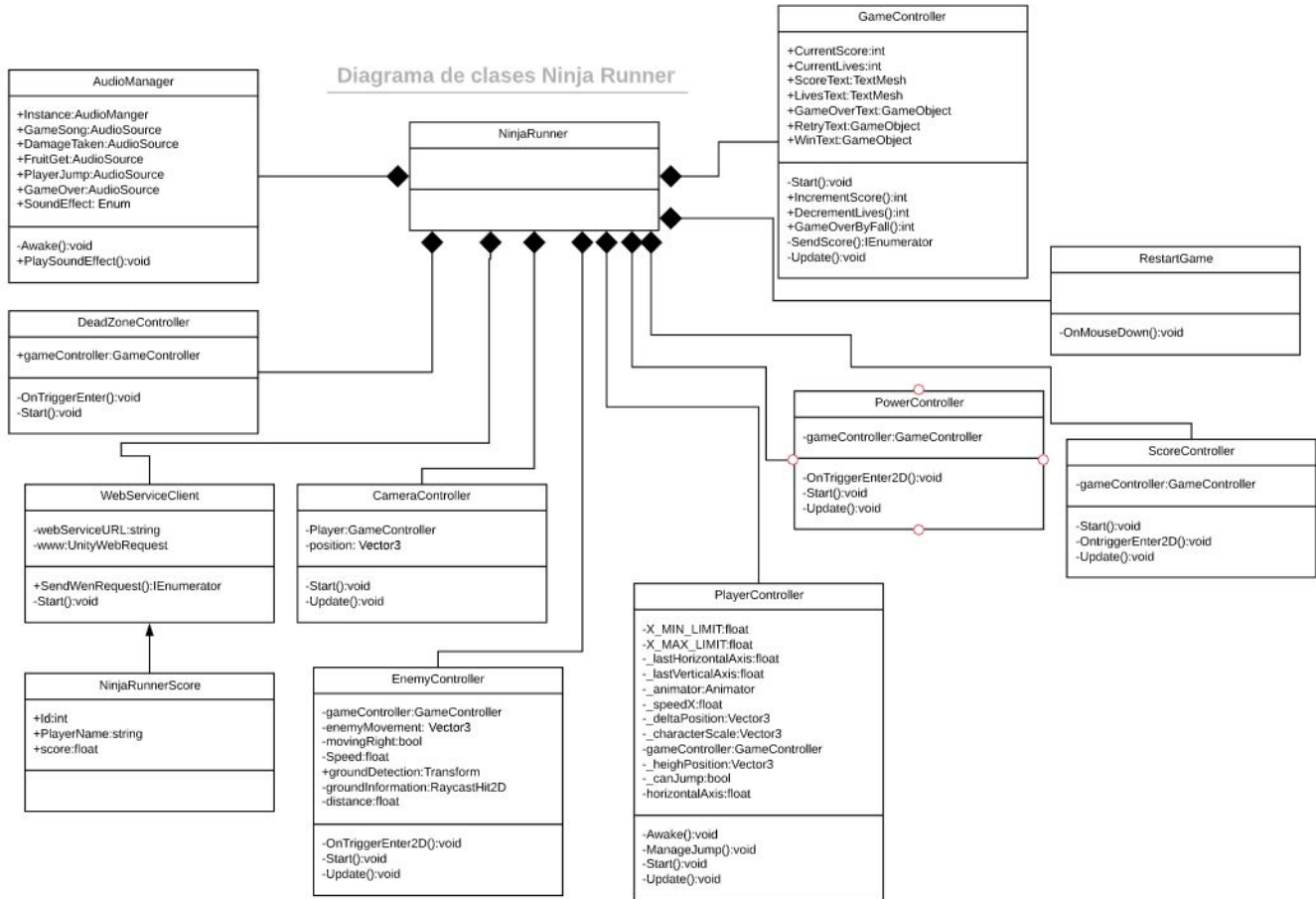


Diagrama de Caso de uso



Diagrama de Estado

Diagrama de Estado Ninja Runner

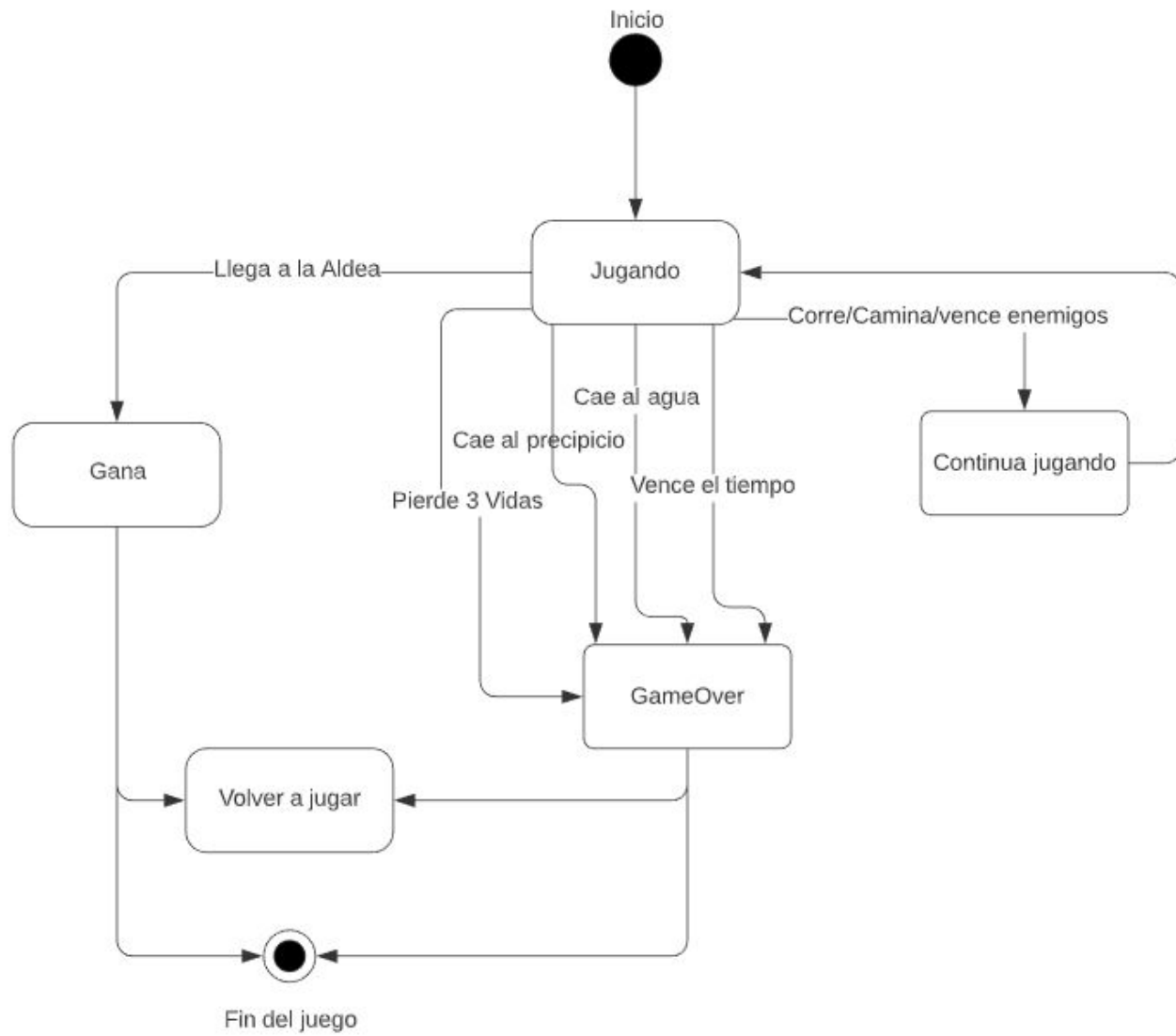


Diagrama de secuencia

Diagrama de secuencia Ninja Runner

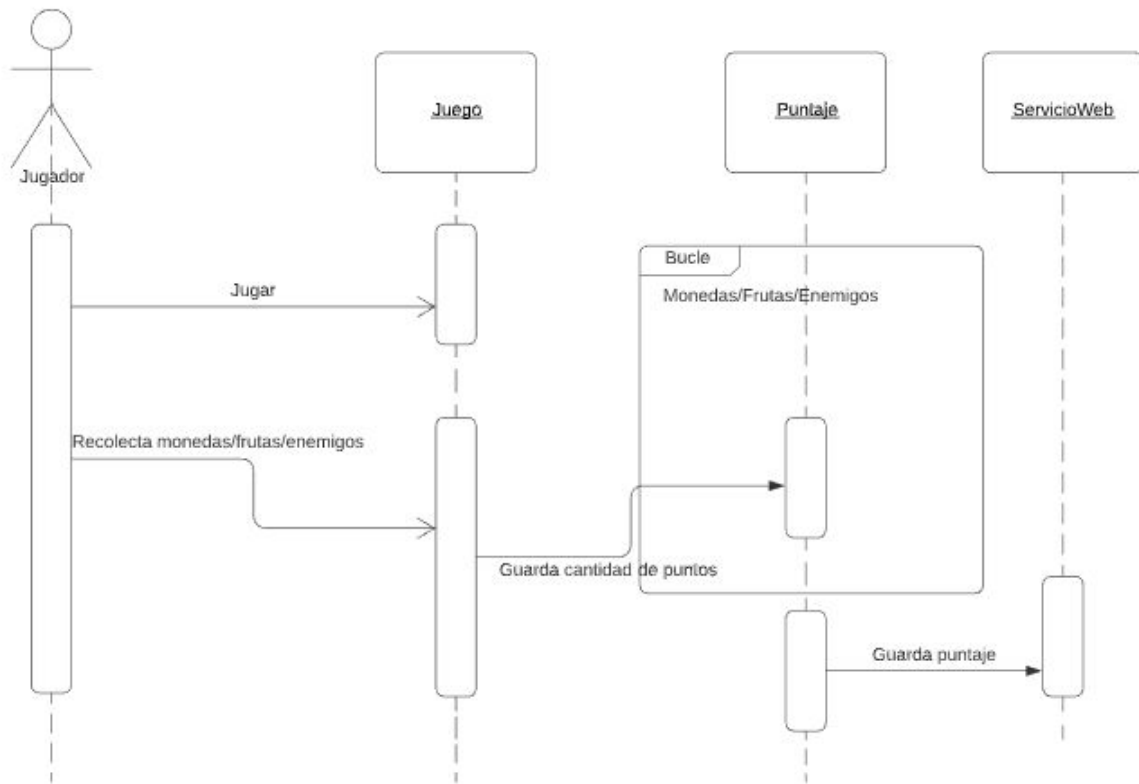
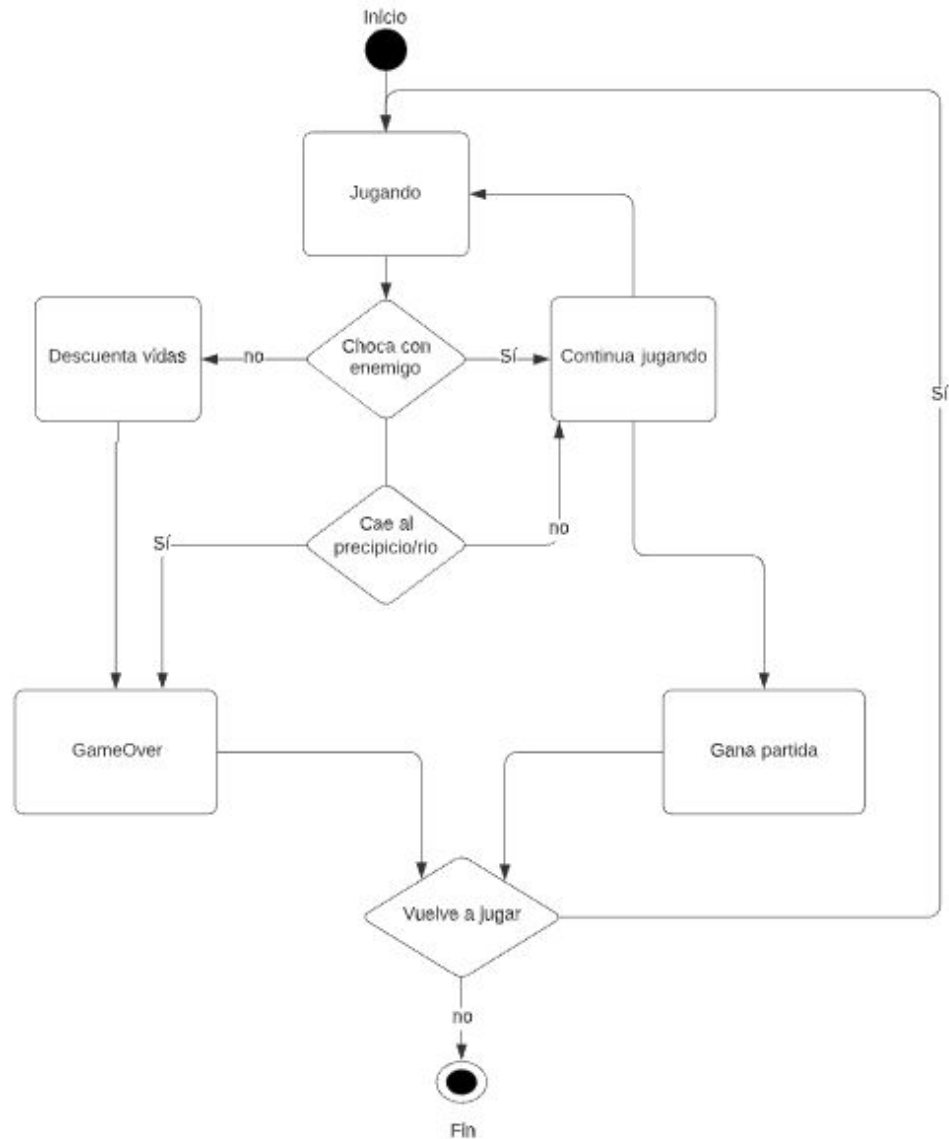


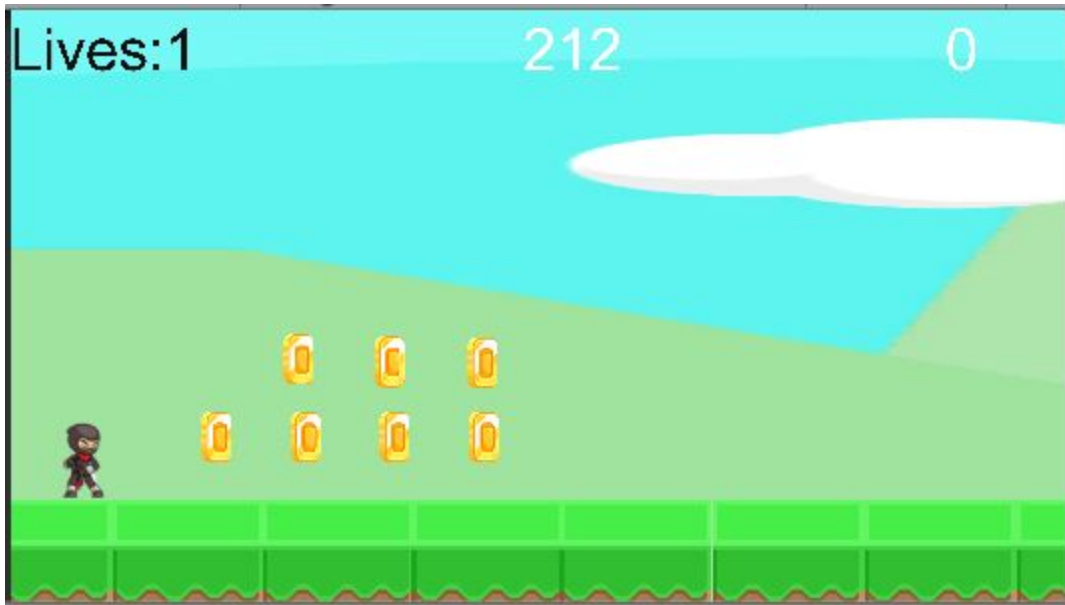
Diagrama de actividad

Diagrama de actividades Ninja Runner

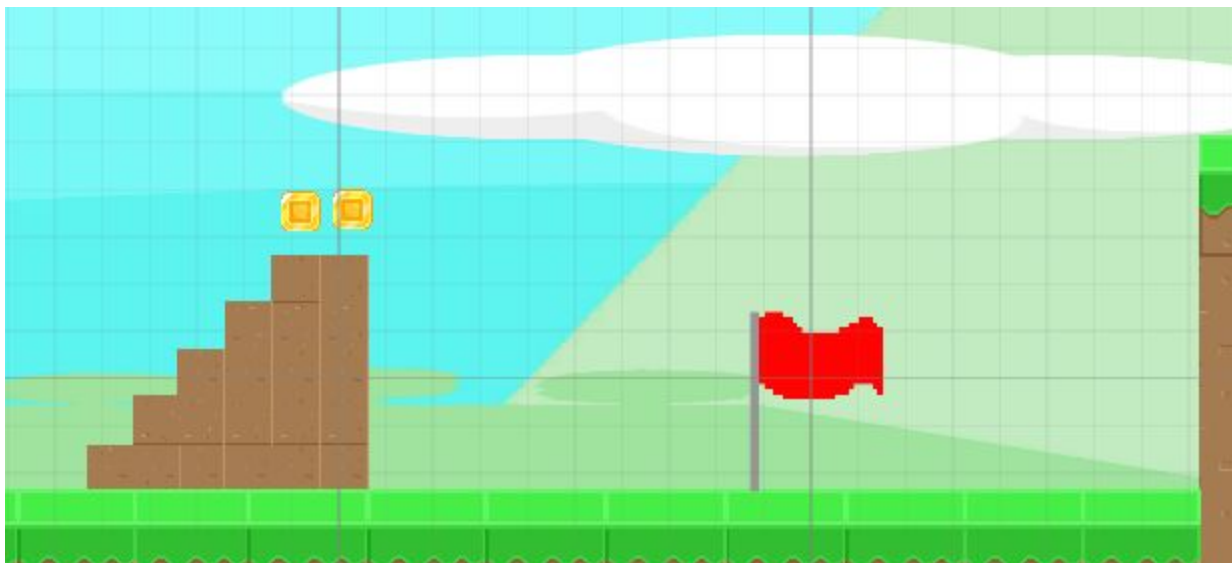


Capturas de pantalla: Muestra de la escena

En funcionamiento



Meta



Enemigos



Para crear el background del escenario del juego se utilizara un Quad con una material para colocar una imagen. Se utilizará un 3D text para el conteo de las vidas del jugador, también para el aumento del score y para que vaya restando el tiempo en el timer

Para el manejo de audio y sonido tanto de background como las reacciones del juego se utilizan varios componentes de Audio Source y se programa un Script para su asignación llamado AudioSource. Cada uno de los audios a utilizar fueron descargados de la tienda de Unity como recurso gratuito.

Para los tiles que conforman el terreno utilizamos Box Collider, y un audio source para el sonido de la collision y un script para la configuración del toque de la tecla, aquí también se maneja la disminución de las vidas.

Se utilizó un script para manejar algunas funcionalidades del juego como el conteo de puntos y la disminución de vidas, asimismo se programa la aparición de los textos y figuras en el juego. Por otro lado se envía el puntaje para su almacenamiento en la base de datos vía el servicio web configurado, mediante el script llamado webserviceclient y el script general que se encarga de las demás funciones del juego es el GameController.

Para manejar la zona muerta se crea un objeto con un Box Collider e Is trigger activado para configurar la destrucción del personaje una vez colisiones en la zona.

Para los enemigos se utilizó un raycast en conjunto de un objeto vacío al cual hacíamos hijo de el enemigo y mediante raycast verificamos que siempre que haya terreno el enemigo pueda moverse en cierta dirección, si en un momento detecta que no hay tierra el enemigo dará la vuelta para el otro lado y repetirá el mismo proceso hasta que no detecte terreno.

Además de todo lo mencionado se creó un script llamado CameraController, este script se encarga de que la cámara, el texto del score, el de la vida y el del timer sigan al jugado todo el tiempo y que de esta forma no sean dejados atrás por el jugador.

Para el almacenamiento de la puntuación en la base de datos SQL se creó un servicio web del tipo Web API y MVC, se definieron las peticiones necesarias para almacenar tanto el nombre del jugador como su puntaje.

CÓDIGO FUENTE

CONDICIONES DE ERROR

- Hay momentos en los que si se hace un mal salto y el jugador cae mal en la plataforma no retoma bien la posición correcta.
- Cuando se pierde la partida por caer al agua el retry text y el game over text no salen en la parte actual de la escena donde esta el jugador sino al principio de la escena.

NOTAS AL PROFESOR

CONCLUSIONES

En conclusión, para el videojuego desarrollado se utilizó la plataforma de desarrollo de videojuego Unity, NinjaRunner es un videojuego sencillo que implementa objetos sencillos con distintos componentes que permiten la programación de animaciones y las reacciones ante detección de colisiones con otros objetos. Además, se implementó un servicio web con la finalidad de almacenar los datos del jugador y el juego en una base de datos. Para concluir podemos decir que los conocimientos adquiridos en la clase pudieron ser aplicados en la creación y desarrollo de este juego cosas como la agregación de fondos mediante el uso de quad y material hasta la implementación de un servicio web para así poder almacenar los puntajes de los jugadores.

RECOMENDACIONES

En una etapa de mejora al juego desarrollado se sugiere implementar sincronización del toque de la tecla con una toca del piano siguiendo el ritmo de la canción.

REFERENCIAS BIBLIOGRÁFICAS

<https://play.google.com/store/apps/details?id=free.os.jump.superbros.adventure.world&hl=es>

<https://bobs-world.uptodown.com/android>

<https://www.youtube.com/watch?v=o0j7PdU88a4&list=PL2AQpVXFsmrnw08tz0RPa1ZqSQG6G08u&index=13&t=0s>

<https://www.youtube.com/watch?v=aRxuKoJH9Y0&list=PL2AQpVXFsmrnw08tz0RPa1ZqSQG6G08u&index=12&t=194s>

<https://www.youtube.com/watch?v=k-75tAys7iI&list=PL2AQpVXFsmrnw08tz0RPa1ZqSQG6G08u&index=11&t=0s>