Desarrollo Videojuego Multijugador Online Metodología de la Investigación

Minati Franco, Nicolas Cataldo, Octavio Fariña

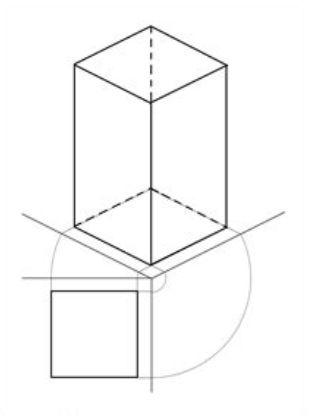
**Tetris 3D**

**Descripción del Videojuego**

Procederemos a desarrollar un videojuego 3D mulitijugador online basado en el clásico Tetris como proyecto final. Hemos tomado como referencia el juego Blockout el cual es una versión del Tetris en 3D.

El videojuego a desarrollar consta de un area de juego tridimensiaonal en forma de prisma donde la base del prisma actue como “suelo” donde se detendran las piezas que van cayendo. Las piezas tendran formas inspiradas en las presentes en Blockout y estas caeran desde la cara opuesta a la base del prisma.

A diferencia de Blockout el jugador tendra la habilidad de alternar entre 5 posiciones o vistas, una vista superior y una vista isometrica por cada una de las 4 esquinas del prisma (simpre situando el “suelo” como base del prisma).

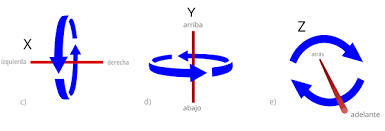


El jugador contará con la posibilidad de rotar la pieza que esté cayendo sobre cualquiera de los 3 ejes, teniendo dos posibles direcciones para cada eje. Para esto utilizaremos un total de 6 teclas destinadas a la rotación de la pieza.

**Eje x Eje y Eje z**

**[Q] [W] [E]**

**[A] [S] [D]**



Cada rotación se dará en intervalos de 90 grados.

Para desplazar inmediatamente la pieza hacia el suelo el usuario debera presionar la tecla ESPACIO.

Al formar un plano completamente cubierto por piezas, el mismo se eliminará y se desplazaran las piezas o partes de las mismas hacia el suelo (bajarán un nivel), se establece una puntuacion por haber completado un plano.

El jugador perdera cuando posicione una pieza y cualquier parte de la misma quede un nivel por encima del nivel máximo.

Con respecto al multijugador desarrollaremos como mínimo uno de dos posibles modos de juego.

**Modo Competitivo:** 2 jugadores se enfrentan, cada uno con su propia area de juego (su propio prisma y piezas), se estableceran distintas mecánicas de ataque, por ejemplo cuando un jugador completa un plano este hará que se le llene un plano con basura, es decir huecos al azar, al rival. El rival que recibe la basura por lo tanto ahora tiene un nivel más ocupado y debe “limpiarlo” (completar el plano incompleto). El plano basura se agregara en el nivel más bajo desplazando así un nivel para arriba todas las piezas del jugador.

**Modo Cooperativo:** se juntan dos primas, uno correspondiente a cada jugador, donde iran cayendo las respectivas piezas de cada prisma. En este caso por más que un jugador complete un plano en su prisma debe también esperar que se complete el plano del mismo nivel del prisma que se encuentra adyacente. De esta manera ambos jugadores deben cooperar al posicionar sus piezas para que no queden huecos en los respectivos planos que intenten formar.

En ambos modos la partida se acaba cuando cualquiera de los dos jugadores posiciones una pieza o parte de pieza por encima del nivel máximo.

Stack Tecnológico

Todo el desarrollo del videojuego será realizado con herramientas de código abierto sobre el sistema operativo Ubuntu.

**Paradigma: Programacion Orientada a Objeto**

Utilizaremos el paradigma de la programación orientada a objetos ya que este facilitará el desarrollo del videojuego al poder representar los elementos que intervienen en el videojuego como objetos en nuestro código.

**Motor Gráfico: Godot**

Godot es un motor de videojuegos 2D y 3D multiplataforma, libre y de código abierto, publicado bajo la Licencia MIT y desarrollado por la comunidad de Godot. El motor es funcional en sistemas Windows, OS X, Linux y BSD. Permite exportar los videojuegos creados a PC, teléfonos móviles, y HTML5.

**Editor 3D: Blender**

Blender es un programa informático multiplataforma, dedicado especialmente al modelado, iluminación, renderizado, la animación y creación de gráficos tridimensionales. También de composición digital utilizando la técnica procesal de nodos, edición de vídeo, escultura (incluye topología dinámica) y pintura digital.

Utilizaremos Blender para crear los modelos 3D de los recursos utilizados (piezas, prisma de juego, etc).

**Editor de Texto: VSCodium**

El editor de texto creado por Microsoft llamado Visual Studio Code presenta tiene su código fuente open source, sin embargo las versiones compiladas del mismo distribuidas por Microsoft incorporan funcionalidades de telemtria propietarias. Para esto utilizaremos VSCodium evitando así tener que compilar nosotros el código fuente de Visual Studio Code, ya que VSCodium es una version compilada de este código fuente mantenida y distribuida por la comunidad open source.

**Lenguaje de Programación:**

Utilizaremos una combinacion de C# y GDScript (llenguaje desarrollado para integrarse con el motor gráfico Godot)

**C#**

C# es un lenguaje de programación multiparadigma desarrollado y estandarizado por la empresa Microsoft como parte de su plataforma .NET.

Aunque C# forma parte de la plataforma .NET, ésta es una API, mientras que C# es un lenguaje de programación independiente diseñado para generar programas sobre dicha plataforma. Ya existe un compilador implementado que provee el marco **Mono - DotGNU**, el cual genera programas para distintas plataformas como Windows Microsoft, Unix, Android, iOS, Windows Phone, Mac OS y GNU/Linux.

C# fue aprobado como un estándar por la ECMA (ECMA-334) e ISO (ISO/IEC 23270) por lo que su definicion, código fuente, etc. fueron establecidos como open source.

**GDScript**

GDScript es un lenguaje de programación de alto nivel con tipado dinámico utilizado para crear contenido para Godot Engine. Utiliza una sintaxis similar a Python (los bloques están basados en la indentación y muchas palabras clave son similares). Su objetivo es optimizar e integrarse estrechamente con Godot Engine, permitiendo una gran flexibilidad en la creación e integración de contenido.