

TetrisCraft

Manuel Cervera Nicolás Romero Gabriel Sepúlveda Cristobal Venegas

23 de junio de 2019

1. Descripción

TetrisCraft es un juego inspirado en la versión tridimensional del popular juego Tetris. La idea nace al incorporar en en Tetris conceptos del juego MineCraft, enfatizando sobre todo su aspecto gráfico. En el juego piezas compuestas por cubos bajan hasta colisionar con otra pieza o con la base de una grilla contenedora con dimensiones de 7x7x12 cubos, la velocidad de las piezas irá aumentando progresivamente en medida que avanza el juego. Existen 12 piezas diversas que el jugador puede acomodar, ya sea moviéndose hacia adelante, atrás o a los lados, o girándola en torno a cualquiera de los 3 ejes x, y o z del espacio tridimensional.

El objetivo del juego es evitar durante la mayor cantidad de tiempo posible que las piezas se acumulen y toquen la parte superior de la grilla contenedora. Para esto el jugador puede rellenar todos los espacios habilitados de uno de los 12 niveles de altura, lo que ocasiona que todos los cubos en ese nivel sean eliminados y que todos los demás cubos que están por encima bajen 1 espacio. Este efecto puede acumularse para hacer desaparecer múltiples niveles a la vez.

Eliminar un nivel de cubos otorga al jugador puntaje asociado a los tipos de cubos eliminados, existiendo 12 tipos de cubos distintos. Los distintos tipos de cubos son distintos tipos de materiales que existen en el mundo de MineCraft, la cantidad de puntos entregados por ellos depende de su rareza en el mundo, haciendo relación entre el puntaje de tetris con los puntos de experiencia de Minecraft. Eliminar múltiples niveles a la vez agrega un multiplicador al puntaje obtenido al igual que en el tetris original.

El juego termina solo cuando alguna pieza toca el límite superior de la grilla y no puede bajar.

2. Mejoras

La principal mejora integrada en TetrisCraft respecto al Tetris 3D original es el agregar componentes referentes al juego MineCraft. Este agregado de componentes se puede apreciar en todo el aspecto visual del juego, el sistema de entrega de puntos y las mecánicas adicionales incorporadas.

2.1. Aspecto Visual

El aspecto visual del juego está hecho para ser el mismo que el del juego MineCraft.

El menú inicial de ambos juegos es el mismo, o muy similar. La textura de los cubos que componen las piezas son las que encontramos en los bloques del mundo de Minecraft y la HUD que muestra el puntaje y las piezas futuras está hecha a similitud de la HUD In-Game de MineCraft.



2.2. Sistema de Puntos

Al igual que en el tetris original el jugador obtiene puntos cada vez que elimina uno de los niveles de altura del juego. Eliminar varios a la vez significa que el puntaje obtenido será multiplicado en base a la cantidad de niveles eliminados. Adicionalmente se ha agregado una mecánica en la que la cantidad de puntos que entrega cada cubo al ser eliminado varía dependiendo de la rareza que tiene el cubo en MineCraft, por lo que eliminar un cubo de diamante otorgará una cantidad de puntaje mucho mayor que uno de tierra. Esto le permite al jugador intentar explotar esta mecánica al eliminar varios niveles de bloque raros a la vez para obtener el mayor puntaje posible.

2.3. Mecánicas

Otra mejora que se ha agregado respecto al tetris 3D convencional es la incorporación de los bloques de dinamita y de bloques que sí respetan la gravedad como la arena y la grava que son propios de MineCraft.

El bloque de dinamita es un bloque que al igual que en MineCraft posee la capacidad de explotar, destruyendo los bloques que se encuentran a su alrededor, la cantidad de bloques destruidos dependerá de la resistencia que tengan los bloques aledaños. Cada bloque destruido reduce la fuerza de la explosión de manera relativa a la resistencia de dicho bloque. Esto permite al jugador destruir bloques de manera sencilla y abrir paso a espacios vacíos que pueda haber tapado debido a un error. También le dan la opción de intencionalmente generar cavernas.^{en} las cuales depositar la dinamita para generar un mayor nivel de destrucción

Minecraft y Tetris comparten el hecho de que los bloques no caen una vez han sido situados en un lugar, con dos principales diferencias. En Tetris cuando se elimina un nivel de altura todos los bloques que están por encima bajan un nivel, mientras que en MineCraft existen dos bloques que siempre caen si es que hay un espacio vacío abajo, siendo estos la arena y la grava. Mientras que la mecánica del Tetris era obligatoria tenerla se optó por incorporar la mecánica

de la arena y la grava de Minecraft, permitiéndole al jugador crear efectos en cadena al acumular dichos bloques en posiciones específicas y facilitando la eliminación de niveles después de que el jugador haya cometido un error dejando espacios vacíos entre los bloques.

3. Problemáticas

La primera problemática con la que se encontró el equipo de desarrollo fue el cómo programar el juego. En un inicio se planeó trabajar solo con lo que traía JavaScript y WebGL de base, pero la complejidad de esto hubiera sido mucho mayor. Frente a esto, después de un tiempo se optó por trabajar con Three.js, esto dio paso a una segunda problemática, como incorporar Three al proyecto, a pesar de ser solo una librería por lo que bastaba con agregarla al proyecto al igual que cualquier otra hoja de código la información provista en internet no era clara y el archivo de descarga era el repositorio de git completo, dando paso a la confusión de cuál fichero había que agregar o cómo agregar el repositorio completo.

Otra problemática que persiste durante algún tiempo fue el cómo organizar la arquitectura del proyecto, la cual fue cambiando constantemente durante el proceso de desarrollo hasta alcanzar un resultado que todos los miembros del equipo considerarán estable.

Adicionalmente una gran cantidad de problemas pequeños fueron surgiendo a medida que se quería agregar un nuevo elemento o mecánica al juego. Como mover las piezas, agregar textos, efectos de backing de luz, sistema de multi-cámara para poder moverse en mundo sin tener que mover la HUD, que controles utilizar, déficit en el sistema de colisiones, etc. No sería una subestimación decir que cada vez que se quería agregar algo nuevo al juego nos encontrábamos con un pequeño obstáculo que superar. Algunos tal vez no tan pequeños como la necesidad de que la cámara de la HUD estuviera en diagonal para que los cubos en ella se vieran de la misma forma que en Minecraft, por lo que todo en esa escena debía moverse en medidas calculadas a base de funciones trigonométricas, o el hecho de que el juego se "laga" cuando se tenía una gran cantidad de cubos en pantalla debido principalmente a los efectos de backing de luz que se agregaron. Problemas que se solucionaron paso a paso.

4. Supuestos

Dado que el juego nace de la idea de impregnar la esencia de Minecraft en el Tetris 3D la cantidad de cosas que se pudieron haber agregado es incontable. Algunas de las ideas más discutidas en el equipo desarrollador es el agregar bloques de agua, lava y fuego cada uno con sus respectivas mecánicas, como los son esparcirse a bloques vacíos (lava y agua), quemar bloques de madera creando nuevos bloques de fuego (lava y fuego) y sus respectivas interacciones, como crear piedra u obsidiana al hacer interactuar lava con agua o apagar el fuego con agua.

El agregar "mobs", como los enderman, los cuales son capaces de cambiar bloques de posición, entre otros.

La creación de un ciclo día y noche que controle la aparición de los mobs y

la posibilidad de matarlos al dejarles caer un cubo encima, lo cual resulta en puntaje adicional para el jugador.

La cantidad de posibilidades continúa de manera indefinida al ir pensando en cómo traspasar la gran cantidad de mecánicas existentes en MineCraft al Tetris 3D.

5. Equipo Desarrollador

El equipo desarrollador está compuesto por 4 integrantes. Las labores de programación fueron distribuidas de manera equitativas entre todos, al igual que las de ideación del concepto de juego, Nicolás Romero encargándose de todo lo que es incorporación del arte y programación de los menús de juego. Cristobal Venegas encargado de la programación del feedback al usuario y pulido. Santiago Cervera encargado de la programación de todos los efectos de estética del juego y pulido. Gabriel Sepúlveda encargado de programación de las mecánicas, arquitectura del juego. Cabe mencionar que el hecho de que uno de los miembros del equipo esté encargado de una de las áreas no quiere decir que haya hecho toda la programación de dicha área. La programación del juego, como se dijo anteriormente, se distribuyó de manera equitativa a medida que los elementos eran necesarios para continuar con el desarrollo del juego.