

DOCUMENTO FINAL

PROYECTO MINECRAFT

William Castro 6000180, Juliana Ordoñez 6000458

Universidad Militar Nueva Granada

Cajicá

Resumen-En este proyecto realizamos una previa investigación y una práctica de implementación para hacer un simulación del video juego Minecraft hecho en three.js con las diferentes geometrías pero principalmente del cubo, haciendo uso de los diferentes temas vistos durante el desarrollo de la materia de computación gráfica (colisiones, CGS, curvas, texturas).

I. INTRODUCCIÓN

En este documento presentamos la propuesta al proyecto de computación gráfica a realizada, iniciamos con un mundo de cubos donde tenemos un personaje, a la audiencia se le muestra la visión principal del personaje en modo primera persona conocido normalmente en el mundo que puede colocar diversos materiales predeterminados, y construir con estos diferentes estructuras cúbicas.

II. OBJETIVO

- Obtener información de cómo implementar las colisiones en cubos.
- Realizar el análisis , diseño y desarrollo de el juego minecraft utilizando la metodología de colisiones en three.js
- Obtener codigos reutilizables como recurso y ejecutarlos en el código elaborado por nosotros inicialmente.
- Hacer uso de las diferentes temáticas vistas en la materia de computación.

III. DESCRIPCIÓN COMPONENTE

Mencionaremos los diferentes componentes que puede tener una aplicación web para el desarrollo del videojuego de minecraft.

A. Metodología

Se hará una breve descripción de los lenguajes , programa, y formatos usados para la implementación del proyecto.

HTML: En sus siglas en inglés de HyperText Markup Language, hace referencia al lenguaje de marcado para la elaboración de páginas web, donde lo utilizamos para crear la página principal donde el usuario podrá interactuar con el menú, también fue usado para crear los textos de cada página alterna.

CSS: En español «Hojas de estilo en cascada», es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado, donde lo utilizamos para darle estilo a la página principal y para estructurar los títulos en los documentos alternos.

JAVASCRIPT: Es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico, lo utilizamos para crear diferentes objetos de entorno gráfico durante el desarrollo del videojuego.

GIT: Es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Lo utilizamos durante todo el proyecto para llevar las diferentes versiones y control de nuestro videojuego.

THREE.JS: Es una biblioteca liviana escrita en JavaScript para crear y mostrar gráficos animados por computadora en 3D en un navegador Web y puede ser utilizada en conjunción con el elemento canvas de HTML5, SVG y WebGL. Todo el proyecto está basado en esta librería donde creamos todo el entorno y las gráficas para animar el videojuego.

GIF: Graphics Interchange Format también llamado Compuserve GIF y más conocido como GIF, es un formato gráfico digital utilizado ampliamente en la World Wide Web, tanto para imágenes como para animaciones, lo

utilizamos para la pagina principal para emular el país de Minecraft.

PNG: Es un formato gráfico basado en un algoritmo de compresión sin pérdida para bitmaps no sujeto a patentes fue desarrollado en buena parte para solventar las deficiencias del formato GIF. Lo utilizamos para crear las texturas de casi todo el entorno del videojuego ya que es muy fácil de trabajar y de implementar en el código.

VISUAL STUDIO CODE:

IV.DISEÑO

A. Menú Principal

Inicialmente creamos una aplicación web , donde utilizamos el lenguaje de HTML y CSS para crear un menú de navegación sobre todo el proyecto. Usamos un gif de fondo para hacer más llamativo.



Fig 1. Imagen menu de creacion de objetos nuevos Minecraft

B.Páginas

En esta parte diseñamos una vinculación con el menú principal, para tener una aplicación compacta y fácil de usar para el jugador.

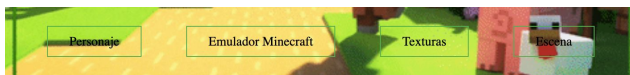


Fig 2. Imagen de interacción de Minecraft

C.Creacion de personaje

En esta sesión se realizó la creación del personaje principal de minecraft haciendo uso de la geometrias cubo, jerarquías, matrices, rotaciones y unión(caminado del personaje). Todo con la librería de Three.js Y finalmente el uso de una imagen de formato PNG donde se observa el modelo y la indumentaria del personaje.

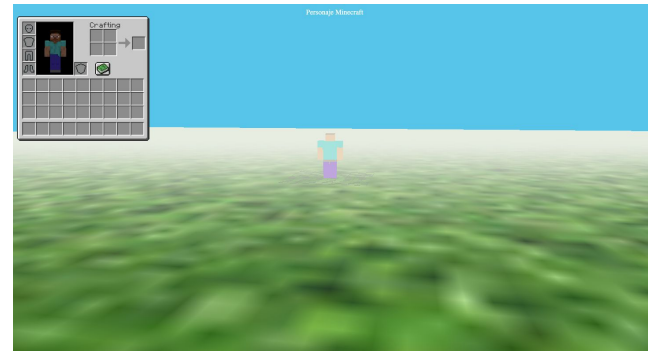


Fig 3. Imagen de Personaje de Minecraft

D.Emulador del Minecraft

En esta sesión se implementó la geometría cubo con una textura básica, creando con Ctrl A el cubo gris y con Ctrl S un cubo más largo .

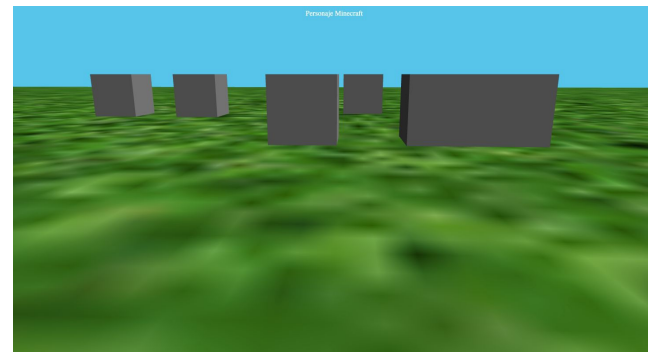


Fig 4. Imagen de Creación de Cubos Minecraft

E.Texturas

En esta sesión se creó un cubo con la geometría cubo aplicando las diferentes texturas para así formar el cubo original del juego minecraft.

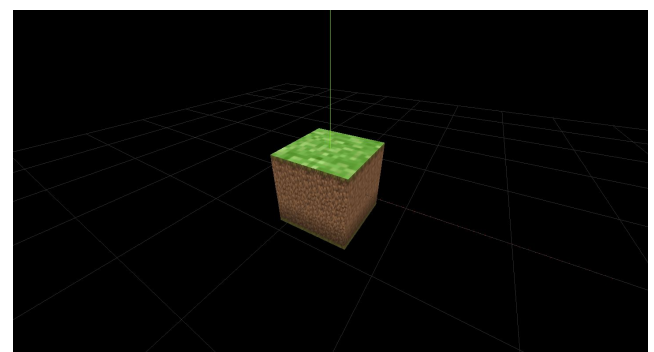


Fig 5. Imagen de textura del cubo Minecraft

Imágenes usadas como texturas en el cubo.

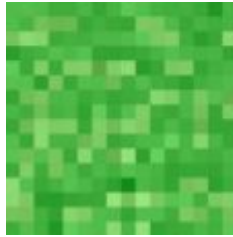


Fig 6 . Imagen de Textura de Pasto Minecraft

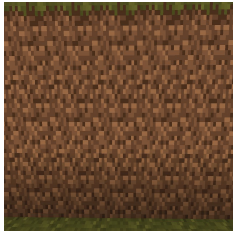


Fig 7 . Imagen de Textura de Tierra Minecraft

F.Escena General

Esta escena la usamos para las diferentes sesiones que son para emular el piso y el cielo de minecraft

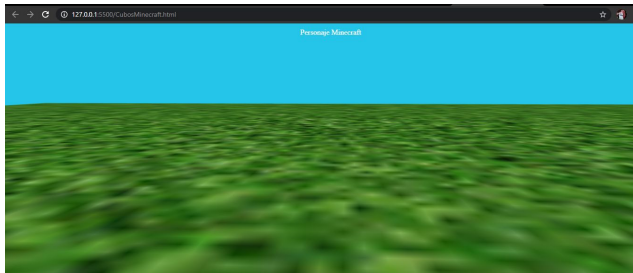


Fig 8 . Imagen de entorno Minecraft

Esta escena se hace con el efecto niebla usado en la parte del personaje



Fig 9 . Imagen de Niebla Minecraft

G.Escena Final

En esta sesión se observa toda una parte del mundo inicial del minecraft haciendo uso de las geometrías cubo , matrices ,ciclos y diferentes cosas mas. Lo tomamos de un video un youtube con su respectivo repositorio <https://www.youtube.com/watch?v=6tSg42ZIR6c>

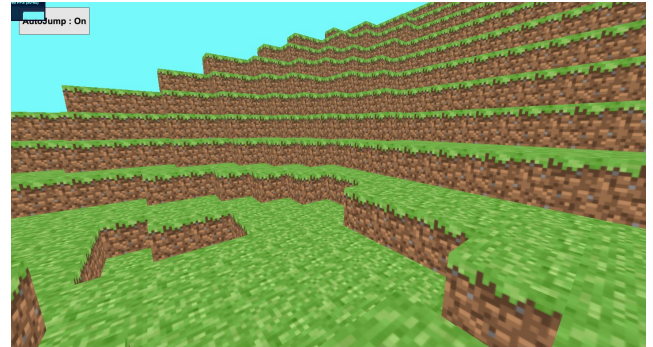


Fig 10 . Imagen de entorno final de Minecraft

V.DIAGRAMAS DE FLUJO

Describe el proceso para documentar, estudiar, planificar, mejorar.

En este diagrama se describe el menú Principal.

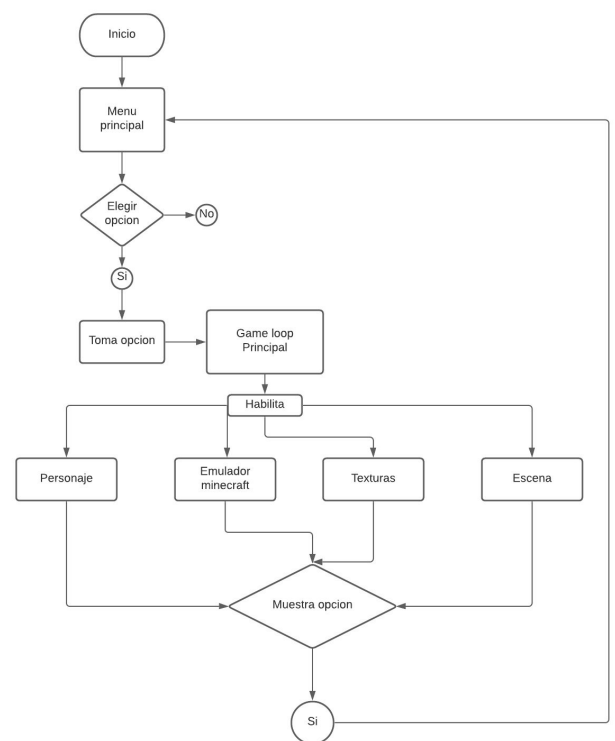


Fig 11 . Diagrama de flujo menú principal Minecraft

En este diagrama hacemos la descripción gráfica del Personaje.

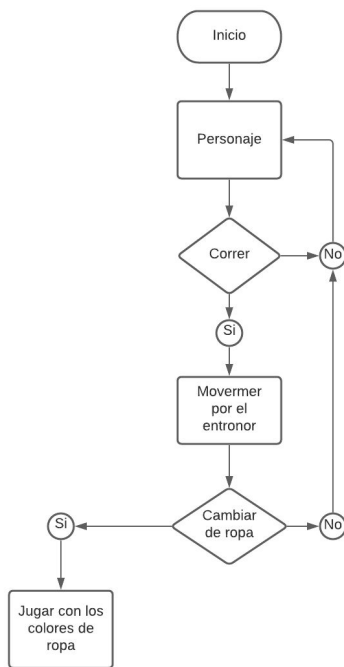


Fig 12 . Diagrama de flujo personaje de Minecraft

En este diagrama se describe el proceso del emulador de Minecraft

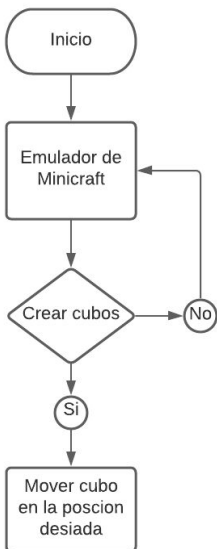


Fig 13 . Diagrama de flujo del emulador de Minecraft

En este de diagrama de la descripción de la escena final

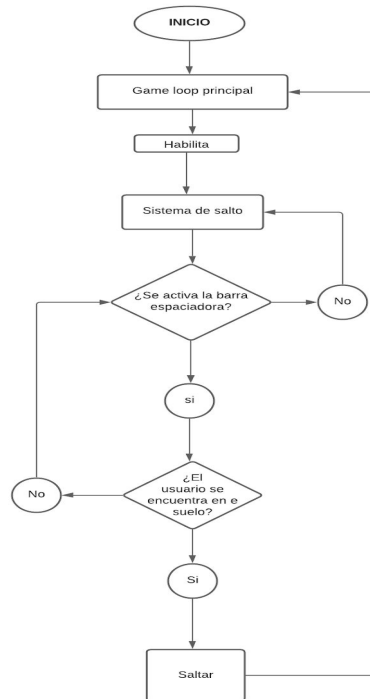


Fig 14 . Diagrama de flujo Escena final Minecraft

VI.DESARROLLO DE TRABAJO

A.Cronograma

Pudimos establecer la organización del trabajo dentro del proyecto con las diferentes actividades a entregar en las determinadas fechas lo cual nos permitió cumplirlas como se esperaban.

CRONOGRAMA		10-sep	17-sep	24-sep
FASES	Definir idea	Buscar id	Idea lista	
	Como?		Investigacion	
	1 Resumen, Introduccion, Objetivos		Resumen	
	2 Marco Teorico, Cronograma,		Marco Teorico	
Creacion	3 Alcance, Diseño		Alace y Diseño	
	Colusiones De Cubos			Entrega Investigacion
	Cracion de Camaras			
Codificacion	Implementacion diseño Minecraft			
	Pruebas			
	Cracion de Presentacion final			
Actividades Pendientes				
Entregas				

Fig 15 . Imagen de cronograma

B. Metodología

En el siguiente diagrama se describe en forma general el proceso del proyecto de minecraft

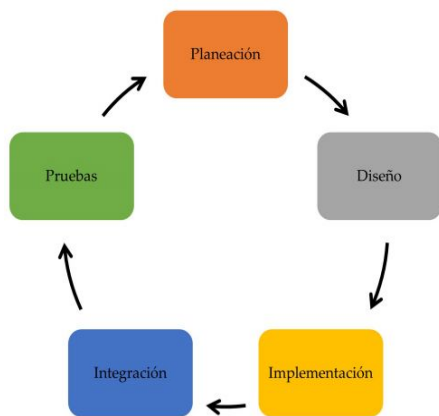


Fig 16 . Descripción de proceso general del proyecto

En este diagrama podemos observar de manera más detallada el proceso en el transcurso del proyecto

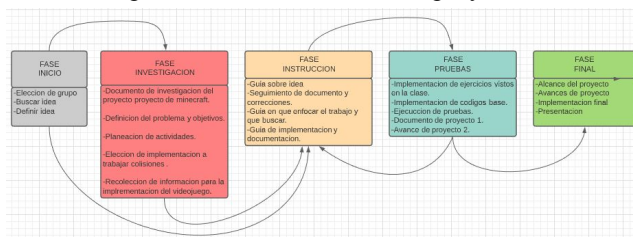


Fig 17 . Descripción de proceso detallado del proyecto



Fig 18 . Entorno de Trabajo del Proyecto

VII.CÓDIGO

La implementación del videojuego se hizo en el programa visual studio code con las diferentes extensiones como lo son go live, java, css , html y la librería de Three.js.

El código completo será anexado a una carpeta drive donde se podrá observar en su totalidad el desarrollo total del proyecto.

Aquí se deja una imagen del código donde se muestra una parte del desarrollo del videojuego.

https://github.com/Andres-Castro-C/Computacion_Grafica_2020-2/tree/master/Proyecto/Minecraft

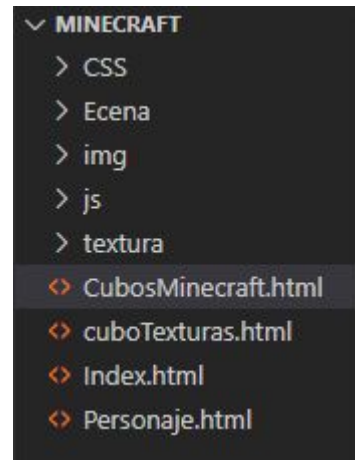


Fig 19 .Explorador de archivos del proyecto Minecraft

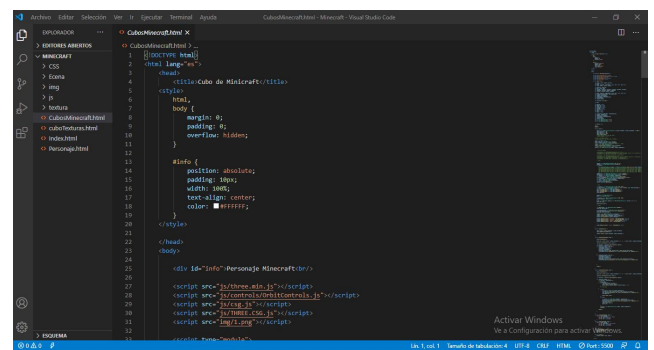


Fig 20 . Entorno de Desarrollo del Proyecto

VIII.BIBLIOGRAFÍA

[1]Mundos tridimensionales,(2017.5 octubre) [en línea]adictosalainformatica.com, disponible :<http://adictosalainformatica.com/category/3d/?print=pdf-search>

[2]Any Object3D?,¿Alguna forma de obtener un cuadro delimitador desde un objeto Object3D de three.js?,(2013, 4 abril)[en

línea],disponible:<https://stackoverflow.com/es/q/4203998>

[3]Colliders: detección de colisiones en videojuego Unity 3D – Academia Android,(2016,24 abril),[en

línea]Academiaandroid.com,disponible:[https://academiaan](https://academiaandroid.com/colliders-deteccion-colisiones-juego-unity-3d/)

droid.com/colliders-deteccion-colisiones-juego-unity-3d/

[4]Colliders: detección de colisiones en videojuego Unity 3D – Academia Android,(2020, 10 enero),[en

línea]Academiaandroid.com,disponible:[http://saber.ucv.ve/](http://saber.ucv.ve/bitstream/123456789/15018/1/Thesis.pdf)

bitstream/123456789/15018/1/Thesis.pdf

[5]Diseño de juegos 3d para web,(2005, 6 mayo),[en

línea]Thefiveplanets.org,disponible:[https://thefiveplanets.org/wp-content/uploads/2016/09/thef](https://thefiveplanets.org/wp-content/uploads/2016/09/thefiveplanets-preview.pdf)

iveplanets-preview.pdf

[6]Descubre que es Minecraft y por que todo el mundo

habla de él,(2016, 2 febrero),[en

línea]Thefiveplanets.org,disponible

<https://www.educaciontrespuntocero.com/noticias/que-es-minecraft-educacion/>