



TÓPICOS DE FÍSICA

Documentación – Proyecto 1er Parcial

López Bañuelos Karen Daniela

Maestro: Jesús Francisco Caro Cota

189368

Ing. en Producción Multimedia

19 de febrero del 2020

Introducción

Durante este semestre en la materia de Tópicos de física, estamos aprendiendo a como emplear nuestros conocimientos ya adquiridos sobre este concepto, estamos aplicándolo en el programa de Unity, que es con el que estamos trabajando actualmente.

Comenzamos aprendiendo sobre Unity y Unity Hub, que es el que estamos utilizando hoy en día, seguimos con los comandos de git, para poder guardar y actualizar la información con la que estamos trabajando.

Estamos aprendiendo todos los conceptos principales y más característicos de esta aplicación, sobre como se utilizan cada uno de los comando que contiene y entender el porque pasa, para así comprenderlos y poder emplearlos en un futuro.

En este parcial, trabajamos con un proyecto, acerca de aplicar los efectos de los cuales aprendimos estos días a los sprites que fueran de nuestro gusto, para ello a continuación se muestra cada una de las definiciones que estuvimos aprendiendo, así como también los conceptos acerca de los efectos que utilizamos en este proyecto.

Desarrollo

En este semestre, en la materia de Tópicos de física, hemos trabajado en el parcial con el programa de Unity, donde en lugar de aprender sobre física, ahora la estamos aplicando con los conocimientos base que ya tenemos, los empleamos para hacer modelos en 2D para después construir a base de la física, videojuegos, material o animación.

Unity es una herramienta de desarrollo de videojuegos creada por la empresa Unity Technologies. En la página web de Unity, en la sección Made with Unity, podemos ver que este software se ha utilizado para crear multitud de juegos conocidos y otros no tan conocidos. También se ha utilizado para crear experiencias de Realidad Virtual interactivas e incluso miniseries. Una de las características más importantes y cómodas de Unity es que soporta la exportación a una cantidad enorme de plataformas.

Nosotros hemos trabajado con Unity, pero en la plataforma de Unity Hub. Unity Hub es una aplicación independiente que simplifica la forma en que encuentra, descarga y administra sus proyectos e instalaciones de Unity. Además, puede agregar manualmente a su Hub versiones del Editor que ya haya instalado en su máquina.

Se puede usar el Hub para:

- Administre su cuenta de Unity y las licencias de editor.

- Cree su proyecto, asocie una versión predeterminada del editor de Unity con el proyecto y gestione la instalación de varias versiones del editor.
- Inicie diferentes versiones de Unity desde su vista de Proyecto
- Administre y seleccione Objetivos de compilación del proyecto sin iniciar el Editor.
- Ejecute dos versiones de Unity al mismo tiempo.
- Agregue componentes a las instalaciones existentes del Editor.

Para el desarrollo de nuestras actividades, fue necesario la utilización de **git** y **powershell**, para poder seguir trabajando con el proyecto a largo plazo y subirlo a **Github**, la plataforma

GitHub es un sistema de gestión de proyectos y control de versiones de código, así como una plataforma de red social diseñada para desarrolladores. Permite trabajar en colaboración con otras personas de todo el mundo, planificar proyectos y realizar un seguimiento del trabajo. **Git**, es un software de control de versiones diseñado por Linus Torvalds. Se define como control de versiones a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo es decir a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración, repositorios online más grandes de trabajo colaborativo en todo el mundo. **Windows PowerShell** es un shell de línea de comandos, un lenguaje de scripting y un entorno para la automatización de tareas creado por Microsoft.

Todo lo que hemos trabajado en este parcial, se llama computer graphics. El campo de los **computer graphics** es un campo amplio y diverso que existe en sección transversal entre la informática y el diseño. Está interesado en todo el proceso de creación de imágenes generadas por computadora, desde la creación de modelos digitales tridimensionales, hasta el proceso de texturizar, renderizar e iluminar esos modelos, hasta la visualización digital de esas representaciones en una pantalla.

Para crear un nuevo proyecto en Unity Hub, tenemos que estar registrados en la plataforma y después activar una licencia, por ser estudiantes, se puede activar una licencia personal, para que sea gratis sin complicaciones. Con los proyectos que nosotros hemos trabajado, se llaman **LWRP Template**.

La **Lightweight Pipeline** es una tubería de renderizado programable disponible con Unity 2018.1. La **LT Pipe** realiza un renderizado directo de un solo paso con eliminación de luz por objeto con la ventaja de que todas las luces están sombreadas en un solo paso. En comparación con el renderizado de vanilla unity forward, que realiza un pase adicional por píxel de luz, el uso de la tubería LT dará como resultado menos llamadas de extracción a expensas de la complejidad del sombreador ligeramente adicional. La tubería admite como máximo 8 luces por objeto y solo admite un subconjunto de características de renderizado de Unity.

En Unity, empezamos trabajando con **Shaders**, un **shader** es una aplicación que se carga en el GPU se utiliza para cargar texturas, deformar modelos, mostrar sprites, mostrar efectos especiales, etc. También aplicamos los **materiales** y las **texturas**. **Materiales** se define cómo debe renderizarse una superficie, incluyendo referencias a las Texturas que utiliza, información de tiling, tintes de color y más. Las opciones disponibles para un Material dependen del Shader que el Material esté usando. **Shaders** son pequeños scripts que contienen los cálculos matemáticos y algoritmos para calcular el Color de cada píxel procesado, en función de la entrada de iluminación y la configuración del Material. **Texturas** son imágenes bitmap. Un Material puede contener referencias a las texturas, de modo que el Shader del Material pueda usar las texturas al calcular el color de la superficie de un GameObject. Además del color básico (Albedo) de la superficie de GameObject, las texturas pueden representar muchos otros aspectos de la superficie de un material, como su reflectividad o rugosidad. Un material especifica un shader específico para usar, y el shader utilizado determina qué opciones están disponibles en el material. Un Shader especifica una o más variables de Textura que espera usar, y el Inspector de Materiales en Unity le permite asignar sus propios Elementos de Textura a estas variables de Textura.

En los trabajos que hemos realizado en clase, hemos utilizados ShaderGraph y PBR. Un **ShaderGraph** le permite construir sus sombreadores visualmente. En lugar de escribir código a mano, crea y conecta nodos en una red gráfica. Se puede crear shaders fácilmente desarrollándolos de manera visual y consultar los resultados en tiempo real.

Para poder trabajar en Unity con Shaders o empezar hacer algún modelado, es necesario insertar una imagen o textura que es con la que vamos a trabajar, para empezar a darle diferentes características como el outline o entintado, por ejemplo.

Para ello utilizamos un **Sprite**, el cual consiste en la máscara o la imagen que el desarrollador de videojuegos asigna a un objeto que puede colisionar, es una imagen usada para representar un ente gráficamente (o parte de él) y poder posicionarlo en el lugar deseado de una escena mayor. Mediante este sistema además se pueden crear animaciones que representen dicho ente cambiando el sprite correspondiente.

Estuvimos trabajando con distintos sprites, los cuales bajamos de internet, estos se pueden descargar desde páginas especiales, o simplemente de internet buscando la imagen deseada. La mayoría de las veces lo que vamos a descargar es una imagen en png, con transparencia, pero cuando lo mandamos a Unity en el programa, debemos cambiarle la configuración como **Sprite**, para que lo lea y nos deje trabajar como se debe en él.

En este proyecto de Unity aplicamos diferentes temas, **el entintado, outline, fresnel y glitch**.

El **entintado** como la palabra lo dice entinta el sprite, es decir, tiene la característica de poder cambiar de color el sprite que estamos utilizando, entintarlo de un solo color por encima de él, cambiando el Alpha.

El **outline** agrega un efecto simple de contorno a componentes gráficos tal como Text o Image. Debe estar en el mismo GameObject que el componente gráfico.

El **Efecto Fresnel** es la observación de que la cantidad de reflectancia que se ve en una superficie depende del ángulo de visión. Los sombreadores de Fresnel en general permiten que la reflexión, la especularidad y otros atributos varíen según el ángulo de visión de una superficie 3D. Un sombreador de Fresnel le permitirá especificar un color especular para partes de una superficie directamente frente a la cámara, y otro color especular que se verá en partes de una superficie que son perpendiculares a la cámara.

El efecto **glitch** es conocido también como el efecto holograma y este hace que tu sprite tenga un tipo de distorsión en la textura como efecto, que le permita tener apariencia como si fuera un holograma.

En este proyecto que hicimos para el parcial, combinamos estos efectos en los shadergraphs con los sprites. Hicimos un solo shader para el outline y el entintado para que estuvieran juntos, esto mismo ocurrió con el fresnel y glitch, que también los unimos en uno sólo.

Conclusión

Para comprender bien el tema y saber usar bien el programa de Unity, el principal secreto es conocer que es cada uno de los comandos o nodos que hay y que se utilizan para la realización de cada efecto, ya que con ello se puede crear un mejor conocimiento acerca de lo que se está trabajando y poder comprender el tema.

Realmente lo que fue este proyecto, ya lo habíamos hecho como actividades en clase, cada uno de los efectos que se pidió, pero el reto de ahora era poder juntarlos en sí, en un mismo shadergraph para que el sprite tuviera dos efectos a la vez.

Los efectos no son tan complicados cuando se logra entender lo que se está trabajando y el porque de cada una de las cosas, este proyecto fue un reto más para analizar y comprender lo que estuvimos viendo en todo el parcial.

Aprender acerca de estos efectos que vimos, nos ayuda mucho ya que son de las características más principales que se le ponen a un sprite, antes de animarlo o pasar al siguiente paso, es esencial conocer esto, para poder después aplicar lo aprendido sin duda ni complicaciones y lograr un mejor resultado.

Referencias bibliográficas

David Erosa García. (2019). ¿Qué es unity?. 18 de febrero del 2020, de OpenWebinars. Sitio web: <https://openwebinars.net/blog/que-es-unity/>

Unity Hub (2019). 18 de febrero del 2019, de Unity Sitio web: <https://docs.unity3d.com/Manual/GettingStartedUnityHub.html>

Gustavo B. (2019). ¿Qué es GitHub y para qué se utiliza? 18 de febrero del 2020, de Hostinger Tutoriales Sitio web: <https://www.hostinger.mx/tutoriales/que-es-github/>

Ándres. (2015). ¿Qué es git?. 18 de febrero del 2020, de Codigo facilito Sitio web: <https://codigofacilito.com/articulos/que-es-git>

What is Computer Graphics? (s.f). 18 de febrero del 2020, de Cornell CIS - Program of computer graphics Sitio web: <https://www.graphics.cornell.edu/about/what-computer-graphics>

Davide Costantini. (2018). PowerShell: una guía completa – parte 1. 18 de febrero del 2020, de The Solving Sitio web: <https://thesolving.com/es/sala-de-servidores/powershell-una-guia-completa-parte-1/>

Lightweight Pipeline. (s.f.). 18 de febrero del 2020, de Unity Sitio web: <https://docs.unity3d.com/Packages/com.unity.render-pipelines.lightweight@4.6/manual/index.html>

Adolfo. (2019). Programación de Shaders: ¿Que es un shader?. 18 de febrero del 2020, de Tejón Procedural Sitio web: <https://tejonprocedural.com/2019/05/13/programacion-de-shaders-que-es-un-shader/>

Materiales, Shader y Texturas. (s.f.). 18 de febrero del 2020, de Unity Sitio web: <https://docs.unity3d.com/es/2018.4/Manual/Shaders.html>

Shader Graph. (s.f.). 18 de febrero del 2020, de Unity Sitio web: <https://unity.com/es/shader-graph>

Raquel Varela. (2019). Sprites. 18 de febrero del 2020, de Geekno Sitio web: <https://www.geekno.com/glosario/sprites>

Sprite. (s.f.). 18 de febrero del 2020, de Fandom Sitio web: <https://videojuegos.fandom.com/es/wiki/Sprite>

Outline (contorno). (s.f.). 18 de febrero del 2020, de Unity Sitio web: <https://docs.unity3d.com/es/2018.4/Manual/script-Outline.html>

Dorian Iten. (s.f.). The Fresnel Effect. 18 de febrero del 2020, de Dorian Iten Sitio web: <https://www.dorian-iten.com/fresnel/>

Jeremy Birn. (s.f.). Fresnel Effect. 18 de febrero del 2020, de 3drender Sitio web: <http://www.3drender.com/glossary/fresneleffect.htm>

NSDG. (2019). Efecto Holograma en Unity 3D. 18 de febrero del 2020, de Sharp Coder Sitio web: <https://sharpcoderblog.com/blog/create-a-hologram-effect-in-unity-3d>

Texturas 2D. (s.f.). 18 de febrero del 2020, de Unity Sitio web: <https://docs.unity3d.com/es/2018.4/Manual/Textures.html>

Working with Shader Graph. (s.f.). 18 de febrero dle 2020, de Unity Sitio web: <https://docs.unity3d.com/Packages/com.unity.render-pipelines.lightweight@6.7/manual/ShaderGraph.html>

Konosoke. (s.f.). SHADERS: ESOS QUE TODOS NOMBRAN Y POCOS SABEN LO QUE SON. 18 de febrero del 2020, de Citro Gamers Sitio web: <http://citrogamers.blogspot.com/2015/09/shaders-esos-que-todos-nombran-y-pocos.html>