Contenido

[1. Autores del trabajo, planificación y entrega 2](#_30j0zll)

[1.1 Autores 2](#_1fob9te)

[1.2 Planificación 2](#_3znysh7)

[1.3 Entrega 2](#_2et92p0)

[2. Requisitos del prototipo a implementar 3](#_tyjcwt)

[2.1 Requisitos funcionales 3](#_3dy6vkm)

[2.2 Otros requisitos 3](#_1t3h5sf)

[3. Criterios de comparación en la implementación 4](#_4d34og8)

[3.1 Criterio 1: Nombre del criterio 4](#_2s8eyo1)

[3.2 Criterio 2: Nombre del criterio 4](#_17dp8vu)

[3.N Criterio N: Nombre del criterio 4](#_3rdcrjn)

[4. Proyecto de implementación de un prototipo del sistema utilizando la tecnología A 5](#_26in1rg)

[4.1 Documentación de diseño 5](#_lnxbz9)

[4.2 Documentación de construcción 5](#_35nkun2)

[4.3 Documentación de pruebas 5](#_1ksv4uv)

[4.4 Documentación de instalación 5](#_44sinio)

[4.5 Manual de usuario 5](#_2jxsxqh)

[5. Proyecto de implementación de un prototipo del sistema utilizando la tecnología B 6](#_z337ya)

[5.1 Documentación de diseño 6](#_3j2qqm3)

[5.2 Documentación de construcción 6](#_1y810tw)

[5.3 Documentación de pruebas 6](#_4i7ojhp)

[5.4 Documentación de instalación 6](#_2xcytpi)

[5.5 Manual de usuario 6](#_1ci93xb)

[6. Comparación de las dos implementaciones 7](#_3whwml4)

[6.1 Evaluación de los criterios en la implementación usando la tecnología A 7](#_2bn6wsx)

[6.2 Evaluación de los criterios en la implementación usando la tecnología B 7](#_qsh70q)

[7. Comparación de la implementación de las tecnologías 8](#_3as4poj)

[8. Conclusiones 10](#_1pxezwc)

# 1. Autores del trabajo, planificación y entrega

## 1.1 Autores

GRUPO M2

Víctor Parrilla López.

Tomás Rubio del Saz.

Francisco José Martínez Bartolomé.

Diego Gerardo Estalrich Cuéllar.

Alejandro San Roque Emery.

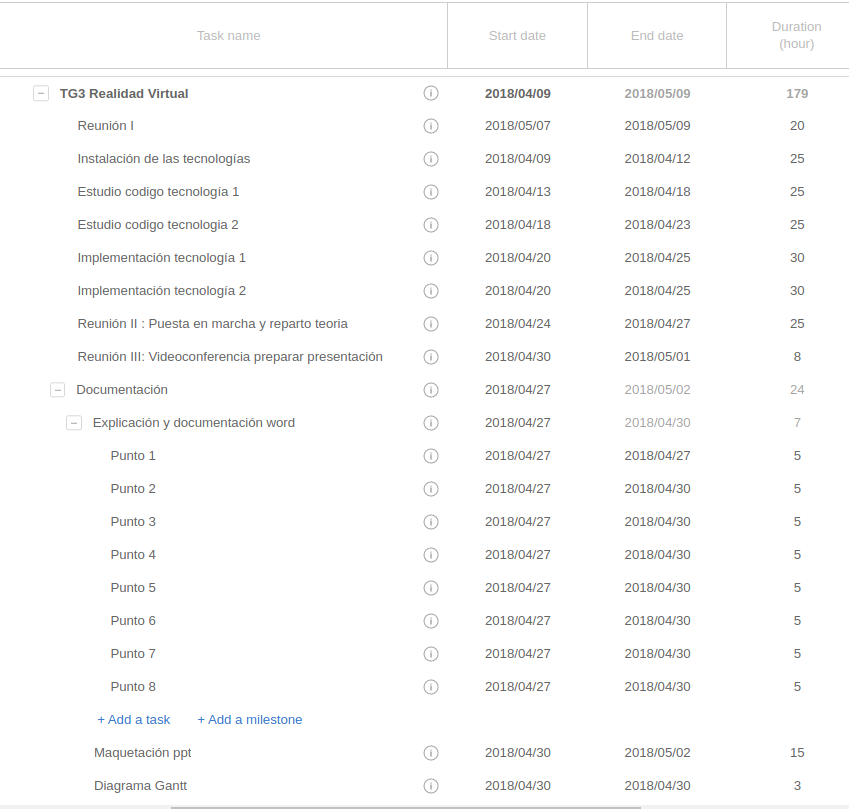
## 1.2 Planificación

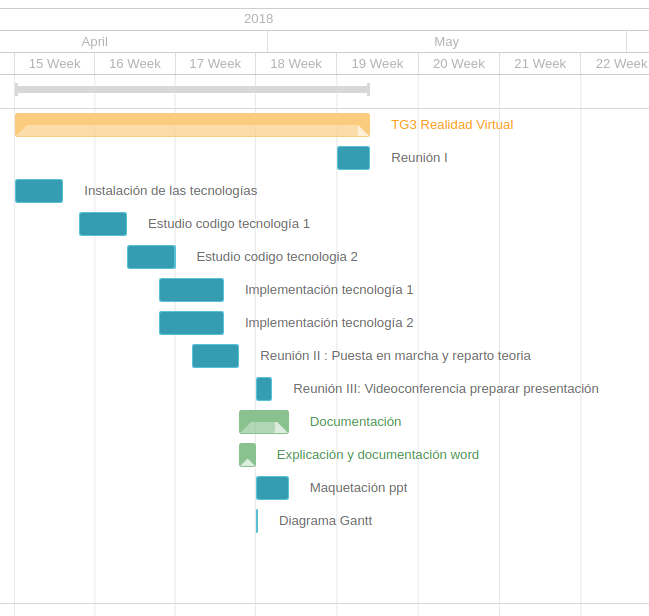
La planificación de este trabajo se puede apreciar en el diagrama de Gantt creado con Ganttpro accediendo al siguiente enace<https://app.ganttpro.com/#!/app/home> y compartido además con el profesor de la asignatura.

El presente trabajo “TG3” esta planificado sobre un mínimo de 225 horas totales, que corresponden a una media de 45 horas por cada integrante del grupo.

Pasaremos a explicar los puntos más relevantes de la planificación:

* En el diagrama Gantt no están divididas las tareas por las personas del equipo, ya que por las características del trabajo la mayoría de las tareas se han realizado de manera conjunta, de tal manera que todos llos participantes del grupo han colaborado en todas las tareas. Solo destaca la planificación y el presente diagrama Gantt que han sido creados por el coordinador de grupo.
* Para la realización del trabajo se han realizado tres reuniones grupales, siendo dos presenciales en la universidad y la ultima por videoconferencia.
* El tiempo invertido para poder implementar un prototipo en cada una de las tecnologías ha sido lo más destacable en cuanto al tiempo real utilizado ya sea por el desconocimiento inicial en la implementación de las tecnologías o por la falta de soltura a la hora de implementación de plataformas de realidad virtual.
* El uso del hardware usado para la implementación del trabajo ha sido limitado, pudiéndose usar solo por una persona a la vez, lo que ha hecho más difícil la realización del presente trabajo.





## 1.3 Entrega

Enlace repositorio GitHUb: https://github.com/victorParrilla/TG3

Hay que tener en cuenta que varios de los apartados se han realizado en común por todos los miembros del equipo de trabajo, por lo que el documento aportado en cada archivo .docx de cada integrante del grupo no es del todo real al medir la tarea realizada por cada integrante.

# 2. Requisitos del prototipo a implementar

# 

En esta sección se mostrarán la lista de requisitos de los prototipos implementados con el fin de luego poder realizar una comparación mas real y exhaustiva. Esta sección está dividida entre “requisitos funcionariales” y “otros requisitos”.

* Los requisitos funcionales corresponden a aquellos requisitos que definen una función del sistema software.
* Los requisitos no funcionales no se centran en el comportamiento “el que hace” si no en el “como lo hace”, juzga la operación de un sistema. A su vez pueden dividirse, por ejemplo, requisito no funcional de seguridad

“Otros requisitos” son en su mayoría requisitos no funcionales, pero como veremos más adelante puede haber más tipos.

El objetivo del proyecto es comparar la implementación de un mismo prototipo de sistema utilizando dos tecnologías diferentes (A y B).

## 2.1 Requisitos funcionales

2.2 Requisitos de diseño de la aplicación

|  |  |
| --- | --- |
| **REQ.** | **DESCRIPCIÓN** |
| RF01 | El sistema moverá la cámara siguiendo el movimiento de la cabeza del usuario. |
| RF02 | El usuario podrá ver imágenes 360 del entorno. |
| RF03 | El usuario podrá salir de la aplicación en todo momento. |
| RF04 | El usuario podrá ocultar/mostrar el contenido gráfico del entorno cuando desee |
| RF05 | La aplicación no necesita de uso de Internet, por lo que se puede usar en todo momento |
| RF06 | El sistema reconoce ambos mandos correctamente |
| RF07 | La aplicación deberá dejar interactuar con los objetos de la escena a través de los mandos. |
| RF08 | El puntero de la aplicación generara continuamente raycast, que generarán eventos al colisionar con objetos válidos. |
| RF09 | El sistema generará dos imágenes desfasadas para que nuestro cerebro las una y genere una única imagen tridimensional, como la visión en el mundo real. |
| RF10 | El usuario podrá ver información del entorno. |
| RF11 | El usuario podrá ver imágenes del entorno. |

## 2.2 Otros requisitos

|  |  |
| --- | --- |
| **REQ.** | **DESCRIPCIÓN** |
| RNF01 | El sistema deberá ofrecer un buen rendimiento, cargando y respondiendo a las acciones del usuario en un mínimo de 1 segundo. |
| RNF02 | La aplicación debe ser fácil e intuitiva. |
| RNF03 | El sistema ha de presentar los elementos gráficos claramente visibles e identificables. |
| RNF04 | Actualmente se podrá ejecutar a partir de una tarjeta gráfica envidia geforce 1060(6gb) |
| RNF05 | El sistema deberá funcionar a más de 30 FPS. |
| RNF06 | EL sistema necesita soporte del controlador |
| RNF07 | La aplicación debe ser utilizada con unas gafas de realidad mixta Lenovo eXplorer |
| RNF08 | La aplicación ha de cumplir con los consejos de usabilidad de realidad virtual |

# 3. Criterios de comparación en la implementación

Se procede a definir los criterios de comparación de la implementación de las dos tecnologías, VR y Unreal, en cuanto a la construcción de los dos prototipos teniendo en cuenta os requisitos ya establecidos en el anterior apartado.

Se trata de criterios del tipo” “horas empleadas en el desarrollo del sistema”, “velocidad de funcionamiento del sistema”, “recursos necesarios”, etc.

## 3.1 Criterio 1: Tiempo de aprendizaje

*Nombre del criterio: Tiempo de aprendizaje*

*Descripción: Con este criterio valoraremos el tiempo previo usado para formarnos en la tecnología y el tiempo desde que arrancamos la aplicación hasta que se empieza con la implementación del diseño.*

*Tipo de valor: Numérico (Horas)*

**3.2 Criterio 2: Tiempo de respuesta en el desarrollo**

*Nombre del criterio: Tiempo de respuesta en el desarrollo*

*Descripción: Con este criterio se valora el tiempo de respuesta de cada una de las tecnologías a la hora de ser implementadas. Se valorará aquella tecnología con el menor tiempo en su implementación sin ser excluyentes ya que tendremos que tener en cuenta más criterios.*

*Tipo de valor: Numérico (segundos).*

**3.3 Criterio 3: Facilidad de implementación**

*Nombre del criterio: Facilidad de uso.*

*Descripción: Con este criterio se valora la facilidad de uso de cada una de las tecnologías.*

*Tipo de valor: Numérico. Escala del 0 al 10.*

**3.4 Criterio 4: Intuición de la Interfaz de usuario.**

*Nombre del criterio: Intuición de la interfaz de usuario.*

*Descripción: Con este criterio valoraremos lo intuitiva que es la interfaz de usuario en cada una de las tecnologías.*

*Tipo de valor: Texto. Se puede clasificar como nada intuitiva, poco intuitiva o muy intuitiva.*

**3.5 Criterio 5: Velocidad de respuesta**

*Nombre del criterio: Velocidad de respuesta*

*Descripción: Se analiza la velocidad de respuesta del prototipo a la hora de su utilización.*

*Tipo de valor: numérico. Escala del uno al 10.*

## 3.6 Criterio 6: Tiempo de configuración

## *Nombre del criterio:Tiempo de configuración*

## *Descripción: Analizaremos en tiempo empleado para configurar el programa para su utilización, incluyendo instalación.*

## *Tipo de valor: Numérico (horas).*

## 

## 3.7 Criterio 7: Calidad de la visibilidad

## *Nombre del criterio:Calidad de la visibilidad*

## *Descripción: Una vez el prototipo esté creado analizaremos la calidad de visibilidad. Esto significa cuál de los dos percibe un mayor realismo.*

## *Tipo de valor: Numérico (escala del 1 al 10).*

## 

***TABLA RESUMEN DE CRITERIOS***

|  |  |
| --- | --- |
| Número/Nombre Criterio | Descripción |
| Criterio 1: Tiempo de aprendizaje | Tiempo desde formación hasta implementación. |
| Criterio 2:Tiempo de respuesta en el desarrollo | tiempo de respuesta implementación. |
| Criterio 3: Facilidad de implementación | facilidad de uso de cada una de las tecnologías. |
| Criterio4:Intuición de la Interfaz de usuario | Intuición interfaz usuario. |
| Criterio 5: Velocidad de respuesta | velocidad de respuesta en utilización. |
| Criterio 6: Tiempo de configuración | *tiempo empleado instalación y configuración.* |
| Criterio 7: Calidad de la visibilidad | *calidad de visibilidad* |

# 4. Proyecto de implementación de un prototipo del sistema utilizando Unity 3D

## 4.1 Documentación de diseño

Se ha decidido realizar un entorno virtual sencillo para verlo con las gafas de realidad virtual mixta - Lenovo Explorer.

Para poder realizar la construcción del entorno virtual, se han utilizado los propios recursos gratuitos que tiene Unity, modelados sencillos y compatibles.

## 4.2 Documentación de construcción

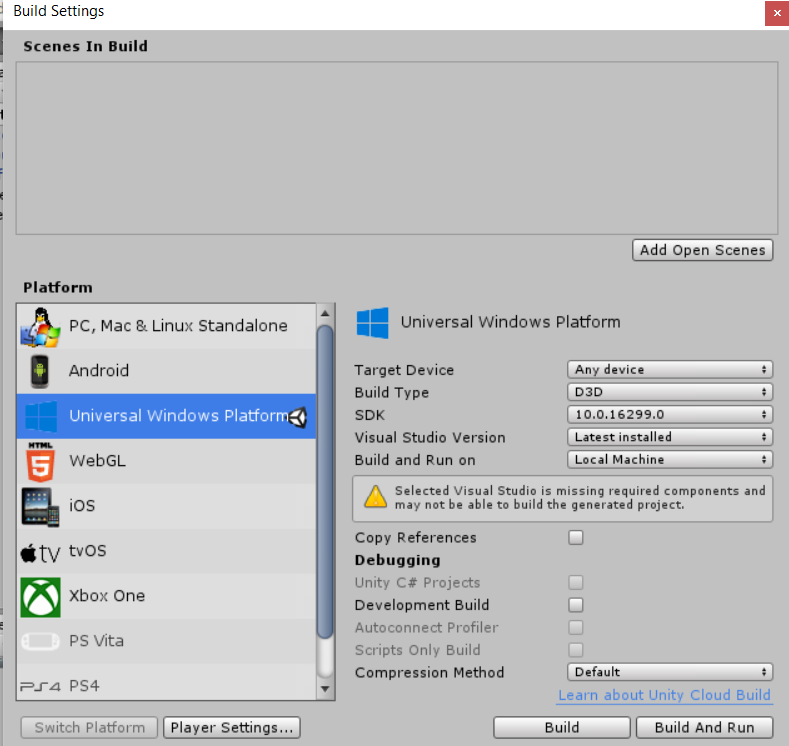
1. Uno de los grandes problemas que hemos encontrado ha sido poder integrar el modelo de gafas Lenovo Explorer de forma nativa, por lo que hemos tenido que recurrir a una búsqueda en profundidad por dentro de los foros de Unity, para poder conseguir los plugins y herramientas necesarias para una correcta implementación.
2. De esta forma ha podido ser todo más sencillo, ya que prácticamente es como un SDK con el que nos dá libertada para poder desarrollar lo que necesitemos.

## 4.3 Documentación de pruebas

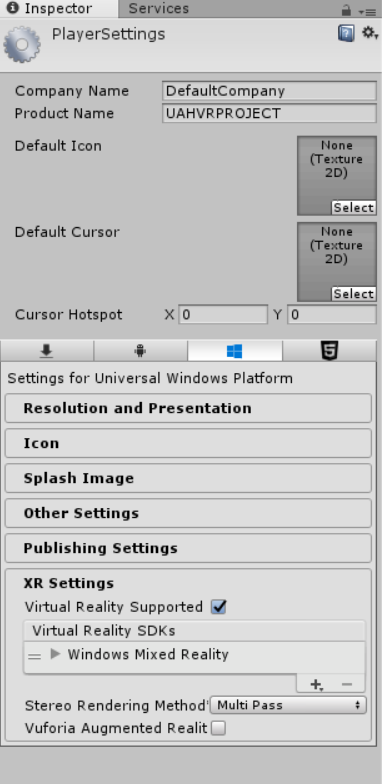
Para la fase de pruebas hemos utilizado el toolkit interno de Unity para poder compilarlo y probar en tiempo real.

## 4.4 Documentación de instalación

Lo primero que debemos hacer es instalar el paquete de Unity 3D, que lo encontramos en la página oficial de Unity (<https://store.unity.com/es/download?ref=personal>). **Es importante a la hora de instalar que se seleccione Universal Windows Plataform y que el SDK de compilación se haya instalado de forma correcta.**

****

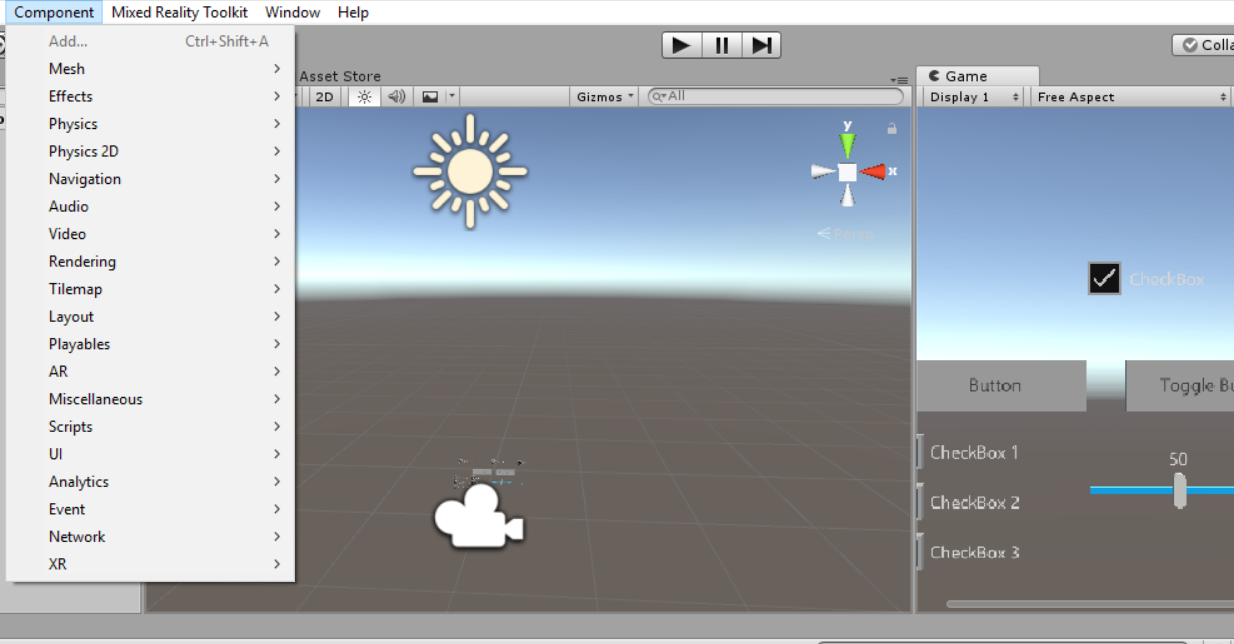
Debemos seleccionar **Player Settings** y comprobar que en **XR Settings** de nuestro proyecto está marcado “Virtual Reality Supported” y que aparece el SDK de Windows Mixed Reality.

****

Una vez que comprobamos que todo está correcto, ya podemos realizar la compilación de nuestro proyecto con las gafas conectadas. Automáticamente se arrancará el servicio de Windows Mixed Reality y mostrará abrirá la aplicación creada.

## 4.5 Manual de usuario

Para crear nuestro diseño en VR podemos utilizar cualquier herramienta de edición 3D que Unity trae por defecto, de manera que para crear un entorno en realidad virtual es relativamente sencillo.



Desde el menú de Component, podemos añadir cualquier Asset, objeto, cámara o efecto que necesitemos para nuestro proyecto. Simplemente seleccionando lo que necesitemos automáticamente se pondrá en nuestra escena. En una parte tenemos la ventana de escena, que es donde creamos nuestro escenario, y la otra ventana es la ventana de Game, que nos muestra lo que veremos una vez compilado y ejecutado el proyecto.

Al tratarse de una experiencia en VR no se puede materializar sin que sea en persona o en vídeo 360.

# 5. Proyecto de implementación de un prototipo del sistema utilizando Unreal Engine

## 5.1 Documentación de diseño

Creación de un escenario virtual en Unreal Engine en el que podamos estar inmersos dentro de él.

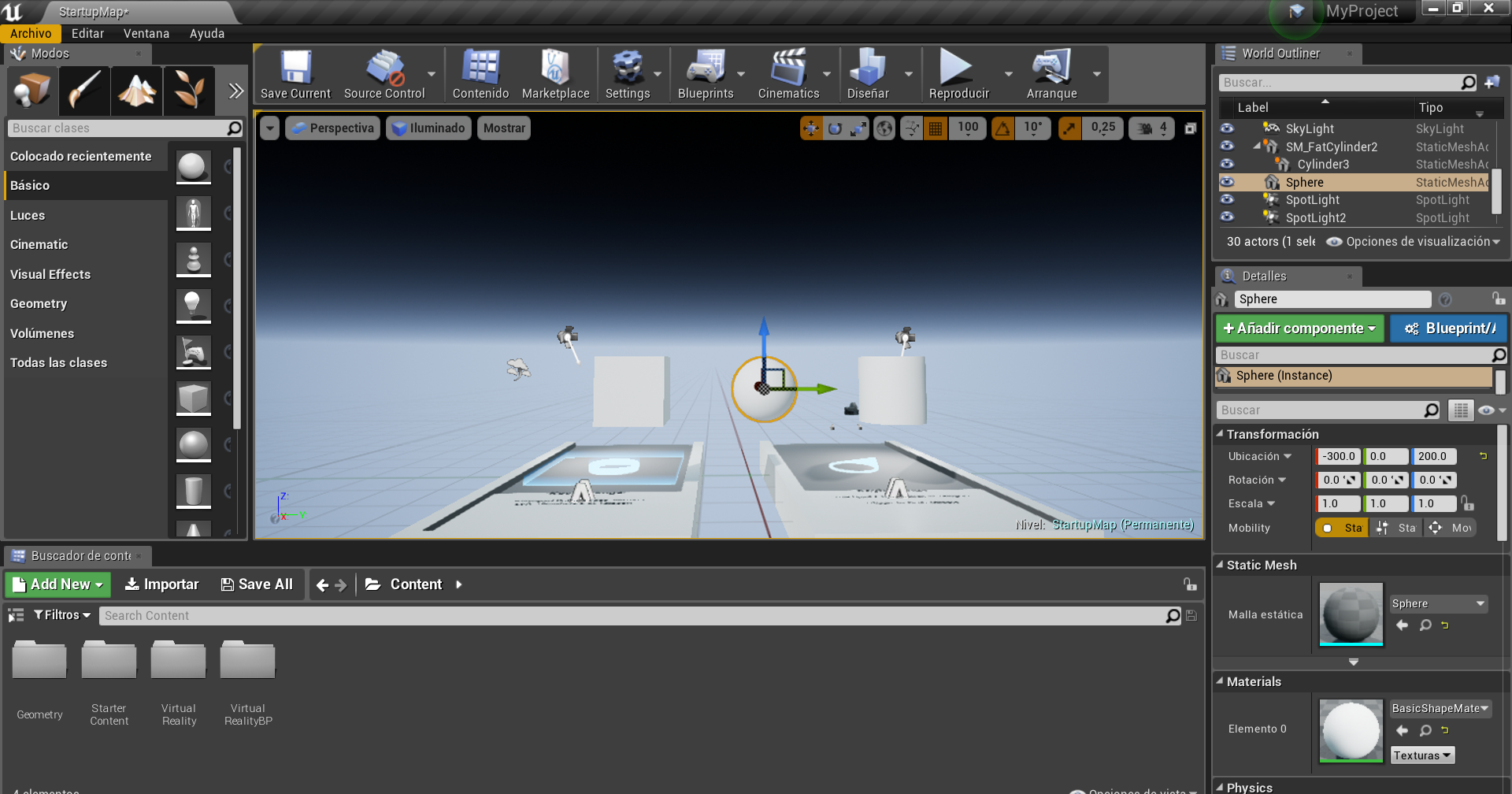
Para el diseño del escenario utilizaremos las herramientas 3D que vienen integradas dentro de Unreal Engine, de forma que podamos interactuar con el entorno en tiempo real.

Sin duda el potencial de Unreal a la hora de crear entornos en VR es superior, ya que de forma nativa nos implementa los controladores y plugins necesarios adaptandolos a cualquier tipo de gafa de realidad virtual. Apoyándose en el controlador de Steam VR, necesario para poder trabajar en Unreal.

En este caso, al igual que en Unity utilizaremos las formas geométricas que encontramos dentro de propio software.

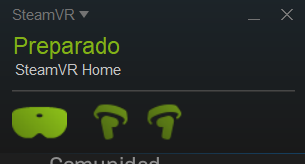
## 5.2 Documentación de construcción

Para ello, utilizaremos las herramientas que nos aparecen por defecto y que vienen integradas con el programa.

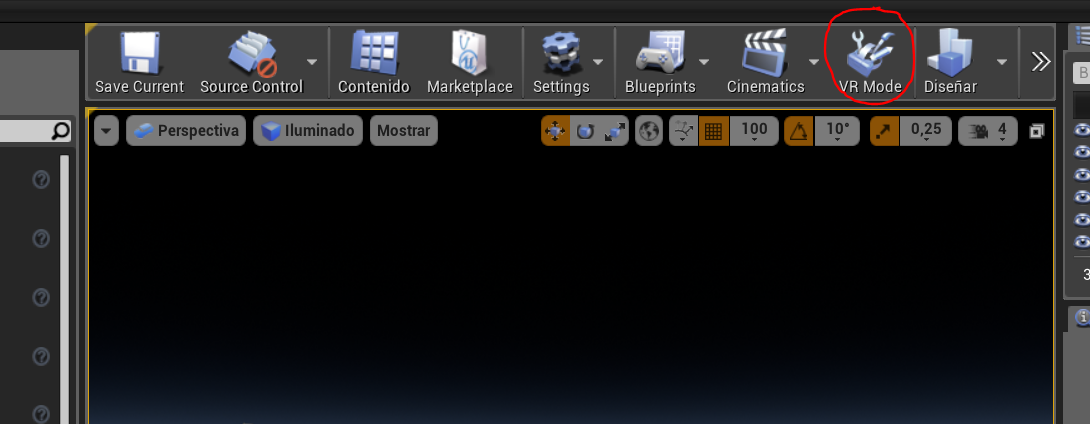


## 5.3 Documentación de pruebas

Para poder realizar la compilación y ejecución de forma correcta del proyecto VR es necesario tener conectadas las gafas desde el momento que abres Unreal para que cargue directamente la herramienta de Steam VR, y así se puedan visualizar las opciones de realidad virtual.



Una vez SteamVR preparado podemos acceder al modo VR dentro de Unreal.



No tendremos que hacer nada más que cargar nuestro proyecto (plantilla en VR) y añadirá de forma automática todos los controladores necesarios para detectar las gafas y los mandos. A diferencia con Unity, aquí los mandos se identificarán como si fueran los de las gafas de HTC VIVE que son compatibles con Realidad Mixta. Al tratarse de una sensación en VR, se presentará en persona o en vídeo 360.

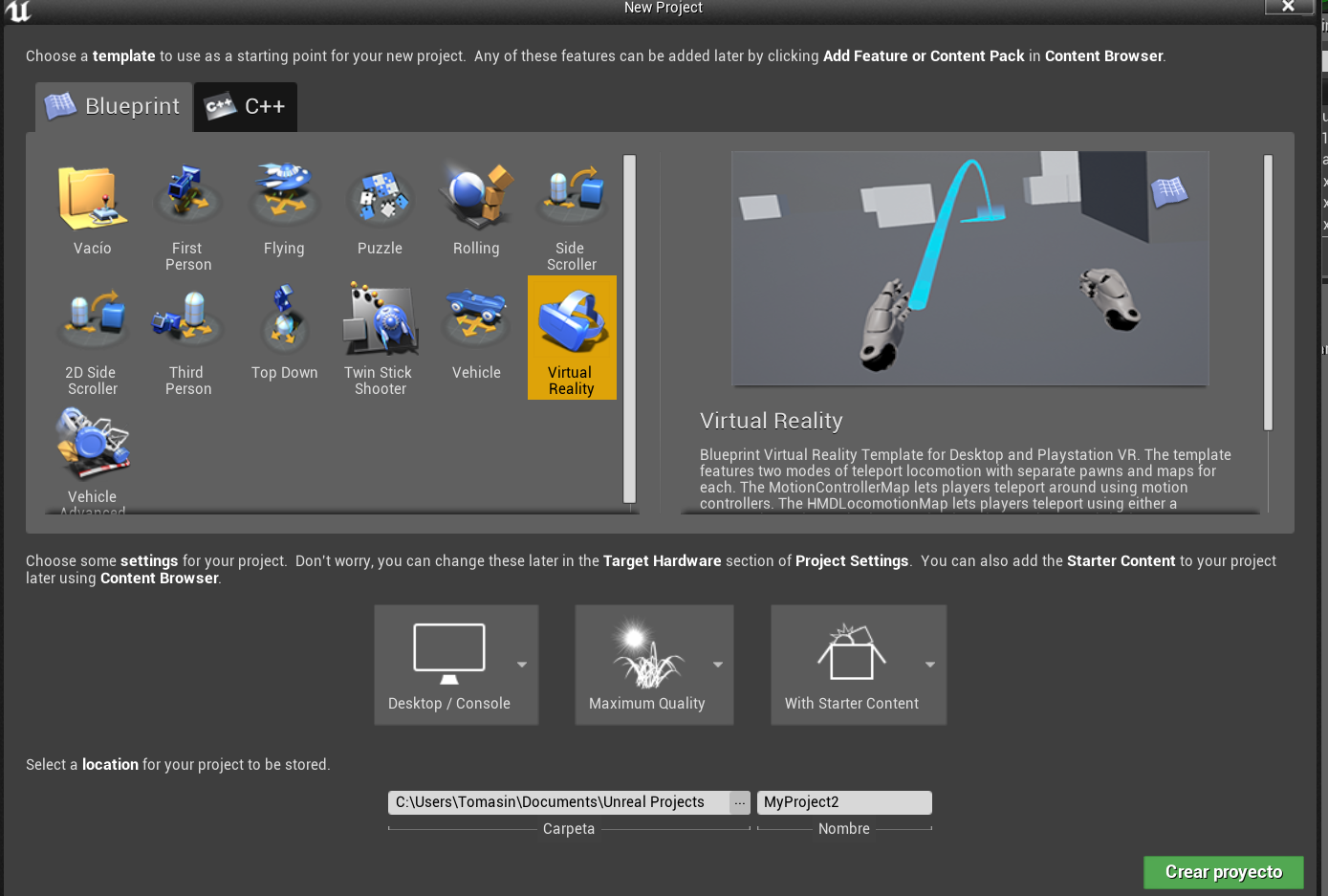
## 5.4 Documentación de instalación

Para poder utilizar Unreal, accedemos a la página web, y nos descargamos el programa de forma gratuita (<https://www.unrealengine.com/en-US/eulacheck?state=https://www.unrealengine.com/en-US/what-is-unreal-engine-4>). Sólo es necesario crearse una cuenta, que es gratuita.

Es un proceso de instalación sencillo en el que únicamente tenemos que tener en cuenta de instalar los samples de VR si queremos empezar de cero y así tener una referencia. Está totalmente en castellano, pero la documentación en su web está en inglés. A diferencia de Unity que encontramos toda la documentación en español.

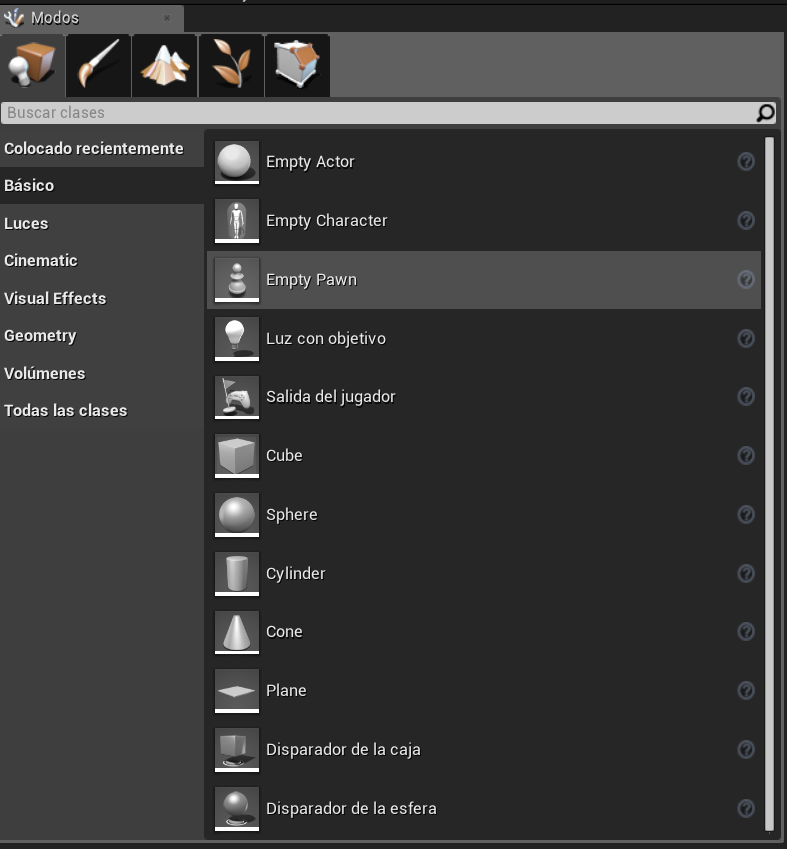
Para crear nuestro proyecto de VR desde cero, sólo necesitamos seguir estos pasos.

Archivo→ Nuevo proyecto → Seleccionamos Virtual Reality, seleccionamos la localización del proyecto y el nombre que le asignamos para guardar → Crear proyecto.



## 5.5 Manual de usuario

Pudiendo añadir nuestros propios diseños o editar los que vienen por defecto. Desde Objetos básicos, luces, cinemáticas, efectos visuales, figuras geométricas o volúmenes.



# 6. Comparación de las dos implementaciones

## 6.1 Evaluación de los criterios en la implementación usando la tecnología Unity 3D

|  |  |
| --- | --- |
| **CRITERIO** | **EVALUACIÓN** |
| Tiempo de aprendizaje | 60 horas |
| Tiempo de respuesta en el desarrollo | 20 horas |
| Facilidad de implementación | 7, al principio cuesta configurar la herramienta y tener todo listo de pluggins y utilidades extra para que todo sea compatible y funcional con el hardware de VR |
| Intuición de la Interfaz de usuario | Muy intuitiva |
| Velocidad de respuesta | 8 |
| Tiempo de configuración | 20 horas, hay varios drivers/pluggins y funcionalidades que hay que buscar e instalar manualmente y documentarse sobre como hacerlo. |
| Calidad de la visibilidad | 7, la calidad ofrecida por la tecnología no es mala ni extremadamente buena, depende de la calidad y perfección que le quiera dar la persona encargada en el apartado visual para obtener resultados óptimos. |

## 6.2 Evaluación de los criterios en la implementación usando la tecnología Unreal Engine

## 

|  |  |
| --- | --- |
| **CRITERIO** | **EVALUACIÓN** |
| Tiempo de aprendizaje | 55 horas |
| Tiempo de respuesta en el desarrollo | 10 horas |
| Facilidad de implementación | 9 |
| Intuición de la Interfaz de usuario | Muy intuitiva |
| Velocidad de respuesta | 7 |
| Tiempo de configuración | 5 horas, prácticamente con la instalación está todo listo y reconoce perfectamente el hardware con el que hay que trabajar, ya que utiliza las herramientas de Steam VR (HT Vive) que son compatibles con el hardware usado (VR Lenovo). |
| Calidad de la visibilidad | 9, posee utilidades y funcionalidades para un acabado óptico y luminoso bastante realista y potente. |

# 7. Comparación de la implementación de las tecnologías

|  |  |  |  |
| --- | --- | --- | --- |
| Criterios | Unity | Unreal | Comentarios |
| TIempo de aprendizaje | Curva de aprendizaje más corta y pronunciada. En 3 dias de aprendizaje hemos logrado hacer modelos sólidos | Curva de aprendizaje más larga. Hemos precisado de una semana aproximadamente hasta comenzar a hacer un modelo serio. | En Unity con menos conocimientos puedes hacer cosas de carácter avanzado |
| Tiempo de respuesta en el desarrollo | Unity tiene un tiempo de respuesta más lento (promedio), debido a que trabaja con algunos lenguajes interpretados. | Unreal ofrece mejores resultados al ofrecer posibilidades de desarrollo a bajo nivel de arquitectura, tiene un tiempo de respuesta más rápida. | Unreal, debido a su configuración más compleja, permite manipular parámetros a más bajo nivel que Unity, a costa de un mayor nivel de dificultad |
| facilidad de desarrollo | Unity es intuitivo, potente y ofrece un gran abanico de recursos para el desarrollo. | Unreal es una herramienta potente, pero compleja, lo que complica su desarrollo. | Unity es una herramienta más indicada a la hora de realizar modelos con conocimientos principiantes |
| facilidad de implementacion | Unity es una herramienta que ha demostrado ser muy polivalente, conservando su faceta de sencillez de uso, permitiendo crear modelos en distintos entornos sin demostrar muchas complicaciones | Unreal al ser una herramienta más compleja, ha reportado más problemas a la hora de desplegar modelos. |  |
| intuicion de la interfaz de usuario | La interfaz de Unity, considerando el gran abanico de opciones que ofrece, ha demostrado ser muy sencilla de usar, con una paleta de herramientas intuitiva. Muchos aspectos del proyecto se han podido hacer sin necesidad de recurrir a ayuda de las distintas fuentes de soporte que ofrece la herramienta. | Unreal en este sentido, tiene una interfaz menos cuidada en este aspecto. Nos hemos encontrado en muchas situaciones en la necesidad de recurrir a ayuda para poder crear el modelo | Unity ofrece una interfaz mucho más intuitiva que Unreal |
| velocidad de respuesta | Unity ofrece una buena velocidad de respuesta. | En este apartado, Unreal nos ha sorprendido muy gratamente. Pese a conllevar mayor dificultad el hecho de configurar parámetros a un nivel más bajo, permite una mayor adaptación a la arquitectura sobre la que trabaja, ofreciendo mayor tiempo de respuesta | Unreal es una herramienta más optimizada y ofrece un tiempo de respuesta más rápida que Unity |
| tiempo de configuracion | Unity no ha precisado de una gran cantidad de tiempo en ser configurada para poder comenzar a operar | La configuración de parámetros en un nivel de abstracción menor, ha conllevado en la necesidad de más tiempo para configurar la herramienta. | Unreal precisa de más tiempo para poder configurar los parametros |
| CALIDAD DE VISIBILIDAD | Unity ofrece grandes resultados en cuanto a nivel de realismo. | Unreal ofrece unos resultados superiores a Unity, con unas texturas más logradas. | Pese a conllevar un nivel de dificultad mucho mayor que Unity, Unreal ofrece resultados muy contundentes, si bien Unity también ofrece unos resultados excelentes. |

# 8. Conclusiones

Podemos concluir el proyecto con una idea de que existe facilidad para poder crear entornos en realidad virtual en ambos programas, pero debido a las diferentes gafas de realidad virtual, no todas funcionan de la misma manera, salvo las más utilizadas que son Oculus Rift y HTC VIVE.

En nuestro caso, utilizando las gafas de realidad mixta de Lenovo Explorer podemos apreciar como hay que ingeniárselas un poco para poder implementar los entornos VR en ambos programas.

A día de hoy se requiere de un hardware muy específico para poder realizar este tipo de proyectos, ya que necesitan de bastante potencia para que la sensación creada en VR sea buena.