

UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

ESCUELA POLITÉCNICA SUPERIOR DE ELCHE

GRADO EN INGENIERÍA INFORMÁTICA EN  
TECNOLOGÍAS DE LA INFORMACIÓN



Desarrollo de un Entorno Docente de Realidad Virtual usando  
Tecnologías Web



TRABAJO FIN DE GRADO

Julio 2023

AUTOR: Antonio García Ruiz

DIRECTOR: Manuel Quesada Martínez

## RESUMEN

La Realidad Virtual (VR) es una tecnología en continuo cambio que ha evolucionado mucho en los últimos años. La RV se utiliza en muchos campos, desde el sector del ocio a través de los Videojuegos hasta el sector de la Educación. Es en este último campo en el que se va a enfocar este proyecto por medio de la implementación de un prototipo de Metaverso Docente.

La motivación principal de este proyecto es crear un entorno docente de RV compartido por alumnos y profesores para distintas asignaturas. Los usuarios podrán acceder a un Mundo Virtual en el que podrán interactuar con distintos tipos de recursos educativos. Para ello, se ha diseñado una aplicación web que ofrece a los usuarios el acceso a tantos Metaversos como asignaturas tenga asociadas. En cada Metaverso el usuario podrá encontrar tres zonas con diferente funcionalidad. La primera zona es un Aula que servirá de punto de encuentro para visualizar vídeos relacionados con los contenidos de la asignatura. La segunda zona será un Laboratorio en el que se podrán visualizar modelos 3D junto a un formulario interactivo con preguntas sobre ellos para afianzar el aprendizaje; por ejemplo, podrá aparecer un modelo anatómico asociado a un formulario con preguntas sobre él. Por último, se ofrecerá a los usuarios de una Sala de Reuniones en la que podrán conversar usando los micrófonos de sus dispositivos, teniendo en cuenta la proximidad en el mundo entre sus avatares.

Para lograr el objetivo anterior se ha tenido que investigar distintas Tecnologías Web y *frameworks* que permitan cumplir con los requisitos del proyecto. Tras el análisis, hemos desarrollado nuestra solución usando las siguientes tecnologías. Para la parte del servidor se utilizará la tecnología Node.js. Las vistas se implementarán como aplicaciones webs usando HTML5, CSS y JS, y para la creación del Metaverso se usará el *framework* A-Frame que facilita la integración e interacción en tiempo real de los modelos 3D a través del DOM. Para la edición y manipulación de los objetivos 3D que componen nuestro entorno hemos usado Blender. Se ha utilizado una metodología ágil de desarrollo de software basada en Scrum, así como herramientas de desarrollo como Visual Studio Code o Bitbucket para el control de versiones.

Como conclusión, este proyecto ha sido un gran reto que ha requerido abordar distintas tecnologías actuales e innovadoras. El contexto de librerías y herramientas definido en el estado del arte nos ha permitido construir un prototipo funcional que cumple el objetivo del trabajo. Además, al final de este se describen, junto a las conclusiones, una serie de líneas de trabajo futuro que permitirían continuar con el trabajo aquí iniciado.

**Palabras clave:** realidad virtual, RV, educación, inmersión, metaverso, A-Frame, NAF, Cardboard, WebXR, WebVR

## AGRADECIMIENTOS

Me gustaría dar gracias a mi madre por apoyarme en todo momento de la vida y en especial en la carrera, por tener una gran paciencia y motivarme.

También a mis compañeros de clase Alfonso, David, Enzo y Alae por estar desde el principio de carrera apoyándonos mutuamente. Que no falten las risas junto a los buenos momentos que no se olvidarán.

También me gustaría agradecer a los profesores de la carrera. En especial a mi tutor Manuel Quesada Martínez, por la confianza que ha depositado en mí, tanto en las asignaturas como este trabajo donde le agradezco toda la ayuda proporcionada.

Por último, agradecer a mis amigos de toda la vida Wilson, Ricky y Gianni, que siempre han estado apoyándome y ayudándome desde el instituto.



# Contenido

1.- Introducción.....	6
1.1- Realidad Virtual y la Educación .....	7
1.2- El Metaverso.....	7
1.2.1- El Metaverso y la industria audiovisual .....	8
1.2.2- Los Metaversos y la Educación.....	10
1.2.3- Posibilidades para el uso y la creación de un Metaverso.....	11
1.3.- Objetivos.....	14
2.- Materiales y Métodos .....	15
2.1- Estado del arte .....	15
2.1.1- Breve reseña histórica de la Realidad Virtual .....	15
2.1.2- Aplicaciones que hacen uso de la RV en Educación .....	16
2.1.3- Plataformas educativas de RV .....	19
2.2- Estado del arte de la Realidad Virtual en la Web .....	22
2.2.1- Estándares para la RV en la web: WebVR y WebXR .....	22
2.2.2- Problemas de compatibilidad de Google CardBoard .....	23
2.2.3- Desarrollo de RV en la web: Three.js, Babylon.js y A-Frame .....	26
2.2.4- A-Frame y sus herramientas.....	29
2.2.5- Resumen y comparativa de <i>frameworks</i> analizados.....	33
2.2.6- Repositorios y modelos 3D para construir el Metaverso .....	33
2.3- Herramientas utilizadas para el entorno de configuración.....	35
2.4- Otras herramientas utilizadas .....	36
2.5- Propuesta de la solución .....	36
3.- Metodología y Resultados.....	38
3.1- Metodología.....	38
3.2- Casos de Uso .....	39
3.3- Diagramas de secuencia y mapa de navegación .....	41
3.4- Diseño de la Base de Datos .....	44
3.5- El uso de modelos 3D en el Metaverso .....	45
3.6- Algunos aspectos destacables de implementación .....	46
3.6.1- Instalación del entorno: Node.js y XAMPP.....	47
3.6.2- Integración de A-Frame en los ficheros HTML5.....	48
3.6.3- Importación de modelos 3D externos en A-Frame.....	48
3.6.4- Carga dinámica de los distintos Metaversos.....	49

3.6.5- Audio de proximidad e interacción entre avatares .....	51
3.6.6- Interacción del usuario con los elementos del Metaverso.....	52
3.7- Ejemplo de uso de la Aplicación.....	55
3.7.1- Despliegue del proyecto.....	55
3.7.2- Registro de usuario e inicio de sesión.....	56
3.7.3- Acceso al Metaverso.....	56
3.7.4- Interacción entre usuarios conectados simultáneamente .....	58
3.7.5- Explorando las distintas salas: vitanto el Aula .....	59
3.7.6- Visitando el Laboratorio .....	60
3.7.7- Visitando la Sala de Reuniones .....	61
3.7.8- Gestión recursos: vídeos .....	62
3.7.9- Gestión recursos: modelos del laboratorio y cuestionario.....	63
4-Conclusiones y trabajo futuro.....	65
4.1 Trabajo futuro .....	66
5. Bibliografía .....	67
ANEXO I: Casos de Usos .....	72



# 1.- Introducción

La educación es un proceso de transferencia cultural del que hacemos uso para transmitir una cantidad de valores y conocimientos a las personas, con el fin de facilitar y promover el enriquecimiento personal además de ayudar a interactuar con el mundo exterior [1].

La palabra educación (enseñanza, acción y efecto de desarrollar las facultades intelectuales) viene de latín *education* (crianza, entrenamiento) derivado del verbo *educare*. Se puede ver que la palabra *education* está formada por el prefijo *ex*, el verbo *ducere* y el sufijo *-tio*, ambos términos provenientes del latín. El prefijo *ex*, da la idea de hacia afuera, *ducere* significa guiar, arrastrar y el sufijo *-tio* que es equivalente a *-cion* que indica acción y efecto. Entonces si juntas las tres cosas formas *educativo* [2].

Los beneficios de la educación difieren según las etapas en las que tiene lugar. Por ejemplo, en Preescolar y las primeras etapas de primaria lo que más destaca es la alfabetización, ya que es donde aprendes a escribir y a leer. En la segunda etapa de primaria hay que destacar la interacción social junto a las actividades formativas, que es donde se comienza a hacer contacto con compañeros y a socializar a través de excursiones y viajes. Secundaria es la etapa de amplificación del aprendizaje y la interacción social ya que estos aspectos aumentan drásticamente comparado con la educación primaria. Por último, en la etapa de la Educación Superior el beneficio es incrementar las oportunidades de trabajo junto al desarrollo de habilidades para resolver problemas, ya que gracias a eso en un futuro se alcanza el ser independiente y acceder a un trabajo [1].

En el contexto de este proyecto, merece la pena analizar el beneficio de la Realidad Virtual (RV) en la educación (ver Figura 1). Incluir una tecnología innovadora como esta permite conservar todos los beneficios anteriores más nuevas ventajas como el acceso remoto a una variedad ilimitada de servicios (materiales, temas, tutores...). Además, el uso de la Realidad Virtual facilita la socialización ya que permite hablar con gente de todo el mundo.



Figura 1 Metaverso en Educación (imagen extraída de [3])

## 1.1- Realidad Virtual y la Educación

El uso de la Realidad Virtual permite recrear un ambiente generado con las tecnologías de la Informática, donde los dispositivos RV admiten de manera implícita la realidad existente. Todo ello permite crear experiencias educativas enmarcadas en una Realidad Mixta que unifica la destreza humana y la inteligencia artificial de la que se hace uso para generar esta RV en la educación [4]. El uso de simulación en educación permite trabajar estos tres puntos:

- Posibilita que los educandos se concentren en un determinado objetivo de enseñanza.
- Permite la reproducción de un determinado procedimiento o técnica.
- Posibilita que todos apliquen un criterio normalizado.

A través de las simulaciones con RV, podremos sumergir al usuario en una interacción en 3D como podría ser en medicina, construcción, videojuegos, educación... (ver Figura 2).



Figura 2 Metaverso en la educación (imagen extraída de [4])

## 1.2- El Metaverso

La primera pregunta que hay que hacerse cuando escuchamos la palabra **Metaverso**, es ¿qué significa? Un Metaverso [5] es un espacio virtual colectivo y persistente, que puede ser visitado y explorado por muchos usuarios. El mundo es generado por un ordenador y se puede visitar a través de la Realidad Virtual o un navegador web a través de dispositivos móviles (ver Figura 3).



Figura 3 Metaverso (imagen extraída de [6])

El término Metaverso fue acuñado por primera vez en 1992 por el escritor de ciencia ficción Neal Stephenson en su novela titulada Snow Crash [7]. Aunque el concepto de los Metaversos ya lleva años entre nosotros, siendo definido como un lugar en que se mezclan el mundo real y el virtual para generar un espacio en común único, su relación con Internet es un concepto relativamente nuevo. Este término fue muy conocido mundialmente tras la conferencia Connect 2021 en la que Mark Zuckerberg anunció un cambio de nombre de Facebook a Meta para enfocarse en un concepto más amplio de tecnologías, entre ellas la RV y el Metaverso (para más información se recomienda al lector la visualización de las *Keynotes* de 2021 [8] y 2022 [9]).

En la actualidad, el Metaverso supone el futuro de las experiencias virtuales interconectadas a través de tecnologías de Realidad Virtual y Realidad Aumentada [5]. Algunas características relacionadas con el Metaverso son:

- Los objetos y el mundo del Metaverso, aunque la gente se desconecte, seguirán existiendo, la persistencia es algo que influye bastante en este sentido.
- Los usuarios pueden interactuar con el mundo virtual y con otros usuarios a través de los movimientos del cuerpo en caso del dispositivo de realidad virtual o de voz si tienes un micrófono, eso hace que sea más divertido el poder entrar.

### 1.2.1- El Metaverso y la industria audiovisual

Hay muchas series y películas que han tratado sobre el Metaverso. Quizás el ejemplo más representativo de esta tecnología es la película **Ready Player One** [10] (ver Figura 4). La temática principal de esta película se centra en el año 2045, donde una persona usa un traje háptico y lentes de realidad para trasladar los movimientos del mundo real al mundo virtual y permitirle sentir lo que ocurre en el mundo digital.





Figura 4 Escena de Ready Player One (imagen extraída de [10])

Otro ejemplo sería **Black Mirror** [11], una serie que muestra los peores rasgos de la humanidad y sus mayores innovaciones. Hay varios capítulos, que tratan como la RV o el Metaverso pasan de ser algo divertido a ser fatal o de terror. Concretamente, los capítulos más impactantes son:

- **Playtesting** [12] (ver Figura 5): es un capítulo en la que una persona entra a un videojuego de terror que está diseñado para nutrirse de los miedos personales del jugador e incorporar elementos audiovisuales en el entorno a base de estos. Por ejemplo, si padeces de aracnofobia, las arañas aparecerán en algún momento del juego, lo que hace que no puedas diferenciar si está en la realidad o en un mundo artificial ya que te encuentras inmerso al cien por cien.



Figura 5 Playtesting (imagen extraída de [13])

- **Joan es Horrible [14] (ver Figura 6):** es otro capítulo donde una persona firma los términos y condiciones de una aplicación permitiendo que la empresa pueda usar todos sus datos. Gracias a un ordenador cuántico, hacen un Metaverso con su vida. A su vez, esa persona tiene otro mundo lo que sucesivamente hace que haya muchos mundos creados artificialmente a partir de una persona real.



Figura 6 Joan es Horrible (imagen extraída de [15] )

### 1.2.2- Los Metaversos y la Educación

Como hemos comentado, hay muchos ejemplos del uso del Metaverso. Por ejemplo, es utilizado para simular distintos ámbitos como el del entretenimiento con los videojuegos, las películas en 3D, o experimentos en ciencia y tecnología. Uno de sus beneficios es que si en lugar de generar experiencias basadas en simulaciones se utilizasen entornos reales estos podrían ser muy caros.

En el contexto de este TFG, el Metaverso es analizado como un producto que comienza a jugar un papel importante en el ámbito educativo, permitiendo a los estudiantes poder observar cosas que en el mundo real no podrían ver, o interactuar con otros estudiantes de distintas partes del mundo. Por ejemplo, en muchas empresas alrededor del mundo, están comenzando a desarrollar tecnología inmersiva 3D orientada a fines educativos [16]–[18]

Para comenzar a aprovechar el Metaverso en un ámbito académico, se requiere familiarizarse con las tecnologías que impulsan los entornos virtuales y que permiten proporcionar a los estudiantes una experiencia más próxima al mundo real. Algunas tecnologías extraídas de [19] son:

- **Realidad Aumentada y Realidad Virtual:** son tecnologías que tienen cosas en común pero no son lo mismo, ya que una el usuario interactúa con algo generado por el ordenador, y la otra es integrar elementos digitales a la realidad.
- **Lentes de realidad virtual y gafas inteligentes:** para que la experiencia sea mucho más inmersiva, se usan dos cosas que son las lentes RV, que

conecta al usuario a entornos digitales y luego las gafas inteligentes, que, aunque se encuentran en primeras etapas de desarrollo, permite combinar elementos virtuales con el entorno físico del usuario.

- **Inteligencia Artificial:** hará análisis en tiempo real y según la necesidad del usuario, personalizará el entorno inmersivo.
- **Blockchain y contratos inteligentes:** al necesitarse sistemas de control y de seguridad, gracias a la tecnología *Blockchain* permitirá que los entornos virtuales ofrezcan gran confianza a los usuarios como a empresas

Tras ver algunas tecnologías, se indica ahora algunos ejemplos en los cuales se podría usar el Metaverso con fines educativos:

- **Campus virtuales en el Metaverso:** una meta para el futuro es que un alumno pueda asistir a su campus virtualmente, que sea idéntico todo y que se pueda convivir simultáneamente con sus compañeros.
- **Simuladores con alta fidelidad contextual:** actualmente ya existen muchos simuladores para estudiantes de medicina, arquitectura, etc. Pero el Metaverso va a llevarlos a un siguiente nivel ya que podrá hacer cosas tan reales como practicar operaciones de corazones, o un vuelo con un nuevo avión.
- **Visitar virtuales alrededor de la Tierra u otros planetas:** gracias al metaverso y a la tecnología, prometen una inmersión completa que podremos escuchar y practicar el idioma de los entornos junto a ver museos, ciudades, etc.
- **Aprendizaje *on the go*:** gracias a la tecnología de las gafas inteligentes, vas por una calle y simplemente con escanear un código QR ya te saldría contenido digital para poder ayudar al estudiante a aprender información sobre ese sitio.

### 1.2.3- Posibilidades para el uso y la creación de un Metaverso

En la actualidad existen soluciones que permiten hacer uso de Metaversos de una forma sencilla. Por ejemplo, un ejemplo muy conocido es Decentraland [20] que es una plataforma de Realidad Virtual donde tu imaginación es el único límite. Por ejemplo, puedes crear y vender coleccionables junto a crear mundos con experiencias interactivas (ver Figura 7). Se recomienda al lector la lectura del siguiente artículo [21] para saber más sobre qué es y cómo entrar en este Metaverso por primera vez.



Figura 7 Primer inicio de sesión en Decentraland (imagen extraída de [21])

Por otro lado, tenemos soluciones que permiten la creación de Metaversos a medida y como servicios. Por ejemplo, la empresa Virtway Events [22] permite crear salas virtuales y personalizarlas que pueden ser usadas con distintos propósitos. Por ejemplo, ofrecen plantillas personalizadas para la realización de talleres online, reuniones de equipos de trabajo, talleres y exposiciones... (ver Figura 8).



Figura 8 Captura de la página principal de Virtway Events (imagen sacada de [22])

Por poner otro ejemplo, se recomienda la lectura del artículo [23] en el que se describe el caso de uso llevado a cabo por diferentes instituciones académicas como la recreación del Paraninfo de la Universidad Pablo de Olavide (ver Figura 9) o la presentación de un acto en el metaverso de la Facultad de Filología (ver Figura 10).



Figura 9 El metaverso que recrea el Paraninfo de la Universidad Pablo de Olavide (imagen extraída de [23])

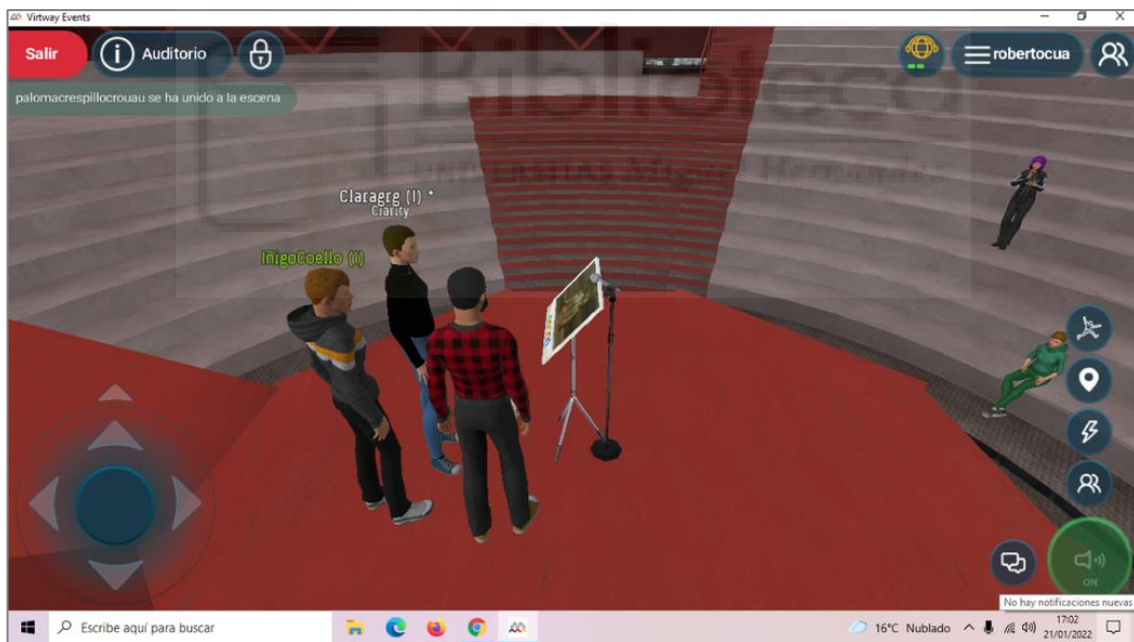


Figura 10 La presentación de un acto en el metaverso de la Facultad de Filología (imagen extraída de [23])

En este TFG queremos tomar estos trabajos como referencia para, en la medida de lo posible crear una solución similar pero enfocada a crear un entorno que facilite el uso de Metaversos en el ámbito docente. Por ejemplo, queremos que sirva a alumnos y profesores de una misma asignatura para tener un punto de encuentro virtual común, y desde el que poder acceder a distintos materiales docentes relacionados con ella.

Queremos que sirva como prueba de concepto para evolucionar, por ejemplo, las reuniones Meet que empleamos habitualmente para impartir la docencia online. Esta evolución consistiría en un Metaverso UMH compuesto por diferentes salas accesibles por grupos de alumnos y profesores que compartan materias.

Además, queremos que nuestra solución pueda ser desplegada por el mayor número de usuarios; por ello queremos que el desarrollo se realice para entornos que utilicen tecnologías web y sean accesibles a través de navegadores web y desde distintos tipos de dispositivos.

### 1.3.- Objetivos

En línea de la discusión anterior, el **objetivo general** de este trabajo es crear una Aplicación Web que integre tecnología de RV para generar un entorno interactivo que pueda ser usado como Metaverso Docente.

Como se ha comentado, este Metaverso servirá de punto de encuentro para profesores y alumnos ofreciendo la posibilidad de compartir distintos tipos de recursos relacionados con una sesión docente. Por ejemplo, se crearán entornos personalizados según diferentes materias gestionadas por el docente, en las que podrá subir vídeos, exámenes o modelos 3D con los que interactuar.

Para conseguir el objetivo principal, es necesario cumplir los siguientes **objetivos secundarios**:

- Realizar un estudio del estado del arte de tecnologías web que permitan integrar la RV de forma sencilla para el usuario, así como integrarse con los distintos tipos de recursos mencionados.
- Realizar un análisis y diseño de software, e implementar un prototipo funcional de la aplicación web de RV poniendo en práctica una metodología de desarrollo ágil.
- Desplegar el prototipo implementado en un servidor donde los usuarios se puedan conectar y probar el sistema.

A **nivel personal** he elegido este proyecto porque el mundo de la Realidad Virtual me parece algo bastante complejo y que da mucho que hablar. Por ejemplo, puedes hacer mil cosas, desde volar un avión hasta visitar otro país como Japón o Estados Unidos que están a kilómetros de aquí. Es como si pudieras cumplir tus sueños sin necesidad de moverte de casa, ya que la simulación usando RV te facilita mucho las cosas.

## 2.- Materiales y Métodos

En este capítulo analizaremos un estudio del estado del arte, desde el análisis de la evolución de las tecnologías de RV y sus dispositivos, hasta el estudio de tecnologías actuales que nos permitan implementar un Metaverso Docente de RV desplegado como en un entorno web.

### 2.1- Estado del arte

Comenzaremos esta sección haciendo un recorrido por la historia de la RV y la repercusión de los Metaversos en la actualidad.

#### 2.1.1- Breve reseña histórica de la Realidad Virtual

Aunque el término de Realidad Virtual empezase a ganar relevancia a principios de los años 90, este concepto es mucho más antiguo siendo una tecnología con una mayor historia. El primer estereoscopio fue inventado en 1838 por Charles Wheatstone y consistía en una especie de gafas que simulaban una imagen en 3D (ver Figura 11). La primera cámara llegaría 10 años más tarde en 1850, y permitía capturar dos imágenes que, observadas a través de un visor, generaban una imagen 3D en el interior del cerebro con la sensación de profundidad [24].



Figura 11 Estereoscopio

En el año 1963, Hugo Gernsback sorprendió con su *The Teleyeglasses*, que fue una televisión portable sujeta a la cabeza como unas gafas (ver Figura 12). El dispositivo permitía ver la televisión de forma, dual y estereoscópica. No podía responder a los movimientos de la cabeza, pero hacía sentir al usuario como si estuviese casi dentro del mundo de la televisión.

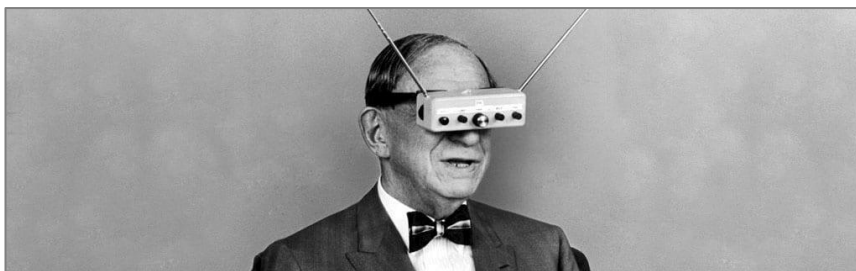


Figura 12 The Teleyeglasses

En el año 1995, Nintendo lanzó Virtual Boy (ver Figura 13) que fue un proyector estilo gafas de Realidad Virtual para crear un efecto monocromático, aunque fuese un fracaso comercial [25].



Figura 13 Virtual Boy

En 2012 se lanzaban las Oculus [26] (ver Figura 14) que permitían la visión de volumetría 3D en un entorno 360° de una forma envolvente. A partir de este momento se disparó la fiebre de la RV y aparecieron muchas gafas como HTC Vive Focus, Valve Index, Playstation VR....



Figura 14 Oculus

### 2.1.2- Aplicaciones que hacen uso de la RV en Educación

Continuando con el análisis iniciado en la Introducción, destacamos en este apartado un ejemplo representativo en el ámbito de los videojuegos que es **VRChat** [27]. VRChat es un videojuego creado en 2017 que está disponible en la plataforma Steam. Propone una dinámica de juego donde poder elegir, copiar o dibujar un avatar para construir un personaje a tu medida. También se puede crear cualquier mundo virtual donde poder moverse o atraer a miles de usuarios reconocibles por sus avatares, que interactúan con el juego de forma casi ilimitada.



Durante la experiencia, VRChat es compatible con guantes y gafas de RV, para así poder comunicarse con gestos con el resto de los usuarios, sincronizar los labios a los del avatar, bailar, agacharse u otros miles de interacciones. En la Figura 15 se muestra un ejemplo donde aparece un grupo de usuarios con distintos avatares jugando a un juego de mesa gracias a la RV.



Figura 15 VRChat

Continuamos nuestro análisis de aplicaciones con ejemplos representativos para explicar la situación actual del uso de RV en aplicaciones relacionadas con el ámbito educativo:

- **Anatomyou VR [28]:** es una aplicación que te permite navegar de manera inmersiva por el interior del cuerpo humano para aprender e identificar los diferentes órganos que lo constituyen. Es posible usarlo en desde dispositivo móviles, así como desde dispositivos de RV (ver Figura 16).

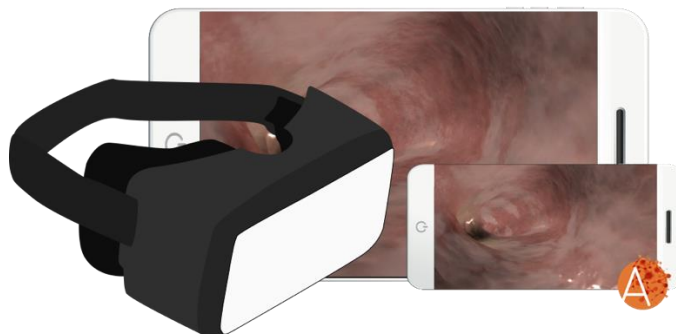


Figura 16 Anatomyou

- **Mondly VR** [29]: podremos aprender una multitud de idiomas a través de la práctica y usando RV. Recrea situaciones cotidianas que se encuentran en el día a día o en un viaje, y donde se requiere de un cierto dominio del idioma. Podemos escoger entre uno de los 30 idiomas disponibles (ver Figura 17).

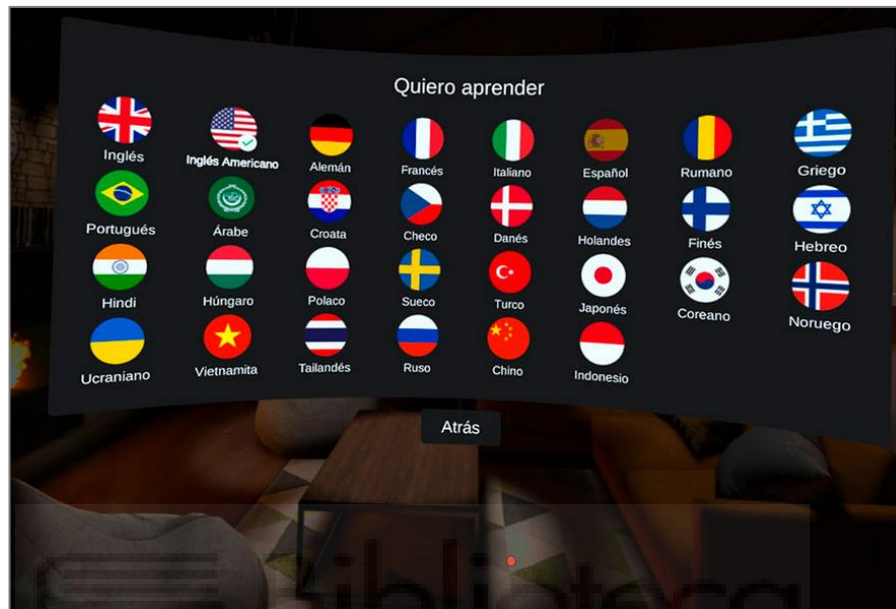


Figura 17 Mondly

- **Unimersiv** [30]: es una plataforma para vivir experiencias educativas en RV (ver Figura 18). Está destinada a estudiantes de todas las edades, proponen el uso de la RV como herramienta para acelerar los aprendizajes. En esta plataforma, se publica regularmente contenido educativo, dando a los usuarios la oportunidad de aprender sobre temas tan diversos como el espacio, la historia o la ciencia.



Figura 18 Unimersiv

### 2.1.3- Plataformas educativas de RV

Por último, vamos a dirigir nuestra búsqueda a plataformas educativas de RV, e incluso a algunas que ofrezcan soluciones o Metaversos personalizables, y que puedan ser utilizados como espacios para la enseñanza y el aprendizaje.

En [31] se exploran las posibilidades formativas de nuevas plataformas audiovisuales 3D como Second Life [32], así como algunas de sus ventajas y desventajas de su uso como herramienta educativa. En dicho trabajo concluyen que el estudio de las dinámicas complejas entre avatares y entre el usuario y el propio Metaverso abren nuevas vías de investigación para el estudio de la educación en el siglo XXI. Estas vías deben ser exploradas y examinadas por la comunidad educativa y académica por su importancia actual y potencial futuro.

Aunque algunas de las soluciones como las comentadas en la introducción han llevado a la práctica acciones concretas, no encontramos ninguna enseñanza que se imparta 100% online usando este tipo de herramientas. La popularidad alcanzada en 2021 de este tipo de entornos ha generado mucha atracción por parte de inversores y la aparición de una gran cantidad de noticias y plataformas dedicadas.

En este trabajo vamos a analizar en profundidad dos de ellas 360 School [33] y Educa 360 [34]. Hemos seleccionado estos dos ejemplos por presentar características que nos han resultado útiles para definir nuestra herramienta:

- **360 School [33] (ver Figura 19):** es un servicio único de educación inmersiva en 360° y RV diseñado para los profesores y sus alumnos. Proporciona acceso a una colección de más de 350.000 imágenes 360° de alta resolución de lugares históricos, culturales y simplemente bellos y relevantes desde el punto de vista educativo. Ofrece herramientas para buscar panorámicas 360° utilizando un mapamundi interactivo y compartirlas con los alumnos por diferentes canales. También permite crear visitas guiadas personalizables. Las visitas guiadas son viajes interactivos y envolventes con panorámicas de 360° muy fáciles de usar en ordenadores, smartphones, tabletas y gafas de Realidad Virtual.

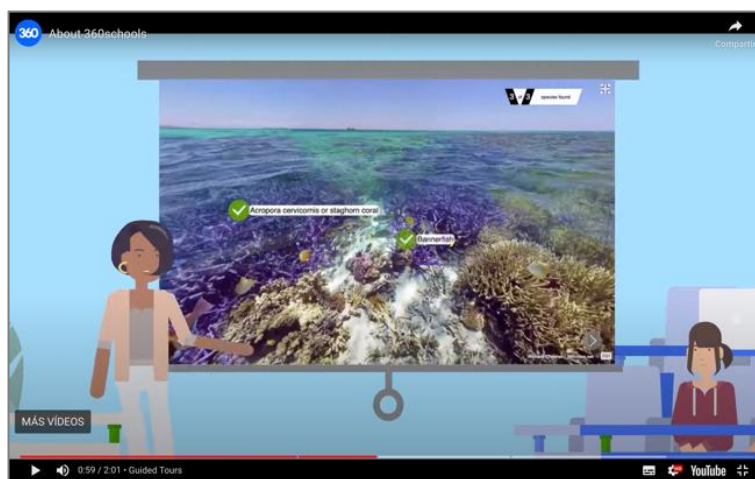


Figura 19 Video de donde se ve el 360° sobre un lugar (imagen extraída de [35])

Por ejemplo, en la Figura 19 se muestra un ejemplo de un viaje personalizado en el que se ha etiquetado una imagen 360° que puede visualizarse en un entorno de RV y a continuación se pueden definir cuestionarios con preguntas sobre dicho recurso (ver Figura 20).

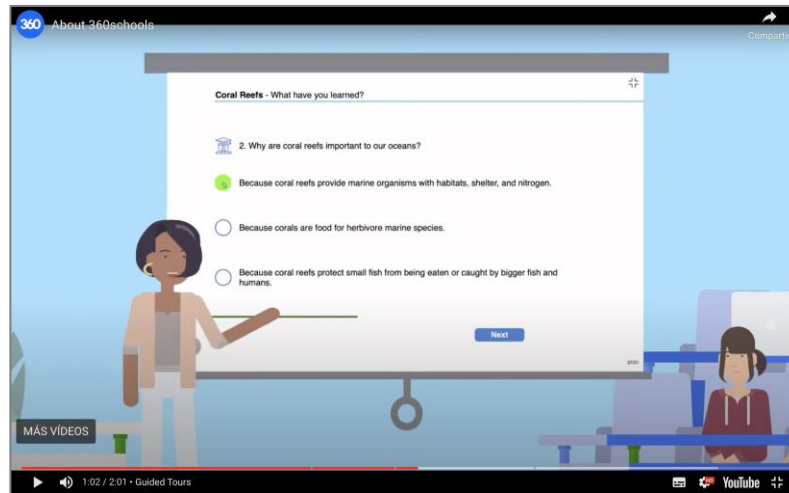


Figura 20 Preguntas relacionadas con el video (imagen extraída de [35] )

- **Educa 360 [34] (ver Figura 21):** permite crear aulas virtuales completas y ofrece diferentes tipos de recursos 3D para personalizarlas. Es una solución mucho más completa que la anterior ya que, en está sí que permite crear Metaversos personalizables y no solo experiencias de RV basadas en imágenes 360°.

Esta solución requiere el pago de licencias con planes que oscilan entre los 19,90 y los 119,90 euros por usuario al mes. También ofrecen tres kits de implantación que incluyen el hardware adecuado para poder llevar a un aula física sesiones virtuales como las que ellos proponen. En ese sentido, vemos que no es una iniciativa de bajo coste, ya que un kit de implantación oscila entre los 4.400 y los 14.200 euros (ver Figura 22).



Figura 21 Inicio de la página Educa360 (imagen extraída de [34])




 <p><b>Kit de Implantación X 4</b></p> <p><b>4.400€</b></p> <p>Antes 4.900€</p> <ul style="list-style-type: none"> <li>• Maleta 4 Dispositivos Educa360</li> <li>• Gafas VR + Touch (4 unidades).</li> <li>• Sistema Baterías Carga (para Touch).</li> <li>• Router Internet 4G (Conexión Vodafone Independiente).</li> <li>• 6+6 meses de Licencia Educa360.</li> <li>• Implantación y configuración.</li> <li>• Aula Virtual personalizada para el Centro en el Metaverso.</li> <li>• Creador de Avatares.</li> <li>• 4 Cuentas - Educa360.</li> <li>• 4 Cuentas - Educamaker.</li> </ul>	 <p><b>Kit de Implantación X 8</b></p> <p><b>7.400€</b></p> <p>Antes 8.900€</p> <ul style="list-style-type: none"> <li>• Maleta 8 Dispositivos Educa360</li> <li>• Gafas VR + Touch (8 unidades).</li> <li>• Sistema Baterías Carga (para Touch).</li> <li>• Router Internet 4G (Conexión Vodafone Independiente).</li> <li>• 6+6 meses de Licencia Educa360</li> <li>• Implantación y configuración.</li> <li>• Aula Virtual personalizada para el Centro en el Metaverso.</li> <li>• Creador de Avatares.</li> <li>• 8 Cuentas - Educa360.</li> <li>• 8 Cuentas - Educamaker.</li> </ul>	 <p><b>Proyecto Implantación</b></p> <p><b>14.200€</b></p> <p>Antes 16.900€</p> <p>Convierte tu Institución Educativa en un centro que esté preparado para ofrecer formación con Realidad Virtual.</p> <ul style="list-style-type: none"> <li>• Plan formativo integral (5 Talleres)</li> <li>• 2 KITS de IMPLANTACIÓN X 4</li> <li>• Orientado a profesores y alumnos.</li> <li>• Certificación y expedición de diplomas acreditativos.</li> <li>• 6+6 meses de Licencia Educa360</li> <li>• Cámara 360° modelo RICOH Theta.</li> <li>• 24 Webinars de formación online.</li> <li>• 15 Cuentas - Educa360.</li> <li>• 15 Cuentas - Educamaker.</li> </ul>
--	---	---

Figura 22 Kits con gafas VR y cuentas para poder usar educa360 (imagen extraída de [34])

Las dos herramientas anteriores han sido consideradas como referencia por diferentes factores. Por un lado, la primera es una herramienta personalizable sin coste que permite crear experiencias de RV muy completas. Por otro lado, la segunda es una herramienta muy completa con un coste elevado de la que nos gusta la idea de poder crear aulas o Metaversos virtuales también personalizables de forma sencilla.

Por tanto, queremos crear una herramienta que combinen aspectos de estas dos, ofreciendo además la posibilidad de un uso democratizado que no repercuta en costes elevados para las organizaciones y usuarios que la utilicen. Esto formará parte de nuestros requisitos a la hora de analizar y seleccionar las tecnologías de desarrollo que permitan implementar nuestra solución. De hecho, este es uno de los motivos por los que centramos nuestro trabajo en tecnologías de desarrollo web y descartamos soluciones específicas para modelos de gafas concretos, como por ejemplo la implementación de soluciones para dispositivos concretos como Oculus Rift S como los realizados en [36], [37].

Desde el punto de vista de la funcionalidad, el análisis de las distintas herramientas descritas anteriormente ha permitido identificar algunos aspectos básicos que deberemos cubrir en nuestra propuesta:

1. **Sin descargas:** para acceder a nuestra aplicación no será necesario instalar, ni descargar nada. Se deberá poder acceder directamente desde un propio navegador web.
2. **Multidispositivo:** se podrá acceder a los metaversos generados desde diferentes tipos de dispositivos. Por ejemplo, si un usuario no dispone de gafas de RV podrá acceder desde un PC normal, aunque limitando de algún modo la experiencia inmersiva. En caso de usar dispositivos de RV,

la herramienta deberá ser compatible con la mayoría de ellos, sin ser necesario implementar versiones específicas de cada fabricante.

3. **Interacción entre usuarios:** chatear, moverse, hablar, compartir, descubrir un mundo lleno de posibilidades deberá ser una funcionalidad común para todos los usuarios, independientemente del dispositivo que utilicen para conectarse.
4. **Trabajo colaborativo y restricción de acceso:** será posible crea aulas y clases en las que el acceso estará restringido a los alumnos matriculados y sus profesores. Se ofrecerán distintos tipos de entornos y recursos personalizables que se podrán integrar en el mundo 3D generado. Cada uno de esos mundos será compartido por todos los usuarios con acceso a ellos.

Todos estos principios deberemos contemplarlos en nuestra solución, por lo que deberemos hacer un análisis de las tecnologías que nos permitan llevarlos a cabo y que discutimos en el siguiente apartado.

## 2.2- Estado del arte de la Realidad Virtual en la Web

Proseguimos nuestro análisis con herramientas específicas que además del 3D nos ofrezcan la posibilidad de montar soluciones de RV accesibles a través del navegador web.

### 2.2.1- Estándares para la RV en la web: WebVR y WebXR

Existen dos APIs que permiten implementar soluciones de Realidad Virtual en el navegador web:

- **WebVR** [38], [39]: es una API experimental JavaScript creada en 2014 que proporciona soporte para dispositivos de RV como el HTC Vive, Oculus Rift o Google Cardboard en un navegador de web de forma práctica y accesible. Cada navegador y fabricante aporta librerías para dar soporte a sus propios dispositivos. En su origen fue diseñada con los siguientes objetivos: (1) detectar los dispositivos de RV disponibles, (2) obtener las características de los dispositivos, (3) obtener la posición y orientación del dispositivo y (4) mostrar imágenes en el dispositivo con la frecuencia de refresco adecuada.
- **WebXR** [40]: surgió en 2018 y es la tecnología actual que ha remplazado a WebVR. WebXR combina la Realidad Aumentada y la Realidad Virtual, permitiendo su uso en entornos web de gran potencia y fácilmente accesibles. Al ser una mezcla, hay dos tipos de WebXR:
  - **WebXR AR:** se trata de una API en los navegadores que proporciona un estándar y permite acceder a los servicios de Realidad Aumentada del Sistema Operativo (ARcore en Android y ARKit en iOS). Combina

el mundo virtual con el mundo real, haciendo que los elementos digitales se integren en el entorno físico a través de la cámara de un dispositivo.

- **WebXR VR:** este tipo es el que usaremos en este proyecto y se basa principalmente en permitir a los usuarios sumergirse completamente en entornos virtuales utilizando auriculares o cascos de RV.

Aunque son todavía tecnologías en desarrollo, el nacimiento de soluciones como WebXR pretende democratizar el acceso a las experiencias inmersivas de RV ya que cualquiera podría usarlos sin necesidad de tener el hardware o software específico para su uso. Por ejemplo, las soluciones basadas en WebVR y WebXR permiten acceder desde juegos hasta simulaciones de visualización arquitectónica. Para ello, no es obligatorio generar soluciones *ad hoc* para el último modelo de casco, ya que son compatibles con la mayoría y se accede a las aplicaciones desarrolladas desde el navegador web.

El uso de este tipo de aplicaciones está creciendo significativamente y ofrece una amplia gama de aplicaciones prácticas y creativas en diversos campos. Desde experiencias de comercio electrónico en Realidad Aumentada que permiten a los usuarios usar las cámaras de sus dispositivos móviles para probar productos antes de comprarlos, hasta hacer visitas de Realidad Virtual en entornos tridimensionales. Todo ello está revolucionando la forma de interactuar con el mundo digital.

### 2.2.2- Problemas de compatibilidad de Google CardBoard

En el año 2014 surgió una iniciativa muy interesante bajo la premisa de transformar *cualquier* teléfono inteligente en una plataforma de RV por menos de 5 dólares: **Google CardBoard** [41]. Esta solución fue muy popular y con soluciones de este tipo se puede acceder a vídeos 360° disponibles, por ejemplo en YouTube, y visualizarlos en un entorno inmersivo con un móvil (ver instrucciones en <https://support.google.com/youtube/answer/6239930?hl=en>) .



Figura 23 Cardboard convierte los teléfonos en visores de contenido 3D (imagen extraída de [42])

En este trabajo priorizamos el desarrollo de una solución que pudiese ser desplegada con esta tecnología, ya que consideramos que poder acceder a nuestro Metaverso docente con cualquier móvil nos permitiría democratizar considerablemente su uso. Sin embargo, nos hemos encontrado el problema de que muchos navegadores de dispositivos móviles han dejado de ser compatibles con WebXR [43]. Es por ello muy importante tenerlo en mente y prestar atención a estos temas de compatibilidad. Por ejemplo, en la Figura 24 se aprecian las versiones de los navegadores de escritorio que son compatibles con la tecnología más actual, WebXR.

Chrome	Edge *	Safari	Firefox	Opera	IE
4-64				10-51	
65-78	12-18	3.1-12.1	2-76	52-65	
<sup>1</sup> 79-109	<sup>1</sup> 79-109	<sup>3</sup> 13-16.2	<sup>2</sup> 77-109	<sup>1</sup> 66-94	6-10
<sup>1</sup> 110	<sup>1</sup> 110	<sup>3</sup> 16.3	<sup>2</sup> 110	<sup>1</sup> 95	11
<sup>1</sup> 111-113		<sup>3</sup> 16.4-TP	<sup>2</sup> 111-112		

Figura 24 Compatibilidades Plataformas VR

Si pasamos por una celda roja que tenga la bandera verde, podemos apreciar que, si se puede habilitar, pero de manera predeterminada están desactivadas (ver Figura 25).



Figura 25 Navegador sin soporte por defecto

Uno de los problemas que hemos encontrado es que para la mayoría de navegadores de dispositivos móviles esta tecnología no es compatible (ver Figuras 26 y 27). Además, aunque en Android funciona correctamente con algunos navegadores, en iOS no es compatible con Safari y obliga a los usuarios a descarga Chrome. De las pruebas realizadas, hemos experimentado que existen ciertos problemas que la hacen poco intuitiva de utilizar para usuarios no expertos; por ejemplo, se puede usar solamente con el protocolo seguro HTTPs



en otro caso surgen problemas de permisos para poder los sensores de los dispositivos.

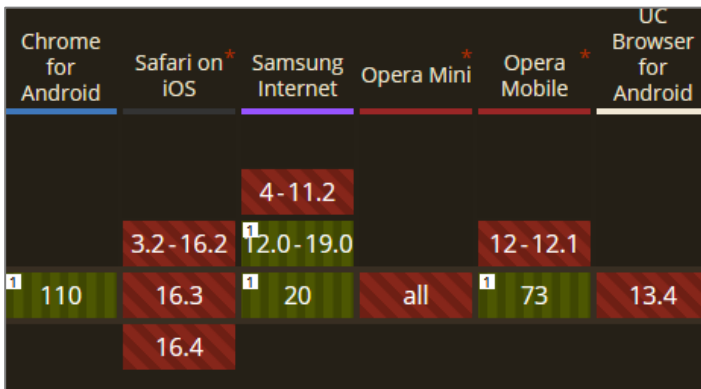


Figura 26 Compatibilidades móvil parte 1

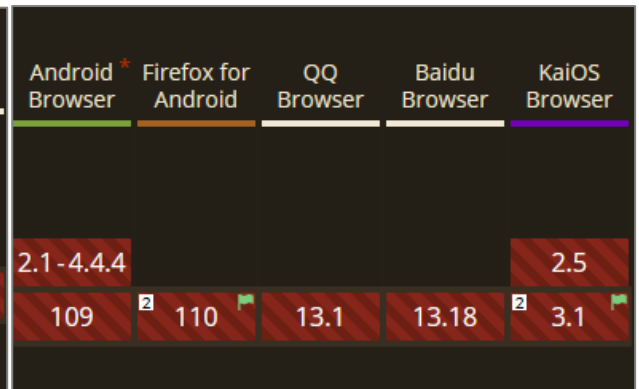


Figura 27 Compatibilidades móvil parte 2

Aunque nuestra idea inicial era hacer una solución compatible con CardBoard, tuvimos que descartarlo por los continuos errores al usar versiones actuales de las APIs y la falta de compatibilidad y apoyo de la industria a esta tecnología.

Durante el proyecto dedicamos un gran esfuerzo para ver si existía la posibilidad de solucionar estos problemas. Probamos y analizamos las siguientes opciones, que fueron descartadas finalmente por suponer demasiados problemas y necesidades de configuración tediosas para los usuarios:

- Problemas con IOS:** este sistema operativo dificulta por defecto el uso de estas tecnologías de VR. Para que sea compatible hay que permitir el *Motion & Orientation Access* en los navegadores. Además, y como se ha comentado anteriormente se requiere el acceso a través de HTTPs ya que es necesario para conceder el permiso de acceso a sensores y otros parámetros y poder usarlo. Una vez hecho esto, se requerirá el uso de Polyfill para adaptar la pantalla como se explica en el siguiente punto.
- Polyfill [44]:** para poder usar RV con Google CardBoard debe usarse esta herramienta para adaptar la visualización al dispositivo. En otro caso habría que personalizar la escena con desplazamiento de vértice para que se duplique (ver Figura 28).



Figura 28 Sin vs Con Polyfill VR

En Android no hemos detectado menos problemas para el uso de CardBoard. Por ejemplo, las pruebas han funcionado bien con el navegador Google Chrome que sí es compatible con WebVR y WebXR. En el caso de usar Firefox o Edge, como vimos en las Figuras 26 y 27 podría fallar.

Todas estas pruebas nos han hecho tomar conciencia de que las tecnologías con las que vamos a trabajar en este TFG se encuentran en desarrollo, y que no hay un interés general en la industria por permitir un uso de estas con cualquier dispositivo. Merece la pena reflexionar que, aunque el uso de CardBoard ha experimentado problemas en los últimos años estas tecnologías WebVR y WebXR sí que son compatibles con la mayoría de los navegadores integrados en dispositivos de RV como Oculus.

### 2.2.3- Desarrollo de RV en la web: Three.js, Babylon.js y A-Frame

Existen *frameworks* de más alto nivel contruidos sobre WebVR y WebXR que han ganado popularidad en los últimos años como Three.js, Babylon.js o A-Frame.

- **Three.js [45] (ver Figura 29):** 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 o WebGL. El código fuente está alojado en un repositorio en GitHub. Three.js simplificó mucho la programación de código en WebGL con una documentación muy detallada pero no para principiantes, ya que va muy al grano y no se extiende para público novel.

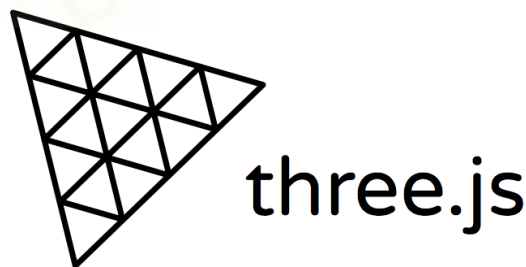


Figura 29 Three.js

En la Figura 30 se muestra un ejemplo extraído de la documentación de three.js donde se explica que es una geometría para esta tecnología. Proporciona varios ejemplos, en este caso con `voxelpainter`, indica como ponerlo en nuestro código para poder implementarlo correctamente. Además, se pueden programar fragmentos de código de forma interactiva y en la nube con herramientas como <https://threejs.org/editor/>.

# Geometry

Geometry is a user-friendly alternative to [BufferGeometry](#). Geometries store attributes (vertex positions, faces, colors, etc.) using objects like [Vector3](#) or [Color](#) that are easier to read and edit, but less efficient than typed arrays.

Prefer [BufferGeometry](#) for large or serious projects.

## Example

[WebGL / geometry / minecraft](#)

[WebGL / geometry / minecraft / ao](#)

[WebGL / geometry / nurbs](#)

[WebGL / geometry / spline / editor](#)

[WebGL / interactive / cubes / gpu](#)

[WebGL / interactive / lines](#)

[WebGL / interactive / raycasting / points](#)

[WebGL / interactive / voxelpainter](#)

```
var geometry = new THREE.Geometry();

geometry.vertices.push(
  new THREE.Vector3( -10, 10, 0 ),
  new THREE.Vector3( -10, -10, 0 ),
  new THREE.Vector3( 10, -10, 0 )
);

geometry.faces.push( new THREE.Face3( 0, 1, 2 ) );

geometry.computeBoundingBox();
```

Figura 30 Documentación Three.js

Para agregar un motor de física tendremos que elegir entre dos Amoo.js y Cannon.js. Hay poca o casi ninguna documentación disponible, por ello, pese a que esta herramienta tiene una activa plantea algunos problemas.

- **Babylon.js** [46] (ver Figura 31): es un moderno motor WebGL JavaScript que abstrae detalles de Three.js. Esta abstracción ayuda a los principiantes y reduce el tiempo para crear nuevas escenas, conlleva que se reduzca hasta aproximadamente un 60% menos.



Figura 31 Babylon.js

Babylon.js usa Canvas al igual que Three.js, solo que el código fuente está escrito en TypeScript y luego se compila en una versión JavaScript pura. Al igual que en three.js, existe una gran documentación y ejemplos interactivos disponibles a través de *playgrounds* (ver Figura 32). Estos ejemplos son de gran utilidad para la comunidad y facilitan su aprendizaje (<https://www.babylonjs-playground.com/>).

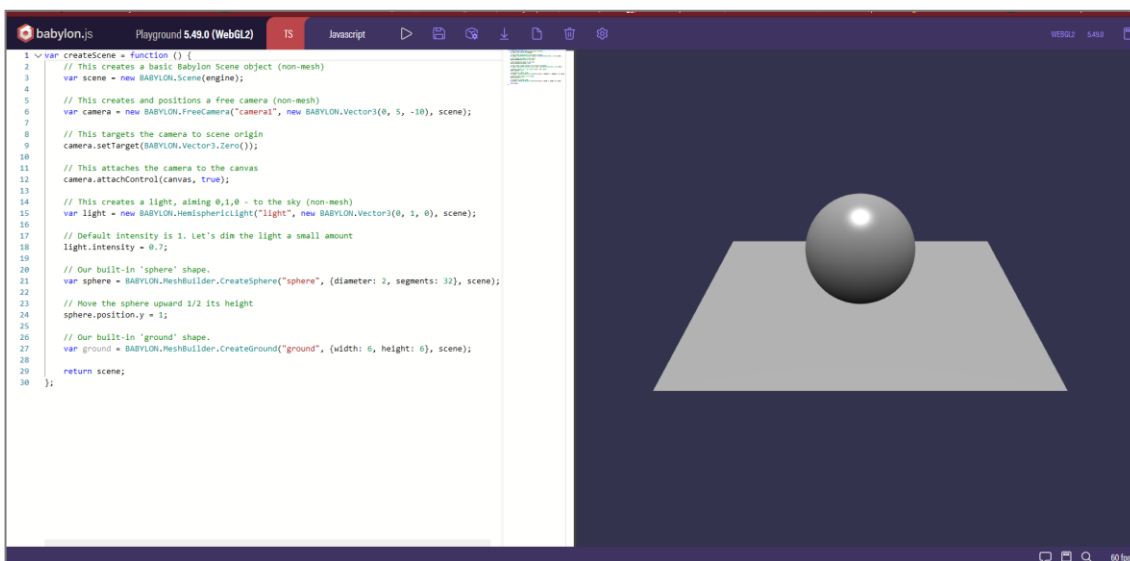


Figura 32 Babylon.js playground

Para simular la física, Babylon permite integraciones de motores de física como *plugins*, un ejemplo sería Babylon JS Physics Imposter (ver Figura 33). Babylon es más amigable y tiene más características interesantes que Three.js.



Figura 33 Babylon.js Physics Imposter

A-Frame es una alternativa a three.js y babylon.js que, además, ha sido la seleccionada en este trabajo. Por ello le dedicamos la siguiente subsección.

## 2.2.4- A-Frame y sus herramientas

A-Frame [47] es una tecnología diseñada para facilitar el uso de WebVR y XR. Hasta su versión 1.0 hace uso de WebVR, en las posteriores ha sido actualizado para trabajar sobre WebXR. A partir de esta versión, este *framework* ha dejado de ser compatible con CardBoard; aunque hemos encontrado numerosos ejemplos que funcionan bien en el navegador web de escritorio, al habilitar su visualización de RV en CardBoard empezaban a producirse errores inesperados. Como hemos comentado, estos errores no ocurren en navegadores actualizados de dispositivos de RV, por lo merece la pena resaltar que este *framework* no está obsoleto sino la compatibilidad de CardBoard con él.

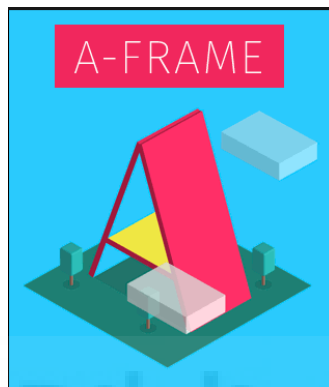


Figura 34 A-FRAME

A-Frame ofrece una extensión de HTML que permite la definición de objetos y escenas 3D en el navegador que se perciben como RV cuando se utilizan dispositivos de RV o móviles compatibles. Fue desarrollado originalmente por Mozilla en 2015, pero ha pasado a ser un proyecto *Open Source* con una amplia comunidad de usuarios.

Entre sus principales ventajas encontramos su sencillez ya que se basa en código HTML, por lo que resulta fácil trabajar añadiendo *scripts* para crear objetos o entornos complejos. A-Frame usa la arquitectura ECS (Entidades-Componentes-Sistema) que hace que cada objeto tenga un comportamiento diferente, la arquitectura está formada por:

- **Entidades:** son objetos contenedores donde los componentes son ligados para otorgarles propiedades.
- **Componentes:** son las propiedades que hacen que una entidad sea diferente a otra.
- **Sistemas:** proporcionan el entorno donde manejar y desarrollar los componentes.

Algunas características de A-Frame son:

- **Arquitectura basada en componentes:** es un potente entorno de three.js, que proporciona una estructura declarativa y reutilizable.

- **Rendimiento:** está optimizado desde cero para WebXR (y en sus primeras versiones para WebVR). Utiliza el DOM, pero sus elementos no tocan el motor del diseño del navegador. Se realizan en memoria con poca sobrecarga en una única llamada.
- **Inspector visual:** proporciona un inspector visual 3D integrado, que permite acceder al código rápidamente y modificarlo fácilmente para poder visualizar los resultados al instante.
- **Diseño basado en componentes:** incluye algunos componentes prediseñados como luces, animaciones, modelos, texto que son compatibles con la mayoría de los dispositivos. También es posible añadirle física e importar objetos más complejos como océanos, montañas...

Para este proyecto se ha usado A-Frame con JavaScript. A-Frame tiene una documentación y comunidad grande y esto hace que sea fácil poder acceder o pedir ayuda. Aunque nació en 2015, si tienes algún error, es fácil encontrarlo en sus foros o a través de publicaciones de usuarios activos en redes sociales (por ejemplo [@PrimaryObjects](#) o [@Michael Blade](#)) [47].

Además, para facilitar el diseño 3D, el inspector visual que ofrece permite modificar los componentes fácilmente, y eso es un alivio que facilita la comprensión del 3D y sus posiciones por coordenadas. En la Figura 35 se muestra como ejemplo una captura de pantalla de su documentación en la que se puede apreciar que es muy completa y ayuda bastante. Esto lo hace muy intuitivo a la hora de empezar desde 0 con él.



Figura 35 Documentación A-Frame

Al igual que en three.js y babylon.js, encontramos numerosos ejemplos que resultan de utilidad para aprender y afianzar conceptos. Por ejemplo, en la Figura 36 se pueden observar el listado de códigos de ejemplo que cubren un amplio abanico de la funcionalidad soportada por esta librería.

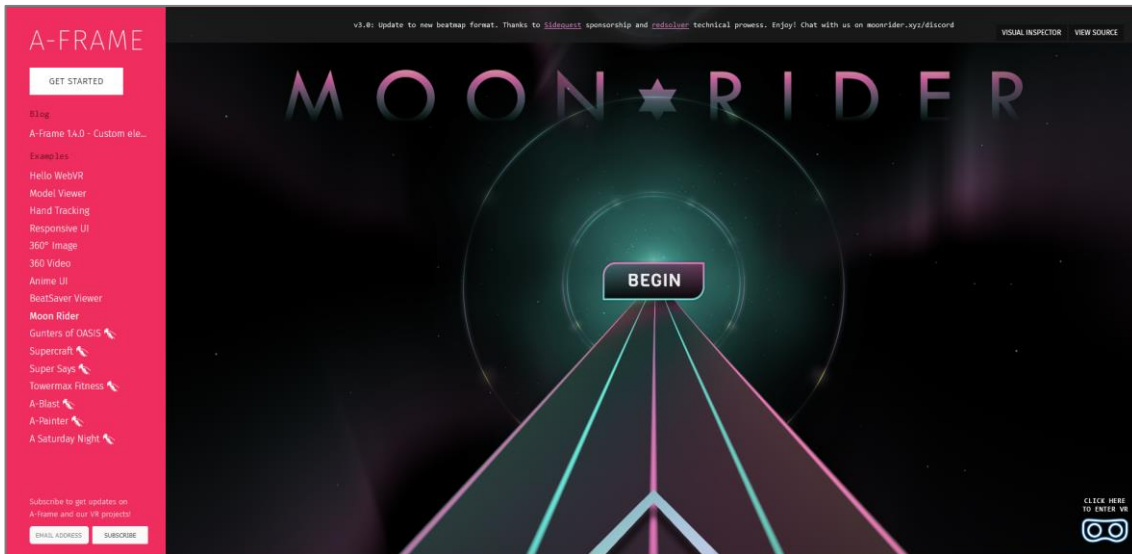


Figura 36 A la derecha se observa un listado de playground de A-Frame

A la hora de usar físicas en A-Frame se puede usar varias librerías como ammo.js, cannon.js u otras que incorporan comandos para poder hacer que el objeto tenga “física corporal” o “colisiones” que “rodeará” al cuerpo. Por ejemplo, en la Figura 37 se puede apreciar como el azul va a rebotar y se va a caer mientras el rojo como es grande, se queda en el sitio sin ningún problema.

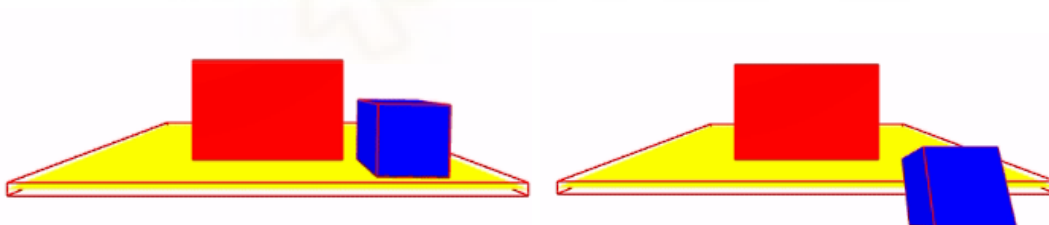


Figura 37 Físicas A-Frame

Por último, vamos a discutir sobre las posibilidades de acceso por parte de múltiples usuarios con A-Frame. Por defecto, un entorno 3D creado con A-Frame permite a los usuarios acceder a él a través del navegador web. Sin embargo, el acceso de cada usuario sería independiente y esto quedaría lejos de nuestro objetivo de lograr un Metaverso en el que varios usuarios puedan conectar a la vez e interactuar entre ellos.

Para solucionar este problema, y después de mucho investigar y probar diferentes opciones, encontramos el proyecto de GitHub networked-aframe [48]] que se define como un *framework* construido sobre A-Frame para escribir aplicaciones de RV multiusuario. Gracias a este *framework* podremos integrar en nuestro proyecto características centradas en la interacción entre los usuarios conectados en el mismo momento, como el audio de proximidad.

Merece la pena destacar que el uso de Networked A-Frame (NAF) exige el uso de Node.js.

- **NodeJS** [49]: es un entorno de tiempo de ejecución de JavaScript que se utiliza para crear aplicaciones escalables del lado del servidor y de red a través de servidores privados virtuales. Es requerido por NAF ya que implementa funcionalidad que permite a la librería una gestión del DOM orientada a realizar cambios en el entorno 3D en tiempo real.

Como hemos comentado A-Frame permite el desarrollo de aplicaciones web que pueden ser desplegadas en navegadores de dispositivos que permiten VR. Existen una gran variedad de dispositivos con un coste base por encima de los 300 euros. Esto dificulta las tareas de prueba y depuración ya que tener acceso a todos ellos resulta difícil. Para paliar los problemas de acceso a todos ellos, existen herramientas o simuladores que simulan la conexión desde un dispositivo de RV usando el navegador de escritorio tradicional. En nuestro caso haremos uso de Immersive Web Emulator.

- **Immersive Web Emulator** [50]: es una extensión de Google Chrome, que nos permitirá el uso de simular los controles de un dispositivo RV sin tenerlo.

En la Figura 38 se muestra una captura de pantalla de esta extensión. Podemos ver como se está probando una aplicación de RV implementada con WebXR a través del navegador de escritorio Chrome. Al utilizar la extensión se habilitará en la parte derecha las opciones del simulador. Permitirán elegir el dispositivo de VR desde el que se quiere acceder. Por ejemplo, en el caso de la Figura 38 se ha seleccionado unas gafas Quest Pro. Según el modelo de gafas seleccionado se habilitarán los controles y dispositivos propios del mismo que podrán usarse e interactuar con la vista simulada de la izquierda.

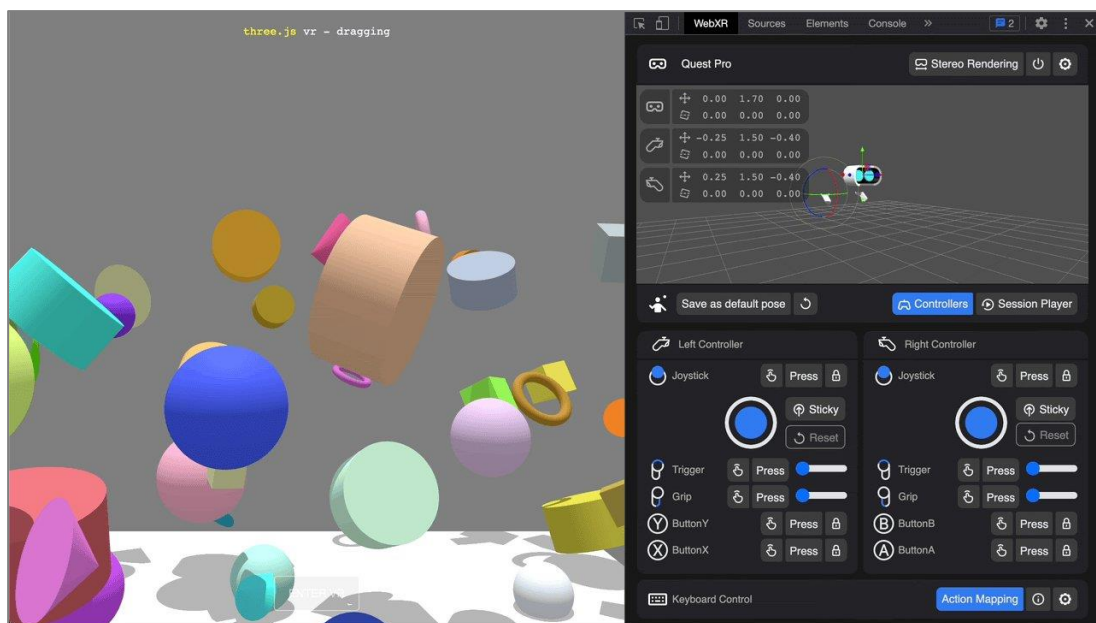


Figura 38 Ejemplo de uso de Immersive Web Emulator (imagen extraída de [50])



## 2.2.5- Resumen y comparativa de *frameworks* analizados

Para concluir este apartado resumimos en la Tabla 1 algunas de las características estudiadas para los tres *frameworks* analizados en este trabajo:

	<b>Babylon.js</b>	<b>Three.js</b>	<b>A-Frame</b>
<b>Documentación</b>	Bastante documentación y extensa	Detallada, pero es difícil de aprender	Extensa, clara y con ejemplos claros
<b>PlayGrounds</b>	Te sale con una base ya hecha, a partir de ahí puedes hacer lo que quieras.	Antiguo, tienes que empezar desde 0 cada vez que quieras usarlo	No hay en sí, simplemente tienes varios ejemplos completos que con ver el código puedes comprenderlo.
<b>Físicas</b>	Tiene muchas librerías que te permite hacer uso de una física para que sea más realista.	Tiene pocas librerías y no hay mucha documentación.	Tiene también bastantes librerías como Babylon.js ya que usaríamos las mismas.
<b>Posibilidad de RV o AR</b>	[51] Si	[52] Si	[53] Si
<b>Compatibilidad con CardBoard</b>	Con webVR sí, pero webXR no	Con webVR sí, pero webXR no	Con webVR sí, pero webXR no.

Tabla 1: Comparativa de las tecnologías Babylon.js, Three.js y A-Frame

## 2.2.6- Repositorios y modelos 3D para construir el Metaverso

Además de texturas y objetos básicos, A-Frame permite importar modelos 3D según diferentes formatos. La creación de nuestro metaverso va a requerir de la integración de diferentes objetos 3D que doten al espacio de atractivo visual. Crear todos los modelos desde cero supera el alcance de este proyecto, por eso se ha hecho uso de modelos públicos y accesibles a través de Internet.

A continuación, se describen los repositorios que se han escogido para obtener los modelos reutilizados en este proyecto:

- **Sketchfab [54] (ver Figura 39):** página web creada en 2012 para almacenar y visualizar diseños 3D. Contiene miles de modelos tanto de pago como gratuitos. Gracias a su visualizador esta web es una de las más reconocidas e importantes.

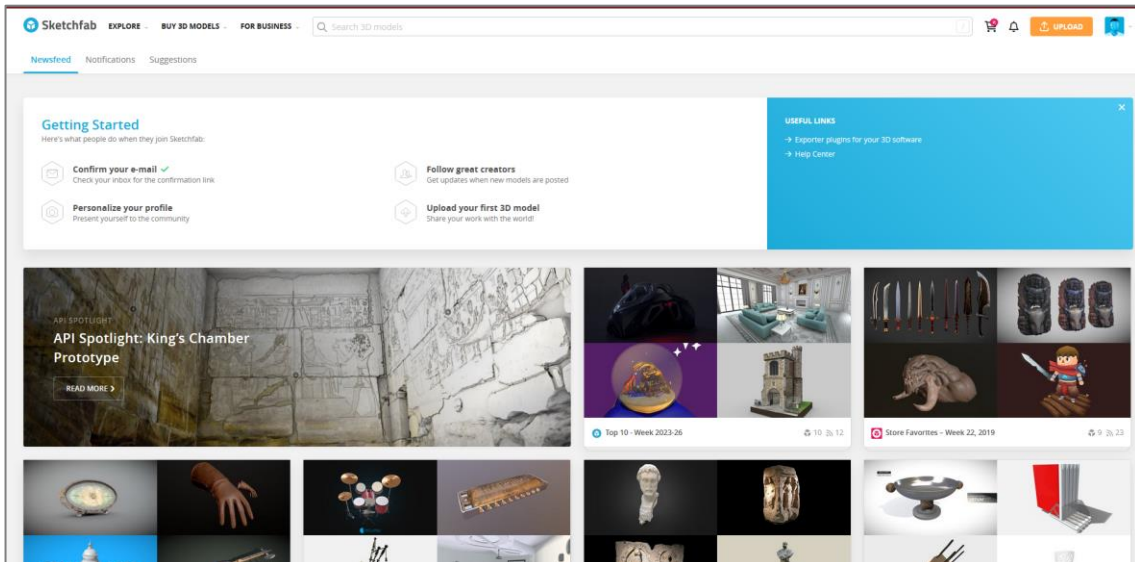


Figura 39 Menú de Sketchfab (imagen extraída de [55])

- **3DWarehouse [56] (ver Figura 40):** es un repositorio de muebles en 3D. Al entrar se encuentran miles de diseños de mesas, bibliotecas, cocinas, etc. También tiene un software que es 3DWarehouse Sketchup, que ayuda a modelar en 3D de código abierto y gratuito para que todo el mundo pueda acceder a esos modelos.

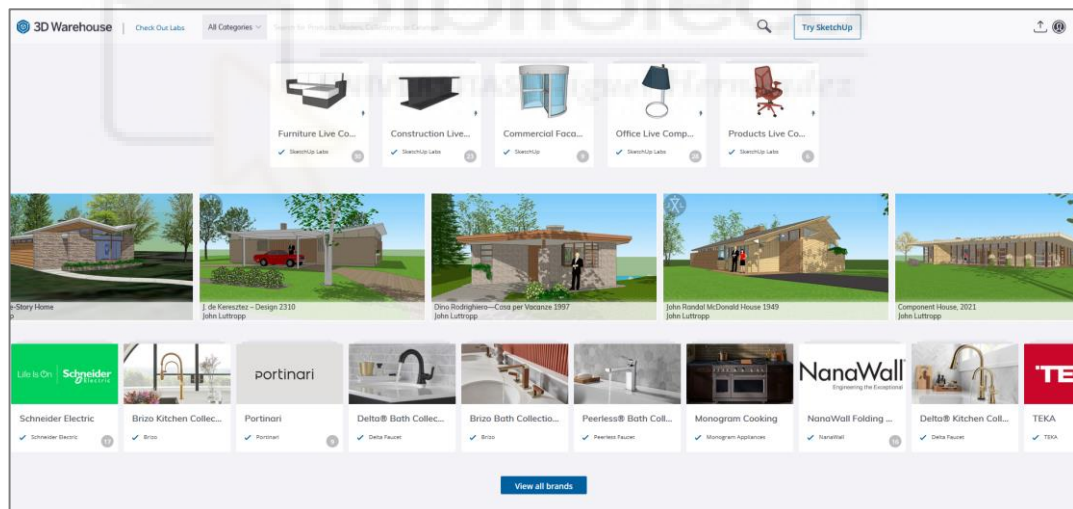


Figura 40 Menú de 3DWarehouse (imagen extraída de [57])

- **Poly.pizza[58] (ver Figura 41):** es otro gran repositorio donde tiene más de 7000 modelos distribuidos bajo licencia Creative Commons y de Dominio Público. En 2020 Google cerró su plataforma Poly, y entonces Zac Lucarelli, creó esta página web para que todo el mundo pudiera seguir teniendo acceso modelos con estética basada en polígonos.

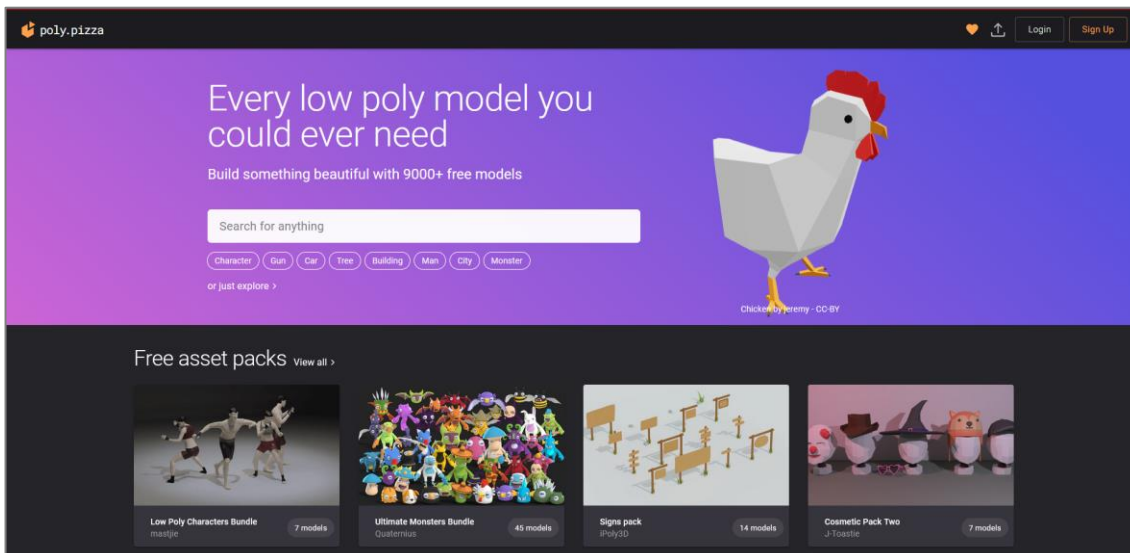


Figura 41 Menú de poly.pizza (imagen extraída de [59])

## 2.3- Herramientas utilizadas para el entorno de configuración

Los apartados anteriores se han centrado en el análisis de tecnologías centradas en la implementación del Metaverso 3D de RV usando tecnologías web. No obstante, nuestra solución también requerirá la implementación de un entorno web tradicional de pequeño alcance que servirá como herramienta de configuración de distintos aspectos relacionados con los Metaversos docentes.

El desarrollo de esta herramienta web de configuración del Metaverso requerirá del uso de tecnologías web y de bases de datos tradicionales. Entre las herramientas y tecnologías utilizadas destacamos las siguientes:

- **XAMMP** [60]: es un paquete de software libre para la gestión de una base de datos, un servidor web Apache y los intérpretes de PHP y Perl. El uso que se le ha dado en este proyecto es el de sistema de gestión de base de datos con MySQL para poder almacenar los datos necesarios.
- **MySQL** [61]: sistema de gestión de base de datos relacional. El uso de esta herramienta es para almacenar todos los modelos, referencias a vídeos, cuestionarios... que se cargan dinámicamente en cada mundo.
- **HTML5** [62]: código que se utiliza para estructurar y desplegar una página web y sus contenidos. Se ha usado principalmente porque es la base del código y sin esto no se podría crear nuestra página web.
- **CSS** [63] y **Bootstrap** [64]: CSS se ha usado para dotar de estilo y diseño a la web implementada para la gestión de salas y recursos. También hemos usado Bootstrap que es una biblioteca multiplataforma que ofrece un conjunto de herramientas de código abierto para el diseño de sitios y aplicaciones web.

## 2.4- Otras herramientas utilizadas

También hemos utilizado otras herramientas como:

- **PhpMyAdmin** [65]: es una herramienta con la intención de manejar la administración de MySQL a través de una página web. Se ha usado para administrar nuestra base de datos pudiendo añadir, borrar y modificar tablas.
- **Visual Studio Code** [66]: editor de código fuente desarrollado por Microsoft, incluye soporte para la depuración, control integrado de control de versiones o resaltado de sintaxis, entre muchas otras funciones. Se ha usado principalmente esta herramienta para escribir el código HTML y NodeJS.
- **Bitbucket** [67]: servicio de alojamiento que se usa para control de versiones. En este caso se ha utilizado como sistema de control de versiones del código desarrollado.

## 2.5- Propuesta de la solución

Para finalizar este capítulo, en la Figura 42 se resume gráficamente como quedan integradas las distintas herramientas y tecnologías que formarán parte de nuestra solución.

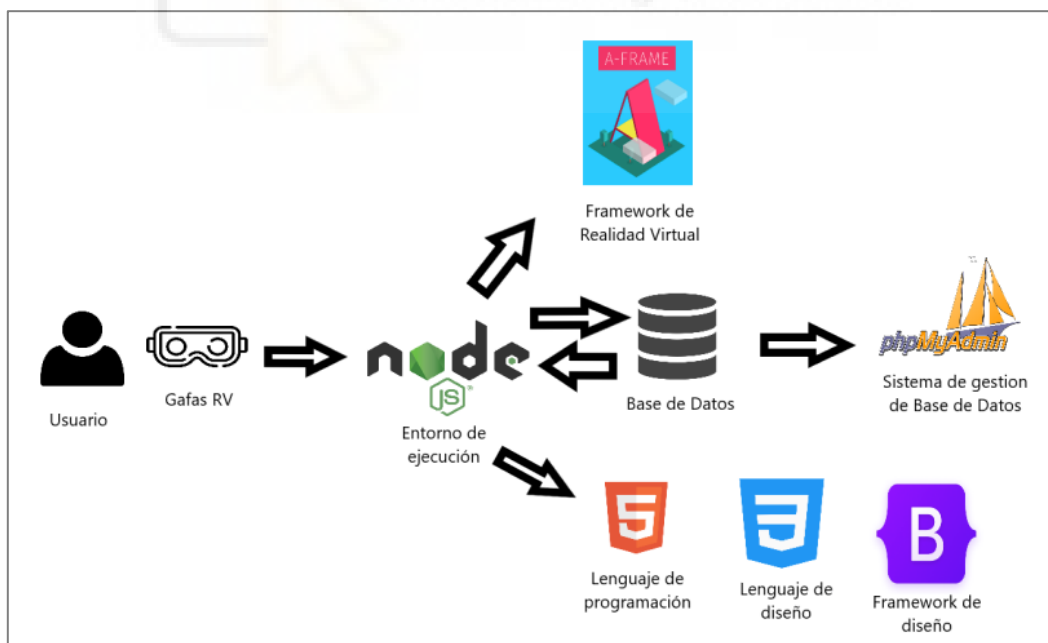


Figura 42 Propuesta de la solución

En el *frontend* el principal protagonista será el A-Frame por ser la tecnología sobre la que construiremos el Metaverso 3D de RV; con un menor alcance tendremos las tecnologías de HTML5, CSS y Bootstrap sobre las que se construirá una sencilla interfaz web de configuración. Estas dos vistas harán uso

de una base de datos relacional a la que accederemos desde las aplicaciones por medio de Node.js y haremos consultas durante el desarrollo de mantenimiento y gestión usando PhpMyAdmin. Por tanto, la tecnología que hará de transmisión entre el *frontend* y el *backend* será Node.js.



## 3.- Metodología y Resultados

En este capítulo se van a describir los aspectos relacionados con la planificación, el diseño y el desarrollo de software. Para finalizar se mostrará una descripción de la funcionalidad de la herramienta implementada.

### 3.1- Metodología

Se ha usado la metodología de desarrollo ágil Scrum [68]. Scrum es un marco de administración utilizado por equipos de trabajo para organizarse por cuenta propia y trabajar en aras de alcanzar un objetivo común.

Hay distintas fases (ver Figura 43):

- **Planificación:** es la fase en la que se establecen las tareas prioritarias y donde se obtiene información breve sobre el proyecto.
- **Sprint:** es el intervalo del tiempo (máximo de un mes) donde se produce el desarrollo de un producto que es entregable potencialmente.
- **Burn Down:** instante donde se mide el progreso del proyecto.

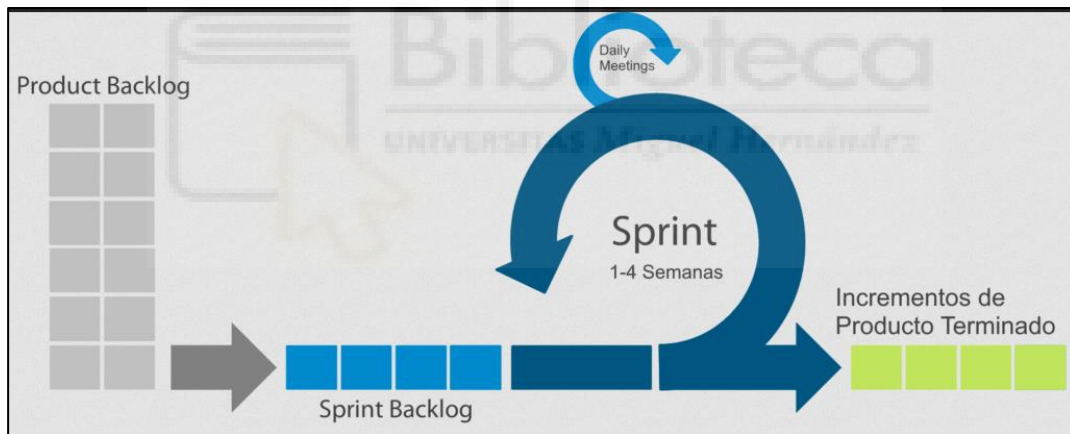


Figura 43 Scrum

En nuestro proyecto, aunque no contamos con un equipo de personas grande, hemos usado Scrum por su agilidad a la hora de avanzar a través de los *sprints*. Debido a los requisitos cambiantes y los constantes retos tecnológicos que iban surgiendo, el director del trabajo y yo establecimos reuniones periódicas, en las que semanalmente nos manteníamos actualizados discutiendo el avance del proyecto a partir de un *feedback* sobre los objetivos establecidos.

A continuación, se va especifican las tareas realizadas en las distintas fases de este trabajo:

- **Fase 0: Investigación y estado del arte**
  - **0.1.** Investigación de herramientas relacionadas.
  - **0.2.** Investigación y elección de tecnología para desarrollo VR.

- **0.3.** Investigación de uso del *framework* A-Frame.
- **0.4.** Investigación y elección de repositorios para modelos 3D.
- **0.5.** Investigación de uso del *framework* Networked A-Frame.
- **Fase 1: Diseño del Metaverso**
  - **1.1.** Instalación y creación del proyecto en Visual Studio Code.
  - **1.2.** Instalación de NodeJS junto a npm.
- **Fase 2: Diseño de la aplicación**
  - **2.1.** Desarrollo de una escena en RV para móvil.
  - **2.2.** Cambio de RV para móvil a dispositivos.
  - **2.4.** Creación de la base de datos.
  - **2.3.** Implementación de modelos y vídeos.
  - **2.4.** Implementación del Networked A-Frame.
- **Fase 3: Despliegue del proyecto**
  - **3.1.** Configuración e inicio del servidor.
  - **3.2.** Configuración y puesta en marcha de Networked A-Frame.
- **Fase 4: Documentación y memoria del proyecto**
  - **4.1.** Análisis y diseño del software
  - **4.2.** Ejemplo de la aplicación.
  - **4.3.** Documentar la memoria.

En la Figura 44 se puede observar el cronograma con el desarrollo temporal de las distintas fases enumeradas.

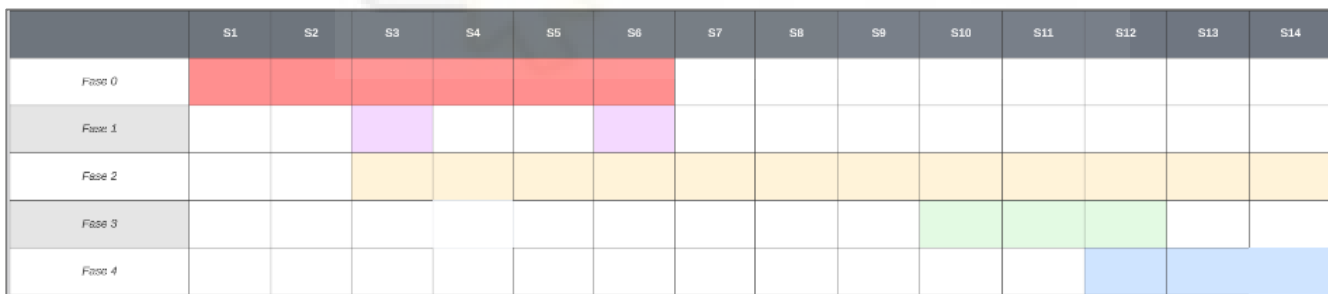


Figura 44 Cronograma de las distintas fases del uso de la metodología Scrum

### 3.2- Casos de Uso

En este apartado describiremos los aspectos más importantes de los Casos de Uso. Principalmente, hemos identificado dos actores: el Profesor y los Alumnos.

El Profesor es el que mayor permiso tiene ya que podrá acceder tanto al Metaverso como a la interfaz web de configuración desde la que gestionar los distintos recursos asociados a las asignaturas. Por ejemplo, el Profesor podrá realizar acciones como: la personalización de los Metaversos para que cambien dinámicamente o asociar alumnos a las distintas asignaturas. Por otro lado, el Alumno solo podrá acceder a las salas de las asignaturas en las que está

matriculado, entendiendo cada sala como un Metaverso diferente y personalizado con los recursos educativos de la asignatura.

En la Figura 45 se muestra un diagrama general que recoge los distintos Casos de Uso identificados, y que son descritos con mayor detalle en el Anexo I. A modo de ejemplo, y por no sobrecargar este apartado, se muestra en la Tabla 2 un ejemplo del Caso de Uso relacionado con el acceso al Metaverso.

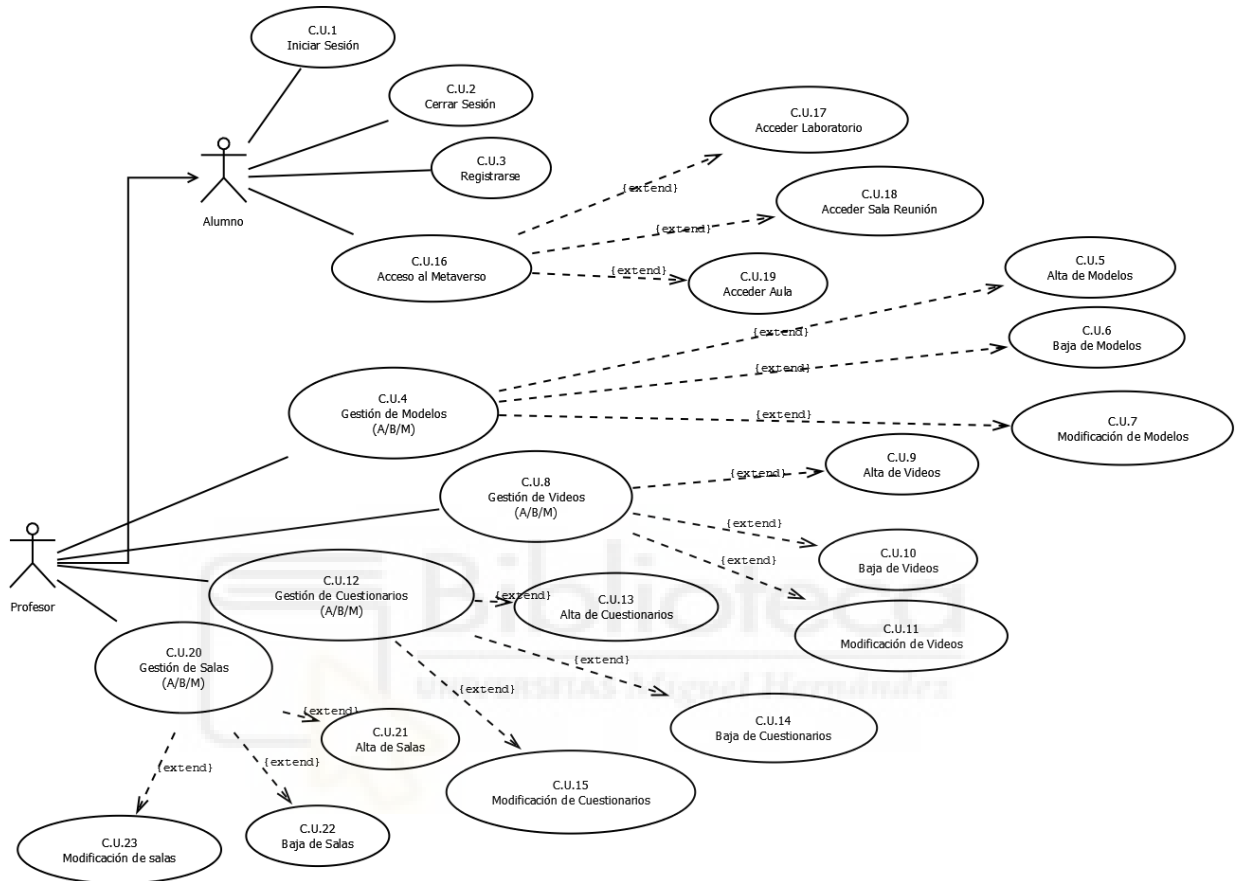


Figura 45 Casos de Usos

<b>CU-16</b>	<b>Acceder al Metaverso</b>
<b>Actores</b>	<u>Profesor y alumno</u>
<b>Descripción</b>	Los usuarios iniciaran sesión en la página web
<b>Precondición</b>	Tener acceso a Internet y cuenta
<b>Secuencia</b>	<ol style="list-style-type: none"> <li>1. El usuario accede a la página web.</li> <li>2. El usuario se autentica correctamente.</li> <li>3. El usuario visualiza su listado de asignaturas.</li> <li>4. El usuario accede al Metaverso de una asignatura haciendo clic sobre ella.</li> <li>5. El usuario accede a la vista 3D de la aplicación donde podrá habilitar la opción de RV.</li> </ol>
<b>Postcondición</b>	<p>Si en el punto 1 no tiene Internet, no podrá acceder a la página.</p> <p>Si en el punto 2 falla el inicio de sesión (por ser incorrecto o no tener cuenta) aparecerá un mensaje indicando el tipo de error.</p>

Tabla 2: Ejemplo del CU-16 desarrollado y relacionado con el acceso al Metaverso



### 3.3- Diagramas de secuencia y mapa de navegación

En nuestro contexto, un Metaverso es un mundo virtual dinámico y compartido con otros usuarios. Tendremos un Metaverso por cada una de las asignaturas, de manera que sirva de punto de reunión para todos los alumnos matriculados y los profesores que las imparten.

Desde el punto de vista de la interacción del usuario con el mundo, en las Figuras 46 y 47 se puede observar un diagrama de secuencia en el que se muestra la interacción de los usuarios con las distintas partes que componen la aplicación desarrollada.

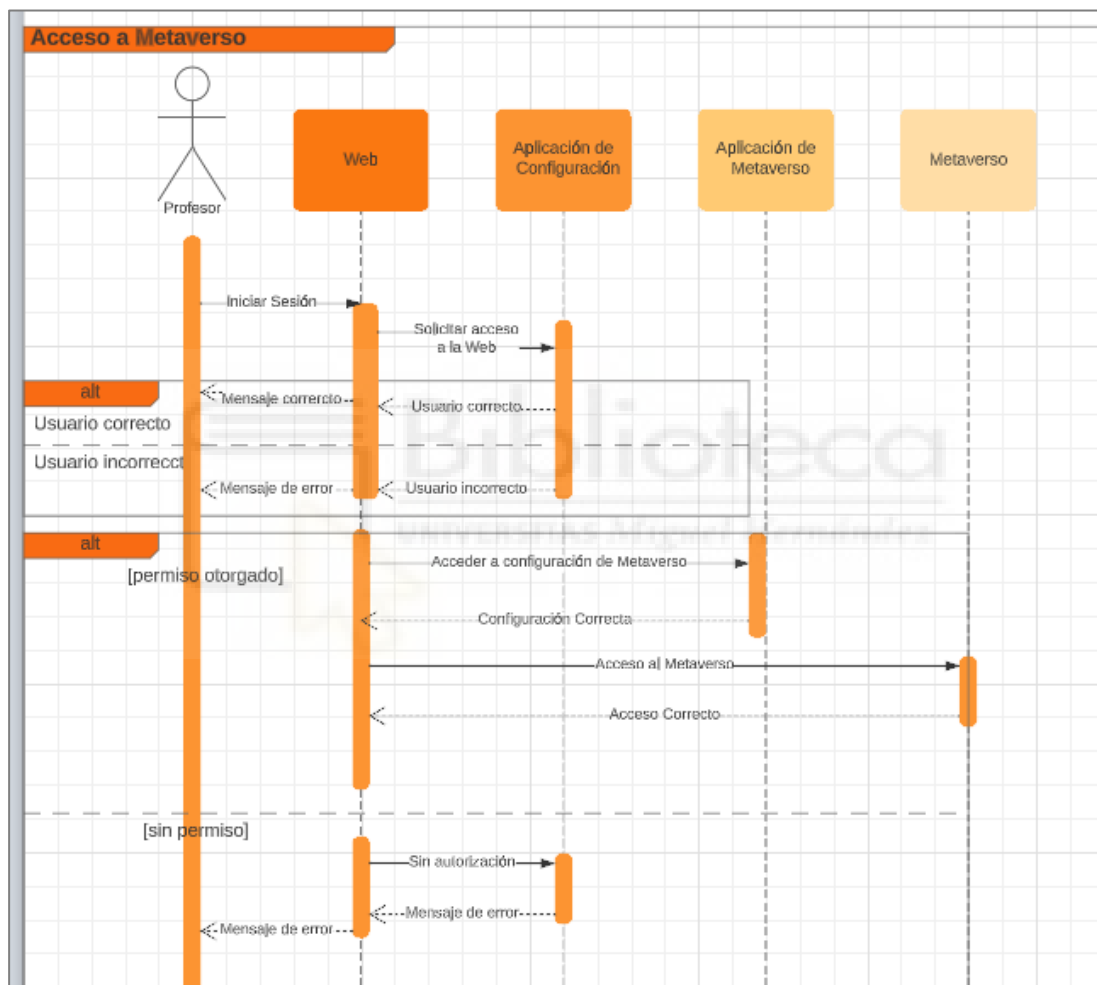


Figura 46 Diagrama de Secuencia para el rol de Profesor

En la Figura 46 se muestra la interacción del profesor. Como vemos este perfil de usuario tendrá acceso a la aplicación de configuración y a la aplicación de Metaverso, que será una aplicación web tradicional desde la que podrá realizar la gestión de las distintas asignaturas que imparte. Por ejemplo, podrá crear una asignatura y asignarle alumnos; tras ello, el sistema creará un nuevo Metaverso y una URL desde la que acceder a él. Además, el profesor también podrá acceder al Metaverso de cada una de sus asignaturas para interactuar con sus alumnos en tiempo real si están conectados.

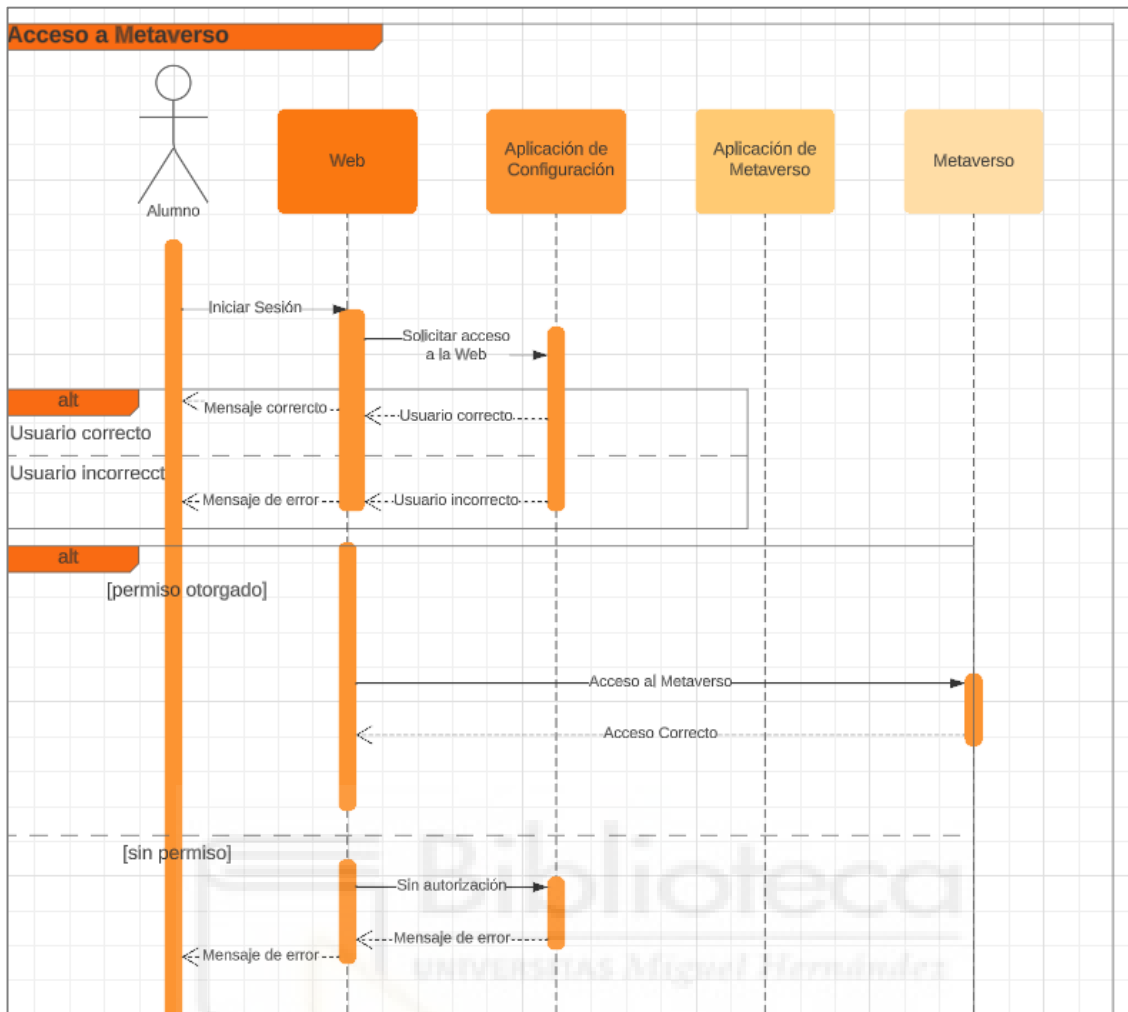


Figura 47 Diagrama de Secuencia para el rol de Usuario

Como ya se ha comentado, un usuario podrá acceder a tantos Metaversos como asignaturas esté matriculado. En cuanto a las posibilidades que ofrece cada uno de dichos Metaversos, en la Figura 48 se representa en un mapa de navegación las distintas opciones ofrecidas por nuestro entorno virtual.

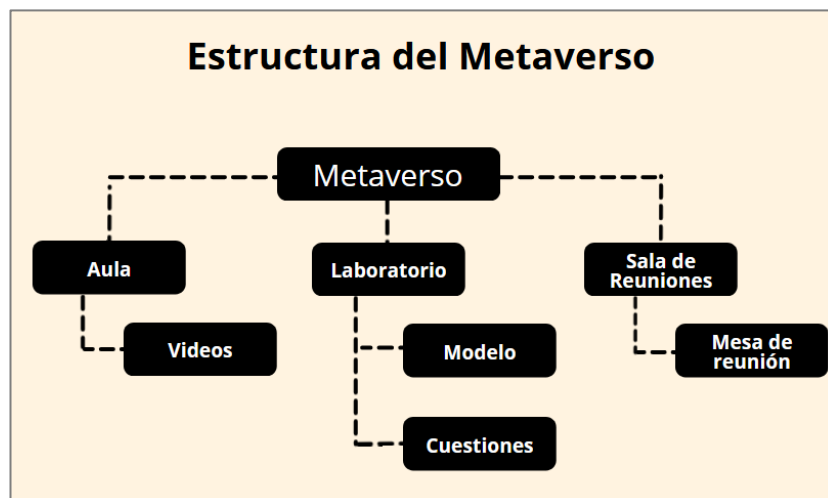


Figura 48 Mapa de navegación del Metaverso

A continuación, describimos brevemente la utilidad de cada uno de los tres tipos de salas que componen nuestro entorno 3D:

- **Aula:** este espacio pretende simular un aula de teoría tradicional dotada de una zona de “proyector” en la que podrían visualizarse recursos educativos basados en vídeos. Estos vídeos serán configurados y subidos por el profesor de la asignatura. Estéticamente estará compuesta por una zona para el alumnado con distintas mesas dispuestas en filas y una mesa para el profesor.
- **Sala de reuniones:** este espacio pretende simular un lugar compartido por los distintos usuarios en el que poder realizar reuniones usando el audio de proximidad. El audio de proximidad utilizará los micrófonos de los dispositivos de cada usuario e interactuará con otros en función de la distancia que exista en el mundo virtual entre sus avatares.
- **Laboratorio:** en algunos contextos educativos el uso de modelos 3D es muy habitual. Por ejemplo, los grados de ciencia de la salud utilizan modelos físicos sobre los que aprender conceptos de anatomía, o en algunas ingenierías son utilizados para aprender mecánica. Hemos querido ofrecer en nuestro Metaverso un lugar que permita llevar estas prácticas docentes a un entorno virtual a través del 3D. Para ello, el laboratorio será un lugar donde se podrá ver los modelos en 3D junto a un cuestionario que serán preguntas sobre dicho modelo.

Por tanto, en esta primera versión de nuestro Metaverso, se le ofrecerá al profesor los siguientes tipos de recursos educativos que podrá subir y relacionar con sus asignaturas:

- **Vídeos:** se podrán asociar itinerarios basados en vídeos que estarán disponibles para los alumnos en el Aula del Metaverso.
- **Modelo 3D:** si el profesorado cuenta con material 3D en formato. glb, podrá subirlo y quedará disponible para poder ser visualizado y manipulado en el laboratorio.
- **Cuestionarios:** se podrán asociar con los modelos 3D cuestionarios con enunciado y 4 opciones que serán accesibles por el alumnado desde el laboratorio. El alumnado podrá explorar libremente el modelo 3D y a su lado podrá interactuar con el cuestionario que contendrá preguntas sobre el modelo. El cuestionario será interactivo y ofrecerá en directo *feedback* sobre si la selección de la respuesta es correcta o no.

Dado el alcance y el límite de este trabajo, hemos implementado un listado limitado de salas y recursos como los descritos anteriormente. Se propondrá como trabajo futuro la creación de más tipos que puedan cubrir un mayor número de prácticas docentes.

### 3.4- Diseño de la Base de Datos

La base de datos es importante en el desarrollo de la aplicación, ya que es donde se almacenarán todos los datos sobre los usuarios, las asignaturas, y los distintos recursos que compondrán el Metaverso de cada una de las asignaturas.

En la Figura 49 se muestra el diagrama Entidad-Relación de nuestra base de datos, cuya creación y mantenimiento se ha realizado usando la herramienta phpmyadmin.

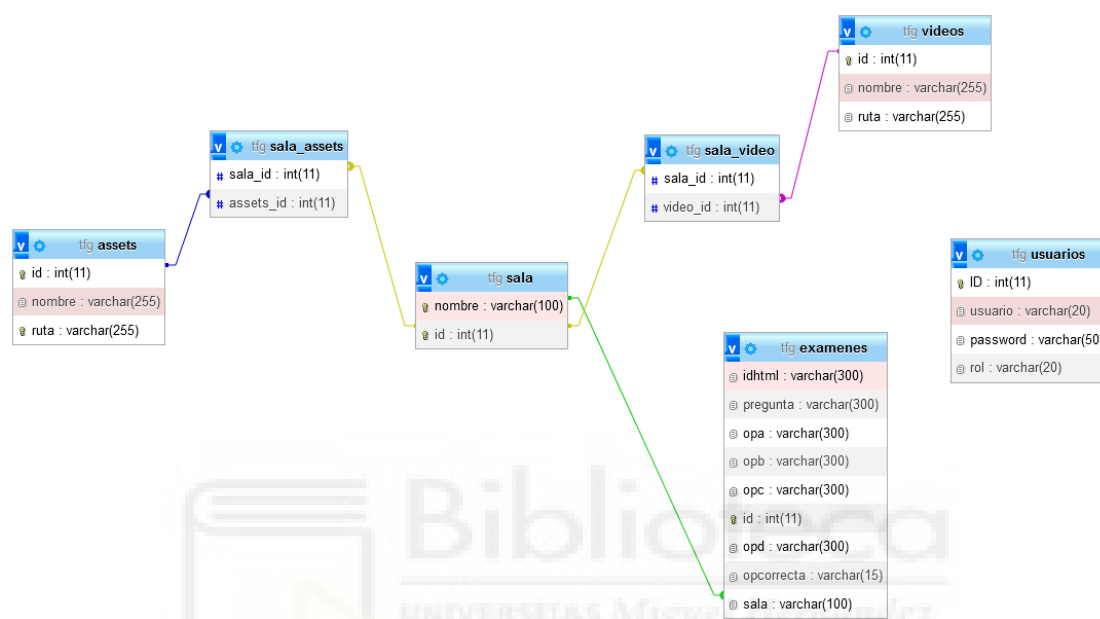


Figura 49 Modelo E-R

A continuación, explicamos en qué consisten cada una de las tablas:

- **Usuarios:** almacenará las credenciales de los usuarios junto a sus roles para poder acceder en el menú a ciertas opciones de desarrollo.
- **Assets:** contendrá el nombre y ruta de todos los modelos 3D. Merece la pena destacar que los modelos 3D estarán relacionados tanto con el material educativo que suba el docente, como con el propio entorno 3D estático que formará parte del Metaverso como árboles, paredes, sillas... El código fuente de los modelos 3D será almacenado directamente en el directorio de la aplicación web, por eso aquí necesitamos guardar su URL de acceso.
- **Sala\_assets:** se utiliza para relacionar cada sala con los assets que la componen.
- **Sala:** contiene información sobre las diferentes salas de que formarán nuestro metaverso. Merece la pena destacar que tendremos una sala por cada asignatura, por tanto, y en línea con la nomenclatura que hemos usado anteriormente entendemos que cada sala será un Metaverso.
- **Sala\_video:** se utiliza para relacionar y almacenar qué vídeos están asociados a cada asignatura.

- **Video:** almacena el nombre del video junto a la URL para acceder a él desde el código. Al igual que los modelos 3D, el vídeo en sí será almacenado directamente como un recurso del proyecto web.
- **Exámenes:** sirve para almacenar las preguntas, las respuestas y la respuesta correcta de los cuestionarios asociados a los modelos 3D.

### 3.5- El uso de modelos 3D en el Metaverso

En este apartado se va a hablar sobre algunos modelos que se han usado para componer el Metaverso. Como se comentó anteriormente, mientras algunos modelos han podido ser reutilizados directamente, otros han sido creados desde cero o modificados a través de sus texturas para lograr una mayor integración.

Por ejemplo, en la Figura 50 se puede ver un modelo que ha sido descargado de Sketchfab [69]; usaremos este árbol para integrarlo en la parte exterior el Metaverso desde la que se podrá acceder a las distintas salas.



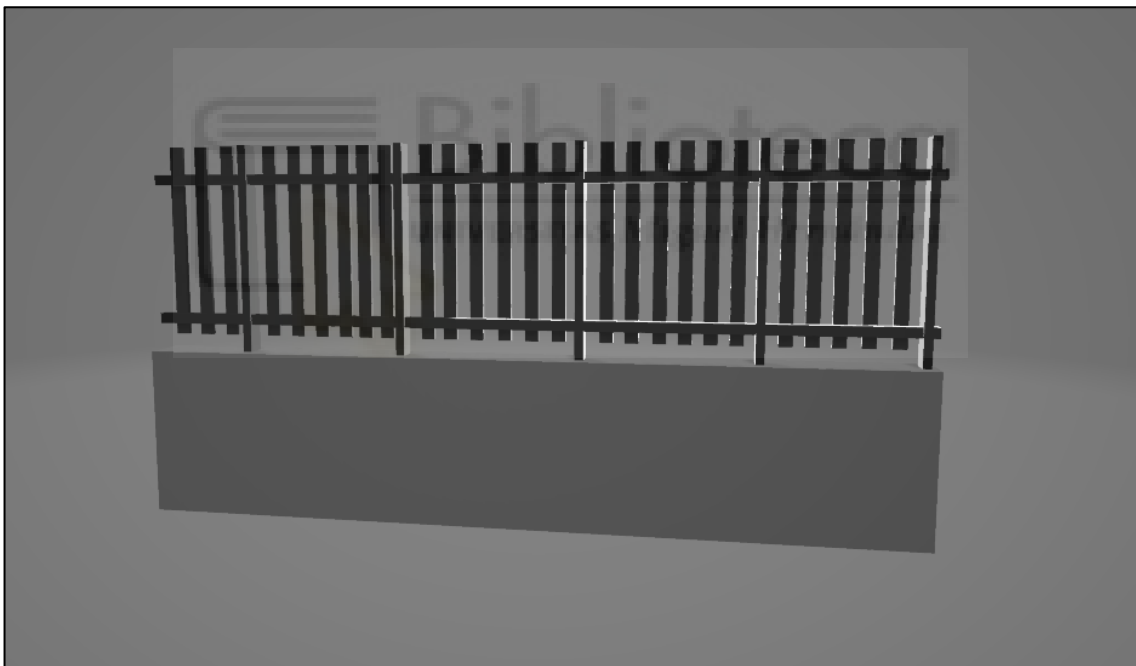
*Figura 50 Modelo de un árbol en. Glb*

Un ejemplo de modelo hecho desde 0 con la aplicación Blender es la silla que se puede observar en la Figura 51. Esta silla formará parte del Aula que servirá de punto común para que los alumnos visualicen vídeos o compartan sus experiencias a través del audio de proximidad.



*Figura 51 Modelo de la silla en. glb*

Por último, en la Figura 52 se muestra un ejemplo de una valla que ha sido reutilizada [70] y que es usada para delimitar el recinto del Metaverso de una asignatura y desde donde podrá accederse a los diferentes espacios.



*Figura 52 Modelo de la valla en formato. glb*

### **3.6- Algunos aspectos destacables de implementación**

En este apartado se resumirán los aspectos relacionados con la implementación del código y que supusieron un desafío durante el proceso de desarrollo. Se debe tener en cuenta que en esta sección se describirán los aspectos más relevantes del código, con el fin de describir el uso de las tecnologías A-Frame y Networked A-Frame, de manera que sirva de documentación útil para alumnos que se animen a continuar con el trabajo futuro.

### 3.6.1- Instalación del entorno: Node.js y XAMPP

Para trabajar con Node.js hay que descargar de la página oficial el ejecutable [71], que incluye también el gestor de paquetes `npm`. Para comprobar que todo ha funcionado correctamente, una vez instalado escribimos en la consola:

1.	<code>Node -version</code>
2.	<code>Npm -version</code>

Si el proceso ha funcionado correctamente, el resultado en la consola debería informar sobre las versiones de ambos productos de forma similar a la Figura 53.

```
C:\Users\Toni>node --version
v16.17.1

C:\Users\Toni>npm --version
9.6.5
```

Figura 53 Versiones node.js y npm

También tendremos que descargar e instalar XAMPP [72]. En nuestro caso no utilizaremos todos los servicios ofrecidos por él. Por ejemplo, no usaremos Apache ya que utilizaremos Node.js por ser requerido por el *framework* Networked A-Frame. Sin embargo, sí utilizaremos MySQL para gestionar y almacenar nuestra base de datos.

En la Figura 54 se muestra el panel de control de XAMPP. Para iniciar sus servicios deberemos hacer clic en el botón “*Start*” asociado a cada uno de ellos.

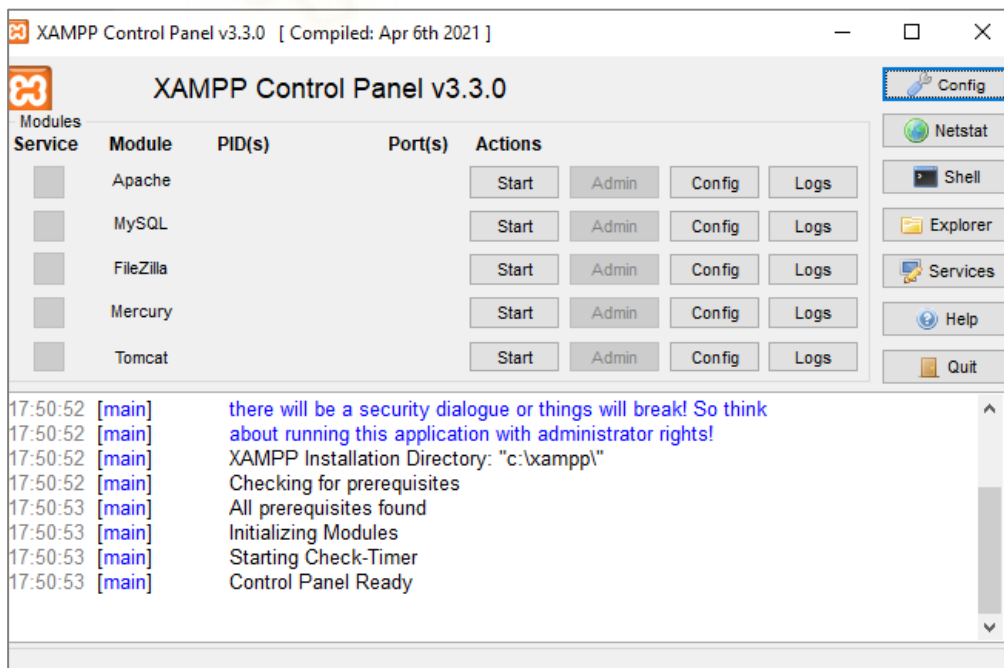


Figura 54 XAMPP

Para la gestión de la base de datos se utilizará phpmyadmin. Con él podremos crear, mantener y consultar la base de datos que será utilizada por nuestra aplicación. En la Figura 55 se muestra un ejemplo de uso de la herramienta phpmyadmin sobre nuestra BBDD.

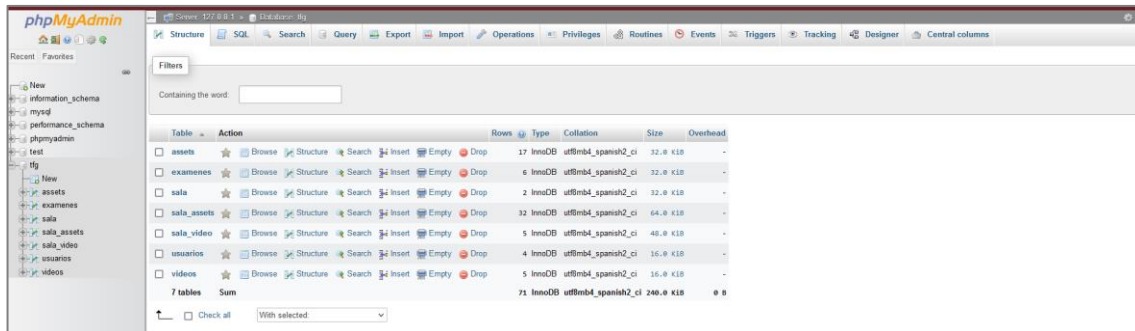


Figura 55 Menú de PHPMYADMIN realizando una consulta sobre nuestra base de datos

### 3.6.2- Integración de A-Frame en los ficheros HTML5

Como hemos comentado previamente, la programación con A-Frame se realiza completamente integrada en un fichero .html tradicional; A-Frame extiende la sintaxis de HTML con nuevos elementos que son los definidos en la documentación del *framework*.

Por ejemplo, en la Figura 56 se muestra una captura de pantalla de un fragmento del fichero principal (*index.html*) donde se definen las texturas que compondrán los suelos y las paredes de nuestro Metaverso. En ella podemos observar el uso de los elementos *a-plane*, *a-sky* o *a-gltf-model* implementados por el *framework*. Además, a través del elemento *a-gltf-model* integramos en el entorno modelos 3D previamente importados, que en este caso corresponden con una fuente central de la parte exterior de nuestro entorno.

```
<a-plane static-body material="src:assets/floor2.jpg;repeat: 100 100" height="60" width="100" rotation="-90 0 0"></a-plane>
<a-sky src="#sky"></a-sky>
<a-gltf-model src="#fuente" scale="1 1 1" rotation=" 0 270 0 " position="3 0.3 -13"></a-gltf-model>
```

Figura 56 Código donde se indica el suelo, cielo y la fuente

### 3.6.3- Importación de modelos 3D externos en A-Frame

Además de las texturas y elementos básicos 3D, A-Frame permite importar modelos más complejos, diseñados externamente en herramientas de edición 3D profesional como Blender y guardados en formatos específicos de estas herramientas [73]. Para importar un modelo externo deberemos hacer un uso combinado de los elementos *a-assets*, *a-assets-item* y *a-entity*. El elemento *a-assets-item* permite importar el modelo externo que será enlazado a través de una URL con el fichero que almacena modelo. Por ejemplo, en la Figura 57 se muestra un ejemplo de código extraído de la documentación



en el que se muestra el uso combinado de las tres etiquetas anteriores. Merece la pena destacar el papel que juega el atributo `id` y a través del cual se establece la relación entre el modelo importado con `a-asset-item` y su uso a través del elemento `a-entity` (especificando el `id` a través del atributo `gltf-model`).

```
<a-assets>
  <a-asset-item id="tree" src="/path/to/tree.glTF"></a-asset-item>
</a-assets>

<a-entity glTF-model="#tree"></a-entity>
```

Figura 57 Importación del modelo con la URL y para meterlo en el mundo (imagen extraída de [73])

### 3.6.4- Carga dinámica de los distintos Metaversos

En este proyecto definiremos los distintos elementos de nuestros Metaversos en un fichero `.html` que servirá como plantilla. En esta plantilla diferenciaremos entre dos tipos de modelos: comunes a todos los Metaversos y específicos de cada asignatura. Por ejemplo, los árboles y la infraestructura de las distintas salas será algo común entre distintas asignaturas, pero los modelos 3D y cuestionarios variarán según la configuración elegida por el profesor.

Como se comentó en el apartado de la BBDD, tenemos guardado en la BBDD cuales son los `assets` disponibles y sus rutas. También queda almacenada la relación entre los `assets` y las diferentes salas o Metaversos. Por ejemplo, en la Figura 58, se puede observar un ejemplo de la información guardada sobre un `assets` que representa la mesa del Laboratorio.

	id	nombre	ruta
<input type="checkbox"/>	3	mesa_lab	assets/mesa_laboratorio.glb

Figura 58 Información de un `asset` almacenado en nuestra BBDD

En la parte del *backend*, cuando un usuario solicite al servidor un Metaverso relacionado con una determinada asignatura, se recuperará la información de los `assets` relacionados con él a través de una consulta realizada sobre la BBDD, que será enviada al cliente usando formato JSON codificado en una variable del JavaScript de la página web. Por ejemplo, en la Figura 59 se muestra como a través de una consulta SQL se obtienen los `assets`, los vídeos y cuestionarios específicos del Metaverso de una asignatura.

```
// Consulta para obtener los assets relacionados con la sala
const assetQuery = `SELECT a.id, a.ruta, s.id, a.nombre
FROM assets a
JOIN sala_assets sa ON a.id = sa.assets_id
JOIN sala s ON sa.sala_id = s.id
WHERE s.nombre = ?
`;
```

Figura 59 Consulta SQL para obtener los `assets` relacionados con una asignatura

Una vez en el *frontend*, se cargarán en el DOM de la página web y de forma dinámica los distintos *assets* contenidos en el JSON mencionado. Esta carga se realizará en JavaScript cuando la página web completa haya sido recibida y cargada por el navegador. Para ello se asociará el procedimiento de carga con el evento `DOMContentLoaded` como se puede observar en la Figura 60. Se puede observar que la variable `assets` es la que contienen la información recuperada de la BBDD por el *backend*.

```
window.addEventListener('DOMContentLoaded', function() {
  const assetsContainer = document.getElementById('assets-container');
  assets.forEach((asset) => {
    const assetItem = document.createElement('a-asset-item');

    assetItem.id = asset.assetId;
    assetItem.src = asset.assetSrc;
    const assetSrc = asset.assetSrc;
    assetItem.setAttribute('src', assetSrc);
    assetsContainer.appendChild(assetItem);
  });
});
```

Figura 60 Lado cliente sacar JSON

En la Figura 61 se puede consultar otro ejemplo de cómo esta información recuperada y codificada en el JSON es utilizada por nuestro fichero base (`index.html`).

```
const exámenes = examenResults.map((row) => {
  return {
    idhtml: row.idhtml,
    pregunta: row.pregunta,
    opa: row.opa,
    opb: row.opb,
    opc: row.opc,
    opd: row.opd,
    opcionCorrecta: row.opcorrecta,
  };
});

const videoUrls = videoResults.map((row) => row.ruta);
const assets = assetResults.map((row) => ({
  assetId: row.nombre,
  assetSrc: row.ruta
}));

const indexPath = path.resolve(__dirname, '../examples/index.html');
const indexContent = fs.readFileSync(indexPath, 'utf8');
const modifiedIndexContent = indexContent
  .replace('{{videoUrls}}', JSON.stringify(videoUrls))
  .replace('{{assets}}', JSON.stringify(assets))
  .replace('{{exámenes}}', JSON.stringify(exámenes));

response.send(modifiedIndexContent);
```

Figura 61 Enviar JSON lado servidor

Merece la pena destacar que, en nuestra plantilla, todos los elementos 3D son localizados en las distintas partes del plano a través del atributo `position` del elemento `a-ghost-model`. También se utiliza su atributo `scale` para ajustarlo en una escala razonable.

### 3.6.5- Audio de proximidad e interacción entre avatares

A través de la funcionalidad ofrecida por la extensión *Networked A-Frame* (NAF) conseguimos que los Maversos estén compartidos por los distintos usuarios que están conectados a la vez. NAF permite hacer una sincronización de entidades y componentes de A-Frame entre los diferentes usuarios. Es decir, cada vez que un usuario se conecte al Metaverso de una asignatura, se generará un avatar que será visible por otros que estuvieran conectados previamente. Además, los movimientos de todos ellos serán compartidos en tiempo real dotando al Metaversos de vida.

Desde el punto de vista de la configuración de nuestro entorno, para crear y unirse a una sala con el NAF habilitado hay que agregar el componente de escena en red a través del atributo `networked-scene` como se puede ver en la Figura 62.

```
<a-scene networked-scene="audio:true; adapter: easyrtc; room: tfg; debug: true;">
```

Figura 62 Etiqueta para crear la escena 3D

Además de la sincronización de entidades a través de avatares, NAF ofrece la funcionalidad para que dichas entidades implementen el audio de proximidad. Esto permitirá a los usuarios conectados poder usar el micrófono y hablar con los otros a través de un chat de proximidad. En la Figura 63 se muestra como una entidad, habilitando su atributo `networked-audio-source`, es configurada para usar su micrófono.

```
<a-entity class="avatar" networked-audio-source>
```

Figura 63 Etiqueta para crear el avatar y habilitarle el audio de proximidad

Además de lo anterior, para que el NAF funcione, se deberá configurar un socket compartido o adaptador cuyo nombre se actualizará dinámicamente con JavaScript, y cuyo ejemplo se puede ver en el código de la Figura 64. En ella se puede apreciar cómo se inicializa a través de la función `easyrtc` que se parametriza con el nombre de cada sala para que no haya interacciones entre los usuarios de los distintos Metaversos.

```
easyrtc.listen(app, socketServer, null, (err, rtcRef) => {  
  console.log("Initiated");  
  if (rtcRef && rtcRef.events) {  
    rtcRef.events.on("roomCreate", (appObj, creatorConnectionObj, nombreSala, roomOptions, callback) => {  
      console.log("roomCreate fired! Trying to create: " + nombreSala);  
      appObj.events.defaultListeners.roomCreate(appObj, creatorConnectionObj, nombreSala, roomOptions, callback);  
    });  
  }  
});
```

Figura 64 Función para crear la sala con el nombre dinámico

### 3.6.6- Interacción del usuario con los elementos del Metaverso

Aunque la mayoría de los elementos 3D del metaverso se mantienen estáticos, el usuario podrá interactuar con los modelos, vídeos y cuestionarios cargados en el Metaverso de una asignatura.

- **Interacción con los vídeos:** se implementará la interacción a través de JavaScript y en el lado del cliente a través del API ofrecida por el elemento `a-video` (ver en la Figura 65). Merece la pena mencionar que esta etiqueta hereda la funcionalidad definida para el elemento `video` original de HTML5, y esta compatibilidad es la que hace que A-Frame sea más intuitivo de utilizar que otras opciones anteriormente mencionadas (`three.js` y `Babylon.js`). Además, usando el elemento `a-image` se definirán tres botones (play, pause y siguiente).

```
<a-video src="#mi-video" rotation=" 0 270 0" position="8.5 3 11" scale="5 3 3"></a-video>
<a-image src="#pause" onclick="Pause()" rotation=" 0 270 0" scale="0.5 0.5 0.5" position="8.5 0.8 12"></a-image>
<a-image src="#play" onclick="Play()" rotation=" 0 270 0" scale="0.5 0.5 0.5" position="8.5 0.7 11"></a-image>
<a-image src="#next" onclick="Next()" rotation=" 0 270 0" scale="0.5 0.5 0.5" position="8.5 0.8 10"></a-image>
```

Figura 65 Etiquetas de vídeo e imagen

Para completar el código de la Figura 65, en la Figura 66 se muestra la implementación de la función JavaScript que permite interactuar en tiempo real con los vídeos a través del DOM.

```
function changeVideo(num_video,videoUrls)
{
    myVideo = document.getElementById('mi-video');
    myVideo.src = videoUrls[num_video];
    myVideo.load();
    myVideo.currentTime = 0;
    myVideo.play();
}

function Pause()
{
    myVideo.pause()
}

function Play()
{
    myVideo.play()
}

function Next() {
    if (num_video >= videoUrls.length-1) {
        num_video=0;
        changeVideo(num_video,videoUrls);
    }
    else
    {
        num_video++;
    }
    changeVideo(num_video,videoUrls);
}
```

Figura 66 Funciones para interactuar con la lista de vídeos

- **Interacción con los objetos del laboratorio:** la carga dinámica de los cuestionarios se hace de forma similar a lo que hemos explicado para los

assets. La parte estática se ha construido, como se muestra en la Figura 67, usando etiquetas de texto, `a-text`, ofrecidas por el *framework*. También se puede apreciar que permitimos la interacción con el modelo 3D relacionado con el cuestionarios a través de dos acciones de rotación.

```

<a-gltf-model id="modelo-element" src="#modelo" position="-14.5 0 -5" scale="0.005 0.005 0.005"></a-gltf-model>
<a-image src="#next" id="next2" rotation="0 90 0" scale="0.5 0.5 0.5" position="-14 1 -9.5"></a-image>
<a-image src="#next" onclick="rotarModelo90()" rotation="0 90 0" scale="0.5 0.5 0.5" position="-13 0.5 -5.5"></a-image>
<a-image src="#next" onclick="rotarModelo89()" rotation="0 279 0" scale="0.5 0.5 0.5" position="-13 0.5 -5"></a-image>
<!--Preguntas-->
<a-entity position="-14.9 2.7 -7" rotation="0 90 0">
  <a-text color="black" value="Cuestionario"></a-text>
</a-entity>
<a-entity position="-14.9 2.2 -7" rotation="0 90 0" scale="0.75 0.75 0.75">
  <a-text id="pregunta" color="black" ></a-text>
</a-entity>
<a-entity position="-14.9 1.7 -7" rotation="0 90 0" scale="0.5 0.5 0.5">
  <a-text id="opa" color="black" ></a-text>
  <a-entity position="2 0 0" class="respuestaIndicador" geometry="primitive: circle; radius: 0.25" material="color: #CCC" visible="false" ></a-entity>
</a-entity>
<a-entity position="-14.9 1.2 -7" rotation="0 90 0" scale="0.5 0.5 0.5">
  <a-text id="opb" color="black" ></a-text>
  <a-entity position="2 0 0" class="respuestaIndicador" geometry="primitive: circle; radius: 0.25" material="color: #CCC" visible="false"></a-entity>
</a-entity>
<a-entity position="-14.9 0.7 -7" rotation="0 90 0" scale="0.5 0.5 0.5">
  <a-text id="opc" color="black" ></a-text>
  <a-entity position="2 0 0" class="respuestaIndicador" geometry="primitive: circle; radius: 0.25" material="color: #CCC" visible="false"></a-entity>
</a-entity>
<a-entity position="-14.9 0.2 -7" rotation="0 90 0" scale="0.5 0.5 0.5">
  <a-text id="opd" color="black" ></a-text>
  <a-entity position="2 0 0" class="respuestaIndicador" geometry="primitive: circle; radius: 0.25" material="color: #CCC" visible="false"></a-entity>
</a-entity>

```

Figura 67 Etiqueta para crear texto en 3D

Por último, también se utilizará JavaScript para cambiar dinámicamente el modelo y el cuestionario visible en el laboratorio. En este caso deberemos realizar una consulta al *backend* que nos devolverá la información del nuevo modelo en un array a cargar en el DOM (ver Figura 68).

```

// Obtén las referencias a los elementos <a-text> en el HTML
const examen = examenes;
let respuestaCorrectaGlobal='';
const preguntaElement = document.querySelector('#pregunta');
const modeloElement = document.querySelector('#modelo-element');
const opaElement = document.querySelector('#opa');
const opbElement = document.querySelector('#opb');
const opcElement = document.querySelector('#opc');
const opdElement = document.querySelector('#opd');
const cambiarModeloBtn = document.querySelector('#next2');
cambiarModeloBtn.addEventListener('click', CambiarModelo);

preguntaElement.setAttribute('value', examen[0].pregunta);
modeloElement.setAttribute('src', '#' + examenes[i].idhtml);
opaElement.setAttribute('value', examen[0].opa);
opbElement.setAttribute('value', examen[0].opb);
opcElement.setAttribute('value', examen[0].opc);
opdElement.setAttribute('value', examen[0].opd);
respuestaCorrectaGlobal=examen[0].opcionCorrecta;
function CambiarModelo()
{
  i++;
  if(i>=examenes.length)
  {
    i=0;
  }
  const allTicks = document.querySelectorAll('.respuestaIndicador');
  allTicks.forEach(tick => tick.setAttribute('visible', 'false'));
  preguntaElement.setAttribute('value', examenes[i].pregunta);
  modeloElement.setAttribute('src', '#' + examenes[i].idhtml);
  modeloElement.removeAttribute('gltf-model');
  respuestaCorrectaGlobal=examenes[i].opcionCorrecta;
  opaElement.setAttribute('value', examenes[i].opa);
  opbElement.setAttribute('value', examenes[i].opb);
  opcElement.setAttribute('value', examenes[i].opc);
  opdElement.setAttribute('value', examenes[i].opd);
}

```

Figura 68 Función para modificar el modelo del Laboratorio en tiempo real

- **Interacción con el cuestionario:** incluiremos la funcionalidad necesaria para que al hacer clic en las repuestas de un cuestionario se validen mostrando si la información es correcta o no. En la Figura 69 se aprecia como se trabaja con los eventos de `click` asociados a los diferentes elementos. Al hacer clic en una opción, comprobamos si es o no la correcta mostraríamos el círculo de color rojo o verde informando al usuario.

```

//Verificar pregunta
function validarRespuesta(respuestaSeleccionada)
{
  const tickElement = document.querySelector(`#${respuestaSeleccionada} ~ a-entity.respuestaIndicator`);
  if (tickElement)
  {
    if (respuestaSeleccionada === respuestaCorrectaGlobal) {
      tickElement.setAttribute('material', 'color: green');
    }
    else
    {
      tickElement.setAttribute('material', 'color: red');
    }
    tickElement.setAttribute('visible', 'true');
    const allTicks = document.querySelectorAll('.respuestaIndicator');
    allTicks.forEach(tick => tick.setAttribute('visible', 'false'));

    tickElement.setAttribute('visible', 'true');
  }
}

opaElement.addEventListener('click', function() {
  validarRespuesta('opa');
});

opbElement.addEventListener('click', function() {
  validarRespuesta('opb');
});

opcElement.addEventListener('click', function() {
  validarRespuesta('opc');
});

opdElement.addEventListener('click', function() {
  validarRespuesta('opd');
});

```

Figura 69 Función para validar la pregunta y ver si es correcta

Evento de clic también es útil en un entorno de RV. Para ello se ha habilitado la etiqueta `<a-cursor>` que aparecerá como un círculo en la parte central de la pantalla como se ve en la Figura 70.

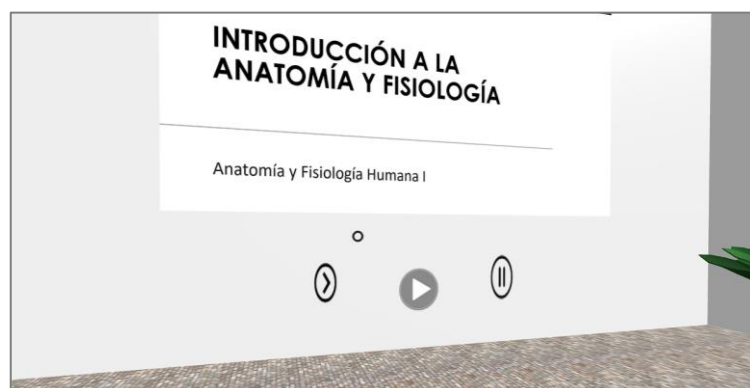


Figura 70 Cursor que actúa como puntero para poder interactuar con el entorno

### 3.7- Ejemplo de uso de la Aplicación

A continuación, se describirá un ejemplo de uso de la aplicación web desarrollada en este Trabajo Fin de Grado.

#### 3.7.1- Despliegue del proyecto

En primer lugar, y como se muestra en la Figura 71, se deberá iniciar el servicio MySQL a través del servidor XAMPP, sino la conexión con la base de datos daría error.

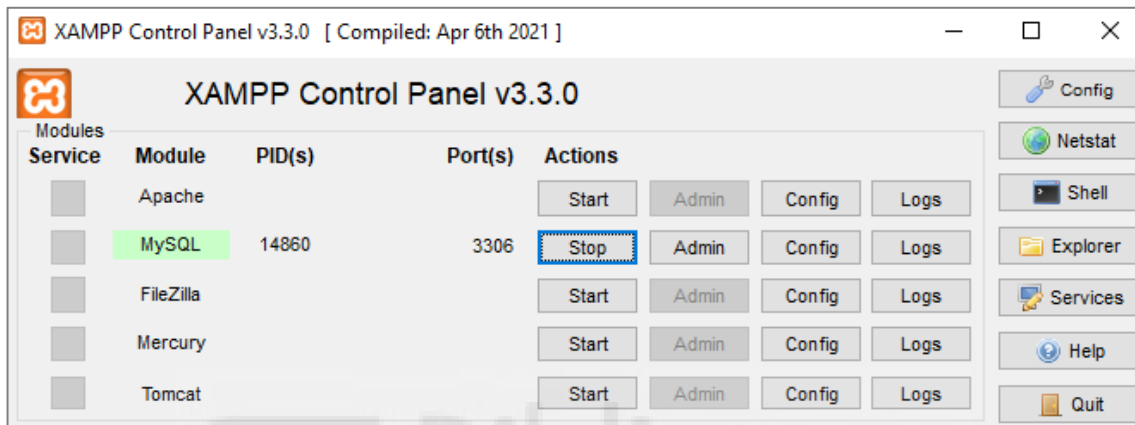


Figura 71 XAMPP

Una vez iniciada la base de datos, se deberá iniciar el servidor Node.js. Para ello, accederemos a la carpeta del proyecto y se arrancará usando el comando `start` como se puede ver en la Figura 72.

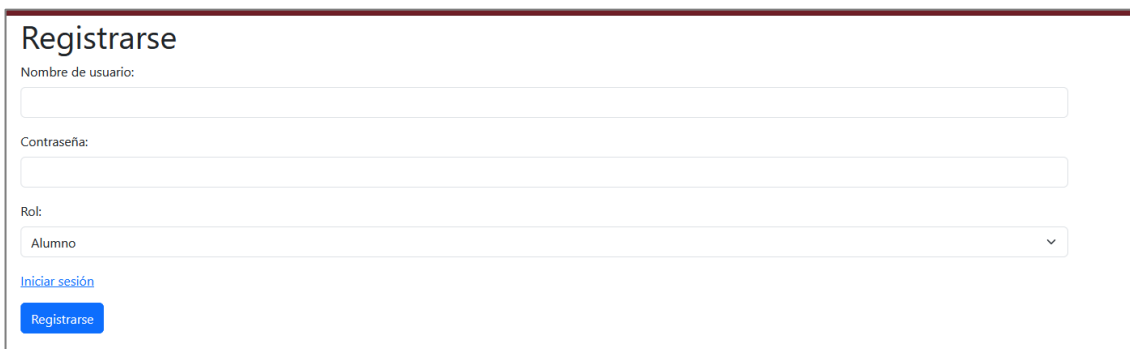
```
C:\xampp\htdocs\prueba\a-blast-master\examples\assets>npm start
> tfg@1.0.0 start
> node ./server/easyrtc-server.js

Aplicacion corriendo puerto 8080
Connected!
```

Figura 72 Comprobación que se inicia correctamente

### 3.7.2- Registro de usuario e inicio de sesión

Para poder acceder a la aplicación, los usuarios deberán registrarse. En la Figura 73 se muestra el formulario de registro en el que se podrá especificar el rol del usuario registrado, en un futuro el proceso de registro se deberá de mejorar como la gestión y asignación de roles.



The screenshot shows a registration form with the following fields and elements:

- Title: Registrarse
- Nombre de usuario: Input field
- Contraseña: Input field
- Rol: Dropdown menu with 'Alumno' selected
- Links: [Iniciar sesión](#) and [Registrarse](#) (highlighted in blue)

Figura 73 Registro de usuarios

Una vez registrados, los usuarios deberán autenticarse a través del formulario de inicio de sesión mostrado en la Figura 74.



The screenshot shows a login form with the following fields and elements:

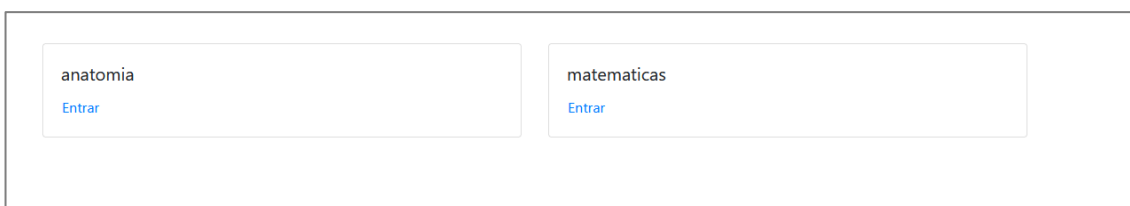
- Title: Iniciar sesión
- Nombre de usuario: Input field
- Contraseña: Input field
- Rol: Dropdown menu with 'Alumno' selected
- Links: [Registrarse](#) and [Iniciar sesión](#) (highlighted in blue)

A watermark for 'Biblioteca UNIVERSIDAD Miguel Hernández' is visible in the background.

Figura 74 Formulario de inicio de sesión

### 3.7.3- Acceso al Metaverso

Una vez autenticado el usuario, tendrá acceso a un listado de asignaturas con las que estará relacionado. Por ejemplo, como se muestra en la Figura 75, a los alumnos les aparecerá un listado de las asignaturas en las que está matriculado y enlaces que le permitirá acceder los Metaversos asociados con ellas.



The screenshot shows a menu with two course options, each with an 'Entrar' link:

- anatomia [Entrar](#)
- matematicas [Entrar](#)

Figura 75 Menú donde se ven los Metaversos matriculados



Al hacer clic en uno de estos enlaces, el usuario será dirigido al entorno 3D del Metaverso. En la Figura 76 se muestra al usuario inmerso en el Mundo Virtual y desde el que podrá acceder a las distintas salas que encontrará en él.



Figura 76 Zona de aparición

Merece la pena destacar que, la captura de la Figura 76 se ha hecho accediendo al Metaverso a través de un navegador web de escritorio. En la parte inferior izquierda de dicha imagen podemos observar que se ha habilitado por defecto el audio de proximidad. Este botón ofrece la posibilidad de silenciarnos para que el resto de los usuarios no nos escuchen.

Además, en la esquina inferior izquierda se habilitará el botón "VR" cuando el navegador desde el que accedemos sea compatible con esta tecnología. Al pinchar en este botón la inmersión en el entorno usando RV será mayor. En la Figura 77 hacemos uso del simulador Immersive Web Emulator para acceder a nuestro Metaversos con RV simulando unas Quest 2.

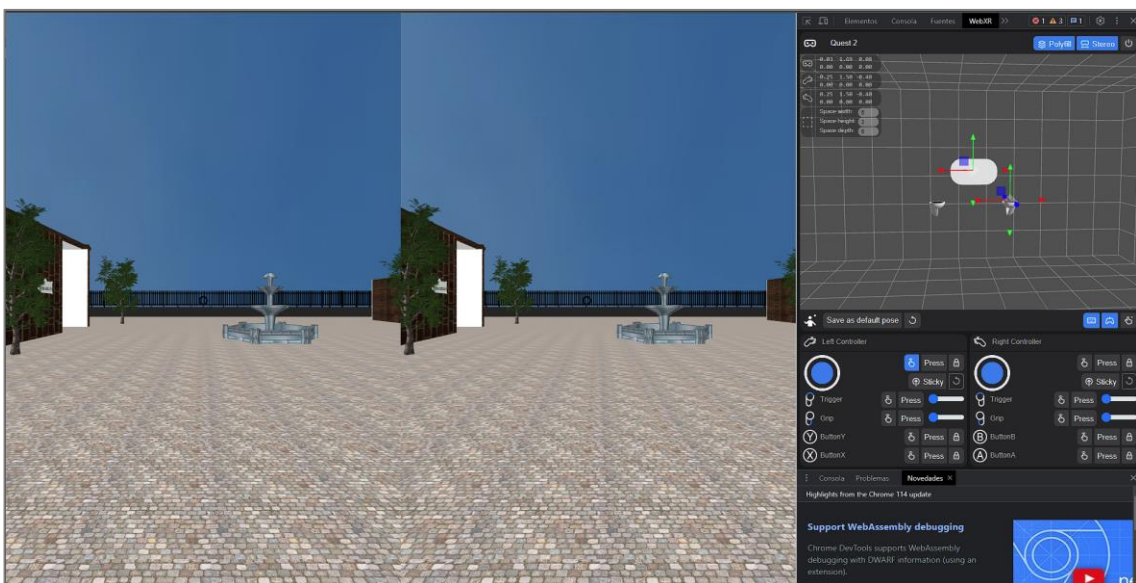


Figura 77 Así se vería si pulsamos el botón VR con la extensión Immersive Web Emulator

### 3.7.4- Interacción entre usuarios conectados simultáneamente

Ahora vamos a mostrar qué ocurriría si, además del usuario anterior, otro nuevo usuario conecta en el mismo instante de tiempo al mismo Metaverso. En la Figura 78 se observa como ambos usuarios son representados por distintos avatares, y en caso de que tengan habilitados sus micrófonos la interacción entre ellos a través de la voz sería en tiempo real.

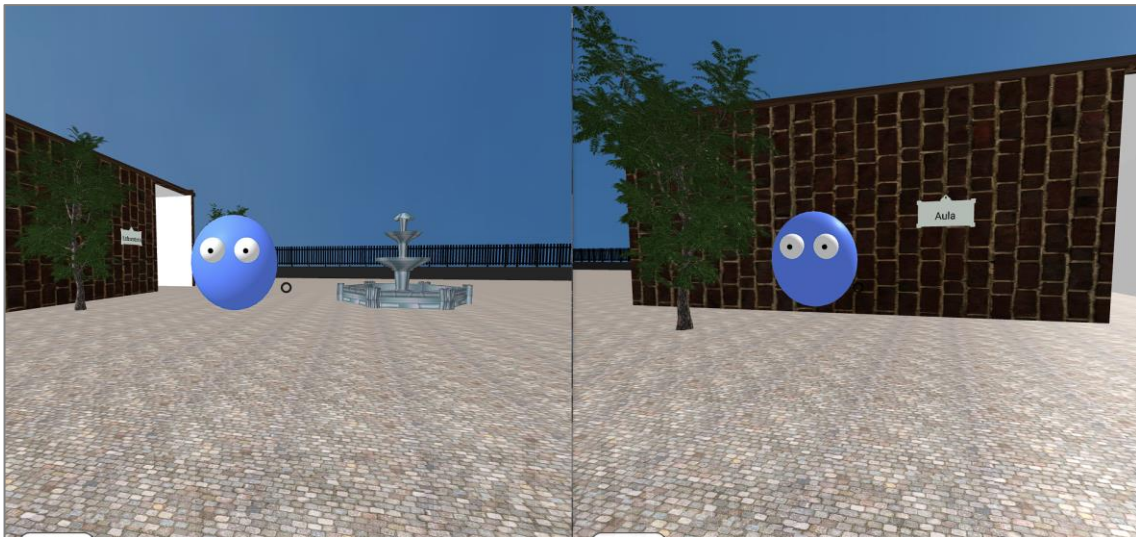


Figura 78 Uso de la NAF para que haya varios usuarios

En la Figura 79, podemos ver como el primer usuario tiene el micrófono muteado y el segundo no, con lo cual el segundo no podría escuchar lo que dice el primer usuario.

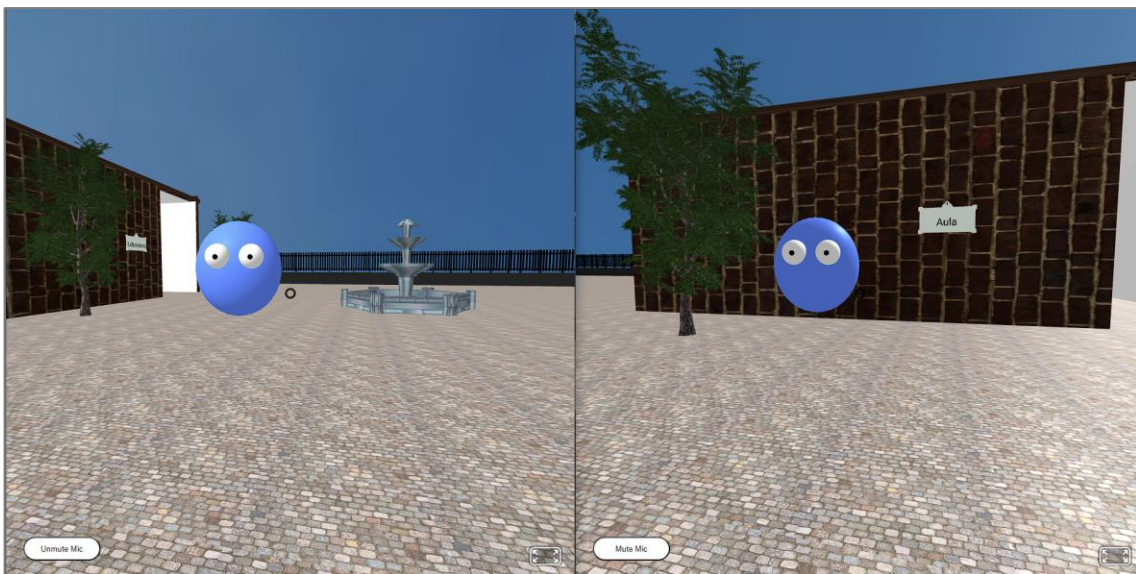


Figura 79 Se puede apreciar como el primer usuario tiene el micrófono muteado y el segundo no

### 3.7.5- Explorando las distintas salas: vitanto el Aula

Una vez conectados, el usuario podrá navegar libremente por el Metaverso. Según el dispositivo desde el que se está realizando la conexión se realizará el movimiento utilizando distintos controles. Por ejemplo, si se ha conectado desde un navegador de escritorio se deberá utilizar el teclado para desplazarse. Por otro lado, si la conexión se ha realizado desde un dispositivo de RV A-Frame permitirá que el movimiento se realice utilizando los mandos o controles propios de cada tipo de dispositivos.

Dado que el Metaverso se componen de tres tipos de espacios, a la entrada de cada uno de ellos se incluirá información sobre la sala como se observa en la Figura 80.



Figura 80 Modelo del aula

Al entrar en un Aula, el usuario se encontrará inmerso en un ambiente como el que se muestra en la Figura 81. Allí tendrá la posibilidad de interactuar con los vídeos subidos por el profesor a través del “proyector” que estará representado en una de las paredes de la clase.



Figura 81 Vista del aula por dentro junto al vídeo y sus botones

### 3.7.6- Visitando el Laboratorio

A continuación, se muestra el Laboratorio donde se interactuará con los modelos 3D y sus cuestionarios asociados. En la Figura 82 se muestra una imagen del usuario colocado justo en la puerta de este.



Figura 82 Modelo del laboratorio

En la Figura 83 se muestra cómo se vería por dentro el Laboratorio. En este caso, podemos ver que hay cargado un modelo anatómico con dos botones que permitirá girarlo 90° tanto a izquierda como a derecha. Además, junto al modelo 3D se puede ver el cuestionario con el que el usuario puede interactuar. En este caso, el usuario ha hecho clic sobre la primera respuesta que es la correcta y se indica con un círculo verde. Por último, en la parte derecha del cuestionario se aprecia un botón "Siguiete". Si el usuario hace clic sobre él se actualizará en tiempo real el laboratorio con el siguiente modelo y cuestionario asociado.

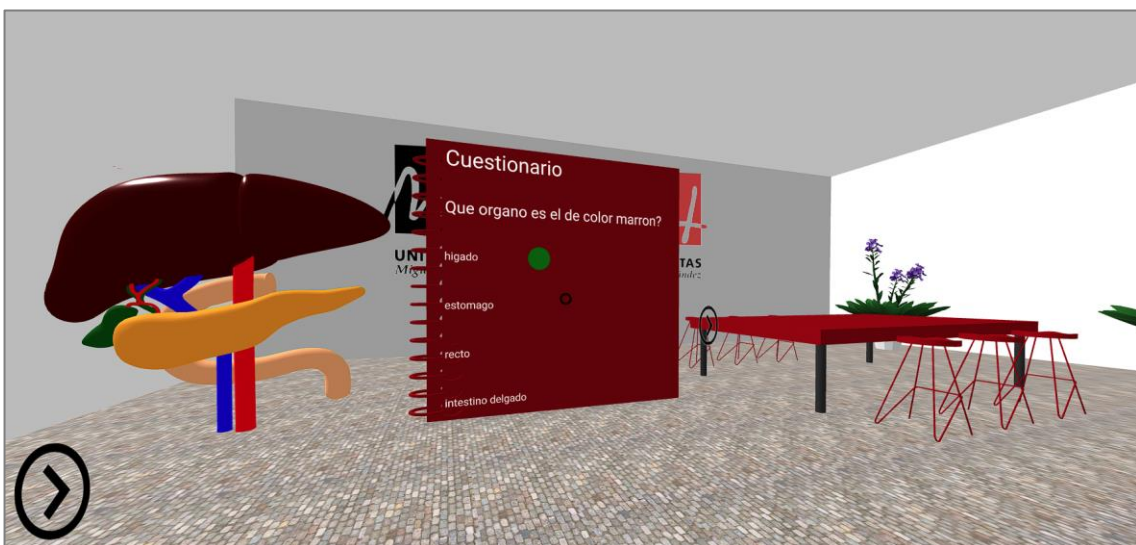


Figura 83 Laboratorio por dentro donde se ve el modelo y las preguntas junto a la respuesta correcta

### 3.7.7- Visitando la Sala de Reuniones

El último edificio que compone nuestro Metaverso y que presentamos en la Figura 84 es la Sala de Reuniones. En la Figura 85 se muestra cómo sería su interior. Esta sala no contiene elementos interactivos. Su propósito es servir de zona común para que los usuarios se reúnan y conversen usando el audio de proximidad en un entorno “cerrado”.



Figura 84 Modelo de la sala de reuniones

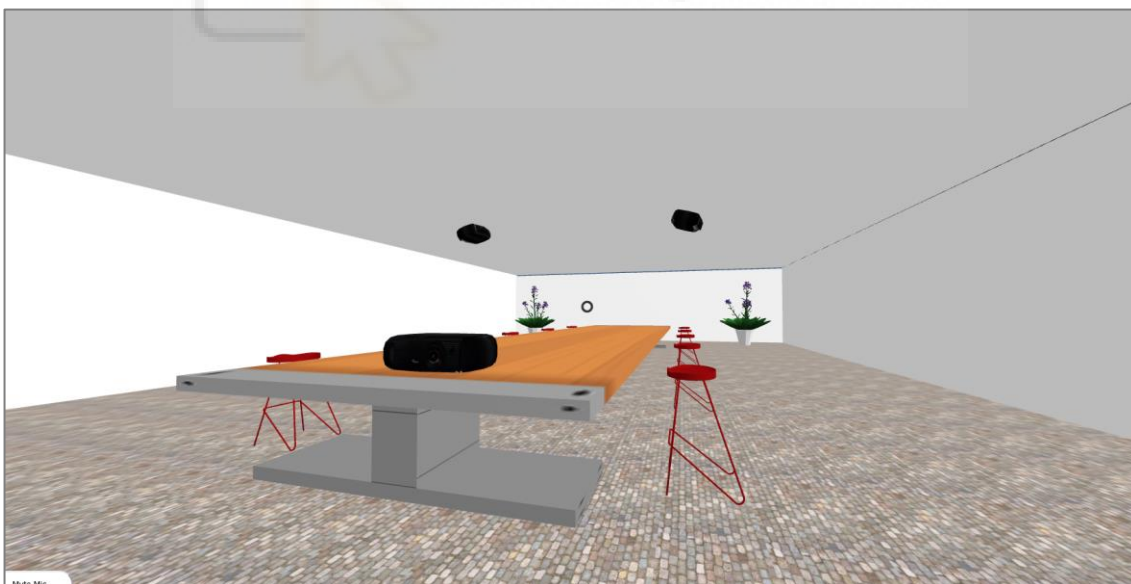


Figura 85 Sala de reuniones por dentro

Una vez explicadas las características de Metaversos, continuamos con las opciones de configuración de los metaversos, que serán accesibles solo por los profesores a través de la interfaz.

### 3.7.8- Gestión recursos: vídeos

En la Figura 86, se puede observar que la vista de los profesores contiene funcionalidad adicional de la que pueden acceder los alumnos. Concretamente, en esta imagen se muestran el listado de asignaturas/salas relacionadas con un profesor.



Figura 86 Listado de los Metaversos con sus opciones correspondientes

Por ejemplo, el profesor podrá modificar los vídeos que serán visibles en el aula. Para ello utilizará la vista que se muestra en la Figura 87 donde aparece un listado con todos los vídeos disponibles. Junto a ellos tendrá la opción de modificar, borrar o añadir un nuevo recurso.

#### Videos de la sala

ID	Nombre	Ruta		
1	anatomia	assets/anatomia.mp4	Borrar	Modificar
2	anatomia2	assets/anatomia2.mp4	Borrar	Modificar
9	anatomia3	/assets/anatomia3.mp4	Borrar	Modificar

Salas   Añadir video

Figura 87 Listado de los videos

En la Figura 88 se muestra el formulario para añadir vídeos. Este formulario permitirá al usuario subir el vídeo y darle un nombre. Al pinchar en aceptar la información del nuevo vídeo será registrada en la BBDD y el fichero del vídeo será copiado a la capeta de recursos de la aplicación web y, a partir de este momento, será accesible desde el Metaverso a través de su URL.

Formulario para añadir un vídeo:

Ingresa el nombre del video

Selecciona un archivo de video

Examinar... No se ha seleccionado ningún archivo.

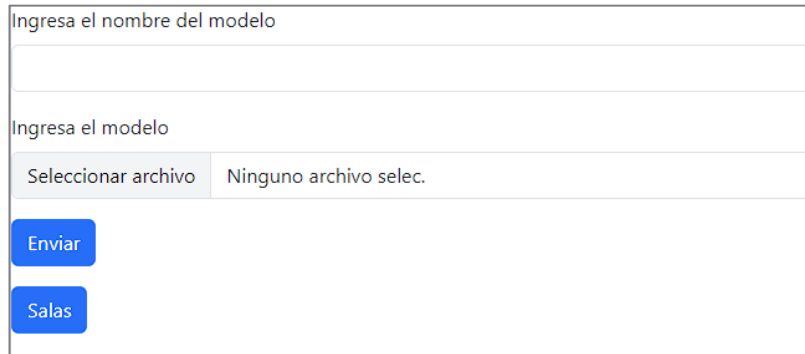
Añadir

Salas

Figura 88 Formulario para añadir un vídeo

### 3.7.9- Gestión recursos: modelos del laboratorio y cuestionario

En la Figura 89 se muestra el formulario para añadir un modelo que será visible en el Laboratorio. De forma similar a los vídeos, el profesor deberá subir el modelo y darle un nombre. Para almacenarlo se seguirá también un flujo similar al de los vídeos.

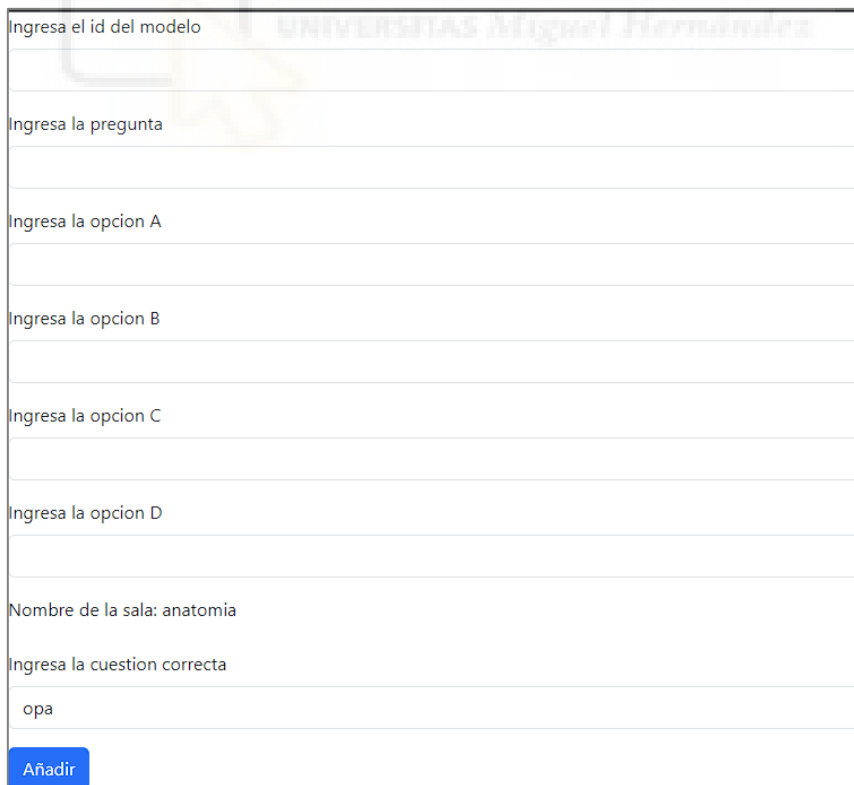


Formulario para añadir un modelo. El formulario contiene los siguientes elementos:

- Un campo de texto con el placeholder "Ingresa el nombre del modelo".
- Un campo de texto con el placeholder "Ingresa el modelo".
- Un botón "Seleccionar archivo" que abre un selector de archivos.
- Un texto "Ninguno archivo selec." que indica el estado del archivo seleccionado.
- Un botón "Enviar" de color azul.
- Un botón "Salas" de color azul.

Figura 89 Formulario para añadir un modelo

Además, los profesores podrán definir nuevos cuestionarios. Por ejemplo, en la Figura 90 se muestra el formulario para añadir un cuestionario. Este formulario permitirá definir y almacenar sus diferentes preguntas y la respuesta válida. Esta información se guardará en la BBDD será cargada dinámicamente desde la vista del Metaverso.



Formulario para añadir un cuestionario. El formulario contiene los siguientes elementos:

- Un campo de texto con el placeholder "Ingresa el id del modelo".
- Un campo de texto con el placeholder "Ingresa la pregunta".
- Un campo de texto con el placeholder "Ingresa la opcion A".
- Un campo de texto con el placeholder "Ingresa la opcion B".
- Un campo de texto con el placeholder "Ingresa la opcion C".
- Un campo de texto con el placeholder "Ingresa la opcion D".
- Un campo de texto con el placeholder "Nombre de la sala: anatomia".
- Un campo de texto con el placeholder "Ingresa la cuestion correcta".
- Un campo de texto con el placeholder "opa".
- Un botón "Añadir" de color azul.

Figura 90 Formulario para añadir un cuestionario

Al igual con el resto de los recursos educativos, los profesores podrán administrar los cuestionarios ya creados. Para ello se utilizará la vista que se muestra en la Figura 91.

Cuestionarios de la sala							
ID del Modelo	Pregunta	Opcion A	Opcion B	Opcion C	Opcion D	Opcion Correcta	Sala
anatomia	¿Que organo es el de color marron?	higado	estomago	recto	intestino delgado	opa	anatomia
anatomia	¿Que organo es el de color amarillo?	higado	intestino delgado	intestino grueso	Estomago	opd	anatomia
anatomia	¿Que organo es el de color verde?	Pancreas	No se	Vesicula biliar	Intestino gruego	opc	anatomia
anatomia	¿Que es lo que esta en color rojo?	Aorta abdominal	Vena roja	Vena azul	Intestino	opa	anatomia

[Salas](#)
[Añadir cuestionario](#)

Figura 91 Listado de los cuestionarios





## 4-Conclusiones y trabajo futuro

El objetivo principal de este proyecto fue crear una Aplicación Web que integre tecnología de RV para generar un entorno inmersivo que pueda ser usado como Metaverso Docente. Una vez finalizado este TFG, podemos afirmar que se ha logrado el objetivo. Para ello, hemos creado un prototipo funcional de una aplicación web que integra tecnologías de RV que permiten a sus usuarios (alumnos y profesores) el acceso a un entorno inmersivo 3D y compartido, un Metaverso, que podrá ser recorrido libremente, y desde el que se podrá acceder a diferentes salas (Aula, Laboratorio y Sala de Reuniones). Además, dentro de cada una de estas salas se podrá interactuar con los propios usuarios a través de audio de proximidad, y con diferentes tipos de recursos educativos (vídeos, modelos 3D y cuestionarios).

Desde el punto de vista técnico, el estudio del estado del arte supuso un reto ya que nunca habíamos trabajado con este tipo de tecnologías en entornos web. Aunque el objetivo inicial fue democratizar el uso de la Realidad Virtual haciendo una aplicación compatible con dispositivos móviles a través de Google CardBoard. El análisis del estado del arte reveló ciertas dificultades de compatibilidad de esta tecnología desde la aparición de WebXR (sustituto natural de WebVR). La elección de WebXR redirigió nuestro camino. El priorizar la elección de la tecnología WebXR, en lugar de la obsoleta WebVR, centramos nuestro desarrollo en su compatibilidad con las implementaciones más actuales de navegadores web lo que permitirá que nuestra solución sea accesible desde diferentes tipos de dispositivos como navegadores de escritorio (sin habilitar la RV) o desde navegadores de dispositivos de RV como Meta Quest 2.

A partir de la elección anterior, establecimos un marco de trabajo basado en tecnologías web usando el *framework* A-Frame y su extensión Networked A-Frame (NAF) para construir la parte 3D de los Metaversos docentes. El uso de estas herramientas ha requerido el despliegue del proyecto usando Node.js. Para la gestión de datos usaremos una BBDD MySQL a través de XAMPP. Además de la interfaz 3D de los Metaversos, se ha implementado una parte de la herramienta usando tecnologías web tradicionales como HTML5, CSS, JS y el *framework* Bootstrap. El uso de Node.js, ha exigido tener que trabajar con el formato JSON para serializar los datos que son transferidos entre nuestro *backend* y nuestro *frontend*. Por último, para la construcción del Metaverso 3D, hemos tenido que utilizar Blender para editar y manipular sus modelos 3D.

Desde el inicio, nos enfrentamos a un proyecto abierto con requisitos cambiantes. Por ello, para la gestión de este se ha seguido la metodología ágil de desarrollo del software basada en Scrum. Esto nos ha permitido iterar sobre los requisitos y avanzar a partir de pequeños hitos marcados en los *sprints*, con un seguimiento y control semanal. También se han usado herramientas que han facilitado la gestión del código como Bitbucket para el control de versiones o el uso de editores como Visual Studio Code.

Creemos que el uso de este tipo de tecnología puede ser útil para facilitar el acceso a un entorno educativo común evitando el desplazamiento físico requerido en un mundo real. Además, en este tipo de entornos se pueden integrar simuladores y otros recursos, que podría ser difícil de almacenar y mantener en una escuela física. Por ejemplo, la interacción con modelos 3D de anatomía requeriría tener que desplazarse al lugar físico donde se encuentra el modelo, y esta limitación no existen en un entorno virtual.

Con este proyecto pretendemos abrir una línea de trabajo centrada en el desarrollo de aplicaciones de RV usando Tecnologías Web. Aunque hemos construido un primer prototipo funciona esperamos que, en un futuro, este siga avanzando y evolucionando. Esperamos que su crecimiento permita explorar nuevas dimensiones que contribuyan a analizar el impacto de estas tecnologías en la educación, en línea con los retos detectados en [31].

## 4.1 Trabajo futuro

A continuación, se presentan algunas líneas de trabajo futuro que surgen una vez finalizado este trabajo:

- **Evaluar el impacto de este tipo de soluciones en la educación:** explorar nuevos tipos de recursos educativos e interacciones que permitan llevar la creatividad de alumnos y profesores a nuevos límites.
- **Incluir IA:** implementar Bots o NPC's que puedan simular personas que realicen tareas concretas dentro del Metaverso. Por ejemplo, un Bot podría realizar la recepción de los alumnos y guiarles en el proceso de uso de las distintas salas y recursos.
- **Mejora de la BBDD:** la BBDD se podría ampliar y optimizar para que las consultas y la cargas sean más fluidas y mejorar los tiempos de respuesta por parte de los usuarios.
- **Mejora de Modelos:** modelos o interacciones de mejor calidad para que los usuarios se sientan más cómodos e integrados con el entorno. Esto requeriría el desarrollo de animaciones que doten a los modelos de mayor dinamismo.
- **Inclusión de avatares:** permitir a los usuarios personalizar su avatar para que no todos sean iguales, así se podría diferenciar un alumno de un profesor y viceversa.
- **Incluir tecnología Blockchain:** sería interesante explorar las posibilidades de esta tecnología en diferentes partes del Metaverso. Por ejemplo, se podría relacionar los avatares con complementos que usen NFT para dotarlos de valor.
- **Mejorar la herramienta de gestión:** se podría mejorar la herramienta de gestión para que quede mejor integrada en un contexto educativo real.

## 5. Bibliografía

- [1] EuroInnova, «Beneficios de la educación». <https://www.euroinnova.edu.es/blog/beneficios-de-la-educacion>
- [2] Comunidad UIC, «Apoyo a la educación, por una mejor sociedad». <https://www.uic.mx/apoyo-a-la-educacion-por-una-mejor-sociedad/>
- [3] Vicmix, «Metaverso y educación: posibilidades en el futuro de la Realidad Extendida». <https://vicmixreality.com/2021/12/15/metaverso-educacion-posibilidades-en-el-futuro-de-la-realidad-extendida/>
- [4] Jorge Calvo Martín, «Beneficios de la Realidad Virtual en Educación. Cómo llevarla al aula.» <https://gaptain.com/blog/beneficios-de-la-realidad-virtual-en-educacion-como-llevarla-al-aula/>.
- [5] Arimetrics, «Qué es Metaverso». <https://www.arimetrics.com/glosario-digital/metaverso>
- [6] Rodrigo Álvarez, «Metaverso: seis maneras en las que el mundo virtual nos va a cambiar la vida». <https://tn.com.ar/tecno/novedades/2022/07/31/metaverso-6-maneras-en-las-que-el-mundo-virtual-nos-va-a-cambiar-la-vida/>.
- [7] Snow crash: A novel, *Snow crash: A nove*. Spectra, 2003.
- [8] Meta, «The Metaverse and How We'll Build It Together -- Connect 2021». <https://www.youtube.com/watch?v=>
- [9] Meta, «Meta Connect Keynote 2022». <https://www.youtube.com/watch?v=hvfV-iGwYX8>
- [10] Esther G.Wood, «Ready Player One: el Spielberg más palomitero». <https://www.televisando.com/ready-player-one-spielberg-herbert-cline>
- [11] Netflix, «Black Mirror». <https://www.netflix.com/es/title/70264888>
- [12] Grupo Rirca, «Black Mirror y “Playtest”; La (¿ansiada?) llegada de la “verdadera” realidad virtual». <https://www.rirca.es/black-mirror-y-playtest-la-ansiada-llegada-de-la-verdadera-realidad-virtual/>.
- [13] Kasey Moore, «Black Mirror “Playtest”: Episode Guide, Cast & Theories». <https://www.whats-on-netflix.com/news/black-mirror-playtest-episode-guide-cast-theories/>
- [14] Cultura Ocio, «Black Mirror 6x01: La historia real detrás de Joan es horrible». <https://www.culturaocio.com/tv/noticia-black-mirror-6x01-historia-real-detras-joan-horrible-20230626101219.html>

- [15] Henrique Cesar Mello, «Trailer da sexta temporada de Black Mirror tira sarro com a própria Netflix; veja... - Leia mais em [https://caras.uol.com.br/tv/trailer-da-sexta-temporada-de-black-mirror-tira-sarro-com-propria-netflix-veja.shtml?utm\\_source=site&utm\\_medium=txt&utm\\_campaign=copypaste](https://caras.uol.com.br/tv/trailer-da-sexta-temporada-de-black-mirror-tira-sarro-com-propria-netflix-veja.shtml?utm_source=site&utm_medium=txt&utm_campaign=copypaste)».
- [16] T. J. , L. Y. J. Lin, *Language Learning in Virtual Reality Environments: Past, Present, and Future*. 2015.
- [17] E. E. M. C. A. G. J. D. Anacona, «Aplicación de los metaversos y la realidad virtual en la enseñanza».
- [18] K. LIM, *The six learnings of Second Life: A framework for designing curricular interventions in-world*. 2009.
- [19] Pearson, «El metaverso en la educación superior: ¿una realidad próxima?»
- [20] Decentraland, «Decentraland».
- [21] Observatorio Blockchain, «¿Qué es y cómo entrar en el metaverso Decentraland por primera vez?»
- [22] Virtway events, «Virtway events».
- [23] Diego J. Geniz, «El metaverso llega a las aulas de Sevilla: Realidad virtual para la educación», 2022.
- [24] Xperimentacultura, «Historia de la Realidad Virtual».
- [25] Wikipedia, «Virtual Boy».
- [26] Meta, «Oculus».
- [27] Memésis, «VRChat, el juego de realidad virtual donde casi todo vale». [https://www.vozpopuli.com/especiales/vrchat-realidadvirtual-acoso\\_0\\_1411059266.html](https://www.vozpopuli.com/especiales/vrchat-realidadvirtual-acoso_0_1411059266.html)
- [28] Gobierno de Canarias, «Anatomyou». <https://www3.gobiernodecanarias.org/medusa/ecoescuela/recursosdigitales/2020/05/08/anatomyou/>
- [29] Realovirtual, «Aprendiendo idiomas en VR con Mondly: ANÁLISIS». <https://www.realovirtual.com/articulos/5913/aprendiendo-idiomas-vr-mondly-analisis>
- [30] Android Latino, «Unimersiv lanza las experiencias educativas VR en Daydream». <https://androidlatino.co/unimersiv-experiencias-educativas-vr-daydream/25362/>
- [31] Israel Marquez, *Metaversos y educación: Second Life como plataforma educativa*. 2011.

- [32] Linden Research Inc., «Second Life». <https://secondlife.com/>
- [33] 360 schools, «Inmersive 360º». <https://schools.360cities.net/>
- [34] Educa 360, «Unete al metaverso educativo». <https://educa360.com/>
- [35] 360 schools, «About 360schools».
- [36] P. Martínez Botella, «Desarrollo de una aplicación para simulación de deficiencia visual mediante realidad virtual», 2022.  
<http://dspace.umh.es/handle/11000/28614>
- [37] J. Rodríguez Ruiz, «Desarrollo de una aplicación de realidad virtual para la simulación de daltonismo », 2022.  
<http://dspace.umh.es/handle/11000/29167>
- [38] Javier Salinas, «¿Qué es la WebVR?» <https://javiersalinas.es/que-es-la-webvr/>
- [39] Wikipedia, «WebVR». <https://es.wikipedia.org/wiki/WebVR>
- [40] Onirix, «Web XR: Explorando la Realidad Extendida en la Web». <https://www.onirix.com/es/webxr-ejemplos-desarrollo-realidad-extendida>
- [41] Wikipedia, «Google Cardboard». [https://es.wikipedia.org/wiki/Google\\_Cardboard](https://es.wikipedia.org/wiki/Google_Cardboard)
- [42] Paul Sawers, «The New York Times to launch slate of VR films, will give out 1M+ Google Cardboard viewers for free». <https://venturebeat.com/media/the-new-york-times-to-launch-slate-of-vr-films-will-give-out-1m-google-cardboard-viewers-for-free/>
- [43] Fyrd, «WebXR Device API». <https://caniuse.com/webxr>
- [44] NAVER FE Platform, «Some Tips for the Web VR Beginners», 2020.  
<https://medium.com/naver-fe-platform/some-tips-for-the-web-vr-beginners-ee12002f36bf>
- [45] Wikipedia, «Three.js». <https://es.wikipedia.org/wiki/Three.js>
- [46] Hitesh Sahu, «Three.js vs Babylon.js», 26 de septiembre de 2019.
- [47] Matthew Jackson, «VR with AFrame vs Three.js vs Babylon.js», 2016.  
<http://www.stellarbuild.com/blog/article/vr-with-aframe-vs-threejs-vs-babylonjs>
- [48] Vicentfretin, «Getting started with Networked-Aframe». Getting started with Networked-Aframe
- [49] Wikipedia, «NodeJS». <https://es.wikipedia.org/wiki/Node.js>
- [50] Meta, «Immersive Web Emulator». <https://chrome.google.com/webstore/detail/immersive-web-emulator/l/related>

- [51] WebXR, «Getting started with WebXR Augmented Reality using Babylon.js», 2021. [https://www.youtube.com/watch?v=RaB\\_Uyqx6Q4](https://www.youtube.com/watch?v=RaB_Uyqx6Q4)
- [52] Nik Lever, «Creating an AR experience using Three.js».
- [53] A-Frame, «A-Frame v1.0.0 - WebXR Support, AR Mode». <https://aframe.io/blog/aframe-v1.0.0/>
- [54] Alicia M., «Sketchfab, la plataforma más grande para visualizar y descargar modelos 3D». <https://www.3dnatives.com/es/sketchfab-visualizar-y-descargar-modelos-3d-230320202/>
- [55] Sketchfab, «3D Model repository». <https://sketchfab.com/feed>
- [56] Enzo Marchionni, «3dWarehouse Diseño de muebles en melamina 3d y otros materiales».
- [57] Trimble, «Repositorio de modelos (3dWarehouse)». <https://3dwarehouse.sketchup.com>
- [58] Zac Lucarelli, «Poly Pizza Press Kit», [En línea]. Disponible en: <https://poly.pizza/docs/press>
- [59] Zac Lucarelli, «Poly Pizza». <https://poly.pizza/>
- [60] Wikipedia, «XAMPP».
- [61] Wikipedia, «MySQL». <https://es.wikipedia.org/wiki/MySQL>
- [62] MDN contributors, «HTML: Lenguaje de etiquetas de hipertexto». <https://developer.mozilla.org/es/docs/Web/HTML>
- [63] Wikipedia, «CSS». <https://es.wikipedia.org/wiki/CSS>
- [64] Wikipedia, «Bootstrap». [https://es.wikipedia.org/wiki/Bootstrap\\_\(framework\)](https://es.wikipedia.org/wiki/Bootstrap_(framework))
- [65] Wikipedia, «phpMyAdmin». <https://es.wikipedia.org/wiki/PhpMyAdmin>
- [66] Wikipedia, «Visual Studio Code». [https://es.wikipedia.org/wiki/Visual\\_Studio\\_Code](https://es.wikipedia.org/wiki/Visual_Studio_Code)
- [67] Wikipedia, «Bitbucket». <https://es.wikipedia.org/wiki/Bitbucket>
- [68] Wikipedia, «Scrum (desarrollo de software)». [https://es.wikipedia.org/wiki/Scrum\\_\(desarrollo\\_de\\_software\)](https://es.wikipedia.org/wiki/Scrum_(desarrollo_de_software))
- [69] Hanandess, «Cedro Andino». <https://sketchfab.com/3d-models/cedro-andino-010430034fcc4602a89d3d7069c7c9f1>
- [70] DaniDBM, «Valla». <https://sketchfab.com/3d-models/cedro-andino-010430034fcc4602a89d3d7069c7c9f1>
- [71] OpenJS Foundation, «Node.js».
- [72] Apache Friends, «Download XAMPP».

[73] A-Frame, «gltf-model ». <https://aframe.io/docs/1.4.0/components/gltf-model.html>



## ANEXO I: Casos de Usos

En las siguientes tablas se incluyen los casos de uso desarrollados y descritos en el capítulo de Materiales y Métodos.

<b>CU-01</b>	Iniciar sesión
<b>Actores</b>	Profesor y alumno
<b>Descripción</b>	Los usuarios iniciarán sesión en la página web
<b>Precondición</b>	Tener internet
<b>Secuencia</b>	1.El usuario accede a la página web. 2.El usuario introduce los datos para poder registrarse. 3.Ok
<b>Postcondición</b>	Si en el punto 1 no tiene internet, no podrá acceder a la página. Si en el punto 2, se falla el registro saldrá un mensaje que le indicara el tipo de error.

<b>CU-02</b>	Cerrar sesión
<b>Actores</b>	Profesor y alumno
<b>Descripción</b>	Los usuarios cerraran sesión en la página web
<b>Precondición</b>	Tener internet y estar logueado
<b>Secuencia</b>	1.El usuario pulsara el botón de cerrar sesión. 2.Ok
<b>Postcondición</b>	Si en el punto 1 no está logueado no podrá acceder al botón correspondiente.

<b>CU-03</b>	Registro
<b>Actores</b>	Profesor
<b>Descripción</b>	El profesor registrara a los usuarios para que puedan entrar.
<b>Precondición</b>	Tener internet
<b>Secuencia</b>	1.El usuario accede a la página web. 2.El usuario introduce los datos para poder registrarse. 3.Ok
<b>Postcondición</b>	Si en el punto 1 no tiene internet, no podrá acceder a la página. Si en el punto 2, se falla el registro saldrá un mensaje que le indicara el tipo de error.



<b>CU-04</b>	Gestión de modelos (A/B/M)
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Un menú donde se podrá acceder a la gestión de modelos (borrar,eliminar y añadir)
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de modelos. 3.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página.

<b>CU-05</b>	Alta de modelos
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Un formulario donde podrá rellenar los datos del modelo y así subirlos a la base de datos.
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de modelos. 3.Pulsar el botón de alta de modelo. 4.Rellenar los datos correctamente. 5.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página. Si en el punto 4 no introduce correctamente los datos, le saldrá un error.

<b>CU-06</b>	Baja de modelos
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Eliminará de la base de datos los modelos.
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de modelos. 3.Pulsar el botón de eliminar. 4.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página.

<b>CU-07</b>	Modificación de modelos
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Se podrá editar el modelo correspondiente
<b>Precondición</b>	Tener internet y cuenta

<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de modelos. 3.Pulsar el botón de editar. 4.Rellenar el formulario correctamente. 5.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página. Si en el punto 4 no se rellena correctamente el formulario dará error.

<b>CU-08</b>	<b>Gestión de vídeos (A/B/M)</b>
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Un menú donde se podrá acceder a la gestión de vídeos (borrar, eliminar y añadir)
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de vídeos. 3.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página.

<b>CU-09</b>	<b>Alta de vídeos</b>
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Un formulario donde podrá rellanar los datos del video y así subirlos a la base de datos.
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de vídeos. 3.Pulsar el botón de alta de vídeo. 4.Rellenar los datos correctamente. 5.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página. Si en el punto 4 no introduce correctamente los datos, le saldrá un error.

<b>CU-10</b>	<b>Baja de vídeos</b>
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Eliminará de la base de datos el vídeo

<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de vídeos. 3.Pulsar el botón de eliminar. 4.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página.

<b>CU-11</b>	Modificación de vídeos
<b>Actores</b>	Profesor
<b>Descripción</b>	Se podrá editar el vídeo correspondiente
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de vídeos. 3.Pulsar el botón de editar. 4.Rellenar el formulario correctamente. 5.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página. Si en el punto 4 no se rellena correctamente el formulario dará error.

<b>CU-12</b>	Gestión de cuestionarios (A/B/M)
<b>Actores</b>	Profesor
<b>Descripción</b>	Un menú donde se podrá acceder a la gestión de cuestionarios (borrar, eliminar y añadir)
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de cuestionarios. 3.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página.

<b>CU-13</b>	Alta de cuestionario
<b>Actores</b>	Profesor
<b>Descripción</b>	Un formulario donde podrá rellanar los datos del cuestionario y así subirlos a la base de datos.
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de cuestionarios.

	3.Pulsar el botón de alta de cuestionario. 4.Rellenar los datos correctamente. 5.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página. Si en el punto 4 no introduce correctamente los datos, le saldrá un error.

<b>CU-14</b>	<b>Baja de cuestionarios</b>
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Eliminará de la base de datos el cuestionario
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de cuestionarios. 3.Pulsar el botón de eliminar. 4.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página.

<b>CU-15</b>	<b>Modificación de cuestionarios</b>
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Se podrá editar el cuestionario correspondiente
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de cuestionarios. 3.Pulsar el botón de editar. 4.Rellenar el formulario correctamente. 5.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página. Si en el punto 4 no se rellena correctamente el formulario dará error.

<b>CU-16</b>	<b>Acceder al Metaverso</b>
<b>Actores</b>	<u>Profesor y alumno</u>
<b>Descripción</b>	Los usuarios iniciaran sesión en la página web
<b>Precondición</b>	Tener acceso a Internet y cuenta
<b>Secuencia</b>	1. El usuario accede a la página web.

	<p>2. El usuario se autentica correctamente.</p> <p>3. El usuario visualiza su listado de asignaturas.</p> <p>4. El usuario accede al Metaverso de una asignatura haciendo clic sobre ella.</p> <p>5. El usuario accede a la vista 3D de la aplicación donde podrá habilitar la opción de RV.</p>
<b>Postcondición</b>	<p>Si en el punto 1 no tiene Internet, no podrá acceder a la página.</p> <p>Si en el punto 2 falla el inicio de sesión (por ser incorrecto o no tener cuenta) aparecerá un mensaje indicando el tipo de error.</p>

<b>CU-17</b>	<b>Acceder Laboratorio</b>
<b>Actores</b>	<u>Profesor y alumno</u>
<b>Descripción</b>	Los usuarios iniciaran sesión en la página web
<b>Precondición</b>	Tener acceso a Internet y cuenta
<b>Secuencia</b>	<p>1.El usuario accede a la página web.</p> <p>2.El usuario introduce los datos en el formulario.</p> <p>3.El usuario pulsa el metaverso al que quiera acceder.</p> <p>4. Podrá acceder al edificio del laboratorio.</p> <p>5.Ok.</p>
<b>Postcondición</b>	<p>Si en el punto 1 no tiene Internet, no podrá acceder a la página.</p> <p>Si en el punto 2, se falla el inicio de sesión ya que es incorrecto o no tiene cuenta saldrá un mensaje que le indicará el tipo de error.</p> <p>Si en el punto 4, el metaverso falla no podrá acceder al laboratorio.</p>

<b>CU-18</b>	<b>Acceder Sala Reunión</b>
<b>Actores</b>	<u>Profesor y alumno</u>
<b>Descripción</b>	Los usuarios iniciaran sesión en la página web
<b>Precondición</b>	Tener acceso a Internet y cuenta
<b>Secuencia</b>	<ol style="list-style-type: none"> <li>1.El usuario accede a la página web.</li> <li>2.El usuario introduce los datos en el formulario.</li> <li>3.El usuario pulsa el metaverso al que quiera acceder.</li> <li>4. Podrá acceder al edificio de reuniones.</li> <li>5.Ok.</li> </ol>
<b>Postcondición</b>	<p>Si en el punto 1 no tiene Internet, no podrá acceder a la página.</p> <p>Si en el punto 2, se falla el inicio de sesión ya que es incorrecto o no tiene cuenta saldrá un mensaje que le indicará el tipo de error.</p> <p>Si en el punto 4, el metaverso falla no podrá acceder a la sala de reuniones.</p>

<b>CU-19</b>	<b>Acceder Aula</b>
<b>Actores</b>	<u>Profesor y alumno</u>
<b>Descripción</b>	Los usuarios iniciaran sesión en la página web
<b>Precondición</b>	Tener acceso a Internet y cuenta
<b>Secuencia</b>	<ol style="list-style-type: none"> <li>1.El usuario accede a la página web.</li> <li>2.El usuario introduce los datos en el formulario.</li> <li>3.El usuario pulsa el metaverso al que quiera acceder.</li> <li>4. Podrá acceder al edificio del aula.</li> <li>5.Ok.</li> </ol>
<b>Postcondición</b>	<p>Si en el punto 1 no tiene Internet, no podrá acceder a la página.</p> <p>Si en el punto 2, se falla el inicio de sesión ya que es incorrecto o no tiene cuenta saldrá un mensaje que le indicará el tipo de error.</p> <p>Si en el punto 4, el metaverso falla no podrá acceder al aula.</p>

<b>CU-20</b>	Gestión de salas (A/B/M)
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Un menú donde se podrá acceder a la gestión de salas (borrar, eliminar y añadir)
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de salas. 3.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página.

<b>CU-21</b>	Alta de salas
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Un formulario donde podrá rellenar los datos de las salas y así subirlos a la base de datos.
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de salas. 3.Pulsar el botón de alta de salas. 4.Rellenar los datos correctamente. 5.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página. Si en el punto 4 no introduce correctamente los datos, le saldrá un error.

<b>CU-22</b>	Baja de salas
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Eliminará de la base de datos la sala
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	1.Iniciar sesión. 2.Pulsar el botón de gestión de salas. 3.Pulsar el botón de eliminar. 4.OK
<b>Postcondición</b>	Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página.

<b>CU-23</b>	Modificación de salas
<b>Actores</b>	<u>Profesor</u>
<b>Descripción</b>	Se podrá editar la sala correspondiente
<b>Precondición</b>	Tener internet y cuenta
<b>Secuencia</b>	<ol style="list-style-type: none"> <li>1.Iniciar sesión.</li> <li>2.Pulsar el botón de gestión de salas.</li> <li>3.Pulsar el botón de editar.</li> <li>4.Rellenar el formulario correctamente.</li> <li>5.OK</li> </ol>
<b>Postcondición</b>	<p>Si en el punto 1 no tiene internet, o no tiene cuenta no podrá acceder a la página.</p> <p>Si en el punto 4 no se rellena correctamente el formulario dará error.</p>

