



UNIVERSIDAD
CATÓLICA
DE CUENCA

UNIVERSIDAD CATÓLICA DE CUENCA

Comunidad Educativa al Servicio del Pueblo

**FACULTAD DE INFORMÁTICA, CIENCIAS DE LA
COMPUTACIÓN E INNOVACIÓN TECNOLÓGICA**

CARRERA DE REALIDAD VIRTUAL Y VIDEOJUEGOS

**Diseño e implementación de un simulador educativo en realidad virtual basado en
arquitectura de software, mecánicas interactivas y game design patterns para el laboratorio
de comercio exterior y logística internacional**

**PROYECTO DE INTEGRACIÓN CURRICULAR PREVIO A LA OBTENCIÓN
DEL TÍTULO DE INGENIERO EN REALIDAD VIRTUAL Y VIDEOJUEGOS**

AUTOR: IKA EDUARDO ZHUNIO MACHUCA

DIRECTOR: ING. JOHN LUIS ESTRADA GUAYTA

CUENCA - ECUADOR

2026

DIOS, PATRIA, CULTURA Y DESARROLL



UNIVERSIDAD CATÓLICA DE CUENCA

Comunidad Educativa al Servicio del Pueblo

**FACULTAD DE INFORMÁTICA, CIENCIAS DE LA
COMPUTACIÓN E INNOVACIÓN TECNOLÓGICA**

CARRERA DE REALIDAD VIRTUAL Y VIDEOJUEGOS

**Diseño e implementación de un simulador educativo en realidad
virtual basado en arquitectura de software, mecánicas
interactivas y game design patterns para el laboratorio de
comercio exterior y logística internacional**

**PROYECTO DE INTEGRACIÓN CURRICULAR PREVIO A LA
OBTENCIÓN DEL TÍTULO DE INGENIERO EN REALIDAD
VIRTUAL Y VIDEOJUEGOS**

AUTOR: IKA EDUARDO ZHUNIO MACHUCA

DIRECTOR: ING. JOHN LUIS ESTRADA GUAYTA

CUENCA – ECUADOR

2026

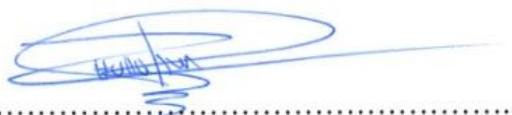
DIOS, PATRIA, CULTURA Y DESARROLLO



DECLARATORIA DE AUTORÍA

Ika Eduardo Zhunio Machuca portador de la cédula de ciudadanía N° **0107340630**. Declaro ser el autor de la obra: **“Diseño e implementación de un simulador educativo en realidad virtual basado en arquitectura de software, mecánicas interactivas y game design patterns para el laboratorio de comercio exterior y logística internacional.”**, sobre la cual me hago responsable sobre las opiniones, versiones e ideas expresadas. Declaro que la misma ha sido elaborada respetando los derechos de propiedad intelectual de terceros y eximo a la Universidad Católica de Cuenca sobre cualquier reclamación que pudiera existir al respecto. Declaro finalmente que mi obra ha sido realizada cumpliendo con todos los requisitos legales, éticos y bioéticos de investigación, que la misma no incumple con la normativa nacional e internacional en el área específica de investigación, sobre la que también me responsabilizo y eximo a la Universidad Católica de Cuenca de toda reclamación al respecto.

Cuenca, **7 de septiembre de 2026**

F: 

Ika Eduardo Zhunio Machuca

C.I. 0107340630

**CERTIFICADO DEL TUTOR**

Yo, John Luis Estrada Guayta certifico que el presente proyecto de titulación, con el título “Diseño e implementación de un simulador educativo en realidad virtual basado en arquitectura de software, mecánicas interactivas y game design patterns para el laboratorio de comercio exterior y logística internacional.”, fue desarrollado por Ika Eduardo Zhunio Machuca, con número de cédula 0107340630, bajo mi supervisión.



F:

John Luis Estrada Guayta

C.I. 1716532781

Dedicatoria

Dedico este trabajo a mis abuelos paternos y maternos, pilares fundamentales de mi vida, por el amor, las enseñanzas y el ejemplo que me han brindado.

De manera especial, a mi abuelita, Mariana Ortega, cuyos sacrificios hicieron posible mi formación. Gracias por apoyarme, acompañarme durante tantas noches y confiar siempre en mí. Este logro también es suyo.

A la memoria de mi abuelo, Alejo Zhunio. Aunque partió antes de verme alcanzar esta meta, su vida no terminó con su ausencia, sino que continúa en quienes tuvimos la dicha de compartirla. Sus enseñanzas, sus acciones y el amor que sembró permanecen en nosotros y forman parte de lo que hoy somos. Por eso, este logro también le pertenece: es uno de los frutos de su legado y de todo aquello que su existencia puso en movimiento.

A mis dos padres y a mis dos madres, por su amor, esfuerzo y apoyo incondicional. De manera especial, a mi padre, Juan Zhunio, por ser, además de un gran padre, un verdadero hermano y compañero de vida. Gracias por caminar a mi lado, comprenderme y brindarme siempre su confianza y apoyo.

A mis hermanos y a mis tíos, por su cariño, compañía, consejos y confianza a lo largo de este camino.

Finalmente, me dedico este logro a mí mismo, por el esfuerzo, la perseverancia y la determinación para seguir adelante frente a cada dificultad.

Agradecimientos

En primer lugar, agradezco a Dios y a la Virgencita por concederme la fortaleza, la paciencia y la perseverancia necesarias para superar las dificultades y continuar durante este proceso de aprendizaje.

Al finalizar esta etapa tan significativa, agradezco sinceramente a todas las personas que estuvieron a mi lado. Sus consejos, palabras de ánimo y muestras de apoyo fueron fundamentales para seguir avanzando y alcanzar esta meta.

A mi tutor, Ing. John Luis Estrada Guayta, por su guía, disposición y enseñanzas durante el desarrollo de este trabajo. Sus conocimientos y recomendaciones fueron esenciales para culminar satisfactoriamente este proyecto. Asimismo, agradezco a los docentes universitarios que participaron en mi formación, por compartir sus conocimientos y prepararme para afrontar los desafíos de la vida profesional.

A mis amigos y compañeros, por su apoyo, compañía y por hacer de esta etapa una experiencia más divertida, agradable y memorable.

También agradezco a una persona que formó parte importante de este camino, por su compañía, apoyo y motivación. Aunque hoy nuestros caminos sean diferentes, siempre valoraré con gratitud su contribución a este proceso.

Finalmente, agradezco a quienes contribuyeron, directa o indirectamente, a mi formación y a la culminación de esta meta.

Resumen

El presente trabajo tuvo como objetivo diseñar e implementar un simulador educativo en realidad virtual para apoyar la práctica de procesos de exportación e importación en estudiantes de la Facultad de Ciencias Económicas y Empresariales. El sistema fue desarrollado en Unity mediante Meta XR SDK y se estructuró mediante una arquitectura modular basada en los patrones Singleton, Observer y Template Method. Se implementaron mecánicas de seguimiento de manos, reconocimiento de gestos, teletransportación, agarre de objetos y validación mediante sockets, además de módulos para administrar tareas, registrar datos, controlar el progreso y gestionar tutoriales interactivos con animaciones de manos virtuales. Las pruebas funcionales permitieron verificar el inicio de la aplicación, la transición entre escenas, la locomoción, el agarre con ambas manos, las tareas, las interfaces y el audio direccional. En Meta Quest 3S, con un límite de 72 fotogramas por segundo, se registraron valores aproximados entre 52 y 72 FPS. Asimismo, una prueba piloto con cinco estudiantes permitió identificar dificultades relacionadas con la locomoción, el alcance de agarre y las poses de las manos, las cuales fueron ajustadas posteriormente. Se concluye que el prototipo establece una base tecnológica funcional para la práctica guiada de procesos de comercio exterior, aunque su contribución al aprendizaje deberá determinarse mediante una validación educativa posterior.

Palabras clave: *realidad virtual, simulación educativa, comercio exterior, arquitectura de software, interacción natural.*

Abstract

This study aimed to design and implement an educational virtual reality simulator to support the practice of export and import processes among students from the Faculty of Economic and Business Sciences. The system was developed in Unity using the Meta XR SDK and structured through a modular architecture based on the Singleton, Observer, and Template Method design patterns. The simulator incorporated hand tracking, gesture recognition, teleportation, object grabbing, and socket-based validation, together with modules for task management, data recording, user progress tracking, and interactive tutorials supported by virtual hand animations. Functional testing verified application startup, scene transitions, locomotion, two-handed grabbing, task execution, interface behavior, and directional audio. On Meta Quest 3S, with a target cap of 72 frames per second, performance ranged approximately from 52 to 72 FPS. In addition, a pilot test with five students identified usability issues related to locomotion, grabbing distance, and hand poses, which were subsequently adjusted. The results indicate that the prototype provides a functional technological foundation for guided practice in foreign trade processes; however, its contribution to learning outcomes still requires further validation through a formal educational evaluation.

Keywords: *Virtual reality, educational simulation, foreign trade, software architecture, natural interaction.*

Índice

Declaratoria de Autoría	II
Certificado del Tutor	III
Dedicatoria.....	IV
Agradecimientos.....	V
Resumen	VI
Abstract.....	VII
Índice	VIII
Índice de Tablas.....	XII
Índice de Ilustraciones	XIII
Capítulo I.....	1
1. Marco Referencial	1
1.1. Introducción	1
1.2. Contextualización del problema	3
1.3. Planteamiento del problema.....	5
1.4. Formulación del problema	6
1.5. Preguntas específicas	6
1.6. Objetivo general.....	7
1.7. Objetivos específicos	7
1.8. Justificación	8
1.9. Alcance	9
1.9.1. Análisis de requerimientos	10
1.9.2. Diseño de la arquitectura de software.....	10
1.9.3. Comprobación técnica y de usabilidad mecánica.....	11
1.9.4. Limitaciones del alcance	12
1.10. Delimitaciones.....	13
1.10.1. Delimitación Espacial	13
1.10.2. Delimitación Temporal	13
1.10.3. Delimitación Tecnológica.....	13
1.10.4. Delimitación Funcional.....	14
1.10.5. Delimitación Metodológica	14
1.10.6. Delimitación Económica.....	14
1.11. Beneficiarios.....	14
1.11.1. Beneficiarios directos	14

1.11.2.	Beneficiarios indirectos	15
1.12.	Metodología general.....	16
1.12.1.	Metodología de investigación	17
1.12.2.	Metodología de desarrollo Kanban.....	19
1.13.	Resultados esperados.....	21
1.14.	Relación con el protocolo de investigación.....	21
Capítulo II.....		23
2.	Marco Teórico	23
2.1.	Antecedentes investigativos / Estado del arte	23
2.2.	Bases teóricas.....	27
2.2.1.	Aprendizaje inmersivo.....	27
2.2.2.	Aprendizaje procedimental y secuenciación	28
2.2.3.	Presencia, agencia y corporeidad.....	28
2.2.4.	Retroalimentación multimodal	29
2.2.5.	Usabilidad en realidad virtual.....	29
2.3.	Bases metodológicas	30
2.4.	Bases tecnológicas	32
2.4.1.	Unity y C#	32
2.4.2.	Meta XR Interaction SDK	32
2.4.3.	Meta Quest 3S	32
2.4.4.	Interacción mediante hand tracking.....	33
2.5.	Arquitecturas, modelos y estándares.....	33
2.5.1.	Organización modular	33
2.5.2.	Programación orientada a objetos.....	34
2.5.3.	Gestión de etapas	34
2.5.4.	Eventos y Observer.....	35
2.5.5.	Validación mediante sockets y tareas	35
2.6.	Herramientas de desarrollo	35
2.6.1.	Unity 6000.3.5f2.....	35
2.6.2.	Meta XR Interaction SDK 85.0.0	36
2.6.3.	Meta Quest 3S	36
2.6.4.	Diagramas de clase.	36
2.7.	Marco conceptual.....	36

2.7.1.	Realidad virtual inmersiva.....	36
2.7.2.	Simulador educativo.....	36
2.7.3.	Head-mounted display (HMD).....	37
2.7.4.	Hand tracking.	37
2.7.5.	Interacción natural.	37
2.7.6.	Grabbable object.....	37
2.7.7.	Hand pose.	37
2.7.8.	Socket.	37
2.7.9.	Teletransportación.	37
2.7.10.	Retroalimentación multimodal.	37
2.7.11.	Audio espacial.....	38
2.7.12.	Persistencia	38
2.8.	Relación entre la teoría y la propuesta tecnológica	38
Capítulo III		41
3.	Manual de Usuario	41
3.1.	Presentación de la propuesta y perfil del usuario	41
3.1.1.	Descripción general de Explorador Global	41
3.1.2.	Perfil del usuario.....	41
3.1.3.	Requisitos mínimos y condiciones de uso	42
3.2.	Acceso e inicio de la experiencia.....	42
3.2.1.	Preparación del espacio y del visor	42
3.2.2.	Inicio de la aplicación.....	43
3.2.3.	Tutorial inicial	43
3.2.4.	Selección del módulo e inicio del recorrido	44
3.3.	Navegación y controles.....	45
3.3.1.	Seguimiento de manos.....	45
3.3.2.	Locomoción gestual y teletransportación	45
3.3.3.	Giro y desplazamiento por pasos.....	47
3.3.4.	Selección de botones y elementos de interfaz	48
3.3.5.	Agarre y manipulación de objetos	49
3.4.	Funciones e interacciones disponibles.....	50
3.4.1.	Sistema de etapas y tareas	50
3.4.2.	Interacción con documentos	50

3.4.3.	Validación de objetos y acciones.....	51
3.4.4.	Progreso y finalización de etapas	51
3.4.5.	Recorridos de exportación e importación.....	51
3.5.	Evidencia de uso	52
3.5.1.	Inicio y selección del recorrido	52
3.5.2.	Inicio de una actividad.....	53
3.5.3.	Interacción y validación durante la actividad	54
3.5.4.	Finalización de una etapa	55
Capítulo IV		57
4.	Manual De Programador	57
4.1.	Arquitectura general del sistema.....	57
4.2.	Tecnologías, motores y herramientas utilizadas	58
4.3.	Organización técnica del proyecto.....	59
4.4.	Arquitectura de clases y componentes.....	62
4.5.	Desarrollo de sistemas principales.....	63
4.5.1.	Sistema central de etapas	63
4.5.2.	Controladores de exportación.....	65
4.5.3.	Controladores de importación	66
4.6.	Desarrollo de mecánicas principales.....	67
4.7.	Integración de sistemas	67
4.8.	Integración entre diseño, arte y programación.....	70
4.9.	Gestión de datos y persistencia	71
4.10.	Configuración de escenas y flujo del simulador	72
4.11.	Pruebas técnicas y validación.....	72
Capítulo V		77
5.	Conclusiones y Recomendaciones	77
5.1.	Conclusiones	77
5.2.	Recomendaciones	78
Bibliografía		80

Índice de Tablas

Tabla 1 Correspondencia entre los fundamentos teóricos y las decisiones aplicadas en Explorador Global	38
Tabla 2 Organización funcional de los módulos, responsabilidades y componentes principales de Explorador Global	60
Tabla 3 Resultados de las pruebas técnicas y de la prueba piloto de usabilidad mecánica	73

Índice de Ilustraciones

Ilustración 1	Presentación del tutorial inicial de interacción	44
Ilustración 2	Secuencia de activación de la locomoción gestual	46
Ilustración 3	Confirmación de un punto de teletransportación	47
Ilustración 4	Control gestual de giro durante la navegación.....	48
Ilustración 5	Selección de un elemento de interfaz mediante hand tracking	49
Ilustración 6	Secuencia básica de aproximación y agarre de un objeto.....	50
Ilustración 7	Selección de los módulos de exportación e importación	52
Ilustración 8	Panel informativo al inicio de una etapa.....	53
Ilustración 9	Manipulación y traslado de un objeto dentro del escenario.....	54
Ilustración 10	Panel de finalización de una etapa de Explorador Global	55
Ilustración 11	Mapa de módulos del repositorio.....	57
Ilustración 12	Paquetes instalados en el proyecto.....	59
Ilustración 13	Jerarquía global de herencias del sistema primera parte.....	63
Ilustración 14	Arquitectura del sistema central de etapas.....	64
Ilustración 15	Controladores del recorrido de exportación.....	65
Ilustración 16	Controladores del recorrido de importación.	66
Ilustración 17	Integración del sistema con Meta XR.....	68
Ilustración 18	Funcionamiento del sistema de sockets	69
Ilustración 19	Comunicación entre tareas e interfaz de progreso.	70
Ilustración 20	Gestión temporal del progreso durante la ejecución.....	71

Capítulo I

1. Marco Referencial

1.1. Introducción

La transformación digital ha favorecido el uso de tecnologías inmersivas en diferentes espacios de aprendizaje. Estas herramientas ofrecen nuevas formas de representar, visualizar e interactuar con situaciones difíciles de reproducir mediante recursos tradicionales. En la educación superior, su incorporación todavía mantiene un carácter principalmente exploratorio y suele desarrollarse en experiencias específicas, más que como parte permanente de los programas académicos (Stracke et al., 2025). Durante los últimos años, el estudio de sus aplicaciones ha crecido de manera constante, especialmente en la educación superior, donde se emplea para apoyar la formación en asignaturas que requieren práctica y la comprensión de procedimientos complejos (Radianti et al., 2020; Pinargote Castro et al., 2024).

La literatura científica registra aplicaciones de realidad virtual en medicina, ingeniería, arquitectura y otras áreas. A través de estos recursos es posible recrear escenarios de práctica y experimentar dentro de condiciones controladas. Las propiedades inmersivas e interactivas pueden favorecer el interés, la participación y el desarrollo de actividades educativas que no resultan fáciles de ejecutar en espacios físicos convencionales (Di Natale et al., 2020).

Uno de los retos habituales en las universidades es ofrecer espacios donde los estudiantes puedan acercarse a procesos profesionales que no son fáciles de reproducir dentro de un aula. La realidad virtual brinda una alternativa para recrear esos contextos mediante experiencias inmersivas semejantes a determinadas situaciones del campo laboral. Distintos estudios reconocen a la simulación como una de las aplicaciones más utilizadas de esta tecnología en la educación superior, pues facilita la representación de

procedimientos complejos sin depender de espacios físicos reales (Jongbloed et al., 2024; Radianti et al., 2020; Stracke et al., 2025).

Esta dificultad puede observarse con claridad en el comercio exterior. Las operaciones de exportación e importación comprenden actividades vinculadas con la preparación de documentos, inspecciones, clasificación de mercancías y transporte internacional. También incluyen procedimientos aduaneros y validaciones administrativas. Para ejecutar estas operaciones resulta necesario comprender cómo se conectan sus etapas y cuál es el recorrido que siguen las mercancías desde el origen hasta el destino. No obstante, los programas universitarios suelen abordar buena parte de estos contenidos mediante exposiciones, revisión de normas y análisis documental. Esto reduce las oportunidades de interactuar y practicar en escenarios cercanos a los que existen en el ámbito profesional.

La revisión de la literatura mostró un aumento de investigaciones sobre realidad virtual aplicada a la educación superior. Sin embargo, la mayoría de las propuestas se concentra en ciencias de la salud, ingeniería y capacitación técnica. Los trabajos dedicados a crear simuladores inmersivos para enseñar procesos de comercio exterior siguen siendo limitados (Pellas et al., 2021; Radianti et al., 2020; Stracke et al., 2025). Esta diferencia abre una posibilidad de estudio desde la Ingeniería de Software, que no se restringe al uso de la realidad virtual como herramienta. También comprende diseñar y desarrollar una solución que represente un proceso real de forma organizada dentro de un entorno virtual interactivo.

Ante esta situación, el trabajo propone diseñar y desarrollar un simulador educativo de realidad virtual para apoyar la enseñanza de procesos de comercio exterior a estudiantes universitarios. La investigación, abordada desde la Ingeniería de Software,

comprende el análisis de requerimientos y el diseño de la arquitectura del sistema. Incluye además la programación de mecanismos de interacción y la creación de un entorno inmersivo que representa las principales actividades de exportación e importación. La solución busca servir como punto de partida para futuras aplicaciones de simulación destinadas a contextos académicos relacionados con el comercio internacional.

1.2. **Contextualización del problema**

El comercio exterior constituye una de las principales actividades que impulsan el desarrollo económico y la integración de los países en los mercados internacionales. Su ejecución abarca un conjunto de procesos administrativos, logísticos y aduaneros que permiten el intercambio de bienes y servicios entre diferentes territorios. Uno de los retos habituales de las universidades es ofrecer espacios donde los estudiantes puedan acercarse a procesos profesionales difíciles de reproducir dentro del aula. Las revisiones recientes muestran que la realidad virtual se ha utilizado para crear experiencias prácticas, recorridos virtuales y simulaciones educativas en distintas disciplinas, aunque su utilidad depende de la actividad y del diseño aplicado (Pellas et al., 2021).

En el ámbito universitario, la enseñanza del comercio exterior busca proporcionar a los estudiantes los conocimientos necesarios para comprender el reglamento activo, la gestión, los procedimientos de exportación e importación, la clasificación arancelaria, las operaciones logísticas y los mecanismos de control aduanero. Sin embargo, gran parte de estos contenidos se desarrollan mediante estrategias de enseñanza tradicionales, apoyadas principalmente en clases orales, estudios de legislación, análisis de caso y revisión analítica. Aunque estas metodologías permiten fortalecer los conocimientos conceptuales, presentan ciertas limitaciones para representar la dinámica operativa de los procesos reales, que deben ejecutarse siguiendo una secuencia determinada (Pinargote Castro et al., 2024).

Esta situación se evidencia en la formación de los estudiantes de la Facultad de Ciencias Económicas y Empresariales, donde los contenidos relacionados con comercio exterior precisan de la comprensión de procedimientos que dependen de la interacción entre diversas entidades y elementos. La preparación de documentos de exportación, el reconocimiento de mercancías, el proceso de aforo, la inspección aduanera y la validación de requisitos legales forman parte de situaciones que pueden reproducirse con cierta dificultad dentro del aula debido a limitaciones de infraestructura, disponibilidad de recursos, condiciones de seguridad y acceso a instalaciones especializadas.

En consecuencia, el proceso de enseñanza se apoya principalmente en recursos gráficos y presentaciones, limitando las oportunidades de interacción directa con entornos similares a los encontrados en el ámbito profesional. Diversos autores coinciden en que la realidad virtual ha permitido superar parcialmente estas limitaciones mediante la creación de espacios inmersivos capaces de representar situaciones complejas, ofreciendo escenarios controlados donde los usuarios pueden interactuar con objetos virtuales y ejecutar procedimientos similares a los desarrollados en contextos reales (Radianti et al., 2020; Pinargote Castro et al., 2024; Magallanes Rodríguez et al., 2021).

La revisión de la literatura demuestra que las aplicaciones de realidad virtual en educación superior se han centrado principalmente en áreas como medicina, ingeniería, arquitectura y formación técnica, donde los simuladores son utilizados para representar procedimientos especializados, entrenar habilidades prácticas y recrear escenarios que implican altos costos o riesgos operativos. No obstante, se observa una limitada disponibilidad de soluciones orientadas específicamente a la simulación de procesos de comercio exterior, principalmente aquellas que involucran varias etapas del proceso de comercialización, orientado hacia la exportación e importación (Radianti et al., 2020).

Dado el caso, el desarrollo de un simulador educativo en realidad virtual representa una oportunidad para dar soporte a las estrategias de enseñanza tradicionales mediante la recreación de espacios inmersivos que permitan representar los procedimientos fundamentales del comercio exterior. Desde la perspectiva de la Ingeniería de Software, esta propuesta implica modelar procesos reales, representar eventos propios de las operaciones aduaneras y desarrollar una arquitectura que facilite la evolución y mantenimiento del sistema. En consecuencia, la contextualización descrita muestra la existencia de un entorno académico donde convergen una necesidad formativa y una oportunidad tecnológica que justifican el desarrollo de la presente investigación.

1.3. **Planteamiento del problema**

La enseñanza del comercio exterior brinda a los estudiantes conocimientos sobre normativa, documentación y procedimientos vinculados con las operaciones de importación y exportación. Sin embargo, los sistemas educativos convencionales tienen dificultades para recrear de manera efectiva la secuencia de estos procesos y la participación de los distintos actores involucrados. La comprensión de las actividades depende, en gran medida, de recursos teóricos y conceptuales, lo que reduce las oportunidades de experimentar situaciones semejantes a las del futuro entorno profesional (Pinargote Castro et al., 2024).

La realidad virtual ha demostrado que puede utilizarse de forma eficiente en diferentes áreas, entre ellas la medicina y diversas ramas de la ingeniería. A pesar de ello, la literatura registra una disponibilidad limitada de simuladores inmersivos dirigidos específicamente a enseñar procesos de comercio exterior en el contexto universitario (Radianti et al., 2020). Esta ausencia abre la posibilidad de crear soluciones tecnológicas que complementen los métodos habituales de enseñanza. Mediante entornos interactivos

podrían representarse escenarios apropiados para practicar actividades relacionadas con la importación y la exportación.

Frente a esta situación, resulta necesario diseñar y desarrollar un simulador educativo de realidad virtual que represente las principales actividades del comercio exterior. El proyecto debe incorporar principios de Ingeniería de Software para construir una arquitectura modular, mantenible y escalable. Una solución con estas características podría complementar la formación de los estudiantes de la Facultad de Ciencias Económicas y Empresariales mediante una herramienta tecnológica centrada en simular situaciones reales que pueden presentarse dentro del entorno laboral.

1.4. **Formulación del problema**

¿Cómo desarrollar un simulador educativo en realidad virtual, basado en una arquitectura de software modular y mecánicas interactivas mediante hand tracking, que permita representar los procesos simplificados de exportación e importación definidos por el proyecto LACELI y ejecutarlos de forma funcional en dispositivos Meta Quest 3 y Meta Quest 3S?

1.5. **Preguntas específicas**

1. ¿Cuáles son los requerimientos funcionales, pedagógicos y de interacción definidos por el proyecto LACELI para representar los recorridos simplificados de exportación e importación en el simulador?
2. ¿Cómo debe estructurarse una arquitectura de software modular que organice las etapas, tareas, validaciones e interacciones del sistema y facilite su mantenimiento y ampliación?

3. ¿Cómo pueden integrarse Unity y Meta XR Interaction SDK para implementar hand tracking, locomoción gestual, manipulación de objetos, tutoriales contextuales y retroalimentación multimodal?
4. ¿Cómo se puede comprobar el funcionamiento técnico, la usabilidad mecánica y el rendimiento inicial del simulador mediante pruebas realizadas en Meta Quest 3S?

1.6. **Objetivo general**

Desarrollar el sistema Explorador Global, un simulador educativo en realidad virtual mediante Unity y Meta XR Interaction SDK, orientado a representar los procesos simplificados de exportación e importación definidos por el proyecto LACELI, mediante interacción inmersiva con Hand Tracking y una arquitectura de software modular.

1.7. **Objetivos específicos**

Analizar los requerimientos funcionales, pedagógicos y de interacción definidos por el proyecto LACELI para determinar los sistemas necesarios en los recorridos simplificados de exportación e importación.

Diseñar una arquitectura de software modular basada en programación orientada a objetos, componentes y eventos, que organice las etapas, tareas, validaciones e interacciones del simulador y facilite su mantenimiento y ampliación.

Implementar en Unity, mediante Meta XR Interaction SDK, los sistemas de hand tracking, locomoción gestual, manipulación de objetos, tutoriales contextuales y retroalimentación multimodal requeridos para la experiencia inmersiva.

Comprobar el funcionamiento técnico, la usabilidad mecánica y el rendimiento inicial del simulador mediante pruebas de ejecución y una prueba piloto con usuarios en Meta Quest 3S.

1.8. **Justificación**

La propuesta tiene valor académico porque incorpora un recurso complementario para representar actividades que deben cumplirse en secuencia. Dentro del entorno controlado de realidad virtual, el usuario puede actuar, observar la respuesta del sistema y repetir las tareas. La presencia y la agencia constituyen posibilidades importantes para el aprendizaje inmersivo, según Makransky y Petersen (2021). La primera corresponde a la percepción de estar dentro del escenario; la segunda aparece cuando las acciones realizadas generan efectos que el usuario puede comprender.

El proyecto no pretende sustituir la enseñanza especializada ni afirmar que, por sí solo, mejora el aprendizaje. Busca ofrecer una herramienta de apoyo que el área académica pueda evaluar posteriormente. Esta posición concuerda con Hamilton et al. (2021), quienes señalan que las experiencias inmersivas necesitan evaluaciones rigurosas antes de relacionarlas con resultados educativos.

En el componente tecnológico, el trabajo presenta una arquitectura modular construida en Unity. Los sistemas encargados de las etapas, tareas, sockets y documentos cumplen responsabilidades definidas. Otros componentes administran los tutoriales, el progreso y las validaciones. La herencia ayuda a reutilizar el ciclo común de las etapas, y los eventos comunican diferentes elementos del sistema sin reunir todas las respuestas en un único componente.

Meta XR Interaction SDK hizo posible incorporar el seguimiento de manos, la manipulación de objetos, los sockets y la locomoción. La interacción mediante gestos forma parte del SDK. El trabajo realizado en el proyecto se concentró en configurarla, enlazarla con el flujo principal y ajustarla a los requerimientos establecidos. Esta aclaración evita presentar como propios los algoritmos o recursos creados por Meta.

La usabilidad constituye otra razón que sustenta la propuesta. Una interacción que parece natural puede demandar más esfuerzo cuando el sistema no ofrece la precisión necesaria, como muestran Hameed et al. (2023). Las dificultades detectadas durante la prueba piloto llevaron a modificar el alcance de agarre, incorporar poses para ambas manos y utilizar teletransporte mediante gestos. Cada uno de estos cambios surgió directamente de lo observado durante la interacción de los participantes.

Los estudiantes y docentes relacionados con el proyecto LACELI son sus beneficiarios directos. Los primeros cuentan con una experiencia interactiva, y los docentes disponen de un recurso que puede complementar sus explicaciones. Para la institución, el proyecto proporciona una base tecnológica que podría ampliarse más adelante con nuevas actividades inmersivas.

El trabajo también guarda relación con la carrera de Ingeniería en Realidad Virtual y Videojuegos, ya que reúne programación orientada a objetos y arquitectura de software. A esto se suman la interacción humano-computador, el desarrollo en realidad virtual y las pruebas realizadas directamente en hardware. También incluye tareas de optimización. El producto muestra cómo estas competencias pueden aplicarse de forma conjunta ante una necesidad institucional.

1.9. Alcance

La propuesta abarcará el desarrollo de un prototipo funcional en realidad virtual. Este prototipo permitirá que el usuario seleccione uno de los recorridos disponibles, avance por una secuencia de etapas y realice actividades mediante seguimiento de manos. Cada módulo presentará instrucciones, tareas, objetos interactivos y condiciones de validación. Los recorridos tendrán una finalidad educativa y no reproducirán de forma completa todos los procedimientos legales, administrativos o aduaneros que intervienen en una operación real de comercio exterior.

1.9.1. Análisis de requerimientos

El proyecto permitirá analizar los requerimientos funcionales, pedagógicos y de interacción proporcionados por el responsable del proyecto LACELI. Este análisis tendrá como finalidad identificar qué funciones deben transformarse en sistemas de software y establecer cómo participará el usuario en los recorridos de exportación e importación.

El análisis comprenderá la revisión de las actividades definidas para cada recorrido, el orden de las etapas, las tareas que deberá completar el usuario y las respuestas que ofrecerá el sistema. También se identificarán los objetos manipulables, los documentos representados, las instrucciones necesarias y las condiciones que permitirán determinar si una acción fue correcta o incorrecta.

Los requerimientos pedagógicos se limitarán a la secuencia educativa entregada por LACELI. La investigación no formulará contenidos propios sobre normativa aduanera ni evaluará la exactitud profesional de los procesos. La responsabilidad del trabajo estará en interpretar los requerimientos recibidos y convertirlos en funciones programables dentro del simulador.

Los requerimientos de interacción incluirán el uso de hand tracking, el desplazamiento dentro de los escenarios, la selección de elementos y el agarre de objetos. También abarcarán la colocación de elementos en espacios determinados, la interacción con interfaces y la orientación mediante ayudas visuales o auditivas.

1.9.2. Diseño de la arquitectura de software

El proyecto permitirá diseñar una arquitectura modular para organizar la lógica del simulador. La estructura se basará en programación orientada a objetos, componentes de Unity y comunicación mediante eventos. Su finalidad será distribuir las responsabilidades entre distintos sistemas y evitar que toda la lógica dependa de un único controlador.

El diseño contemplará un sistema para administrar el recorrido general y la transición entre etapas. Cada etapa tendrá tareas concretas, condiciones de inicio y criterios de finalización.

La arquitectura también incorporará módulos destinados a la manipulación de documentos, validación mediante sockets, tutoriales contextuales, audio, locomoción e interacción con objetos. El diseño buscará que los componentes puedan reutilizarse en diferentes etapas y que una modificación localizada produzca el menor impacto posible sobre los demás sistemas.

Se elaborarán diagramas para representar la organización de los módulos, las principales relaciones entre clases y el funcionamiento de los sistemas centrales. Estos recursos servirán para documentar la solución y facilitar su comprensión técnica. No se planteará una arquitectura distribuida, una plataforma web ni un sistema basado en microservicios, debido a que la aplicación funcionará directamente en un visor autónomo.

1.9.3. Comprobación técnica y de usabilidad mecánica

El proyecto permitirá comprobar el funcionamiento técnico, la usabilidad mecánica y el rendimiento inicial del simulador en Meta Quest 3S. La comprobación técnica revisará el inicio de la aplicación, la selección de módulos, la carga de escenas y la transición entre etapas. También observará el funcionamiento del teletransporte, el seguimiento de manos, el agarre, los sockets, las tareas, las interfaces, los tutoriales y el audio espacial.

La usabilidad mecánica se revisará mediante una prueba piloto con cinco estudiantes de distintas carreras. Esta prueba buscará determinar si los participantes pueden comprender y utilizar las mecánicas básicas sin depender de conocimientos previos sobre comercio exterior. La información se obtendrá mediante observación directa y comentarios verbales.

La prueba piloto servirá para identificar dificultades relacionadas con el desplazamiento, la selección de objetos, el agarre y la comprensión de las ayudas. Las observaciones podrán utilizarse para realizar ajustes en las interacciones antes de cerrar el desarrollo. Esta actividad tendrá un carácter exploratorio y no constituirá una evaluación educativa formal.

El rendimiento inicial se observará directamente en Meta Quest 3S con un límite configurado de 72 FPS. Se revisará el comportamiento de la aplicación durante la ejecución y se conservarán capturas como evidencia de momentos específicos. Los registros no se utilizarán como un promedio estadístico ni como una prueba prolongada de rendimiento, sino como una comprobación técnica inicial del prototipo.

1.9.4. Limitaciones del alcance

El trabajo no evaluará la adquisición de conocimientos sobre exportación o importación. Tampoco aplicará una comparación anterior y posterior, un grupo de control o instrumentos destinados a medir aprendizaje. Por esta razón, los resultados no permitirán afirmar que el simulador mejora el desempeño académico de los estudiantes. Una validación educativa con una muestra más amplia deberá desarrollarse en una investigación posterior.

Explorador Global no funcionará como una plataforma completa de gestión del aprendizaje. Quedarán fuera del alcance la administración de cursos, el registro de calificaciones, la generación de reportes docentes y el seguimiento permanente de estudiantes. Tampoco se incorporarán funciones multiusuario, conexión con servicios externos o comunicación en tiempo real.

La propuesta no incluirá simulaciones económicas avanzadas, actualización automática de normativa ni conexión con instituciones aduaneras. Los documentos, decisiones y actividades que aparezcan dentro del entorno corresponderán a una

representación educativa simplificada. Su contenido seguirá las directrices proporcionadas por LACELI y no deberá utilizarse como sustituto de procedimientos oficiales de comercio exterior.

El resultado esperado será un prototipo funcional que integre la arquitectura, las mecánicas y los recorridos definidos dentro del alcance. La contribución principal se concentrará en analizar los requerimientos, diseñar la estructura del software, implementar los sistemas inmersivos y comprobar su funcionamiento inicial en el dispositivo seleccionado.

1.10. Delimitaciones

1.10.1. Delimitación Espacial

El proyecto fue desarrollado para la carrera de Negocios Internacionales de la Universidad Católica de Cuenca. La aplicación representa escenarios virtuales vinculados con actividades simplificadas de exportación e importación. Las pruebas técnicas se realizaron utilizando un visor Meta Quest 3S en espacios controlados y libres de obstáculos.

1.10.2. Delimitación Temporal

El proceso de desarrollo, revisión y ajuste se efectuará entre abril y finales de agosto de 2026. Durante este periodo se deben mantener reuniones semanales con el responsable del proyecto LACELI para presentar avances, revisar las funciones implementadas y solicitar modificaciones. La evaluación corresponde a la versión disponible al finalizar dicho periodo.

1.10.3. Delimitación Tecnológica

El desarrollo utiliza Unity 6000.3.5f2, C# y Meta XR Interaction SDK 85.0.0. Los dispositivos objetivo son Meta Quest 3 y Quest 3S; las pruebas descritas se realizaron en Quest 3S.

1.10.4. Delimitación Funcional

La versión actual funciona mediante hand tracking y no requiere controladores físicos. El progreso se mantiene en memoria con una clase estática y se elimina al cerrar la aplicación.

1.10.5. Delimitación Metodológica

La prueba piloto contó con cinco estudiantes de diferentes carreras. Se aplicaron observación directa y comentarios verbales. No existió grupo de control, pretest, posttest ni escala cuantitativa; por tanto, no se determina la efectividad educativa frente a metodologías tradicionales.

1.10.6. Delimitación Económica

El proyecto se desarrolló con los recursos tecnológicos disponibles para el equipo y la institución. Se utilizó Unity como motor de desarrollo, Meta XR Interaction SDK y un visor Meta Quest 3S para las pruebas.

El estudio no incluye un análisis financiero, presupuesto de implementación institucional, costos de mantenimiento, adquisición masiva de visores ni estimación de retorno de inversión. Tampoco contempla gastos relacionados con infraestructura de servidores o almacenamiento, debido a que la aplicación funciona localmente y no incorpora servicios en la nube ni persistencia de datos.

1.11. Beneficiarios

1.11.1. Beneficiarios directos

Los estudiantes de la Facultad de Ciencias Económicas y Empresariales que utilicen Explorador Global serán los principales beneficiarios directos. El simulador les ofrecerá un recurso complementario para aproximarse a los recorridos simplificados de exportación e importación definidos por LACELI. Dentro del entorno virtual podrán

revisar instrucciones, manipular objetos, reconocer documentos representados y completar actividades organizadas por etapas.

El beneficio estará en disponer de una experiencia interactiva que acompañe los contenidos impartidos por el área académica. Los estudiantes podrán repetir las actividades durante una sesión, observar las respuestas del sistema y familiarizarse con el orden general de los recorridos. El simulador no reemplazará las clases ni garantizará por sí mismo una mejora del aprendizaje, pero proporcionará un espacio controlado para practicar las acciones incluidas en la propuesta.

Los docentes relacionados con las asignaturas de comercio exterior también serán beneficiarios directos. Dispondrán de una herramienta tecnológica que podrá complementar explicaciones, demostraciones y actividades formativas. El entorno virtual facilitará la presentación visual de una secuencia que, mediante recursos tradicionales, se explica principalmente con documentos, imágenes o exposiciones.

Los responsables del proyecto LACELI formarán parte de los beneficiarios directos porque contarán con un prototipo desarrollado a partir de sus requerimientos funcionales y pedagógicos. La aplicación podrá servir como base para revisar los recorridos propuestos, identificar posibles mejoras y planificar evaluaciones educativas posteriores. Además, la arquitectura modular facilitará que se analice la incorporación de nuevas actividades en futuras versiones.

1.11.2. Beneficiarios indirectos

La Universidad Católica de Cuenca será beneficiaria indirecta porque el proyecto ampliará su experiencia en el desarrollo y uso de tecnologías inmersivas. El simulador constituirá una referencia institucional sobre la integración de realidad virtual en actividades académicas y podrá favorecer la colaboración entre la Facultad de Ciencias

Económicas y Empresariales, la carrera de Ingeniería en Realidad Virtual y Videojuegos y el proyecto LACELI.

Los estudiantes de Ingeniería en Realidad Virtual y Videojuegos también podrán beneficiarse indirectamente. La arquitectura, los diagramas y la documentación técnica podrán utilizarse como material de consulta para comprender la organización de un proyecto de realidad virtual basado en etapas, tareas, eventos, seguimiento de manos y validaciones. Su reutilización deberá respetar la autoría del trabajo y las licencias de Unity, Meta XR y los recursos incorporados.

Otros docentes, investigadores y estudiantes interesados en tecnologías educativas podrán consultar la experiencia como antecedente para futuras propuestas. El proyecto podrá aportar una base para estudios posteriores sobre usabilidad, interacción mediante hand tracking y evaluación educativa de simuladores inmersivos. Cualquier ampliación deberá realizarse dentro de un nuevo alcance y con los instrumentos de validación correspondientes.

De forma indirecta, las futuras cohortes de estudiantes de la Facultad de Ciencias Económicas y Empresariales podrían utilizar versiones posteriores del simulador, siempre que la institución decida mantenerlo, actualizarlo e incorporarlo dentro de sus actividades. Este posible beneficio se plantea como una proyección y no como un resultado asegurado por la presente investigación.

1.12. **Metodología general**

La investigación combinó una metodología de investigación aplicada con una metodología de desarrollo basada en Kanban. La primera permitió estudiar el problema, revisar los requerimientos y definir los criterios necesarios para orientar la propuesta.

Kanban, en cambio, se utilizó para organizar las actividades de programación y controlar el avance de los componentes del simulador.

El trabajo se desarrolló entre abril y julio de 2026. Durante este periodo se realizaron revisiones funcionales periódicas para comprobar los avances, identificar dificultades y ajustar los requerimientos cuando era necesario. La investigación y el desarrollo mantuvieron una relación directa: la información obtenida durante el análisis orientó la construcción del software, y las pruebas realizadas sobre los incrementos permitieron refinar las interacciones.

El proceso se organizó de acuerdo con los objetivos específicos. Primero se analizaron los requerimientos funcionales, pedagógicos y de interacción. Después se diseñó la arquitectura modular. La siguiente fase correspondió a la implementación en Unity mediante Meta XR Interaction SDK. Finalmente, se comprobó el funcionamiento técnico, la usabilidad mecánica y el rendimiento inicial del simulador en Meta Quest 3S.

1.12.1. Metodología de investigación

El estudio utilizó un enfoque de investigación aplicada porque empleó conocimientos de programación, arquitectura de software y realidad virtual para desarrollar una solución destinada a una necesidad concreta. El resultado principal fue un prototipo tecnológico y no la formulación de una teoría nueva.

La investigación tuvo un alcance descriptivo. Se identificaron y documentaron los requerimientos del simulador, las funciones de sus módulos, las mecánicas de interacción y las condiciones necesarias para representar los recorridos simplificados de exportación e importación. También se describieron la arquitectura del software, las herramientas utilizadas y el comportamiento observado durante las pruebas.

El diseño fue no experimental, ya que no se manipularon variables para establecer relaciones de causa y efecto. La prueba piloto se orientó a observar el uso de las mecánicas y no a comparar grupos ni medir cambios en el aprendizaje. Por esta razón, los resultados se limitaron a la comprobación técnica y a una revisión inicial de usabilidad.

Las técnicas utilizadas fueron la revisión documental, el análisis de requerimientos, la observación directa y las pruebas funcionales. La revisión documental permitió estudiar antecedentes sobre realidad virtual educativa, aprendizaje procedimental, seguimiento de manos, locomoción y usabilidad. El análisis de requerimientos ayudó a transformar las actividades definidas para los recorridos educativos en componentes y funciones de software.

La observación directa se aplicó durante una prueba piloto con cinco estudiantes de distintas carreras. La selección no buscó medir conocimientos de comercio exterior, sino comprobar si participantes sin experiencia previa podían comprender las mecánicas principales.

Los comentarios verbales de los participantes se utilizaron como información complementaria para reconocer dificultades de interacción. No se aplicó un cuestionario estandarizado, una medición previa y posterior ni un grupo de control.

La comprobación técnica se realizó directamente en Meta Quest 3S. Se revisaron el inicio de la aplicación, la selección de los módulos, la carga de escenas y las transiciones entre etapas. También se comprobaron la locomoción gestual, el agarre con ambas manos, los sockets, las tareas, los tutoriales, las interfaces y el audio espacial. El rendimiento se observó mediante registros puntuales con un límite configurado de 72 FPS.

La metodología de investigación quedó organizada en cuatro momentos:

1. Análisis: revisión de antecedentes, estudio de los requerimientos y definición de las etapas, tareas e interacciones.

2. Diseño: elaboración de la arquitectura, distribución de módulos y definición de las relaciones entre componentes.

3. Implementación: programación de los sistemas en Unity e integración de Meta XR Interaction SDK.

4. Comprobación y refinamiento: ejecución de pruebas técnicas, observación de usuarios y aplicación de ajustes sobre las interacciones.

1.12.2. Metodología de desarrollo Kanban

Para organizar el desarrollo del simulador se utilizó Kanban. Esta metodología permitió representar visualmente las actividades, controlar su estado y trabajar de forma incremental. Las funciones no se implementaron todas al mismo tiempo se dividieron en tareas más pequeñas que avanzaban por el tablero según su progreso.

El tablero se estructuró mediante las siguientes columnas:

- 1.Pendiente: actividades identificadas que todavía no habían iniciado.

- 2.En proceso: tareas que se encontraban en análisis, diseño o programación.

- 3.En revisión: componentes implementados que requerían una comprobación funcional.

- 4.Finalizado: tareas verificadas e integradas con los demás sistemas.

Cada actividad se registró como una unidad de trabajo. Las tareas podían corresponder a una etapa del recorrido, un sistema general o una modificación puntual. Entre ellas se incluyeron la gestión de etapas, el sistema de tareas, las validaciones

mediante sockets, la locomoción, el agarre de objetos, las interfaces y los tutoriales contextuales.

El flujo comenzaba con la selección de una actividad pendiente. Después se analizaba su función, las dependencias necesarias y su relación con los demás módulos. La tarea pasaba a la columna En proceso durante la programación y configuración en Unity. Una vez obtenida una versión funcional avanzaba a En revisión donde se comprobaba su comportamiento. Si aparecían errores o cambios, regresaba al estado anterior; cuando cumplía con lo requerido se trasladaba a Finalizado.

Kanban facilitó incorporar cambios sin detener completamente el desarrollo. Cuando un requerimiento se modificaba la actividad se actualizaba o se generaba una nueva tarea. El resto de los componentes podía continuar su avance según el orden de prioridad y sus dependencias.

El desarrollo tuvo carácter incremental porque cada función terminada se integraba progresivamente con las anteriores. Primero se construyeron los sistemas necesarios para controlar el flujo general. Después se incorporaron las mecánicas y las funciones específicas de cada recorrido. Las pruebas se realizaron durante todo el proceso y no únicamente al terminar el prototipo.

Las revisiones funcionales se efectuaron de manera periódica. En ellas se verificaba si los incrementos cumplían con los requerimientos y si podían integrarse con el resto del simulador. Las observaciones generaban correcciones o nuevas actividades dentro del tablero Kanban. Este ciclo continuaba hasta que el componente alcanzaba el comportamiento esperado.

1.13. **Resultados esperados**

El proyecto dispone de:

1. Un prototipo funcional ejecutable en Meta Quest 3 y Meta Quest 3S.
2. Un recorrido educativo simplificado: exportación.
3. Una arquitectura modular documentada mediante diagramas.
4. Sistemas reutilizables de etapas, tareas, sockets, documentos, tutoriales y validación.
5. Interacción mediante hand tracking con cualquiera de las dos manos.
6. Locomoción gestual integrada con el flujo de la aplicación.
7. Tutoriales audiovisuales y contextuales.
8. Evidencias de funcionamiento y rendimiento en Meta Quest 3S.
9. Resultados de una prueba piloto inicial de usabilidad mecánica.
10. Manual del usuario y manual técnico para mantenimiento y ampliación.

Estos resultados permiten demostrar la viabilidad técnica de la propuesta. La comprobación de su impacto en el aprendizaje corresponde a una fase posterior.

1.14. **Relación con el protocolo de investigación**

El protocolo aprobado planteó el diseño e implementación de un simulador educativo en realidad virtual basado en arquitectura de software, mecánicas interactivas y patrones de diseño para el laboratorio de comercio exterior y logística internacional.

El desarrollo mantiene correspondencia con el tema aprobado. Se implementó un simulador en Unity, se integró Meta XR, se organizó la lógica mediante programación orientada a objetos y se construyeron sistemas de interacción, etapas, tareas, documentos y validación.

La metodología incremental declarada en el anteproyecto se aplicó mediante entregas progresivas y reuniones semanales. Los requerimientos fueron revisados por el responsable del proyecto LACELI y el producto evolucionó a partir de sus observaciones.

Existen dos precisiones respecto del planteamiento inicial. Primero, el sistema no incorpora persistencia porque esta función dejó de ser requerida; el progreso se conserva únicamente durante la ejecución. Segundo, la validación realizada fue técnica y de usabilidad mecánica. No se efectuó una comparación de aprendizaje frente a metodologías tradicionales.

Estas diferencias no modifican el núcleo de la propuesta, pero deben reflejarse en los objetivos y conclusiones para conservar coherencia entre lo planificado, lo desarrollado y lo evaluado.

Capítulo II

2. Marco Teórico

2.1. Antecedentes investigativos / Estado del arte

La realidad virtual ha ganado espacio dentro de la educación porque puede construir escenarios tridimensionales, incorporar actividades interactivas y acercar al usuario a situaciones que no siempre pueden reproducirse en un aula. Su uso va más allá de presentar información por medio de un visor. Las propuestas con mayor desarrollo combinan desplazamiento, instrucciones, manipulación de objetos y respuestas del sistema, de modo que el estudiante participe dentro de la experiencia en lugar de limitarse a observarla.

Radianti et al. (2020) revisaron distintas aplicaciones de realidad virtual inmersiva empleadas en educación superior. Entre los campos encontrados estuvieron medicina, ingeniería y ciencias naturales, además de otras áreas de formación. Los estudios incluían laboratorios, recorridos virtuales, simulaciones y actividades prácticas. La revisión también detectó que varios trabajos dedicaban más atención al componente tecnológico que al diseño pedagógico y a la evaluación del aprendizaje. Esto indica que construir un entorno inmersivo funcional no basta para demostrar su utilidad educativa. También hace falta definir las actividades, orientar al usuario y establecer cómo se evaluará la experiencia.

Una revisión que abarcó diez años de investigaciones fue realizada por Di Natale et al. (2020). El estudio examinó experiencias de realidad virtual inmersiva en diferentes niveles educativos y encontró que la inmersión y la interacción podían despertar interés, además de favorecer la participación. Los resultados cambiaban según la forma en que cada aplicación había sido diseñada. Algunas propuestas se concentraban en recorridos

visuales, pero otras pedían al usuario manipular elementos, resolver problemas o practicar procedimientos específicos.

Pellas et al. (2021) obtuvieron resultados parecidos al analizar la literatura científica sobre realidad virtual inmersiva en educación. Las aplicaciones revisadas tenían distintos mecanismos de interacción y diferentes niveles de participación. Tampoco compartían un único propósito. Se encontraron simulaciones, visitas inmersivas, laboratorios virtuales y experiencias creadas para desarrollar habilidades. Entre sus dificultades aparecieron el diseño de interfaces, la usabilidad y el proceso de adaptación inicial al dispositivo.

Luo et al. (2021) estudiaron investigaciones sobre realidad virtual desarrolladas entre 2000 y 2019 en educación escolar y superior. Esta tecnología había sido aplicada tanto para trabajar conocimientos conceptuales como para apoyar habilidades prácticas. Sin embargo, los dispositivos, la duración de las intervenciones, las metodologías y las formas de evaluación presentaban diferencias importantes. Por esta razón, los resultados de una aplicación no pueden trasladarse directamente a cualquier otra experiencia de realidad virtual.

Una revisión más reciente fue presentada por Stracke et al. (2025). Los autores encontraron que la realidad virtual inmersiva todavía no forma parte habitual de todos los programas universitarios. Muchas experiencias continúan ejecutándose en laboratorios o durante actividades específicas, en lugar de integrarse de manera permanente en las asignaturas. También observaron que ciertos estudios no detallaban suficientemente el entorno utilizado, los dispositivos o las condiciones de aplicación. Esa falta de información dificulta repetir y comprobar los resultados. Una propuesta tecnológica

debería registrar la versión del motor, el visor, los sistemas de interacción y las condiciones en que se efectuarán sus pruebas.

Los estudios sobre resultados educativos ofrecen conclusiones favorables, aunque no totalmente iguales. Wu et al. (2020) encontraron que la realidad virtual inmersiva mediante visores podía influir de manera positiva en el desempeño de aprendizaje. El efecto variaba de acuerdo con la actividad y con las características de los participantes. La tecnología, en este caso, actuaba como una herramienta de apoyo y no como el único elemento responsable de los resultados.

El metaanálisis de Çoban et al. (2022) reunió 48 estudios con una muestra total de 3179 estudiantes de educación escolar y superior. Los resultados mostraron un efecto general positivo, pero de magnitud pequeña. También se presentaron diferencias según el nivel educativo, el área de formación y el recurso empleado para establecer la comparación. Esto obliga a valorar cada experiencia dentro de su propio contexto, sin asumir beneficios generales antes de evaluarla.

Yu y Xu (2022) examinaron 43 estudios empíricos relacionados con realidad virtual. Sus resultados incluían efectos sobre habilidades intelectuales, actitudes, estrategias cognitivas y destrezas motoras. No todos los niveles educativos presentaban el mismo comportamiento. Los autores recomendaron considerar la motivación, la satisfacción y el interés del usuario, aunque mantuvieron la necesidad de evaluar cada aplicación con base en sus objetivos concretos.

Conrad et al. (2024) analizaron la efectividad de la realidad virtual inmersiva en procesos educativos y de formación. Los entornos donde el usuario manipula, participa y aplica de forma práctica lo aprendido pueden aprovechar mejor las características del medio. Aun así, la realidad virtual sigue siendo una herramienta. El aprendizaje depende

del diseño educativo y de cómo se aplica la metodología. Para una propuesta centrada en actividades secuenciales, las mecánicas deberían relacionarse con acciones que cumplan una finalidad definida y no incluirse únicamente para aumentar la cantidad de objetos interactivos.

Los conceptos de presencia y agencia también han sido utilizados para estudiar el aprendizaje inmersivo. Makransky y Petersen (2021) formularon el modelo cognitivo-afectivo del aprendizaje inmersivo, conocido como CAMIL. Dentro de este modelo, la presencia describe la percepción de encontrarse en el entorno virtual. La agencia aparece cuando el usuario comprende que sus acciones producen determinados resultados. Otros factores incluidos son la motivación, el interés, la corporeidad, la autoeficacia, la autorregulación y la carga cognitiva.

Sentirse dentro del entorno puede aumentar la participación, pero no asegura por sí mismo un mejor aprendizaje. Makransky y Mayer (2022) compararon una visita virtual inmersiva con una presentación mostrada en pantalla. La primera produjo niveles mayores de presencia, disfrute e interés. También se registraron resultados favorables en evaluaciones inmediatas y posteriores. Los investigadores relacionaron los hallazgos con elementos cognitivos y afectivos, lo que refleja la necesidad de acompañar la inmersión con una actividad bien organizada y un objetivo educativo comprensible.

La cantidad de información presentada constituye otro problema que debe considerarse. Un mismo escenario puede reunir botones, objetos, sonidos, textos y otros estímulos visuales. Albus et al. (2021) analizaron el empleo de señales en realidad virtual y observaron que destacar la información relevante podía favorecer la comprensión y reducir la carga cognitiva innecesaria. Estas señales ayudaban a dirigir la atención hacia los elementos relacionados con la tarea. El antecedente ofrece una base para planificar

animaciones, instrucciones, indicadores o sonidos, procurando que tales ayudas no llenen el escenario de información.

Hamilton et al. (2021) realizaron una revisión de estudios cuantitativos sobre realidad virtual inmersiva utilizada como herramienta pedagógica. Aunque encontraron resultados prometedores, también identificaron problemas metodológicos. Entre ellos se encontraban muestras pequeñas, instrumentos distintos y diseños experimentales que no podían compararse fácilmente. Los autores advierten que una experiencia educativa necesita instrumentos y resultados claramente definidos. Una revisión inicial de usabilidad, por tanto, no debería presentarse como una demostración de aprendizaje.

2.2. Bases teóricas

2.2.1. Aprendizaje inmersivo

La realidad virtual inmersiva puede favorecer ciertos resultados de aprendizaje, pero su efecto no es uniforme en todos los niveles, asignaturas o formas de evaluación. El metaanálisis de Çoban et al. (2022) identificó un efecto general positivo de magnitud pequeña y diferencias relacionadas con el nivel educativo, el campo de estudio y el recurso utilizado como comparación. Por esta razón, la inmersión no debe presentarse como garantía automática de aprendizaje.

El aprendizaje inmersivo se produce cuando el usuario participa en una experiencia digital que genera sensación de presencia y capacidad de actuación. Makransky y Petersen (2021) proponen el modelo cognitivo-afectivo del aprendizaje inmersivo, conocido como CAMIL. Este modelo identifica la presencia y la agencia como posibilidades psicológicas centrales de la realidad virtual. La presencia se relaciona con

la percepción de encontrarse dentro del entorno, la agencia expresa la sensación de que las acciones propias producen consecuencias.

CAMIL también reconoce que la motivación, el interés, la autoeficacia, la corporeidad, la carga cognitiva y la autorregulación pueden modificar los resultados. Por esta razón, la experiencia incluye tutoriales y señales que reducen la incertidumbre inicial.

2.2.2. Aprendizaje procedimental y secuenciación

El conocimiento procedimental se relaciona con saber cómo ejecutar una serie de acciones. No consiste solamente en recordar conceptos, sino en reconocer el orden, seleccionar elementos y responder a condiciones del entorno. Jongbloed et al. (2024) señalan que la realidad virtual resulta especialmente adecuada para ensayar procedimientos, debido a que combina representación espacial, repetición y ejecución directa. Los efectos de la realidad virtual pueden aparecer en habilidades intelectuales, estrategias cognitivas, actitudes y destrezas motoras. Sin embargo, el resultado depende de las características de los participantes y de la actividad educativa desarrollada (Yu & Xu, 2022).

El simulador no reproduce la complejidad normativa de una operación comercial. Su función consiste en presentar un recorrido pedagógico definido por el área solicitante y traducirlo a acciones observables. La fidelidad buscada es funcional: que el estudiante identifique una secuencia y participe en ella, no que ejecute una gestión aduanera válida.

2.2.3. Presencia, agencia y corporeidad

La presencia depende de la coherencia entre estímulos visuales, movimientos y respuesta del entorno. La agencia requiere que el resultado de una acción sea comprensible. Makransky y Petersen (2021) indican que la inmersión puede favorecer el interés, pero una carga cognitiva excesiva puede reducir sus beneficios.

El hand tracking contribuye a la corporeidad al representar las manos del usuario dentro del escenario. La interacción directa evita traducir cada movimiento a botones de un mando; aun así, exige zonas de agarre accesibles, poses apropiadas y objetos situados dentro del campo de seguimiento. Las mejoras realizadas después de la prueba piloto responden a este equilibrio entre naturalidad y precisión.

2.2.4. Retroalimentación multimodal

Una experiencia inmersiva puede presentar varios estímulos de manera simultánea, lo que dificulta identificar la información relevante. Las señales visuales ayudan a dirigir la atención y pueden reducir la carga cognitiva innecesaria cuando destacan elementos relacionados directamente con la tarea (Albus et al., 2021).

La variedad de señales evita que toda la orientación dependa de texto. También permite que la retroalimentación permanezca vinculada con el lugar de la acción. El AudioSource tridimensional se ubica en la siguiente zona y aumenta su intensidad percibida conforme el usuario se aproxima. Esta decisión apoya la navegación sin introducir flechas permanentes que saturen la escena.

2.2.5. Usabilidad en realidad virtual

La usabilidad comprende la posibilidad de completar acciones con claridad, esfuerzo razonable y pocos errores. En realidad virtual intervienen factores adicionales: adaptación al visor, estabilidad de seguimiento, alcance físico, orientación espacial y selección de objetos.

Hameed et al. (2023) muestran que la interacción con manos no siempre supera a los controladores en rapidez o precisión. Por ello, el requisito de utilizar hand tracking debe acompañarse de mecanismos compensatorios

La prueba piloto con cinco estudiantes de distintas carreras tuvo carácter exploratorio. Mediante observación directa y comentarios verbales se identificaron dificultades mecánicas. No se midió aprendizaje ni se aplicó una escala cuantitativa; sus resultados se utilizaron para refinar el producto.

2.3. **Bases metodológicas**

El proyecto siguió un enfoque de investigación aplicada, debido a que respondió a una necesidad concreta mediante la construcción de un producto tecnológico. El trabajo tuvo alcance descriptivo en la identificación de requerimientos y carácter tecnológico en el diseño, implementación y prueba del simulador.

La metodología de desarrollo fue progresiva. Entre abril y finales de julio de 2026 se realizaron reuniones semanales con Adrián Alvarado, responsable del proyecto LACELI. En cada sesión se presentaban avances, se revisaba la respuesta de las interacciones y se solicitaban ajustes cuando el comportamiento no coincidía con las necesidades funcionales o pedagógicas.

El ciclo de trabajo se organizó de la siguiente manera:

1. Recepción o actualización de requerimientos.
2. Análisis de la función solicitada.
3. Diseño del componente o modificación del sistema.
4. Implementación en Unity.
5. Prueba técnica en el entorno de desarrollo.
6. Presentación del incremento en la reunión semanal.
7. Aprobación o solicitud de cambios.

8. Integración del componente aceptado con los demás módulos.

Este procedimiento explica las diferencias entre el documento inicial y el producto final. Los requerimientos evolucionaron durante el desarrollo y fueron ajustados mediante revisión directa. No corresponde denominar el proceso Scrum, porque no se utilizaron formalmente sus roles, eventos y artefactos. La descripción más precisa es desarrollo incremental con validación funcional semanal.

La validación comprendió dos niveles. El primero fue técnico: se comprobó la ejecución en Meta Quest 3S, el cambio de escenas, la navegación, la respuesta de tareas, la selección de objetos y el rendimiento. El visor fue configurado con un límite de 72 FPS. Las capturas registraron valores aproximados entre 52 y 72 FPS, con varios momentos cercanos al límite. Estos datos constituyen evidencia puntual y no un promedio estadístico.

El segundo nivel fue una prueba piloto de usabilidad con cinco estudiantes de diferentes carreras. El propósito no fue evaluar conocimientos de comercio exterior, sino determinar si las mecánicas podían comprenderse y utilizarse sin experiencia previa. La información se obtuvo mediante observación directa y comentarios. A partir de ella se integró la locomoción gestual de Meta XR, se mejoró el alcance de los objetos agarrables y se añadieron poses de mano.

Las limitaciones metodológicas deben quedar explícitas. No existió un grupo de control, una medición previa y posterior ni un instrumento cuantitativo de usabilidad. Por tanto, los resultados no permiten afirmar una mejora del aprendizaje ni generalizar la aceptación del sistema. Hamilton et al. (2021) advierten precisamente que la evaluación de experiencias inmersivas requiere instrumentos y resultados bien definidos. En esta fase, la evidencia demuestra viabilidad técnica y aporta observaciones iniciales de uso.

2.4. Bases tecnológicas

2.4.1. Unity y C#

El simulador fue implementado en Unity 6000.3.5f2. El motor concentra el manejo de escenas, objetos, animaciones, físicas, audio, interfaces y compilación para dispositivos de realidad virtual. La lógica se programó en C# mediante componentes asociados con objetos de escena.

Unity utiliza MonoBehaviour como base para scripts que participan en el ciclo de ejecución. Esta estructura favoreció la separación por responsabilidades: los controladores de etapa administran la secuencia; los validadores comprueban objetos; los gestores de tareas contabilizan acciones; y los componentes de interfaz muestran instrucciones y progreso.

2.4.2. Meta XR Interaction SDK

La integración inmersiva utiliza Meta XR Interaction SDK 85.0.0, acompañado por Meta XR Core SDK y los componentes necesarios del paquete All-in-One 85.0.0. El SDK proporciona el seguimiento de manos, los interactores, los objetos agarrables, los sockets y los eventos de locomoción.

Es importante delimitar la autoría técnica. El reconocimiento de manos, el gesto base de locomoción y clases como SnapInteractable pertenecen al SDK de Meta. El aporte del proyecto fue configurarlos, conectarlos con la lógica principal y adaptarlos a las actividades del simulador. También se animó en Unity un prefab de mano suministrado por Meta para construir demostraciones contextuales.

2.4.3. Meta Quest 3S

La aplicación fue probada en Meta Quest 3S como dispositivo autónomo. Esta condición impone restricciones distintas a las de un computador: memoria limitada,

presupuesto de procesamiento y necesidad de mantener una frecuencia visual estable. El límite de 72 FPS se estableció como referencia operativa.

Caserman et al. (2021) explican que la inestabilidad visual, el movimiento y la latencia pueden relacionarse con malestar en experiencias inmersivas. La teletransportación se utilizó como mecanismo principal de locomoción para reducir desplazamientos continuos y ofrecer control directo sobre el destino.

2.4.4. Interacción mediante hand tracking

El seguimiento de manos ofrece una alternativa a los controladores físicos y puede hacer que la manipulación parezca más directa. Aun así, las tareas de agarrar y colocar objetos dependen de la precisión del sistema, el diseño de los elementos interactivos y la forma en que la aplicación representa el contacto (Masurovsky et al., 2020).

La experiencia final prioriza el uso de manos sin controladores físicos. El gesto de locomoción puede ejecutarse con cualquiera de las dos manos. El usuario mantiene la mano curvada y realiza un gesto de clic aproximando el pulgar hacia la articulación central del índice; después orienta el destino y confirma el desplazamiento con un nuevo clic.

Buckingham (2021) advierte que el seguimiento de manos ofrece naturalidad, pero debe considerar oclusión, campo de visión y precisión. Esta observación fundamenta la ampliación del alcance de agarre, la configuración bilateral de poses y la incorporación de tutoriales.

2.5. Arquitecturas, modelos y estándares

2.5.1. Organización modular

El repositorio contiene 92 scripts distribuidos en módulos como Core, Data, Documents, Interaction, PlayerMovement, Scoring, SocketSystem, Stages, Subtitle,

TaskSystem, Tutorial, UI y Validation. Esta separación facilita localizar responsabilidades y reduce la concentración de lógica en un único controlador.

La arquitectura es modular y orientada a componentes. No constituye una arquitectura por capas estricta ni utiliza microservicios. Unity organiza el comportamiento mediante objetos y componentes; sobre esa base, el proyecto agrupa controladores y gestores según su función.

2.5.2. Programación orientada a objetos

La programación orientada a objetos se evidencia en clases, encapsulación, herencia y polimorfismo. BaseStageController define el comportamiento común de una etapa, mientras que controladores concretos de exportación e importación sobrescriben métodos como OnStartStage, OnMiddleStage y OnFinishStage.

La clase base evita repetir la secuencia de inicio, espera, teletransporte y finalización. Las subclases agregan la lógica particular. Esta estructura se aproxima al patrón Template Method porque el algoritmo general permanece en la clase base y ciertos pasos se especializan mediante métodos sobrescritos.

2.5.3. Gestión de etapas

StageManager mantiene la secuencia de navegación y decide si el siguiente paso implica cargar una escena o teletransportar al usuario dentro de la escena actual. Se implementa como Singleton y utiliza DontDestroyOnLoad para conservar una instancia entre escenas.

GameFlowManager escucha la finalización de una etapa y solicita el avance. SceneStageGroup agrupa controladores cuando varias actividades ocurren en la misma escena. El resultado es una máquina secuencial de progreso; sin embargo, no se debe

afirmar que existe una implementación formal del patrón State, porque no se encontraron clases de estado intercambiables con una interfaz común.

2.5.4. Eventos y Observer

Los módulos se comunican mediante eventos de C# y UnityEvent. Por ejemplo, TaskNotificationUI se suscribe a cambios de TaskManager, mientras GameFlowManager escucha la finalización del controlador activo. Este mecanismo se relaciona con Observer: un componente publica un cambio y los interesados reaccionan sin que el emisor deba controlar toda la respuesta.

La combinación permite configurar acciones desde el Inspector y conservar conexiones explícitas en código.

2.5.5. Validación mediante sockets y tareas

SocketValidator hereda de SnapInteractable y verifica si el tipo del objeto coincide con el esperado. La respuesta se ejecuta mediante eventos de objeto correcto o incorrecto. Esta lógica separa la detección de la consecuencia visual o auditiva.

TaskManager administra una colección de tareas configuradas desde el Inspector. Cada tarea puede ejecutar una acción inmediata, una acción posterior y un evento al completar el conjunto. TaskNotificationUI presenta el progreso. Esta separación permite reutilizar el sistema en distintas etapas sin escribir una interfaz específica para cada una.

2.6. Herramientas de desarrollo

2.6.1. Unity 6000.3.5f2

Unity 6000.3.5f2 constituyó el entorno principal de construcción. Desde el editor se organizaron escenas, prefabs, animaciones, interfaces, eventos y referencias. C# se utilizó para controlar el flujo y conectar los componentes.

2.6.2. Meta XR Interaction SDK 85.0.0

Meta XR Interaction SDK 85.0.0 aportó las bases de interacción para manos, agarre, sockets y locomoción. La elección se relaciona con el dispositivo objetivo y con el requerimiento de prescindir de controladores. El proyecto integró estos recursos sin atribuirse la creación de los algoritmos de seguimiento.

2.6.3. Meta Quest 3S

Las pruebas se realizaron en Meta Quest 3S. Las herramientas de monitoreo del visor permitieron observar FPS y carga durante la ejecución. Las capturas constituyen evidencia de momentos específicos; para futuras evaluaciones sería recomendable registrar sesiones completas y calcular promedio, percentiles y caídas sostenidas.

2.6.4. Diagramas de clase.

La organización de clases, métodos y módulos fue documentada mediante diagramas. Estos recursos serán especialmente útiles en el capítulo técnico, donde podrá explicarse el flujo entre StageManager, controladores, tareas, sockets, interfaces y sistemas tutoriales.

2.7. Marco conceptual

2.7.1. Realidad virtual inmersiva.

Entorno digital tridimensional presentado mediante un visor que responde a los movimientos del usuario y busca producir sensación de presencia.

2.7.2. Simulador educativo.

Aplicación que representa actividades o situaciones con propósito formativo. En este proyecto, la representación es simplificada y se concentra en la secuencia de acciones.

2.7.3. Head-mounted display (HMD).

Dispositivo colocado en la cabeza que presenta imágenes estereoscópicas y registra orientación o posición.

2.7.4. Hand tracking.

Seguimiento de la posición y configuración de las manos sin depender de controladores físicos.

2.7.5. Interacción natural.

Forma de entrada que intenta aproximarse a acciones cotidianas, como señalar, agarrar o realizar gestos.

2.7.6. Grabbable object.

Objeto virtual configurado para ser seleccionado y sostenido mediante un interactor.

2.7.7. Hand pose.

Configuración visual de la mano virtual asociada con una forma de agarre.

2.7.8. Socket.

Zona que recibe un objeto y puede validar si corresponde con el tipo esperado.

2.7.9. Teletransportación.

Técnica de locomoción en la que el usuario selecciona un punto y el sistema cambia su posición sin representar todo el recorrido continuo.

2.7.10. Retroalimentación multimodal.

Respuesta que combina recursos visuales, auditivos o funcionales para comunicar el resultado de una acción.

2.7.11. Audio espacial.

Sonido cuya dirección e intensidad percibida dependen de la posición de la fuente y del usuario.

2.7.12. Persistencia

Conservación de datos después de cerrar una aplicación. Explorador Global no incorpora esta capacidad.

2.8. Relación entre la teoría y la propuesta tecnológica

Tabla 1 Correspondencia entre los fundamentos teóricos y las decisiones aplicadas en Explorador Global

Fundamento	Decisión aplicada	Evidencia en el sistema
Aprendizaje procedimental	Dividir los recorridos en acciones ordenadas	Controladores de exportación e importación
Presencia y agencia	Permitir que el usuario ejecute y valide acciones	Tareas, sockets y eventos
Segmentación	Organizar la experiencia por etapas	StageManager, SceneStageGroup y BaseStageController
Retroalimentación multimodal	Combinar demostración, sonido y respuesta del sistema	Mano tutorial, AudioSource 3D y validadores
Interacción natural	Priorizar el uso de manos sin controladores	Meta XR Interaction SDK y hand tracking
Usabilidad	Ajustar alcance y poses después de observar usuarios	Objetos agarrables y poses bilaterales

Reducción de desplazamiento continuo	Integrar gestual	teletransportación	Sistema principal de locomoción
Modularidad	Separar responsabilidades por sistemas		Organización del repositorio en módulos
Polimorfismo	Reutilizar un flujo común y especializar etapas		Herencia de BaseStageController
Comunicación desacoplada	Notificar cambios mediante eventos		Eventos C# y UnityEvent
Rendimiento en visor autónomo	Establecer límite y observar FPS		Pruebas en Meta Quest 3S a 72 FPS
Alcance sin persistencia	Conservar progreso solo durante la sesión		Clase estática ProgressData

Nota: Elaboración propia.

La relación de la tabla 1 evidencia que las decisiones principales no se adoptaron únicamente por disponibilidad tecnológica. La división por etapas responde al carácter procedimental; los tutoriales reducen la incertidumbre; el audio espacial orienta la atención; y los eventos permiten conectar sistemas sin concentrar todas las responsabilidades.

La validación piloto también cumple una función de enlace entre teoría y desarrollo. La literatura advierte que el hand tracking puede presentar problemas de precisión y carga de uso. La observación de los estudiantes permitió reconocer

dificultades equivalentes y transformarlas en cambios concretos: locomoción gestual integrada, mayor alcance de agarre y poses para ambas manos.

Finalmente, el simulador debe entenderse como una solución tecnológica de apoyo. Su aporte se encuentra en la arquitectura, la integración de Meta XR, la construcción de sistemas reutilizables y la adaptación de las interacciones a usuarios sin experiencia previa. La evaluación del aprendizaje de comercio exterior requeriría un estudio posterior con instrumentos, muestra y diseño comparativo específicos.

Capítulo III

3. Manual de Usuario

3.1. Presentación de la propuesta y perfil del usuario

3.1.1. Descripción general de Explorador Global

Explorador Global es una simulación educativa de realidad virtual que representa actividades vinculadas con procesos de exportación e importación mediante escenarios tridimensionales, documentos, paneles informativos y objetos logísticos. La experiencia está dirigida principalmente a estudiantes universitarios de la Carrera de Negocios Internacionales de la Facultad de Ciencias Jurídicas de la Universidad Católica de Cuenca.

La aplicación organiza las actividades de forma secuencial y permite que el usuario consulte instrucciones, seleccione opciones, interactúe con elementos del entorno y avance entre etapas. La representación tiene un propósito académico y no sustituye los procedimientos oficiales, la normativa vigente ni la práctica profesional del comercio exterior.

3.1.2. Perfil del usuario

El usuario principal de Explorador Global corresponde a estudiantes de la Carrera de Negocios Internacionales de la Facultad de Ciencias Jurídicas de la Universidad Católica de Cuenca, quienes utilizan la simulación como recurso de apoyo para interactuar con actividades relacionadas con procesos de exportación e importación.

No se requiere experiencia avanzada previa con dispositivos de realidad virtual. El usuario puede presentar un nivel básico o incluso no haber utilizado anteriormente un visor, debido a que la aplicación incorpora un tutorial inicial y ayudas contextuales

destinadas a explicar las acciones esenciales de navegación e interacción. Para utilizar adecuadamente la experiencia, se espera que el estudiante pueda seguir instrucciones breves, identificar información presentada en paneles y ejecutar las acciones indicadas durante cada etapa.

3.1.3. Requisitos mínimos y condiciones de uso

Para utilizar la versión documentada de Explorador Global se requieren las siguientes condiciones:

- Un visor Meta Quest 3 con la aplicación Explorador Global instalada.
- Un espacio físico despejado que permita mover los brazos con seguridad durante la interacción.
- Iluminación ambiental suficiente para favorecer el reconocimiento de las manos por las cámaras del visor.
- Las cámaras externas del visor deben permanecer descubiertas durante la sesión.
- Las manos deben mantenerse, cuando sea posible, dentro del campo de seguimiento del visor para conservar una interacción estable.

Antes de comenzar, el usuario debe comprobar que no existan obstáculos, muebles u otras personas dentro del área inmediata de movimiento. Aunque el desplazamiento virtual se realiza principalmente mediante teletransportación, las tareas de selección y manipulación requieren movimientos reales de las manos y los brazos.

3.2. Acceso e inicio de la experiencia

3.2.1. Preparación del espacio y del visor

El usuario debe colocarse el visor dentro de un área libre de obstáculos y verificar que las cámaras dispongan de iluminación suficiente. Al iniciar el seguimiento, ambas manos deben situarse frente al visor hasta que aparezca su representación virtual. Si una

mano deja de reconocerse temporalmente, debe volver a colocarse dentro del campo de visión de las cámaras hasta recuperar el seguimiento.

3.2.2. Inicio de la aplicación

Una vez preparada el área de uso, el usuario ejecuta Explorador Global desde el visor. La experiencia presenta el entorno inicial y activa el sistema de seguimiento de manos. Antes de continuar con los módulos de comercio exterior, se muestra una guía inicial destinada a familiarizar al usuario con las acciones básicas de interacción.

3.2.3. Tutorial inicial

El tutorial introduce los gestos necesarios para desenvolverse dentro de la aplicación. Durante esta guía, el usuario observa la demostración, reproduce la acción indicada y comprueba la respuesta del sistema antes de continuar. La guía se concentra en el desplazamiento, la selección y la manipulación mediante seguimiento de manos.

Presentación del tutorial inicial de interacción



Nota. Captura de pantalla del tutorial inicial de Explorador Global, utilizado para introducir las acciones básicas antes de comenzar el recorrido.

La captura muestra la guía presentada al inicio de la experiencia. Su función es preparar al usuario antes de las actividades de exportación o importación y reducir la necesidad de aprender los controles durante una tarea del recorrido.

3.2.4. Selección del módulo e inicio del recorrido

Después del tutorial, el usuario accede a la selección del recorrido y elige entre los módulos disponibles de exportación e importación. La elección determina la secuencia de actividades, documentos y escenarios que se presentarán durante la sesión. Ambos

recorridos comparten los mismos principios de navegación e interacción descritos en este manual.

3.3. Navegación y controles

3.3.1. Seguimiento de manos

La interacción principal utiliza seguimiento de manos. Las manos virtuales representan la posición y orientación de las manos reales y permiten ejecutar las acciones comunes sin depender de controladores físicos. La aplicación admite el uso de la mano izquierda y de la derecha en las interacciones documentadas.

Para mantener una respuesta estable, el usuario debe evitar ocultar las manos durante periodos prolongados y procurar que los gestos se realicen frente al visor. Cuando el seguimiento se pierde, la acción debe repetirse después de que la mano vuelva a ser reconocida.

3.3.2. Locomoción gestual y teletransportación

El desplazamiento principal se realiza mediante locomoción gestual y teletransportación. El usuario adopta la postura mostrada por el tutorial, activa el gesto correspondiente, orienta el destino y confirma el movimiento. La aplicación traslada al usuario al punto indicado sin representar un desplazamiento continuo entre ambas posiciones.

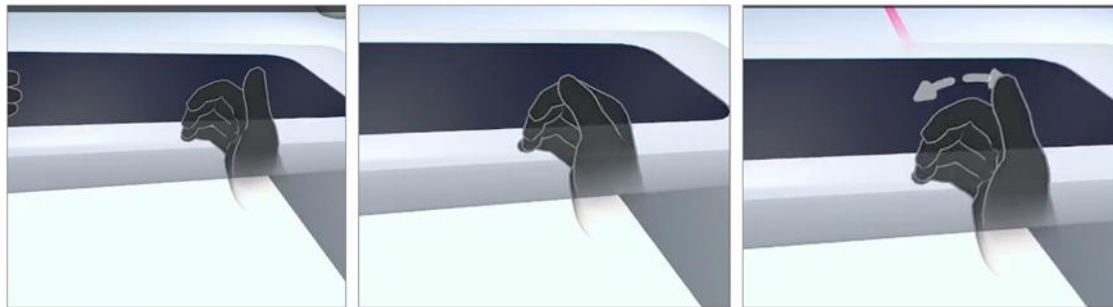
1. Adopte la postura gestual indicada por el tutorial y mantenga la mano dentro del campo de seguimiento.
2. Active el gesto acercando el pulgar hacia la zona indicada del dedo índice.
3. Oriente el destino hacia una posición válida del escenario.
4. Repita el gesto de confirmación para ejecutar la teletransportación.

La Ilustración 4 resume los tres momentos iniciales necesarios para activar la locomoción gestual antes de seleccionar el destino.

Ilustración

2

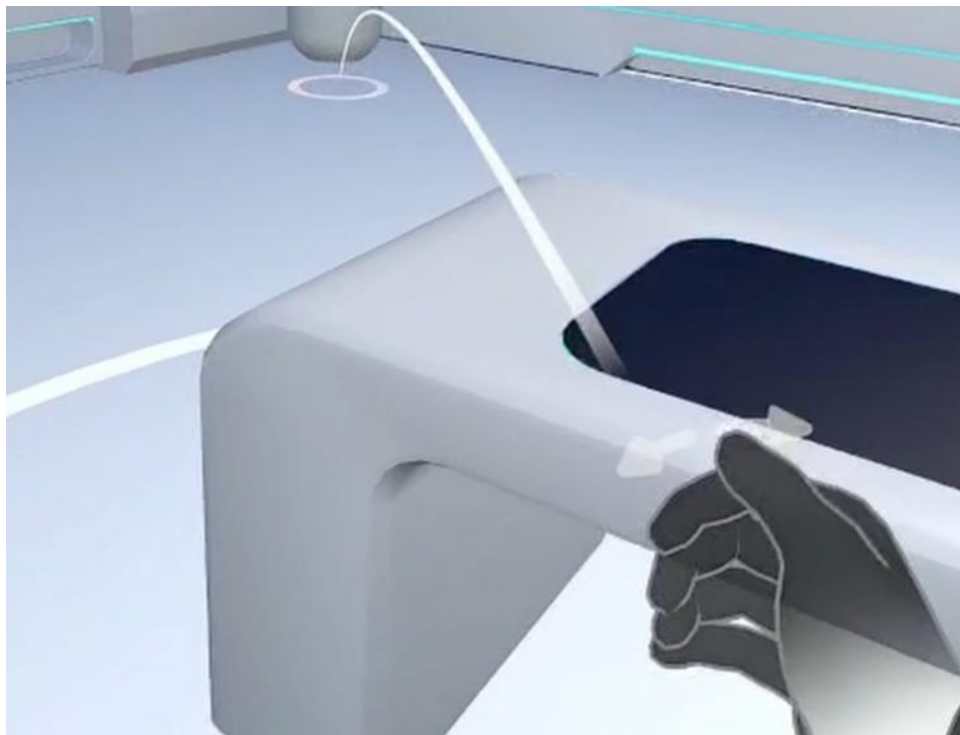
Secuencia de activación de la locomoción gestual



Nota. Capturas de pantalla de Explorador Global. (a) postura inicial de la mano; (b) aproximación del pulgar hacia el dedo índice; (c) activación del gesto de locomoción.

La secuencia permite observar que el desplazamiento no se activa con un único movimiento aislado. Primero se adopta la postura de la mano, después se aproxima el pulgar al índice y, finalmente, el sistema reconoce el gesto de locomoción. Mantener esta secuencia facilita que el usuario identifique el momento en que puede orientar el destino dentro del escenario.

Una vez activada la locomoción, el usuario orienta el destino y confirma el punto seleccionado mediante el mismo sistema gestual, como se muestra en la Ilustración 5.

Ilustración**3***Confirmación de un punto de teletransportación*

Nota. Captura de pantalla de la locomoción gestual utilizada para seleccionar y confirmar un punto de teletransportación.

La imagen documenta el momento en que el usuario confirma el desplazamiento con la mano. El sistema de teletransportación permite cambiar de posición dentro del escenario manteniendo el control directo sobre el destino.

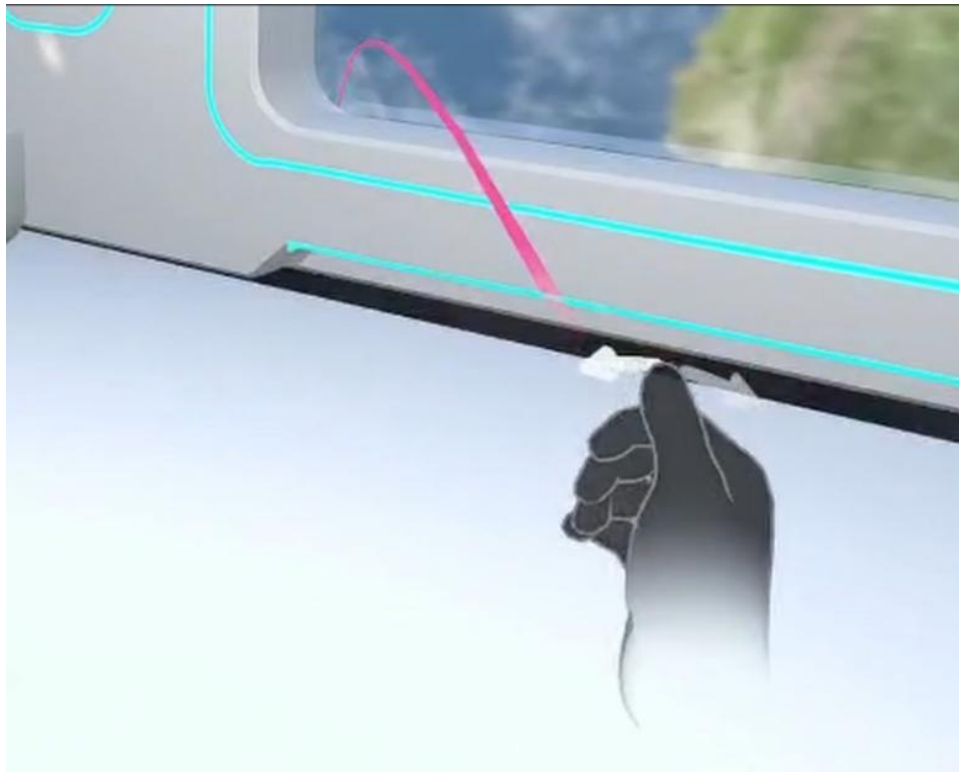
3.3.3. Giro y desplazamiento por pasos

Además de la teletransportación, el sistema documentado permite realizar ajustes de orientación y pequeños desplazamientos. El giro se ejecuta mediante un movimiento lateral de la mano mientras se mantiene la postura de locomoción. El desplazamiento por pasos se realiza con el movimiento del pulgar hacia delante o hacia atrás, según la dirección requerida.

Ilustración

4

Control gestual de giro durante la navegación



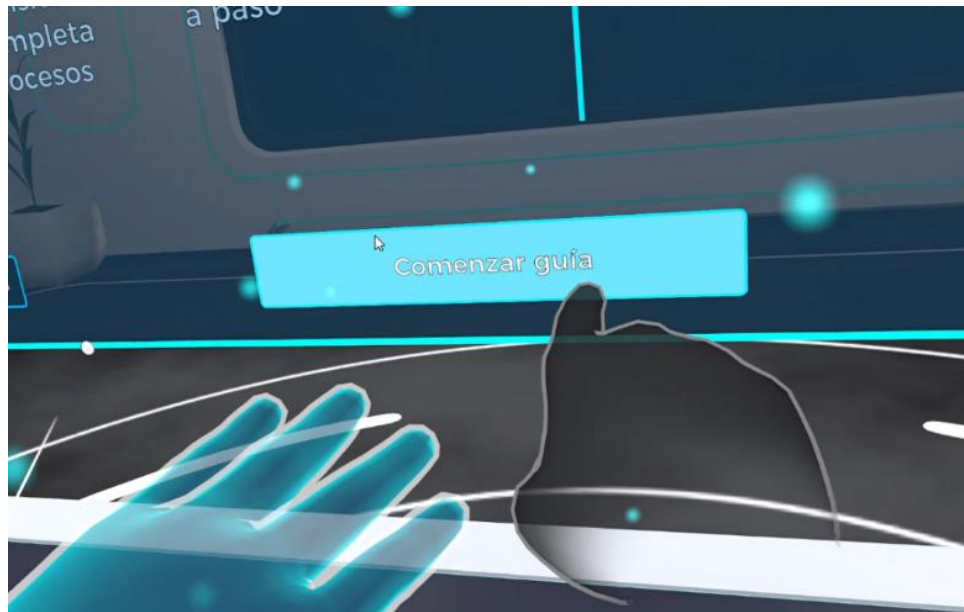
Nota. Captura de pantalla del gesto utilizado para ajustar la orientación del usuario dentro del escenario.

La captura muestra el control de giro integrado al sistema de locomoción. Esta acción permite reorientar la vista sin exigir un desplazamiento físico amplio dentro del área real de uso.

3.3.4. Selección de botones y elementos de interfaz

Los paneles de la aplicación incluyen botones y elementos interactivos para comenzar etapas, seleccionar opciones y confirmar acciones. Para utilizarlos, el usuario orienta la mano hacia el elemento, comprueba que el indicador de selección se encuentre sobre el objetivo y ejecuta el gesto de pulsación. Después debe esperar la respuesta visual o sonora antes de repetir la acción.

Selección de un elemento de interfaz mediante hand tracking



Nota. Captura de pantalla del indicador de selección situado sobre un elemento interactivo de la interfaz.

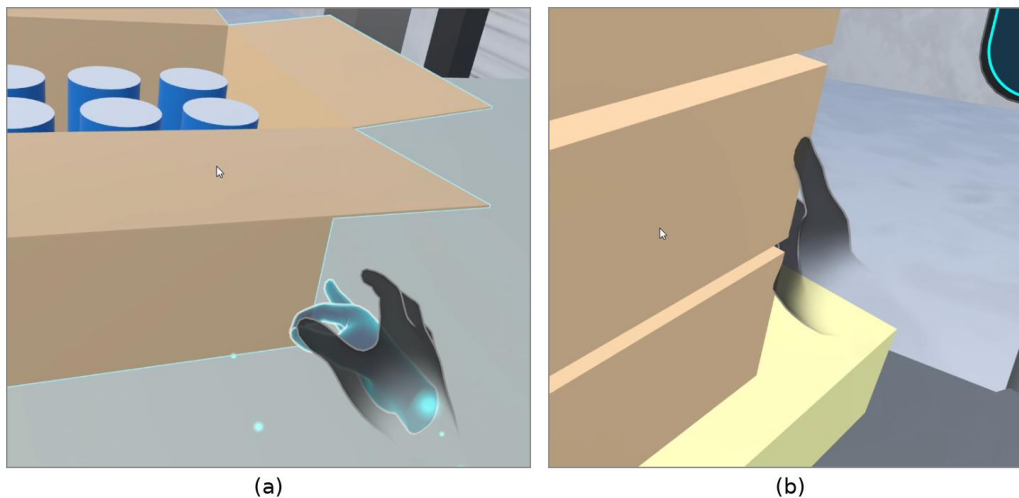
La imagen evidencia la relación entre la orientación de la mano y el elemento seleccionado. El indicador permite comprobar qué control recibirá la acción antes de efectuar la confirmación.

3.3.5. Agarre y manipulación de objetos

Los objetos configurados como manipulables pueden agarrarse con cualquiera de las dos manos. El usuario aproxima la mano al elemento, cierra la mano para sujetarlo, lo desplaza o inspecciona según la actividad y abre la mano para liberarlo. La representación virtual de la mano se adapta al agarre configurado para los objetos que lo requieren.

1. Acerque la mano al objeto que desea manipular.
2. Cierre la mano hasta que el sistema reconozca el agarre.
3. Mueva o inspeccione el objeto de acuerdo con la instrucción de la etapa.
4. Abra la mano para soltar el objeto en la posición requerida.

Secuencia básica de aproximación y agarre de un objeto



Nota. Capturas de pantalla de Explorador Global. (a) aproximación de la mano a un objeto manipulable; (b) objeto sostenido mediante seguimiento de manos.

La evidencia muestra una de las interacciones físicas comunes del simulador. El agarre permite trasladar elementos y colocarlos en las posiciones solicitadas por las actividades del recorrido.

3.4. Funciones e interacciones disponibles

3.4.1. Sistema de etapas y tareas

Los recorridos se organizan mediante etapas consecutivas. Al comenzar una actividad, el usuario recibe un panel con información sobre el objetivo y las acciones requeridas. Después de iniciar la etapa, debe completar las tareas visibles o indicadas por el sistema antes de avanzar. Algunas etapas se desarrollan dentro del mismo escenario y otras implican una transición hacia un entorno diferente.

3.4.2. Interacción con documentos

En las actividades que utilizan documentación, el usuario consulta la información presentada y ejecuta la acción solicitada sobre los elementos interactivos. Según la tarea,

puede seleccionar datos, arrastrar elementos hacia una posición determinada o confirmar la información mediante los controles de la interfaz. El estado mostrado por el sistema permite identificar si una acción continúa pendiente, requiere corrección o fue aceptada.

3.4.3. Validación de objetos y acciones

Determinadas actividades requieren colocar un objeto en una zona específica o seleccionar una opción válida. Cuando la acción coincide con lo solicitado, el sistema presenta la respuesta correspondiente y la tarea puede avanzar. Si el elemento o la opción no son correctos, la actividad permanece pendiente y el usuario debe corregir la acción antes de continuar.

3.4.4. Progreso y finalización de etapas

La interfaz de progreso informa la etapa actual y el avance alcanzado durante la sesión. Una etapa finaliza cuando el sistema reconoce las tareas necesarias y habilita la transición correspondiente. Si el recorrido no avanza, el usuario debe comprobar que no existan actividades pendientes.

Ante una etapa que no finaliza, se recomienda revisar:

- Las tareas indicadas en el panel de progreso;
- Los objetos que todavía no han sido colocados;
- Los documentos o campos que continúan pendientes;
- Las zonas de validación relacionadas con la actividad; y
- Las ayudas visuales o sonoras que indiquen la siguiente acción.

3.4.5. Recorridos de exportación e importación

Explorador Global dispone de recorridos de exportación e importación que utilizan los mismos mecanismos generales de interacción, pero presentan actividades, documentos y decisiones diferentes. El usuario debe seguir las instrucciones de cada etapa

y completar las acciones en el orden establecido por el sistema. Este manual se concentra en la operación de la aplicación y no reemplaza la explicación académica del contenido aduanero representado.

3.5. Evidencia de uso

Este apartado presenta una secuencia documentada del uso de Explorador Global mediante capturas obtenidas durante la ejecución de la aplicación. Las evidencias muestran el acceso al recorrido, el inicio de una actividad, la interacción con elementos del escenario y la finalización de una etapa, con el propósito de relacionar los controles y funciones descritos anteriormente con su aplicación dentro de la experiencia.

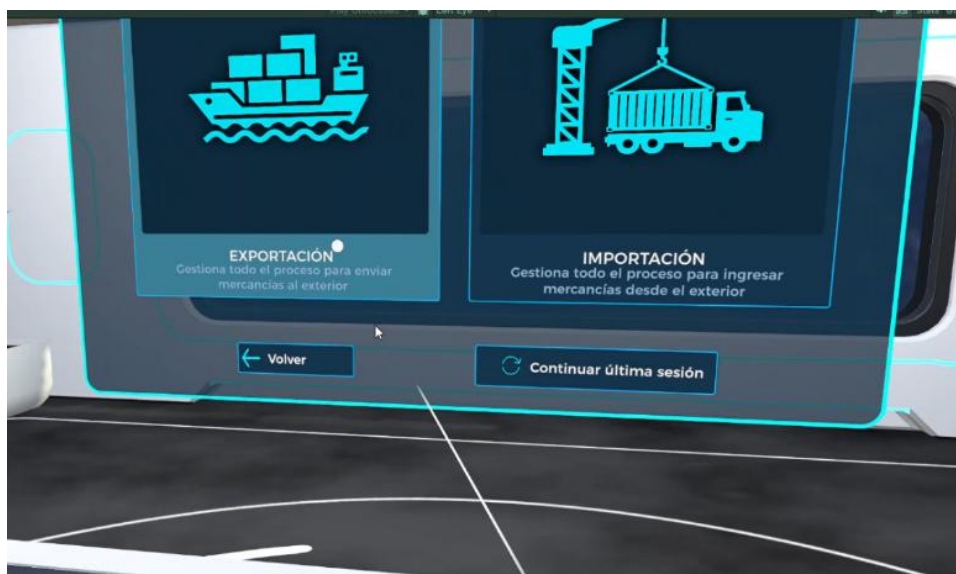
3.5.1. Inicio y selección del recorrido

Después de completar el tutorial, el usuario selecciona el módulo que desea ejecutar. La pantalla diferencia los recorridos de exportación e importación y permite iniciar la secuencia correspondiente mediante la misma interacción gestual utilizada en los demás paneles.

Ilustración

7

Selección de los módulos de exportación e importación



Nota. Captura de pantalla de la interfaz de selección de módulos de Explorador Global.

La interfaz muestra el punto de decisión que determina el recorrido de la sesión. Una vez confirmada la opción, el sistema presenta las etapas asociadas al módulo seleccionado.

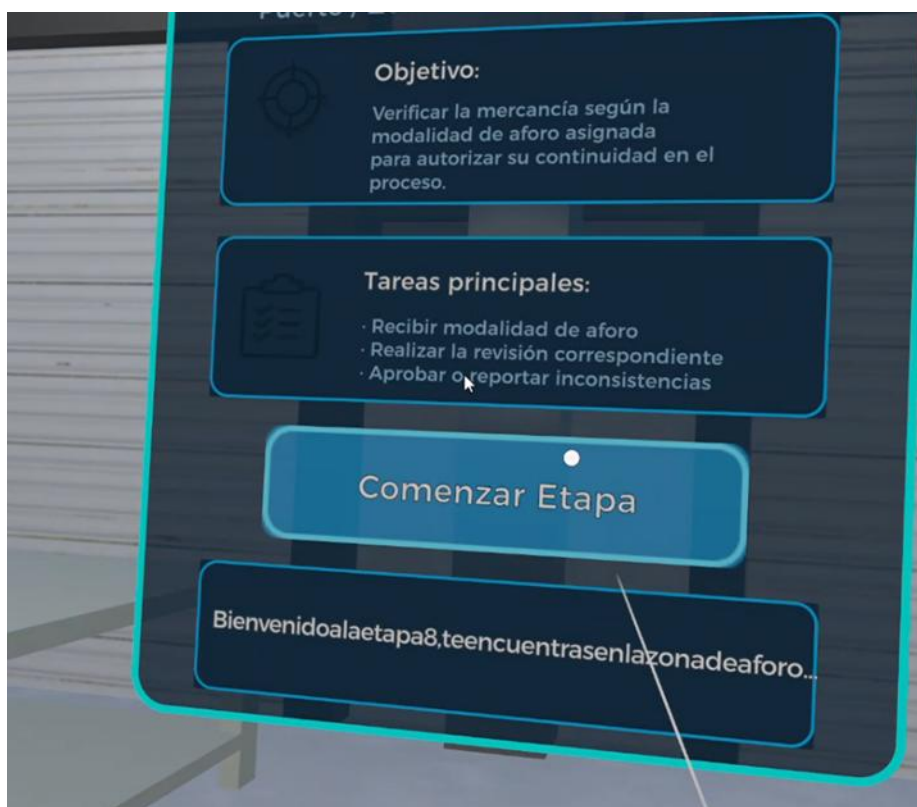
3.5.2. Inicio de una actividad

Al acceder a una etapa, el sistema presenta un panel informativo con el objetivo y las acciones que deben completarse. El usuario revisa esta información antes de iniciar la actividad para identificar qué elementos del escenario o de la interfaz deberá utilizar.

Ilustración

8

Panel informativo al inicio de una etapa



Nota. Captura de pantalla del panel de una etapa antes de comenzar la actividad correspondiente.

La evidencia permite observar cómo se presenta la información previa a la acción. El panel funciona como referencia para relacionar el objetivo de la etapa con las tareas que el usuario debe ejecutar dentro del entorno.

3.5.3. Interacción y validación durante la actividad

Durante el recorrido, el usuario combina navegación, selección de interfaz y manipulación de objetos según las instrucciones disponibles. Cuando una actividad requiere trasladar un elemento, el objeto puede sujetarse, moverse y colocarse en la posición indicada antes de que el sistema registre la acción.

Ilustración

9

Manipulación y traslado de un objeto dentro del escenario



Nota. Captura de pantalla de Explorador Global durante la manipulación y traslado de un objeto mediante seguimiento de manos.

La captura evidencia una interacción física utilizada durante las tareas del simulador. El usuario sujeta y desplaza el objeto hasta la posición indicada por la actividad; cuando se cumple la condición requerida, el sistema registra la acción y permite continuar con el flujo correspondiente.

3.5.4. Finalización de una etapa

Cuando las tareas requeridas han sido completadas, el sistema presenta la confirmación correspondiente y administra la transición hacia la siguiente parte del recorrido. Esta respuesta permite al usuario reconocer que la actividad actual terminó antes de continuar.

Ilustración

10

Panel de finalización de una etapa de Explorador Global



Nota. Captura de pantalla del panel mostrado al completar una etapa de Explorador Global.

La imagen documenta el cierre de una actividad y la comunicación del estado final al usuario. La confirmación funciona como punto de transición entre la tarea completada y la siguiente etapa del recorrido.

En conjunto, la secuencia evidencia el flujo básico de uso de la aplicación: preparación e ingreso, selección del recorrido, consulta de instrucciones, ejecución de interacciones, validación de tareas y finalización de la etapa. Las funciones específicas

varían entre exportación e importación, pero mantienen los mismos principios generales de navegación y respuesta descritos en este capítulo.

Capítulo IV

4. Manual De Programador

4.1. Arquitectura general del sistema

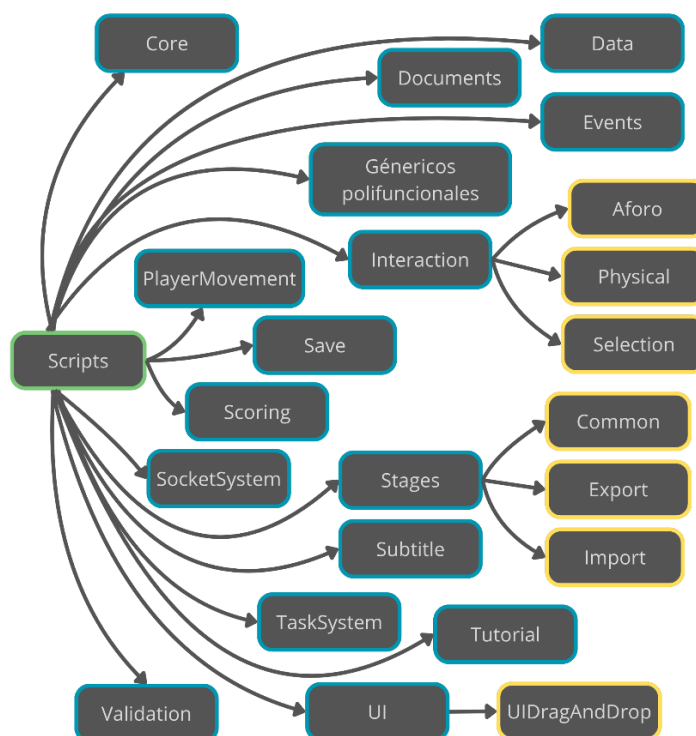
Explorador Global utiliza una arquitectura modular orientada a componentes. El repositorio contiene 92 scripts distribuidos en carpetas funcionales. La estructura no corresponde a microservicios ni a una arquitectura por capas estricta; se apoya en el modelo de componentes de Unity y separa las responsabilidades por sistemas.

Los módulos principales son Core, Data, Documents, Events, Interaction, PlayerMovement, Save, Scoring, SocketSystem, Stages, Subtitle, TaskSystem, Tutorial, UI y Validation.

Ilustración

11

Mapa de módulos del repositorio



Nota: Elaboración propia.

El diagrama presenta la organización general de los scripts de Explorador Global. Desde la carpeta principal Scripts se distribuyen los módulos funcionales encargados de datos, documentos, eventos, interacción, movimiento, etapas, tareas, tutoriales, interfaz y validación. También muestra subdivisiones específicas, como los controladores de aforo, interacción física y selección, además de los recorridos de exportación e importación. Esta separación facilita localizar los componentes y mantener cada responsabilidad dentro de un grupo definido.

4.2. Tecnologías, motores y herramientas utilizadas

El desarrollo se realizó en Unity 6000.3.5f2 mediante C#. La interacción inmersiva emplea Meta XR Interaction SDK 85.0.0 y los componentes requeridos de Meta XR Core SDK y Meta XR All-in-One SDK. La aplicación fue compilada y probada en Meta Quest 3S, aunque el dispositivo objetivo incluye Quest 3 y Quest 3S.

Las animaciones de tutorial se construyeron con el sistema de animación de Unity. Para la orientación auditiva se utilizaron componentes AudioSource tridimensionales. Las interfaces fueron creadas con las herramientas de UI de Unity y conectadas mediante UnityEvent.

Ilustración

12

Paquetes instalados en el proyecto

▼ Packages - Asset Store		
AllSky Free - 10 Sky / Skybox Set	1.1.0	✓
Autosaver	1.1	✓
Construction Props MegaPack	1.0	✓
Free Casual SoundFX Pack	1.0	✓
Free Sound Effects Pack	1.0	✓
Free Stylized Textures	1.0	✓
Industrial Building Materials Pack	1.0	✓
Industrial Fuse Box Pack	1.0	✓
Industrial Lockers Pack	1.0	✓
Industrial Props Equipment Mega Bundle	1.0	✓
Industrial Racks Collection	1.0	✓
Meta - Voice SDK - Immersive Voice Commands	85.0.0	🔗
Meta MR Utility Kit	85.0.0	🔗
Meta XR All-in-One SDK	85.0.0	🔗
Meta XR Audio SDK	85.0.0	🔗
Meta XR Core SDK	85.0.0	🔗
Meta XR Haptics SDK	85.0.0	🔗
Meta XR Interaction SDK	85.0.0	🔗
Meta XR Interaction SDK Essentials	85.0.0	🔗
Meta XR Platform SDK	85.0.0	🔗
Procedural Walk Animation	1.0.0	✓
Quick Outline	1.1	✓
RPG Essentials Sound Effects - FREE!	1.0	✓
Skybox Series Free	4.3	✓

La captura muestra parte de los paquetes y recursos incorporados al proyecto de Unity. Entre ellos aparecen Meta XR Core SDK, Meta XR Interaction SDK, Meta XR All-in-One SDK y otros componentes de la versión 85.0.0, utilizados para integrar seguimiento de manos, interacción inmersiva y compatibilidad con Meta Quest. También se observan paquetes gráficos y sonoros empleados en la construcción de los escenarios. La imagen documenta las dependencias disponibles durante el desarrollo, pero no implica que todos los recursos mostrados intervengan directamente en cada sistema.

4.3. Organización técnica del proyecto

La organización por carpetas permite ubicar rápidamente los scripts asociados con una función. Stages contiene controladores comunes y específicos de exportación e

importación; Interaction agrupa funciones como aforo, selección y comportamiento físico; SocketSystem valida objetos; TaskSystem administra actividades; Tutorial controla las ayudas; y UI reúne elementos de interfaz y arrastrar y soltar.

Las escenas representan segmentos del recorrido. Algunos cambios se realizan mediante carga de escena y otros mediante teletransportación dentro del entorno actual. Las referencias a objetos, sonidos, animaciones y eventos se configuran desde el Inspector.

Tabla 2 Organización funcional de los módulos, responsabilidades y componentes principales de Explorador Global

Módulo	Propósito	Clases o scripts representativos	Dependencias principales
Core	Coordinar el flujo general y representar el modo de simulación seleccionado.	GameFlowManager, SimulationContext, SimulationMode	Stages, Unity Engine
Data	Mantener temporalmente la etapa actual, el número de etapas y el proceso seleccionado.	ProgressData, ProgressDataEditor, ProgressUI	UI, TextMeshPro, Unity Engine
Documents	Representar y administrar los documentos utilizados en las actividades de exportación e importación.	BaseDocument, CommercialInvoice Document, PackingListDocument, BillOfLadingDocument, CertificateOfOrigin Document, DocumentField, DragDropSlot	UI, sistema de validación, Unity Engine
Events	Estructura prevista para comunicar eventos generales del simulador.	GameEventBus, GameEvents, StageEventPayload	Unity Engine, C# Events

Interaction	Gestionar las mecánicas de selección, aforo, interacción física y respuesta de objetos.	AforoManager, BaseAforoController, AutomaticAforoController, DocumentaryAforoController	Meta XR, Unity Physics, Validation
PlayerMovement	Integrar la locomoción gestual de Meta XR con el sistema principal de desplazamiento.	PanelLocomotionBroadcaster	Meta XR Interaction SDK, Stages, Unity Engine
Save	Módulo reservado para una posible implementación futura de persistencia.	SaveData, SaveManager	Ninguna dependencia funcional en la versión actual
Scoring	Administrar valores o resultados obtenidos durante determinadas actividades.	Scripts de puntuación y presentación de resultados	TaskSystem, UI, Unity Engine
SocketSystem	Recibir objetos y comprobar si corresponden con el tipo esperado.	SocketValidator, SocketSelectionHandler, SocketItem	Meta XR SnapInteractable, Unity Events, Validation
Stages	Controlar las etapas, su secuencia, cambios de escena, teletransportación y finalización.	BaseStageController, StageManager, StageEntry, SceneStageGroup, GameFlowManager, controladores de exportación e importación	Core, Unity SceneManager, VRFade, TaskSystem
Subtitle	Mostrar textos de apoyo, indicaciones o subtítulos durante la experiencia.	SubtitleManager	TextMeshPro, UI, Unity Engine
TaskSystem	Registrar tareas, validar su finalización y comunicar el progreso a la interfaz.	TaskManager, TaskEntry, TaskNotificationUI	C# Events, Unity Events, UI, Stages

Tutorial	Presentar ayudas audiovisuales y demostraciones animadas de las interacciones.	TutorialManager, TutorialStep, TutorialInput	Animator, OVR Input, Unity Events, AudioSource
UI	Gestionar paneles, botones, elementos arrastrables, contenedores y comparación de datos.	UIDraggableItem, UIItemContainer, UICompareStringItemContainer, UICompareStringContainerManager	Unity EventSystem, TextMeshPro, Documents, Validation
Validation	Comprobar condiciones, selecciones y respuestas antes de completar una actividad.	Validadores específicos configurados para cada mecánica	TaskSystem, SocketSystem, UI, Unity Events
Generics o componentes polifuncionales	Proporcionar comportamientos reutilizables para activar objetos, ejecutar eventos, controlar colisiones y contar documentos.	ActiveActionOnEnableWithDelay, ActiveAndDesactiveOBJ, ActOneObjWhitUnityEvent, Balanza, DocumentTriggerCounter	Unity Events, Physics, Documents, TaskSystem

4.4. Arquitectura de clases y componentes

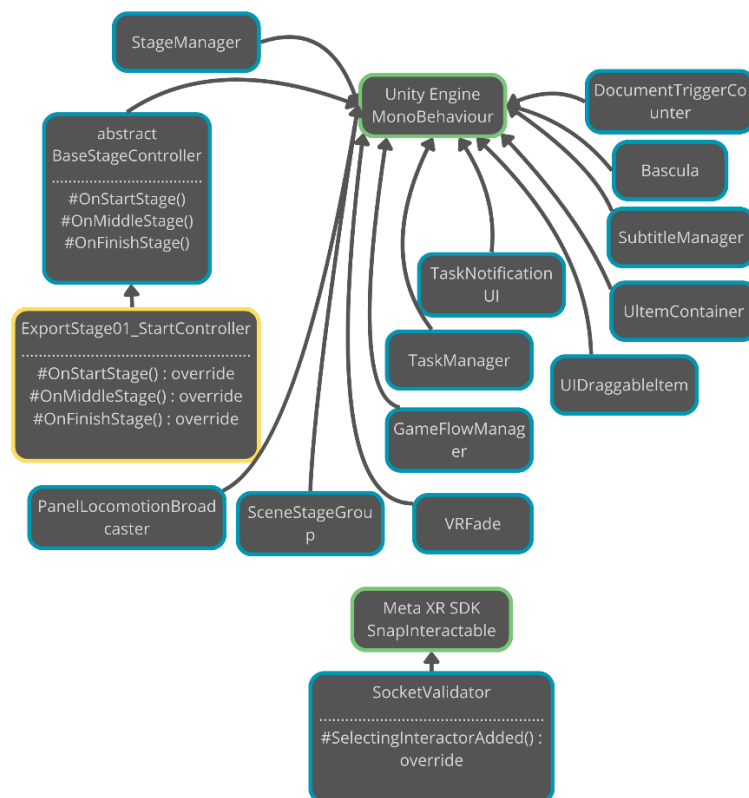
La mayor parte de los comportamientos hereda de MonoBehaviour. El proyecto utiliza herencia cuando existe una secuencia compartida y composición cuando un objeto necesita combinar capacidades configurables.

BaseStageController es la clase abstracta principal para las etapas. Define campos comunes, tiempos, referencias de teletransporte, eventos y el ciclo de inicio, desarrollo y cierre. Los controladores concretos sobrescriben los puntos necesarios.

Ilustración

13

Jerarquía global de herencias del sistema primera parte



El diagrama resume las relaciones de herencia más representativas de la aplicación. La mayoría de los componentes programados extienden MonoBehaviour, lo que permite incorporarlos a objetos de las escenas y configurar sus referencias desde el Inspector de Unity. También se muestra la especialización de BaseStageController mediante controladores concretos y la extensión de componentes proporcionados por Meta XR, como SnapInteractable. Estas relaciones permiten compartir comportamientos comunes y modificar únicamente la lógica particular de cada clase derivada.

4.5. Desarrollo de sistemas principales

4.5.1. Sistema central de etapas

GameFlowManager localiza el controlador activo y se suscribe al evento de finalización. Cuando recibe la notificación, solicita a StageManager la

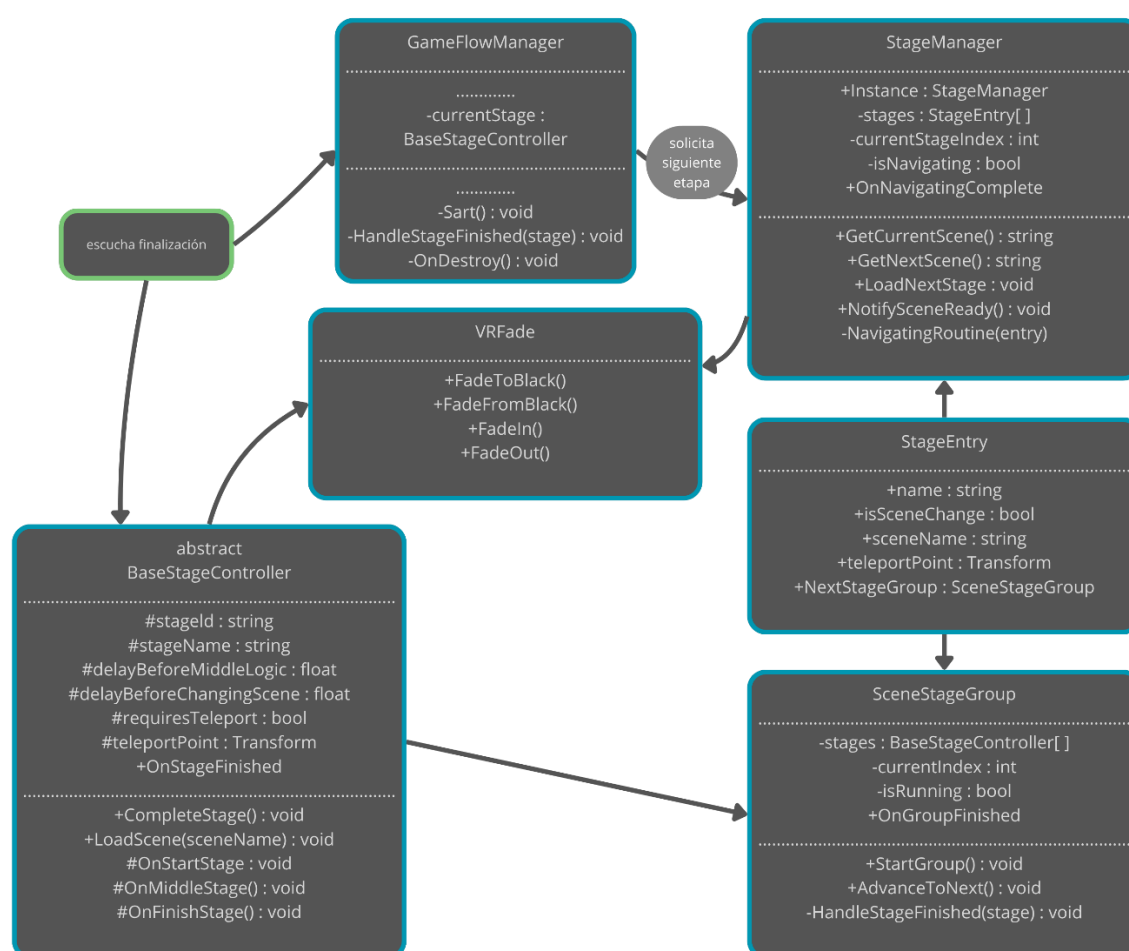
siguiente entrada. StageManager determina si debe cargar otra escena o ejecutar un teletransporte dentro de la actual.

StageEntry conserva el nombre, el tipo de transición, la escena, el destino y el grupo siguiente. SceneStageGroup administra conjuntos de controladores cuando varias actividades pertenecen a una misma escena.

Ilustración

14

Arquitectura del sistema central de etapas.



El diagrama representa la coordinación entre GameFlowManager, StageManager, StageEntry, SceneStageGroup, BaseStageController y VRFade. GameFlowManager supervisa la etapa activa y detecta su finalización, mientras StageManager determina cuál debe ejecutarse después. Cada StageEntry contiene la información necesaria para realizar

la transición, y SceneStageGroup administra varias actividades dentro de una misma escena. VRFade interviene durante las cargas o teletransportaciones para suavizar visualmente el cambio.

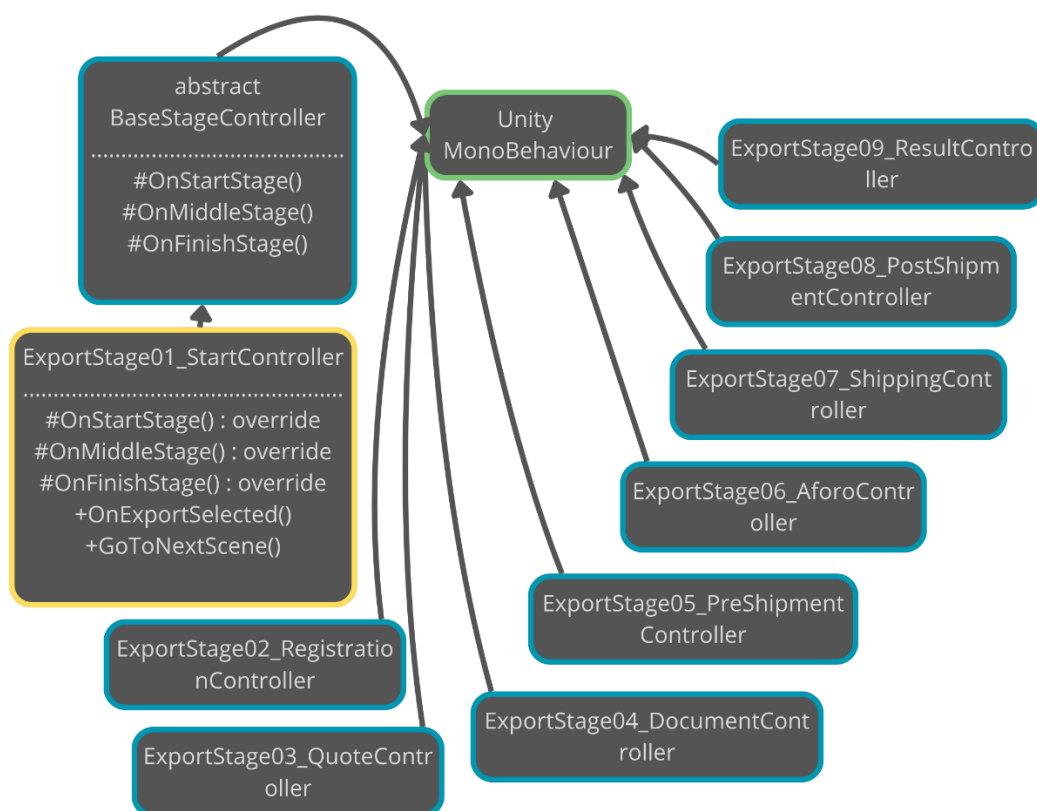
4.5.2. Controladores de exportación

El recorrido de exportación está formado por controladores numerados según su etapa. Cada clase contiene la lógica necesaria para registro, cotización, documentos, preembarque, aforo, embarque, posenvío y resultados, según la secuencia educativa aprobada.

Ilustración

15

Controladores del recorrido de exportación.



La ilustración organiza las clases encargadas de las etapas del módulo de exportación. Los controladores numerados representan actividades de inicio, registro, cotización, gestión documental, preembarque, aforo, embarque, posenvío y presentación

de resultados. Los componentes comparten la base proporcionada por Unity y, cuando corresponde, heredan el ciclo común definido en BaseStageController. Cada clase conserva la lógica y los eventos particulares de su etapa, aunque todas participan en la misma secuencia educativa.

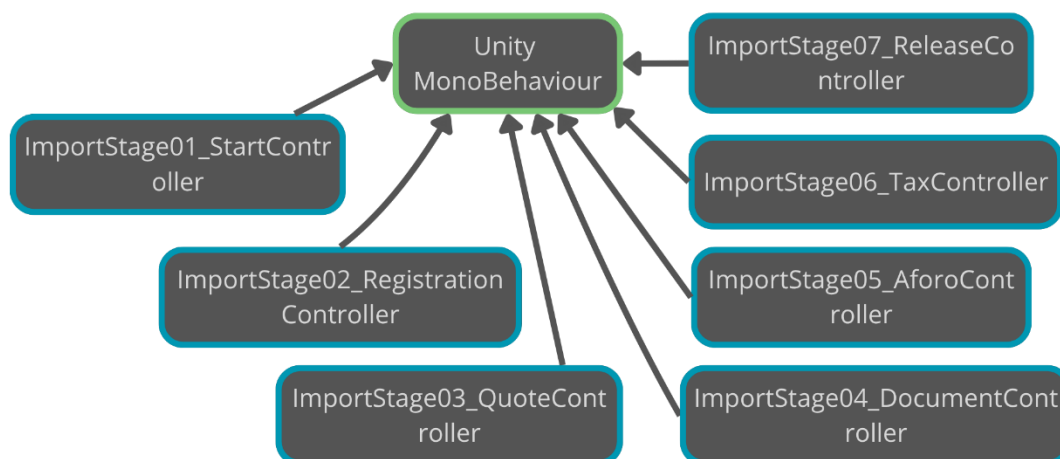
4.5.3. Controladores de importación

El módulo de importación se organiza mediante controladores para inicio, registro, cotización, documentos, aforo, tributos y liberación. Las clases comparten el modelo de componentes, aunque sus tareas y eventos se configuran por separado

Ilustración

16

Controladores del recorrido de importación.



El diagrama muestra los controladores que integran el recorrido de importación: inicio, registro, cotización, documentos, aforo, tributos y liberación. Cada componente hereda de MonoBehaviour y administra las acciones asignadas a una etapa determinada. Aunque las clases utilizan una estructura semejante, sus tareas, objetos y validaciones se configuran de manera independiente dentro de Unity.

4.6. **Desarrollo de mecánicas principales**

Las mecánicas comprenden locomoción, selección, agarre, colocación, validación, arrastrar y soltar, comparación de cadenas y activación de tareas. El hand tracking permite utilizar cualquiera de las dos manos.

La locomoción gestual pertenece a Meta XR; el proyecto la integró con el flujo principal y la configuró para activar, orientar y confirmar el teletransporte. Los objetos agarrables incorporan poses bilaterales. Los sockets reciben elementos y verifican su tipo.

El tutorial utiliza un prefab de mano de Meta animado en Unity. La demostración permanece activa hasta que la acción se valida y después se activa el audio espacial de la tarea siguiente.

4.7. **Integración de sistemas**

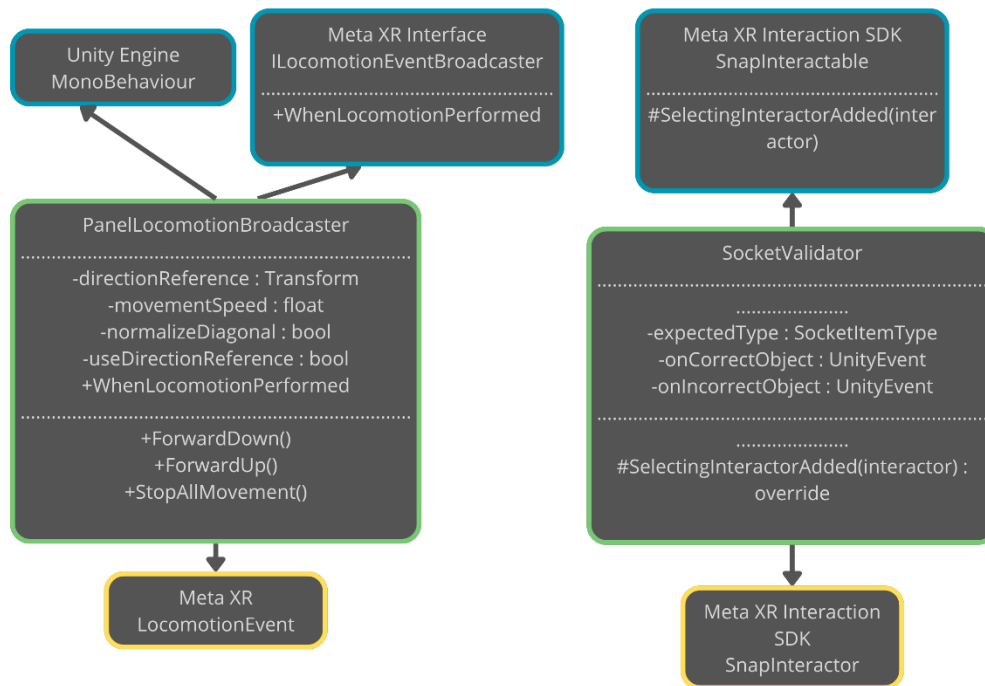
La integración se realiza mediante referencias serializadas, eventos de C# y UnityEvent. Este modelo permite conectar botones, validadores, tareas, animaciones, sonido y cambios de etapa desde el Inspector.

TaskManager emite cambios de progreso; TaskNotificationUI los recibe y actualiza la interfaz. SocketValidator ejecuta eventos según el objeto sea correcto o incorrecto. Los controladores de etapa escuchan o reciben estas respuestas para completar su lógica.

Ilustración

17

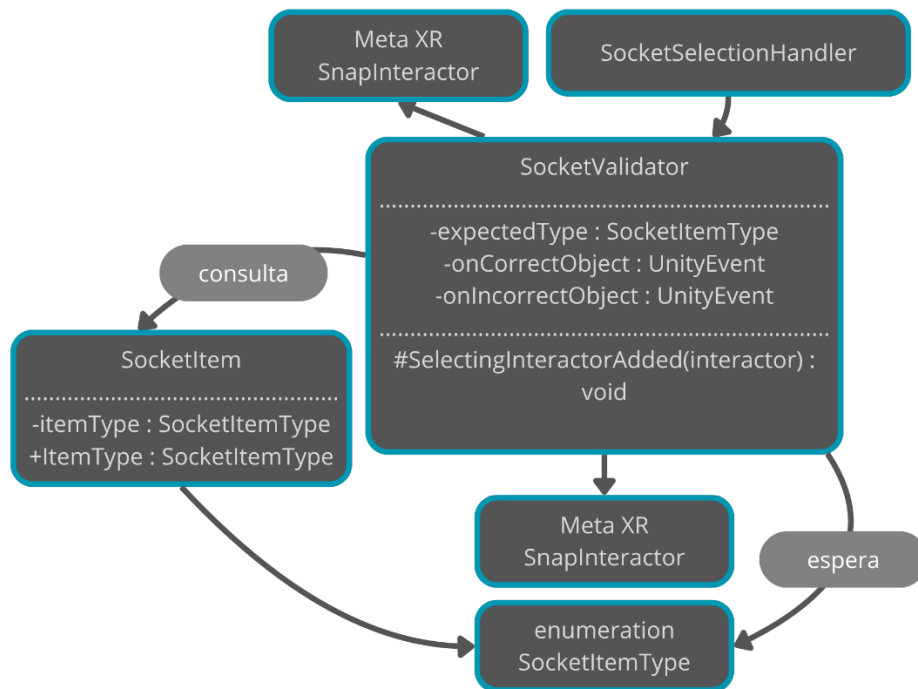
Integración del sistema con Meta XR.



La ilustración presenta dos puntos principales de integración con Meta XR. En el primero, **PanelLocomotionBroadcaster** conecta el sistema de locomoción de la aplicación con los eventos gestuales proporcionados por Meta XR. En el segundo, **SocketValidator** amplía el comportamiento de **SnapInteractable** para revisar el tipo de objeto colocado y ejecutar respuestas de acierto o error. El proyecto no desarrolla las funciones internas de seguimiento de manos del SDK; configura sus componentes y los enlaza con la lógica del simulador.

Ilustración**18**

Funcionamiento del sistema de sockets

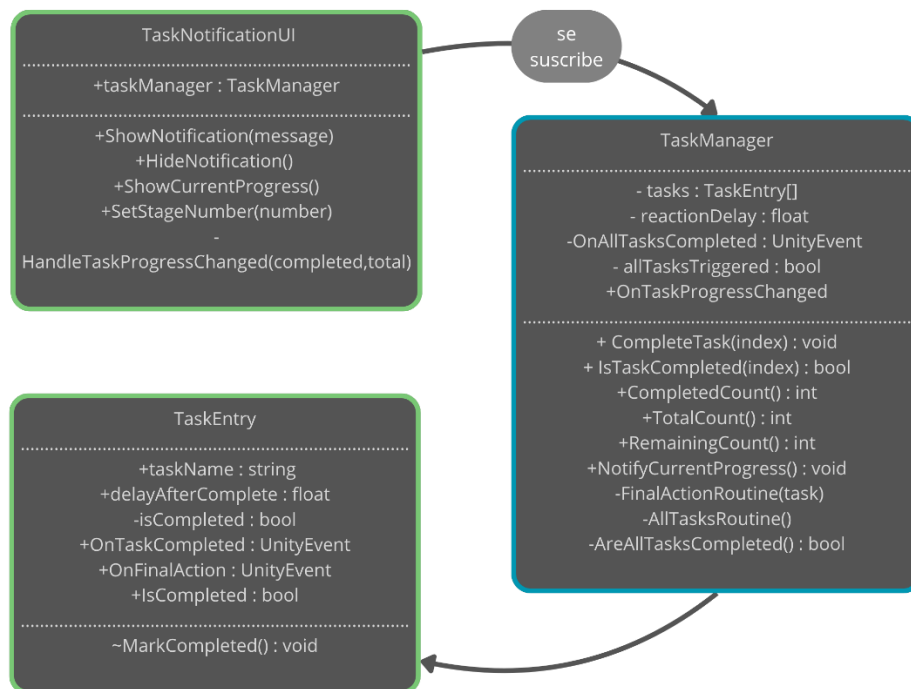


El diagrama explica el proceso utilizado para validar objetos dentro de una zona receptora. **SocketValidator** recibe la interacción generada por **SnapInteractor**, consulta la información del objeto mediante **SocketItem** y compara su tipo con el valor esperado de **SocketItemType**. Según el resultado, ejecuta el evento configurado para una colocación correcta o incorrecta. **SocketSelectionHandler** complementa este comportamiento gestionando la respuesta asociada con la selección.

Ilustración

19

Comunicación entre tareas e interfaz de progreso.



La ilustración representa el intercambio de información entre TaskEntry, TaskManager y TaskNotificationUI. Cada TaskEntry conserva los datos y el estado de una actividad. TaskManager administra el conjunto de tareas, registra cuáles fueron completadas y emite cambios de progreso. TaskNotificationUI se suscribe a dichas actualizaciones para mostrar mensajes, el número de etapa y el avance alcanzado. La comunicación mediante eventos evita que la interfaz tenga que comprobar continuamente el estado de cada tarea.

4.8. Integración entre diseño, arte y programación

Los escenarios y recursos visuales se incorporaron como objetos de Unity. La programación añadió comportamientos sin modificar la finalidad visual de cada recurso. Las referencias se configuraron desde el Inspector para permitir que animaciones, paneles y sonidos respondieran a eventos.

Las manos tutoriales constituyen un ejemplo de integración. El modelo fue proporcionado por Meta, la animación fue creada en Unity y el sistema de validación controla su desaparición. De forma semejante, los paneles combinan diseño visual, texto y métodos programados.

4.9. Gestión de datos y persistencia

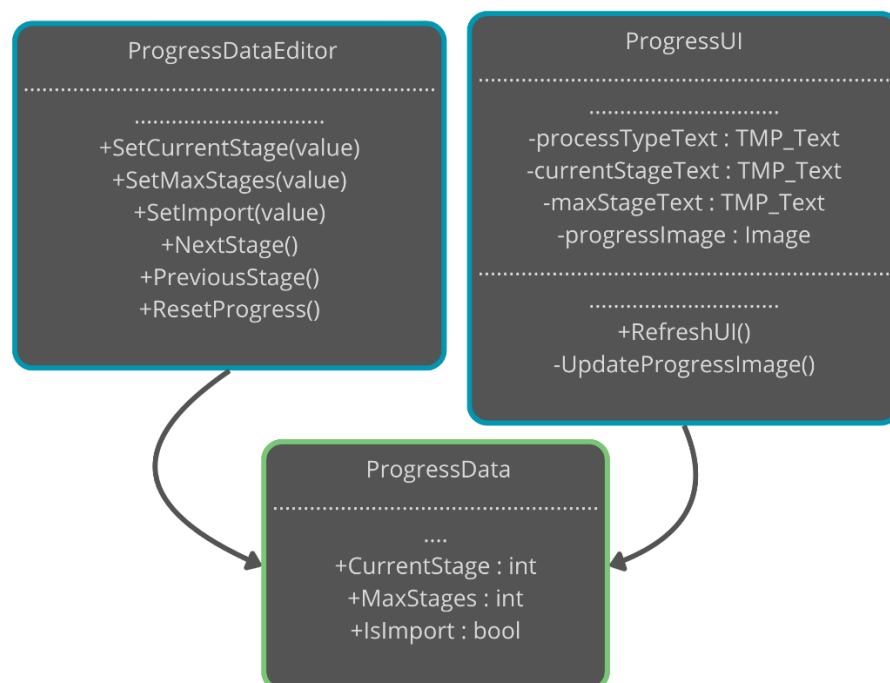
La versión actual no implementa persistencia. ProgressData es una clase estática que mantiene etapa actual, número máximo de etapas y tipo de proceso durante la ejecución. Al cerrar la aplicación, la información se pierde.

SaveData y SaveManager se encuentran vacíos y no forman parte del funcionamiento. Tampoco existe uso de PlayerPrefs, JSON, archivos locales o base de datos. Esta decisión responde a los requerimientos definidos para el alcance actual.

Ilustración

20

Gestión temporal del progreso durante la ejecución.



El diagrama muestra cómo ProgressDataEditor y ProgressUI utilizan la clase estática ProgressData. ProgressDataEditor modifica la etapa actual, el número máximo de etapas y el tipo de recorrido, mientras ProgressUI consulta esos valores para actualizar los textos y la representación visual del progreso. La información permanece disponible únicamente mientras la aplicación está abierta; no se guarda en archivos, perfiles ni bases de datos después del cierre.

4.10. **Configuración de escenas y flujo del simulador**

El flujo comienza con el tutorial y el menú de selección. Luego se carga el recorrido elegido. StageManager conserva el índice de avance y consulta la entrada siguiente. Cuando la entrada requiere otra escena, utiliza SceneManager; cuando las actividades permanecen en el mismo escenario, mueve el rig del usuario y activa el grupo correspondiente.

Las transiciones utilizan VRFade para oscurecer la vista antes de una carga o teletransportación. Esto oculta cambios bruscos de posición y mantiene continuidad visual.

4.11. **Pruebas técnicas y validación**

Las pruebas técnicas comprobaron ejecución en Meta Quest 3S, carga de escenas, teletransporte, agarre, sockets, tareas, tutoriales, interfaces y audio. El visor se configuró con límite de 72 FPS.

La prueba piloto reunió cinco estudiantes de distintas carreras. Se observó directamente su interacción y se recibieron comentarios verbales. La evaluación se limitó a mecánicas y usabilidad; no midió conocimientos de comercio exterior.

Las principales modificaciones fueron la integración de locomoción gestual, la ampliación del alcance de agarre y la configuración de poses para ambas manos. Estos cambios evidencian un proceso de mejora basado en observación de usuarios.

Tabla 3 Resultados de las pruebas técnicas y de la prueba piloto de usabilidad mecánica

Elemento evaluado	Procedimiento de prueba	Resultado observado	Ajuste o decisión adoptada
Inicio de la aplicación	Se ejecutó la compilación en Meta Quest 3S y se comprobó el acceso al tutorial y al menú principal.	La aplicación inició correctamente y permitió acceder a la selección de los módulos de exportación e importación.	Se mantuvo el tutorial inicial con reproducción en bucle y opción para omitirlo.
Locomoción	Los participantes se desplazaron por los escenarios utilizando el sistema inicial y, posteriormente, la locomoción gestual de Meta XR.	En la prueba inicial se observaron dificultades para desplazarse sin controladores. La integración del gesto facilitó la selección del destino y el teletransporte.	Se integró el sistema de locomoción gestual de Meta XR con el flujo principal y se habilitó su uso con cualquiera de las dos manos.
Agarre de objetos	Los usuarios intentaron seleccionar, agarrar, mover y soltar los objetos requeridos en diferentes etapas.	Algunos elementos resultaban difíciles de alcanzar o seleccionar con hand tracking.	Se amplió el alcance de selección de los objetos agarrables para facilitar su manipulación.
Poses de mano	Se observó la representación de las manos al sostener diferentes objetos con la mano izquierda y derecha.	Los objetos podían manipularse con ambas manos, pero necesitaban una representación de	Se configuraron poses específicas para ambas manos en los objetos agarrables.

		agarre coherente.	más
Sistema de sockets	Se colocaron objetos correctos e incorrectos en las zonas receptoras configuradas.	El sistema identificó el tipo del objeto y ejecutó los eventos correspondientes para selecciones válidas o no válidas.	Se mantuvo la validación mediante SocketValidator y eventos configurables desde el Inspector.
Sistema de tareas	Se completaron las acciones configuradas en diferente orden y se observó la actualización del progreso.	Las tareas se registraron durante la ejecución y el sistema reconoció cuándo se completó el conjunto requerido.	Se conservaron los eventos individuales y el evento general de finalización administrados por TaskManager.
Tutorial inicial	Se reprodujo el video de locomoción al comenzar la experiencia y se comprobó la opción para omitirlo.	El video mostró de forma continua el uso del desplazamiento y la teletransportación mediante hand tracking.	Se configuró su reproducción en bucle y se mantuvo la opción de omisión para usuarios familiarizados con la mecánica.
Tutorial contextual	Los participantes observaron la mano animada e intentaron reproducir las acciones de agarrar y ubicar objetos.	La demostración facilitó la identificación del movimiento esperado y desapareció al completarse correctamente la interacción.	Se mantuvo la animación activa hasta la validación de la acción.

Audio espacial	Se activaron las señales sonoras y los usuarios se desplazaron hacia la siguiente zona de interacción.	El sonido permitió identificar la dirección de la siguiente actividad y aumentó su intensidad percibida al acercarse a la fuente.	Se utilizaron componentes AudioSource 3D de Unity como guía direccional.
Interfaces	Se probaron botones, paneles, instrucciones, selección de módulos y elementos de arrastrar y soltar.	Las interfaces respondieron a las acciones y permitieron acceder a las funciones configuradas.	Se conservaron conexiones mediante UnityEvent y las interfaces del sistema de eventos de Unity.
Cambios de escena	Se completaron etapas que requerían cargar otra escena y etapas que utilizaban teletransportación dentro del mismo entorno.	El sistema ejecutó cambios de escena y desplazamientos internos de acuerdo con las entradas configuradas.	Se mantuvo la coordinación entre StageManager, SceneManager, SceneStageGroup y VRFade.
Estado del progreso	Se avanzó por varias etapas y posteriormente se cerró y reinició la aplicación.	El progreso permaneció disponible durante la ejecución, pero se eliminó después de cerrar el simulador.	Se confirmó que la versión actual utiliza datos estáticos temporales y no implementa persistencia.
Rendimiento	Se ejecutaron diferentes escenas en Meta Quest 3S con el visor limitado a 72 FPS y se observaron las métricas mostradas en pantalla.	Se registraron valores aproximados entre 52 y 72 FPS. Varias mediciones se situaron entre 69 y 72 FPS, aunque se presentó una caída puntual cercana a 52 FPS.	Se determinó que el sistema puede aproximarse al límite configurado, pero requiere mediciones continuas y optimización adicional en las

escenas con
mayor carga.

La tabla reúne los procedimientos aplicados para comprobar el inicio de la aplicación, locomoción, agarre, poses de mano, sockets, tareas, tutoriales, audio espacial, interfaces, cambios de escena, gestión del progreso y rendimiento en Meta Quest 3S. También presenta los resultados observados y las decisiones adoptadas después de cada comprobación. La información corresponde a pruebas funcionales y a una prueba piloto con cinco estudiantes; no representa una evaluación del aprendizaje de comercio exterior.

Capítulo V

5. Conclusiones y Recomendaciones

5.1. Conclusiones

1. La programación se distribuyó en 92 scripts organizados por módulos, entre ellos Stages, Interaction, SocketSystem, TaskSystem, Tutorial, UI, Documents y Validation. Esta separación ayudó a mantener responsabilidades concretas y facilitó los cambios solicitados durante el desarrollo. No todos los componentes cumplen la misma función: algunos controlan el recorrido, otros comprueban acciones del usuario y varios administran elementos visuales, documentos o actividades.

2. El proyecto emplea Singleton en StageManager, Template Method en BaseStageController y una comunicación similar a Observer mediante eventos de C# y UnityEvent. Estos patrones contribuyeron a coordinar el recorrido, reutilizar el ciclo común de las etapas y comunicar acciones entre componentes con menos dependencias directas.

5. La prueba piloto con cinco estudiantes permitió identificar dificultades mecánicas relacionadas con el desplazamiento y el agarre de objetos. A partir de la observación directa y los comentarios recibidos, se integró la locomoción gestual, se amplió el alcance de agarre y se configuraron poses para ambas manos. Estos ajustes mejoraron la facilidad de interacción observada durante las pruebas, aunque el tamaño de la muestra no permite generalizar los resultados.

6. Las pruebas técnicas en Meta Quest 3S comprobaron el funcionamiento del inicio, la carga de escenas, el teletransporte, el agarre, los sockets, las tareas, los tutoriales, las interfaces y el audio espacial. Con el visor limitado a 72 FPS se observaron registros puntuales aproximados entre 52 y 72 FPS, con varias mediciones cercanas a 69–72 FPS.

Estos valores demuestran que el prototipo puede aproximarse al límite configurado, pero todavía requiere mediciones continuas y optimización en las escenas con mayor carga.

7. Explorador Global constituye una base tecnológica para apoyar la práctica guiada de las secuencias definidas por LACELI. Sin embargo, el trabajo realizado únicamente comprobó el funcionamiento técnico y la usabilidad mecánica del prototipo; no evaluó la adquisición de conocimientos sobre comercio exterior. Su aporte educativo deberá determinarse posteriormente mediante una validación con estudiantes, instrumentos definidos y criterios de aprendizaje medibles.

8. El sistema de sockets comprueba si el objeto ubicado corresponde al tipo esperado. A partir de esa validación, los eventos configurados pueden aceptar la acción, mostrar una respuesta o mantener pendiente la actividad. El sistema de tareas registra las acciones completadas durante la ejecución y comunica el progreso a la interfaz. Los datos permanecen únicamente mientras la aplicación está abierta, debido a que el proyecto no incorpora una base de datos ni un mecanismo de persistencia. Al cerrar el simulador, la información temporal se elimina. Esta decisión coincide con el alcance definido por los responsables del proyecto.

5.2. **Recomendaciones**

1. Se recomienda realizar una validación educativa posterior con una muestra más amplia de estudiantes pertenecientes a la población objetivo. Esa evaluación debería medir aspectos diferentes de la usabilidad mecánica, como comprensión de la secuencia, identificación de documentos y capacidad para completar las actividades después de utilizar el simulador. Una comparación antes y después de la experiencia ofrecería datos más claros sobre posibles cambios en el aprendizaje.

2. Las futuras pruebas técnicas necesitan sesiones más prolongadas y un registro continuo del rendimiento. Las capturas actuales muestran valores aproximados de entre 52 y 72 FPS, aunque no representan un promedio. Un análisis posterior podría registrar la tasa de cuadros por segundo en cada escena, el consumo de CPU y GPU, el tiempo de carga, la temperatura del visor y los momentos donde ocurren caídas de rendimiento. Con esa información sería posible identificar con mayor precisión los objetos, materiales o procesos que necesitan optimización.

3. El simulador debe probarse también en Meta Quest 3, además de las comprobaciones realizadas en Meta Quest 3S. Aunque ambos dispositivos pertenecen a la misma familia y forman parte del alcance tecnológico, las pruebas documentadas corresponden al segundo visor. Revisar la aplicación en los dos equipos ayudaría a comprobar que las interacciones, la interfaz, el audio y el rendimiento mantienen un comportamiento adecuado.

4. Resulta conveniente mantener el uso sin controladores físicos como criterio principal. Las ayudas gestuales deben conservar instrucciones claras y estar disponibles cuando el usuario las necesite, sin interrumpir de forma innecesaria el recorrido. El tutorial inicial podría incorporar opciones para repetir únicamente el gesto que causa dificultad, en lugar de reproducir nuevamente toda la explicación.

Bibliografía

Albus, P., Vogt, A., & Seufert, T. (2021). Signaling in virtual reality influences learning outcome and cognitive load. *Computers & Education*, 166, Article 104154.

Buckingham, G. (2021). Hand tracking for immersive virtual reality: Opportunities and challenges. *Frontiers in Virtual Reality*, 2, Article 728461.

Caserman, P., Garcia-Agundez, A., Gámez Zerban, A., & Göbel, S. (2021). Cybersickness in current-generation virtual reality head-mounted displays: Systematic review and outlook. *Virtual Reality*, 25(4), 1153–1170.

Çoban, M., Bolat, Y. İ., & Göksu, İ. (2022). The potential of immersive virtual reality to enhance learning: A meta-analysis. *Educational Research Review*, 36, Article 100452.

Conrad, M., Kablitz, D., & Schumann, S. (2024). Learning effectiveness of immersive virtual reality in education and training: A systematic review of findings. *Computers & Education: X Reality*, 4, Article 100053.

Di Natale, A. F., Repetto, C., Riva, G., & Villani, D. (2020). Immersive virtual reality in K–12 and higher education: A 10-year systematic review of empirical research. *British Journal of Educational Technology*, 51(6), 2006–2033.

Hameed, A., Möller, S., & Perkis, A. (2023). How good are virtual hands? Influences of input modality on motor tasks in virtual reality. *Journal of Environmental Psychology*, 92, Article 102137.

Hamilton, D., McKechnie, J., Edgerton, E., & Wilson, C. (2021). Immersive virtual reality as a pedagogical tool in education: A systematic literature review of quantitative learning outcomes and experimental design. *Journal of Computers in Education*, 8(1), 1–32.

Jongbloed, J., Chaker, R., & Lavoué, E. (2024). Immersive procedural training in virtual reality: A systematic literature review. *Computers & Education*, 221, Article 105124.

Luo, H., Li, G., Feng, Q., Yang, Y., & Zuo, M. (2021). Virtual reality in K–12 and higher education: A systematic review of the literature from 2000 to 2019. *Journal of Computer Assisted Learning*, 37(3), 887–901.

Magallanes Rodríguez, J. S., Rodríguez Aspiazu, Q. J., Carpio Magallón, Á. M., & López García, M. R. (2021). Simulación y realidad virtual aplicada a la educación. *RECIAMUC*, 5(2), 101–110.

Makransky, G., & Mayer, R. E. (2022). Benefits of taking a virtual field trip in immersive virtual reality: Evidence for the immersion principle in multimedia learning. *Educational Psychology Review*, 34(3), 1771–1798.

Makransky, G., & Petersen, G. B. (2021). The cognitive affective model of immersive learning (CAMIL): A theoretical research-based model of learning in immersive virtual reality. *Educational Psychology Review*, 33(3), 937–958.

Masurovsky, A., Chojecki, P., Runde, D., Lafci, M., Przewozny, D., & Gaebler, M. (2020). Controller-free hand tracking for grab-and-place tasks in immersive virtual reality: Design elements and their empirical study. *Multimodal Technologies and Interaction*, 4(4), Article 91.

Pellas, N., Mystakidis, S., & Kazanidis, I. (2021). Immersive virtual reality in K–12 and higher education: A systematic review of the last decade scientific literature. *Virtual Reality*, 25(3), 835–861.

Pinargote Castro, M. A., Muñoz Pilozo, A. G., & Orellana Londoño, C. L. (2024). El rol de la realidad virtual en la educación superior. *Ciencia Latina Revista Científica Multidisciplinar*, 8(3), 9037–9045.

Radianti, J., Majchrzak, T. A., Fromm, J., & Wohlgenannt, I. (2020). A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda. *Computers & Education*, 147, Article 103778.

Stracke, C. M., Bothe, P., Adler, S., Heller, E. S., Deuchler, J., Pomino, J., & Wölfel, M. (2025). Immersive virtual reality in higher education: A systematic review of the scientific literature. *Virtual Reality*, 29(2), Article 64.

Wu, B., Yu, X., & Gu, X. (2020). Effectiveness of immersive virtual reality using head-mounted displays on learning performance: A meta-analysis. *British Journal of Educational Technology*, 51(6), 1991–2005.

Yu, Z., & Xu, W. (2022). A meta-analysis and systematic review of the effect of virtual reality technology on users' learning outcomes. *Computer Applications in Engineering Education*, 30(5), 1470–1484.