

Evaluación de los Retos en la Cooperación en Entornos de Realidad Aumentada Colaborativa.

Evaluation of Challenges in Cooperation within Collaborative Augmented Reality Environments.

Dulce Esperanza Nuñez Rodriguez^{1*}, Bryan Alberto Olmos Ramírez^{2*}, Gustavo Adolfo Murillo Gutiérrez^{5***}, Rocio Alfonsina Lizárraga Morales^{3**}, Uriel Haile Hernandez Belmonte^{4**}

¹Lic. En Artes Digitales, División de Ingenierías Campus Irapuato-Salamanca, Universidad de Guanajuato.

²Departamento de Arte y Empresa, División de Ingenierías Campus Irapuato-Salamanca, Universidad de Guanajuato.

³Departamento de Arte y Empresa, División de Ingenierías Campus Irapuato-Salamanca, Universidad de Guanajuato.

[¹d.nunezrodriguez@ugto.mx](mailto:d.nunezrodriguez@ugto.mx), [²ba.olmosramirez@ugto.mx](mailto:ba.olmosramirez@ugto.mx), [³ra.lizarragamorales@ugto.mx](mailto:ra.lizarragamorales@ugto.mx), [⁴uh.hernandez@ugto.mx](mailto:uh.hernandez@ugto.mx), [⁵ga.murillogutierrez@ugto.mx](mailto:ga.murillogutierrez@ugto.mx)

Resumen

La colocación multiusuario en experiencias de realidad extendida (XR) es clave para mejorar la interacción y colaboración efectiva en entornos educativos, profesionales y sociales, potenciando la sensación de presencia compartida y facilitando tareas conjuntas entre usuarios remotos o locales. Este proyecto presenta la implementación y evaluación de la colocación multiusuario en experiencias de Realidad Extendida (XR) utilizando Unity 6 y el Meta XR Core SDK, con pruebas en dispositivos Meta Quest 3 y 3s. Se diseñó y desarrolló una aplicación demostrativa tipo el juego de Tic-Tac-Toe también conocido en México como el juego del gato, en la que varios usuarios pueden compartir y sincronizar objetos virtuales en un mismo espacio físico a través de anclajes espaciales y servicios de red multijugador. El trabajo documenta el proceso técnico para configurar colocación, gestionar la propiedad de objetos virtuales y sincronizar interacciones en tiempo real. Los resultados muestran que la integración de Meta XR facilita el desarrollo de experiencias colaborativas, aunque persisten retos como la precisión del anclaje, la compatibilidad entre dispositivos y la gestión de propiedad en objetos interactivos. Se concluye que, pese a las limitaciones identificadas, la solución propuesta es viable para prototipos colaborativos en XR, y se recomienda continuar explorando optimizaciones y pruebas en escenarios más complejos.

Palabras clave: colocación; Cooperación;Realidad Aumentada ; Unity; Meta XR Core; multiusuario.

Introducción

La Realidad Extendida (XR, por sus siglas en inglés *Extended Reality*) es un campo emergente que integra tecnologías como la realidad virtual (RV), la realidad aumentada (RA) y la realidad mixta (RM), con el objetivo de fusionar lo físico y lo digital en experiencias interactivas e inmersivas (Nebeling & Madier, 2019). Estas tecnologías han ganado gran relevancia en la última década por su potencial para transformar áreas como la educación, el diseño, la medicina y la industria manufacturera (Bhattacharyya et al., 2019; Picard, Sun, & Botev, 2024).

En particular, la RV se caracteriza por sumergir completamente al usuario en un entorno digital generado por computadora, en el cual se sustituye por completo la percepción del entorno físico. Este tipo de experiencias han demostrado ser altamente efectivas para aplicaciones formativas, simulaciones de alto riesgo y entrenamiento profesional, gracias a su capacidad de generar presencia, concentración y retroalimentación inmersiva (Nebeling & Madier, 2019).

Dentro de este marco, la RA se distingue por permitir la superposición de objetos digitales sobre el entorno físico del usuario, en tiempo real, a través de dispositivos como teléfonos móviles, tabletas o visores dedicados (Apple Inc., 2025; Meta, 2024). Esta capacidad de extender el entorno con información digital ha convertido a la RA en una herramienta clave para el desarrollo de experiencias interactivas en contextos colaborativos.

Una de las aplicaciones más prometedoras de la RA es la colocación multiusuario, que permite a dos o más participantes visualizar e interactuar con los mismos objetos aumentados en un espacio físico compartido. Este tipo de interacción resulta especialmente valiosa en entornos educativos, simulaciones médicas y entrenamientos colaborativos, ya que promueve la coordinación espacial, el aprendizaje activo y el sentido de presencia compartida (Almerico, Jenkins, & Brooks, 2024; Masneri et al., 2023).

Sin embargo, el diseño de experiencias de RA colaborativa plantea importantes retos técnicos, tales como la alineación espacial precisa entre dispositivos, la sincronización en red y la gestión de las propiedades de los objetos interactivos (Nebeling & Madier, 2019; Murillo et al., 2025). Para superar estas barreras, se han desarrollado diversos frameworks y APIs, como Meta XR Core SDK o ARKit/ARCore, que ofrecen implementar la colocación y redes multijugador de forma más accesible y reproducible (Unity Technologies, 2025; Meta for Developers, 2025).

Recientes trabajos han propuesto marcos y arquitecturas de referencia para abordar estos desafíos. Por ejemplo, Picard, Sun y Botev (2024) presentan un framework de código abierto para realizar estudios multiusuario con XR.; Masneri et al. (2023) documentan una arquitectura educativa colaborativa en RA con fases de diseño, implementación y evaluación; y Bhattacharyya et al. (2019) exploran el diseño de juegos síncronos en AR con usuarios colocados. Estas propuestas coinciden en la necesidad de garantizar consistencia espacial y baja latencia para lograr una experiencia fluida y significativa.

En este contexto, el trabajo de Ruiz, Belmonte y González (2024) destaca la importancia de la sincronización espacial y la gestión eficiente de datos en red como factores clave para crear entornos XR colaborativos robustos. La evolución constante de estas herramientas demuestra el creciente interés por resolver los retos asociados a la RA compartida y facilitar su aplicación en escenarios reales.

El presente proyecto se sitúa en esta línea de investigación y desarrollo. Se implementó una aplicación colaborativa basada en un juego clásico (Tic-Tac-Toe) utilizando Unity y Meta XR Core SDK, con pruebas en dispositivos Meta Quest 3 y Meta Quest 3s. A través de esta aplicación, se exploran los desafíos de la colocación multiusuario, la gestión de propiedad y la sincronización de objetos en tiempo real, evaluando además la experiencia del usuario mediante un cuestionario estructurado. Este enfoque permite identificar tanto las limitaciones técnicas actuales como las oportunidades de mejora en el diseño de experiencias XR colaborativas.

Metodología

Este estudio empleó un enfoque experimental para analizar los retos de la colocación en XR colaborativa, eligiendo Unity 6 y *Meta XR All-in-One SDK* por su flexibilidad, compatibilidad con dispositivos Meta Quest y capacidad para integración multiusuario avanzada. A diferencia de otras soluciones como VRTK, MRTK o ARCore/ARKit, que ofrecen modelos de colocación ligados a servicios propietarios en la nube o con compatibilidad restringida a ciertos dispositivos, el SDK de Meta XR permite implementar sesiones colaborativas locales sin depender de suscripciones externas o infraestructura adicional. Esta característica resultó determinante para su elección, ya que facilita la implementación en condiciones controladas sin requerir conectividad externa ni dependencias de servicios de terceros (Meta Platforms, 2025; Bhattacharyya et al., 2019).

El aporte principal de esta investigación es presentar y validar una metodología práctica para la creación de experiencias XR colaborativas, resaltando tanto los desafíos técnicos como las ventajas frente a otras soluciones existentes. Esta aproximación metodológica se alinea con propuestas como la de Picard, Sun y Botev (2024), quienes destacan la importancia de marcos abiertos y modulares para el desarrollo de estudios multiusuario en XR, permitiendo una integración flexible y escalable de herramientas como Unity y sistemas de red para la creación de entornos colaborativos. Para estructurar este análisis y asegurar la claridad en cada etapa del proceso, el desarrollo del proyecto se organizó en cinco fases consecutivas: selección y justificación de herramientas, preparación del entorno, diseño y prototipado, implementación técnica, y validación de la experiencia colaborativa. Esta división permitió abordar de manera sistemática tanto los aspectos conceptuales como los retos técnicos que surgen en la creación de aplicaciones XR multiusuario.

Herramientas y Tecnologías

En el ecosistema XR actual, existen frameworks ampliamente utilizados como *Google ARCore* y *Apple ARKit*. *ARCore* implementa el servicio de *Cloud Anchors*, que permite a múltiples usuarios compartir anclajes en la nube para experiencias colaborativas persistentes entre dispositivos Android e iOS (Google LLC, 2025). Por otro lado, *ARKit* utiliza objetos tipo *ARAnchor* que representan posiciones y orientaciones en el mundo real, y puede persistir estos anclajes entre sesiones mediante *ARWorldMap*, asegurando coherencia espacial en dispositivos Apple (Apple Inc., 2025; PrepareFor, 2024).

Aunque ambos frameworks permiten la colocalización mediante servicios externos o suscripciones, su compatibilidad se limita generalmente a sus ecosistemas específicos. En contraste, el Meta XR SDK facilita la implementación de colocalización de manera local, sin requerir infraestructura en la nube, ofreciendo una solución flexible y eficiente para entornos colaborativos con visores XR Meta Quest, lo cual fundamentó su elección en este trabajo.

Se emplearon plataformas y dispositivos de vanguardia en XR: Unity 6 como motor principal de desarrollo; el *Meta XR All-in-One SDK* (versión 77.0.0), específicamente sus módulos de interacción, simulación, audio y gestión de avatares; y las bibliotecas *Unity Netcode for GameObjects* y *Unity Multiplayer Services* para la sincronización y gestión de sesiones multiusuario en red. La programación se realizó en C# y la validación se llevó a cabo utilizando visores XR Meta Quest de la serie 3 (Meta Quest 3 y 3s en adelante referidos como “visores XR Meta Quest”), garantizando compatibilidad total con las funcionalidades del SDK.

Estas herramientas ofrecen un entorno flexible y bien documentado, permitiendo implementar y evaluar soluciones de colocalización de forma accesible y replicable. Su uso garantiza que los resultados obtenidos sean relevantes tanto para el sector académico como para desarrolladores XR interesados en crear entornos colaborativos avanzados.

Preparación del entorno

Esta fase consistió en la preparación de una escena base en Unity 6 (versión 6000.0.45f1) para garantizar la compatibilidad con los visores XR Meta Quest, siguiendo las recomendaciones técnicas de la documentación oficial de Meta XR SDK (Meta Platforms, 2025). Debido a que los visores Meta funcionan con un sistema operativo basado en Android, fue necesario configurar el entorno de desarrollo para esta plataforma desde la ventana *Build Settings* de Unity (Figura 1).

Se importó el paquete *Meta XR All-in-One SDK* (versión 77.0.0), integrando únicamente los módulos necesarios para la experiencia colaborativa: interacción, simulación, audio y gestión de avatares. Con el objetivo de optimizar el rendimiento, se desactivaron funciones no esenciales como el mapeo de mallas, *raycasting* ambiental y oclusión.

La configuración incluyó la activación del *XR Plugin Management* con el proveedor *OpenXR*, una interfaz abierta y libre desarrollada por el Khronos Group que permite una integración estandarizada entre Unity y diversos dispositivos XR (Khronos Group, 2024). Esta interfaz es necesaria para habilitar funciones específicas del ecosistema Meta Quest, como soporte de dispositivo, detección de colocalización, gestión de sesiones, utilidades visuales y sincronización de anclajes espaciales (Figura 2).

También se integraron los *Unity Multiplayer Services* para la autenticación, gestión de red y sesiones compartidas. Para facilitar las pruebas sin necesidad de hardware físico, se utilizó el *Meta XR Simulator*, una herramienta incluida en el SDK que permite la simulación de interacciones XR directamente desde el editor de Unity. Finalmente, se calibró la escena ajustando las cámaras, el escalado de referencia y el sistema de seguimiento espacial, dejando el entorno listo para las fases experimentales siguientes (Figura 3).

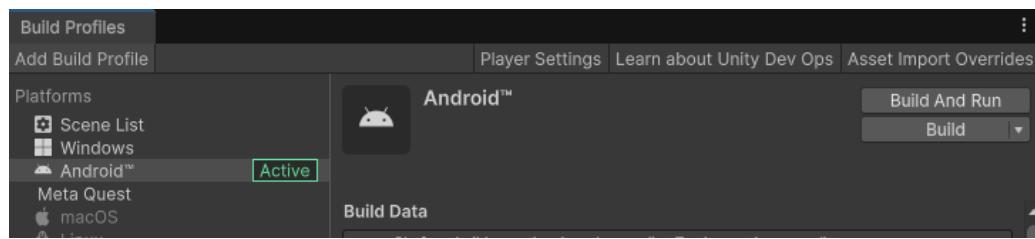


Figura 1. Configuración del perfil de compilación en Unity, con la plataforma Android seleccionada como destino activo. Esta selección es esencial para el despliegue en visores XR Meta Quest, ya que permite habilitar las funciones XR específicas del sistema operativo Android.

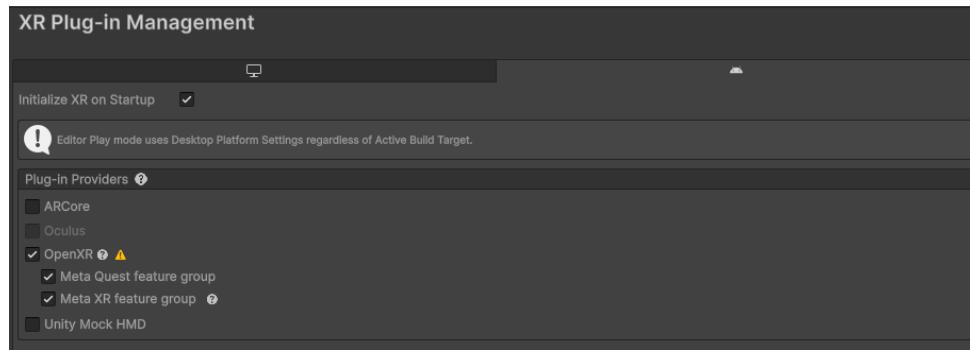


Figura 2. Panel de gestión de plug-ins XR en Unity. Se muestra la activación del proveedor OpenXR y del grupo de características específicas para visores XR Meta Quest. Esta configuración es necesaria para habilitar las funciones de colocalización, anclajes espaciales y otras características avanzadas de XR ofrecidas por el Meta XR SDK.

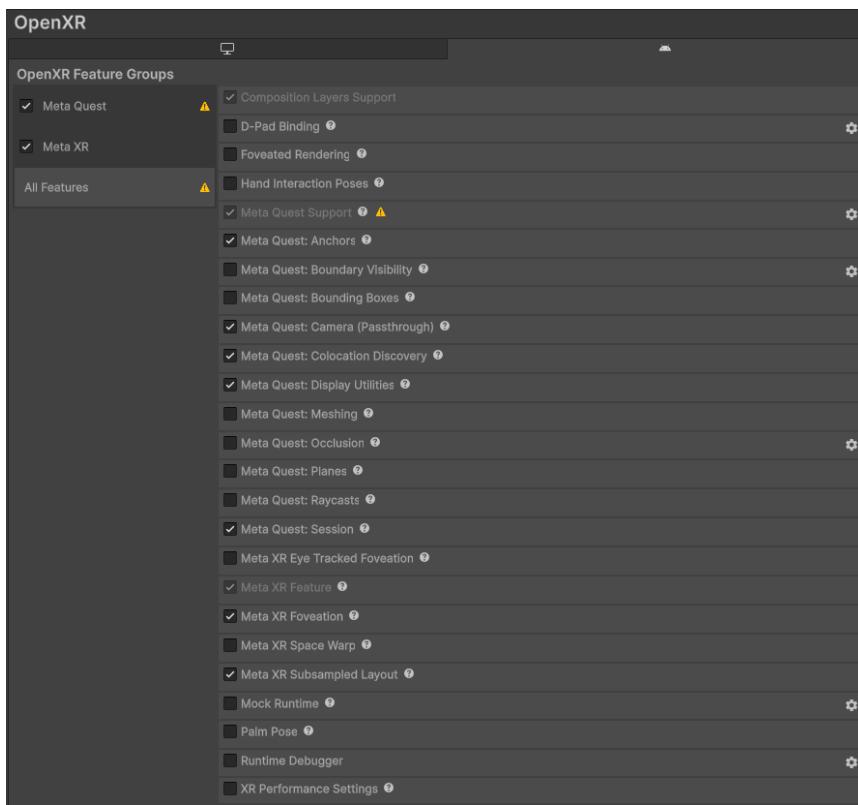


Figura 3. Activación de módulos esenciales del grupo OpenXR en Unity para Meta Quest, incluyendo soporte para colocalización, anclajes espaciales, Space Warp y utilidades de visualización, necesarios para experiencias XR colaborativas.

Se seleccionaron los componentes básicos del sistema colaborativo: objetos interactivos sincronizados (Networked Grabbable Object) para permitir la manipulación compartida, anclajes espaciales compartidas (Shared Spatial Anchors) para garantizar la alineación precisa de los objetos entre usuarios, módulos multijugador para la sincronización en tiempo real y la representación visual de los participantes mediante avatares virtuales. Esta integración de elementos permitió crear una base sólida para experimentar y evaluar la colocalización en un entorno XR colaborativo.

Prototipado

Se realizó una sesión de lluvia de ideas para explorar diversas opciones de implementación para mostrar la cooperación en un entorno colocalizado de realidad aumentada. Como resultado, se eligió desarrollar el juego clásico de Tic Tac Toe (gato) como aplicación demostrativa. La simplicidad de las reglas y las interacciones del Tic Tac Toe permite observar de manera clara y directa los retos de la colocalización, ya que requiere que dos usuarios comparten un mismo espacio virtual y puedan colocar, mover y visualizar piezas en tiempo real. Esto facilita el estudio de aspectos como la sincronización de los objetos entre usuarios, cómo se perciben los movimientos y cómo se mantiene la coherencia espacial, aspectos clave de la colocalización.

Además, este juego consta de interacciones que son básicas en entornos colaborativos, como la capacidad de tomar, soltar y posicionar objetos en un tablero común, acciones que representan la esencia de la mayoría de las aplicaciones multiusuario en XR. La selección de un juego con reglas simples y mecánicas claras permite modelar de forma controlada la interacción sincrónica entre usuarios, siguiendo principios propuestos por Bhattacharyya et al. (2019) para el diseño de experiencias colaborativas basadas en realidad aumentada. De este modo, se puede evaluar la alineación de anclajes espaciales, la latencia en la sincronización y la experiencia de usuario al interactuar con elementos compartidos.

Para fundamentar la elección y diseñar la dinámica esperada, se elaboró un prototipo físico en plastilina ilustrado en la Figura 4. Esta maqueta permitió visualizar y anticipar los desafíos prácticos de la colocalización, tales como la organización del espacio, la manipulación de piezas y la percepción compartida del tablero. Esta actividad fue clave para adaptar el diseño virtual a las necesidades de la experiencia colaborativa y justificar la implementación del juego Tic Tac Toe como herramienta efectiva para analizar los retos de la colocalización en XR.



Figura 4: Prototipo físico de Tic Tac Toe en plastilina, mostrando la disposición de las casillas, piezas y áreas personales para simular la interacción colaborativa.

Implementación Técnica del Sistema

La creación y sincronización de anclajes espaciales fue una de las acciones técnicas centrales en este proyecto. Un anclaje espacial es una referencia virtual persistente que se asocia a una ubicación y orientación específica en el espacio físico real, permitiendo que los objetos digitales se posicione y mantengan su alineación con respecto al entorno y a los usuarios, incluso si los dispositivos se mueven o cambian de contexto. Entre sus principales características destacan la persistencia a lo largo del tiempo, la precisión en la localización y la capacidad de ser compartidas entre múltiples dispositivos en una sesión colaborativa.

El uso de anclajes espaciales resulta fundamental para la colocalización, ya que proporcionan un punto de referencia común entre todos los participantes, asegurando que los objetos virtuales sean percibidos en la misma posición física desde cualquier visor. En este proyecto, los anclajes compartidos son generadas automáticamente por el sistema tomando como base la posición inicial del dispositivo anfitrión (host), lo cual resuelve el reto inicial de alinear y sincronizar el espacio virtual para todos los usuarios, tal como se detalla en la documentación oficial de Meta Platforms (2025), los anclajes compartidos son generadas automáticamente por el sistema a partir de la posición del visor anfitrión, eliminando la necesidad de

intervención manual por parte del usuario y asegurando una referencia espacial consistente entre dispositivos, garantizando una experiencia colaborativa consistente y precisa.

La sincronización de acciones se implementó mediante Unity Netcode for GameObjects, una biblioteca que facilita la comunicación y gestión de objetos en red entre múltiples usuarios dentro de Unity. Para simplificar la integración, se emplearon los módulos proporcionados por los Building Blocks de Meta XR, que incluyen componentes preconfigurados para las funcionalidades clave del entorno colaborativo.

En particular, se utilizó el Prefab Colocation, el cual integra automáticamente los siguientes elementos:

- Network Manager: Componente principal encargado de inicializar, gestionar y monitorear la red multijugador, coordinando la conexión entre dispositivos y el intercambio de datos en tiempo real.
- Local Matchmaking: Módulo que facilita la detección y emparejamiento automático de usuarios en la misma red local, permitiendo el acceso rápido y sencillo a sesiones colaborativas sin configuración manual.
- Colocation Controller: Controlador específico que administra las sesiones de colocalización, sincronizando las posiciones y acciones de los usuarios sobre los objetos compartidos y asegurando la alineación espacial entre todos los participantes.

Esta integración automatizada permite la gestión eficiente de la conectividad multijugador local, reduciendo la necesidad de lógica personalizada adicional para la creación o unión a sesiones, y favoreciendo una experiencia de usuario fluida y coherente.

- Gestión de Propiedad de Objetos:

Durante el desarrollo se identificaron retos asociados al sistema de propiedad de objetos en Unity Netcode. De forma predeterminada, únicamente el usuario propietario de un objeto tiene permisos para interactuar con él o modificar su estado en la red, lo que impedía que los clientes interactúen libremente con los objetos generados por otros usuarios o por el host. Para superar esta limitación, se optó por un enfoque basado en la generación de objetos propios de cada usuario. Cada participante puede generar sus propias piezas a través de un sistema centralizado denominado Object Manager, gestionado mediante el script CubeSpawner. Este script permite la creación controlada de piezas, asignando automáticamente la propiedad al usuario que solicitó su generación. La acción de generar una pieza se vincula a botones interactivos dentro del entorno virtual compartido, asegurando que la propiedad sea coherente con el usuario activo y permitiendo la manipulación autónoma de los objetos generados. Este enfoque garantiza el cumplimiento de las reglas de propiedad en red y mantiene la consistencia del estado de los objetos durante la sesión multijugador (Figura 5).

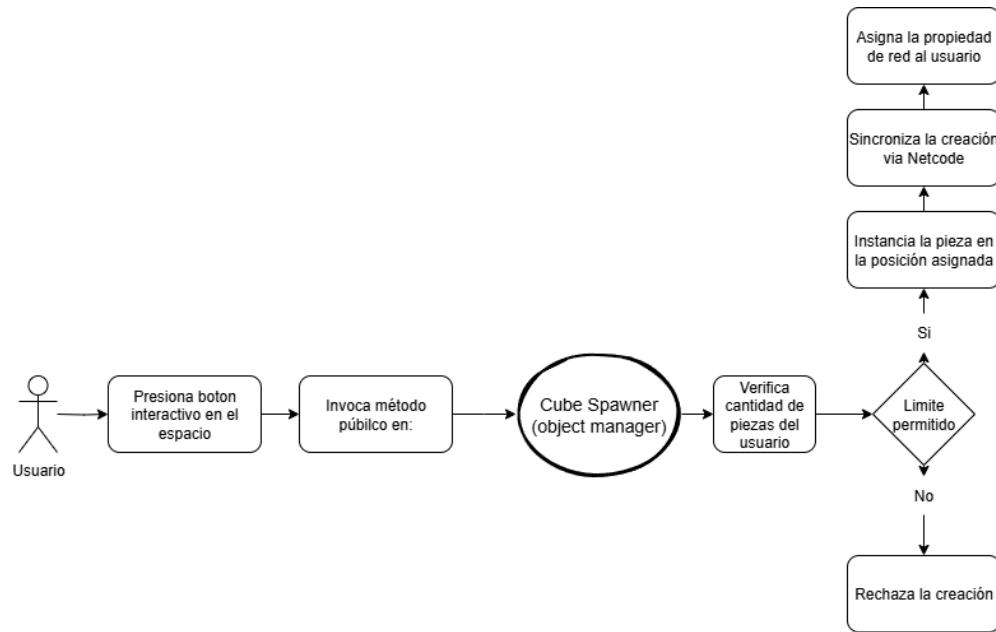


Figura 5: Diagrama del proceso de generación y asignación de propiedad de los objetos interactivos en la sesión multijugador.

Evaluación de la experiencia colaborativa

La evaluación de la experiencia colaborativa se realizó mediante un instrumento diseñado específicamente para captar la percepción subjetiva de los usuarios tras interactuar con el entorno Tic Tac Toe en XR. Este proceso fue implementado y modelado a partir del enfoque propuesto por Masneri et al. (2023), donde se enfatiza la importancia de analizar no solo el rendimiento técnico del sistema, sino también la experiencia del usuario desde una perspectiva educativa, interactiva y colaborativa.

El cuestionario fue concebido como una herramienta estructurada para recopilar valoraciones relacionadas con la usabilidad, la intuitividad del sistema, el funcionamiento de las mecánicas de colocalización, las dinámicas propias del juego Tic Tac Toe adaptado a RA, y el valor general de la experiencia lúdica en un entorno extendido. La estructura del instrumento responde a la necesidad de evaluar dimensiones complementarias del sistema, reconociendo que la percepción de presencia compartida, la claridad de las acciones disponibles y la fluidez de la interacción inciden directamente en la efectividad de los sistemas colaborativos en XR. Cada afirmación fue redactada para ser valorada por el usuario en una escala de tipo Likert del 1 al 5, donde 1 representa “totalmente en desacuerdo” y 5 “totalmente de acuerdo”. Asimismo, se incluyó un campo abierto al final del cuestionario para que los participantes pudieran compartir comentarios, sugerencias o reflexiones adicionales respecto a la experiencia.

El instrumento se dividió en seis secciones temáticas que agrupan un total de 24 ítems y una pregunta abierta (**Tabla 1**).

Este instrumento no solo permitió estructurar la evaluación de forma clara y coherente, sino que también ofrece una base replicable para futuras pruebas en otros entornos XR colaborativos. La combinación de escalas cuantitativas y espacio abierto cualitativo proporciona un panorama completo sobre la experiencia de usuario, clave para el desarrollo iterativo y centrado en el usuario de este tipo de sistemas.

Tabla 1. Cuestionario de evaluación de la experiencia colaborativa

Ítem	Enunciado	Categoría
1	El sistema me resultó simple de manejar.	Usabilidad
2	Pude aprender a usar la aplicación muy rápidamente.	Usabilidad
3	Las distintas funciones están bien integradas.	Usabilidad
4	Creo que podría usar la aplicación sin ayuda técnica.	Usabilidad
5	Me sentí seguro al interactuar con la aplicación.	Usabilidad
6	La interfaz me pareció muy intuitiva.	Intuitividad
7	No tuve que aprender nada nuevo para usarla.	Intuitividad
8	Pude entender al instante qué debía hacer en cada paso.	Intuitividad
9	La navegación dentro de la aplicación fue fluida.	Intuitividad
10	Las acciones (tomar, arrastrar, soltar, etc.) respondieron de manera precisa.	Intuitividad
11	Me sentí conectado con el otro jugador mientras compartíamos el mismo espacio virtual.	Funcionalidades de colocalización
12	Fue fácil coordinarme con el otro jugador para colocarnos de forma conjunta en el área de juego.	Funcionalidades de colocalización
13	La interacción con los objetos que veía el otro era fluida y sin interrupciones-	Funcionalidades de colocalización
14	Sentí que ambos teníamos el mismo nivel de control sobre los objetos compartidos.	Funcionalidades de colocalización
15	Percibí que podíamos colaborar eficazmente para avanzar en la partida.	Funcionalidades de colocalización
16	La experiencia de colocalización me hizo sentir que estábamos en el mismo lugar, aunque cada uno usara su propio visor.	Funcionalidades de colocalización
17	Fue natural generar mis piezas (X u O) y ubicarlas en la cuadrícula 3×3 sin dificultad.	Mecánicas del juego en AR
18	Las indicaciones flotantes (botón, letras, mensajes) eran claras y me orientaron durante todo el juego.	Mecánicas del juego en AR
19	La interacción con los objetos (botón, piezas, cuadrícula) se sintió rápida y sin interrupciones.	Mecánicas del juego en AR
20	Sentí que el entorno AR me sumergía más en el juego que si fuera en pantalla plana.	Experiencia lúdica general
21	Compartir el mismo espacio virtual hizo más entretenida y colaborativa la partida.	Experiencia lúdica general
22	Disfruté la experiencia y me sentí motivado a jugar varias rondas.	Experiencia lúdica general
23	Recomendaría esta experiencia a otras personas.	Experiencia lúdica general
24	¿Tienes sugerencias o comentarios adicionales sobre la experiencia de juego y la colocalización en esta aplicación?	Comentarios libres

Resultados

La implementación del entorno de realidad extendida colaborativa logró establecer una experiencia funcional de colocalización entre dos usuarios utilizando visores XR Meta Quest. El sistema permite a cada participante generar, manipular e interactuar con objetos virtuales de forma sincronizada, dentro de un espacio compartido, alineado espacialmente mediante anclajes generados de forma automática.

Sincronización espacial

Se comprobó que la creación automática de los anclajes espaciales mediante el SDK de Meta permite que ambos usuarios visualicen los elementos interactivos en posiciones consistentes. A lo largo de múltiples pruebas, se mantuvo la alineación entre dispositivos tras la inicialización de la sesión (Figura 7 y Figura 8).



Figura 7. Perspectiva desde dispositivo A.



Figura 8. Perspectiva desde dispositivo B

Interacción Multijugador

Los usuarios pueden generar sus propias piezas mediante botones interactivos vinculados al sistema *CubeSpawner* (Figura 9 y 10), lo cual garantiza la propiedad individual de los objetos y permite interacciones independientes y sincronizadas. Esta solución resolvió el problema inicial de manipulación limitada por restricciones del Netcode.



Figura 9. Usuario presionando botón para generar su pieza



Figura 10. Usuario sosteniendo su pieza y colocándola en un espacio dentro del tablero.

Estabilidad de sesión y Flujo general

La arquitectura implementada mediante prefabs de building blocks proporcionó una estructura robusta para la creación y conexión de sesiones. El sistema de matchmaking local funcionó sin fallas en múltiples ejecuciones.

Evaluación de los usuarios

Tras aplicar el instrumento descrito en la Tabla 1, se consolidaron las respuestas de diez participantes, cuyos promedios por categoría temática se muestran en la Tabla 2. En general, la experiencia fue valorada positivamente en todas las dimensiones evaluadas. Es importante destacar que aproximadamente el 70 % de los participantes utilizaron por primera vez un visor de realidad aumentada, lo cual aporta un contexto relevante a la interpretación de los resultados, ya que sugiere que el sistema fue comprensible incluso para usuarios sin familiaridad previa con tecnologías XR.

Tabla 2. Resultados promedio de la evaluación por categoría

Categoría	Promedio
Usabilidad	4.7
Intuitividad	4.5
Funcionalidades de colocalización	4.6
Mecánicas del juego en AR	4.7
Experiencia lúdica general	4.9

Conclusiones

El presente trabajo permitió validar con éxito la implementación de un entorno colaborativo basado en la colocalización espacial utilizando tecnología de realidad extendida (XR). Las implementaciones técnicas realizadas (documentadas mediante evidencias visuales en las Figuras 7 a 10) demuestran que las herramientas seleccionadas (Meta XR SDK, Unity y Netcode) resultan efectivas y adecuadas para desarrollar interacciones colaborativas síncronas en redes locales, destacando particularmente la funcionalidad automática de generación y sincronización de anclajes espaciales, así como la gestión eficiente de propiedad de objetos virtuales.

Entre los aprendizajes fundamentales destaca la importancia crítica de una correcta preparación inicial del entorno de desarrollo. La configuración detallada del entorno Android y la integración precisa de las características del SDK resultaron esenciales para garantizar la viabilidad y estabilidad técnica del sistema de colocalización. Asimismo, el uso de visores XR Meta Quest permitió observar y validar en condiciones reales la eficacia del sistema implementado, evidenciando claramente el potencial de interacción colaborativa en tiempo real entre usuarios ubicados en un mismo espacio virtual.

La evaluación realizada con usuarios proporcionó resultados mayoritariamente positivos en todas las dimensiones analizadas. Se destacó la accesibilidad e intuitividad del sistema, especialmente entre usuarios sin experiencia previa en XR, aunque también se identificaron áreas concretas susceptibles de mejora según la percepción de usuarios más experimentados. Particularmente relevante fue la observación del impacto directo que tiene la correcta sincronización y alineación espacial sobre la calidad de la experiencia percibida por los usuarios, lo que subraya la importancia técnica y perceptual de estos aspectos en sistemas colaborativos XR.

Finalmente, como principales limitaciones identificadas en este estudio, destacan las restricciones técnicas institucionales relacionadas con la conectividad y accesibilidad a ciertos servicios específicos, como los ofrecidos por Meta XR, las cuales impactaron momentáneamente en el desarrollo de pruebas. Para futuras investigaciones, se recomienda asegurar previamente la infraestructura técnica necesaria y consultar regularmente la documentación técnica oficial actualizada. Asimismo, se subraya la importancia de seguir un proceso estructurado y sistemático en todas las etapas del desarrollo, lo que facilitará una implementación más fluida y eficiente de sistemas colaborativos en entornos XR.

Bibliografía/Referencias

Almerico, N., Jenkins, H., & Brooks, M. S. (2024). Collaborative Augmented Reality in Higher Education: A Systematic Review. *Computers & Education*. [https://doi.org/10.1016/S0003-6870\(24\)00137-6](https://doi.org/10.1016/S0003-6870(24)00137-6)

Apple Inc. (2025). Creating a Collaborative Session. <https://developer.apple.com/documentation/arkit/creating-a-collaborative-session>

Bhattacharyya, P., Nath, R., Jo, Y., Jadhav, K., & Hammer, J. (2019). Brick: Toward A Model for Designing Synchronous Colocated Augmented Reality Games. Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19), 1-9. <https://doi.org/10.1145/3290605.3300553>

Masneri, S., Domínguez, A., Pacho, G., Zorrilla, M., Larrañaga, M., & Arruarte, A. (2023). A collaborative AR application for education: From architecture design to user evaluation. *Virtual Reality.* <https://doi.org/10.1007/s10055-023-00731-4>

Meta for Developers. (2024). Meta XR Core SDK Documentation. <https://developer.oculus.com/documentation/unity/meta-xr-core-sdk>

Murillo Gutierrez, G. A., Jin, R., Ramirez Paredes, J. P. I., & Haile Hernández Belmonte, U. (2025). A Framework for Collaborative Augmented Reality Applications. Companion Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D'25). <https://doi.org/10.1145/3722564.3728390>

Nebeling, M., & Madier, K. (2019). 360proto: Making Interactive Virtual Reality & Augmented Reality Prototypes from Paper. Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19), 1-13. <https://doi.org/10.1145/3290605.3300826>

Picard, S., Sun, N., & Botev, J. (2024). XR MUSE: An Open-Source Unity Framework for Extended Reality-Based Networked Multi-User Studies. *Virtual Worlds,* 3, 404-417. <https://doi.org/10.3390/virtualworlds3040022>

Unity Technologies. (2025). Unity Manual – Multiplatform XR Development. <https://docs.unity3d.com/Manual/XR.html>

Google LLC. (2025). Cloud Anchors allow different users to share AR experiences. Google Developers. Recuperado de <https://developers.google.com/ar/develop/cloud-anchors>

PrepareFor. (2024). What is ARAnchor exactly? Stack Overflow. Recuperado de <https://stackoverflow.com/questions/52893075/what-is-aranchor-exactly>

Apple Inc. (2025). ARWorldMap and ARAnchor persistence in ARKit. Apple Developer Documentation. Recuperado de <https://developer.apple.com/documentation/arkit/arworldmap>

Khronos Group. (2024). OpenXR specification overview (Version 1.1). Khronos OpenXR. Recuperado de <https://www.khronos.org/openxr>