
MANUAL DE USUARIO: CONTROL ROBOT UR MEDIANTE LAS HOLOLENS 2

Beca de Colaboración:
Álvaro Javier Orcajo Domingo

Curso 2023-2024

ÍNDICE

1.	Introducción.....	2
2.	Puesta en Marcha y Configuración.....	3
2.1.	Tipo de conexión.....	3
2.2.	Configuración de IP	3
2.3.	Modelo del Robot. Seguridad	5
2.4.	Anclaje.....	6
2.5.	Home.....	7
3.	Paneles de la interfaz. Utilidad	7
3.1.	Panel de Conexión.....	7
3.2.	Panel de Diagnóstico.....	8
3.3.	Panel Joystick.....	8
3.4.	Panel Programación del Robot	9
3.5.	Panel de Movimiento del robot con la mano	10

1. Introducción

En esta beca de colaboración se ha hecho una aplicación para la innovación en la enseñanza de robótica mediante realidad virtual, en este caso mediante el dispositivo de Microsoft HoloLens 2. Esta aplicación, realizada con la herramienta de desarrollo Unity junto al Mixed Reality Toolkit v3 de Microsoft, enseñará a los alumnos las diferentes tipos de movimientos (MoveJ, MoveL, MoveP) que dispone un robot de Universal Robots UR3e.

Además, mediante la realidad mixta, se podrá ver el rastro del recorrido exacto realizado por el robot, creando así un entorno inmersivo y vivencias únicas para el usuario, con una gran retención de información y por lo tanto aprendizaje. Esta última afirmación se ha demostrado mediante mi trabajo de fin de grado realizado durante 2022-2023 que se puede ver la documentación siguiente:

[Trabajo de Fin de Grado – Álvaro Javier Orcajo Domingo](#)

Este proyecto no solo se dispone de la innovación en la enseñanza, sino que también permite controlar el robot en realidad mixta, un proyecto innovador que puede servir como base para otros futuros proyectos. Potenciando así también el aprendizaje de tanto alumnos como usuarios de todo el mundo en el ámbito de comunicación mediante sockets, entre un robot UR y la Realidad Mixta o Unity.

Destacar el agradecimiento al [proyecto de Roman Parak](#), que me ha servido como aprendizaje para la conexión de sockets.

2. Puesta en Marcha y Configuración

Primero de todo, para poder conectar el robot y que funcione correctamente, hay que tener en cuenta varios factores. Porque la aplicación se ha intentado hacer con la mayor simpleza posible, pero primero es necesario configurar el dispositivo.

Por razones de seguridad, antes de hacer cualquier movimiento del robot mediante esta aplicación, es recomendable poner la velocidad al 35% en el Teach Pendant. Además prestar especial atención al apartado 2.3. Modelo del Robot.

Por otro lado el gripper (en caso de tenerlo) inicialmente tiene que estar abierto, además de estar conectada a la salida digital de herramienta.

2.1. Tipo de conexión

Existe la conexión a través de un **cable ethernet**, que en caso de conectarse desde un ordenador se necesitará un puerto para tal cable. Tanto en las HoloLens 2 como otros ordenadores pueden no disponer de este puerto, en ese caso simplemente se tendría que utilizar un adaptador de USB a cable Ethernet.

Se puede realizar la comunicación mediante **conexión inalámbrica**, que bajo mi punto de vista es la más cómoda, pero el robot necesitaría un router conectado.

2.2. Configuración de IP

Si se quiere utilizar desde el ordenador en la aplicación de Unity (está destinado para realidad mixta pero se puede utilizar desde el ordenador), se tendrá que modificar la IP (según sea inalámbrica o ethernet) en los ajustes de red en el ordenador.

En cambio si se conectan a las HoloLens 2 (que es lo usual), se tendrá que configurar en los ajustes dentro del sistema operativo de las HoloLens 2. En las siguientes imágenes se observa los parámetros a cambiar, en la figura 1 en el caso del ordenador, en la figura 2 en caso de las HoloLens 2:

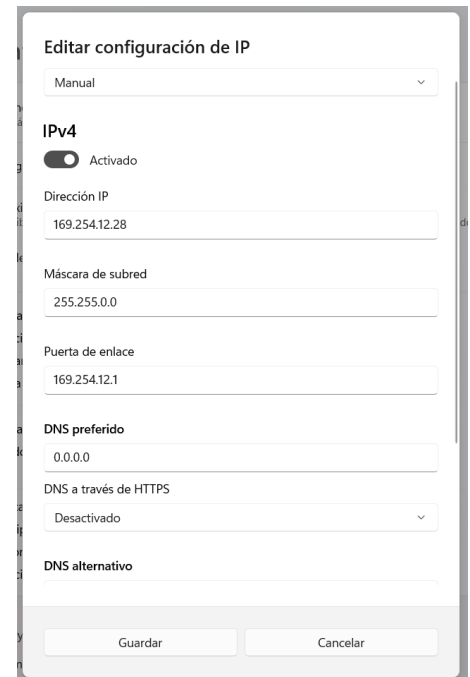
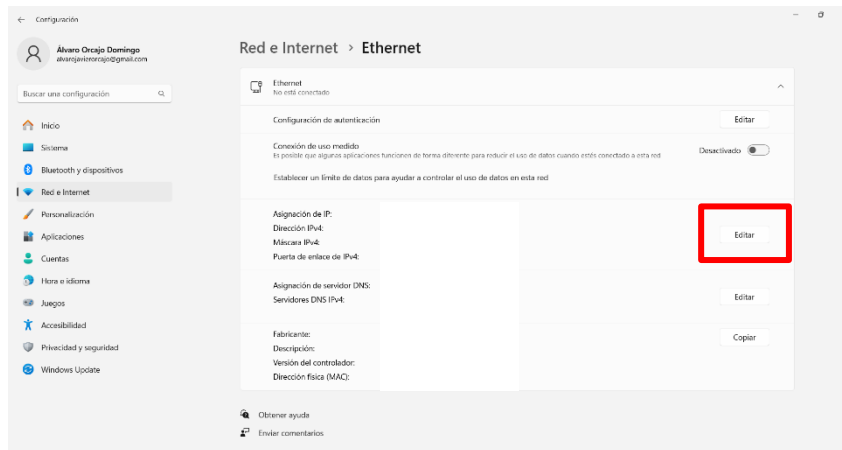


Figura 1. Configuración IP en ordenador

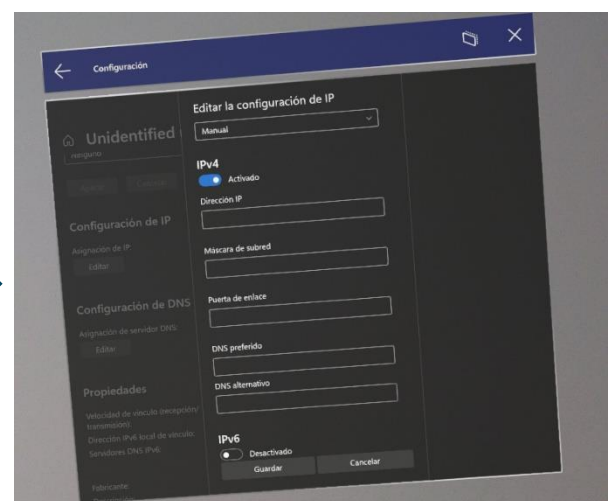
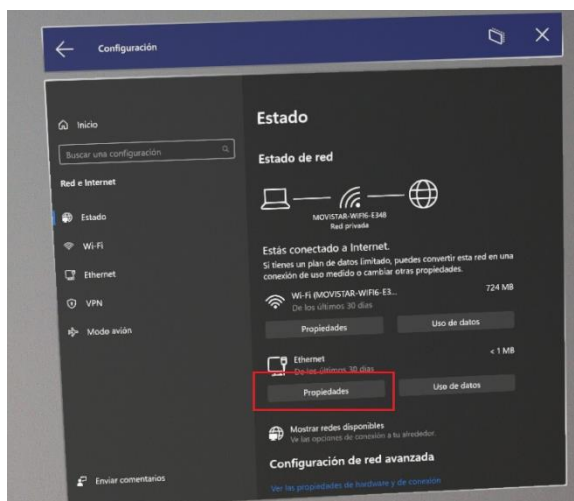


Figura 2. Configuración IP en las HoloLens 2

Una vez hecho lo anterior, hay que indicar la dirección IP dentro de la aplicación, que tiene que ser la misma que la del dispositivo. Se ha hecho una interfaz con un teclado con posibilidad de teclearlo, pero si se va a utilizar la aplicación siempre para una misma dirección IP, se recomienda predefinirlo dentro de Unity en el ordenador antes de compilar y depurar a las HoloLens 2, ya que será mucho más cómodo. Esto se cambiaría en: `Assets\UR_Control_Scripts\UIPanel_Control.cs` dentro de la función "void Start()" la variable llamada "ip_address_txt.text" (en la línea 95 del código). Si se hace esto, simplemente bastará con darle al botón conectar en el panel de conexión.

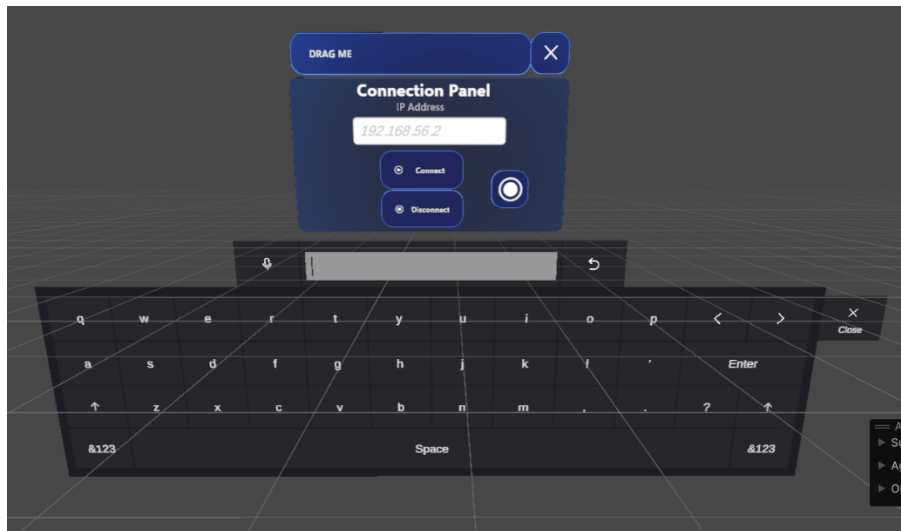


Figura 3. Panel de Conexión, donde se introduce la dirección IP

Una vez configurado, existe otra forma de conectarse con el robot y las HoloLens 2, esto es iniciando la aplicación en el ordenador desde Unity y utilizar la aplicación Holografic Remoting en las HoloLens 2. Esto hará que no sea necesaria la configuración de IP en las HoloLens 2, ya que pueden dar errores de configuración en el dispositivo. En mi experiencia, una de las veces en las que se configuró la dirección IP en el dispositivo, por un motivo que desconozco al volver a poner en automático la configuración de red resulta ser que no se configuraba automáticamente y seguía con la configuración manual, teniendo así problemas de red por establecer una dirección IP que no corresponde con la WIFI. Hasta que no me conecté con Windows Connect no pude solucionar el error.

2.3. Modelo del Robot. Seguridad

La utilización del robot UR3e con esta aplicación es crucial e imprescindible, tanto para la seguridad de los usuarios como para que funcione correctamente. Si se utiliza otro modelo podrían haber movimientos totalmente imprevisibles, incluso si la cinemática estuviese bien hecha, daría errores como los de la siguiente imagen:

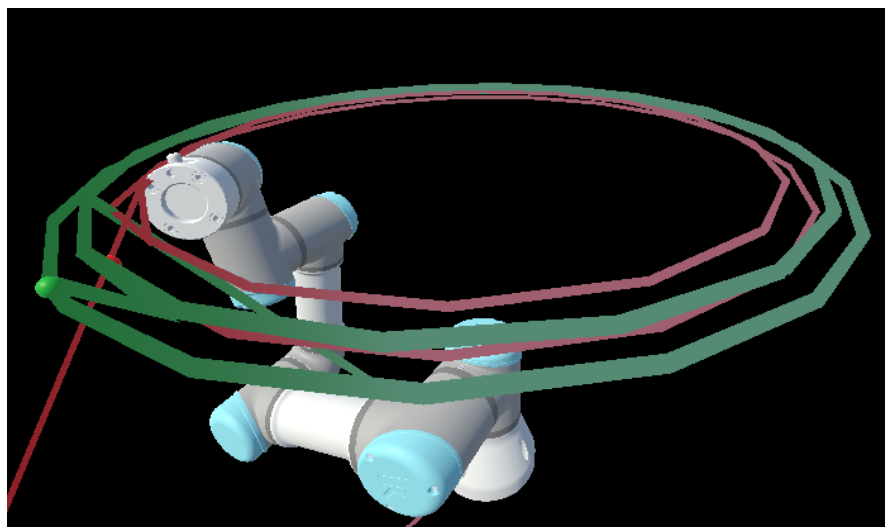


Figura 4. Error con modelo UR3 (con cinemática hecha)

Por otro lado, para garantizar seguridad, se ha hecho una esfera que representa el alcance del robot en la que los movimientos solo se mandarían en caso de que estén dentro de ella.

2.4. Anclaje

El anclaje es importante para tener el correcto origen de coordenadas, el anclaje utilizado es uno manual, el cual se coloca en la base del robot mediante las propias manos en realidad mixta. El anclaje se trata de un objeto diseñado para que sea lo más sencillo y práctico a la vez. Se muestra en la siguiente imagen:

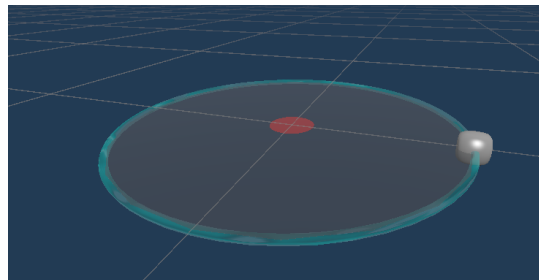


Figura 5. Objeto para hacer el anclaje del robot

Este objeto se puede manipular con las manos, tanto a distancia como cerca, como si de un objeto real se tratase. El objeto se colocará en la base del robot, de forma que la base coincida con el círculo rojo del objeto del anclaje como se muestra en la siguiente imagen:

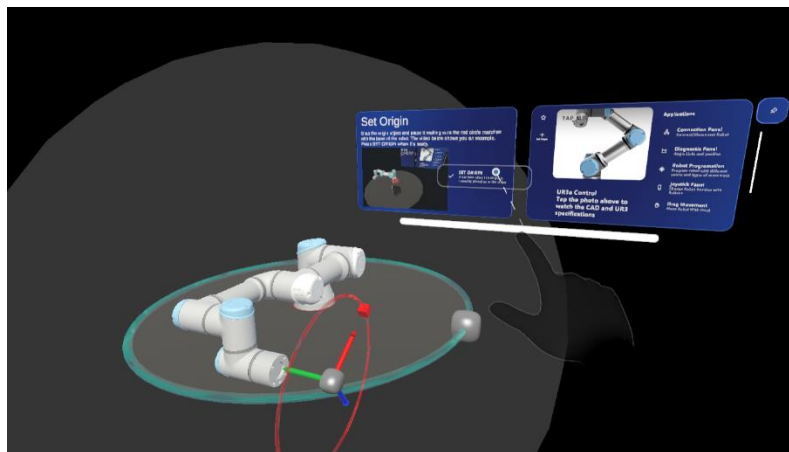


Figura 6. Ejemplo en el que el objeto de anclaje está situado a la perfección en la base del robot

Además, para que el usuario sepa lo que hacer, hay un texto explicativo junto con un vídeo.

Con esto se consigue que el eje de coordenadas de la aplicación casi coincida con el real del robot. Debido a que es un anclaje manual hecho con realidad mixta, habrá un error de como máximo 5 cm si se hace bien.

El procedimiento sería colocar el objeto de anclaje como se indica y luego darle al botón “set origin”. En el caso de necesitar una rotación, existen dos botones para ello, estos botones harán girar el robot

5 grados en la dirección deseada. Si se necesitase cambiar algo del anclaje, siempre estará disponible el botón de set origin en el panel principal.

2.5. Home

Se ha implementado un botón en el panel principal llamado “Reset Scene”, con el que se reinicia la aplicación, en caso de que haya algún error o bug que no se haya registrado.

3. Paneles de la interfaz. Utilidad

Antes de describir a los paneles de funcionalidades, se ha creado una interfaz que gestione toda la aplicación, con la que se podrá navegar a todas las funcionalidades existentes. Este panel principal es el siguiente:

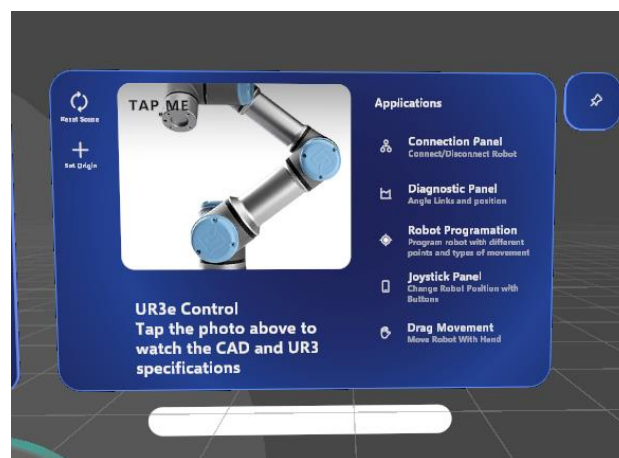


Figura 7. Panel Principal

Para el desarrollo de la aplicación, se ha decidido hacer 4 paneles con un quinto en el que se podría desarrollar en un futuro. Los paneles se describirán a continuación, se mostrará algún fallo existente del Mixed Reality Toolkit que he encontrado y además se mostrarán las posibles líneas futuras que podrían tener cada uno de ellos.

3.1. Panel de Conexión

En este panel se introduce la dirección IP del robot, para ello se ha implementado un teclado que actualmente se encuentra en desarrollo por parte de Microsoft. Esto significa que el teclado funciona pero puede no funcionar a la perfección. El panel se muestra en la figura 3.

Si se pulsa la barra de texto, se abrirá el teclado. Por otro lado los botones de conectar y desconectar inician o finalizan la comunicación con el robot UR mediante sockets. El otro símbolo muestra el estado de conexión, verde en caso de estar conectado y rojo en caso de no estarlo.

3.2. Panel de Diagnóstico

En este panel se muestran a tiempo real las posiciones articulares y cartesianas del robot. Tiene un panel superior para mover el panel con facilidad. En la figura 8 se muestra el panel.



Figura 8. Panel de Diagnóstico

3.3. Panel Joystick

En este panel se podrá mover el robot con comandos del URScript "speedl". Además dispone de botones para poder ajustar tanto la velocidad como la aceleración del robot, esto se pone porque hay ciertos cambios de coordenadas que necesitan de un gran movimiento de las articulaciones (necesidad de velocidades pequeñas) y por otro lado hay cambios de coordenadas que no necesitan de un gran movimiento de articulaciones (necesidad de velocidades altas). Los botones de movimiento moverán al robot según el eje elegido, luego los de rotación harán una rotación respecto al eje elegido. En la figura 9 se muestra el panel.

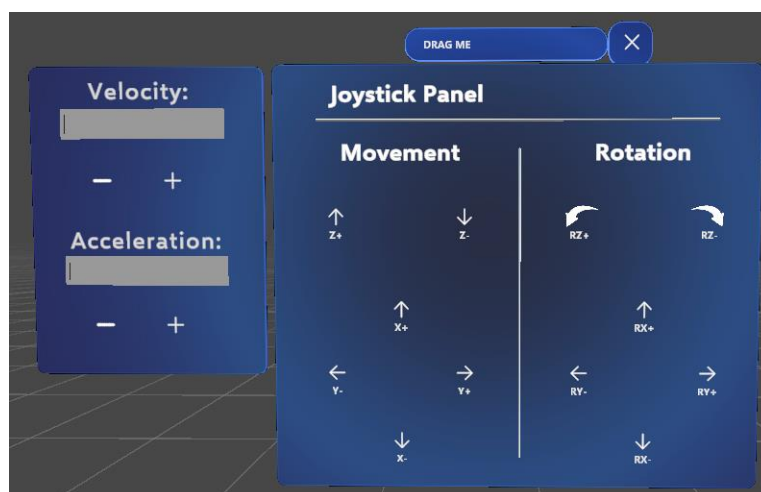


Figura 9. Panel Joystick

3.4. Panel Programación del Robot

En este panel se podrá programar el robot, mandándole varios puntos que se puede mover con la mano tanto desde cerca como desde lejos. Se destaca que hay un panel de texto que avisa de posibles errores, como puede ser el que el punto está fuera de rango. En ese caso el punto no se podrá mandar al robot por seguridad. Cuando se colocan los puntos que se quieran, estos se ejecutarán de manera secuencial, pulsando el botón de “send”.

Una línea futura de este panel, sería el crear una forma de mandar rotaciones, porque hasta ahora se mantienen las rotaciones iniciales. Como esta herramienta está destinada para la docencia a los alumnos de robótica, se recomienda el colocar el robot inicialmente de forma que el tipo de movimiento se trace con claridad manteniendo la rotación. Además de que se ha visto innecesario mandar rotaciones, al tener una finalidad didáctica de los tipos de movimientos. En la figura 10 se muestra el panel.

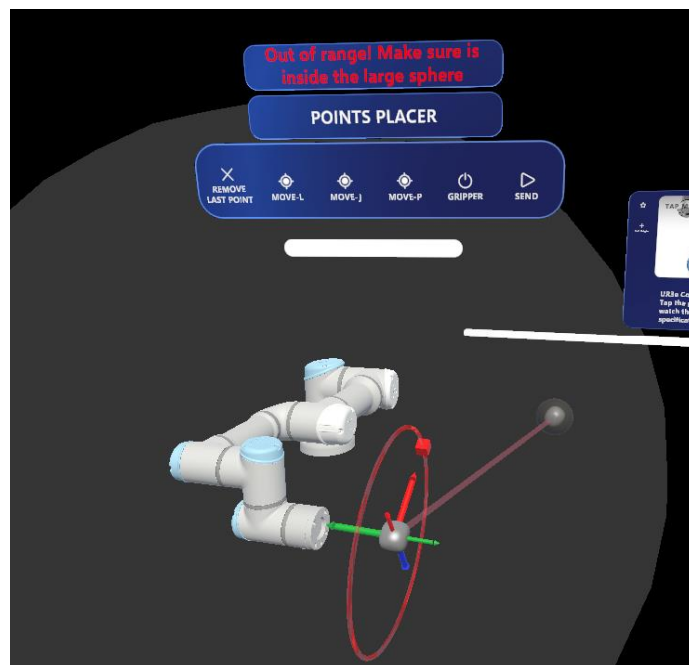


Figura 10. Panel de Programación del robot

Destacar que a la hora de crear un punto, se traza una línea que simula el movimiento que va a realizar el robot. Se pueden elegir tres tipos de movimiento, MoveL, MoveJ y MoveP. Luego hay una acción de gripper, que lo que haría sería mover la pinza o herramienta digital instalada en el robot.

Es importante decir que aparte de mandar puntos, el robot al moverse crea un rastro verde que marca el recorrido del robot en todo momento. Esto hace que se pueda anclar el CAD del robot lejos del robot real, hacer una programación del robot en el Teach Pendant y que al ejecutar el programa se pueda ver el recorrido del robot realizado por parte del objeto virtual con total seguridad y con gran aprendizaje al ver las diferencias entre tipos de movimiento. Esto se muestra en el vídeo de resultados mostrado en el apartado 5.

3.5. Panel de Movimiento del robot con la mano

Este panel se introduce como línea futura, ya que quedaba fuera del alcance de este proyecto. Consistiría en poder coger el extremo del robot, incluso cualquier joint del mismo y poder moverlo arrastrándolo con la mano, siempre manteniendo una velocidad que garantice la seguridad del usuario.

4. Editor de Unity

En caso de querer modificar el proyecto ya existente, se explica en los siguientes pasos como poder hacerlo:

1. Primero descarga el proyecto del repositorio github: [UR HoloLens2 Unity](#)
2. Una vez descargado, se recomienda descomprimirlo en C:/ ya que puede dar problemas de longitud de ubicación de archivo.
3. Se abre "Unity Hub", se da al botón "Add" y se elige la carpeta "UR_Hololens2" en la dirección \UR_Hololens2_Unity\UnityProjects
4. Una vez abierto el proyecto, en la ventana "Project" en Unity nos dirigimos a la carpeta de "Scenes" y abrimos la escena "PruebaUR".
5. Puede ser que nos pidan importar el paquete de TMP (es un paquete de caracteres). Una vez importado para que se apliquen los cambios, reiniciamos unity y volvemos a abrir el proyecto.
6. Una vez abierto estaría listo para editar a su gusto el proyecto. En caso de error al dar al play mode, tener siempre en cuenta si está o no activo el "Holographic Remoting", esto se podrá comprobar abriendo la ventana del mismo en "Mixed Reality/Remoting/Holografic Remoting for Play Mode"

5. Resultados

Los resultados observados en este proyecto se pueden ver gráficamente en el siguiente vídeo:

[Video – UR HoloLens2 Unity](#) <https://youtu.be/5jKLdiqAFml>

6. Contacto

En caso de tener alguna duda o problema, puedes contactar conmigo vía email, estaré encantado de ayudarte:

Álvaro Javier Orcajo Domingo

alvarojavierorcajodomingo@gmail.com