



UNIVERSIDAD IBEROAMERICANA
Estudios con Reconocimiento de Validez Oficial por
Decreto Presidencial Del 3 de abril de 1981

CIENCIAS DE LA INGENIERÍA

**Realidad mixta aplicada a sistemas de
automatización en la Industria 4.0**

T E S I S

PARA OBTENER EL GRADO DE MAESTRO EN CIENCIAS
DE LA INGENIERÍA

PRESENTA:

Joel Arango Ramírez

DIRECTOR

Dr. Eduardo Gamaliel Hernández Martínez

México, CDMX, Febrero 2024

Agradecimientos

Desde la idea, investigación, planeación, e implementación del proyecto pensaba los nombres de las personas que tenía que plasmar por todo el apoyo, tiempo y experiencia que me transmitieron, pero ahora que me encuentro en el cierre, creo que tengo que decir que son tantos los nombres y personas que sería injusto omitir a alguno.

He tenido apoyo desde hace muchos años para llegar a esto, actualmente gente que recién entró a mi vida ha contribuido con un pequeño o enorme grano en este gran logro.

Estos logros no son de una persona, es de encontrarse en el ecosistema correcto con todas las personas, familia, amigos, lugares, trabas, momentos que piensas que no puedes más, te vas al suelo pero ese equipo te da su confianza y apoyo logrando que te levantes de nuevo y luches más fuerte cada día, pero solo existe una cosa importante ¡jamás rendirse!, eso nunca es opción.

Agradezco la paciencia, la ayuda, aunque en ocasiones doy mucha lata pero puedo asegurar que todo lo hago con el corazón, jamás imaginé llegar hasta este punto y ahora pienso que apenas es el inicio de todo lo que viene por descubrir, experimentar, echar a perder y crear, solo sé que los éxitos son paso a paso y que cuando son excelentes no son fáciles, para ser sincero me encanta que no sea fácil, eso es lo que nos lleva a probar hasta donde podemos, siempre lo he dicho y lo repito el límite está en nosotros, la lucha más difícil es convencernos qué tan fuertes podemos ser y todo lo que podemos ofrecer a nuestro entorno, ese pequeño grano de arena que pensamos que no sirve en ocasiones de nada pero la vida me ha demostrado que los pequeños movimientos son los que más impacto causan.

Gracias a todo este proceso de vida que me ayudó a comprender algo que no soltaré jamás, mi capacidad, mi valor y mi potencial, así como todo lo que puedo ofrecer de ahora en adelante, me aferraré a dedicar mi vida entera a hacer lo que me gusta más y para lo que estoy muy dispuesto a prepararme con todo mi ímpetu y fuerza, reitero apenas siento que empiezo pero creo que en mi mente la ruta es clara ahora.

En verdad este proyecto me enseñó demasiado, me acercó a personas impresionantes que admiro y respeto por toda esa capacidad y humildad con la cual apoyan y comparten ese conocimiento, buenos consejos siempre empujando hacia adelante, en verdad mis respetos, son increíbles agradezco todos los consejos y toda la paciencia de corazón la agradezco.

Antes que nada una disculpa si no puntualizo nombres, pero en verdad no puedo más que decir que este proyecto no lo hice yo, lo hicimos todo un equipo que jugó su rol en cada uno de sus apartados de manera impresionante, sin cada uno de ellos esto jamás se hubiera logrado, en verdad millones de agradecimientos a todos y cada uno, no solo personas también los espacios que me permitieron permanecer horas intentando, fallando y creando.

Personalmente, me siento muy orgulloso y agradecido de la Universidad, los que me conocen más, saben que lo digo con el corazón, agradezco y amo mi Universidad, mi hogar

que me ha dado tanto y espero retribuir un poco de todo eso que me ha ofrecido en mi vida.

En verdad, estaré eternamente agradecido con todos los que hicieron posible este logro, me queda claro sin todo ese enorme equipo esto jamás se hubiera logrado.

Los logros nacen cuando nos atrevemos a comenzar y tenemos la garra de no dejarlos escapar, por más que pensemos que no es posible, por más obstáculos que encontremos, todo es posible si luchas con todas tus fuerzas.

Inmensas gracias a todos.....

Índice general

Índice general	IX
Índice de figuras	XI
Índice de tablas	XV
1 Resumen	1
1.1 Resumen ejecutivo	1
1.2 Justificación	1
1.3 Objetivo General	2
1.4 Objetivo Específico	3
1.5 Originalidad	3
1.6 Metodología	4
1.7 Cronograma de Actividades	5
1.8 Infraestructura Disponible	6
2 Estado del Arte	7
2.1 Introducción	7
2.2 Antecedentes Históricos	7
2.2.1 Realidad Virtual	8
2.2.2 Realidad Aumentada	11
2.2.3 Realidad Mixta	13
2.3 Clasificación de Robots	14
2.3.1 Robots Fijos o Estacionarios	16
2.3.2 Cinemática Directa e Inversa	18
2.3.3 Cobot UR3	20
3 Conceptos y Tecnologías en el contexto de la Industria 4.0.	23
3.1 Introducción	23
3.2 Aplicación	24
3.2.1 Aplicación en la Industria 4.0	24

3.2.2	IoT	25
3.3	Tecnología para la realidad virtual y aumentada	25
3.3.1	Hardware y Software	26
3.3.2	Dispositivos para la realidad virtual y aumentada	26
4	Diseño y Protocolos	31
4.1	Introducción	31
4.2	Tecnología para Cobots	31
4.2.1	Modbus	31
4.2.2	URX	33
4.2.3	Protocolo TCP/IP	34
4.3	Tecnología para la Realidad Mixta	35
4.3.1	Unity	35
4.3.2	Blender	37
4.3.3	Visual Studio	38
4.3.4	Visual Studio Code	38
4.3.5	HoloLens 2	38
5	Desarrollo	41
5.1	Introducción	41
5.1.1	Modelado Digital UR3	41
5.1.2	Interfaz de Integración HoloLens 2	46
5.1.3	Desarrollo de la plataforma en Unity	51
5.1.4	Menú	52
5.1.5	Firestore	56
5.1.6	Panel Observa	56
5.1.7	Panel práctica	59
5.1.8	Panel programar	60
5.1.9	Adquisición de Datos UR3	64
5.1.10	Implementación HoloLens 2	68
5.1.11	Cuestionario de evaluación	69
6	Resultados	71
6.1	Introducción	71
6.1.1	Contribución a la Industria	77
6.1.2	Contribución a la Educación	79
7	Conclusiones	83

Índice de figuras

1.1	"Metodología"	4
2.1	"Gafas de Pigmalión"	10
2.2	"A lo largo del río durante el festival Qingming"	10
2.3	"Fotografía Estereoscópica"	10
2.4	"Estereoscopio"	11
2.5	"Sensorama"	11
2.6	"I Met A Morph" instalación educativa de Myron Krueger	12
2.7	Estructura mecánica de brazo robótico	16
2.8	Tipos de robots	17
2.9	Cinemática Directa e Inversa	18
2.10	UR3	21
3.1	Google glass 2	26
3.2	Microsoft HoloLens 2	27
3.3	Apple Vision Pro	27
3.4	Meta Quest Pro y Quest 3	28
3.5	Epson Moverio BT-300	28
3.6	Vuzix M3000 XL	29
4.1	Red Modbus	32
4.2	"Capas Protocolo TCP-IP"	35
4.3	3D Unity	36
4.4	UR3 Modelo 3D Unity	36
4.5	Blender	37
4.6	UR3 Modelo 3D	37
4.7	Microsoft	38
5.1	Flujo de la plataforma	42
5.2	UR3 Modelo 3D	43
5.3	UR3 Modelo 3D eje coordenadas en eje "z"	43

5.4	Grados de libertad	44
5.5	Tabla de Denavid-Hartenberg	44
5.6	Modelo 3D	45
5.7	Modelo 3D UR3.glb	46
5.8	Crear nuevo proyecto 3D	47
5.9	Build Settings	47
5.10	MRTK Logo	48
5.11	MRTK	48
5.12	MRTK Project Path	48
5.13	MRTK Discover Features	49
5.14	MRTK Project Configurator OpenXR	50
5.15	MRTK Project Settings	50
5.16	MRTK Interaction Prtofile	51
5.17	MRTK TextMeshPro	52
5.18	Escena Unity y Mixed Reality Toolkit	52
5.19	Assets del proyecto	53
5.20	Flujo de la plataforma	53
5.21	Microsoft Hand Interaction Profile	53
5.22	Flujo de conexión de módulos	54
5.23	Build Settings	55
5.24	Menú principal	55
5.25	Componentes anidados	55
5.26	Componentes no anidados	56
5.27	Firebase	56
5.28	Menú de aplicaciones	57
5.29	Modelo de observación en Unity	57
5.30	Conexión GameObject a script	58
5.31	Comportamiento Unity - UR3	60
5.32	Modelo de práctica	61
5.33	Prueba de manipular objetos Unity	61
5.34	Módulo Programación Unity	62
5.35	Botonera Unity	63
5.36	Jerarquía UR3 Unity	64
5.37	Articulaciones UR3 Unity	64
5.38	Función de acción de botón Unity	65
5.39	Diagrama de flujo adquisición de datos UR3	66
5.40	Librerías Python Visual Studio Code	66
5.41	Código interfaz gráfica de conexión	67

5.42	Conexión con el robot UR3	67
5.43	Configuración de compilación	68
5.44	Archivos ejecutables	69
5.45	Configuración Visual Studio	69
5.46	Carga correcta	69
6.1	Menú	72
6.2	Módulo Observar	73
6.3	Módulo Practicar	74
6.4	Módulo Programar	75
6.5	Programación con conexión a base de datos	76
6.6	Experiencia de uso de plataforma	78
6.7	Pruebas en Laboratorio	79
6.8	Universidad Iberoamericana	80

Índice de tablas

2.1	Tabla de valores de Denavid-Hartenberg	19
6.1	Latencia UR3 a Firebase	73
6.2	Pruebas de Usuarios	77

Capítulo 1

Resumen

1.1 Resumen ejecutivo

Se presenta el desarrollo de una plataforma de realidad mixta que posibilita a un usuario, la operación remota de dispositivos de robótica y automatización, dirigido a la industria a través de un entorno virtual. Esta plataforma constituye un sistema ciberfísico que permite la teleoperación de sistemas automatizados de forma más eficiente, productiva y segura para un operador y la industria. Se implementa el control de brazo manipulador UR3 permitiendo la observación, capacitación y ejecución del funcionamiento del UR3 desde cualquier lugar, solo contando con una red con conexión a internet estable y adecuado ancho de banda.

Al proporcionar a un operador una plataforma de operación en tiempo real con tecnología en realidad aumentada, mejora su percepción y comprensión de la operación de los dispositivos, esto lo conduce a una toma de decisiones más eficiente.

La hipótesis central que este trabajo pretende abordar, es la posibilidad de reforzar habilidades; proporcionando una mejor representación visual para los operadores del entorno y las tareas a ejecutar, reduciendo fatiga y exposición de zonas peligrosas con el uso de realidad mixta y robots teleoperados.

1.2 Justificación

Este es un proyecto de maestría debido a la complejidad del planteamiento, desarrollo e implementación. Utiliza diferentes técnicas, entre las que destacan:

- Programación de alto nivel
- Manejo de bases de datos
- Servicios y seguridad en la nube
- Big Data
- Técnicas de control avanzado y robótica

Estos temas no son abordados a nivel licenciatura al requerir un conocimiento más amplio y técnico de estas nuevas tecnologías. El software interactivo de realidad mixta que plantea este proyecto involucra el conocimiento de nuevas tecnologías, que no necesariamente están establecidas y aplicadas para el área de la ingeniería y la industria. La implementación involucra nuevos conocimientos prácticos y experimentación que se logra a través de estudios de posgrado. En el año 2019 se registró una pandemia a nivel mundial, lo cual obligó llevar a gran parte de la industria a la suspensión parcial o total de las actividades productivas, los sectores más afectados fueron las actividades inmobiliarias, empresariales y de manufactura, entre otras [1]. El impacto en la sociedad de todo el mundo llevó a poner en práctica el distanciamiento social para mitigar el problema de salud pública, instruyendo a gran parte de la industria y a los órganos administrativos a limitar actividades a lo más esencial posible, esto deriva en la adaptación de las necesidades de la demanda en la industria, permitiendo en algunos rubros laborar desde casa para así evitar la propagación de la enfermedad, adaptando a las empresas a nuevos métodos de manejo y procesos a distancia [2].

Basado en estos puntos podemos localizar una necesidad y otorgar soluciones, permitiendo la teleoperación permitiendo mitigar este tipo de problemas, con esto un operador puede realizar sus funciones a distancia en algo que hasta hoy en día solo ha sido posible realizar físicamente en la planta, manteniendo la operación de actividades de forma segura. Este es solo uno de los diversos usos a implementar con estas tecnologías emergentes que permiten la operación remota.

Finalmente, se resalta que mi experiencia en el área de estudios es afín al desarrollo de la tesis.

- En el campo de la programación, así como manejo de datos.
- Práctica y certificación en la operación del brazo robótico UR3, que se encuentra en la Universidad Iberoamericana.
- Animación 3D.
- Base de datos relacional como no relacional aplicado a proyectos, tanto educativos como profesionales.
- Diversos proyectos de mecatrónica.
- Sistemas de instrumentación y control.

1.3 Objetivo General

El objetivo general del proyecto se enuncia en la siguiente fase:

Diseñar, desarrollar e implementar un sistema de control a distancia de robots manipuladores industriales a través de un sistema de realidad mixta en el marco de la Industria 4.0

1.4 Objetivo Específico

Para lograr el objetivo general, se proponen los siguientes objetivos específicos:

- OE1. Revisión bibliográfica sobre robótica industrial y realidad virtual en el contexto de la Industria 4.0.
- OE2. Levantamiento de requerimientos y diseño de una interfaz de realidad aumentada en una plataforma de desarrollo en 3D para el control de robots manipuladores industriales.
- OE3. Enlazar la interfaz de realidad mixta con los controladores de los robots manipuladores en el laboratorio de robótica de la Ibero, creando una interacción en tiempo real a distancia.
- OE4. Realizar análisis y optimización de desempeño de la solución a través de la realización de una tarea de automatización específica.

1.5 Originalidad

Para el marco teórico, existen diversas publicaciones a nivel internacional donde podemos observar el estudio del comportamiento de la realidad virtual y aumentada desde el ámbito de los videojuegos [3], educación [4] hasta la industria [5]. Esto aporta un gran avance tecnológico al desarrollo de la realidad mixta y la industria, posibilitando un fuerte artífice inicial a futuras mejoras o investigaciones en esta área de tecnologías emergentes. La mayoría de estas investigaciones no son implementadas debido a los recursos que genera este tipo de investigación. La mayoría se basan en investigación simulada y no en su aplicación. En México encontramos muy poca, o casi nula, investigación o implementación de estas tecnologías. En nuestro país la investigación va más dirigida al área comercial. El punto principal es la venta de productos o marcas, así como recorridos virtuales [6], donde el usuario solo se convierte en un espectador, limitando seriamente su nivel de interacción con la aplicación.

En nuestro conocimiento, este trabajo de investigación será el primer desarrollo nacional de diseño de una interfaz de realidad mixta en una plataforma de desarrollo 3D dirigida a la industria que permita operación remota. Existen diferentes estudios de simulación de interfaces en realidad aumentada [7] sin embargo, no plantean conexiones remotas. Esta implementación se convertirá en una referencia para futuras investigaciones en México, lo cual permitirá que la Universidad Iberoamericana se convierta en una institución líder en el desarrollo de diversas aplicaciones de realidad virtual, aumentada y mixta en el campo de la Industria 4.0 y posteriores; generando un fuerte avance no solo en la industria, también en campos como la medicina y la educación.

1.6 Metodología

La metodología del proyecto se descompone en dos etapas anuales, donde se relacionan las actividades para cumplir los objetivos específicos, como se puede observar en la siguiente *figura 1.1*:

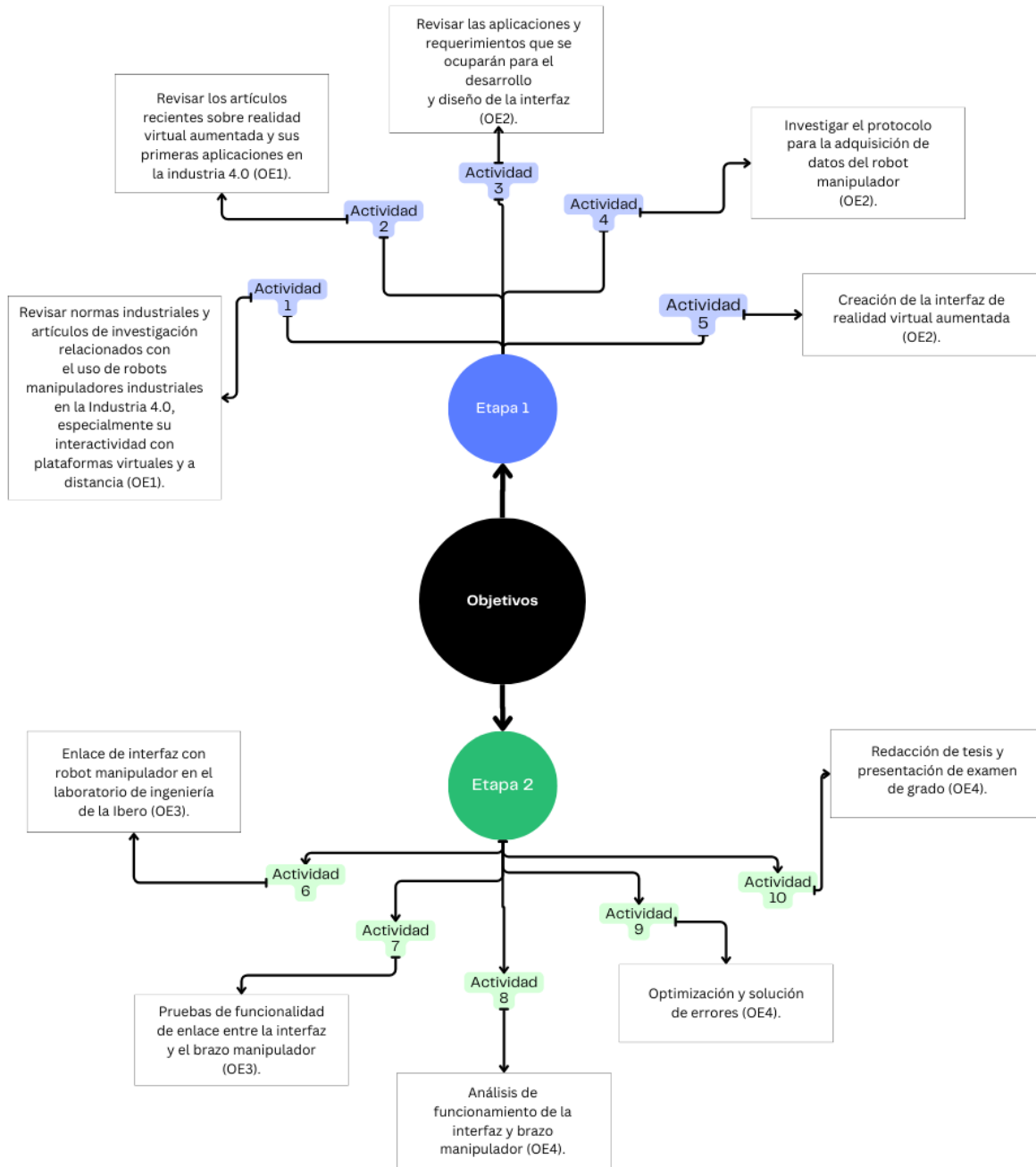


Figura 1.1: "Metodología"

1.7 Cronograma de Actividades

Las actividades mencionadas en la metodología se resumen en el siguiente cronograma.

1er Año	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre
Act.1		x	x	x		
Act.2				x	x	x

1er Año	Enero	Febrero	Marzo	Abril	Mayo	Junio
Act.3	x	x	x			
Act.4			x	x	x	
Act.5			x	x	x	x

2do Año	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre
Act.6		x	x	x	x	
Act.7			x	x	x	x

2do Año	Enero	Febrero	Marzo	Abril	Mayo	Junio
Act.8	x	x	x			
Act.9		x	x	x	x	
Act.10		x	x	x	x	

Desglose de flujo de materiales, actividades y resultados.

Materiales

- HoloLens 2
- Ordenador Windows
- Robot UR3

Actividades

- Conexión HoloLens 2 - Firebase.
- Conexión Brazo robótico UR3 - Firebase.
- Interacción HoloLens 2 - Brazo robótico UR3.
- Pruebas de posicionamiento espacial de los objetos.
- Pruebas de ergonomía en usuarios.
- Pruebas de adaptación a la plataforma.
- Pruebas de funcionalidad en aplicación.

Resultados

- Conexión a distancia entre HoloLens 2 y UR3.
- Observar comportamiento en el gemelo virtual en HoloLens 2 del modelo UR3 del laboratorio de la Ibero.
- Manipulación del modelo digital en los HoloLens 2 por el usuario.
- Programación de articulaciones del modelo digital en los HoloLens 2 reflejando a distancia los movimientos en el brazo UR3 en el laboratorio de la Ibero.
- Operación de usuarios de la plataforma en tiempo real.

1.8 Infraestructura Disponible

La Universidad Iberoamericana cuenta con toda la infraestructura necesaria para desarrollar proyectos de investigación de tecnologías emergentes como son de realidad virtual en el INIAT, hasta inteligencia artificial en el departamento de ciencia de datos, programación de brazos colaborativos y líneas de producción industrial en los laboratorios de electrónica, entre otras. La Universidad cuenta con varios brazos robóticos UR3, de los cuales fue extraída toda la información para la investigación e implementación de la plataforma, también ha sido pionera en el desarrollo de tecnologías de robótica móvil con la implementación de robots omnidireccionales.

- Laboratorio de electrónica.
- Laboratorio de robótica.
- Laboratorio de análisis de movimiento.

Capítulo 2

Estado del Arte

2.1 Introducción

El objetivo de este capítulo es sentar las bases sobre el desarrollo de una plataforma de realidad mixta que permita el óptimo funcionamiento de la industria, esto debido a la implementación de operación a distancia por un operario, brindando una herramienta de soporte para el control de robots colaborativos, permitiendo proteger la integridad de los operadores y reducir costos de operación.

La plataforma tiene la función de resolver los problemas de comunicación y enlace entre el usuario y el robot UR3 mediante un entorno virtual, por medio de la recopilación de información que se enlaza de la nube, creando un entorno de IOT (Internet of things). Con esto, podemos tener un control del brazo manipulador en el entorno de realidad mixta. El enfoque teórico se centra en los procesos de intercambio de datos, así como el comportamiento e investigación de los procesos y aplicación de las nuevas tecnologías en la industria. El enfoque práctico propone una arquitectura que facilite los procesos de operación, entre el usuario y el dispositivo, a controlar de una forma eficaz y segura.

Como primer objetivo es indispensable conocer los antecedentes de estas tecnologías, su evolución, sus aplicaciones actuales y su impacto futuro. La realidad virtual, aumentada y mixta, así como la creciente Industria 4.0 son tecnologías emergentes que se encuentran evolucionando actualmente en industria, educación y capacitación; es necesario explorar los conceptos básicos necesarios para implementar estos dispositivos de forma correcta y útil.

2.2 Antecedentes Históricos

En la actualidad, los sistemas de realidad virtual (RV), realidad aumentada (RA) y realidad mixta (RX) se localizan en una creciente exponencial de aplicaciones y usos en diversos campos o dispositivos, creando una interacción e integración con diferentes tecnologías emergentes [8]. Se desarrollan múltiples experimentos en el campo de lo virtual para así generar un paralelismo con lo real, usando dispositivos móviles como smartphones y tabletas. Los sistemas de realidad virtual y aumentada se podrían denominar como nuevas tecnologías

informáticas para crear un entorno simulado en el ámbito de la automoción. La realidad virtual y aumentada permiten crear entornos en 3D, ya sea utilizando lentes VR y RA, o dispositivos móviles generando experiencias inmersas en un mundo artificial 3D, o añadiendo elementos a entornos reales [9].

La realidad virtual es posible definirla como un sistema informático que nos permite la generación en tiempo real de representaciones virtuales en un ecosistema inmerso. Se podría decir que estas son tomadas realmente como ilusiones, ya que son percibidas a partir de una realidad simulada, sin ningún soporte físico, tratándose de un entorno totalmente digital e inmerso. Estas tecnologías se destacan como una excelente herramienta en la reconstrucción de la herencia cultural, la medicina, la simulación en videojuegos, estructuras de construcción y diseño; la meta principal es hacer que el usuario sienta o perciba que está en otro lugar o añada aditamentos a su entorno generando flexibilidad a las tareas.

La realidad aumentada tiene algunas diferencias con respecto a la realidad virtual. Una de las más importantes es la interacción del usuario en una visión real con adiciones virtuales sin aislar al usuario de su entorno, a diferencia de la realidad virtual. La realidad aumentada, el usuario tiene una interacción sobre una perspectiva de la realidad, añadiendo asistencias visuales en 3D. Algunos estudios plantean desde la vista de un agente virtual que vive en un mundo inmerso observando estas tecnologías [10]. De este modo, el usuario combina imágenes reales y virtuales de forma interactiva en tiempo real, lo que permite añadir información virtual a los elementos que disponemos en el mundo real.

La realidad aumentada cada día ocupa más espacio en diversos campos, como la capacitación en la industria o la educación con sistemas didácticos de aprendizaje [11]. Esta tecnología genera un excelente apoyo a la industria, creando entornos seguros para los operadores, obteniendo aplicaciones de interacción entre humano - máquina con diferentes dispositivos implementados para la mejora de los procesos, minimizando costos.

2.2.1 Realidad Virtual

Es importante entender la evolución de la realidad virtual desde sus inicios a principios del siglo XX. Cerca de 1935, el escritor de ficción Stanley G. Weinbaum [12], en su cuento titulado "Las gafas del Pigmalion" como se observa en la siguiente *figura 2.1* [13], nos da una referencia específica de unas gafas de Realidad Virtual. En su cuento, menciona el invento de un profesor que permite a quien use las gafas transportarse a otros mundos y lugares a través de ellas [14].

Es necesario meter en contexto los inicios de la fotografía, las representaciones más tempranas que intentó generar una sensación envolvente fue la Panoramic Painting, con bastante éxito en Europa en el siglo XIX, de los mayores exponentes fue Franz Roubaud, pintor Ruso que dejó un gran legado de pinturas de recreación histórica bélica, como "La batalla de Borodino"; más antigua aún es la obra en China equiparable a la de Leonardo Da Vinci de Zhang Zeduan [15] quien realizó diversas pinturas sobre la vida cotidiana y costumbres de la

época, la más conocida es “A lo largo del río durante el Festival Qingming”, se logró apreciar en forma de cuadro animado en una pantalla de 6 metros y medio de alto por 128 metros de largo, durante la Expo Mundial de Shanghai en 2010, como se observa en la *figura 2.2* [15].

El siglo XIX es el siglo de la fotografía y de su vertiente más innovadora, la fotografía estereoscópica. El primer estereoscopio fue inventado en 1838 por el británico Charles Wheatstone [16] y consistía en una especie de gafas que simulaban una imagen 3D. Posibilitaba reproducir el efecto de la visión binocular observando separadamente con nuestros ojos dos imágenes tomadas desde puntos de vista a distancia pupilar o mayor, en la *figura 2.3* [16] podemos observar como consiste en posicionar dos imágenes con la misma separación, se montan en un visor de tal manera que cada ojo observe lo que corresponde, esto producirá un efecto de relieve como sucede con los cuerpos tridimensionales, la *figura 2.4* [17] muestra como era el diseño de las gafas Estereoscópicas. La primera cámara fotográfica estereoscópica llegaría unos 10 años más tarde, hacia 1850. Permitía capturar dos imágenes, que observadas a través de un visor, generaban una imagen 3D en el interior del cerebro, o más bien la sensación de profundidad. Esta técnica fue muy popular durante más o menos un siglo. En la exposición de teatro, música y amistad [en torno a Ángel Barrios] en 2007, se utilizaron visores de fotografías estereoscópicas, a modo de elementos museográficos, que permitían ver las fotografías históricas en 3D. La historia de las gafas 3D y el estereoscópico se encuentran muy relacionadas, ya que se utilizó la mayoría de los avances, aunque no en todos los casos, ya que la estereoscópica se extiende en un espectro más amplio de las gafas 3D. Las primeras gafas de Realidad Virtual fueron creadas en 1960 por Morton Heiling, el cual es conocido como el padre de la Realidad Virtual, creó una tecnología inmersiva basada en técnicas cinematográficas, creando Sensorama *figura 2.5* [15], era una máquina bastante aparatosa, que permitía a los usuarios disfrutar de una experiencia multisensorial, se diseñó para ser un equipo portátil en la cabeza, se le denominó como aparato de televisión estereoscópico para uso individual. En la actualidad podría parecer un casco rudimentario, pero el concepto parecía muy similar a los actuales, esto dio a considerarse como las primeras gafas de Realidad Virtual de la historia [18]. El término Realidad Virtual se acuñó por Jaron Lanier y Tom Zimmerman en 1987, siendo de los principales artífices en desarrollar “Guantes de datos”, que supuso un gran paso en el campo de la Realidad Virtual Háptica. Posterior a esto, fueron creciendo exponencialmente los avances en investigación de nuevos lentes de Realidad Virtual. El funcionamiento de las gafas de Realidad Virtual resulta interesante, ya que concentra diferentes tecnologías integradas en un mismo dispositivo, el sistema de visualización se conforma por dos pantallas en las que un usuario puede observar una secuencia de vídeos en cada uno de los ojos. Adaptando el contenido y mediante la visión estereoscópica, el usuario lo percibe como un espacio tridimensional, dinámico e interactivo. Además, incluyen sensores inerciales y de posicionamiento para poder reaccionar a los movimientos que realiza un usuario, esto permite tener una experiencia totalmente interactiva, permitiendo a esto cuando existe un movimiento de la cabeza o manos en el mundo real se replica en el mundo virtual. Normalmente, el software que se utiliza para crear las experiencias de Realidad Virtual es básicamente un motor de creación de videojuegos que es capaz de renderizar imágenes con una alta resolución e hiperrealista, hoy en día está más dirigido a la industria de entretenimiento en videojuegos.

Renderizado

PYGMALION'S SPECTACLES

By **STANLEY G. WEINBAUM**

Author of "The Black Flame," "A Martian Odyssey," etc.

© 1935 by Continental Publications, Inc.



Figura 2.1: "Gafas de Pigmalión"



Figura 2.2: "A lo largo del río durante el festival Qingming"

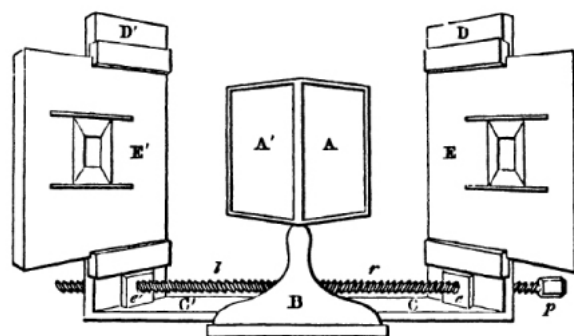


Figura 2.3: "Fotografía Estereoscópica"

Es el proceso que transforma la información de una escena tridimensional, geometría, texturas, iluminación en fotogramas 2D que puede visualizar el usuario. En el caso de la Realidad Virtual se calculan dos fotogramas a la vez, esto para cada uno de los ojos.



Figura 2.4: "Estereoscopio"



Figura 2.5: "Sensorama"

2.2.2 Realidad Aumentada

Es interesante analizar de qué forma y en que época tiene su surgimiento esta tecnología, la primera ocasión que se utilizó el concepto data de 1901 por el escritor Frank Baum, quien imaginó por primera vez unas gafas electrónicas que permitían visualizar información adicional sobre personas que se tenía enfrente [19]. El artista informático Myron W. Krueger, por el año de 1973 inventa la primera instalación de Realidad Aumentada la cual mezclaba diferentes cámaras de vídeo que contaba con un sistema de proyección como observamos en la *figura 2.6* [20], creo un entorno muy interactivo que funcionaba con los movimientos de los usuarios usando sombras y movimiento.

El término de Realidad Aumentada lo crean Jaron Lainer y Thomas Caudell en 1989 [21], el primer sistema que funcionó fue de Louis Rosenberg, desarrollado en el laboratorio Armstrong de las Fuerzas Armadas Estadounidenses (USAF) en 1992, cuyo nombre era Virtual Fixtures y se trataba de un sistema robótico increíblemente complejo diseñado para compensar la falta de procesadores gráficos en 3D, este permitía superponer información sensorial

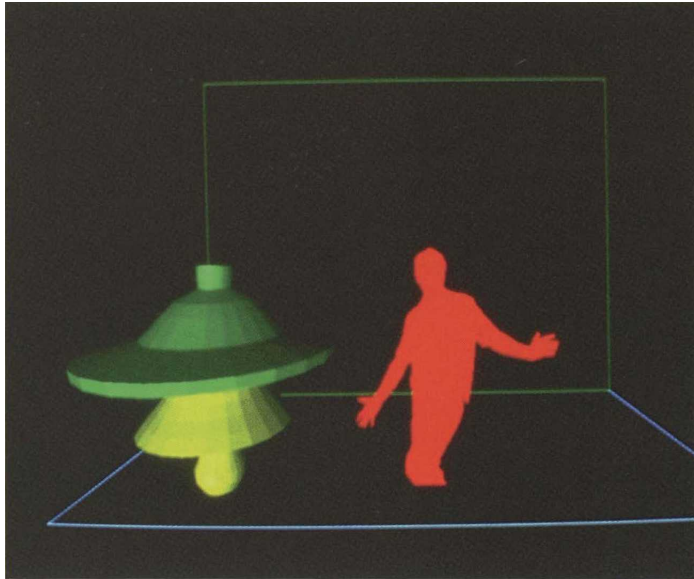


Figura 2.6: "I Met A Morph" instalación educativa de Myron Krueger

en un área de trabajo para mejorar la productividad humana.

En los 90's inició una amplia variedad de aplicaciones que utilizan este concepto para resolver problemas en ámbitos como el diseño y la industria.

- Juego de móvil ARQuake, desarrollado por Bruce Thomas en 2000
- ARToolkit, desarrollo Adobe Flash en 2009
- Google anuncia la beta de Google glass en 2013
- Microsoft crea HoloLens en 2015, con soporte para Realidad Aumentada.

Debido a los avances tecnológicos, en la actualidad es posible ocupar la realidad Aumentada en diferentes dispositivos como computadoras, smartphones y en gafas de realidad mixta. Así como en una amplia gama de sectores:

- Arqueología, Arte, Arquitectura
- Comercio, Administración
- Construcción, Diseño Industrial
- Educación, traducción
- Gestión de emergencias, recuperación de desastres, Medicina
- Juegos, deportes, entretenimiento y turismo

- El ejército
- Navegación

A pesar de que el concepto se inventó hace ya algunos años, la Realidad Aumentada hoy en día es una tecnología emergente, debido a la necesidad de grandes avances tecnológicos en visualización y potencia computacional que lo catapulte a alcanzar su punto más alto. Pasó de ser un sueño a una realidad en solo un siglo, gracias a la investigación y avances que empujan a las nuevas tecnologías a cada día tener un papel más importante en la sociedad, creando aplicaciones que integren la realidad aumentada a la vida diaria, mejorando la productividad, eficiencia y la calidad en las experiencias del usuario final, se observa un potencial ilimitado. Lo que es necesario cuestionar es su posible implementación e impacto que generara en las generaciones presentes y futuras.

2.2.3 Realidad Mixta

La realidad mixta se considera como la nueva ola en la informática, es cada vez más popular entre las empresas y consumidores, debido a la liberación de experiencias limitadas a solo pantallas, al ofrecer interacción instintiva con los datos de nuestros espacios cotidianos [22]. La realidad aumentada móvil ofrece las soluciones de realidad mixta más populares en la actualidad en las redes sociales. La realidad mixta es la mezcla de universos físicos y digitales, estos permiten una interacción en 3D lo más natural, posible e intuitivo entre personas, equipos y el entorno. Se basa en la visión artificial, el procesamiento gráfico, tecnologías de visualización, informática, la nube, entre otras.

El término de realidad mixta fue introducido en 1994 por Paul Milgram y Fumino Kishino, en el documento es “A Taxonomy of Mixel Reality Visual Display”, en el cual se aborda el concepto de continuidad de la virtualidad y la taxonomía de las pantallas visuales.

Desde ese entonces, la aplicación de realidad mixta ha ido más allá de las pantallas a fin de incluir

- - Comprensión del entorno (mapeo y anclaje espacial)
- - Comprensión humana (seguimiento de manos, ocular y entrada de voz)
- - Sonido espacial
- - Ubicación y posicionamiento en los espacios físico y virtual
- - Recursos 3D en espacio de realidad mixta

En las últimas décadas, la relación entre personas y equipos evoluciona de manera conjunta por medio de métodos de entrada, surgió una nueva disciplina que se conoce como la in-

teracción entre el ser humano y equipos, la entrada humana ahora tiene la posibilidad de introducir componentes externos como teclados, ratón, táctiles, voz y seguimientos.

El avance de los sensores y potencia de procesamiento han apoyado a crear mejores percepciones informáticas de los entornos en función de los métodos de entrada avanzada, los cual nos permiten conocer:

- Posición corporal de un sujeto en el mundo físico
- Objetos, superficies y límites
- Iluminación ambiental
- Sonido ambiente
- Reconocimiento de objetos
- Ubicación espacial

A medida que el usuario se mueve en su entorno físico, estos movimientos se pueden mapear en una realidad digital, creando un gemelo virtual. Los límites físicos influyen en las experiencias de la realidad mixta, como juegos o instrucciones basadas en tareas en sistemas de producción con las entradas y percepción del ambiente, las experiencias comienzan a combinarse entre la realidad física y digital.

Estas dos realidades demarcan los extremos de un espectro conocido como continuo de la virtualidad, en un extremo del espectro se tiene la realidad física en donde se localiza al usuario y a contra extremo el del espectro del espacio digital.

2.3 Clasificación de Robots

La clasificación de los robots está determinada por diferentes factores, uno de ellos es basado en su generación [23]. Estos se conocen como instrumentos o herramientas diseñadas con la finalidad de apoyar en actividades que para una persona es poco más difícil de realizar, en sus inicios la idea principal era realizar las tareas repetitivas que pudieran ayudar a las personas; sin embargo, el avance a la ciencia y la tecnología ha permitido desarrollar actividades muy parecidas o mejor a como las podría desarrollar una persona.

Clasificación según la generación de los Robots:

- Primera generación - Manipuladores
- Segunda generación - Robot de aprendizaje

- Tercera generación - Con control sensorial
- Cuarta generación - Robots inteligentes
- Quinta generación - Inteligencia artificial

Los robots presentan diferentes características que permiten distinguir sus clasificaciones dependiendo de sus aplicaciones y prestaciones [24], los ejemplos antes vistos muestra su avance cronológico, se los clasifica basado en su segmento al que se enfoca su actividad de ejecución como la industria, la educación, el sector salud entre otras.

Tipos de robots según su movilidad:

- Robots articulados: Capacidades reducidas, excelentes para mover productos y la manipulación de herramientas.
- Vehículo de guiado automático (AGV): Movilidad definida, usualmente necesitan supervisión de un humano.
- Robots móviles autónomos (AMR): Presentan formas humanas y con funciones parecidas a las personas.
- Cobots: Su diseño principal trabajo colaborativo entre ellos y los humanos, apoyando con tareas repetitivas y en algunos casos peligrosas.

Tipos de robots por función o sector:

- Industriales: Fuerte enfoque en cadenas de producción, actividades rutinarias y repetitivas.
- Doméstico: Ayudan con las tareas de limpieza, vigilancia, entre otras.
- Educativos: Destinadas al apoyo cognitivo o aprendizaje por práctica y experimentación inicial.
- Militares: Apoyo para los ejércitos en aspecto de operación, transporte o detección de explosivos.
- Médicos: De gran apoyo en el sector sanitario ayudando a trasladar insumo, intervenciones quirúrgicas o apoyo a personas con movilidad reducida.

Como es posible observar sus funcionalidades son variadas, depende en gran parte su operación, pero cuál es su definición realmente, la Real Academia Española [25] lo describe como “una máquina o ingenio electrónico programable que es capaz de la manipulación de objetos y realizar diversas tareas u operaciones”, capaz de imitar figuras y movimientos de un ser animado.

2.3.1 Robots Fijos o Estacionarios

La categoría de robots fijos, agrupa los robots que permanecen en una ubicación fija para realizar sus tareas, teniendo la limitación de desplazamiento impuesta por sus grados de libertad, los rangos de movilidad de sus articulaciones, normalmente usados en la sub clasificación de robot industrial manipulador [26].

Esta categoría de robots fueron los primeros en ver su implementación en los procesos industriales, en estos últimos tiempos se encuentran fuertemente posicionados con tecnología actualizada y mayor campo de aplicación. Este tipo de robots ejecutan tareas básicamente de manipulación de objetos en entornos industriales, mecanizado, paletizado, soldadura de puntos, embalaje, pintado, corte láser, entre otras.

Últimamente, este tipo de robots, gracias a la variedad de aplicaciones, disminución en tamaños y los avances tecnológicos, permiten ocuparlos en instituciones educativas con la finalidad de ofrecer un preámbulo de su aplicación y programación de estos dispositivos.

Uno de los robots más utilizados para la industria y la educación, son los brazos robóticos que presentan funciones o ergonomía muy parecida a un brazo humano, normalmente programable y mecánico, es importante conocer su estructura mecánica para comprender su funcionamiento. Gran parte de estos robots guarda cierta similitud con la anatomía de las extremidades superiores del cuerpo humano, a lo cual se hace la referencia a los distintos elementos del robot como cintura, hombro, brazo, codo, antebrazo, muñeca, mano. Estas extremidades se encuentran conectadas con juntas o articulaciones que permiten el movimiento de los eslabones que componen el brazo de forma consecutiva o anidada, como podemos observar en la siguiente *figura 2.7* [27], estas articulaciones pueden ser lineales o rotacionales dependiendo de que el eslabón gire o se mueva sobre algún eje, al conjunto de eslabones y articulaciones normalmente son denominados cadena cinemática, estas pueden ser abiertas o cerradas dependiendo si cada eslabón se conecta mediante articulaciones exclusivamente al anterior y al siguiente exceptuando el primero, que está fijo en un soporte y el último queda en libertad o se conecta a un efector final. Para la cadena cinemática cerrada, usualmente se puede llegar desde cualquier eslabón a cualquier otro esto mediante al menos dos caminos.

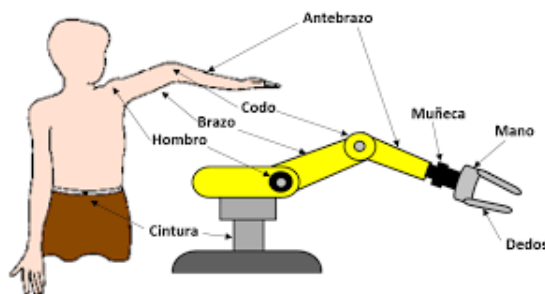


Figura 2.7: Estructura mecánica de brazo robótico

La mayoría de robots manipuladores son cadenas cinemáticas abiertas con articulaciones de tipo rotacional o prismática por lo general, es sencillo interpretar cuantos grados de libertad presenta, ya que coincide con el número de articulaciones de los cuales compone, usualmente

son más usados los robots con cadena cinemática abierta en la industria.

Existen diferentes tipos de robots dependiendo de sus tipos de articulaciones, estas se clasifican según el movimiento permitido entre los eslabones. Existen cinco básicos que son rotacional, prismático, cilíndrico, plano y esférico. En la industria son más utilizadas las articulaciones prismáticas y rotacionales.

Esto resume cada uno de los movimientos que permite una articulación entre dos eslabones de una cadena de un grado de libertad, el número total de grados de libertad se obtiene mediante la suma de cada uno de los grados de libertad de cada articulación. En la siguiente tabla 2.8 [28] podemos observar algunos tipos de robots más utilizados en la industria y sus características.

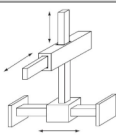
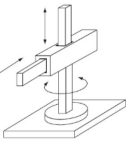
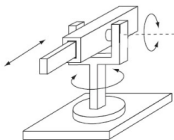
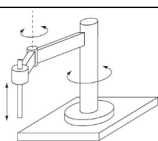
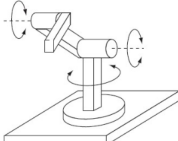
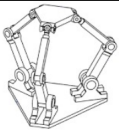
Tipo	ESQUEMA	CARACTERÍSTICA
Robot Cartesiano		El posicionando se hace en el espacio de trabajo con las articulaciones prismáticas. Esta configuración se usa bien cuando un espacio de trabajo es grande y debe cubrirse, o cuando la exactitud consiste en la espera del robot. Posee tres movimientos lineales, es decir, tiene tres grados de libertad, los cuales corresponden a los movimientos localizados en los ejes X, Y y Z.
Robot Cilíndrico		El robot tiene un movimiento de rotación sobre una base, una articulación prismática para la altura, y una prismática para el radio. Este robot ajusta bien a los espacios de trabajo redondos. Puede realizar dos movimientos lineales y uno rotacional, o sea, que presenta tres grados de libertad.
Robot Esférico o Polar		Este tipo de robot cuenta con dos articulaciones rotacionales y una lineal. Dos juntas de rotación y una prismática permiten al robot apuntar en muchas direcciones, y extender la mano a un poco de distancia radial. Los movimientos son: rotacional, angular y lineal. Este robot utiliza la interpolación por articulación para moverse en sus dos primeras articulaciones y la interpolación lineal para la extensión y retracción.
Robot SCARA		Similar al de configuración cilíndrica, pero el radio y la rotación se obtiene por uno o dos eslabones. Este brazo puede realizar movimientos horizontales de mayor alcance debido a sus dos articulaciones rotacionales. También puede hacer un movimiento lineal (mediante su tercera articulación).
Robot Angular o Antropomórfico		Presenta una articulación con movimiento rotacional y dos angulares. Aunque el brazo articulado puede realizar el movimiento llamado interpolación lineal (para lo cual requiere mover simultáneamente dos o tres de sus articulaciones), el movimiento natural es el de interpolación por articulación, tanto rotacional como angular.
Robot de Estructuras paralelas		Posee brazos con articulaciones prismáticas o rotacionales concurrentes. Un robot paralelo es un mecanismo de cadena cinemática cerrada en el cual una plataforma móvil se encuentra unida a una base por varias cadenas cinemáticas independientes. El robot paralelo consiste de una base fija conectada a una plataforma móvil mediante extremidades.

Figura 2.8: Tipos de robots

Para calcular el comportamiento del brazo robótico es esencial hacer un modelado matemático donde se abordan cada uno de sus elementos que permiten conocer grados de libertad, rango de movilidad y distancias de las articulaciones. Conocemos dos tipos de ecuaciones para poder determinar estos parámetros, la cinemática directa y la inversa, la cual será abordada más adelante.

2.3.2 Cinemática Directa e Inversa

La cinemática de un robot se encarga del estudiar el movimiento del mismo con respecto a un sistema fijo de referencia, no considerando fuerzas y momentos que originan estos movimientos.

La cinemática directa busca conocer la posición y orientación del efector final del brazo robótico con respecto a un sistema de coordenadas tomadas como referencia conocida como “q” y la cinemática inversa conocer los valores de cada ángulo de las articulaciones basado en conocer la posición final del efector “x, y, z, α , β , γ ”, como es posible observar en la siguiente *figura 2.9* [29].

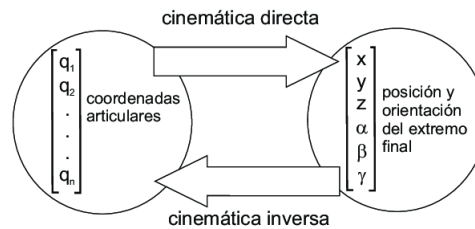


Figura 2.9: Cinemática Directa e Inversa

El modelo cinemático directo es muy funcional para localizar la relación entre la posición y la rotación final con respecto a los ángulos de cada una de las uniones, con respecto a su sistema de coordenadas de referencia.

Existen diferentes métodos geométricos para resolver la cinemática directa, pero para este caso se ocupa el método más usado que es el de Denavid-Hartenberg (D-H) [30]. Este es un método sistemático propuesto en 1955 que sitúa los sistemas de coordenadas que se asocian a cada eslabón y poder obtener la cadena cinemática del brazo robótico a través de matrices de transformación homogénea, solo es válido para robots de cadena abierta.

$$A_i = R_z, \theta_i \text{ Trans}_z, d_i \text{ Trans}_z, a_i R_x, \alpha_i = \begin{vmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$\begin{vmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$\begin{aligned}
 & \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \\
 &= \begin{vmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i \alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{vmatrix}
 \end{aligned}$$

Es importante respetar la posición del desarrollo de las matrices de transformación y rotación, donde [31] en primer lugar genera una rotación sobre “z” seguido de traslaciones y por última rotación en el eje “x”, donde θ_i es una rotación alrededor del eje de $Z_i - 1$, d_i es una traslación a lo largo del eje $Z_i - 1$, a_i es una traslación a lo largo del eje de X_i y α_i es una rotación alrededor del eje X_i .

$\alpha_i, a_i, d_i, \theta_i$, son los parámetros de Denavid-Hartenberg de los enlaces en " i ", y se asocian al enlace " i " y la articulación en " i ", estos parámetros son:

- Torsión de enlace
- Longitud del enlace
- Distancia de articulación
- Ángulo de articulación

Esto en parte a que la matriz A_i es de una sola variable.

Es importante remarcar que las rotaciones tienen que ser sobre el eje x o z de lo contrario los cálculos no serán correctos.

La tabla de parámetros del algoritmo Denavid-Hartenberg, [2.1] depende del tipo de brazo robótico y la posición inicial para conocer sus parámetros ver la siguiente tabla.

Articulación	θ	d	a	α
1	x	x	x	x

Tabla 2.1: Tabla de valores de Denavid-Hartenberg

La cinemática inversa consiste, en determinar los valores que tiene que adoptar cada una de las articulaciones del brazo robótico, que le permita alcanzar la posición y orientación del

extremo del efector conocida. Uno de los métodos más utilizados es el desacoplo cinemático, donde los tres primeros ejes del robot se desacoplan haciendo el cálculo de la cinemática directa de Denavid-Hartenberg para ajustar la posición y los últimos tres para conocer la orientación.

Para la cinemática inversa usualmente es un problema complejo, la cual consiste en crear una serie de ecuaciones cuya solución puede variar en diferentes casos.

2.3.3 Cobot UR3

El brazo robótico UR3 es un robot industrial, compacto, ultraligero y colaborativo de la empresa Universal Robots. Es ideal para implementar directamente dentro de maquinaria, espacios pequeños o salones de enseñanza, gracias a su reducido tamaño 2.10 [32].

UR3 es un robot muy flexible y ligero, perfecto para tareas de ensamblaje y bancos de trabajo automatizados, al tener un peso de 11 kg con una carga útil de 3 kg, rotación de 360° en todos sus ejes y rotación infinita en su efector final.

Para su programación cuenta con una caja de control con:

- 16 entradas digitales
- 16 salidas digitales
- 2 entradas analógicas
- 2 salidas analógicas
- 4 entradas digitales en cuadratura

Los protocolos de comunicación están a cargo de los protocolos:

- Modbus TCP
- PROFINET
- Ethernet/IP
- USB 2.0, USB 3.0

Cuenta con una frecuencia de control de 500 Hz y una pantalla táctil de 12" con interfaz gráfica Polyscope.



Figura 2.10: UR3

Capítulo 3

Conceptos y Tecnologías en el contexto de la Industria 4.0.

3.1 Introducción

El concepto de producción inteligente cada día está revolucionando de mayor forma los procesos industriales, el intercambio de información, la facilidad con la que podemos manipular y almacenar una gran cantidad de información. Esto, aunado a las nuevas tecnologías, como la realidad aumentada que se ha desarrollado en los últimos años, nos permite establecer una interacción entre la realidad y la parte visual gráfica, permitiendo la automatización en la industria donde el operador tiene las ventajas de crear un mundo totalmente virtual desde un escenario real. Potenciando su utilidad tanto para capacitación de personal en ambiente digital, como para la operación y manipulación de maquinaria o herramientas que se ocupan en la industria, manteniendo procesos más prácticos y seguros para los operarios.

Los retos que se presentan para esta tecnología de realidad aumentada en la industria aún son extensos. Existen empresas que la ocupan para capacitar a sus empleados en ambientes seguros y controlados, un ejemplo es UPS que ocupa estas tecnologías para capacitar a los operadores de automóviles de reparto [33].

En este documento se plantea la investigación para desarrollar una plataforma de realidad mixta que permita a un usuario, a través de un mundo virtual, operar dispositivos robóticos como el brazo colaborativo UR3. Para tal efecto es indispensable recopilar toda la información de los dispositivos, dirigiendo la información a la nube para poder procesarla en diferentes entornos, desempeñar las tareas en el ambiente virtual que serán ejecutadas en el ambiente real durante el uso de la plataforma.

3.2 Aplicación

3.2.1 Aplicación en la Industria 4.0

La complejidad de los procesos y los cambios tecnológicos dan la apertura a campos de investigación y aplicación de nuevas tecnologías. Aquí es donde la industria inteligente, también denominada Industria 4.0, se convierte en una pieza fundamental en el avance de las nuevas tecnologías que impactan al sector de la industria. Es considerada como la cuarta revolución industrial y su misión es transformar a las empresas en una organización inteligente para conseguir mejores resultados, apoyándose de las tecnologías digitales emergentes [34].

La primera revolución industrial se caracteriza por la introducción de equipos mecánicos impulsados por la energía de vapor. La segunda se enfoca en la producción en masa usando división de tareas y el uso de la energía eléctrica. La tercera se sustenta en el uso de la electrónica y la informática para obtener una producción automatizada. La cuarta es una nueva era, que da un salto cuantitativo y cualitativo en la organización, así como la gestión de cadenas de valor. Esta etapa de la Industria 4.0 se caracteriza por la automatización, conectividad y globalización de la información. Permite convertir tareas que eran muy complejas en procesos más sencillos, de mayor precisión y rapidez. En los procesos, este avance tecnológico es una herramienta clave para la automatización. Esta tecnología permea no solo en las industrias, también en los hogares, facilitando al usuario la eficiencia y la conectividad con todo el ecosistema. Aquí es donde la IOT (Internet of Things), o internet de las cosas, juega un papel más que importante en este avance [35].

Los recursos como el almacenamiento en la nube, la gran cantidad de datos “Big Data” y el internet de las cosas “IOT” son cruciales para la interconexión de los dispositivos. Los continuos avances en el área de la era digital dan un panorama más amplio de la dirección a seguir se encuentran amplios recursos, como la realidad aumentada en los dispositivos móviles, los ordenadores o refrigerador inteligente en el hogar, lo cual permite crear aplicaciones que faciliten de forma más instantánea el intercambio de información y agilicen la toma de decisiones de forma precisa y segura. Los usuarios no necesariamente necesitarían estar en sus áreas de trabajo para poder resolver eventos que puedan afectar el flujo de un sistema de producción, ya que se cuenta con un amplio flujo de datos convirtiéndose en fábricas inteligentes [36].

El desarrollo, aplicaciones o algoritmos que permitan facilitar y asegurar la eficiencia de los procesos en la industria, es muy necesario, tanto del lado del operador como del proceso en general. La interacción entre las innovaciones con la IOT, la posibilidad de obtener una gran cantidad de información, poder tener una conexión en la nube y gestionar los procesos, abre un panorama cada vez más real de cómo se puede llegar a una mayor eficiencia, con un menor costo y a mayor seguridad [37].

3.2.2 IoT

En los últimos años, el internet de las cosas o “Internet of things” (IoT), se ha convertido en una de las tecnologías emergentes más importantes, debido a que cada día estamos más interconectados. La internet de las cosas, describe la red de objetos físicos que incorporan sensores, hardware y software, con el fin de conectarse entre ellos e interconectar la información de datos a través de internet [38]. En la actualidad existen más de 7 mil millones de dispositivos interconectados, se prevé que este dato aumente a 22 mil millones para el 2025.

Con la tecnología actual es posible la conexión y recopilación de datos de diferentes dispositivos, desde uso cotidiano hasta coches, electrodomésticos, monitores, entre otros millones más de dispositivos integrados, posibilitando la comunicación entre personas de forma fluida y segura. Gracias a las tecnologías como big data, la nube, la ciencia de datos y la informática de bajo costo es posible compartir y recopilar información con la mínima intervención humana.

Estos adelantos tecnológicos se deben a los nuevos avances en las comunicaciones y una serie de tecnologías emergentes, ya que el internet de las cosas existe desde hace tiempo atrás, algunas de ellas son:

- Sensores de bajo costo y potencia
- Conectividad
- Plataformas en la nube
- Aprendizaje analítico y automático
- Inteligencia artificial

Pero como estas tecnologías se aplican a nivel industrial, principalmente, son ocupadas en entornos industriales con control de sensores, instrumentación y dispositivos de conexión con la nube, generando un intercambio masivo de información. Recientemente en la industria se utiliza la comunicación entre dos máquinas o (M2M), generando de esta forma el control inalámbrico entre dispositivos. Debido a esta capacidad de interconexión, el internet de las cosas se le conoce como la cuarta revolución industrial, gracias a su gran impacto y generador de desarrollo de tecnologías.

3.3 Tecnología para la realidad virtual y aumentada

La realidad virtual y aumentada es una tecnología que juega un papel importante en la creciente Industria 4.0. Cada día, más empresas se apoyan en ella para poder tener procesos más eficientes, menores costos y riesgos hacia sus operadores. La realidad aumentada permite acceder a gran información contenida en la nube y hacer más eficientes los procesos entre la máquina y el operario.

Varias empresas se han unido para generar modelos en 3D, con el objetivo de que los programadores puedan crear contenido de aprendizaje que eficiente los procesos de las máquinas, evitando así riesgos de operación. Existen lentes de realidad aumentada, tabletas o dispositivos móviles que pueden ocuparse para tener una integración de contenido gráfico basado en toda la información proporcionada por el fabricante, o la inclusión de Big Data, la cual nos permite extraer toda la información del dispositivo a manipular y subirlo a la nube, creando la interacción entre el contenido gráfico y una vista de la realidad del espacio de ejecución. Con esto, el operador puede manipular gráficamente, desde unos lentes u otro dispositivo, las acciones que realizará el ecosistema, creado en 3D y aplicando las instrucciones en el ambiente real [39].

3.3.1 Hardware y Software

Algunos de los principales fabricantes de dispositivos para realidad aumentada son empresas como Google, Epson, Microsoft, Vuzix y Apple, entre otros. Estas empresas han creado gafas para que el usuario tenga practicidad de movimiento. [40] Hoy en día, los dispositivos móviles también han incorporado aplicaciones que permiten interactuar con la realidad aumentada. Algunos ejemplos son:

3.3.2 Dispositivos para la realidad virtual y aumentada

Google Glass Enterprise Edition 2 La primera versión de las Google glass no logró pasar las primeras pruebas de los desarrolladores. Unas de las grandes mejoras de la segunda edición es un procesador Snapdragon XR1, el primer chipset de Qualcomm verdaderamente orientado a ese segmento. El nuevo chip facilita la tarea a los OEM “Original Equipment Manufacturer” con optimizaciones especiales, capacidades de inteligencia artificial, mayor interactividad, menor consumo de energía y eficiencia térmica mejorada, como muestra la *figura 3.1* [41]. Otras novedades incluyen sensores de movimiento en la cabeza, una nueva cámara de 8 mega pixeles y tecnología de detección de voz con tres micrófonos de formación en haz. Las nuevas gafas inteligentes de Google prometen una autonomía de hasta 8 horas.



Figura 3.1: Google glass 2

Microsoft HoloLens 2 La actualización de las HoloLens es el cambio de procesador de Intel a ARM, el cual incorpora núcleos más poderosos, baja los consumos y aumenta la velocidad de activación del dispositivo. Este dispositivo está diseñado para realidad aumentada y dirigido principalmente a la industria permitiendo un poderoso equipo de desarrollo e implementación, gracias a que cuenta con todo el ecosistema integrado, microprocesador, almacenamiento, batería y conexiones inalámbricas, lo cual lo posiciona entre las mejores gafas de desarrollo de realidad aumentada y mixta, con un diseño muy estético como observamos en la *figura 3.2* [42].



Figura 3.2: Microsoft HoloLens 2

Apple Vision Pro Apple con su nuevo dispositivo, las Vision Pro, ver *figura 3.3* [43]. es una revolucionaria computadora espacial que fusiona el contenido digital con el mundo físico, esto permite a los usuarios interactuar con el contenido digital como si estuviera físicamente presente en el mismo espacio, cuenta con un sistema de ultraalta definición con 23 millones de píxeles distribuidos en dos pantallas.



Figura 3.3: Apple Vision Pro

Meta Quest Meta presenta diferentes opciones a la realidad virtual y aumentada, con sus dos modelos más actuales como son los Quest Pro y Quest 3 apuesta hacia una ruta de la realidad mixta buscando ser una de las marcas líderes en esta tecnología, ofreciendo grandes beneficios en bajo costo y variedad de aplicaciones. Los modelos ver *figura 3.4* [44]. presentan unos lentes cada vez más livianos y con la actualización en las cámaras que ahora son a color mejorando la sensación de los entornos de realidad mixta, es

posible interactuar de manera natural con el mundo virtual sin desconectarte de tu espacio físico también permitiendo la colaboración a distancia, estas gafas mantienen un diseño fuertemente dirigido a la colaboración remota y procesos creativos en equipo.



Figura 3.4: Meta Quest Pro y Quest 3

Epson Moverio BT-300 Las gafas Moveiro BT-300 se presentan como las más ligeras del mundo y son tecnología OLED. Las gafas de Epson destacan por ser las primeras en el mercado que permiten acceder a una vista en primera persona en las tareas de pilotaje de drones. Estas presentan una resolución de 720p con un procesador Intel Atom, cámara de 5 mp 16 gb de almacenamiento interno y 2 gb de memoria RAM como observamos en la *figura 3.5* [45]. Cuenta con control táctil interno, así como con un mando externo que es muy útil para desarrollar diversas tareas.



Figura 3.5: Epson Moverio BT-300

Vuzix M3000 XL Vuzix es una de las empresas más consolidadas en el terreno de las gafas industriales de realidad aumentada. Estas fueron pensadas especialmente para la industria y apuesta por un procesador Intel Atom, 2 gb de Ram, 16 de almacenamiento interno, así como conectividad wifi y bluetooth. Cuenta con un panel de visor a todo color, cámara HD de 16 mp y estabilizador de imagen, doble micrófono con cancelación de ruido y ofrece un puerto micro USB para transferencia de datos, ver *figura 3.6* [46].

En el software para la creación de realidad aumentada existe una gama muy amplia. Cada día son más las posibilidades de encontrar una aplicación acorde a las necesidades de cada usuario, la mayoría son open source, lo cual permite usar modelos existentes, así como



Figura 3.6: Vuzix M3000 XL

crear los propios basados en las necesidades de cada proyecto. Algunos de los programas más destacados son:

- Metaverse
- ActionBound
- Roar
- Zapworks
- Augmented Class
- Aumentaty Author
- ARCrowd
- LayAR
- HP Reveal
- Zappar
- Blippar

Unas de las aplicaciones más poderosas para la creación de contenidos para realidad aumentada son Unity y Vuforia las cuales se describen más adelante. A diferencia de las aplicaciones enumeradas anteriormente, estas requieren de conocimientos de programación más avanzados, lo cual las convierte en herramientas más utilizadas a nivel industrial y el entretenimiento digital como videojuegos y la creación de contenidos de realidad virtual, aumentada y mixta, aplicable tanto en gafas como en dispositivos móviles.

Capítulo 4

Diseño y Protocolos

4.1 Introducción

En este apartado se busca desglosar de manera puntual cada uno de los procesos y protocolos utilizados para la implementación de la plataforma.

4.2 Tecnología para Cobots

4.2.1 Modbus

Modbus fue creado por Modicon, ahora Schneider Electric, a finales de los años 70 para la comunicación entre controladores lógicos programables PLC, Actualmente continúa siendo el protocolo más usado para conectar dispositivos en la industria. En la actualidad, Modbus es el protocolo de comunicación más utilizado en la industria de la automatización y el medio más común para conectar dispositivos electrónicos automatizados [47]. Al ser un protocolo de comunicación abierta facilita su implementación en diversos entornos, esto permite transmitir información a través de redes en serie entre dispositivos electrónicos, se observa en la *figura 4.1* [48] la red de funcionamiento del protocolo.

Cuando se genera un enlace se ubica al dispositivo que emite una solicitud o petición, maestro Modbus y quien suministra la información a este se le conoce como esclavo Modbus, esto dictamina que un dispositivo esclavo no puede ofrecer información, este debe esperar que el dispositivo maestro se la solicite, el maestro escribirá datos en los registros de un dispositivo esclavo y leerá los datos de los registros de un dispositivo esclavo [49].

En una red Modbus estándar, se localiza un maestro y la posibilidad de 247 esclavos, en los cuales cada uno con una dirección de esclavo única de 1 a 247, esto indica que también el maestro puede escribir información en los esclavos.

La especificación del protocolo se publica abiertamente y está libre de derechos, lo que permite una libre instalación para los fabricantes y sus equipos, sin la necesidad de pagar derechos de autor, esto lo solidificó como un protocolo bastante común, en la industria.

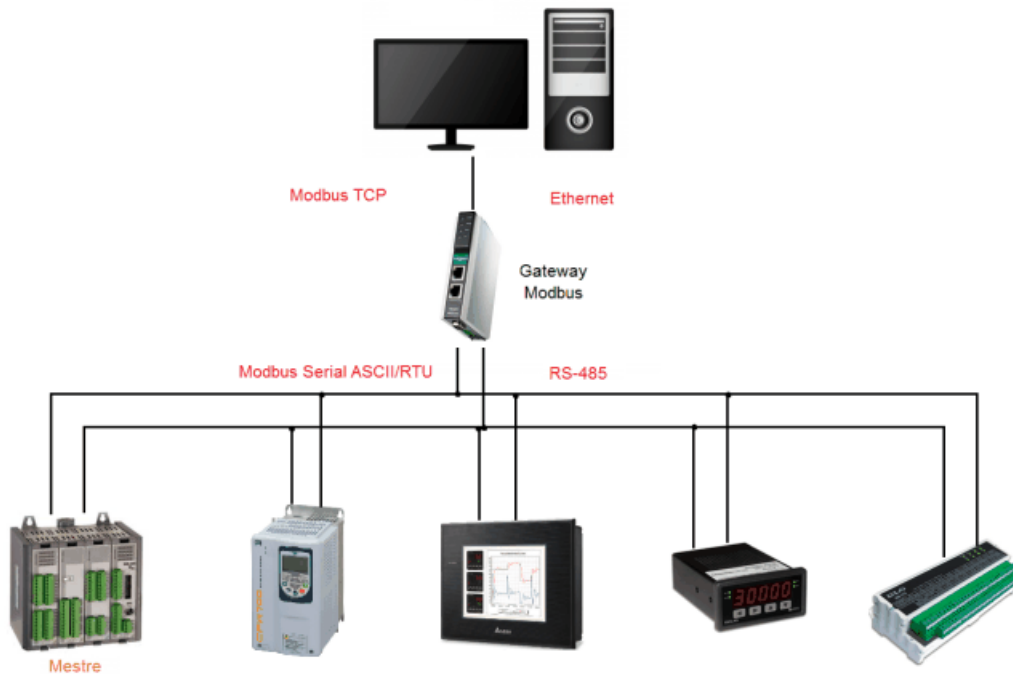


Figura 4.1: Red Modbus

Modbus se usa generalmente para transmitir señales de los dispositivos de instrumentación y control a un controlador principal o un sistema de recolección (SCADA). Destaca su uso en múltiples aplicaciones maestro-esclavo para monitoreo y programación de dispositivos, comunicación entre dispositivos inteligentes, sensores e instrumentación industrial, es un protocolo ideal para aplicaciones donde se requiere una comunicación inalámbrica.

Existen varios tipos de versiones en el protocolo modbus para el puerto serie y Ethernet, que es utilizada para cubrir las necesidades específicas de los sistemas de automatización industrial en la industria, ejemplo Modbus TCP es utilizado para Ethernet, Modbus RTU y Modbus ASCII para los puertos serial. En consecuencia, Modbus TCP hace que la definición de maestro y esclavo sea menos obvia, porque la Ethernet permite la comunicación entre pares, la definición de cliente y servidor son los elementos más conocidos en las redes basadas en Ethernet.

En este contexto el esclavo se convierte en el servidor y el maestro en el cliente, en este sistema se localiza más de un cliente que obtenga los datos del servidor, en terminología de Modbus este significa que habilita múltiples maestros, así como múltiples esclavos.

En lugar de definir maestro y esclavo en un dispositivo físico por dispositivo, el diseñador en este caso es el encargado de crear las asociaciones lógicas entre la funcionalidad de maestro y esclavo, lo cual permite posibilidades de configuración en el entramado del diseño y los comandos de ejecución.

Modbus TCP es utilizado con más frecuencia en:

- PLC (controladores industriales)
- Sistemas SCADA (visualizaciones y control básico de procesos industriales)
- Sensores y actuadores.

En un inicio de la investigación, el protocolo Modbus, sería el encargado para la conexión con el UR3; sin embargo, existe una librería que será explicada más adelante URX, esta será la que proporcionará la conexión.

4.2.2 URX

Antes de explicar el funcionamiento de la biblioteca URX es importante conocer qué es una biblioteca y cuál es su utilidad.

Las librerías o bibliotecas en programación [50] son conjuntos de archivos de código que su objetivo principal es facilitar o agilizar la programación, son esenciales para evitar duplicidad de código, ya que varias de estas líneas de código ya han sido resueltas por otros programadores y así dar mayor agilidad reduciendo el tiempo de desarrollo. En esta investigación se ocuparán bibliotecas diversas, de cliente para monitoreo y control externo e interno de los cobots. Con esta biblioteca es posible crear código que facilite la ejecución e implementación, así como los cálculos matemáticos en diferentes plataformas siempre y cuando cuente con una conexión Ethernet el cobot [51]. En varios casos no es forzoso el uso de estas bibliotecas cliente, esto permite facilitar en algunas ocasiones los procesos. Esto se realiza conectando el robot por I-O (input-output), buses de campo normales directo a la consola de programación del robot, evitando la complejidad de la integración.

Existen diferentes bibliotecas diseñadas por diversas entidades entre empresas o desarrolladores libres, estas son algunas de ellas:

- Universal Robots Client Library
- Universal Robots ROS2 Library
- RTDE Client Library
- UR RTDE
- Sintef Client Library.

Para esta investigación se implementará Sintef Client Library, esta biblioteca fue implementada por la empresa Noruega Siftef, su desarrollo es basado en Python que se encuentra alojada en Git Hub y es compatible con RTDE que es el protocolo (Realtime Data Exchange)

permite el intercambio de información del robot en tiempo real, esta biblioteca se encuentra en Github (<https://github.com/SintefManufacturing/python-urx>).

Algunos de los protocolos necesarios y más importantes para este tipo de comunicación son:

- RTDE(Realtime data Exchange) en el puerto TCP30004 (monitoreo y control de registros)
- Servidor de panel en el puerto TCP 29999 (carga, inicia y detiene el programa)
- Modo intérprete en el puerto TCP30020 (control de movimiento)
- Interfaces primaria, secundaria y en tiempo real (dentro del software de UR se utiliza para la comunicación entre la interfaz del usuario y el control del software)
- Sintef Client Library.

4.2.3 Protocolo TCP/IP

Los protocolos son conjuntos de normas que permiten el intercambio de información entre máquinas y programas de aplicación, es necesario seguir las normas que dicta el protocolo para permitir la interpretación de los mensajes, TCP/IP podemos interpretarlo en término de capas o niveles de la siguiente manera como se observa en la *figura 4.2* [52]. IP(Protocolo de Internet) y TCP(Protocolo de control de transmisión) tienen su origen a principio de 1980 y fueron adoptados por la red ARPANET en 1983, integrada por cientos de computadoras de universidades, centros de investigación militares y algunas empresas. A mediados de los 80's fue creado el protocolo TCP/IP con la finalidad de contar con un lenguaje común a todas las computadoras conectadas a Internet[53].

La dirección IP es la encargada de la identificación del equipo para su localización, es un número único para cada equipo u "host" se representa por 4 cifras separadas por puntos y se determina con un límite de 255, (255.255.255.255) con esta dirección de IP queda identificado el equipo del usuario y la red a la que esta pertenece.

- Capa de aplicación
- Capa de transporte
- Capa de red
- Capa de interfaz de red
- Hardware.

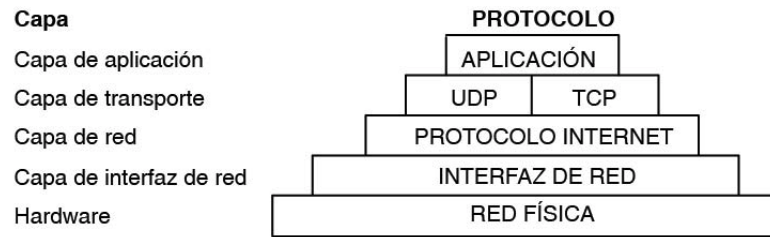


Figura 4.2: "Capas Protocolo TCP-IP"

Son las capas necesarias definidas por TCP/IP para mover la información del remitente al destinatario.

Los programas de aplicación envían datos a uno de los protocolos de las capas de transporte de internet, UDP (User Datagram Protocol) o TCP (Transmission Control Protocol), este es el encargado de recibir los datos de la aplicación, los divide en paquetes más pequeños y añade una dirección de destino, de ahí pasa a la siguiente capa de protocolo.

La capa de red de Internet coloca el paquete en un datagrama de IP, colocando la cabecera y la cola, posterior a ello decide donde enviará el datagrama pasándolo a la capa de interfaz de red.

4.3 Tecnología para la Realidad Mixta

4.3.1 Unity

Unity es un motor de desarrollo o motor de videojuegos multiplataforma creado por Unity Technologies ver [54], *figura 4.3* hace referencia a un software el cual tiene una serie de rutinas de programación que permite el diseño, la creación y el funcionamiento de un entorno interactivo. Dentro de las funcionalidades que tiene un motor de videojuegos estan:

- Motor gráfico para renderizar gráficos 2D y 3D.
- Motor físico que simule las leyes de la física.
- Animaciones.
- Sonidos.
- Inteligencia artificial.
- Programación o scripting.

Unity es una herramienta que permite crear principalmente videojuegos para diversas plataformas (PC, Consolas de videojuegos, móviles, etc.) mediante un editor visual y programación vía script, pero gracias al poder y versatilidad de la plataforma cada día aumenta



Figura 4.3: 3D Unity

los sectores a los cuales es dirigido. Se ha convertido en uno de los principales creadores de contenido para realidad virtual, aumentada y mixta, tanto en dispositivos móviles como la mayoría de gafas que se encuentran en el mercado de desarrollo. Uno de los puntos fuertes que ostenta Unity es la gran comunidad de usuarios que tiene. Esto permite tener acceso a multitud de documentación, foros, repositorios, así como comunidades de resolución de dudas y avances, explicando técnicas y nuevos métodos.

La plataforma Unity aporta la interacción con el modelo cinemático en 3D creado con Blender, en esta parte del proyecto se crea un script en lenguaje C Sharp para interactuar con el modelo diseñado del brazo UR3, (*figura 4.4*).

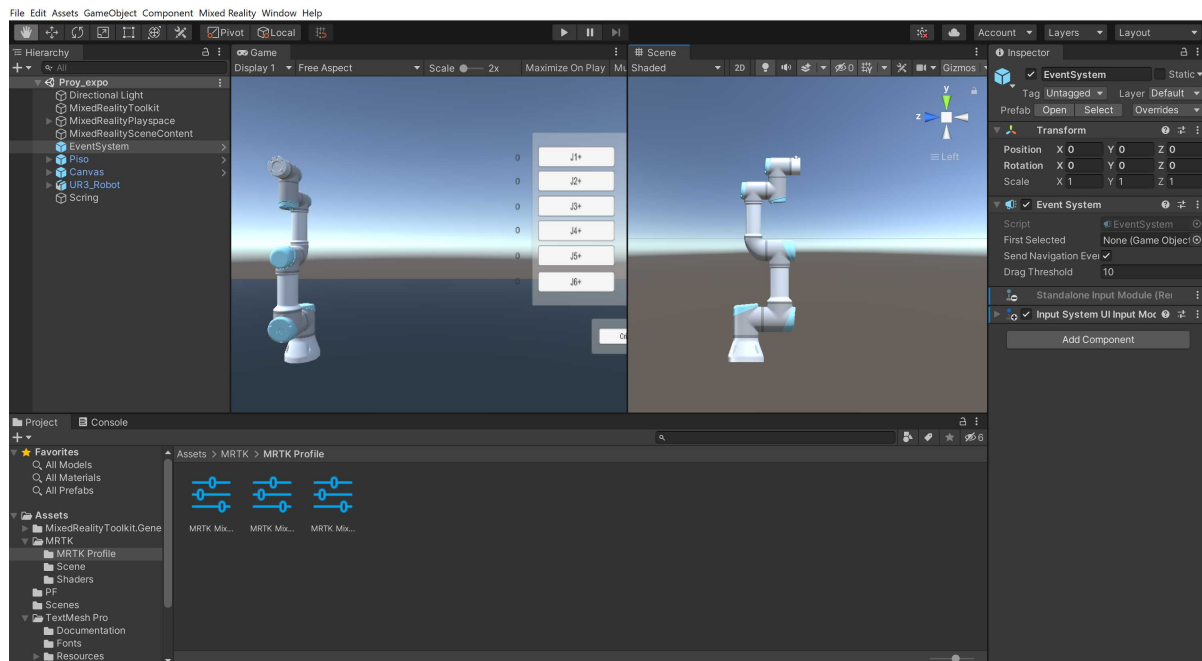


Figura 4.4: UR3 Modelo 3D Unity

4.3.2 Blender

Blender es una suite dedicada a la creación de modelos 3D, ofrece una gama amplia de herramientas esenciales [55], que incluyen Modelado, Renderizado, Animación, Rigging, Edición de Vídeo, VFX, Composición, Texturizado y muchos tipos de Simulaciones. (*figura 4.5*)



Figura 4.5: Blender

Su adaptabilidad a individuos independientes y estudios dedicados al diseño y modelado, su proceso de desarrollo es muy práctico en el diseño, al ser una aplicación que funciona en diferentes plataformas como Linux, MacOS y Windows, aunado a ser un software de código abierto proporciona todas las herramientas para la creación de modelos o animaciones para diversas plataformas. Su interfaz emplea OpenGL esto proporciona una experiencia muy consistente en todo el espectro de plataformas de hardware y software compatibles, Blender cuenta con una vasta comunidad en la cual se pueden encontrar tips, soluciones y ejemplos para mejorar y agilizar los procesos de modelado, ver *figura 4.6*.

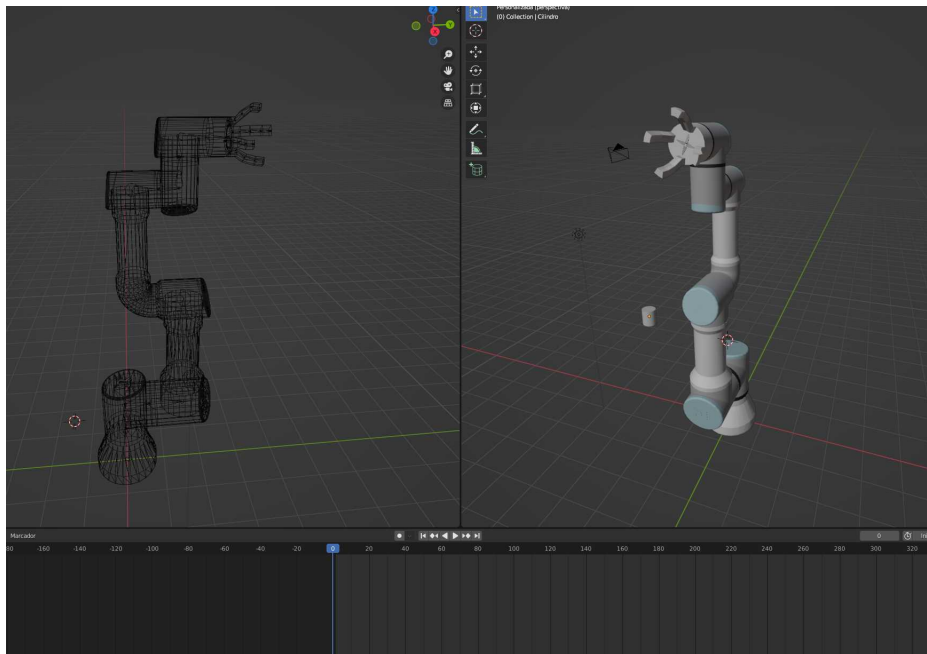


Figura 4.6: UR3 Modelo 3D

Blender permite modelos 3D, así como imágenes fijas y efectos visuales.

En la paquetería de Blender se creó el modelo 3D del UR3 que posteriormente se exporta a Unity en extensión .glb, en Blender se configura la separación de cada una de las articulaciones por capas para permitir programar cada una de las acciones que se ejecutan en el brazo al ser manipulado en las gafas.

4.3.3 Visual Studio

Visual Studio es un IDE de programación que permite crear código para diferentes lenguajes de programación, permite editar, depurar y compilar código, esta plataforma está más diseñada para la plataforma de Windows.

Para el proyecto Visual Studio fue el encargado de compilar e implementar la ejecución de la programación de Unity en las gafas HoloLens 2, así como la creación del código en Python que conecto el brazo UR3 con la base de datos Firebase.

4.3.4 Visual Studio Code

Es un editor de código fuente que se ejecuta en diferentes plataformas, una ventaja es que cuenta con una amplia variedad de extensiones para diferentes lenguajes de programación.

En Visual Studio Code se creó toda la programación para Unity y los diferentes módulos de interconexión en la plataforma y la conexión con la base de datos de los lentes HoloLens 2.

4.3.5 HoloLens 2

HoloLens 2 es un dispositivo de realidad mixta fabricado por Microsoft, ver *figura 4.7* [56], es un dispositivo holográfico, autónomo, ergonómico y sin ataduras, con aplicaciones preparadas para la empresa con el fin de aumentar la precisión y la producción de los usuarios [57].



Figura 4.7: Microsoft

Las lentes HoloLens 2 fueron principalmente creadas para solucionar problemas en la industria, así como agilizar y economizar procesos, ya que permiten generar escenarios de soporte remoto, creando una comunicación directa entre un experto y los miembros de personal de campo.

La posibilidad de conectar varios usuarios al mismo tiempo agiliza de manera eficiente los procesos de comunicación, cuenta con herramientas como Microsoft Teams la cual permite realizar videoconferencias colocando de manera práctica al operador en un contacto directo con el experto remotamente, posibilitando indicar mediante anotaciones gráficas como es posible solucionar la avería.

1. La característica de las HoloLens 2 son [58]:

- Seguimiento de la mano
- Comandos por voz
- Seguimiento ocular
- Mapeo espacial
- Gran campo de visión

Las gafas de HoloLens 2 llevarán todo el control de flujo de interacción entre el usuario y el UR3, se alojará el modelo digital con su programación en las gafas conectadas a internet para tener comunicación con el brazo UR3 que se encuentra a distancia y conectado a una red de internet.

Capítulo 5

Desarrollo

5.1 Introducción

El proyecto se desarrolla a partir de la aplicación de realidad mixta para la Industria 4.0, dado que es una tendencia mundial y en la cual México no ha incursionado con la debida fuerza. El proyecto tiene un enfoque de diseño de arquitectura de realidad mixta integrada a controladores y sistemas de comunicación industriales con un trabajo conceptual importante, pero también de desarrollo de hardware y software.

Las siguientes secciones dan un panorama general del comportamiento de la Industria 4.0 y sus oportunidades de combinarse con la realidad virtual, aumentada y mixta. Aborda las principales tecnologías que están disponibles para la realización de este proyecto en el marco del fuerte impacto del internet de las cosas, la Industria 4.0 y el avance en los dispositivos de inmersión. El proyecto se divide en dos secciones que conectan en un punto para lograr la integración completa. La primera parte se enfoca en el desarrollo e implementación de la plataforma de realidad aumentada en los visores HoloLens 2, así como su conexión a la base de datos. La segunda parte se desarrolla la conexión, interacción y recolección de datos entre el brazo robótico UR3 y la base de datos en la nube en tiempo real. Para crear una plataforma interconectada como podemos observar en la *figura 5.1*

El objetivo medular del proyecto se centra en la creación de una plataforma de realidad mixta, operada a distancia desde unos lentes de realidad aumentada, intercambiando información en la nube con un brazo robótico UR3 que se encuentra en los laboratorios de la universidad. Con eso se busca la eficiencia, capacitación y cuidado tanto de operadores en la industria como alumnos en instituciones educativas. Se diseñó una operación sencilla e intuitiva gracias a la evolución de las tecnologías que permiten una interacción fluida y visualmente práctica para los usuarios, permitiéndoles una interacción muy amigable y práctica.

5.1.1 Modelado Digital UR3

Blender es la plataforma de diseño de creación del modelo debido a sus cualidades, Blender fue la plataforma de realización del modelo en 3D del brazo UR3, creando cada una de sus articulaciones en diferentes capas, esto permite modificar de manera independiente cada una

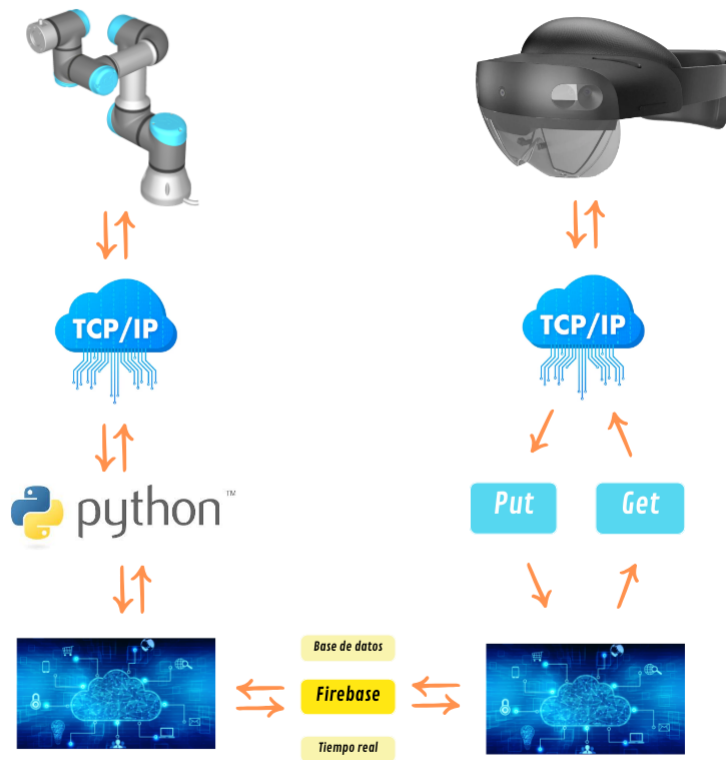


Figura 5.1: Flujo de la plataforma

de ellas como observamos en la *figura 5.2*, colocando texturas y cualidades físicas necesarias para crear un modelo lo más preciso al real.

En la creación del modelo, es importante poner gran atención a detalles que resultan cruciales debido a que de lo contrario puede generar gran complejidad en la implementación de la programación y cálculos matemáticos, los ejes de coordenadas son similares con el estudio real de la cinemática del robot UR3, con respecto al modelo recordar que basado en los parámetros de cinemática directa para el cálculo de Denavit-Hartenberg, tomando en cuenta los ejes cartesianos z y x , a lo cual se busca que el eje de rotación en el modelo 3D trabaje con respecto al eje cartesiano z como lo observamos en la *figura 5.3*, esto evita errores de cálculo al desarrollar la tabla de valores de Denavit-Hartenberg cuando se implemente la cinemática directa e inversa del brazo UR3.

Basado en el análisis del brazo robótico UR3 y su posición con cada una de sus 6 articulaciones, es posible obtener los valores de la tabla Denavit-Hartenberg. Como es posible

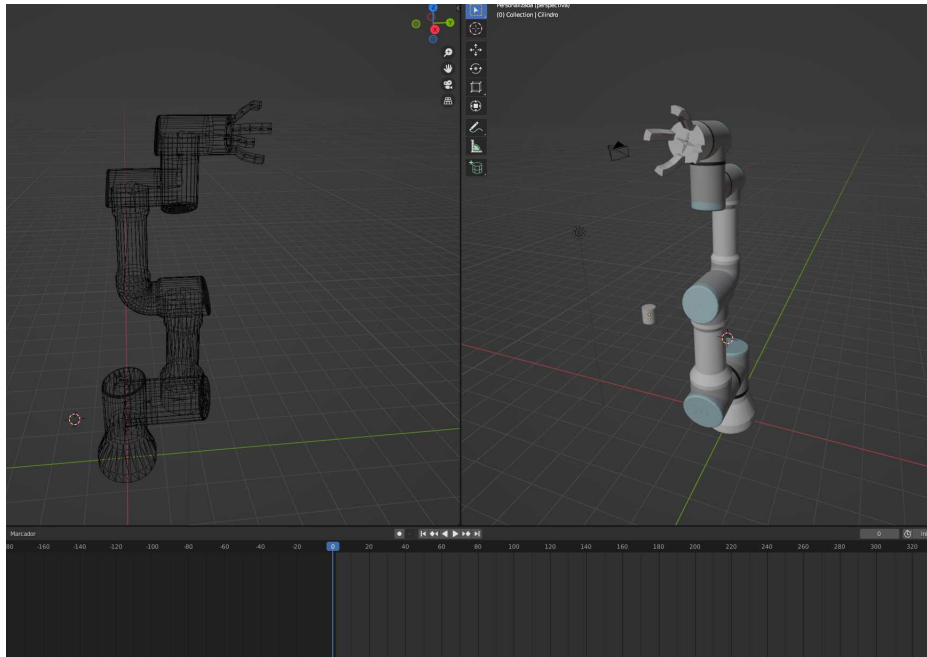


Figura 5.2: UR3 Modelo 3D

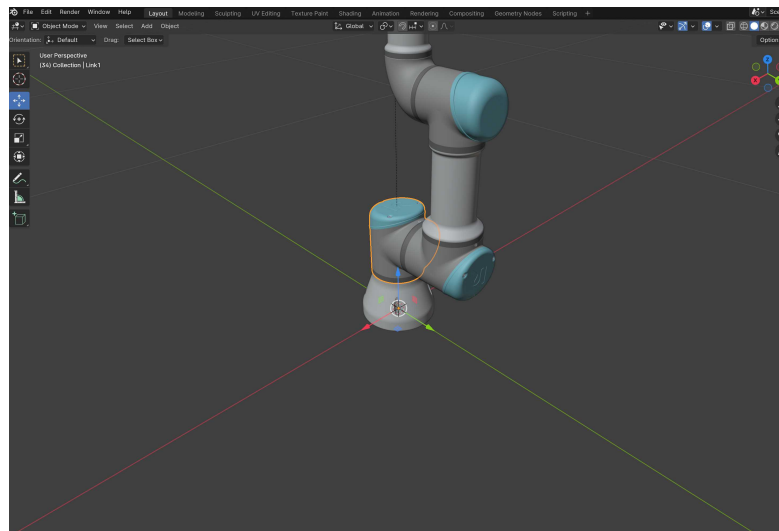


Figura 5.3: UR3 Modelo 3D eje coordenadas en eje "z"

observar en la *figura 5.4* [59], cada articulación tiene una orientación y sentido en el plano cartesiano con referencia el sentido de cada vector, recordar que la posición de cada uno de los vectores es importante, ya que muestran el movimiento que sigue cada articulación.

Como se observa en el modelo del brazo UR3, en el Link 0 es importante colocar el modelo en la posición (0,0) del primer eslabón en el plano cartesiano, de lo contrario se obtiene un desfase de rotación o dirección lo cual puede resultar en aberraciones en los resultados y no precisar su movimiento en la base de cada articulación que presente rotación, las articulaciones posteriores se coloca el set point de cada articulación modelada en el punto donde intercepta con su siguiente eslabón debido a que se toma como el punto de referencia de eje de rotación.

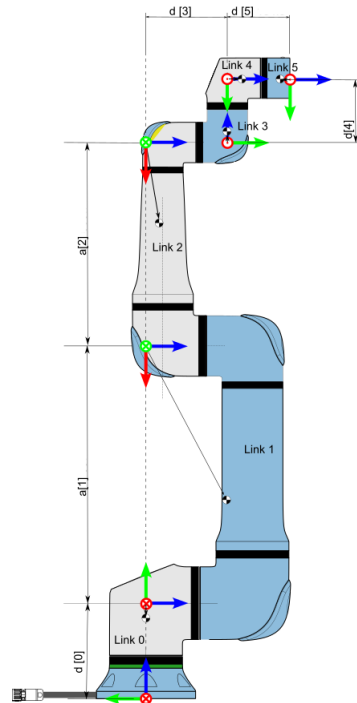


Figura 5.4: Grados de libertad

En el diseño con respecto a las proporciones es necesario tener a consideración los valores que presenta la tabla de distancias entre eslabones propuesta por el fabricante y la tabla de cinemática directa de Denavit-Hartenberg que es posible observar en la *figura 5.5* [59], con esto se tienen los valores precisos para la programación del sistema de ecuaciones necesario para modificar su rotación o traslación con respecto a cada articulación, obteniendo un movimiento fluido, así como una interacción directa con el modelo, experimentando el comportamiento más parecido al modelo real ver *figura 5.6*.

UR3							
Kinematics	theta [rad]	a [m]	d [m]	alpha [rad]	Dynamics	Mass [kg]	Center of Mass [m]
Joint 1	0	0	0.1519	$\pi/2$	Link 1	2	[0, -0.02, 0]
Joint 2	0	-0.24365	0	0	Link 2	3.42	[0.13, 0, 0.1157]
Joint 3	0	-0.21325	0	0	Link 3	1.26	[0.05, 0, 0.0238]
Joint 4	0	0	0.11235	$\pi/2$	Link 4	0.8	[0, 0, 0.01]
Joint 5	0	0	0.08535	$-\pi/2$	Link 5	0.8	[0, 0, 0.01]
Joint 6	0	0	0.0819	0	Link 6	0.35	[0, 0, -0.02]

Figura 5.5: Tabla de Denavid-Hartenberg

Seguido de crear el modelo y colocar cada cualidad en cada una de las articulaciones, se exporta el modelo.

El modelo digital creado en Blender se exporta con la extensión .glb, esta permite exportar el archivo con todas sus propiedades, que después será importado en Unity. Como se observa en la *figura 5.7*, esta extensión importa todas sus características en el software de Unity

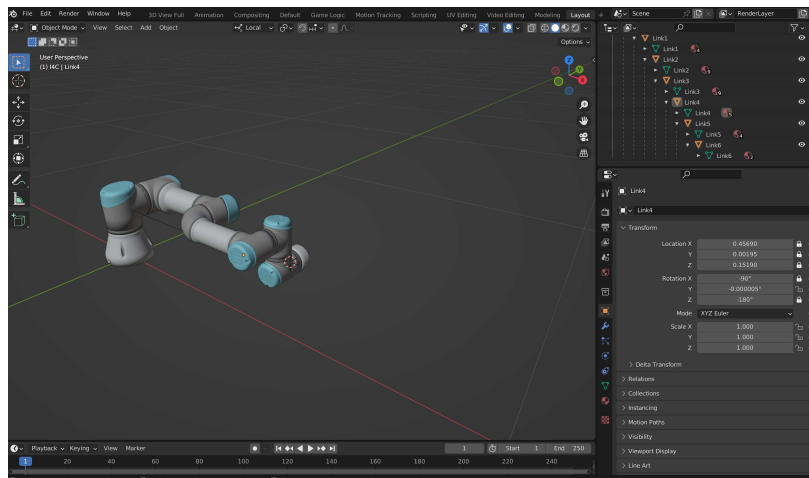


Figura 5.6: Modelo 3D

permitiendo separar cada articulación, facilitando su programación con un Script y manejarlas de forma independiente.

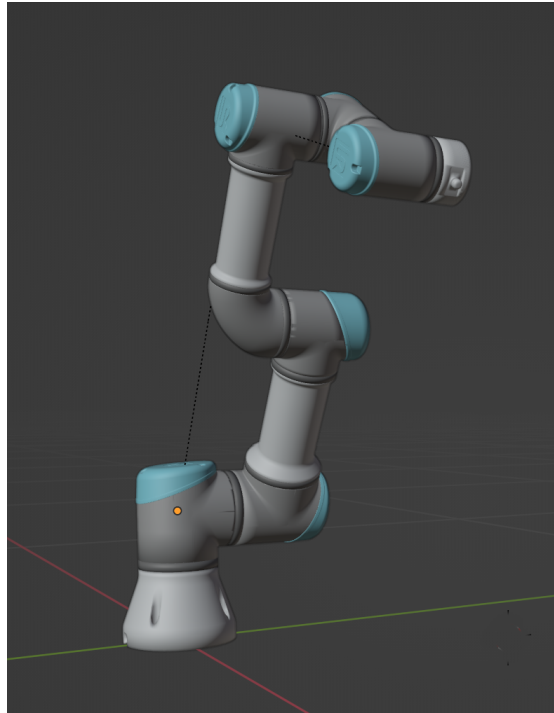


Figura 5.7: Modelo 3D UR3.glb

5.1.2 Interfaz de Integración HoloLens 2

Para la interfaz de integración entre el modelo y los lentes de realidad aumentada HoloLens 2, Unity es el software encargado de crear la interacción entre ambos dispositivos, Unity permite cargar el modelo con todas sus características pre-programadas en Blender, tiene la habilidad de cargar programación de Script en lenguaje “C Sharp” en casi cualquier programa de creación de código. Es importante configurar el proyecto de forma que sea compatible con los lentes HoloLens 2, es necesaria una preparación especial en Unity cargando la paquetería de realidad mixta.

Lo primero es abrir el programa en el Unity Hub se crea un nuevo proyecto 3D, es aconsejable crear una carpeta previamente, donde se coloque solo el proyecto y así poder tener un orden correcto del proyecto. Un dato importante a considerar es ocupar la versión correcta compatible con los HoloLens 2, en el editor de versión colocar la versión 2020.3.28f1, en la parte alta de la ventana, como se observa en la *figura 5.8* si no se encuentra en el editor es indispensable descargarla en la misma plataforma.

Abierto el proyecto ingresar a File -> Build Settings y configurar el proyecto para su compatibilidad con los lentes, seleccionar Universal Windows Platform, en Targeted Device -> HoloLens, en arquitectura -> ARM64 que es la arquitectura del microprocesador de los lentes, seleccionar la opción de Switch Platform y cerrar Build Settings, en la siguiente *figura 5.9* se puede observar la pantalla de configuración de los parámetros.

Descargar la herramienta de MRTK *figura 5.10* [60] es posible localizarla en Windows

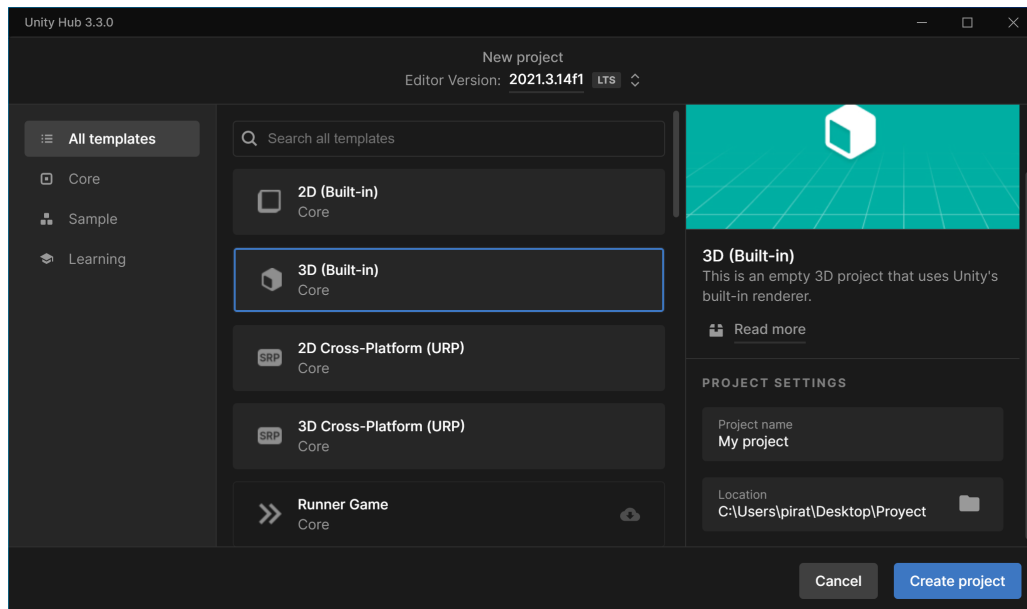


Figura 5.8: Crear nuevo proyecto 3D

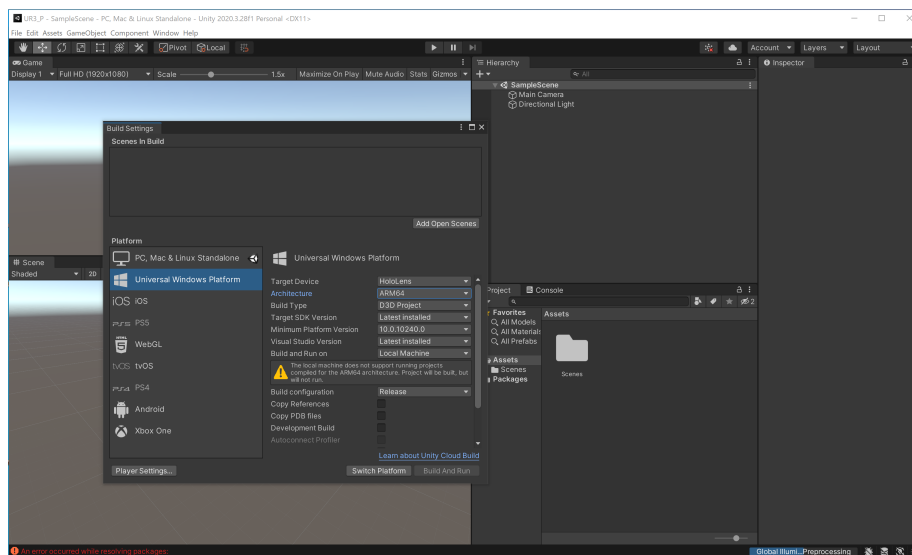


Figura 5.9: Build Settings

Download Center MRTK en el buscador de internet, es importante saber que esta herramienta y la configuración de los lentes solo es posible en la plataforma de Windows 10 o posterior, no es compatible con Apple, otra ruta de descarga es en el siguiente enlace:

<https://learn.microsoft.com/en-us/windows/mixed-reality/develop/unity/welcome-to-mr-feature-tool>

Una vez descargada la herramienta de MRTK se ejecuta MixedRealityFeatureTool ver *figura 5.11*, se selecciona el botón de "start" en la pantalla ver *figura 5.12* y en Project



Figura 5.10: MRTK Logo

Path seleccionar la carpeta donde se encuentre el proyecto presionar Discover Features. En la pestaña de Mixed Reality Toolkit -> Mixed Reality Toolkit Foundation y en Platform Support -> Mixed Reality OpenXR Plugin como se observa en la *figura 5.13*, seleccionar Get Features para así importar y aprobar las características del paquete, salir de Mixed Reality Tool y regresar al proyecto de Unity.

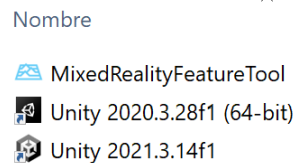


Figura 5.11: MRTK

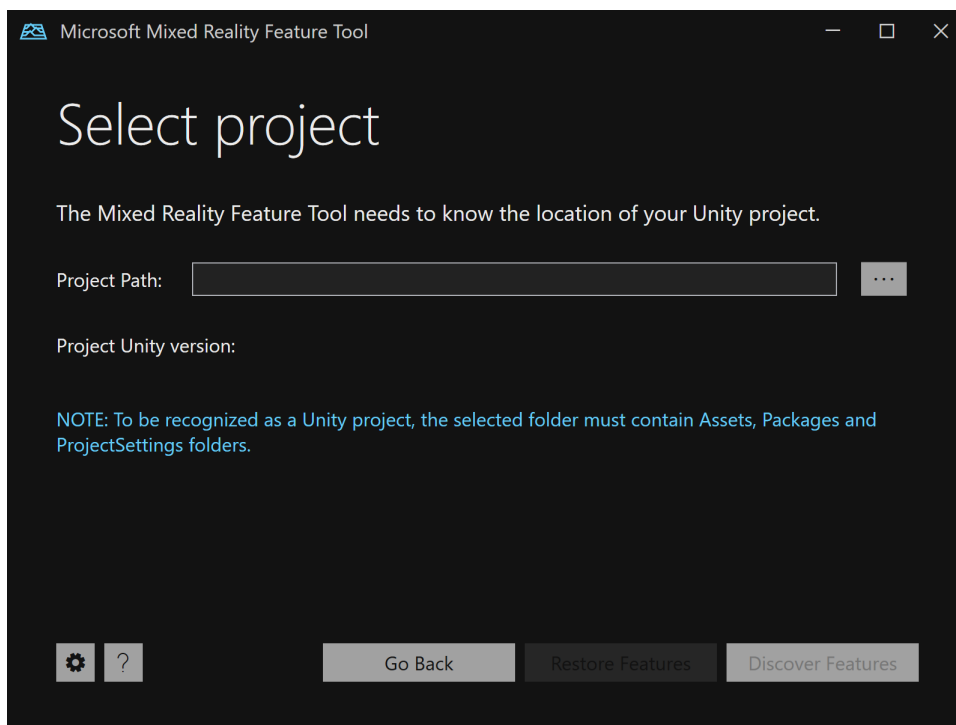


Figura 5.12: MRTK Project Path

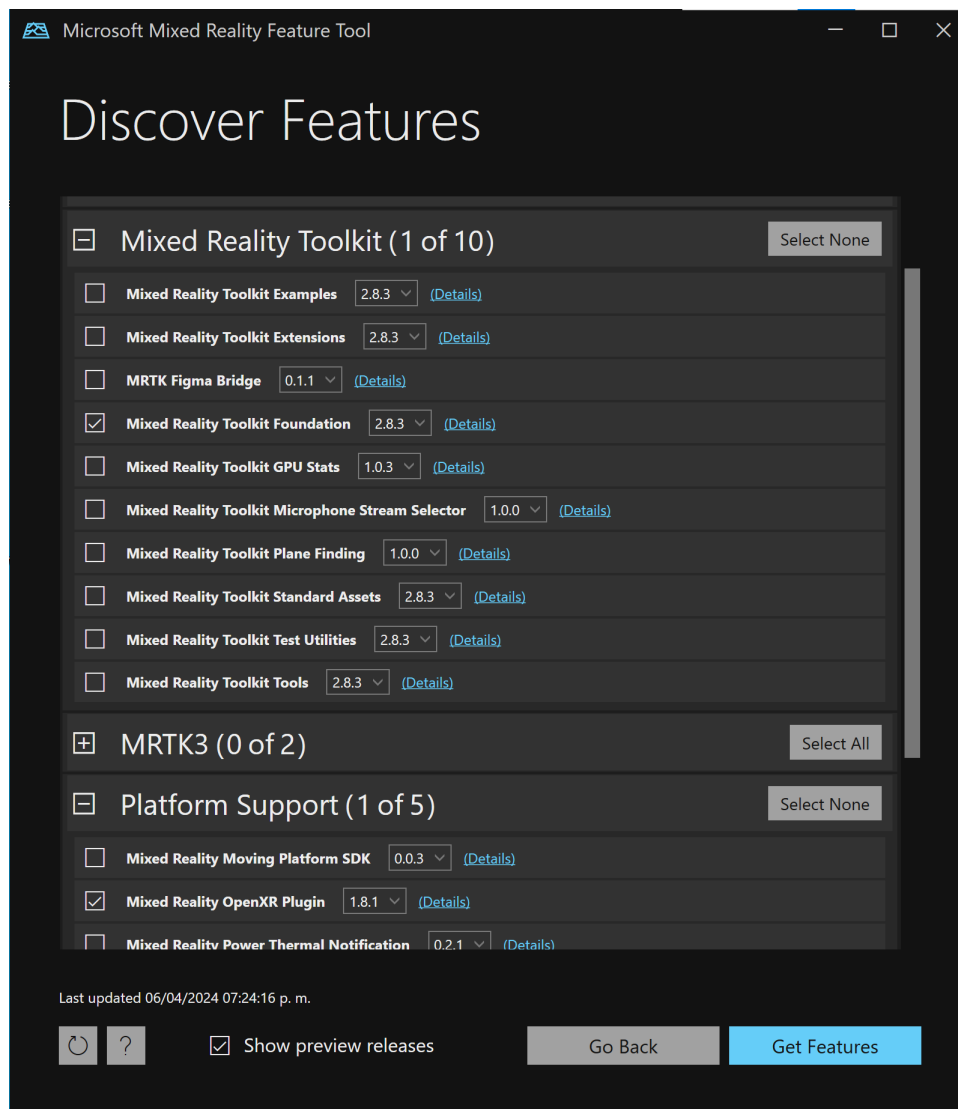


Figura 5.13: MRTK Discover Features

Una vez abierto Unity nuevamente, comenzará a importar todas las características necesarias para poder configurar el proyecto para realidad aumentada, posterior a esto solicita reiniciar Unity, realizarlo, abierto nuevamente el programa aparece la ventana de configuración del proyecto ver *figura 5.14* seleccionar Unity OpenXR plugin, se abre la ventana de Project Settings -> XR Plug-in Management en la pestaña con el símbolo de Windows seleccionar OpenXR -> Microsoft HoloLens feature group regresar a MRTK Project Configurator y seleccionar Next como se observa en la *figura 5.15*.

Nuevamente en la opción de OpenXR -> Depth Submission Mode -> Depth 16 Bit, en Interaction Profiles se tienen la opción de añadir las interacciones que permite la herramienta en los lentes HoloLens 2, en el caso del proyecto seleccionar Microsoft Hand Interaction Profile la cual permite la manipulación de objetos con el sistema de mapeo espacial que cuentan los lentes permitiendo la ubicación de las manos en el espacio ver *figura 5.16*, regresar a MRTK Project Configurator y aplicar los cambios, seleccionar nuevamente Next.

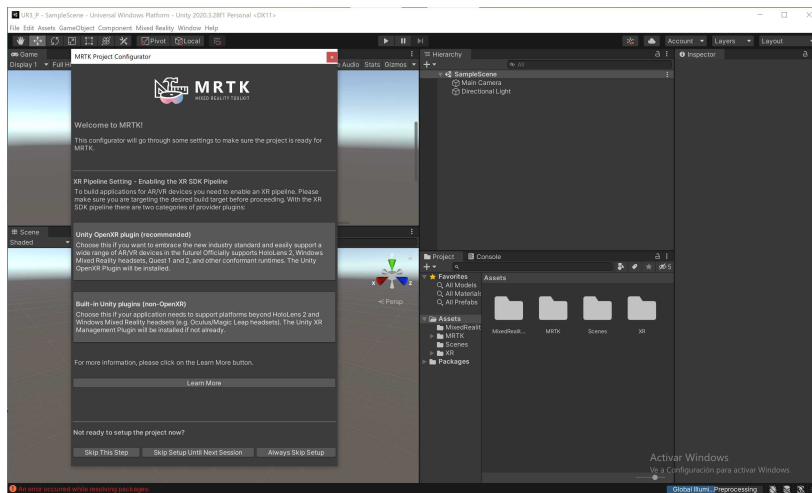


Figura 5.14: MRTK Project Configurator OpenXR

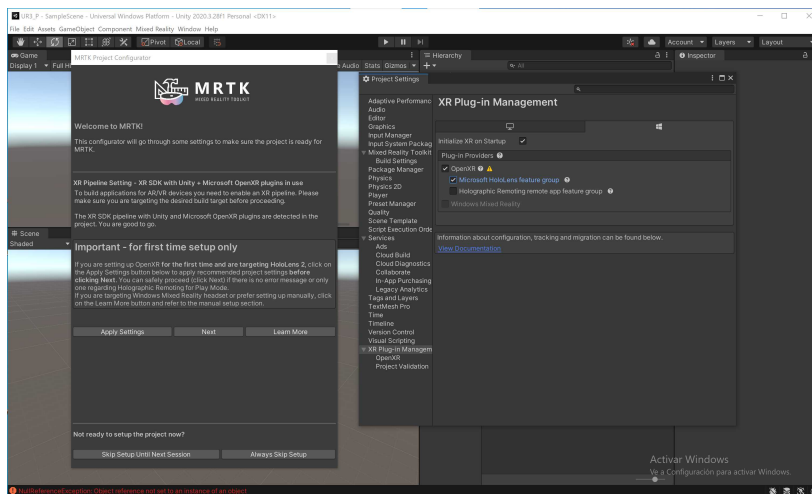


Figura 5.15: MRTK Project Settings

En las pestañas de opciones de Unity se agrega una nueva pestaña Mixed Reality, abrir Toolkit -> Utilities -> Configurar Project for MRTK, en MRTK Project Configurator seleccionar Next, nuevamente Next para poder configurar ver *figura 5.17*, seleccionar Import TMP Essentials y aplicar Done, de esta forma ya se encuentra configurado el proyecto para la implementación de las herramientas de realidad mixta en los HoloLens 2.

Por último, crear una nueva escena y en las pestañas de opciones de Unity ingresar a la pestaña de Mixed Reality -> Toolkit -> Add to scene and configure, esto ajusta los parámetros necesarios para trabajar con las propiedades correctas que requiere el proyecto de realidad mixta como se observa en la siguiente *figura 5.18* donde añade los complementos en la pestaña de jerarquías del proyecto, de esta forma queda configurada la escena en Unity que permite trabajar con los componentes en realidad mixta y la carga de proyectos en los HoloLens 2.

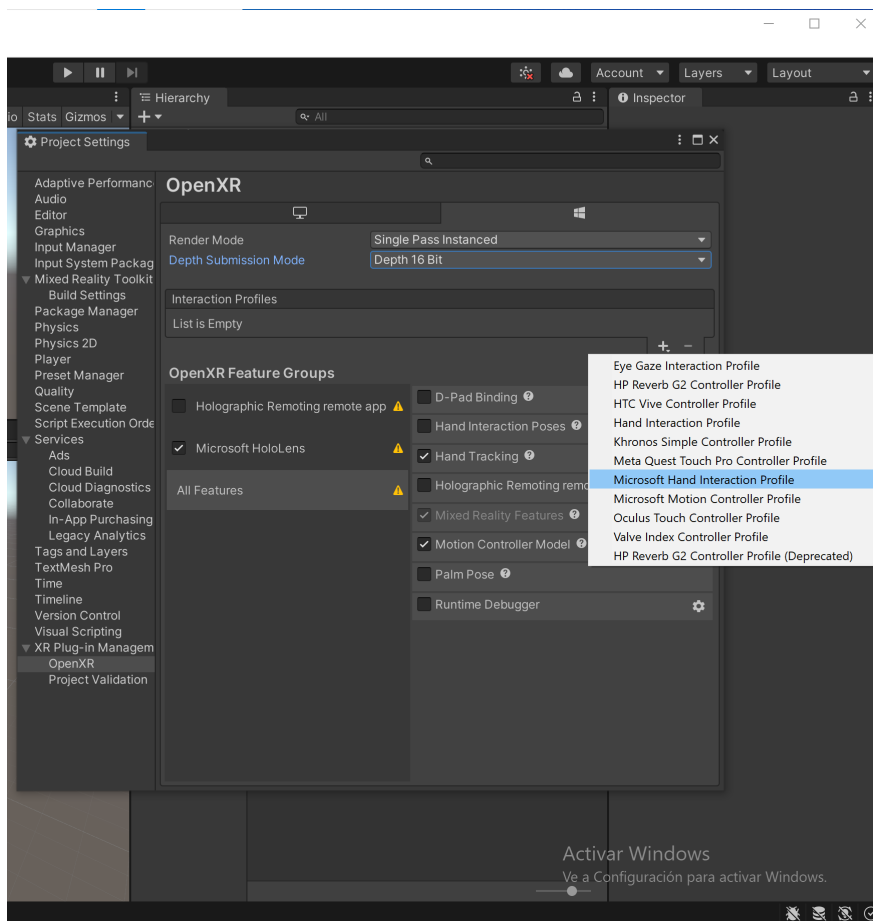


Figura 5.16: MRTK Interaction Profile

5.1.3 Desarrollo de la plataforma en Unity

Una vez configuradas las propiedades del proyecto para trabajar con realidad mixta y el uso de las HoloLens 2, el siguiente paso es importar el modelo del UR3 en formato .glb en la carpeta de modelos del proyecto, por orden y flujo correcto del proyecto se crean carpetas en el asset de cada elemento (Escenas, Objetos, Texturas, Script) observar la *figura 5.19*.

El flujo de la plataforma se diseñó para implementarse en los lentes de realidad mixta HoloLens 2, pero fue indispensable realizar varias pruebas solo en la plataforma antes de realizar la implementación, se creó un diagrama de flujo para comprender de manera más práctica cuál es el comportamiento requerido por la plataforma a ver *figura 5.20*, el proyecto se genera en 4 escenas diferentes que se interconectaran donde cada uno de ellos tendrá su programación diferente basada en la necesidad de cada módulo de operación.

Es necesario cargar tres scripts en cada objeto que se requiera manipular los controles de mando de mapeo espacial con las manos, ver *figura 5.21*, sin estos tres scripts que contiene el paquete de instalación de MRTK no es posible manipular los objetos en la plataforma al cargar en los HoloLens 2, en el botón de añadir componente escribir el nombre de cada uno de los scripts y cargarlo.

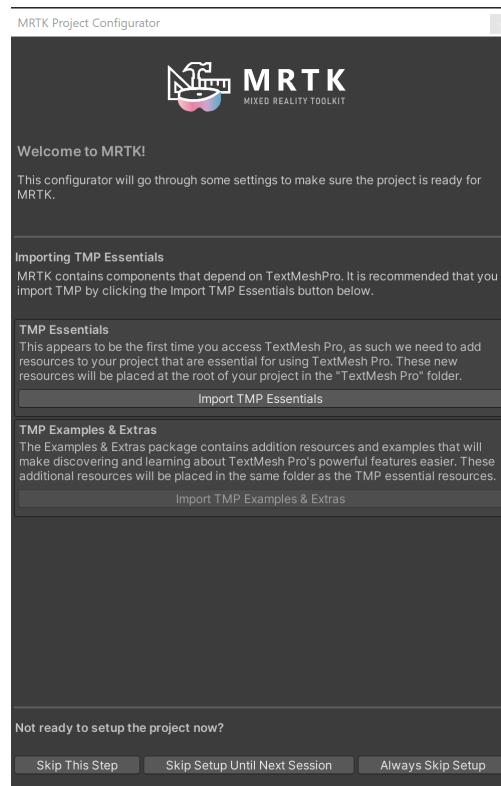


Figura 5.17: MRTK TextMeshPro

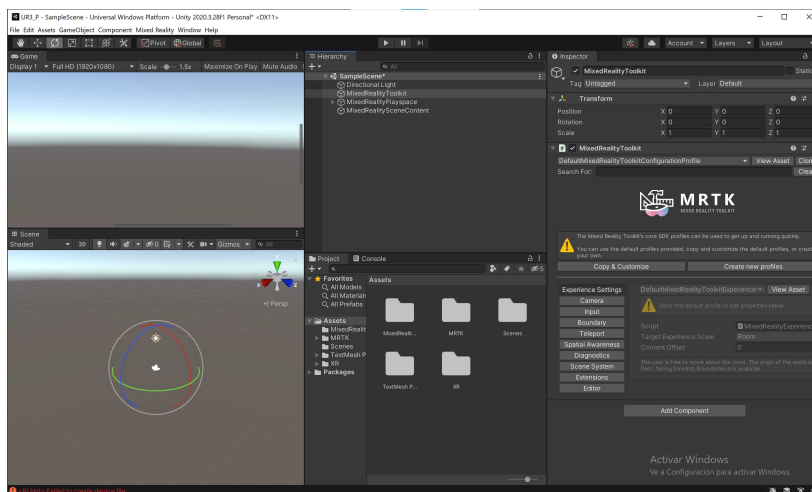


Figura 5.18: Escena Unity y Mixed Reality Toolkit

5.1.4 Menú

El primer módulo implementado fue el menú, ya que es la columna vertebral de enlace entre los módulos de la aplicación. Este panel se encarga de interconectar los 3 módulos adyacentes y ofrecer al usuario una navegación sencilla y práctica dentro de la plataforma, se implementaron 3 botones donde cada uno dirige a una función diferente en la plataforma, se encuentra dividida por:

- Observar

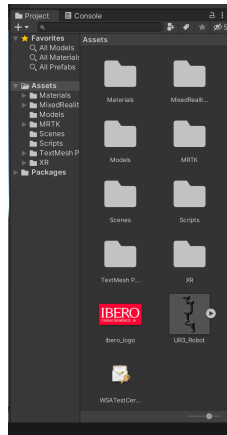


Figura 5.19: Assets del proyecto

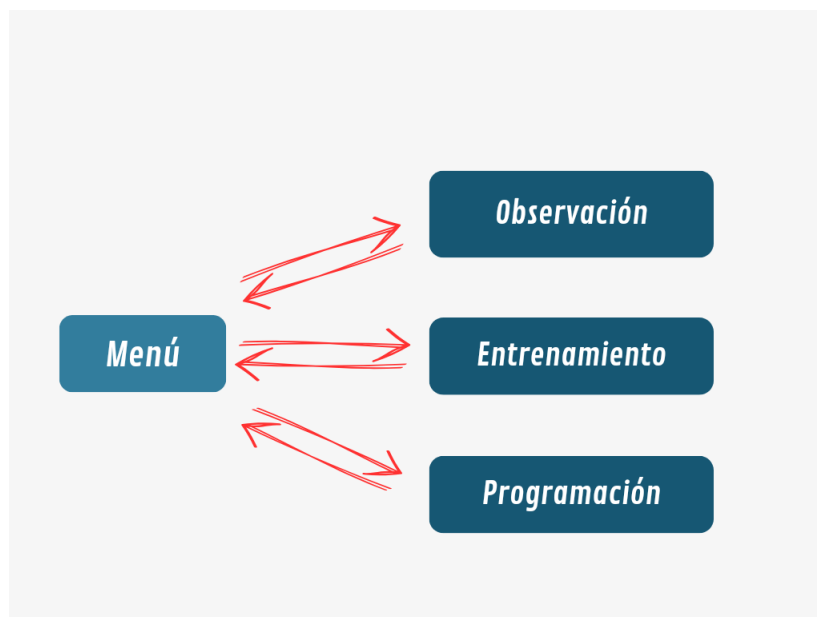


Figura 5.20: Flujo de la plataforma

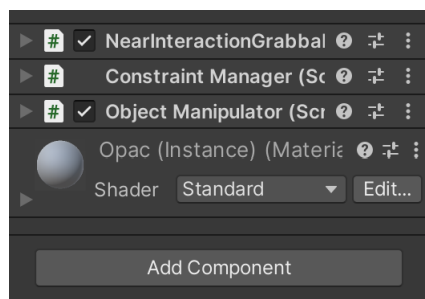


Figura 5.21: Microsoft Hand Interaction Profile

- Práctica
- Programar

Para cada uno de los módulos, así como el menú, se creó un script encargado de realizar

las funciones necesarias para cada acción e interconexión.

La creación, del script o código, es desarrollado en Visual Studio Code y Unity, el código es el encargado de orquestar las acciones que tiene que realizar la animación y la conexión e intercambio de datos entre las gafas y la base de datos. Se recomienda colocar los scripts en un objeto vacío y nombrarlo de una forma sencilla de reconocer.

En el script del menú es creado el mapa de direcciones entre cada una de las secuencias creadas de forma bidireccional, la programación del menú está dirigida a controlar el flujo del programa ver *figura 5.22*, es importante cargar cada una de las escenas en el Build Settings, *figura 5.23* esto es necesario, ya que al compilar el proyecto reconocerá los módulos a compilar que se ejecutaran en la plataforma.

```

1  /// Proyecto de Maestría de Realidad extendida en la Industria 4.0
2  /// Agosto 2022 - Mayo 2024
3  /// Alumno: Ing. Joel Arango Ramirez
4  /// Versión 1.3
5
6  using System.Collections;
7  using System.Collections.Generic;
8  using UnityEngine;
9  using UnityEngine.SceneManagement;
10
11  @ Script de Unity (7 referencias de recurso) | 0 referencias
12  public class direccion : MonoBehaviour
13  {
14      // enrutamiento del menu
15
16      @ Referencias
17      public void loadFisicas(string sceneFisica)
18      {
19          SceneManager.LoadScene(sceneFisica);
20      }
21
22      @ Referencias
23      public void loadEnsamble(string sceneEnsamble)
24      {
25          SceneManager.LoadScene(sceneEnsamble);
26      }
27
28      @ Referencias
29      public void loadDirecta(string sceneDirecta)
30      {
31          SceneManager.LoadScene(sceneDirecta);
32      }
33
34      @ Referencias
35      public void loadMenu(string sceneMenu)
36      {
37          SceneManager.LoadScene(sceneMenu);
38      }
39  }

```

Figura 5.22: Flujo de conexión de módulos

El panel del menú es re-escalable y permite movilidad espacial con las manos. El movimiento espacial es posible por las características de los sensores de mapeo de las gafas HoloLens 2.

El diseño del menú se creó pensando en un tamaño donde sea práctico y fácil de manipular por el usuario, ver *figura 5.24*.

Algunas características importantes en el menú, es la forma de la construcción colocando como “child” o hijos la mayoría de los componentes como se puede ver en la *figura 5.25*, esto es debido a que al mover de posición la base del menú, requiere un movimiento en conjunto de los demás elementos, a lo cual se coloca en escalera los componentes de lo contrario su movimiento sería dispar, como se observa en la *figura 5.26*.

En cada uno de los botones se implementó el perfil de interacción con las manos para hacer posible la selección de los botones cuando sean actualizados en las gafas.

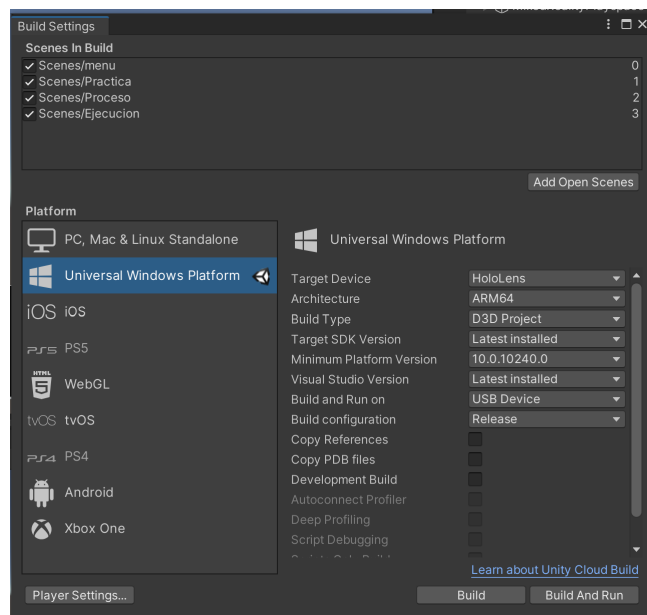


Figura 5.23: Build Settings

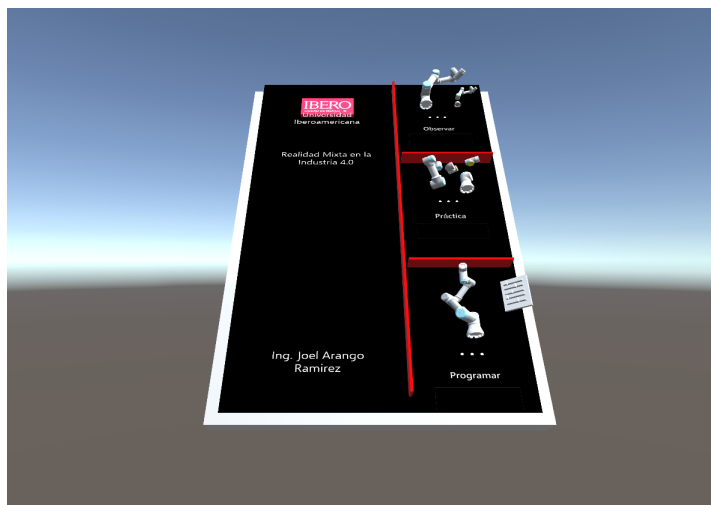


Figura 5.24: Menú principal

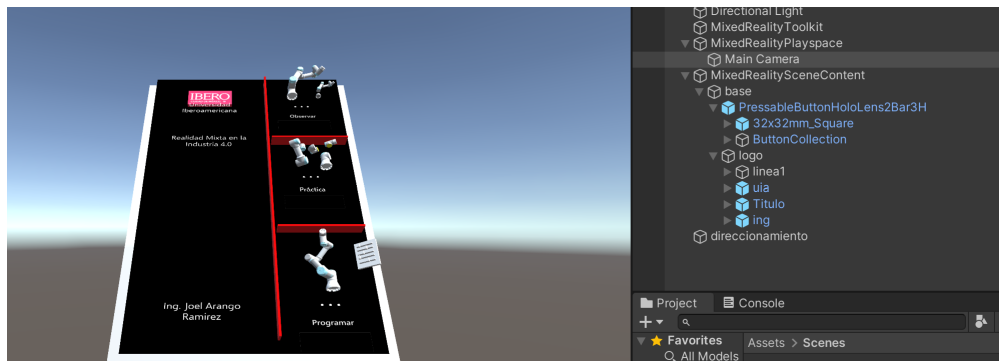


Figura 5.25: Componentes anidados

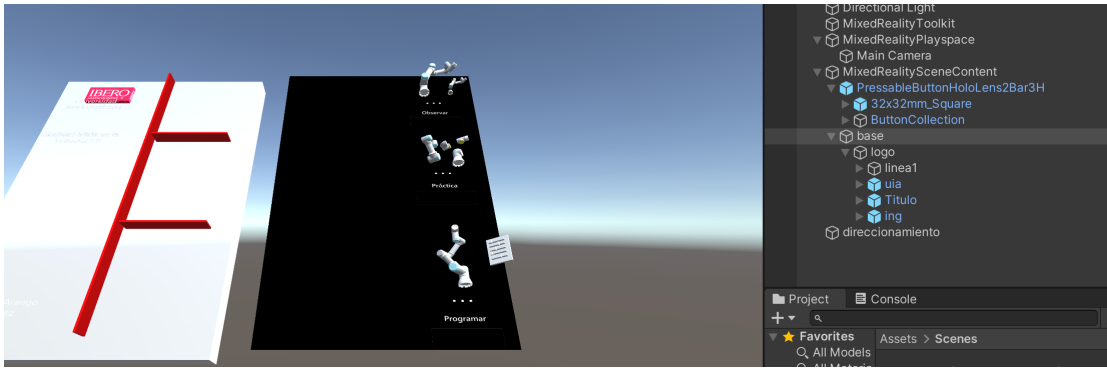


Figura 5.26: Componentes no anidados

5.1.5 Firebase

Firebase es una plataforma de desarrollo en la nube, que ayuda a crear e implementar aplicaciones y sistemas de bases de datos en tiempo real, entre otras. Es un sistema respaldado por Google brindando seguridad en los protocolos de comunicación [61], Firebase se ha posicionado como una de las plataformas más fuertes y segura para la transferencia y almacenamiento de datos en tiempo real. *figura 5.27* [62]



Figura 5.27: Firebase

Para el proyecto se crea una base de datos en la plataforma de Firebase, permitiendo la interacción entre el brazo real UR3 y los Lentes HoloLens 2, se crea una cuenta nueva gratuita y se agrega un proyecto nuevo con el formato de base de datos en tiempo real, automáticamente genera un identificador único que será usado para enlazar la plataforma en el script de Unity y de Python.

5.1.6 Panel Observa

El primer panel que se encuentra por seleccionar en el menú es “Observa” *figura 5.28*, este panel permite observar el comportamiento que tiene el brazo UR3 real en su funcionamiento.

En este módulo se busca mostrar el funcionamiento y comportamiento que tiene el brazo UR3 real, permitiendo observar el proceso a distancia, obteniendo visualmente algún comportamiento erróneo o que fluye de manera normal como fue programado, como se observa en

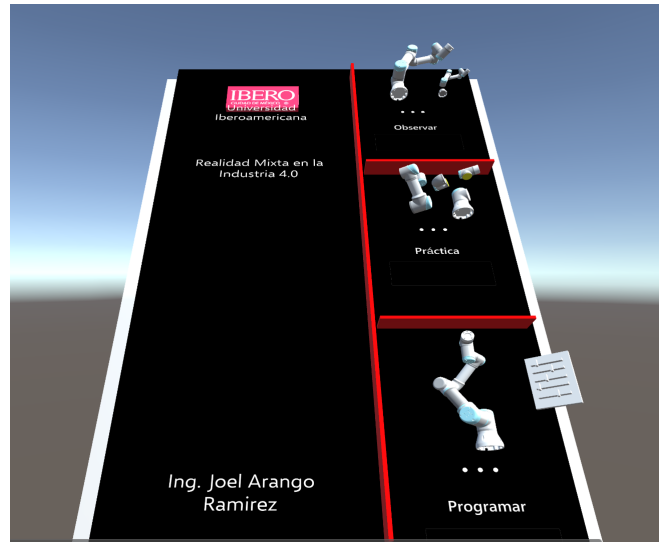


Figura 5.28: Menú de aplicaciones

la *figura 5.29*, toma gran relevancia si es necesario capacitar usuarios o simplemente mostrar como se comporta cada una de sus articulaciones, ya que es una copia digital del modelo real y convirtiéndose en un auxiliar al aprendizaje del comportamiento del UR3.

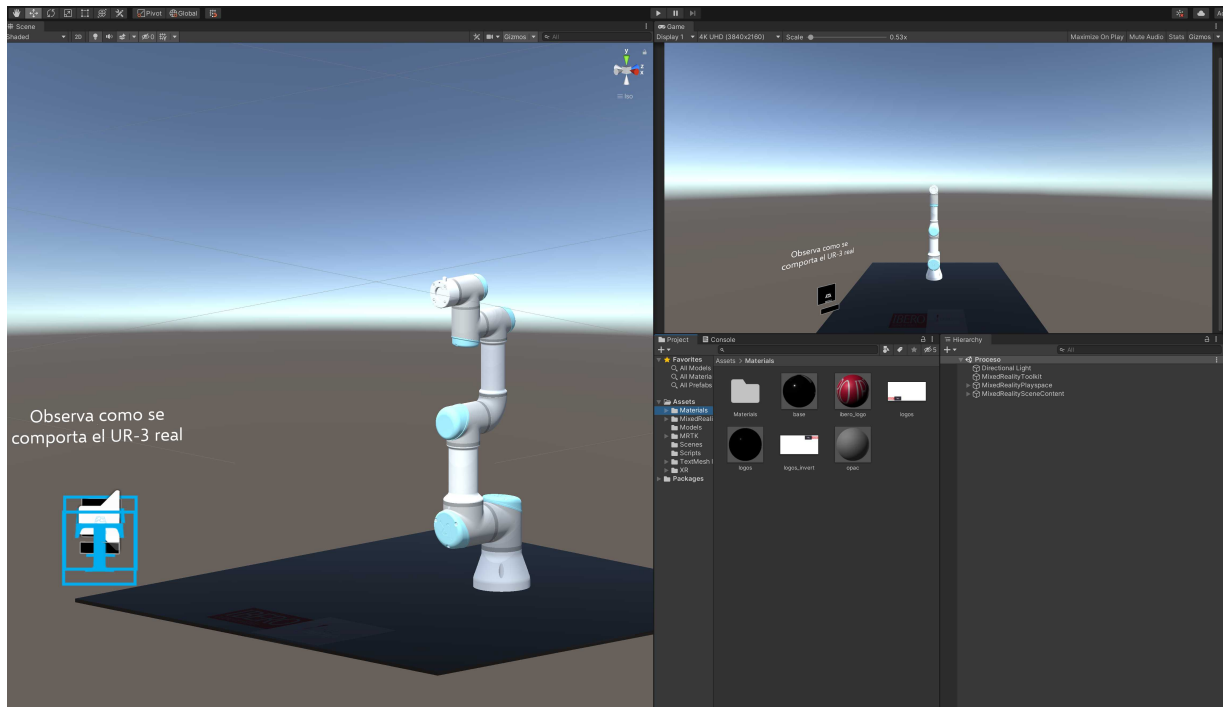


Figura 5.29: Modelo de observación en Unity

Para este caso solo es posible la manipulación espacial y escalar de la base que soporta el modelo, ya que no permite la manipulación ni movimiento de ninguna de las articulaciones al solo presentarse como un modelo de observación.

El desarrollo del script de observación es necesario añadir cada una de las articulaciones como un GameObject por separado, para posteriormente programar cada articulación con cada valor obtenido por la base de datos y así reconoce a cada componente de la plataforma, esto permite su interacción con los demás componentes, se colocan como una variable pública para que sean visibles en Unity, es necesario arrastrar cada articulación con su par y esto implementara la interacción con el script como se observa en la *figura 5.30*.

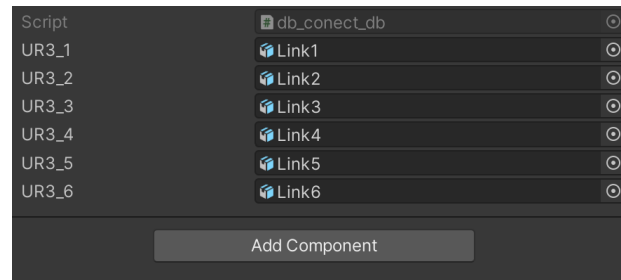


Figura 5.30: Conexión GameObject a script

Para hacer posible la interacción entre el gemelo virtual y el brazo UR3, es necesario la interconexión con la base de datos donde recopila toda la información necesaria para hacer que cada una de las articulaciones tenga movilidad, estos datos son recabados del brazo real del UR3 en la base de datos en tiempo real Firebase.

El protocolo utilizado para esta conexión con Firebase fue TCP/IP permitiendo el intercambio de paquetes de información, entre la base de datos en la nube y el script que se implementa en el brazo digital, para esto se añaden las librerías `UnityEngine.Networking` es una clase que contiene constantes para los canales de la red predeterminados, lo cual permite la apertura de las comunicaciones.

La segunda biblioteca que se implementó es la encargada de tratar con datos de tipo Json, ya que la base de datos de Firebase se comunica a través de estos datos al ser una base de datos no relacional. La paquetería `Newtonsoft.Json` permite la implementación y comunicación de estos tipos de estructuras, aligerando el peso de la información y haciendo más eficiente el intercambio de datos. Con la implementación de las bibliotecas en el script, se configura el protocolo de conexión con la base de datos de Firebase y la plataforma en los HoloLens 2.

Lo que respecta al script la configuración de la base de datos es necesario colocar la URL de referencia de Firebase, esta se encuentra en el encabezado del proyecto en la plataforma de la nube, la estructura de petición de datos es por el protocolo Get en el script, que se encarga de solicitar los valores de cada articulación almacenados en la nube.

Lo principal es crear un `UnityWebRequest` para la conexión con la base de datos, donde se coloca get como petición y la URL de referencia, este protocolo permite solicitar los datos en formato Json a la base de datos.

Al ser Json los datos la forma de tratamiento es diferente a una cadena de datos normales, para ello es necesario serializar o deserializar los datos, Json es un formato liviano para el intercambio de datos, la serialización es un proceso de codificación de un objeto en un medio de almacenamiento con el fin de transmitir por la red como una serie de bytes. El proyecto se requiere deserealizar los datos que provienen de la base de datos en la nube, para poder tratarlos y conectarlos con cada una de las articulaciones, es importante seguir el protocolo de la cinemática directa, ya que se tiene que hacer un ajuste de grados según la tabla de Denavit-Hartenberg y así controla cada una de las articulaciones por separado hasta llegar al efector final.

Este módulo de la aplicación está dirigido al aprendizaje y observación del funcionamiento del brazo virtual con respecto al real, permite al usuario tener una idea clara de flujo y comportamiento que se tiene con respecto al movimiento de ángulos de cada articulación, preparando al usuario para trabajar más adelante con el control de programación.

En las pruebas realizadas con el brazo UR3 se logró observar que recibe los datos de forma correcta, tomando la misma posición que el UR3 real, como se observa en las siguientes *figuras 5.31*.

5.1.7 Panel práctica

El desarrollo del módulo de práctica es el menos complejo, ya que está diseñado para que el usuario tenga una interacción directa con el modelo, donde lo más importante es la manipulación del gemelo virtual sin tener ninguna repercusión con el modelo real, esto hace que el modelo no tenga una comunicación con la nube y a su vez con el brazo real minimizando la parte de programación.

La plataforma de práctica, se desarrolló creando una base circular en el espacio donde reposa el UR3 que puede tomar con sus manos el usuario y manipular libremente, se desactivaron las físicas para evitar que la gravedad lleve el módulo al suelo y el usuario tenga distracciones o pérdidas del modelo en la práctica, se asociaron dos botones a la base, uno para regresar al menú principal y el segundo para regresar el modelo a la base en caso de que el usuario lo pierda espacialmente como se observa en la *figura 5.32*.

En el área de Jerarquía del módulo, se crea un objeto vacío donde se coloca el script que implementa la programación entre la base y el modelo para ejecutar la instrucción de regresar el modelo a la base.

Es posible el movimiento espacial del brazo UR3 así como la escalabilidad del modelo, como observamos en la *figura 5.33*, para permitir esto se añadió la funcionalidad de manipulación con las manos, como se comenta anteriormente en el texto donde para manipular un objeto son necesarios 3 scripts que contiene la paquetería MRTK, de lo contrario no sería posible realizar dicha acción. Para este módulo la base no es manipulable espacialmente.

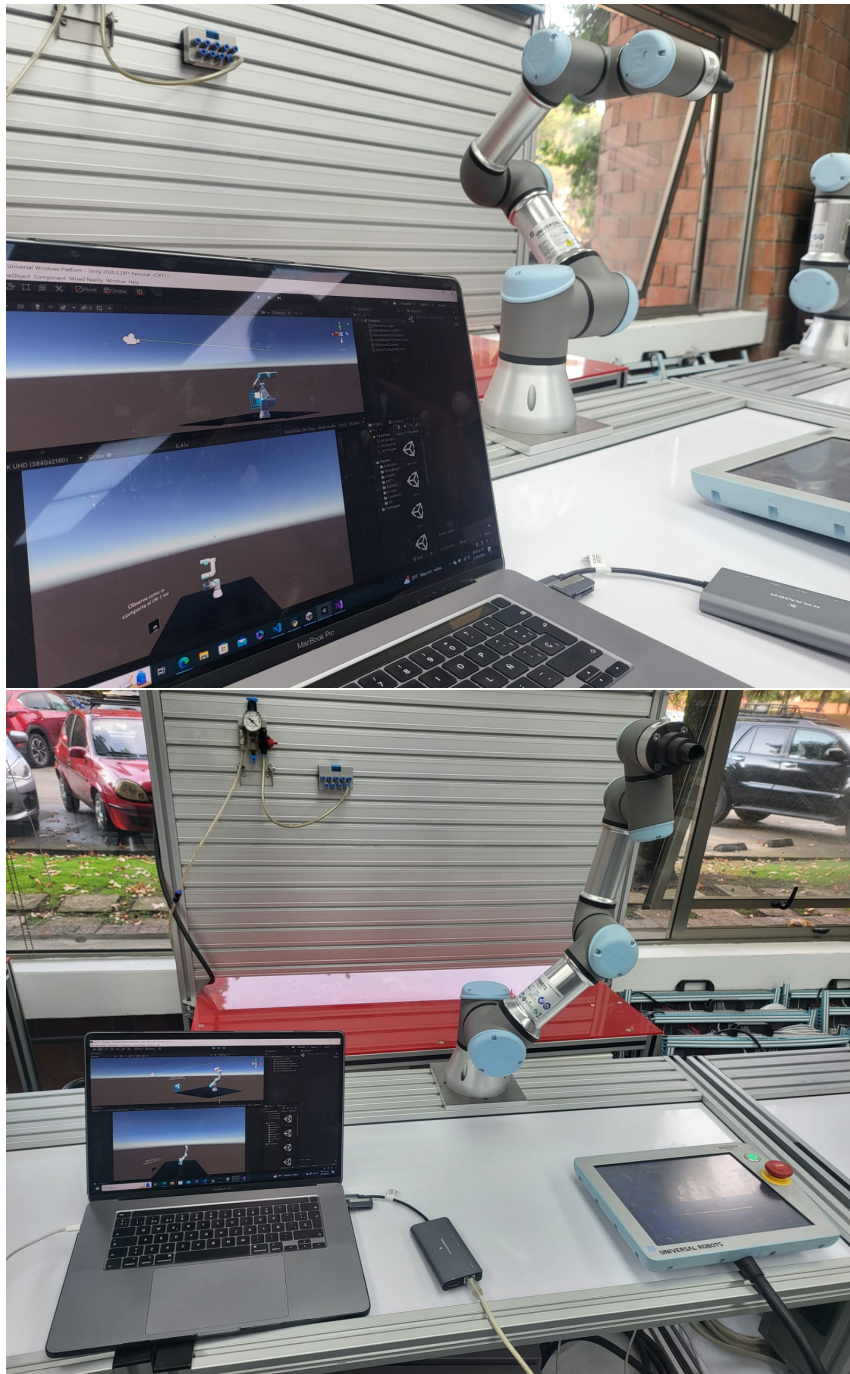


Figura 5.31: Comportamiento Unity - UR3

5.1.8 Panel programar

El panel más complejo es el de programar, está diseñado para tener una conexión bidireccional entre los lentes HoloLens 2, el servidor y el brazo UR3 como observamos el flujo en la siguiente *figura 5.34*, esto permite una interacción entre todas las partes al mismo tiempo lo cual genera una complejidad entre la comunicación.

El diseño de este módulo mantuvo la estructura, el modelo UR3 sobre una plataforma que

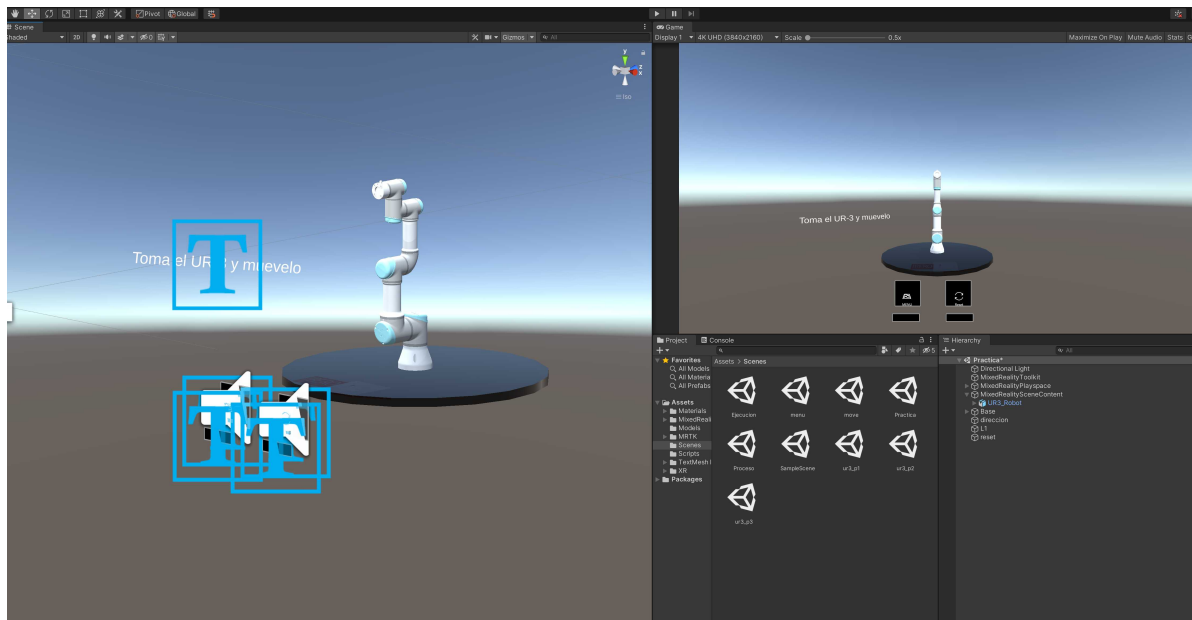


Figura 5.32: Modelo de práctica

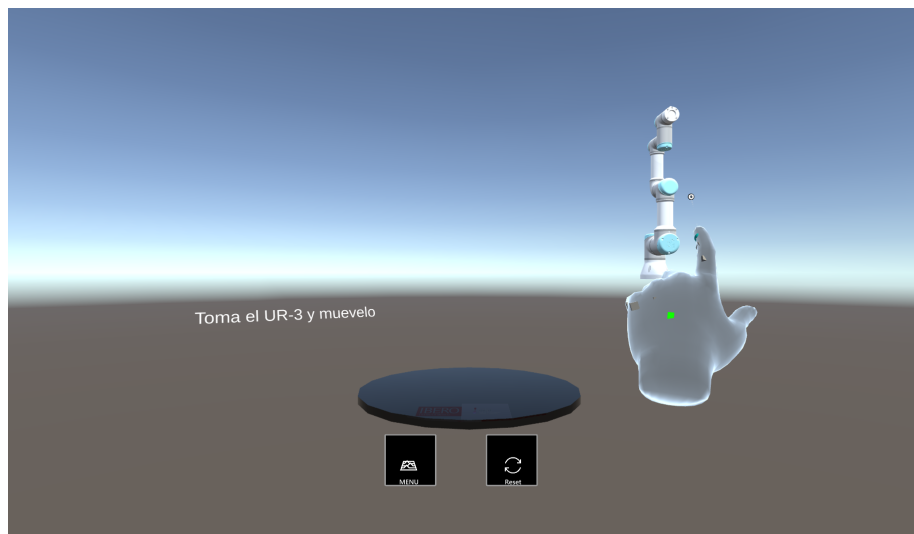


Figura 5.33: Prueba de manipular objetos Unity

es manipulable de tamaño y colocación, pero no así el brazo UR3. A diferencia del módulo de observación, se implementó un panel de botones que está diseñado para mover cada una de las articulaciones por separado del brazo UR3, como observamos en la *figura 5.35*.

El panel de programación está diseñado con dos botones para cada una de las articulaciones, esto permite al usuario tener el control de cada articulación en ambas direcciones, la configuración en la ramificación de Unity como “hijos” de la base a la última articulación del efector final, permite que al mover la articulación primera mueve las que se encuentren asociadas en la jerarquía, como se observa en la *figura 5.36*, esta jerarquía es indispensable para el correcto flujo del UR3 generando inercia entre las articulaciones.

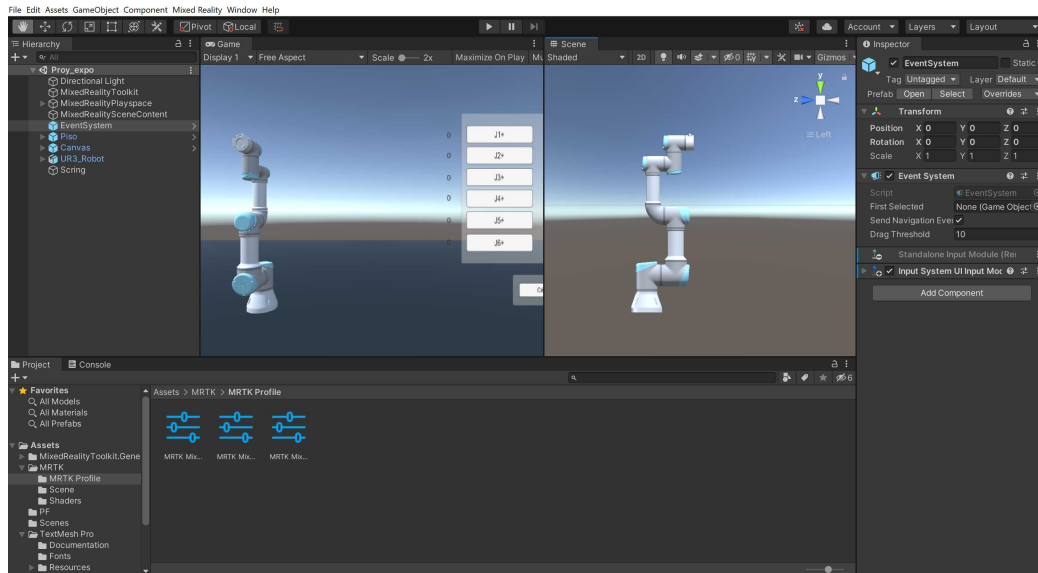


Figura 5.34: Módulo Programación Unity

Se incluyeron 3 botones más en el panel de programación, el primero es una secuencia realizada previamente por el usuario que envía las posiciones de las articulaciones a la base de datos para posteriormente ejecutara el brazo real, en el botón siguiente la base con el brazo UR3 regresa espacialmente y de tamaño a la posición inicial donde se encontraba cuando se seleccionó la opción de programación en el menú inicial, el tercero manda al UR3 al punto programado como “home” que es la posición usualmente como se inicia el proceso del brazo, en la parte baja del panel de programación se coloca el botón de Menú que regresa a la selección de los módulos.

Con respecto a la programación del script se crea un GameObject vacío, donde se coloca el archivo del programa que ejecutará la operación de interacción entre la aplicación en los lentes HoloLens 2 la base de datos y el brazo UR3 de forma bidireccional, sobre el inspector del GameObject creado, se añade el script donde se coloca cada una de las articulaciones del brazo UR3 digital, como se observa en la *figura 5.37*,

Al direccionar el panel de programación del menú, se ejecuta una función encargada de solicitar los valores de la base de datos, colocando el UR3 digital en la posición final de valores que se guardó del brazo real, para evitar errores en el intercambio de datos de programación por articulación. La función para realizar la petición es igual a la ocupada en el módulo de observación donde se aplicó la función “Get”, esto permite solicitar información e interacción con la base de datos y solicitar la última posición almacenada de cada articulación, colocando el UR3 virtual en la posición correcta.

Ya colocado el modelo UR3 en la posición última, el usuario tiene la posibilidad de controlar cada una de las articulaciones con el panel de programación continuando la secuencia de posiciones de cada articulación. Después, se creó una clase nueva de “strings” o cadenas, para almacenar los datos que se intercambiarán entre el modelo y la base de datos con la propiedad “get; set” que permite acceder y enviar la información con el servidor, cada cadena almacenará el valor de cada articulación.

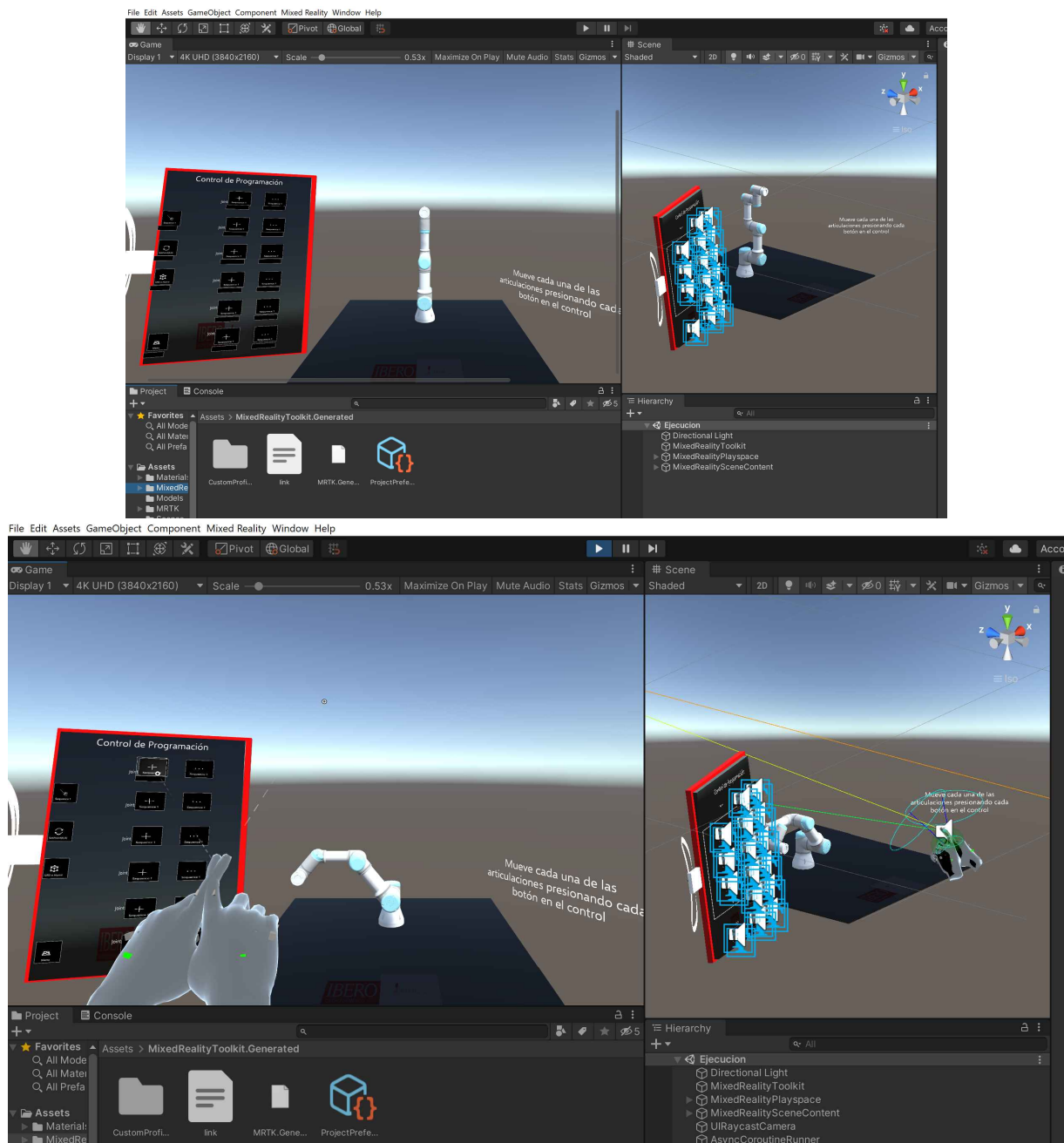


Figura 5.35: Botonera Unity

Con respecto a la programación del script para el envío de datos a la nube, se requiere la función “Put” que permite actualizar los valores en la base de datos cada que se presiona cualquier botón de las articulaciones, dato importante es convertir los valores recabados a formato Json con la función “JsonUtility.ToJson” de la clase creada para almacenar los valores, esta instrucción permite convertir los valores almacenados en una cadena en datos Json, que son los aceptados por la base de datos, una vez convertidos los datos se serializan y se envían a la base de datos en el servidor.

Para el envío de datos es importante colocar el paquete en la función de conexión con la

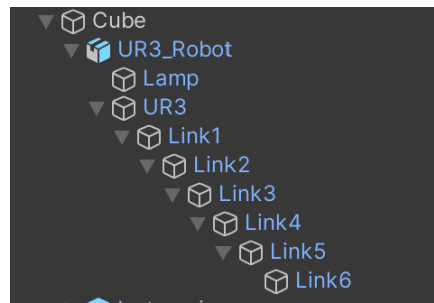


Figura 5.36: Jerarquía UR3 Unity

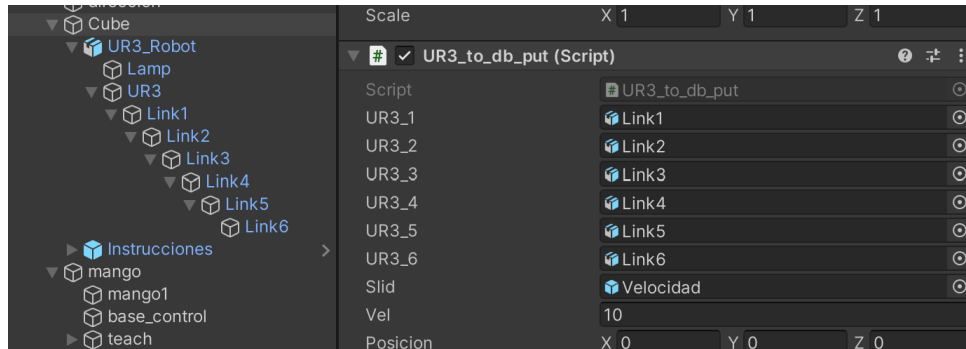


Figura 5.37: Articulaciones UR3 Unity

base de datos creando un “UnityWebRequest.Put(vínculo con servidor, cadena de los datos)” esto permite conectarse con la base de datos en el servidor, a su vez que envía la actualización de los datos.

Para cada adquisición de datos de los botones, se generó una función que se encarga de almacenar y ejecutar el valor en el UR3 virtual y enviarlo a la base de datos en el servidor, es importante configurar cada una de las funciones creadas en las preferencias de cada botón, ya que detecta que se pulsó el botón y ejecute la instrucción, para ello es necesario ir al inspector de cada botón y en el panel de eventos básicos crear un evento “OnClick()”, esta función permite detectar cuando se presiona un botón, y ejecutar la acción programada. Cargar el GameObject donde se colocó el código y seleccionar la ruta de la función creada para ejecutar la acción como se puede observar en la *figura 5.38*.

5.1.9 Adquisición de Datos UR3

Para la adquisición de datos del Brazo UR3 con la base de datos se muestra un diagrama general del flujo que seguirá esta conexión como se muestra en la siguiente *figura 5.39*. En el editor de código Visual Studio Code, se crea el código en Python con la librería URX que tiene la capacidad de tener una conexión bidireccional con el servidor y la base de datos en tiempo real, se añaden las librerías necesarias para la comunicación de ambos dispositivos, como observamos en la *figura 5.40*.

La librería de Firebase administra y valida la comunicación creando el enlace con la base de datos en la nube e interacción segura en tiempo real con los datos.

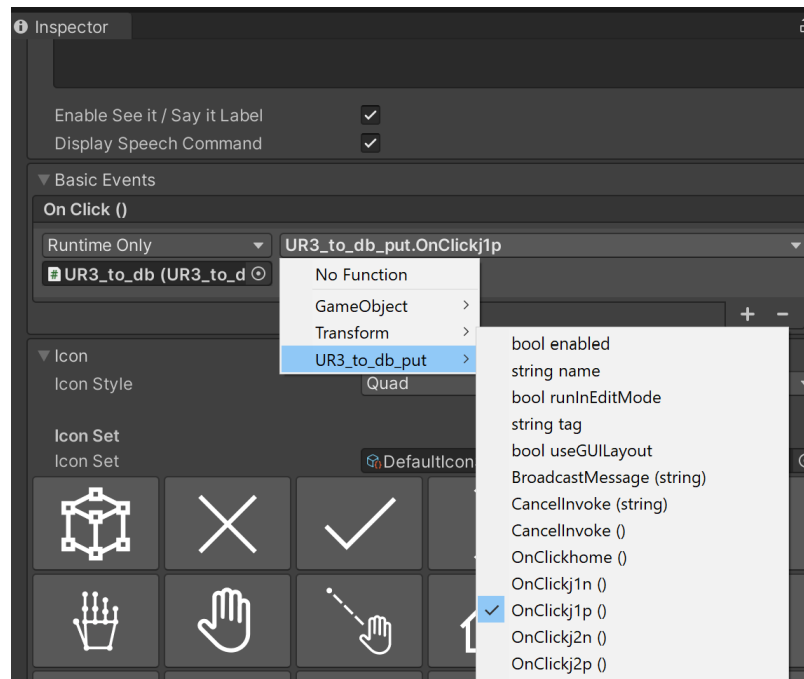


Figura 5.38: Función de acción de botón Unity

TKinter es una interfaz gráfica para Python, con esta librería se crea la aplicación gráfica para la computadora, esta interfaz permitirá que el usuario se conecte por medio de IP con el brazo UR3, facilitando al usuario seleccionar el brazo que utilizará. Se crea en código la ventana que solicita al usuario la IP del brazo a conectar, botones de conexión, desconexión, home, subida y bajada de información con la base de datos de Firebase, como se observa en la *figura 5.41*.

La configuración inicial de conexión, es la implementación de comunicación entre la computadora y el brazo UR3 por medio del protocolo IP creando una red local entre ambos, en la computadora y el UR3 se colocó una IP estática para mantener la conexión estable, los dispositivos se conectan por Ethernet entre ellos, y para el envío de datos a la nube la computadora se conecta a internet por wifi.

La conexión con la base de datos se genera descargando las credenciales de Firebase que permite tener los permisos necesarios para ingresar y manipular los valores, las credenciales se descargan de la plataforma de Firebase desde la página web, se guarda en la carpeta del proyecto donde más adelante será requerida.

En Python se configura la conexión con Firebase, con la siguiente función en Python

“`firebase_admin.credentials.Certificate(ruta de credenciales)`”, inicializar la conexión con la base de datos “`firebase_admin.initialize_app(firebase_sdk,'databaseURL': 'url de la base de datos')`”, con este proceso el enlace queda activo.

La librería Urx está diseñada para tener una comunicación bidireccional de datos entre el brazo UR3 y el programa de Python, se crea una variable que almacena la IP en un objeto

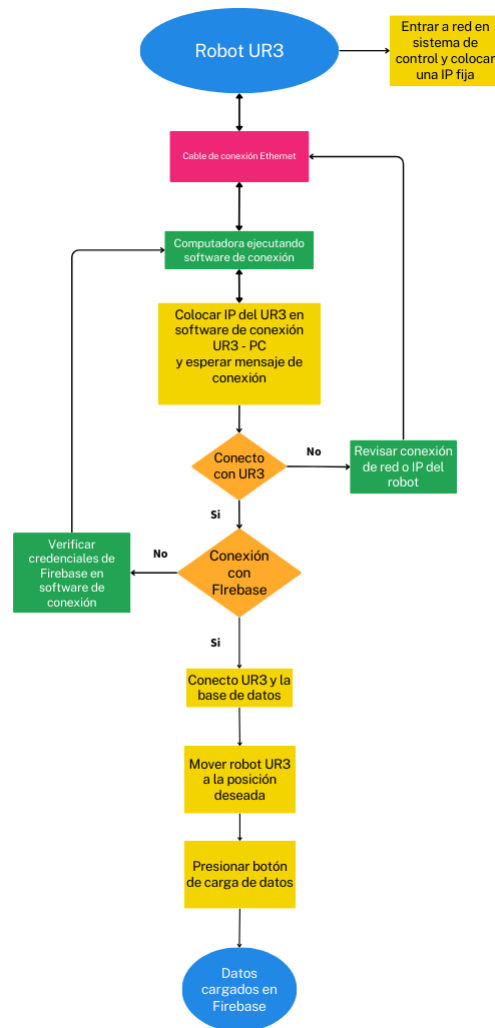


Figura 5.39: Diagrama de flujo adquisición de datos UR3

```
#!/usr/bin/env python3.6
import firebase_admin
from firebase_admin import credentials
from firebase_admin import db
import urx
import math
import keyboard
import time
import sys
from PIL import Image, ImageTk
from tkinter import Tk, Label, Button, Entry, messagebox
```

Figura 5.40: Librerías Python Visual Studio Code

de la librería como “`urx.Robot(IP)`”, esto genera la conexión con el UR3.

```

#Creamos la ventana de la aplicación principal

win_program_n = Tk()
win_program_n.geometry("500x600")
win_program_n.title("Conexión UR3 - DB")

img = Image.open('')
img = img.resize((100,50), Image.ANTIALIAS)
img = ImageTk.PhotoImage(img)
lbl_img = Label(image = img)
lbl_img.place(x=200,y=30)

```

Figura 5.41: Código interfaz gráfica de conexión

La petición de datos al UR3 para subir a la base de datos se realiza llamando la función `getj()` o `getl()` dependiendo del tipo de movimiento que se encuentra en la estructura de programación del robot, los datos son tratados por la biblioteca “`math.degrees()`” esta despliega en grados el valor que recibe de cada una de las articulaciones, se crea una variable que almacenará cada uno de los datos recabados del brazo UR3 y se nombre a cada articulación para subir los datos individualmente, ya que esto permitirá la interacción de cada valor.

Ya con los datos guardados en una variable se aplica un “`ref.update(datos)`” para enviar a Firebase. Conectados los dispositivos por Ethernet ya con las IP estáticas, se ejecuta el programa “Conexión UR3-DB” en la computadora, se coloca la IP asignada al brazo UR3 en el campo de datos vacío, como se observa en la *figura 5.42* se presiona “cargar IP”.



Figura 5.42: Conexión con el robot UR3

Esto crea el vínculo de intercambio de datos entre el UR3 y la base de datos de Firebase,

5.1.10 Implementación HoloLens 2

Realizadas las pruebas entre Unity y la base de datos de Firebase, comprobando su intercambio de datos y movimientos correctos, se implementa la plataforma en los lentes HoloLens 2, es necesario tener la configuración previa de módulos en Build Settings o configuración de compilación como se observa en la siguiente *figura 5.43*, con esto permite cargar todos los módulos y sus interconexiones.

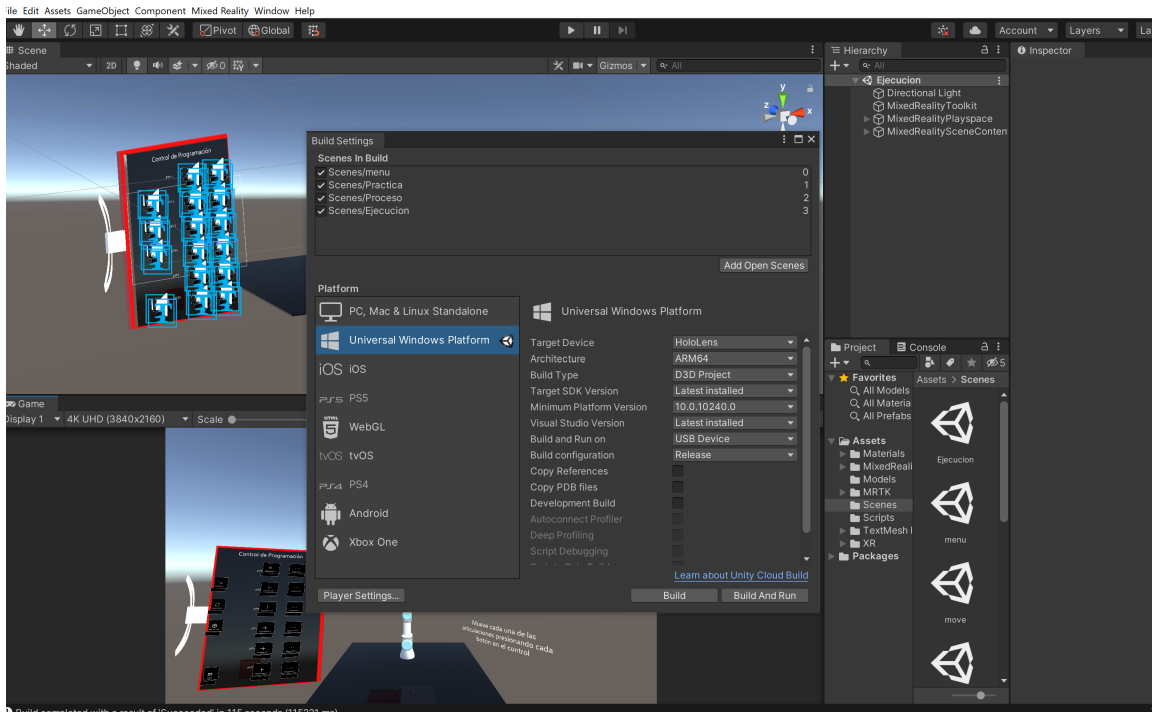


Figura 5.43: Configuración de compilación

En la configuración de compilación activar HoloLens en el objetivo del proyecto, así como la arquitectura ARM64 que ocupan los HoloLens 2, cuando se configure el proyecto en Visual Studio para implementar en los HoloLens 2 se configurarán los mismos parámetros. Ya lista la configuración, se crea una carpeta nueva para almacenar la programación de construcción del proyecto, presionar el botón “build” para construir el archivo que ejecutará la programación en los lentes y guardar en la carpeta creada.

Al ingresar a la carpeta donde se compiló el proyecto de Unity, aparecen los siguientes archivos como se observa en la *figura 5.44*, se ejecuta el archivo con la terminación “sln” que abrirá automáticamente sobre Visual Studio. Conectar los HoloLens 2 con el cable de USB-c a la computadora para iniciar la comunicación.

Una vez abierto Visual Studio se necesita configurar la plataforma de compilación en los HoloLens 2 como podemos observar en la siguiente *figura 5.45*, relase, ARM64, Dispositivo. Iniciar la carga de los archivos y ejecutable en las gafas en compilar. Es importante observar que la carga fue exitosa en el dispositivo como observamos en la *figura 5.46*, de lo contrario intentar nuevamente.

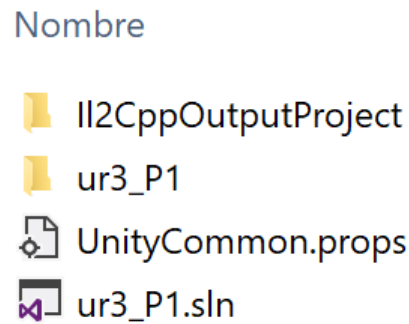


Figura 5.44: Archivos ejecutables

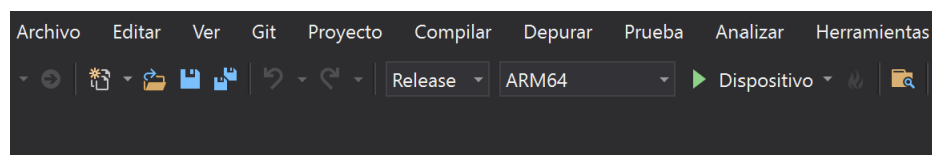


Figura 5.45: Configuración Visual Studio

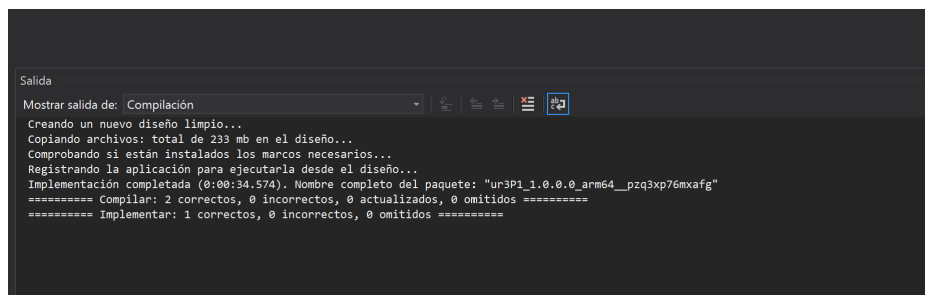


Figura 5.46: Carga correcta

Terminada la carga, desconectar el cable de la computadora, encender los HoloLens 2, abrir el menú en las gafas, abrir la opción de todas las aplicaciones y buscar el proyecto que se cargó para realizar pruebas de funcionalidad.

5.1.11 Cuestionario de evaluación

Se generó un cuestionario de evaluación, para medir parámetros de pruebas específicas en usuarios. Esto permite recabar información importante permitiendo mejorar la experiencia del usuario.

Las preguntas se seleccionaron basado en la experiencia derivada de las pruebas al crear la plataforma. El cuestionario consta de 5 preguntas aplicadas a los usuarios de las gafas HoloLens 2, que determinan su funcionalidad, estas preguntas son:

- Tiempo de adaptación en minutos.

- Facilidad de uso.
- Comodidad del dispositivo.
- Posición de objetos.

Un parámetro evaluado fue el tiempo que los usuarios mantienen las gafas puestas, sometiendo el uso de máximo 15 minutos por usuario.

La retroalimentación de la experiencia en el usuario, permite tener parámetros de uso e implementación y mejorar las funciones de la plataforma.

Capítulo 6

Resultados

6.1 Introducción

En esta sección se abordan los resultados obtenidos de esta investigación realizada en el desarrollo e implementación de la plataforma en las gafas HoloLens 2, a un brazo robótico UR3 teleoperado que se localiza en los laboratorios de la Universidad Iberoamericana, su implementación en el modelo de Industria 4.0, sus ventajas, aplicaciones y repercusiones tanto para un operador como las empresas o instituciones educativas que lo implementen.

Hoy en día la mayoría de las instalaciones tanto en la industria como en diversas instituciones educativas se encuentran interconectadas por sistemas en la nube, los resultados obtenidos muestran un panorama muy favorable para su implementación tanto en la industria como en la educación, las pruebas se realizaron en un ambiente controlado en las instalaciones de la Universidad Iberoamericana.

La aplicación se desarrolló en tres módulos, como se observa en el documento, con la finalidad de abarcar la capacitación y operación a los usuarios, facilitando el proceso de familiarización con los brazos colaborativos UR3.

La primera prueba realizada fue dirigida a observar el flujo y correcto funcionamiento de la plataforma implementada en los lentes HoloLens 2, en esta prueba se experimenta como el usuario opera el sistema, la comodidad, posición espacial de los objetos, tiempo de adaptación y el uso en general.

Sé probó el menú y cada uno de los módulos, que su funcionalidad sea correcta y como están diseñadas, como podemos observar en las siguientes *figuras 6.1, 6.2, 6.3, 6.4*.

Las pruebas de velocidad de intercambio de datos entre los HoloLens 2 y la base de datos Firebase, se realizaron observando el tiempo que tardaba en ejecutar un cambio la base de datos y el modelo, para esta prueba casi es imperceptible, esto debido a que las pruebas se realizaron todo el tiempo en las instalaciones de la Universidad Iberoamericana, la cual cuenta con una red de aproximadamente 900 Mbps, esta permite obtener una buena conexión entre los lentes y la base de datos. Pruebas realizadas con una red comercial de 100 Mbps

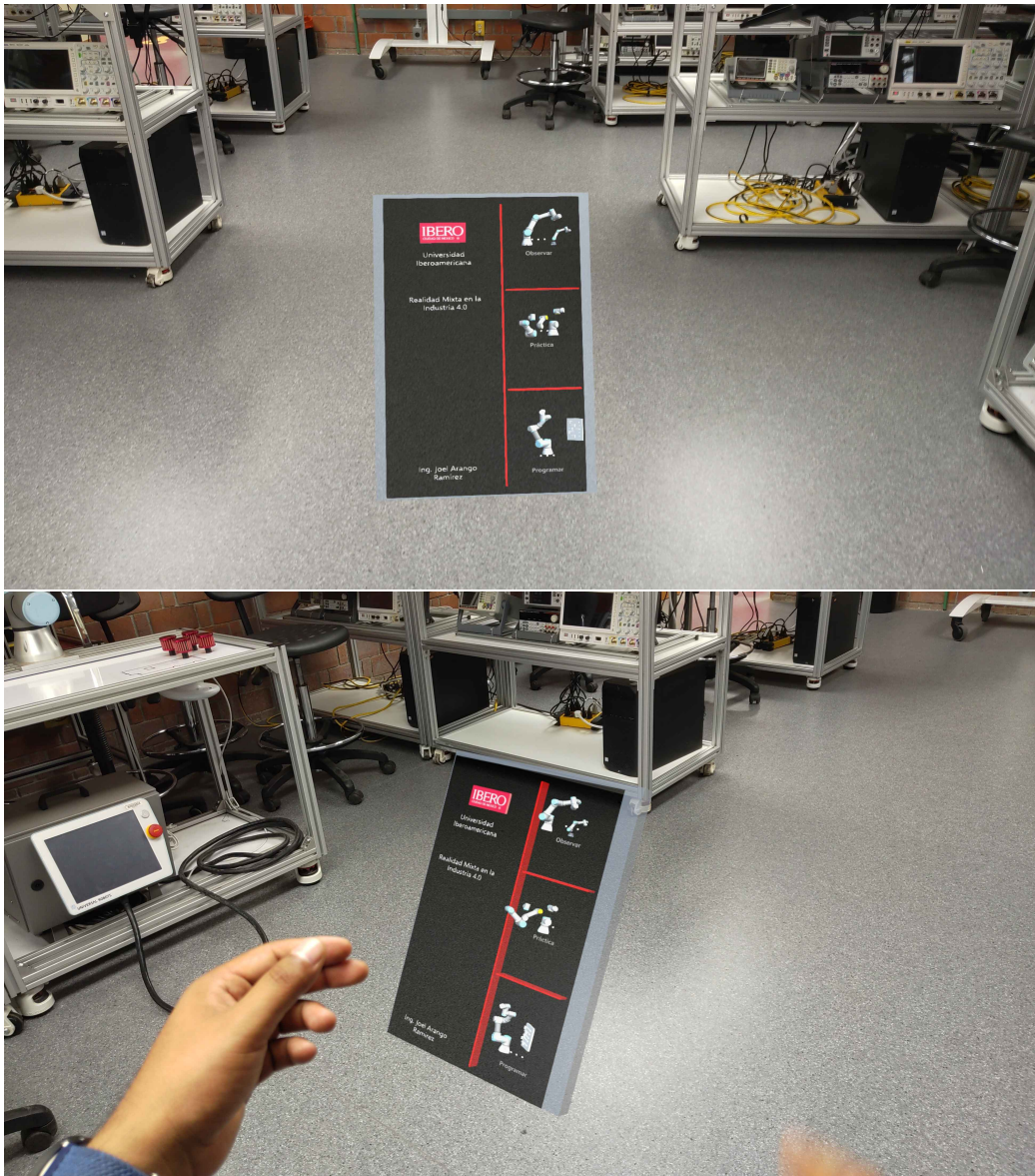


Figura 6.1: Menú

también mostraron un excelente rendimiento casi imperceptible en reiteradas ocasiones.

Con respecto a la medición de latencia de datos del UR3 con la base de datos Firebase se realizaron mediciones por código, se realizaron variaciones de tiempo de espera de envío o recepción de datos. Los parámetros de latencia de datos obtenidos son los observados en la *tabla 6.1*.

Es importante resaltar que las pruebas realizadas se implementaron únicamente en los laboratorios de la Universidad Iberoamericana, la cual cuenta con un sistema elevado de bloqueo de seguridad con respecto al intercambio de datos, lo cual afecta de manera significativa la operación del flujo de datos con la base de datos en tiempo real.

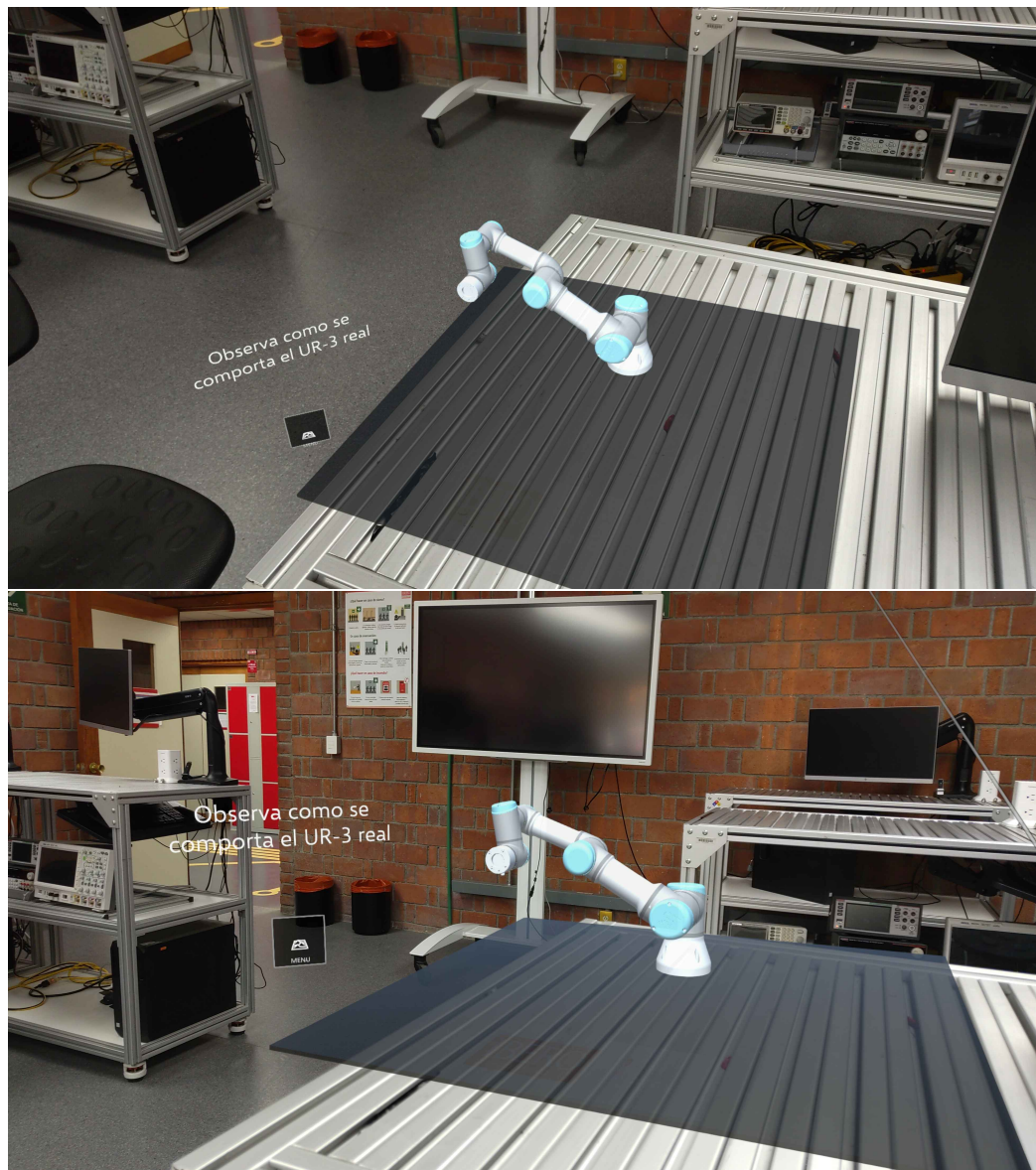


Figura 6.2: Módulo Observar

Tabla 6.1: Latencia UR3 a Firebase

Mediciones	Parámetro en segundos	Estabilidad
30	1	Inestable
30	2	Inestable
30	3	Intermedio
30	4	Estable
30	5	Estable

La variación de parámetros en el código dictamina cada cuantos segundos se envían o solicitan los datos a la nube, se observó que la cantidad de intercambio de datos a Firebase llenaba el bus de datos y detenía o atascaba la aplicación, esto podría deberse a los bloqueos en la red de la Universidad al detectar gran flujo de información.



Figura 6.3: Módulo Practicar

Es importante resaltar que el intercambio de datos entre el UR3 y Firebase es constante a lo cual el riesgo de llenado de bus de datos está latente, contrario con la conexión entre los HoloLens 2 y Firebase esta se encuentra espaciado en tiempo de intercambio de los datos cada que existe un cambio de información en la nube.

La prueba va dirigida a la comunicación entre ambos dispositivos de la plataforma. En esta apartado es posible apreciar que se logra la manipulación del brazo real UR3 desde los HoloLens 2 en cada una de sus articulaciones *figuras 6.5*, se puede observar que la relación de datos entre el brazo real y el gemelo digital logran colocarse en la misma posición tanto en lectura de datos, como el envío de comandos de cada articulación al modelo real.

La siguiente prueba se realizó analizando las sensaciones de 20 usuarios que ocuparon los HoloLens 2 con la plataforma ya instalada, cada usuario los operó por un tiempo de 15 minutos, la prueba fue la manipulación del modelo digital espacialmente y la programación



Figura 6.4: Módulo Programar



Figura 6.5: Programación con conexión a base de datos

de movimiento de control de botones interactuando con la movilidad de cada una de las articulaciones, esta prueba se realizó sin estar conectado a la base de datos de la nube, los datos recabados fueron los que se observan en la *tabla 6.2*, los usuarios fueron alumnos de diferentes carreras (Ingenieras, Diseño, Arquitectura, Comunicación).

En esta prueba se buscó conocer el nivel de adaptabilidad a la plataforma por parte del usuario así como a los dispositivos de realidad mixta en este caso los HoloLens 2, los datos recabados en los usuarios muestran que los lentes son dispositivos muy cómodos para una operación de tiempos prolongados, todas las calificaciones fueron positivas en este apartado.

En la mayoría de los casos no rebasaba los 5 minutos de adaptabilidad al entorno y manipulación de objetos, en 3 casos rebasaron este tiempo de adaptación, 1 usuario no logró adaptarse al sistema de manipulación espacial de objetos.

Para la mayoría de usuarios, la manipulación de objetos fue complicada; sin embargo, comentaron que con mayor tiempo de uso se convertiría en una actividad sencilla, esto a que en la mayoría era el primer acercamiento a estos dispositivos.

Solo una persona no le fue posible la adaptación a la aplicación, le parecía muy complejo como tomar los objetos y solicitó reducir el tiempo de uso.

Tabla 6.2: Pruebas de Usuarios

Usuarios	Tiempo de adaptación (minutos)	Facilidad de uso	Comodidad dispositivo	Tiempo de uso (minutos)	Posición de objetos
2	Menos de 5	Fácil	Ligero	15	Buena
9	Menos de 5	Intermedio	Ligero	15	Buena
3	Menos de 5	Intermedio	Ligero	15	Muy alta
2	Menos de 5	Complicado	Ligero	15	Buena
3	Más de 5 - menos de 8	Difícil	Ligero	15	Lejano
1	Más de 5	Muy difícil	Ligero	10	Buena

Tres usuarios comentaron, que los objetos se encontraban espacialmente alejados de ellos y que tenían que moverse frontalmente para interactuar con los objetos, pero que no causaba una complicación mayor. Un usuario comentó que las plataformas de los objetos, así como el menú, se localizaban muy alto espacialmente.

Los datos generales recabados de la plataforma tiene un alto nivel de oportunidad de implementación, la mayoría de usuarios expresaron que al ser una tecnología muy nueva es importante un acercamiento más directo y constante con su utilización, así como mayor tiempo de adaptación, pero en general lo ocuparían. Ver *figura 6.6*.

La última prueba realizada fue la operación de usuarios familiarizados con robótica y utilización y programación del brazo robótico UR3, esta prueba arrojó datos más precisos así como contundentes de la utilización y desempeño de la plataforma, la posibilidad de su implementación en diversos campos. Los comentarios obtenidos de esta prueba orientan a que apoyarían fuertemente como una introducción previa al uso del dispositivo real, ya que su función es muy intuitiva y didáctica, esto podría prevenir accidentes o mal uso de los brazos robóticos *figura 6.7*.

Fue posible presentar los avances de la investigación e implementación de la plataforma en diferentes eventos de la Universidad Iberoamericana, donde diferentes usuarios tuvieron la oportunidad de probar el sistema implementado en las HoloLens 2, como se observa en las siguientes *imágenes 6.8*.

6.1.1 Contribución a la Industria

En la investigación de las nuevas tecnologías emergentes, así como la implementación cada vez más fuerte de la interconexión de dispositivos hacia plataformas alojadas en la nube, es



Figura 6.6: Experiencia de uso de plataforma

posible vislumbrar que cada día más dispositivos de la industria se encontrarán conectados. Sucesos como la pandemia de 2019, donde es indispensable no detener los procesos industriales sin descuidar la seguridad de los operarios, esta herramienta apoya de forma muy práctica a cubrir dichas necesidades de mantener los procesos de la industria activos sin detener la economía y manteniendo a los operarios seguros, permitiendo trabajar de forma remota la operación de los diferentes dispositivos conectados en las empresas, esto reduce costos a las empresas y brinda seguridad al operario.

La contribución de esta plataforma de realidad mixta creada, plantea una nueva forma de como se puede innovar en la industria en un corto plazo, donde se busca mayor eficiencia a bajos costos y mejores prestaciones para los operadores, cubriendo el avance tecnológico exponencial en estos días.

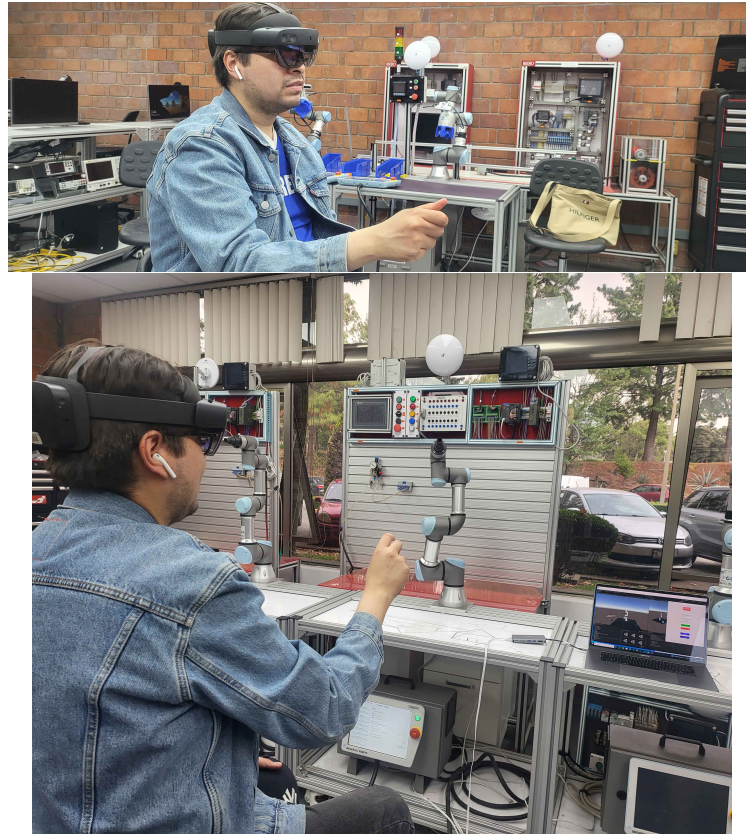


Figura 6.7: Pruebas en Laboratorio

Algunos ejemplos:

- Control a distancia de procesos de bandas transportadoras, observando desde los lentes el funcionamiento en tiempo real y los datos de cada módulo.
- Controlar vehículos autónomos, creando rutas y eficiencia en sus movimientos por la planta gracias a la retroalimentación de datos en tiempo real, conociendo la posición de cada uno y la cadena de flujo de todos los vehículos con la posibilidad de modificar sus rutas.
- Control completo de cada elemento de una planta, observando el comportamiento de cada elemento o modificarlo si es el caso, permitiendo corregir o conocer perturbaciones en la producción.

6.1.2 Contribución a la Educación

Los modelos educativos cada día plantean más la conjunción entre la tecnología y el aprendizaje, esta herramienta proporciona la facilidad de enseñar a un alumno sin exponerlo a peligros, minimizando en gran parte la mala operación de los dispositivos al permitir una práctica previa con estos dispositivos, con esto los estudiantes tienen una idea más clara del comportamiento de los modelos interconectados antes de operar los modelos reales.



Figura 6.8: Universidad Iberoamericana

Esta investigación demostró que no solo en la operación de dispositivos robóticos de la industria es útil, en la mayoría de carreras es posible la creación e implementación de aplicaciones de realidad mixta que permitirían un aprendizaje más práctico, sencillo e interactivo cambiando paradigmas de la forma como se implementa la educación hoy en día, algunos ejemplos:

- Biomédica, la posibilidad de implementar la anatomía de un cuerpo y explorarlo en un salón de clase o a distancia con dispositivos de realidad mixta, da la posibilidad que el profesor y los alumnos puedan manipular los diferentes componentes del cuerpo desde diferentes lugares, que un alumno pueda desde su hogar realizar una práctica y pedir asesoría a distancia de su profesor observando el mismo modelo y los cambios realizados

en tiempo real.

- Arquitectura, crear modelos a escala con medidas físicas similares al que se plantea construir, experimentando materiales o diseños previo a su ejecución, otorga la posibilidad de observar el comportamiento de los materiales obteniendo la información de sensores implementados a distancia previniendo fallos de las estructuras.
- Ingenierías, control de todo el ecosistema de un sistema de producción donde el alumno puede experimentar el comportamiento de cada elemento en el flujo de operación antes de implementarlo, conociendo el resultado cuando entre en operación, manipulando los componentes de forma virtual para después experimentarlo de forma real, programando desde la plataforma todas las necesidades.

Estas son solo algunas de las amplias posibilidades que se detectaron en esta investigación y su implementación en la educación.

Capítulo 7

Conclusiones

Es posible concluir que el proyecto desarrollado superó los objetivos planteados, ya que el planteamiento principal abarcaba la teleoperación del Brazo UR3 que se encuentra en los laboratorios de la Universidad Iberoamericana con unas gafas de realidad aumentada; sin embargo, fue posible ampliar la investigación y los resultados de esta investigación.

Se logró la implementación de tres módulos en las gafas HoloLens 2, que permiten no solo observar el comportamiento del UR3 sino la manipulación por parte del usuario, así como la programación del brazo a distancia. Esto permite una capacitación completa de un operador, permitiendo observar, manipular y programar el UR3 real desde su gemelo virtual.

Los retos a afrontar fueron amplios; sin embargo, nos permitió obtener mucha experiencia y entender el comportamiento de estos sistemas. Un gran reto fue que no existe mucha documentación de estas tecnologías al ser muy nuevas. El primer análisis a investigar fue encontrar una solución a la conexión tanto del brazo UR3 y la implementada en los HoloLens 2 a la nube en bases de datos en tiempo real, buscando una plataforma segura y rápida en el intercambio de datos, ya que de esta solución deriva gran parte del resultado de esta investigación.

Esto permitió observar el alcance que se puede lograr con esta tecnología que da la apertura a más dispositivos y campos de aplicación, no solo de la robótica, sino también en educación, entre otros.

Otro tema importante que se trató fue la implementación de realidad mixta y su programación en dispositivos ópticos como los lentes HoloLens 2, estos lentes son usados en la industria a lo cual no es muy usual su uso en la investigación y desarrollo, se centran más en la implementación desarrollada por la misma marca, esto presentó grandes inconvenientes debido a que la documentación es muy limitada en este ámbito y su implementación más compleja, es posible encontrar más información de desarrollo en lentes de realidad virtual; sin embargo, fue posible la creación y puesta en marcha de la plataforma en realidad mixta, esto abrió un panorama más amplio del uso de estas nuevas tecnologías, permitiendo observar que no solo es posible su uso en la industria, educación, medicina etc.

Un cambio importante que ayudó en una implementación más rápida fue ocupar la paquetería de Python de URX y no el protocolo Modbus, ya que las librerías de URX están dirigidas a la comunicación con los brazos UR3, esto permitió una interacción más directa. Al ser compatible con Python permite enlazar más fácil los datos a la base de datos en la nube, ya que se cuentan con una gran variedad de librerías que aminoran el proceso, este cambio en la programación permite hacer un código más corto y limpio.

Una vez conseguida la implementación de la plataforma, fue posible poner en marcha la aplicación en los lentes así como en el brazo UR3, probando la comunicación a distancia bidireccional, comprobando que cada articulación responde ante los cambios solicitados desde la plataforma y observar los cambios casi instantáneos en el robot UR3. Con esto podemos concluir que el proceso de teleoperación se cumplió con éxito, que era la base central del proyecto. La intercomunicación entre los dispositivos.

Como podemos observar a lo largo del documento, se puso en marcha la implementación de la aplicación, los procesos y pruebas a usuarios para conseguir una retroalimentación del sistema, a lo cual se concluye que es un sistema que puede ser implementado gracias a su interfaz que resulta sencilla, cómoda y con un tiempo rápido de adaptación, siendo funcional capacitando y evitando posibles daños tanto a usuarios y dispositivos robóticos, basado en los usuarios más del 95 por ciento de los usuarios lo utilizaría, esto permite una buena sensación de la posibilidad de implementación.

La investigación permitió la presentación del proyecto en congreso y publicación en “International Conference of Control, Dynamic Systems, and Robotics (CDSR 2024)” de Toronto, Canadá. Movilidad para el estudio e implementación de esta tecnología en “La Pontificia Universidad Javeriana” de Bogotá en Colombia.

Podemos concluir como satisfactorio el resultado de la investigación, ya que fue posible solucionar el tema principal planteado, así como añadir mejoras que no estaban previstas. Las pruebas realizadas superaron las expectativas, los usuarios precisaron que es una tecnología que apoyaría la capacitación no solo a las industrias, también en la educación, con contenido práctico en múltiples disciplinas. Genera un gran aporte a futuras investigaciones, que ayuden a desarrollar más herramientas con nuevas tecnologías emergentes.

La operabilidad de la plataforma aporta un gran avance tecnológico para futuras ampliaciones de investigación de estas tecnologías. Fue posible probar su eficiencia y aceptación por un cierto número de usuarios, cumpliendo con su función de enlazar la información, ayudando al usuario a entender una nueva forma de experiencia de programación y monitoreo, abriendo un campo muy amplio de implementaciones o mejoras de este y otros sistemas parecidos.

Capítulo 8

Recomendaciones

El sistema desarrollado podría ser mejorado implementando cinemática inversa, esto permite un sistema más robusto y completo abarcando las cualidades del brazo UR3.

Realizar pruebas en un ambiente industrial para obtener información complementaria del comportamiento con una operación rutinaria.

Añadir más datos relacionados con el UR3 en los HoloLens 2 como temperatura del robot, valores de articulaciones y efector, proceso ejecutado, paro de emergencia, etc. Permitiendo tener una información más completa al operador.

Implementar un sistema de cámaras que permita al operador observar en las gafas el gemelo virtual y el desempeño del brazo UR3 de diferentes ángulos.

Medir con una red de internet dedicada para tener control de los sistemas de seguridad de intercambio de datos, buscando obtener un sistema más fluido en su proceso.

La latencia obtenida en este proyecto no fue un tema central de estudio en el proyecto, pero se observó que es posible optimizar los tiempos de conexión e intercambio de datos.

La plataforma desarrollada demostró ser útil tanto para la industria como en la educación, se muestra conveniente desarrollar cursos de operación y capacitación en clases dirigidas al uso de los brazos UR3 en la Universidad previo al contacto del alumno con el modelo real.

Bibliografía

- [1] [Online]. Available: <https://www.cepal.org/es/comunicados/impactos-la-pandemia-sectores-productivos-mas-afectados-abarcaran-un-tercio-empleo-un>
- [2] [Online]. Available: <https://www.c3.unam.mx/noticias/noticia180.html>
- [3] J.-S. Chern and Y.-C. Mao, “Effects of immersive virtual reality puzzle solving video games on cognition, motor control, and functional behavior in people with schizophrenia spectrum disorders,” pp. 30–35, 2023.
- [4] T. Vidal, I. Navarro, A. Sánchez, F. Valls, L. Giménez, and E. Redondo, “Virtual reality for enhanced learning in artistic disciplines of degree of video games,” pp. 1–5, 2021.
- [5] H. Flatt, N. Koch, C. Röcker, A. Günter, and J. Jasperneite, “A context-aware assistance system for maintenance applications in smart factories based on augmented reality and indoor localization,” pp. 1–4, 2015.
- [6] [Online]. Available: <https://www.forbes.com.mx/museo-de-memoria-y-tolerancia-usa-realidad-aumentada-para-atraer-nuevo-publico/>
- [7] W. Matcha and D. R. A. Rambli, “Development and preliminary investigation of augmented reality experiment simulation (arex) interface,” pp. 1–5, 2011.
- [8] R. R. Calderón and R. S. Arbesú, “Augmented reality in automation,” *Procedia Computer Science*, vol. 75, pp. 123–128, 2015, 2015 International Conference Virtual and Augmented Reality in Education. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1877050915036893>
- [9] A. Ciprian Firu, A. Ion Tapîrdea, A. Ioana Feier, and G. Drăghici, “Virtual reality in the automotive field in industry 4.0,” *Materials Today: Proceedings*, vol. 45, pp. 4177–4182, 2021, 8th International Conference on Advanced Materials and Structures - AMS 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2214785320397224>
- [10] Z. Zhang, D. Weng, H. Jiang, Y. Liu, and Y. Wang, “Inverse augmented reality: A virtual agent’s perspective,” pp. 154–157, 2018.
- [11] F.-K. Chiang, X. Shang, and L. Qiao, “Augmented reality in vocational training: A systematic review of research and applications,” *Computers in Human Behavior*, vol. 129, p. 107125, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0747563221004489>

-
- [12] [Online]. Available: <https://sobreverso.com/metaverso/las-gafas-del-pigmalion-realidad-virtual/>
- [13] [Online]. Available: <https://www.innovae.com/la-tecnologia-de-realidad-virtual/>
- [14] [Online]. Available: <https://expansion.mx/tecnologia/2022/02/17/como-llegamos-hasta-el-metaverso>
- [15] [Online]. Available: <https://xperimentacultura.com/historia-de-la-realidad-virtual/>
- [16] [Online]. Available: <https://www.fotoconmac.com/wheatstone-sir-charles/>
- [17] [Online]. Available: <https://en-clase.ideal.es/2024/05/08/elemento-de-la-semana-del-museo-de-ciencias-del-ies-padre-suarez-216-estereoscopio-de-mano/>
- [18] [Online]. Available: https://as.com/meristation/2018/01/10/reportajes/1515567480_172151.html
- [19] [Online]. Available: <https://blog.sinergis.com.mx/la-realidad-aumentada-llego-para-quequedarse#:~:text=La%20primera%20vez%20que%20se,las%20personas%20que%20ten%C3%ADa%20delante.>
- [20] [Online]. Available: <https://digitalartarchive.siggraph.org/artwork/myron-w-krueger-i-met-a-morph/>
- [21] [Online]. Available: <https://granasoft.es/realidad-virtual/que-es-la-realidad-virtual/>
- [22] [Online]. Available: <https://learn.microsoft.com/es-es/windows/mixed-reality/discover/mixed-reality>
- [23] [Online]. Available: <https://roboticapuno.blogspot.com/2013/01/clasificacion-de-los-robots.html>
- [24] [Online]. Available: <https://www.telefonica.com/es/sala-comunicacion/blog/tipos-de-robots-clasificacion-aplicaciones-y-ejemplos/>
- [25] [Online]. Available: <https://dle.rae.es/robot>
- [26] M. B. JR Figueroa, WM Montalvo, “Cinemática y dinámica de robots móviles con ruedas,” Marzo 2023.
- [27] [Online]. Available: <https://continuemos estudiando.abc.gob.ar/wp-content/uploads/2023/02/cuadernillo-de-actividades-6ta-entrega-morfologia-basica-de-un-robot-industrial-educacion-tecnologica.pdf>
- [28] [Online]. Available: <https://www.tecnologia-tecnica.com.ar/pdfrobotica/PDFmorfologiaderobotindustrial.pdf>
- [29] [Online]. Available: https://www.researchgate.net/figure/Figura-216-Relacion-entre-cinematica-directa-e-inversa_fig15_287995531
- [30] [Online]. Available: https://en.wikipedia.org/wiki/Denavit%20%80%93Hartenberg_parameters
-

- [31] [Online]. Available: <https://www.kramirez.net/Robotica/Material/Presentaciones/CinematicaDirectaRobot.pdf>
- [32] [Online]. Available: <https://www.universal-robots.com/es/productos/robot-ur3/>
- [33] [Online]. Available: <https://blog.inmersys.com/empresas-que-est%C3%A1n-capacitando-con-vr>
- [34] M. Kozek, “Transfer learning algorithm in image analysis with augmented reality headset for industry 4.0 technology,” pp. 1–5, 2020.
- [35] S. Sureshkumar, C. P. Agash, S. Ramya, R. Kaviyaraj, and S. Elanchezhian, “Augmented reality with internet of things,” pp. 1426–1430, 2021.
- [36] H. Subakti and J.-R. Jiang, “Indoor augmented reality using deep learning for industry 4.0 smart factories,” vol. 02, pp. 63–68, 2018.
- [37] D. Mourtzis, J. Angelopoulos, and N. Panopoulos, “Challenges and opportunities for integrating augmented reality and computational fluid dynamics modeling under the framework of industry 4.0,” *Procedia CIRP*, vol. 106, pp. 215–220, 2022, 9th CIRP Conference on Assembly Technology and Systems. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2212827122001822>
- [38] [Online]. Available: <https://www.oracle.com/mx/internet-of-things/what-is-iot/>
- [39] J. Kovar, K. Mouralova, F. Ksica, J. Kroupa, O. Andrs, and Z. Hadas, “Virtual reality in context of industry 4.0 proposed projects at brno university of technology,” pp. 1–7, 2016.
- [40] J. K. Bologna, C. A. Garcia, A. Ortiz, P. X. Ayala, and M. V. Garcia, “An augmented reality platform for training in the industrial context,” *IFAC-PapersOnLine*, vol. 53, no. 3, pp. 197–202, 2020, 4th IFAC Workshop on Advanced Maintenance Engineering, Services and Technologies - AMEST 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2405896320301786>
- [41] [Online]. Available: <https://alegerglobal.com/es/shop-es/gafas-inteligentes/google-glass-2-2/>
- [42] [Online]. Available: <https://www.lib.ncsu.edu/devices/hololens-2>
- [43] [Online]. Available: <https://www.eleconomista.com.mx/tecnologia/Vision-Pro-de-Apple-tardara-tiempo-para-que-el-mercado-lo-adopte-consideran-analistas-20230600.html>
- [44] [Online]. Available: <https://www.pcguide.com/vr/meta-quest-pro-v-3/>
- [45] [Online]. Available: <https://www.muycomputer.com/2016/02/22/moverio-bt-300-gafas-inteligentes/>
- [46] [Online]. Available: <https://alegerglobal.com/es/shop-es/gafas-inteligentes/vuzix-m300xl-2/>

- [47] [Online]. Available: <https://modbus.org/>
- [48] [Online]. Available: <https://blog.ars-electronica.com.ar/que-es-modbus-funcionamiento>
- [49] [Online]. Available: <https://www.cursosaula21.com/modbus-que-es-y-como-funciona/>
- [50] [Online]. Available: <https://datos.gob.es/es/blog/11-librerias-para-crear-visualizaciones-de-datos>
- [51] [Online]. Available: <https://www.universal-robots.com/developer/insights/client-libraries-for-external-monitoring-and-control/>
- [52] [Online]. Available: <https://www.goconqr.com/apunte/14997748/protocolo-tcp-ip-y-puertos-logicos>
- [53] [Online]. Available: https://www.revista.unam.mx/vol.5/num8/art51/sep_art51.pdf
- [54] [Online]. Available: <https://docs.unity3d.com/Manual/index.html>
- [55] [Online]. Available: https://docs.blender.org/manual/es/dev/getting_started/about/introduction.html
- [56] [Online]. Available: <https://blogs.microsoft.com/blog/2012/08/23/microsoft-unveils-a-new-look/>
- [57] [Online]. Available: <https://www.microsoft.com/es-mx/hololens>
- [58] [Online]. Available: <https://www.microsoft.com/es-es/hololens/hardware#documentar-experiencias>
- [59] [Online]. Available: <https://www.universal-robots.com/articles/ur/application-installation/dh-parameters-for-calculations-of-kinematics-and-dynamics/>
- [60] [Online]. Available: https://en.wikipedia.org/wiki/Mixed_Reality_Toolkit
- [61] [Online]. Available: <https://firebase.google.com/?hl=es>
- [62] [Online]. Available: <https://firebase.google.com/brand-guidelines?hl=es-419>