

Gemelo digital en RA de Planta de Automatización de la Universidad Santo Tomás

Daniel Sneider Espinel García

Trabajo de grado para optar el título de Ingeniero Mecatrónico

Director

David Leonardo Martínez Herrera

Maestría en Ingeniería

Universidad Santo Tomás, Bucaramanga

División de Ingenierías y Arquitecturas

Ingeniería Mecatrónica

2023

Dedicatoria

Este proyecto va dedicado con un gran y profundo agradecimiento a mis padres, cuyo apoyo incondicional, confianza y amor desde el primer día han sido el pilar fundamental en mi camino. Agradezco sinceramente sus invaluable consejos y los valores que han guiado cada paso de esta etapa. A su vez, dedico este logro a mis amigos cercanos, comenzando por Silvia, quien ha sido una parte esencial en todo mi proceso educativo, brindándome un compañerismo inquebrantable, lo cual aprecio enormemente y quiero hacer saber que valoro mucho tu amistad has dejado huella en mí, haber compartido contigo ha sido esencial en mi crecimiento educativo, y que personas como tu son importantes para mí. Además, mi reconocimiento y gratitud se extiende a personas como Andrea, Sergio, Germán, Kevin, Elizabeth, Jacqui, Laura, Felipe, Diego, Duvan y a Gabriela, quienes han sido fuentes de apoyo incondicional a lo largo de este trayecto.

Agradecimientos

Primeramente, agradezco a la universidad Santo Tomas por brindarme la educación necesaria y haberme formado en todos los ámbitos tanto personal como académicamente, así mismo, agradezco al ingeniero David Leonardo Martínez Herrera por guiarme en este proceso, ayudarme en lo que he necesitado y en la corrección de este material. También agradezco al ingeniero Hernán Hernández por la vocación y por el tiempo prestado por haber estado disponible y dispuesto a ayudarme en todo momento. Igualmente, al ingeniero Emerson Olaya por haber sido parte de este proceso enriquecedor académicamente. Solo queda decir gracias ya que de una u otra manera fueron parte de esto.

Contenido

Introducción	14
1. Gemelo digital en RA de Planta de Automatización de la Universidad Santo Tomás	15
1.1 Planteamiento del problema	15
1.2 Objetivos	16
1.2.1 Objetivo general	16
1.2.2 Objetivos específicos	16
1.3 Aportes del trabajo	17
1.4 Organización del Documento	18
1.4.1 Introducción	18
1.4.2 Marco conceptual	18
1.4.3 Metodología	18
1.4.4 Desarrollo	18
1.4.5 Recomendaciones	18
1.4.6 Conclusiones	19
2. Marco de Referencia	19
2.1 Organización del Documento	19
2.1.1 Introducción	19
2.1.2 Automatización Industrial	20
2.1.3 Gemelo Digital	21
2.1.4 WebSocket	23
2.1.5 Node-RED	23
2.1.6 Node.JS	24
2.1.7 Ubuntu	26

2.1.8 ARcore.....	28
2.2 Marco Conceptual	29
2.2.1 Realidad Aumentada.....	29
2.2.2 Automatización Industrial en Ingeniería Mecatrónica	31
2.2.3 Gemelo Digital	31
2.3 Marco Legal	31
2.4 Marco Tecnológico y Científico	32
2.5 Condiciones Iniciales	34
3. Diseño Metodológico.....	34
3.1 Criterios de Diseño.....	35
3.1.1 Gemelo Digital	35
3.1.2 Interactividad intuitiva.....	36
3.1.3 Comunicación entre el gemelo digital y el plc	37
3.1 Descripción de etapas y tareas	38
3.3. Descripción de los productos finales.....	42
3.4. Conclusión del Capitulo.....	43
4.Desarrollo.....	44
4.1. Diseño.....	45
4.1.1 Etapa 1	45
4.1.2 Etapa 2	45
4.1.3 Etapa 3	51
4.1.4 Etapa 4	61
4.1.5 Etapa 5	71

4.2. Simulaciones-pruebas- caracterización	74
4.2.1 Pruebas	74
4.2.2 Modificación del proyecto de Unity	79
4.2.3 Pruebas de estabilidad.	80
4.3. Resultados	82
4.3.1 Aplicativo móvil.	82
4.3.2 Servidor web server.	83
4.3.2 Gemelo digital.....	83
4.3.2. Flujo de datos en Node-RED	83
4.4. Discusión de resultados	84
5. Conclusiones y Trabajos Futuros.....	85
5.1. Conclusiones	85
5.2. Trabajos futuros.....	86
6. Recomendaciones	87
Referencias.....	88

Lista de tablas

Tabla 1. <i>Comparación entre software de diseño</i>	35
Tabla 2. <i>Características de motor gráfico Unity</i>	36
Tabla 3. <i>Resultados esperados</i>	82

Lista de figuras

Figura 1. <i>Proceso de Realidad Aumentada</i>	20
Figura 2. <i>Logo de Node-Red</i>	23
Figura 3. <i>logo de node.js</i>	24
Figura 4. <i>Entorno de ejecución</i>	25
Figura 5. <i>Logotipo de Ubuntu</i>	26
Figura 6. <i>Logotipo de ARcore</i>	29
Figura 7. <i>Estructura de desglose de trabajo EDT</i>	38
Figura 8. <i>Electroválvula FPD270</i>	45
Figura 9. <i>Sensor capacitivo</i>	46
Figura 10. <i>Sensor de flujo de efecto Hall</i>	46
Figura 11. <i>Valvula selenoide</i>	47
Figura 12. <i>codo de pvc</i>	47
Figura 13. <i>Base de planta de automatización</i>	47
Figura 14. <i>Tanque de plástico</i>	48
Figura 15. <i>Selección de área de la pieza según sus vértices.</i>	48
Figura 16. <i>Asignación de zona seleccionada al material.</i>	49
Figura 17. <i>Asignación de zona seleccionada al material.</i>	50
Figura 18. <i>Asignación de zona seleccionada al material.</i>	50
Figura 19. <i>Objetos configurables</i>	51
Figura 20. <i>Selección de versión en Unity.</i>	52
Figura 21. <i>Entorno inicial de Unity.</i>	52
Figura 22. <i>Opción para adición de componente</i>	53

Figura 23. <i>Posicionamiento de objetos referenciados a un mapa de puntos</i>	53
Figura 24. <i>Integración de piezas 3d en Unity</i>	54
Figura 25. <i>Script con objetos de la escena</i>	55
Figura 26. <i>Árbol de objetos de la planta</i>	55
Figura 27. <i>Visualización de planta de automatización.</i>	56
Figura 28. <i>Visualización lateral de planta digital</i>	56
Figura 29. <i>Visualización de datos.</i>	57
Figura 30. <i>Creación de memoria global</i>	57
Figura 31. <i>Colliders en Unity</i>	58
Figura 32. <i>Pruebas</i>	59
Figura 33. <i>Posicionamiento de la planta</i>	59
Figura 34. <i>Interfaz inicial</i>	60
Figura 35. <i>Implementación de método</i>	61
Figura 36. <i>Proceso de transmisión de información</i>	61
Figura 37. <i>Comando de instalación del “WS”</i>	62
Figura 38. <i>Creación del servidor en Node.js</i>	62
Figura 39. <i>forEach</i>	62
Figura 40. <i>Ejecución del código y conexión del cliente</i>	63
Figura 41. <i>Configuración de la máquina virtual</i>	63
Figura 42. <i>Dirección IP pública.</i>	64
Figura 43. <i>Protocolo SSH</i>	64
Figura 44. <i>UltraVNC.</i>	65
Figura 45. <i>Máquina virtual ejecutando el websocket.</i>	66

GEMELO DIGITAL EN RA DE PLANTA DE AUTOMATIZACIÓN	10
Figura 46. <i>Menú de NuGet en Unity</i>	66
Figura 47. <i>Biblioteca de NuGet.</i>	67
Figura 48. <i>Librería WebSocketSharp</i>	67
Figura 49. <i>Estructura base del cliente en Unity.</i>	67
Figura 50. <i>Prueba de la conexión.</i>	68
Figura 51. <i>Instalación de librería node-red-contrib-s7</i>	68
Figura 52. <i>Nodos s7</i>	68
Figura 53. <i>Configuración nodo WebSocket Node-Red.</i>	69
Figura 54. <i>Configuración de nodo s7</i>	69
Figura 55. <i>Tabla de variables</i>	70
Figura 56. <i>Nodos de Node-RED</i>	71
Figura 57. <i>Nodos de Node-RED</i>	71
Figura 58. <i>Creación de base de datos</i>	72
Figura 59. <i>Tabla de variables</i>	73
Figura 60. <i>Pruebas del PLC</i>	73
Figura 61. <i>Tabla de variables</i>	74
Figura 62. <i>Uso del comando ws.Send</i>	74
Figura 63. <i>Tabla de variables</i>	75
Figura 64. <i>Proceso de Filtrado en Nodo de Función.</i>	76
Figura 65. <i>Nodo básico</i>	76
Figura 66. <i>Envío de información mediante el websocket</i>	76
Figura 67. <i>Código para acceso a datos compartidos entre scripts.</i>	77
Figura 68. <i>Tablero de conexiones del PLC y la planta piloto.</i>	77

Figura 69. <i>Configuración de variables analógicas.</i>	78
Figura 70. <i>Ejemplo de la normalización de datos</i>	79
Figura 71. <i>Orden de las escenas para configurar el proyecto.</i>	79
Figura 72. <i>Probando la aplicación en San gil, Santander.</i>	80
Figura 73. <i>Tiempo de envío y tiempo de llegada en milisegundos.</i>	81

Resumen

La pandemia-covid19, generó la necesidad de buscar nuevas alternativas que faciliten los procesos de enseñanzas, sin dejar a un lado la calidad de la educación, por lo tanto, es esencial plantear alternativas que implementen tecnologías innovadoras que favorezcan tanto a los estudiantes como a los profesores a la hora de impartir clases de forma remota. Adquirir conocimientos fundamentales y habilidades en el manejo de sistemas de control de plantas en un entorno remoto es clave para el desarrollo del futuro profesional.

Una de las tecnologías de vanguardia es la realidad aumentada, por lo tanto, la propuesta de este proyecto es la digitalización de la planta de la universidad Santo Tomás mediante un gemelo digital en 3d. Esto incluye la construcción del gemelo digital, la comunicación con un controlador lógico programable (PLC) y el desarrollo de una aplicación de realidad aumentada. Con esto se quiere garantizar una transición efectiva de un modelo de educación más avanzado comparado con la actualidad. En este contexto, la aplicación de tecnologías de la actualidad, como la realidad aumentada, se ha vuelto imprescindible para enriquecer la experiencia de aprendizaje tanto para estudiantes como para profesores.

Palabras clave: Gemelo digital, Realidad aumentada, Automatización, PLC, Mecatrónica.

Abstract

The pandemic generated the need to look for new alternatives that facilitate the teaching processes, without leaving aside the quality of education, therefore, it is essential to propose alternatives that implement innovative technologies that favor both students and teachers when teaching classes remotely. Acquiring fundamental knowledge and skills in the management of plant control systems in a remote environment is key to the development of the future professional. One of the cutting-edge technologies is augmented reality, therefore, the proposal of this project is the digitization of the plant of St. Thomas Aquinas University through a digital twin in 3D. This includes the construction of the digital twin, the communication with a programmable logic controller (PLC) and the development of an augmented reality application. This is to ensure an effective transition to a more advanced education model compared to the current one. In this context, the application of current technologies, such as augmented reality, has become essential to enrich the learning experience for both students and teachers.

Keywords: Digital twins, Augmented reality, PLC, Automation, Mechatronics.

Introducción

Después de la pandemia, surgió la necesidad de utilizar al máximo la tecnología de tal manera que, los procesos de enseñanza no resultaran afectados ante la falta de presencialidad, por ende, implementar técnicas nuevas que faciliten el aprendizaje se ha convertido en algo esencial, basándose en pilares de tecnología avanzada que son fundamentales para el progreso en todos los ámbitos. En este contexto, el presente anteproyecto surge como una respuesta a la necesidad de tener una educación de calidad de forma remota a través de soluciones como la realidad aumentada o inmersión

En este nuevo paradigma educativo, se hace evidente que la innovación tecnológica se rige como el motor que impulsa la evolución de los métodos de enseñanza y aprendizaje. La presente propuesta se nutre de esta realidad, buscando no solo arreglar las dificultades que surgen con la educación remota, sino también transformarla en una experiencia enriquecedora y efectiva.

A lo largo de las siguientes secciones, se explorarán en detalle los fundamentos teóricos y metodológicos de la propuesta, así como los beneficios tangibles que se esperan obtener al implementar estas soluciones. Con el fin de garantizar una transición fluida hacia un modelo educativo mejorado y adaptado a las necesidades contemporánea de la época

En este informe, se desglosará la metodología que se seguirá para la realización del proyecto, incluyendo la investigación, el diseño, la implementación y la evaluación de la solución propuesta. Además, se destacará la importancia de la colaboración interdisciplinaria y el uso de herramientas tecnológicas de vanguardia para lograr resultados óptimos.

1. Gemelo digital en RA de Planta de Automatización de la Universidad Santo Tomás

1.1 Planteamiento del problema

En la pandemia del covid-19, se presentaron diferentes actores que buscaron soluciones que permitieran continuar desarrollando las actividades cotidianas de forma remota. Países como Colombia, donde estuvieron en una situación que puso en incertidumbre a gran parte del sistema educativo, esto debido a que limitadas entidades contaban con plataformas que pudieran garantizar la calidad del servicio de la educación. En consecuencia de los acontecimientos sociales se llevó a que las instituciones educativas buscaran alternativas para remplazar la experiencia que se vivía realizando una práctica, según estima UNESCO la pandemia afectó en el 2020 a más de 220 millones de estudiantes de nivel superior alrededor del mundo lo que conllevó a buscar soluciones como la propuesta por el TEC DE MONTERREY el cual implementó modelos de laboratorios en línea donde los estudiantes podían realizar prácticas observando en directo los resultados de sus experimentos de manera física.

Debido a que la pandemia reafirmó los problemas presentes al momento de adquirir conocimientos fundamentales y la deficiencia de habilidades en el manejo de los sistemas de control de plantas (PLC) en la práctica y al aplicarlo en un entorno laboral. Este proyecto busca implementar la realidad aumentada en áreas de la ingeniería como la automatización industrial llevando a desarrollar prácticas de forma interactivas del mundo físico con información adicional virtual suministrada por un ordenador, superpuesta al mundo real, facilitando así la transición de los conceptos teóricos hasta la aplicación del conocimiento. Este entorno inmersivo otorga al estudiante tener percepciones detalladas de los conceptos abstractos de forma accesible. La realización del proyecto permitiría que los estudiantes que tengan dificultades al momento de

llevar a cabo la práctica puedan tener una mayor comprensión, simulando un proceso real sin tener riesgo alguno. Para esto es importante responder la pregunta ¿cómo se puede lograr el desarrollo del gemelo digital de la planta con RA que cuente con las condiciones suficientes para garantizar la calidad del conocimiento?

1.2 Objetivos

1.2.1 Objetivo general

Digitalizar la Planta 1 del aula de Automatización de la Universidad Santo Tomas sede Bucaramanga enfocada en realidad aumentada utilizando programas de modelado, motores gráficos multiplataforma 2D y 3D, y protocolos de comunicación entre PLC, servidor y aplicación, con el propósito de generar una herramienta que permita realizar prácticas de forma semejante a la real.

1.2.2 Objetivos específicos

Construir un gemelo digital 3D de la Planta 1 del aula de Automatización mediante una suite de creación 3D (Blender) para el aprendizaje e interacción de los estudiantes de pregrado y especialización con equipos del área de automatización.

Definir las características de los objetos construidos en Blender mediante el motor gráfico (Unity) para facilitar la adquisición y transmisión de información del usuario al dispositivo.

Comunicar el gemelo digital con el dispositivo de PLC por medio de un servidor con el objetivo observar y manipular variables del dispositivo real.

Desarrollar el aplicativo móvil de realidad aumentada mediante el kit de herramientas AR proporcionadas por Unity con el fin posibilitar la interacción con el usuario.

1.3 Aportes del trabajo

El diseño del gemelo digital será útil para su eventual modificación y como referencia, guía y/o fundamento para el estudio, conocimiento y/o diseño de futuros sistemas de controles que integren realidad aumentada o virtual dentro y fuera de un entorno académico.

El resultado esperado del gemelo digital implicaría una configuración con el PLC S7-Siemens de la planta de automatización, asegurando un funcionamiento estable. Esto garantizaría la ausencia de complicaciones al utilizar la aplicación móvil. El sistema de control no debería requerir cambios en los elementos seleccionados, ni demandar extensas pruebas con el PLC, ahorrando así tiempo y recursos.

Las entidades o personas beneficiadas con este proyecto además del autor será la universidad Santo Tomás u empresas que cuenten con un PLC-s7 SIEMENS. De igual forma podrán beneficiarse los estudiantes y futuros ingenieros al momento de adquirir habilidades en su proceso de aprendizaje respecto al sector de automatización, a su vez estudiante necesiten trabajos de referencias para futuros proyectos relacionado con realidad aumentada.

El proyecto podrá servir como una alternativa al enfoque de aprendizaje convencional y como una fuente de estímulo para fortalecer los fundamentos académicos de los estudiantes, especialmente en el campo de la ingeniería.

1.4 Organización del Documento

1.4.1 Introducción

Se contextualiza al lector del tema del proyecto, se describe el por qué, los objetivos y la finalidad, presentando una visión general de lo que será encontrado en las siguientes páginas.

1.4.2 Marco conceptual

Se realiza el marco de referencia donde se presenta la base teórica y conceptual necesaria para comprender el problema y la investigación, a su vez, se incluyen antecedentes relevantes que facilitaron el desarrollo del proyecto.

1.4.3 Metodología

Se describe a detalle la metodología llevada a cabo en la investigación y se explica paso a paso como se realiza el proyecto.

1.4.4 Desarrollo

se presentan los hallazgos de la investigación, de igual forma, se muestra el diseño, simulaciones y pruebas.

1.4.5 Recomendaciones

Se habla sobre algunos factores a tener en cuenta al momento de intentar desarrollar una aplicación mejorada con realidad aumentada o bien sea realidad virtual.

1.4.6 Conclusiones

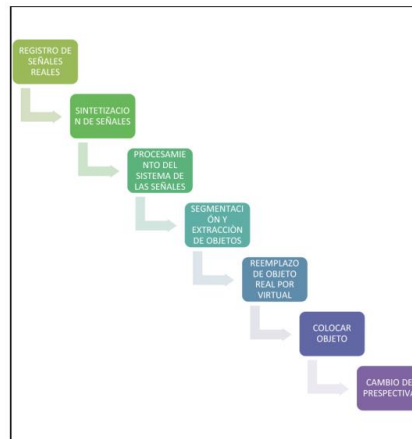
Se resumen los puntos clave del documento y se destacan las conclusiones generales del trabajo.

2. Marco de Referencia

2.1 Organización del Documento

2.1.1 Introducción

En los años 60 se observó la primera interfaz que agregaba a su desarrollo conceptos de RA. Sin embargo, fue en 1990 que el investigador Boeing Tom Caudel dio a conocer el término de Realidad Aumentada como un concepto tecnológico [1]. La RA es una tecnología que posibilita la superposición del mundo físico mediante el tratamiento de imágenes en tiempo real. En otras palabras, es una fusión del mundo físico con el mundo digital donde es posible interactuar con los dos mediante un dispositivo tecnológico, permitiendo al usuario la visualización de un entorno enriquecido, contribuyendo así a una percepción de la realidad más amplia o completa a través de información gráfica. Para el desarrollo de una tecnología de RA es necesario tener en cuenta los siguientes elementos: desde un dispositivo que capture imágenes del mundo exterior (realidad), el elemento donde se proyectarán las imágenes capturadas; por consiguiente, un elemento para el procesamiento de las imágenes proyectadas; por último, se necesita un activador de la realidad aumentada, es decir, un elemento de localización (GPS, brújula o acelerómetro).

Figura 1. *Proceso de Realidad Aumentada*

Tomado de: “C. R. Bello. 2017 ”[2]

Hay diversos sectores donde se aplica la realidad aumentada. En el campo del entretenimiento, por ejemplo, se utiliza en videojuegos y en el cine. También se utiliza en publicidad, ya que permite crear campañas impactantes y creativas. En educación, se utiliza para crear materiales interactivos y dinámicos que permiten a los estudiantes aprender de una forma más atractiva y participativa. En este caso el enfoque de interés es en la educación. La implementación de la RA en la educación ayudará a mejorar los procesos de aprendizaje, según los autores “Pedro & Martinez, 2012” [3] plantean que la RA es una alternativa metodológica en la educación primaria y parten de la idea de que toda herramienta multimedia que sirva para presentar material académico con fines educativos complementa los métodos de enseñanza tradicionales.

2.1.2 Automatización Industrial

El uso de la realidad aumentada (RA) en la automatización de procesos industriales en el campo de la mecatrónica es una aplicación innovadora que combina la automatización de procesos industriales y las tecnologías de RA para aumentar la eficiencia, la seguridad y la productividad

en entornos industriales. Algunos Ejemplos de casos de uso de la RA en la automatización industrial en mecatrónica:

2.1.2.1 Asistencia en mantenimiento y reparación. La RA se utiliza para proporcionar a técnicos y operarios información en tiempo real sobre el estado de los equipos industriales. Los trabajadores pueden utilizar gafas de RA para ver las instrucciones de reparación superpuestas en los equipos o recibir asistencia remota de expertos durante los trabajos de reparación.

2.1.2.2 Educación y Formación. La RA se utiliza para formar a los trabajadores en el manejo de equipos complejos. Los sistemas de formación basados en RA permiten a los estudiantes interactuar con modelos virtuales de máquinas y procesos, lo que simplifica la comprensión y reduce el tiempo de formación.

2.1.3 Gemelo Digital

El término gemelo digital (Digital Twin), lo aborda por primera vez el Dr Michel Grieves de la universidad de Michigan junto con Jhon Vickers en su trabajo de la Nasa en el año 2010 (con respecto al trabajo de desarrollo de productos realizado en la NASA), a comienzo de la década de los 2000 existían mecanismos para capturar la información sobre objetos físicos o unidades individuales eran primitivos, había una cantidad limitada de datos con el que se podían caracterizar los modelos digitales, con el paso del tiempo la adopción de sistemas MES (Manufacturing Execution System) ha permitido capturar mayor cantidad de información, facilitando la construcción de modelos digitales más complejos con una exactitud mayor [4]. Los gemelos digitales son modelos virtuales, réplicas de productos, procesos y medios de fabricación existentes

o futuros que nos permiten modelar e integrar información relevante y el comportamiento de los sistemas de producción en una pantalla o en medios inmersivos. Este proceso ayuda a comprender y anticipar problemas potenciales.

Los gemelos digitales usualmente se utilizan para actividades de validación de diseño o dimensionamiento. Para la implementación de gemelos digitales se hace necesario conocer el tipo de sistema mecatrónico, los parámetros de diseño y los sistemas de control de los mecanismos.

El estudio de los gemelos digitales plantea que la transición de eslabones mecánicos a contrapartes informáticos facilita indagar de forma completa el manejo de equipos y el análisis de los diferentes componentes de una máquina. Se concluye que el gemelo digital es mucho más que dibujos computacionales animados ya que con ellos se pueden considerar aspectos inherentemente prácticos sin implicar daños costosos en el momento en el que se utiliza alguna máquina sin tener experiencia alguna.

El poner en ejecución un gemelo digital puede ser una tarea difícil esto a que debe cumplir en totalidad las diferentes actividades que ejecuta. Los autores Orive, D., López, A., Estévez, E. & Marcos, M. (2021) [5] plantea separar de la siguiente manera:

1. Lo primero que se genera es un modelo del sitio que se va a simular, teniendo en cuenta las actividades específicas que este modelo debe seguir.

2. Como siguiente paso se plantea un proyecto de automatización el cual cuente con las características del gemelo digital es decir cuente con la misma lógica.

3. Para finalizar se usan arquitecturas de la industria 4.0, además de garantizar que se pueda transmitir esta información ya sea con protocolos de comunicación u otro tipo de transmisión.

2.1.4 WebSocket

En 2011, el Grupo de Trabajo en Ingeniería de Internet (IETF) estandarizó el protocolo RFC2 -6455, conocido como el protocolo WebSocket. A partir de ese momento, la mayoría de los navegadores web comenzaron a implementar clientes que soportan este protocolo. Además, se crearon bibliotecas Java para permitir la implementación de WebSocket desde otros clientes, como las aplicaciones Android3 o JavaFX4. Una solución al paradigma mencionado anteriormente es utilizar una única conexión TCP para el tráfico bidireccional, que es precisamente lo que hace el protocolo WebSocket. WebSocket define una Interfaz de Programación de Aplicaciones (API) que establece conexiones socket entre un servidor y un navegador web. [6]

El Protocolo de WebSocket le da el usuario una comunicación entre un servidor y un cliente, esto crea un enlace entre los dos dispositivos permitiendo que entre ellos se envíen información basada en mensajes a través de TCP [6].

2.1.5 Node-RED

Figura 2. Logo de Node-Red



Tomada de “Resources. (s/f). Nodered.org.”.2023. [7]

Según E. Ollora Zaballa [8] “Node-RED es un editor de flujos que también utiliza el navegador para poder interconectar nodos”. Los nodos se pueden conectar a un navegador. La herramienta contiene nodos que representan una funcionalidad muy básica, como activadores que crean activadores para producir estimulación en procesos o herramientas muy complejas como brokers o clientes MQTT. Node-RED se basa principalmente en la creación de funciones en código JavaScript para poder generar, procesar o modificar mensajes recibidos por el servidor. La característica principal de Node-RED es su capacidad de generar muchos flujos complicados. Es por esto por lo que este tipo de herramientas son muy útiles cuando se tiene que recibir, cambiar, adaptar y almacenar datos.

2.1.6 Node.JS

Figura 3. *logo de node.js*

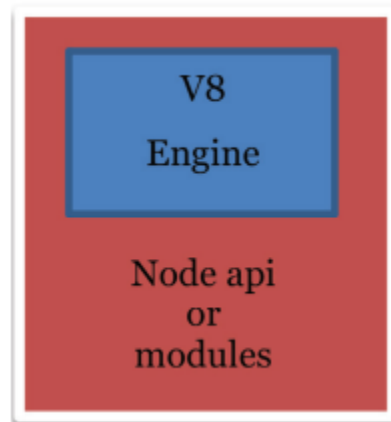


Tomada de: “Node.js” .2017[9]

Node.js es un entorno de código abierto que permite compilar código de JavaScript esto todo sin la necesidad de un navegador, es utilizado por desarrolladores ara construir aplicaciones de red en el lado del servidor de manera escalable, rápida y confiable, puede manejar solicitudes de HTTP y flujos de datos, soporta sistemas de archivos y puede administrar varios procesos de back-end simultáneamente. [9]

Es un entorno de tiempo de ejecución de código abierto, basado en eventos, este puede manejar múltiples conexiones sin necesidad de subprocesos múltiples. Es ideal para construir tipos de aplicaciones escalables.

Figura 4. *Entorno de ejecución*



Tomado de: “Desarrollo backend para aplicaciones ”.2023[10]

Algunas características de Node.js, destacables son las siguiente:

2.1.6.1 Eficiencia. Node.js utiliza un modelo de E/S controlado por eventos y sin bloqueo que le permite manejar una gran cantidad de conexiones simultáneas sin consumir recursos del sistema.

2.1.6.2 Gran comunidad y ecosistema. Node.js tiene una gran comunidad de desarrolladores y una amplia gama de bibliotecas y módulos disponibles a través de npm (Node Package Manager).

2.1.6.3 JavaScript en ambos lados. Node.js permite a los desarrolladores utilizar JavaScript tanto en el lado del cliente como en el del servidor, lo que facilita la creación de aplicaciones web completas utilizando un único lenguaje de programación.

2.1.6.4 Escalabilidad. Node.js es conocido por su capacidad para manejar aplicaciones escalables y de alto rendimiento, lo que lo hace popular para aplicaciones de transmisión y en tiempo real.

Cuando trabajas con Node no tienes la preocupación de que se bloquee el proceso ya que de por si no existe en Node esto da la facilidad de escalar los sistemas ya que no se cuenta con esta limitante [10].

2.1.7 Ubuntu

Figura 5. Logotipo de Ubuntu



Tomado de: Ubuntu 2022.[11]

Es una distribución de LINUX de código abierto, creada a principios del año 2004 El creador de Ubuntu fue Mark Shuttleworth, lo hizo con el fin de tener un sistema operativo que fuera accesible para todas las personas, a su vez, que tuviera la marca de la filosofía básica del paquete de software [12]. La distribución Ubuntu es utilizada en una variedad de entornos, desde escritorios personales, hasta sistemas integrados. Ubuntu presenta una interfaz similar a la

ofrecida generalmente en Windows, por ende, tiene una forma variada y fácil de instalación, los requerimientos mínimos para poder instalar el software son:

- Memoria RAM de 512 Mb
- Espacio en Disco Duro de 2 Gb
- Arquitectura de 32 o 64 bits

2.1.7.1 Características de Ubuntu.

2.1.7.1.1. Entorno de escritorio. Ubuntu tiene un entorno que llaman Unity, estos entornos proporcionan una interfaz gráfica.

2.1.7.1.2 Software libre. La mayoría del software en las distribuciones se realiza bajo licencias de código abierto. Según Martínez Juliá [13] Esto implica lo siguiente:

- El programa fuente donde se genera es accesible para cualquier persona que desee visualizarlo.
[14]
- Quien quiera realizar cambios a dicho programa (código fuente), lo puede hacer sin permiso previo.
[14]
- Se puede distribuir el software inicial como el software con modificaciones. [13]

2.1.7.1.3 Amigable para el usuario. Ofrece una experiencia amigable con el usuario, su instalación es sencilla, la interfaz es intuitiva y se puede tener acceso a una amplia variedad de aplicaciones mediante el gestor de paquetes.

En Ubuntu, se encuentran varias variantes que se adaptan a diferentes necesidades, algunas de esas variantes son las siguientes:

2.1.7.1.4 Ubuntu Desktop. Este es el modelo predeterminado para las variantes de Debian, y se centra en características orientadas a usuarios de escritorio.

Ubuntu Server ofrece una interfaz de línea de comandos en lugar de la interfaz gráfica Unity.

2.1.7.1.5 Ubuntu Core. Esta variante es altamente versátil y se puede adaptar a dispositivos como enrutadores, refrigeradores, drones y Raspberry Pi.

2.1.8 ARcore

Google ha desarrollado una plataforma para simplificar la creación de experiencias de realidad aumentada. Utiliza varias APIs, como la “ARCore API”, “Sceneform API” y “Sceneform grandle plugin”, que permiten a los dispositivos con cámara integrada detectar su entorno y aplicar un mapeo de puntos para interpretar las posiciones de una manera sencilla. La versión utilizada es la 1.22.0. También se han utilizado bibliotecas propias como “GoogleARCore” [14].

Figura 6. *Logotipo de ARcore*



Tomado de: ¿Que es AR Foundation ?.2022 [15]

2.2 Marco Conceptual

2.2.1 Realidad Aumentada

La realidad aumentada (RA) es una tecnología que fusiona elementos reales y virtuales con el fin de brindar una mejor experiencia y comprensión del entorno. Los elementos de la RA son los siguientes:

2.2.1.1 Entorno Real. La RA se basa en el entorno físico del usuario, que puede registrarse mediante dispositivos como cámaras, sensores o gafas de realidad aumentada.

2.2.1.2. Cámara. Dispositivo para capturar las imágenes del mundo real, que funcionará como guía de introducción al mundo real.

2.2.1.3. Procesador. Elemento responsable de combinar las imágenes y procesar la información.

2.2.1.4. Elementos Virtuales. Los elementos virtuales son objetos digitales, gráficos o información superpuestos al entorno real. Estos elementos pueden incluir imágenes, vídeo, texto, modelos 3D o información contextual.

2.2.1.4.1 Software. Programa con soporte lógico que gestiona el proceso.

2.2.1.4.2 Marcador. Imagen procesada por la computadora a la que se le superpone mediante la programación definida de la imagen los objetos 3d.

2.2.1.4.3 Interacción en tiempo real. El valor fundamental de la RA es la interacción en tiempo real entre el usuario y los elementos virtuales. Esto significa que los objetos virtuales pueden responder inmediatamente a las acciones o movimientos del usuario.

2.2.1.5. Visualización. La visualización de la RA puede realizarse a través de diversos dispositivos, como gafas de realidad aumentada, teléfonos inteligentes, tabletas o pantallas especiales. Estos dispositivos permiten al usuario ver tanto el entorno real como los elementos virtuales que se superponen a él.

2.2.1.6. Seguimiento y Reconocimiento. El seguimiento y reconocimiento del entorno físico y de los objetos es necesario para que la RA funcione correctamente. Los sistemas de RA utilizan cámaras y sensores para detectar y seguir los elementos del mundo real, de modo que los elementos virtuales puedan colocarse con precisión.

2.2.1.7. Interacción en la RA. Describe cómo los usuarios interactúan con los objetos y la información en entornos de RA, incluyendo gestos, comandos de voz y dispositivos de entrada.

2.2.2 Automatización Industrial en Ingeniería Mecatrónica

Según Groover (2000) [16] La automatización en los sistemas productivos se describe como una tecnología que implica la utilización de sistemas mecánicos y electrónicos integrados con sistemas informáticos para la operación y control de la producción.

2.2.3 Gemelo Digital

Se refiere a un modelo de simulación que duplica las propiedades o el comportamiento del sistema en cuestión. Este término subraya que el modelo es una réplica precisa del sistema, abarcando varios aspectos y capaz de responder a preguntas detalladas en diversas disciplinas o en relación con el sistema en su conjunto.

2.3 Marco Legal

El decreto 1412 de 2017, emitido por el presidente de la república Juan Manuel Santos, establece un marco legal que busca regular los numerales 23 y 25 del artículo 476 del Estatuto tributario. Este decreto es basado en la ley 1341 de 2009, que regula el sector de las Tecnologías de la información y las Comunicaciones (TIC), El decreto tiene como objetivo “Fomentar el desarrollo de contenidos y aplicaciones”, así como “Estimular y fomentar el crecimiento de la industria de las tecnologías de la información y las comunicaciones para contribuir al crecimiento económico, la competitividad, la creación de empleo y las exportaciones” [17].

El decreto establece un marco regulatorio para una amplia variedad de servicios y contenidos digitales, incluyendo animación digital, desarrollo de videojuegos, diseño y edición sonora, edición gráfica, impresión aditiva, inteligencia artificial, modelado 2D y 3D, programación, realidad virtual y aumentada, entre otros. Cada uno de estos servicios de educación virtual y desarrollo de contenidos digitales se define y se regula en el contexto del decreto.

2.4 Marco Tecnológico y Científico

Del proyecto a ejecutar se conocen trabajos relacionados con el tema principal, sin embargo, la mayoría de las aplicaciones que se tienen ya desarrolladas se basan principalmente en el sector industrial con realidad aumentada (laboratorios).

Las practicas remotas se realizan desde hace más de dos décadas, inicialmente se aplicaba en entornos de relacionados con el espacio siendo algunos telescopios y radares, con el surgimiento del internet esta práctica se extendió a más tipos de laboratorios aumentando drásticamente el uso de estos. (Fagua & Rojas, 2013) [18].

Se ha encontrado el proyecto de Valderrama Serrato, J. N. (2018) [19], Este texto se refiere al diseño y creación de un laboratorio remoto de automatización con aplicaciones de realidad aumentada que permiten visualizar un proceso industrial en un contexto de manufactura. Este proyecto se llevó a cabo con el objetivo de demostrar, a través de conceptos como los laboratorios remotos, la realidad aumentada y la Industria 4.0, la formulación de un entorno de ejercicios de automatización que permita la contextualización de procesos industriales fuera de los laboratorios tradicionales. La metodología de este proyecto se realizó en tres fases: investigación de Tecnologías de Realidad Aumentada: El proyecto comenzó con una fase de investigación para explorar las tecnologías de realidad aumentada disponibles y relevantes para su aplicación en el

laboratorio. Construcción del Sistema Cliente-Servidor: Se construyó un sistema cliente-servidor para el laboratorio remoto. Los usuarios pueden autenticarse en este sistema, y el acceso al laboratorio se logra utilizando el protocolo RDP (Remote Desktop Protocol) para conectarse al servidor del laboratorio. Desarrollo de Aplicaciones de Realidad Aumentada (RA): Se desarrollaron aplicaciones de RA y ejercicios de automatización que permiten a los estudiantes interactuar con los procesos industriales en un entorno virtual de RA. [19]

El laboratorio de acceso remoto ofrece la oportunidad de ver cinco ejercicios de automatización industrial en realidad aumentada, lo que proporciona a los estudiantes experiencia práctica y visual relacionada con la programación de controladores lógicos programables (PLC) y su integración en entornos de fabricación flexible o Industria 4.0. Se concluye que el diseño del laboratorio remoto tiene un impacto relevante en la educación, ya que permite a los estudiantes interactuar con la realidad aumentada y adquirir una comprensión sólida de conceptos de automatización industrial e integración de tecnologías de vanguardia en la industria 4.0

Por otro lado, el proyecto de Daquilema, M. Á., Checa, D. S., & Martin, C. A. (2020) [20], Se basa en el diseño de entornos de realidad aumentada para la monitorización y mantenimiento de procesos o máquinas industriales. Este trabajo propone desarrollar un entorno de realidad aumentada para la monitorización y mantenimiento de procesos o máquinas industriales. Esta tecnología ofrece numerosas ventajas, como reducción del tiempo de formación para mantenimientos complejos, reducción del riesgo laboral, visualización de variables PLC importantes en el proceso, visualización rápida de instrucciones de funcionamiento o acceso remoto a ayuda de expertos, etc.

Para finalizar, se encontró la tesis de Flórez Farfán, Y. G. (2020). [21] El objetivo era presentar un diseño para una aplicación piloto de realidad aumentada que apoye el mantenimiento

instrumental de una planta de aglomeración de cobre. La aplicación se diseñó en base a los componentes de control existentes, lo que facilitó su integración. Además, la aplicación permitirá al personal de mantenimiento de la planta acceder rápidamente a la información esencial, lo que permitirá un diagnóstico temprano de cualquier fallo o problema del sistema. De esta manera, el responsable podrá resolverlo en el menor tiempo posible, reduciendo así los tiempos de paradas no programadas.

2.5 Condiciones Iniciales

El base inicial de este proyecto está basado en la tesis de maestría “ESTRATEGIA VIRTUAL ENFOCADA A LA INDUSTRIA 4.0 SOBRE UNA PLANTA PILOTO INDUSTRIAL [22], la cual suministra las bases para llevar a cabo el proyecto.

3. Diseño Metodológico

La metodología utilizada en este proyecto es descriptiva, de igual forma se realiza una investigación exploratoria, debido a que, se pretende realizar pruebas y experimentos para comprender el funcionamiento en temas como tecnologías de vanguardias como la realidad aumentada en el sector de Automatización Industrial de la Facultad de Ingeniería Mecatrónica. Esta investigación exploratoria permite descubrir detalles de la realidad aumentada en la automatización industrial, explorando posibilidades y limitaciones antes de una implementación completa.

3.1 Criterios de Diseño

Para la construcción del gemelo digital con realidad aumentada se tuvieron en cuenta los siguientes criterios de diseño de selección:

3.1.1 Gemelo Digital

El gemelo digital debe ser igual a la planta 1 del aula de automatización. El desarrollo del modelo virtual es realizado en el software de diseño Blender.

Tabla 1. *Comparación entre software de diseño*

Software	Licencia	Modelado 3D	Renderizado	Simulación	Integración con juegos	Multiplataforma
Blender	Código abierto	Sí	Sí	Sí	Sí	Sí
ZBrush	Comercial	No	No	No	No	No
Cinema4D	Comercial	Sí	Sí	Sí	No	Sí
3ds Max	Comercial	Sí	Sí	Sí	No	No
Autodesk Maya	Comercial	Sí	Sí	Sí	Sí	Sí

Se elige el software de diseño Blender debido a que es accesible para cualquier persona, a su vez, cuenta con una diversa gama de capacidades, desde el modelado hasta simulaciones, Blender es versátil y puede usarse para cualquier tipo de proyectos relacionados a la creación de video juegos.

3.1.2 *Interactividad intuitiva.*

La aplicación de realidad aumentada deberá facilitar a los usuarios la interacción con las piezas mediante toques. La app tendrá que ser estable, esta es creada con el motor gráfico Unity.

Tabla 2. *Características de motor grafico Unity*

CARACTERÍSTICAS DE	VENTAJAS	DESVENTAJAS
UNITY		
Desarrollo de modelado 3D	Unity es una plataforma versátil que permite crear modelos 3D y escenas para aplicaciones de AR.	Puede haber una curva de aprendizaje para usuarios nuevos en el modelado 3D
Interactividad intuitiva	Unity proporciona herramientas y componentes para desarrollar interacciones intuitivas en AR.	
Multiplataforma	Unity es compatible con una amplia variedad de plataformas.	La adaptación a diferentes plataformas puede requerir tiempo adicional de desarrollo.
Integración de AR	Unity ofrece soporte nativo para el desarrollo de aplicaciones de Realidad Aumentada (AR).	
Creación de Experiencias Inmersivas	Unity permite la creación de experiencias de AR que involucran a los usuarios de manera directa.	
Amplias funcionalidades	Ofrece una amplia gama de funciones y herramientas para	

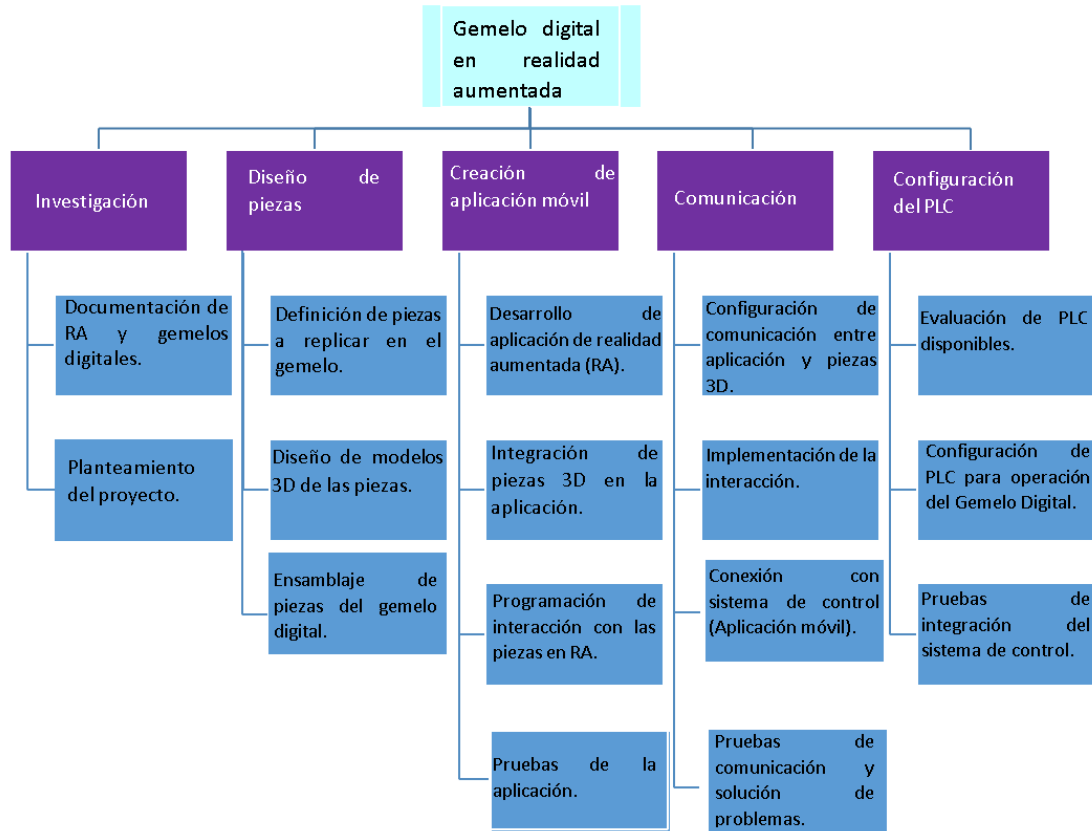
CARACTERÍSTICAS DE	VENTAJAS	DESVENTAJAS
UNITY		
	desarrollo de aplicaciones de AR.	

3.1.3 Comunicación entre el gemelo digital y el plc

La comunicación debe ser estable, para así minimizar las interrupciones al momento de tener en uso la aplicación. La comunicación se establece mediante el protocolo de comunicación Websocket.

3.1 Descripción de etapas y tareas

Figura 7. Estructura de desglose de trabajo EDT



La primera etapa del proyecto es la fase de investigación. Se realiza una revisión detallada de literatura relacionada con la digitalización de plantas industriales, realidad aumentada y automatización. Las tareas para llevar a cabo en la primera fase son las siguientes:

- Investigar aplicaciones de realidad aumentada en educación.
- Analizar proyectos similares de gemelos digitales enfocados a un entorno industrial
- Recopilar recursos y literatura correspondientes a gemelos digitales y realidad aumentada.

Posteriormente a esto se realiza el planteamiento del proyecto para así iniciar la siguiente fase. Las tareas incluyen:

- Definir objetivos generales y específicos.

- Establecer alcance del proyecto y tareas que se deben cumplir.
- Establecer el plan del proyecto, incluyendo cronograma y recursos.

En la segunda etapa se identifican y definen las piezas específicas a diseñar para replicar el gemelo digital. Las tareas involucran:

- Identificar las piezas claves para la construcción 3D del gemelo digital.
- Documentar las especificaciones de las piezas seleccionadas.

Luego, se realiza el modelado de forma precisa de las piezas 3D en el software llamado Blender de la planta de automatización. Las tareas comprenden:

- Modelar las piezas necesarias de forma precisa.
- Realizar pruebas de precisión en los modelos 3D de cada una de las piezas.

Una vez terminadas todas las piezas, se integraron, para finalmente obtener el gemelo digital. Las tareas incluyen:

- Ensamblar las piezas modeladas, respetando la disposición física de cada pieza en la planta de automatización.
- Comprobar la conexión y disposición de las piezas, teniendo en cuenta la planta real.

En la tercera etapa se desarrolla la aplicación de realidad aumentada (RA), que va a permitir a los usuarios interactuar con el gemelo digital de la planta de automatización. Las tareas específicas incluyen:

- Programar la aplicación utilizando una plataforma de desarrollo.
- Integrar funcionalidades de realidad aumentada, como el seguimiento de objeto y el reconocimiento de marcadores.
- Diseñar la interfaz de usuario de la aplicación con el fin de garantizar una experiencia de usuario intuitiva.

Una vez, teniendo creada la aplicación, se integran los modelos de las piezas 3D a esta, permitiendo a los usuarios la interacción con las piezas. Las tareas para integrar los modelos de las piezas 3D a la aplicación consisten en:

- Importar los modelos de cada pieza a la aplicación de RA
- Programar la interacción con las piezas permitiendo la manipulación de estas.

Después de haber sido previamente configurada las piezas para su interacción se programan interacciones específicas. Las tareas involucran:

- Programar la interacción de los usuarios con las piezas, incluyendo funciones como rotación, zoom y visualización de información técnica.
- Integrar información técnica y descriptiva de las piezas para su visualización en RA, proporcionando a los usuarios datos relevantes.

Finalmente, se hacen pruebas de la aplicación de realidad aumentada para garantizar su funcionamiento correcto. Las tareas específicas incluyen:

- Hacer pruebas de uso y experiencia de usuario para evaluar la facilidad de uso y la satisfacción del usuario.
- Identificar y solucionar problemas o errores en la aplicación de RA.
- Validar que la aplicación funcione de manera eficaz en una variedad de dispositivos móviles.

En la etapa cuatro, se configura la comunicación entre la aplicación de RA y la planta de automatización asegurando una interacción fluida. Las tareas específicas abordan:

- Establecer los protocolos de comunicación necesarios entre la aplicación y planta de automatización

- Garantizar la sincronización y transferencia de datos para una experiencia de usuario sin interrupciones.

Una vez establecidos los protocolos de comunicación, se implementan las funciones de interacción entre la aplicación de RA y las piezas, permitiendo a los usuarios interactuar de manera efectiva. Las tareas incluyen:

- Programar la interacción entre los usuarios y las piezas en el Gemelo Digital.
- Validar que la interacción funcione de manera precisa y fluida.

Posteriormente, se establece la conexión entre la aplicación móvil y el sistema de control, permitiendo a los usuarios controlar y supervisar las piezas. Las tareas abordan:

- Configurar la conexión entre la aplicación móvil y el sistema de control industrial.
- Desarrollar interfaces de control y supervisión dentro de la aplicación móvil para permitir una gestión efectiva de las operaciones.

Luego, se llevan a cabo pruebas de la comunicación entre la aplicación móvil y el sistema de control. Las tareas comprenden:

- Realizar pruebas de comunicación para verificar la eficiencia y confiabilidad del sistema.
- Identificar y solucionar problemas de comunicación, latencia o errores que pudieran surgir durante la interacción.

En la quinta etapa, se utiliza el Controlador Lógico Programable (PLC) disponible para operar el Gemelo Digital. Las tareas específicas involucran:

- Seleccionar el PLC que cumple con los requisitos específicos del proyecto, garantizando una integración efectiva.

Luego, se configura el PLC disponible para que sea compatible con el Gemelo Digital y permita el control de las piezas modeladas. Las tareas incluyen:

- Programar el PLC para que pueda comunicarse y controlar las piezas del Gemelo Digital de manera precisa y eficiente.
- Integrar el PLC en el sistema de control general del Gemelo Digital, asegurando una operación coordinada.

Para terminar, se realizan pruebas para garantizar el correcto funcionamiento del PLC y el sistema con el Gemelo Digital y la aplicación de RA. Las tareas involucran:

- Realizar pruebas de integración para verificar la comunicación efectiva entre el PLC y las piezas del Gemelo Digital.
- Validar que el sistema de control respondiera adecuadamente a las interacciones de los usuarios, garantizando un control preciso y seguro.

3.3. Descripción de los productos finales.

A lo largo del desarrollo del proyecto, se han creado varios productos que aportan a la solución del problema planteados, se presentan estos productos finales como una contribución al campo de ingeniería mecatrónica en la universidad Santo Tomás.

En la primera etapa, el producto final es el Modelo 3D del gemelo digital, el modelo incluye la representación de cada una de las piezas seleccionadas. El modelo es la base para la experiencia de realidad aumentada, a medida que, se vayan desarrollando los siguientes productos. En la siguiente etapa se desarrolla una aplicación de realidad aumentada con el fin de permitir a los usuarios la interacción con el gemelo digital. La aplicación fue programada a través del motor gráfico digital Unity. Esta herramienta proporciona una interfaz intuitiva y práctica. Además, se estableció la comunicación entre la aplicación de realidad aumentada y el PLC, permitiendo controlar la planta de automatización real, a su vez, supervisar las operaciones de la planta de

forma remota. Finalmente, se realizaron pruebas de validación para verificar y garantizar el correcto funcionamiento del sistema de control. Los productos finales se presentan de forma detallada en el capítulo 4, este incluye fotografías, pruebas; esto representan una solución que combina el modelado 3D, realidad aumentada y comunicación con sistemas de control industrial. Estos resultados tendrán un impacto en la implementación de modelos de aprendizajes educativos y la formación en un contexto industrial.

3.4. Conclusión del Capítulo

El proyecto de gemelo digital con realidad aumentada es un ejemplo destacado de cómo la combinación de tecnologías como la realidad aumentada y modelado 3D puede aportar beneficios significativos en el ámbito educativo. Un aspecto fundamental para lograr la culminación de este proyecto se basa en la implementación de criterios de diseño apropiados. Estos criterios garantizan la precisión del gemelo digital, una interactividad intuitiva y una comunicación estable con el Controlador Lógico Programable (PLC). Además, la selección de software y herramientas se llevó a cabo, considerando minuciosamente sus ventajas y desventajas. El enfoque metodológico empleado dividió el proyecto en etapas bien definidas, cada una con tareas específicas. Esto permitió un desarrollo ordenado y metódico, facilitando la gestión y supervisión del proyecto en su conjunto. Los productos finales de este proyecto no se limitan solo a un modelo 3D y una aplicación de realidad aumentada, sino que también incluyen una integración efectiva con sistemas de control industrial. En resumen, el proyecto de gemelo digital con realidad aumentada es un ejemplo interesante de cómo la tecnología puede transformar la forma en que aprendemos y nos entrenamos en el mundo actual.

4.Desarrollo

La planta base para el gemelo digital con realidad aumentada que se va a modelar tiene la responsabilidad de llevar a cabo diversas prácticas de laboratorio en áreas como sistemas SCADA, Sistema de Control Distribuido (DCS), instrumentación industrial y laboratorio, IoT, automatización industrial y laboratorio, Instrumentación Virtual, control de procesos y mantenimiento industriales.

En la Facultad de Mecatrónica de la Universidad Santo Tomás, se encuentra una planta de automatización instrumentada que consta de 24 dispositivos industriales, tuberías de PVC de media pulgada y dos tanques cuadrados. Esta planta se utiliza para llevar a cabo prácticas de laboratorio relacionadas con la medición de nivel de agua, temperatura y flujo de agua.

La planta piloto está diseñada de manera que puede conectarse a un módulo PLC S7-1200, ya que incluye un tablero de distribución general que funciona a 24 voltios de corriente continua (24V DC). Además, cuenta con terminales de conexión y luces piloto de color verde y rojo que permiten visualizar el estado de los actuadores y sensores. Asimismo, dispone de un interruptor electromagnético y un botón de emergencia en la parte superior para situaciones críticas.

En cuanto a la organización de la planta, esta cuenta con un diagrama de tuberías (PI&D) que proporciona la nomenclatura de cada dispositivo de acuerdo con su ubicación en el diagrama. Para asegurar el correcto funcionamiento de los dispositivos, la planta tiene incorporada dos tarjetas electrónicas con componentes de electrónica de potencia. Estas tarjetas desempeñan un papel fundamental en la protección y regulación de la corriente eléctrica de los dispositivos, al tiempo que facilitan la conexión con el PLC S7-1200 mediante su tablero de control.

4.1. Diseño

4.1.1 Etapa 1

Esta ya fue desarrollada, esto se comprueba con el capítulo 1 y 2 del proyecto.

4.1.2 Etapa 2

4.1.2.1 Modelado de piezas en Blender. El modelado 3D de las piezas, se realizó en el Software de diseño Blender, las piezas fueron escaladas respecto a la planta de Automatización de la universidad Santo Tomás de Aquino. Es imprescindible contar con conocimientos previos del software, ya que, las guías proporcionan poca información de funciones específicas en este.

Figura 8. Electroválvula FPD270



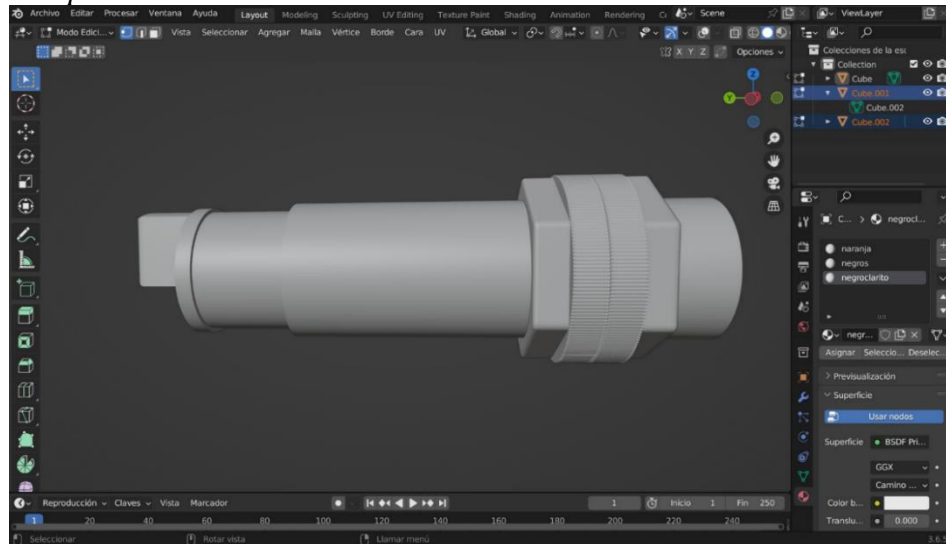
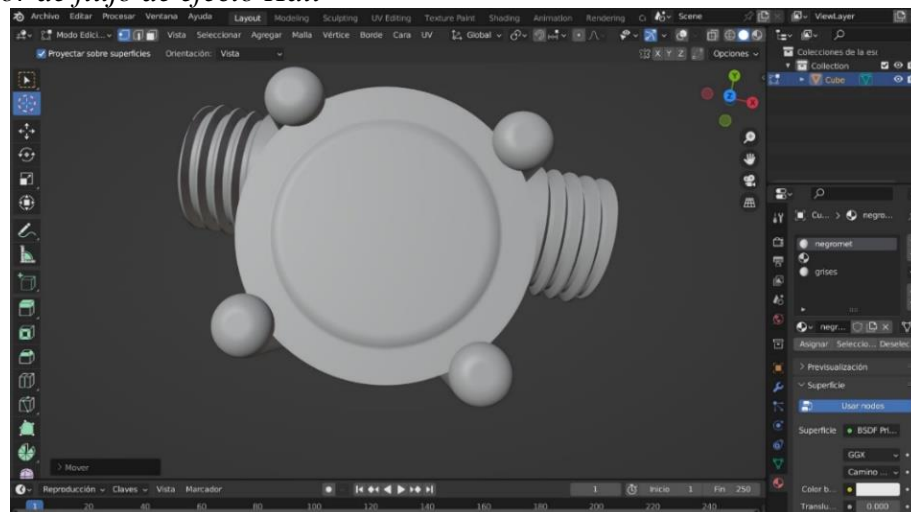
Figura 9. *Sensor capacitivo***Figura 10.** *Sensor de flujo de efecto Hall*

Figura 11. Valvula selenoide



Figura 12. codo de pvc

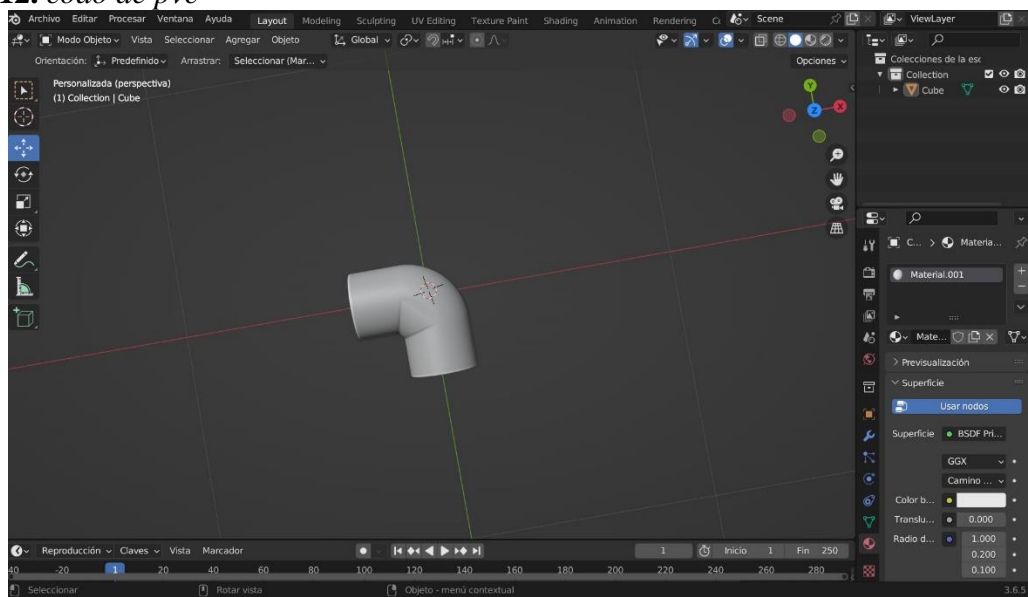


Figura 13. Base de planta de automatización.

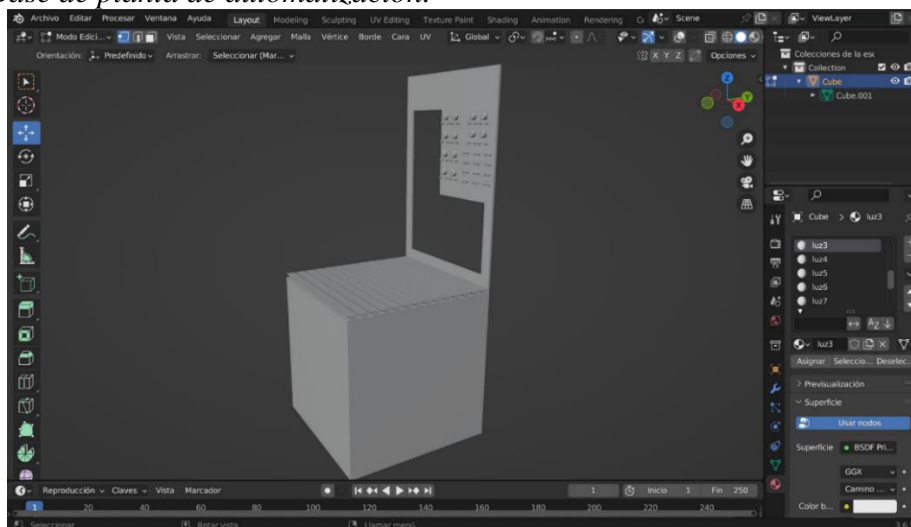
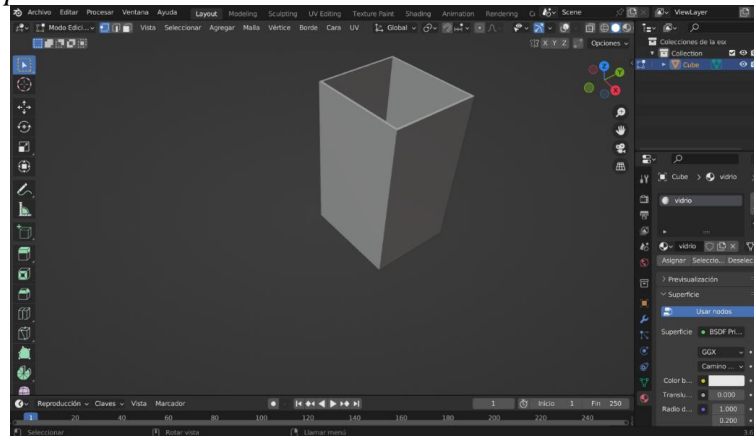
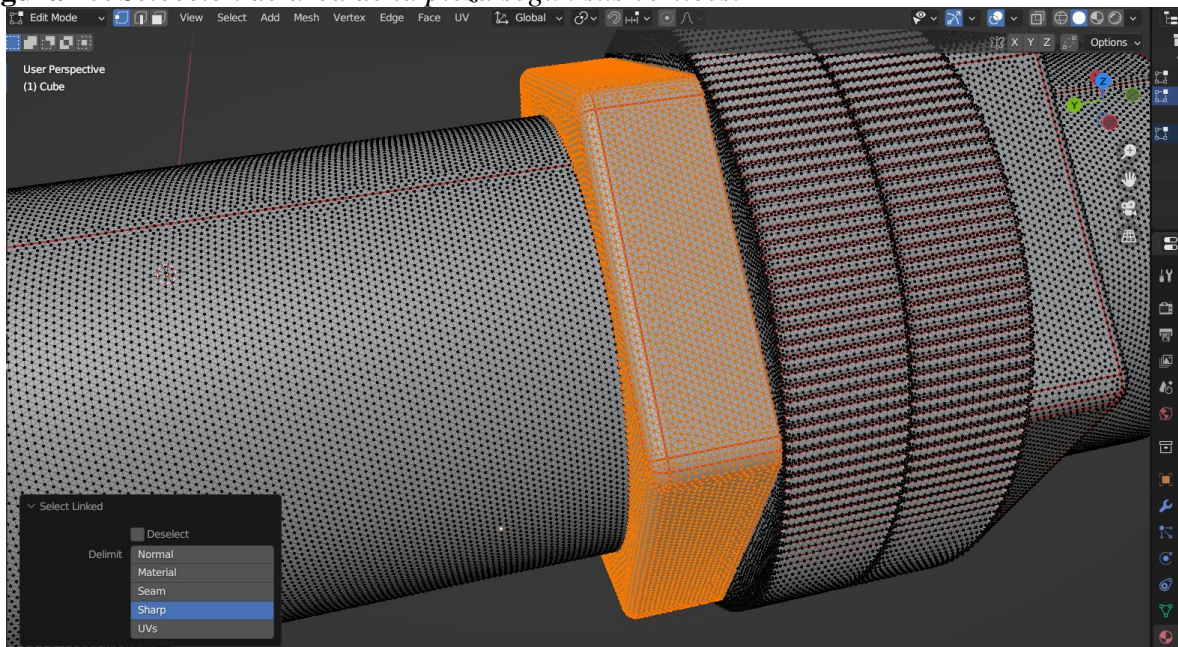


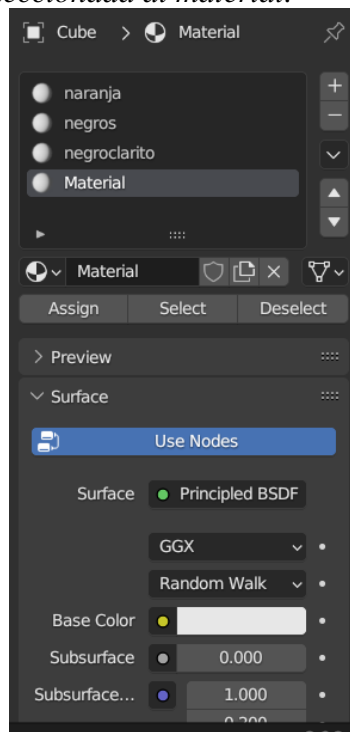
Figura 14. *Tanque de plástico.*

Para darle características a las piezas desarrollada en Blender, después de tener el diseño de la pieza completo, se entra en modo edición o “edit mode” el cual permite seleccionar áreas de la pieza según sus vértices, sus aristas o en general con puntos generados por el programa.

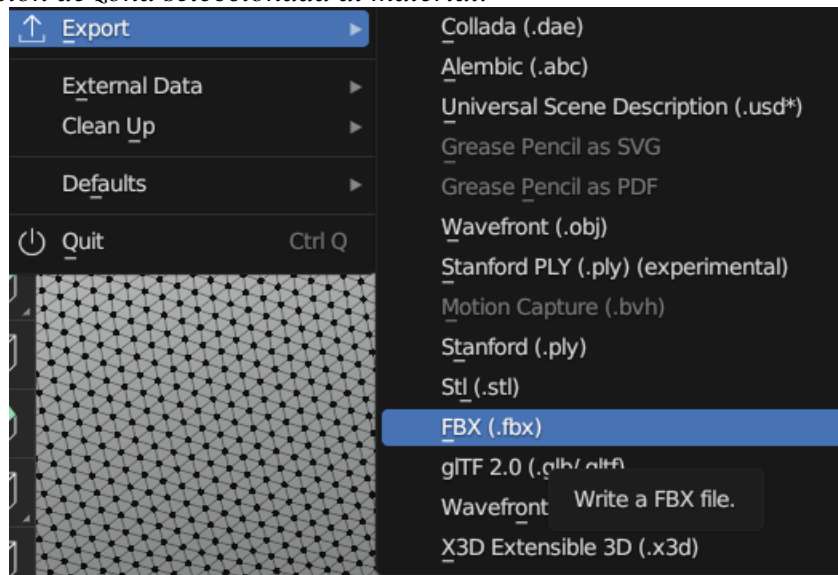
Figura 15. *Selección de área de la pieza según sus vértices.*

Teniendo el área definida, se procede a crear una propiedad a la cual va a pertenecer esta puede ser un “shader, un material, una animación entre otros, en este caso se crea un material el cual se podrá modificar propiedades desde el código en unity. Es importante que la zona seleccionada sea asignada a el material de lo contrario no se podrá modificar después de exportada.

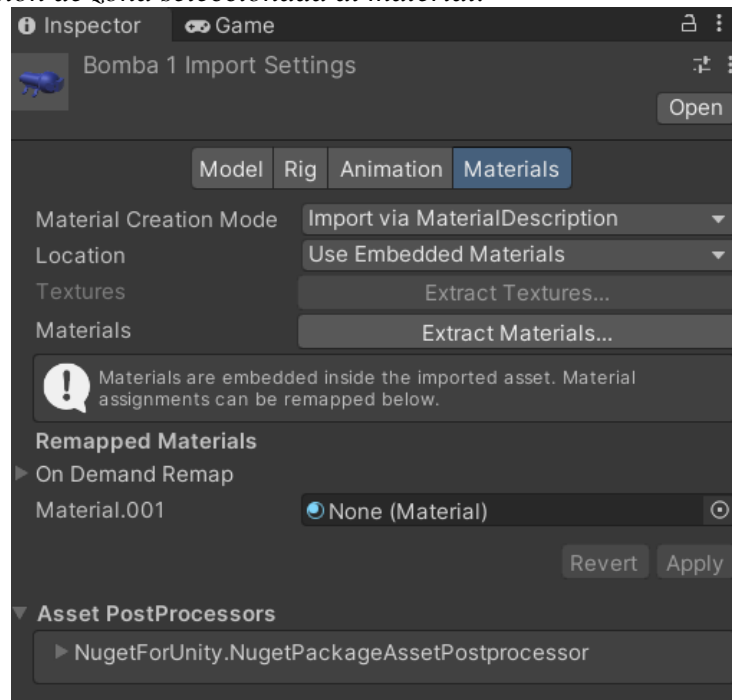
Figura 16. *Asignación de zona seleccionada al material.*



Teniendo las propiedades definidas de la pieza se tiene que exportar el archivo, el archivo para que sea utilizado en Unity tiene que estar en tipo fbx.

Figura 17. *Asignación de zona seleccionada al material.*

Al adjuntar las piezas en Unity tenemos que extraer materiales y características para poderlos modificar en Unity.

Figura 18. *Asignación de zona seleccionada al material.*

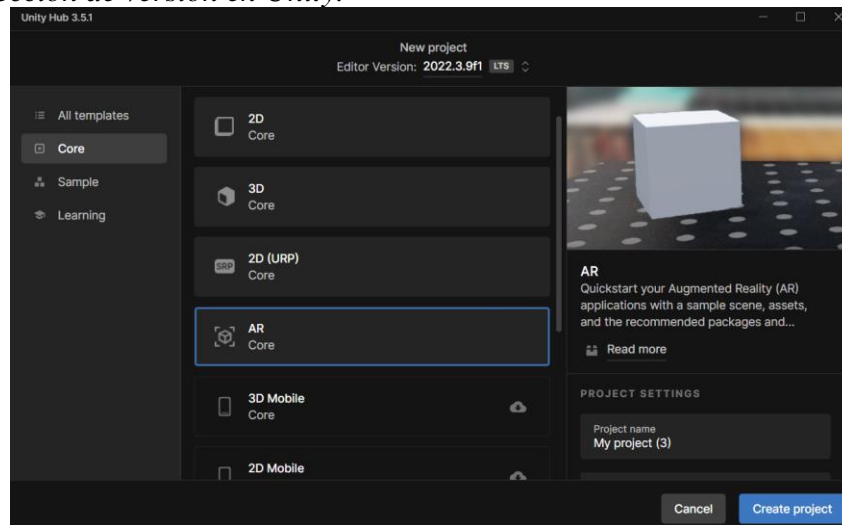
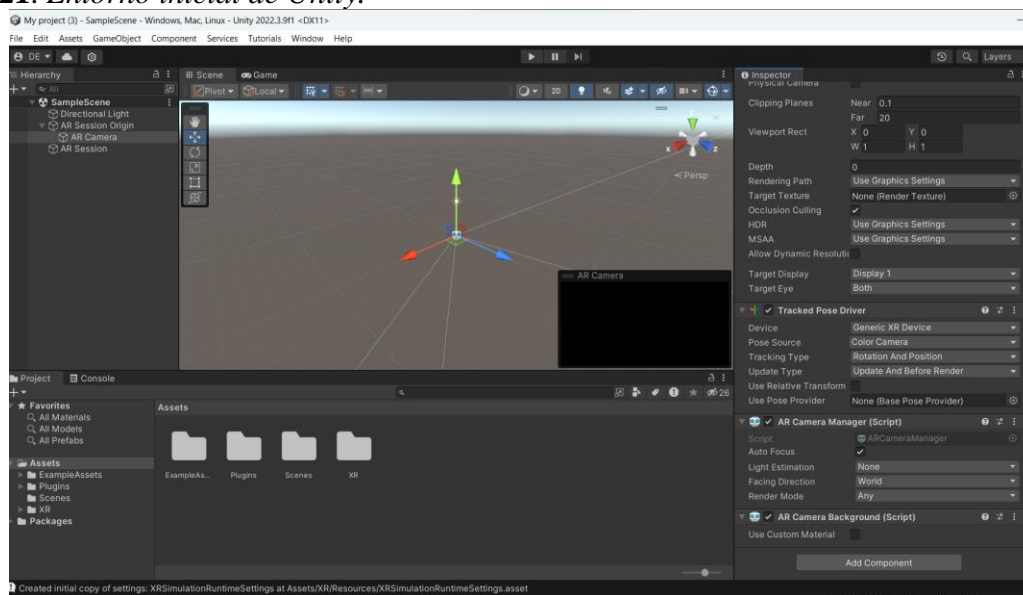
Como resultado de la extracción se obtendrá el modelo y los materiales como objetos configurables.

Figura 19 .Objetos configurables.

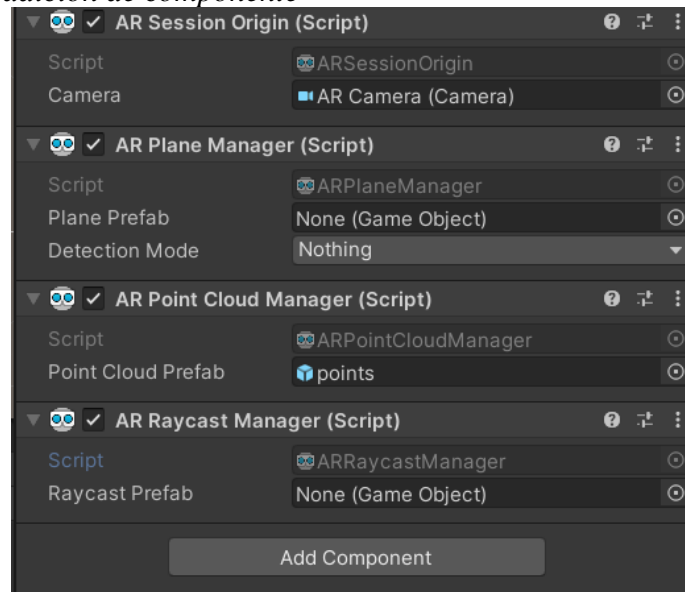


4.1.3 Etapa 3

4.1.3.1 Creación de aplicación móvil. Para crear una aplicación compatible con ARCore se debe configurar adecuadamente desde la creación del proyecto en Unity, para esto se selecciona la versión de Unity a utilizar, se recomienda usar una versión LTS ya que esto garantiza la estabilidad a el momento de usar la plataforma, luego de esto el proyecto debe ser de tipo AR todo esto se realiza desde el “Unity Hub”, con esta configuración hecha se obtiene un proyecto vacío con los templates de ARCore.

Figura 20. Selección de versión en Unity.**Figura 21.** Entorno inicial de Unity.

Con la creación del proyecto se van a generar los elementos que permiten el funcionamiento de la realidad aumentada, el objeto principal que se encarga del procesamiento sería AR sesión, en este objeto debemos añadir componentes como lo son el Arplane manager, el AR anchor manager, el ar raycast manager.

Figura 22. Opción para adición de componente

Todos estos componentes cuentan con parámetros iniciales para su funcionamiento los cuales se deben ajustar a través del Script dependiendo de las necesidades que tenga el usuario, en el caso actual se configuro para la detección de planos horizontales donde se pudiera ubicar la planta esto usando el ar plane manager donde se añade un script , además, este se mezclará con un mapeo por puntos el cual mejorara la superficies, esto va a generar una lista de posibles superficies, esta lista servirá para tomar decisiones de donde posicionar.

Figura 23. Posicionamiento de objetos referenciados a un mapa de puntos

```

if(isinitialposition)
{
    Vector2 middle = new Vector2(Screen.width / 2, Screen.height / 2);
    aRRaycastManager.Raycast(middle, hits, TrackableType.Planes);

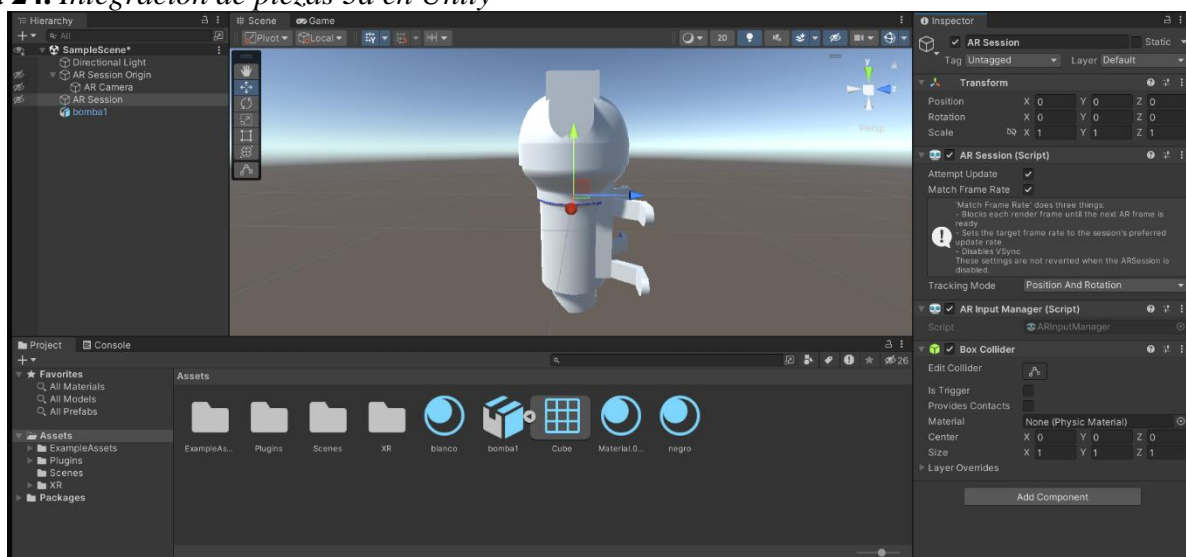
    if(hits.Count>0)
    {
        transform.position = hits[0].pose.position;
        transform.rotation = hits[0].pose.rotation;
        arpointer.SetActive(false);
        isinitialposition = false;
    }
}

```

Teniendo la lista de hits en las cuales se puede situar el modelo se recorre los planos buscando cual cumple la característica de ubicarse centrado en la pantalla, para obtener la posición de la pantalla se pueden extraer de Screen el cual esta referenciado a la AR camera. Teniendo esta configuración podemos situar un Prefab en realidad aumentada.

4.1.3.2 Integración de modelos de piezas 3D a Unity. Para importar modelos a unity se tienen que agregar el proyecto teniendo en cuenta de que estos modelos deberán estar en el tipo FBX luego de estar añadidos podrán ser agregados a el árbol de proyectos de la escena, desde ahí podrá añadirles componentes y propiedades a los materiales de este, además que serán visibles en la escena inicialmente

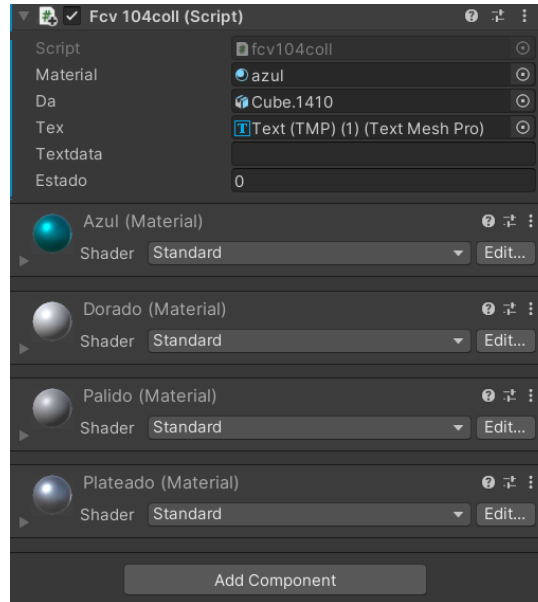
Figura 24. Integración de piezas 3d en Unity



Para poder detectar cuando un usuario toca un objeto es importante que el objeto cuente con un BOXCOLLIDER, esta propiedad permite detectar colisiones con otros objetos generados

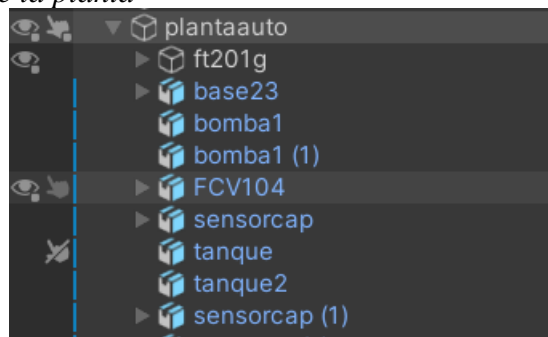
en la escena. Teniendo las propiedades y los componentes definidos se añade un script a el objeto para poder modificar cualquiera de estos.

Figura 25. *Script con objetos de la escena*



4.1.3.2 Ensamblaje en Unity. Es importante incluir todos los objetos iniciales en un objeto padre, esto con el fin de facilitar la modificación de la posición general del objeto o escala.

Figura 26. *Árbol de objetos de la planta*



Para empezar, ubicamos el objeto base en las posiciones XYZ (0,0,0) de esta manera queda referenciado a el origen, los modelos que son repetitivos como las tuberías y los codos se crea un prefabricado para poder añadir varios a las escenas sin tener que volver a configurarlos luego de tener todas las conexiones y piezas en la planta se obtiene como resultado.

Figura 27. *Visualización de planta de automatización.*

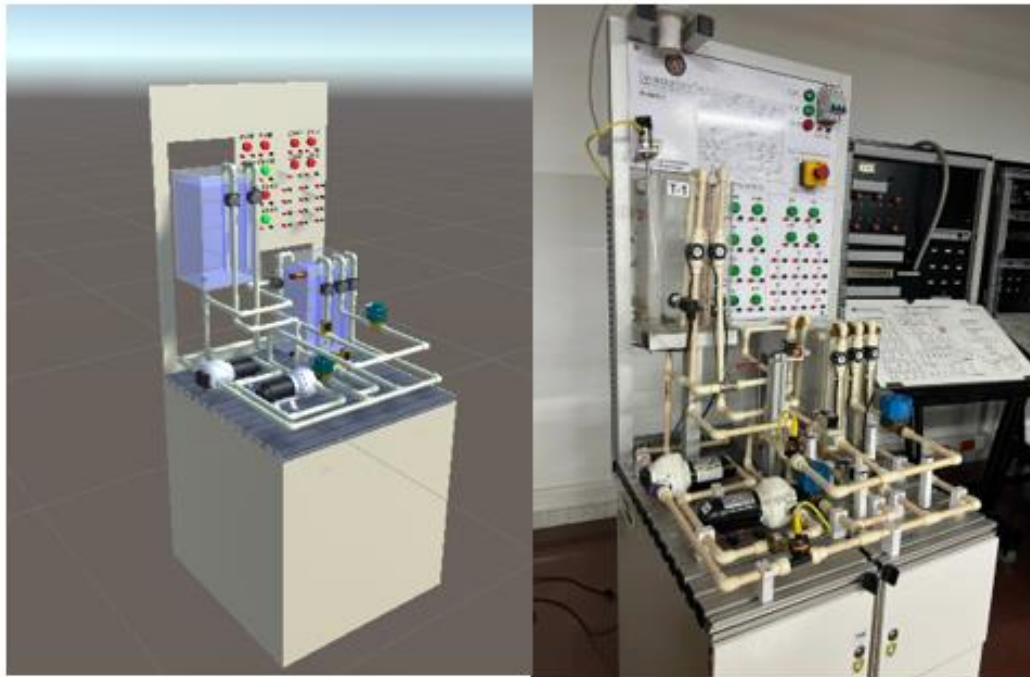
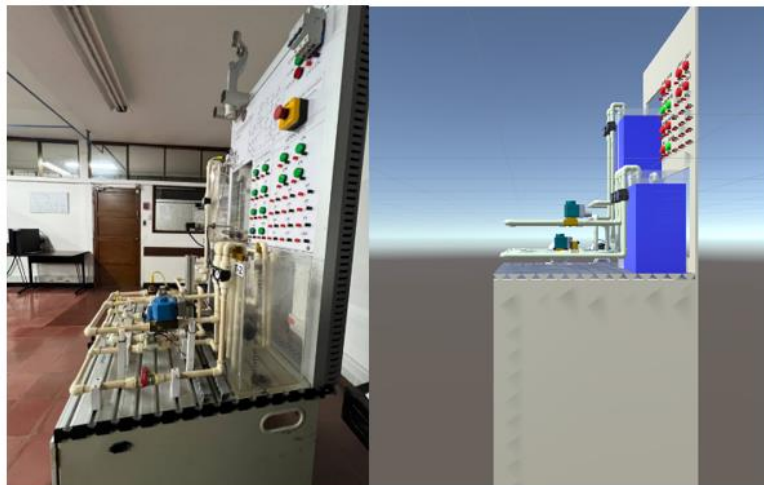
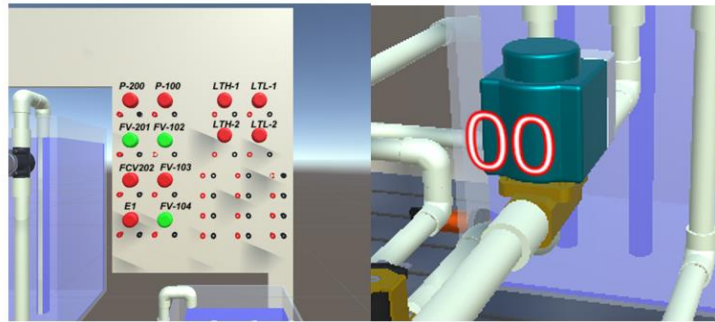


Figura 28. *Visualización lateral de planta digital*



4.1.3.3 Visualización de la información. El modelo cuenta con válvulas proporcionales y sensores de flujo las cuales se controlan con una variable tipo int, sensores capacitivos válvulas solenoides y bomba de agua las cuales solo cuentan con dos estados por lo cual son variables booleanas , para visualizar estos datos en el modelo se añaden objetos tipo texto los cuales se modifican desde el código accediendo a la propiedad TEXT, en cuanto a las variables de tipo booleana se cuenta con un tablero de led en el cual se modifica las texturas dependiendo del estado.

Figura 29. Visualización de datos.



El caso específico del nivel del tanque se implementó una textura que cambia su escala en el eje y simulando el nivel para esto se debe incluir un script el cual accede a una memoria global que contiene el nivel del agua y lo escala a la referencia, esta memoria global esta creada en un objeto de unity diseñado para poder acceder a él sin ser referenciado, solo con el nombre del objeto y una función pública se puede acceder a esa memoria.

Figura 30. Creación de memoria global

```

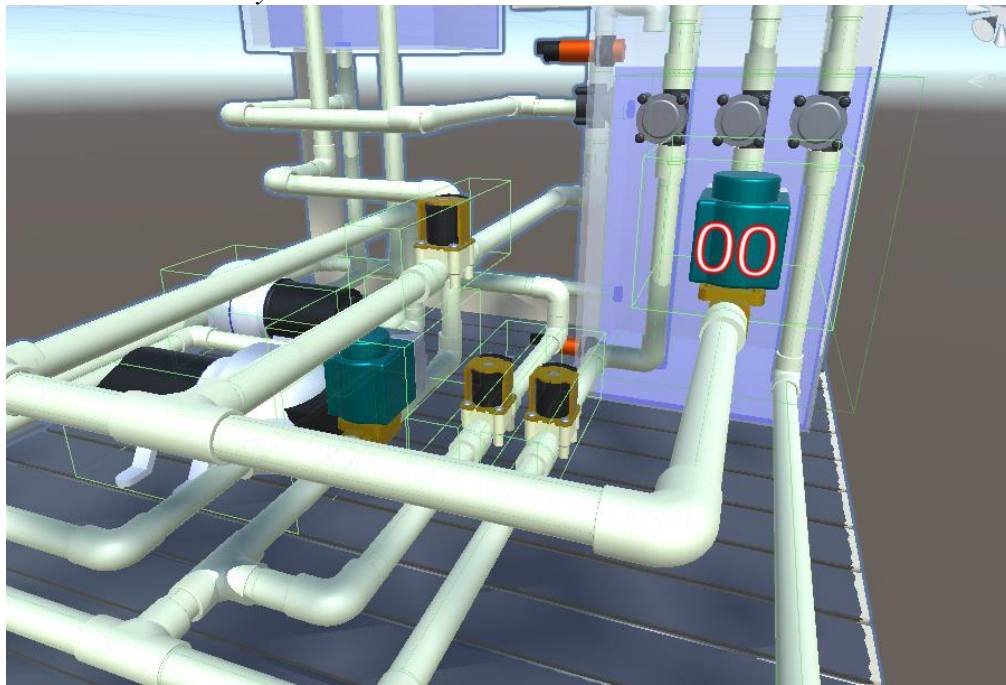
void Update()
{
    int data = nodestruir.Instance.nivel;
    if (data != data2)
    {
        tex.text = data.ToString();

        transform.localScale = new Vector3(19, data/10, 19);
        data2 = data;
    }
}

```

4.1.3.4 Detección de colisiones con objetos en la RA. Detectar colisiones en unity se puede realizar de varias maneras, pero en el caso de aplicaciones móviles se usa el raycast el cual es un generador de rayos que salen de un punto específico y mantienen una dirección vectorial, este rayo interactúa a su paso con los objetos que se encuentra, pero para interactuar los objetos tiene que incluir alguna propiedad de collider la cual se puede usar con diferentes figuras y tamaños.

Figura 31. *Colliders en Unity*



Nota: Los cuadros de color verde en la figura 31 son las colisiones.

Para determinar la posición y la dirección en la que saldrá el rayo generado por raycast se tomó el punto que nos proporción el touch, para acceder a el touch se debe tener referenciado el objeto ARcamera y se accederá atreves de `Input.touches()`, luego de haber generado los rayos se procede a encontrar cuales objetos colisionaron con ellos, el raycast permite cambiar propiedades o ejecutar funciones del objeto en el que colisiona como se muestra a continuación.

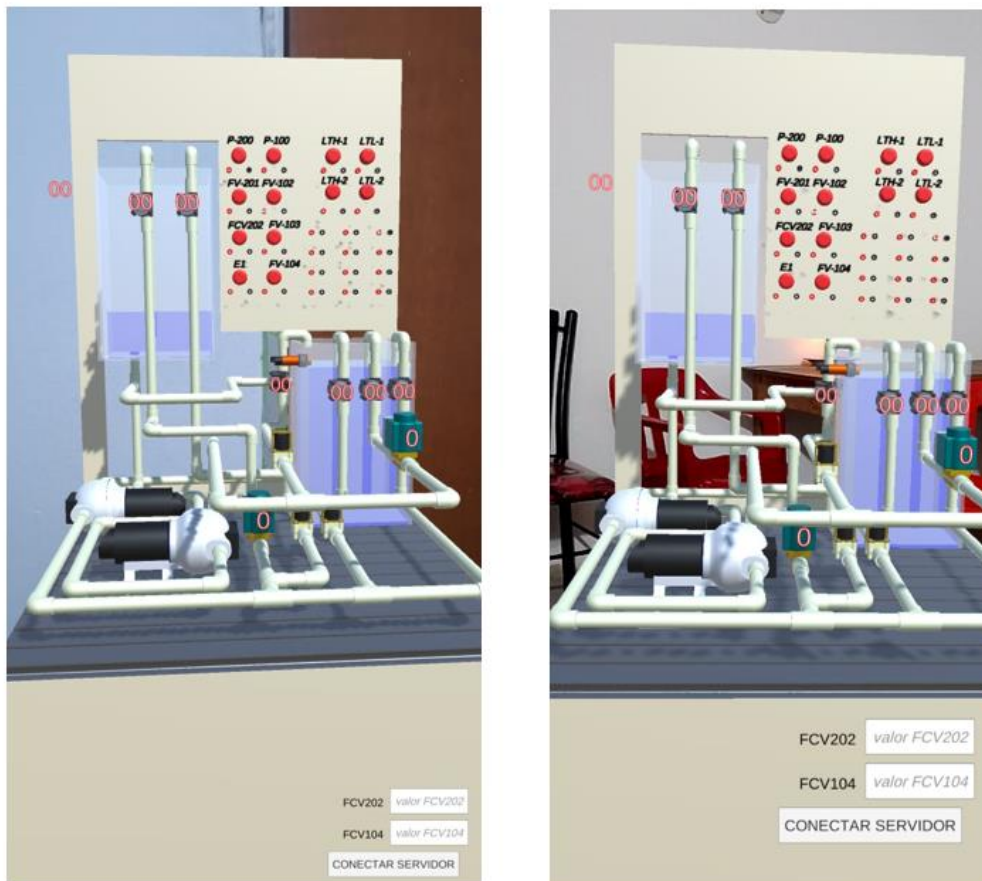
Figura 32. Pruebas

```

if (Input.touchCount > 0 && Input.touches[0].phase == TouchPhase.Began)
{
    Ray ray = arcamera.ScreenPointToRay(Input.touches[0].position);
    RaycastHit hit;
    if (Physics.Raycast(ray, out hit))
    {
        if (hit.collider != null)
        {
            Color newcolor = Random.ColorHSV();
            hit.collider.GetComponent<MeshRenderer>().material.color = Color.red;
            //material2.color = Random.ColorHSV();
        }
    }
}
// Start is called before the first frame update

```

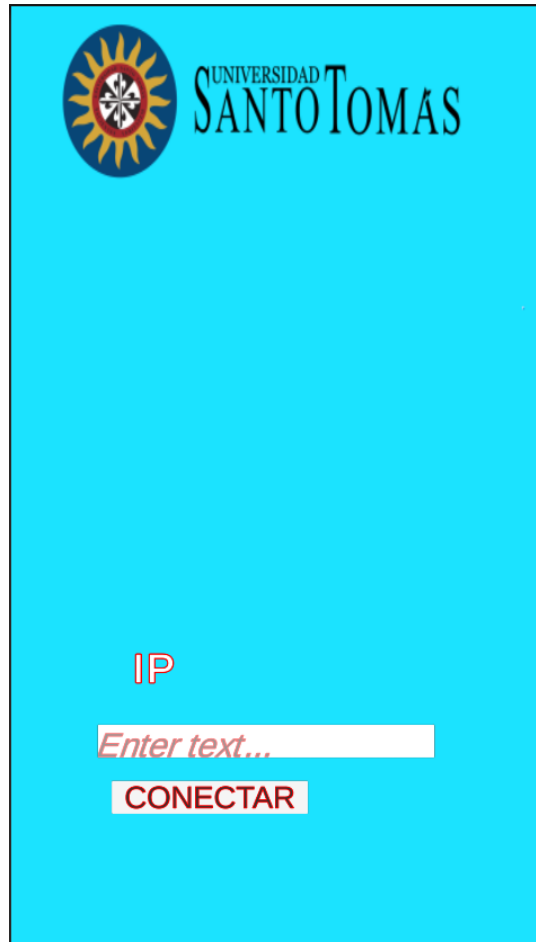
En la siguiente figura se podrá observar cuando se realizan pruebas de posicionamiento de la planta desde diferentes entornos.

Figura 33. Posicionamiento de la planta

Nota: Son pruebas realizadas desde diferentes entornos y distintos dispositivos móviles.

La interfaz inicial de la aplicación es sencilla ya que se le facilitará el uso de esta a cualquier usuario porque no tiene complejidad al momento comprender como funciona.

Figura 34. *Interfaz inicial*



4.1.3.5 Problemas. En el desarrollo de la etapa 3, se presentaron los siguientes problemas:

- El uso de raycast detecta las colisiones pero en este caso en específico no permitió cambiar valores de funciones ni de variables del objeto, pero si permitía cambiar propiedades físicas del objeto como tamaño, posición o escala por lo que se implementó un método en el cual se cambia el color del objeto por un color en específico que no se use en ninguna parte del objeto, teniendo esta propiedad modificada se usa una comparación para saber si el objeto

cambio de color y así ejecutar una función, después de ejecutar la función se devuelve el color original a el material como se evidencia en la siguiente figura.

Figura 35. *Implementación de método*

```
color = da.GetComponent<MeshRenderer>().material.color;

//Debug.Log(color);
if (color == Color.red)
{
    string aux = "FCV202:" + textdata;
    nodestruir.Instance.enviar(aux);
    da.GetComponent<MeshRenderer>().material.color = material.color;
}
}
```

- Los servicios de ARCore no se encuentran disponibles en todos los dispositivos móviles Android ya que estos deben ser compatibles con ARCore, sin embargo, si está disponible en la gran mayoría por lo que se debe verificar antes de usar.

4.1.4 Etapa 4

En esta etapa se desarrollan los protocolos necesarios para transmitir la información desde el aplicativo móvil a el plc físico el cual controla las plantas piloto.

Figura 36. *Proceso de transmisión de información*



4.1.4.1 Configuración de Web socket. Para la configuración del web socket es necesario usar el entorno de node js, además, de un módulo de llamado “WS”. Para la instalación del módulo ws se tiene que ejecutar en la terminal el siguiente comando:

Figura 37. *Comando de instalación del “WS”*

```
PS C:\Users\Usuario\Desktop\nodejsss> npm install ws
```

Luego de tener el módulo” WS” añadido, se configura parámetros como la configuración IP, posteriormente, se determina el puerto a usar teniendo en cuenta que esté disponible (en este caso se utiliza el puerto 8080) y se establece el servidor.

Figura 38. *Creación del servidor en Node.js*

```
const WebSocket = require('ws')
const wss = new WebSocket.Server({ port: 8080 }, ()=>{
  console.log('server started')
})
```

Se configura el web socket para recibir y enviar información a todos los clientes que se conecten.

Figura 39. *forEach*

```
wss.on('connection', function connection(ws) {
  console.log("A client just connect: ");
  ws.on('message', (data) => {
    console.log("data received: " +data)
    wss.clients.forEach((client)=>{
      client.send(''+data);
    })
  })
})
wss.on('listening', ()=>{
  console.log("Server Listening!!")
})
```

Para poder ejecutar este código se utiliza la instrucción node en la terminal junto con el nombre del archivo.

Figura 40. Ejecución del código y conexión del cliente

```
PS C:\Users\Usuario\Desktop\nodejsss> node index.js
server started
Server Listenning!!
A client just connect:
```

El servidor se establece en la red local, lo cual permite a clientes que estén la misma red conectarse a esta. Al no contar con una IP pública no se puede establecer conexión desde un sitio remoto. Para solucionar esto, se recurre a la creación de una máquina virtual en Google cloud, esta proporciona una dirección IP pública y permite que el servidor esté en ejecución a tiempo completo. La máquina virtual se crea con el sistema operativo de Ubuntu.

Figura 41. Configuración de la máquina virtual

Disco de arranque ?

Nombre	instance-4
Tipo	Disco persistente balanceado nuevo
Tamaño	50 GB
Tipo de licencia ?	PAYG (pago por uso)
Imagen	Ubuntu 18.04 LTS Pro Server

CAMBIAR

Al crear la máquina virtual se va a generar una IP pública y se podrá configurar el servidor a través de SSH

Figura 42. Dirección IP pública.

<input type="checkbox"/>	Estado	Nombre ↑	Zona	Recomendaciones	En uso por	IP interna	IP externa	Conectar
<input type="checkbox"/>	<input checked="" type="checkbox"/>	instance-3	southamerica-west1-a			10.194.0.4 (nic0)	34.176.155.44 (nic0)	SSH ▾

Figura 43. Protocolo SSH

```

SSH en el navegador
SUBIR ARCHIVO  DESCARGAR ARCHIVO  !  ☰  ⚙️

Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-1115-gcp x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Mon Oct 23 02:48:16 UTC 2023

System load:  0.0          Processes:    173
Usage of /:   10.4% of 48.27GB  Users logged in:  0
Memory usage: 5%          IP address for ens4: 10.194.0.4
Swap usage:  0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.
   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Infrastructure is enabled.

0 updates can be applied immediately.

New release '20.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sun Oct 22 23:59:35 2023 from 35.235.241.32
luisantonioespinelbustos@instance-3:~$

```

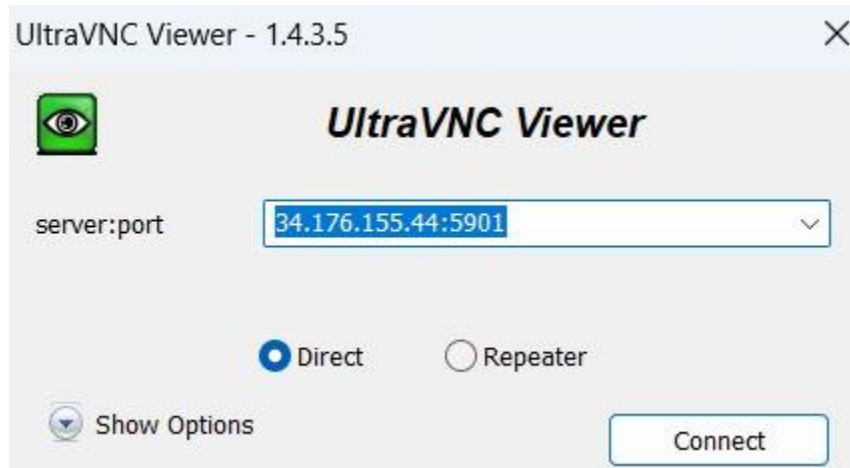
Debido a que el protocolo SSH no es muy manejable para el usuario, se opta por configurar la máquina virtual para lograr acceder de forma remota a un escritorio, para esto se deben ejecutar los siguientes comandos:

- Sudo apt update
- Sudo apt upgrade
- Sudo adduser demo
- Sudo usermod -aG sudo,adm demo
- Sudo apt install xfce4 xfce4-goodies
- Sudo apt install tightvncserver
- Sudo apt install irefox

- Vncserver

Después de ejecutar los comandos se podrá acceder remotamente a través de un escritorio remoto, en este caso UltraVNC.

Figura 44. *UltraVNC.*



Luego de ingresar la dirección IP, se tendrá acceso al escritorio remoto, en este se podrá instalar a través de la terminal Node.js se deberá ejecutar los siguientes comandos:

- `curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.2/install.sh | bash`
- `source ~/.bashrc`
- `nvm install 16`

Tras haber realizado los pasos anteriores, se deben repetir los pasos que se encuentran desde la página 67 hasta la página 99 en la máquina virtual, obteniendo como resultado un websocket al que se pueda acceder desde cualquier sitio.

Figura 45. Máquina virtual ejecutando el websocket.

```

{} package-lock.json  Js index.js x
Js index.js > ...
6  }
7  wss.on('connection',function connection(ws)
8  {
9      console.log("A client just connect:");
10     ws.on('message',(data)=>{
11         console.log("data received: "+data)
12         wss.clients.forEach((client)=>{
13             client.send(' '+data);
14         })
15     })
16 })
17 wss.on('listening',()=>{
18     console.log("server listening")
19 })

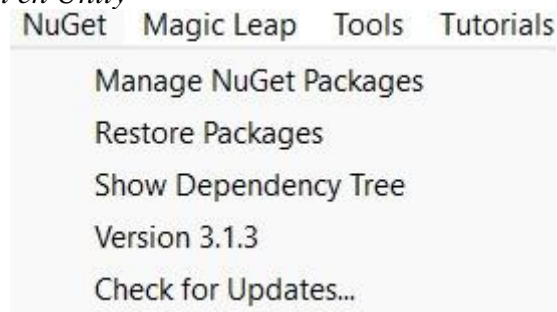
```

```

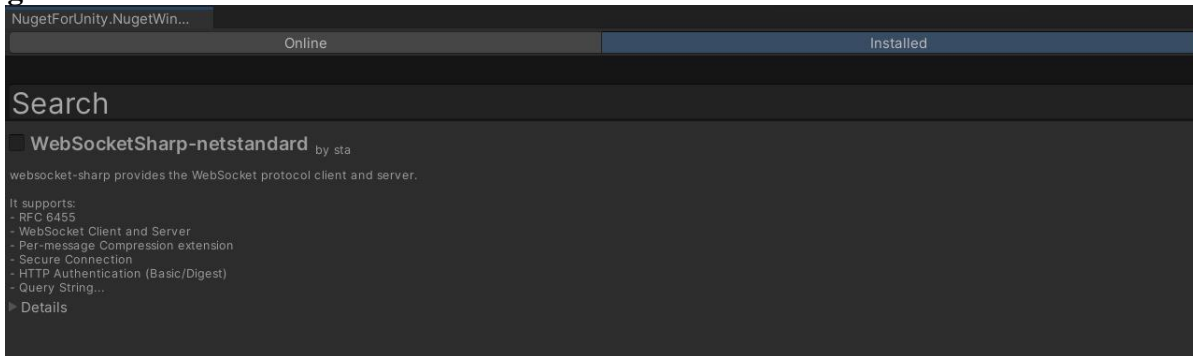
A client just connect:
data received: 15
data received: 29
data received: 5
data received: 15
data received: 29
^C
o luisantoniospelbustos@instance-3:~/Desktop$ node index.js
server started
server listening
A client just connect:
data received: 29

```

4.1.4.2 Web socket en unity. Para que Unity se pueda conectar como un cliente al Websocket, se utiliza WebSocket Sharp, el cual es una package de NuGet.

Figura 46. Menú de NuGet en Unity

Buscar en la librería WebSocketSharp – netstandard e instalar.

Figura 47. *Biblioteca de NuGet.*

Para establecer Unity como un cliente se debe generar un script en el cual se incluya la librería Websocket, con esto se tendrá acceso a la clase websocket mediante la cual se podrá conectar al servidor, a su vez, enviar y recibir información.

Figura 48. *Librería WebSocketSharp*

```
using WebSocketSharp;
```

Para conectarse al servidor se debe conocer la IP y el puerto por donde se va a comunicar.

Figura 49. *Estructura base del cliente en Unity.*

```
string data = nodestruir.Instance.ip2();
data = "ws://" + data + ":3389";
Debug.Log(data);
ws = new WebSocket(data);

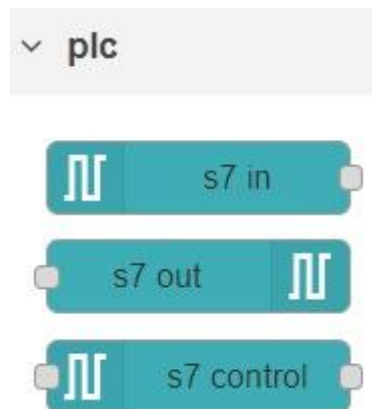
ws.Connect();
ws.OnMessage += (sender, e) =>
{
    Debug.Log("Message Received from " + ((WebSocket)sender).Url + ", Data : " + e.Data);
    data2 = e.Data;
    bandera = false;
};
```

Figura 50. Prueba de la conexión.

4.1.4.3 Node-Red. NodeRed es un entorno que a través de nodos permite la interacción entre diferentes protocolos de comunicación. En el proyecto actual se quiere interactuar con los PLCs de Siemens S7, para esto fue necesario instalar la librería “node-red-contrib-s7”.

Figura 51. Instalación de librería node-red-contrib-s7

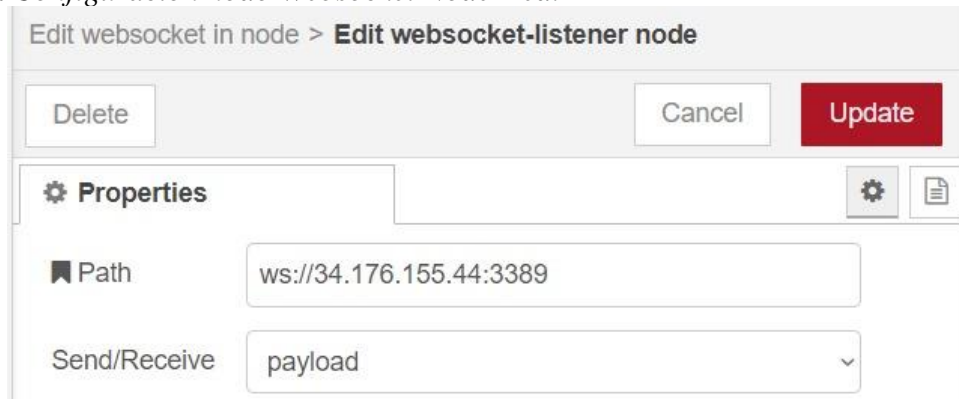
Después de completar la instalación, se habilitarán tres nodos.

Figura 52. Nodos s7

El primer paso para usar Node-Red, es ejecutar una terminal CMD y usar el comando node-red, esto generará un servidor que correrá en la dirección “http://127.0.0.1:1880/” , en la cual si se accede a ella se podrá configurar todos los nodos disponibles en Node-Red. El primer

paso para recibir información del Web Socket configurar el nodo llamado “websocket in”, hay que añadir la dirección IP al Websocket y el modo de configuración de recepción.

Figura 53. Configuración nodo Websocket Node-Red.



Edit websocket in node > **Edit websocket-listener node**

Delete Cancel Update

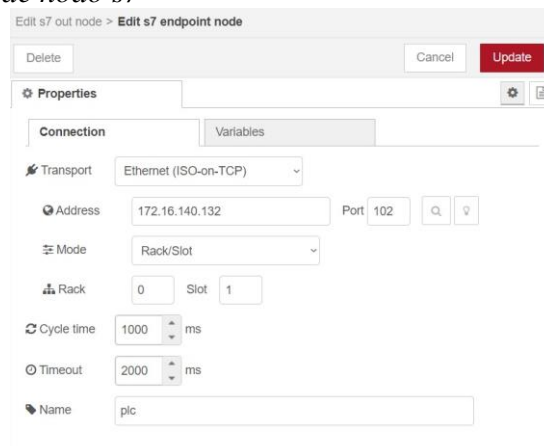
⚙️ Properties 📄

📄 Path ws://34.176.155.44:3389

Send/Receive payload ▾

En la configuración del nodo S7, se debe seleccionar el tipo de transporte, en este caso Ethernet TCP, la dirección la cual corresponde a la IP que tenga configurada el PLC, el rack y el slot se tomarán de la configuración del dispositivo en “TiaPortal”, el ciclo de tiempo de ejecución dependerá de las necesidades del usuario.

Figura 54. Configuración de nodo s7



Edit s7 out node > **Edit s7 endpoint node**

Delete Cancel Update

⚙️ Properties 📄

Connection Variables

📄 Transport Ethernet (ISO-on-TCP)

📄 Address 172.16.140.132 Port 102 🔍 🗑️

📄 Mode Rack/Slot

📄 Rack 0 Slot 1

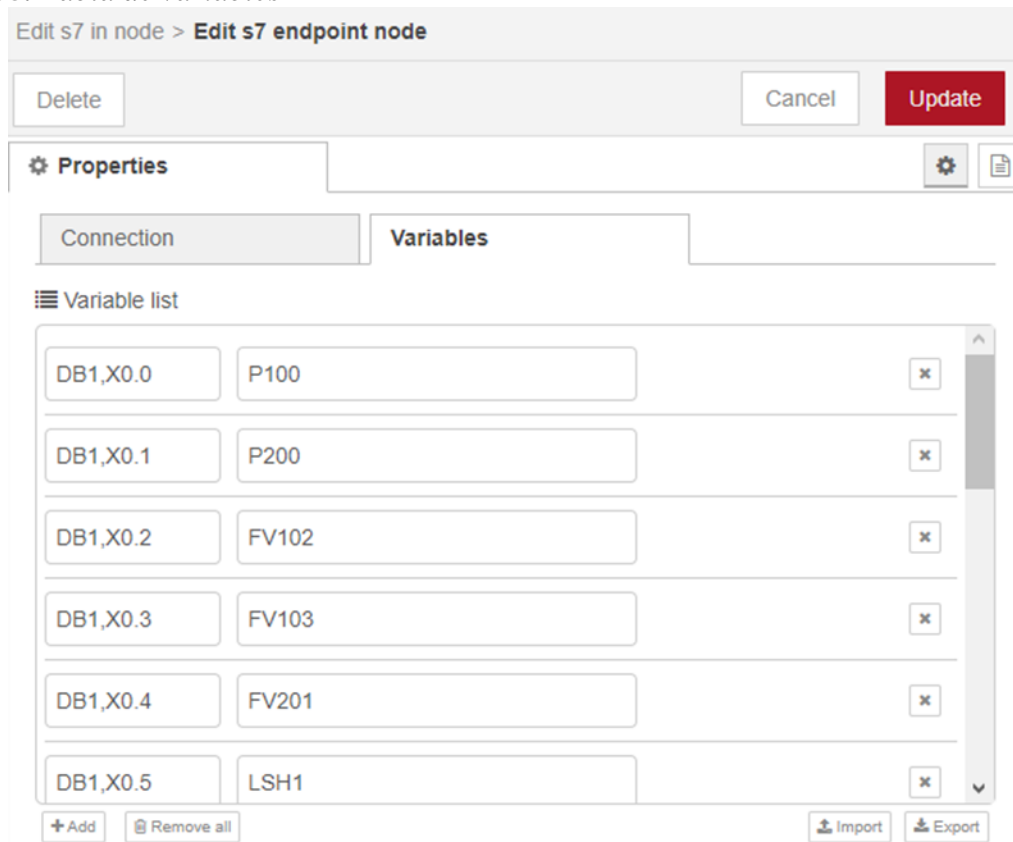
📄 Cycle time 1000 ms

📄 Timeout 2000 ms

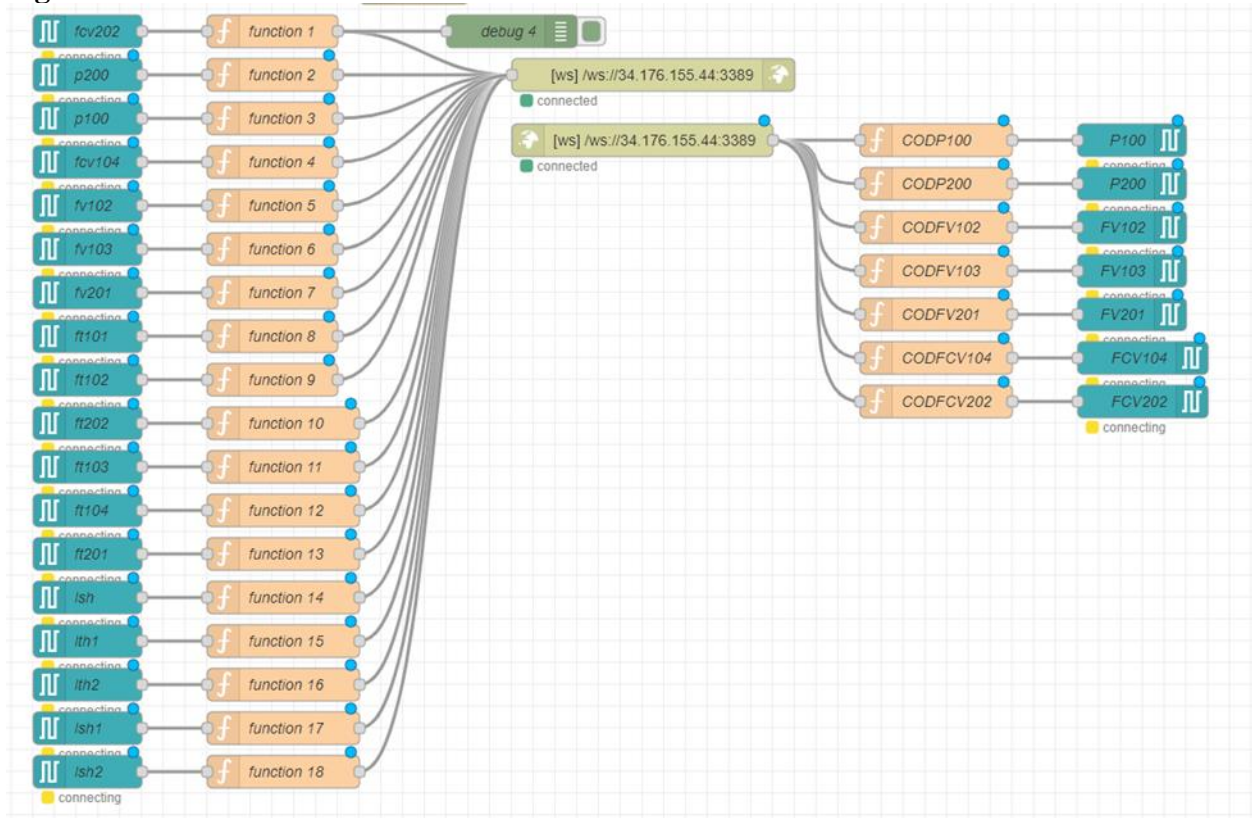
📄 Name plc

Para poder acceder a las variables del PLC se debe configurar con una nomenclatura que depende del tipo de variable, esta nomenclatura se podrá consultar en el siguiente enlace: [node-red-contrib-s7 \(node\) – Node-RED \(nodered.org\)](https://nodered-contrib-s7.github.io/).

Figura 55. *Tabla de variables*

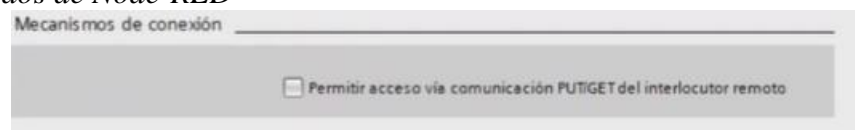


Luego de tener la table de variables definidas se debe generar cada nodo como entrada y cada nodo en salida, en esta casa tendremos 20 nodos de entradas que son las variables que podemos leer del dispositivo y de salida tendremos siete que son las variables que podemos modificar para su funcionamiento.

Figura 56. Nodos de Node-RED

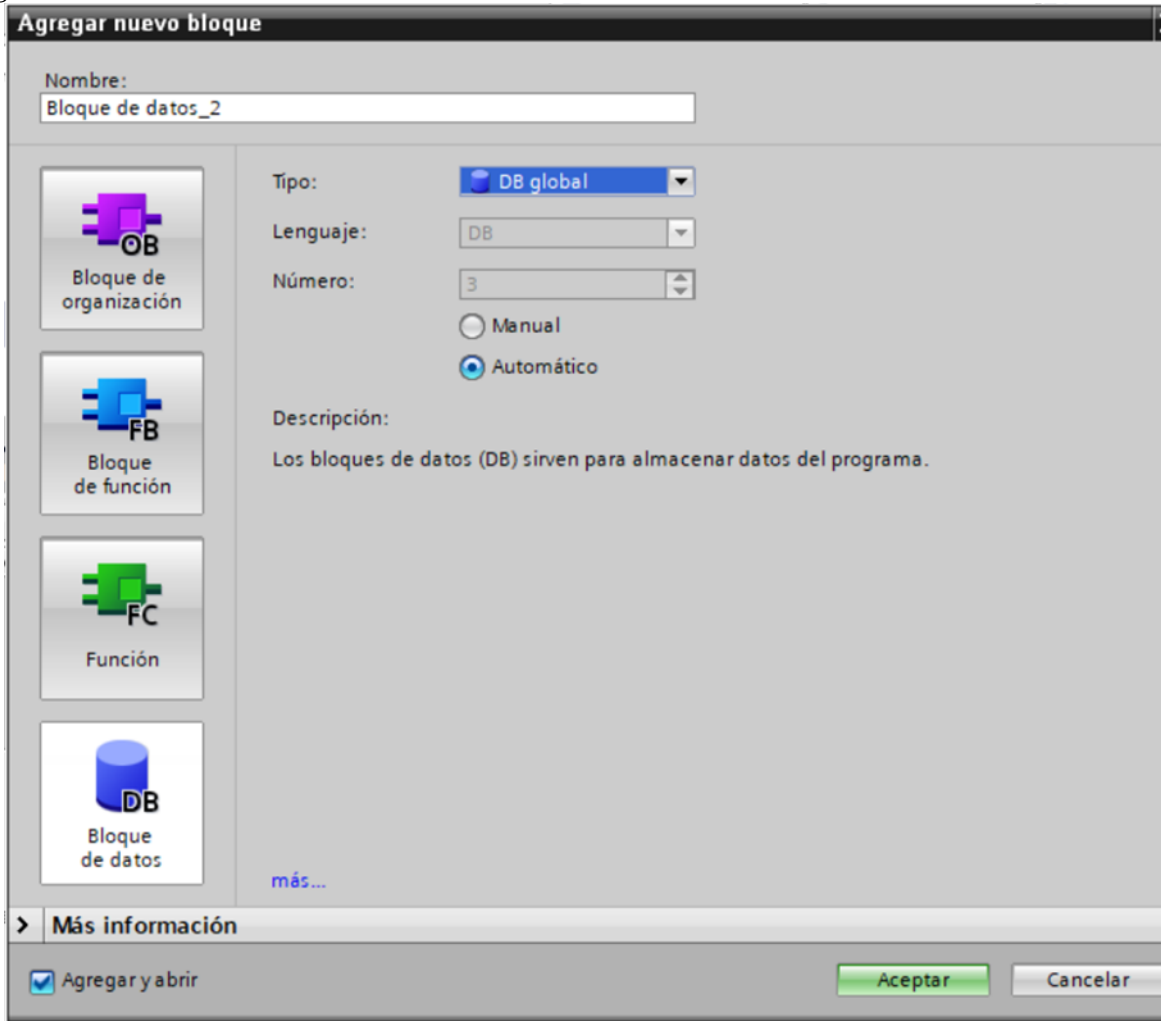
4.1.5 Etapa 5

4.1.5.1 Configuración PLC. Para el correcto funcionamiento de la librería “node-red-contrib-s7” en tia portal se le deben aplicar una configuración inicial, esto se encuentra en las propiedades del plc donde se debe activar la opción de permitir el acceso vía comunicación PUTGET del interlocutor remoto

Figura 57. Nodos de Node-RED

Después de tener esta opción se tiene que crear una base de datos.

Figura 58. Creación de base de datos



Teniendo la base creada de datos se le debe asignar un nombre y un tipo a la variable, este nombre nos permitirá se referenciado como una variable en el programa de Ladder.

Figura 59. *Tabla de variables*

	Nombre	Tipo de datos
1	Static	
2	P100	Bool
3	P200	Bool
4	FV102	Bool
5	FV103	Bool
6	FV201	Bool
7	LSH1	Bool
8	LSH2	Bool
9	LTH1	Bool
10	LTH2	Bool
11	LSH	Int
12	FCV104	Int
13	FCV202	Int
14	FT101	Int
15	FT102	Int
16	FT202	Int
17	FT103	Int
18	FT104	Int
19	FT201	Int

Teniendo las variables definidas tenemos que compilar el bloque, al compilar el bloque nos generara las ubicaciones para poder acceder desde remoto a las memorias configuradas.

Figura 60. *Pruebas del PLC*

Tipo de datos	Offset	Valor de arranq...	Valor de observación	Remanen...	Accesible d...	Es
Bool	0.0	false	TRUE			
Bool	0.1	false	FALSE			
Bool	0.2	false	TRUE			
Bool	0.3	false	FALSE			
Bool	0.4	false	TRUE			
Bool	0.5	false	FALSE			
Bool	0.6	false	FALSE			
Bool	0.7	false	FALSE			
Bool	1.0	false	FALSE			
Int	2.0	0	90			
Int	4.0	0	25			
Int	6.0	0	0			
Int	8.0	0	0			
Int	10.0	0	0			
Int	12.0	0	0			
Int	14.0	0	0			

Ventana Ayuda
 Establecer conexión online Deshacer conexión online
 AGO0811y Bloques de programa Bloque de datos_1 [D01]
 Intervar valores actuales Instantánea Copiar instantáneas a valores de arranque
 Propiedades Información
 través de la dirección IP=172.16.140.118.
 rido correctamente.
 27/10/2023 14:11:45
 27/10/2023 15:22:18
 27/10/2023 15:22:22
 27/10/2023 15:22:22
 27/10/2023 15:22:25
 27/10/2023 15:22:26
 Main (DR1) PLC_1

Figura 61. *Tabla de variables*

Bloque de datos_1									
	Nombre	Tipo de datos	Offset	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ..	Valor de a..
1	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	P100	bool	0.0	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	P200	bool	0.1	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	FV102	bool	0.2	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	FV103	bool	0.3	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	FV201	bool	0.4	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	LSH1	bool	0.5	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	LSH2	bool	0.6	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	LTH1	bool	0.7	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	LTH2	bool	1.0	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
11	LSH	nt	2.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
12	FCV104	nt	4.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
13	FCV202	nt	6.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
14	FT101	nt	8.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
15	FT102	nt	10.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
16	FT202	nt	12.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
17	FT103	nt	14.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
18	FT104	nt	16.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
19	FT201	nt	18.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

4.2. Simulaciones-pruebas- caracterización

4.2.1 Pruebas

Estando configurados los diferentes dispositivos se debe realizar pruebas de como enviar datos entre ellos, la primera prueba realizada es el envío de un dato del gemelo a el web socket, para enviar datos basta con usar el comando `ws.Send`.

Figura 62. *Uso del comando `ws.Send`*

```

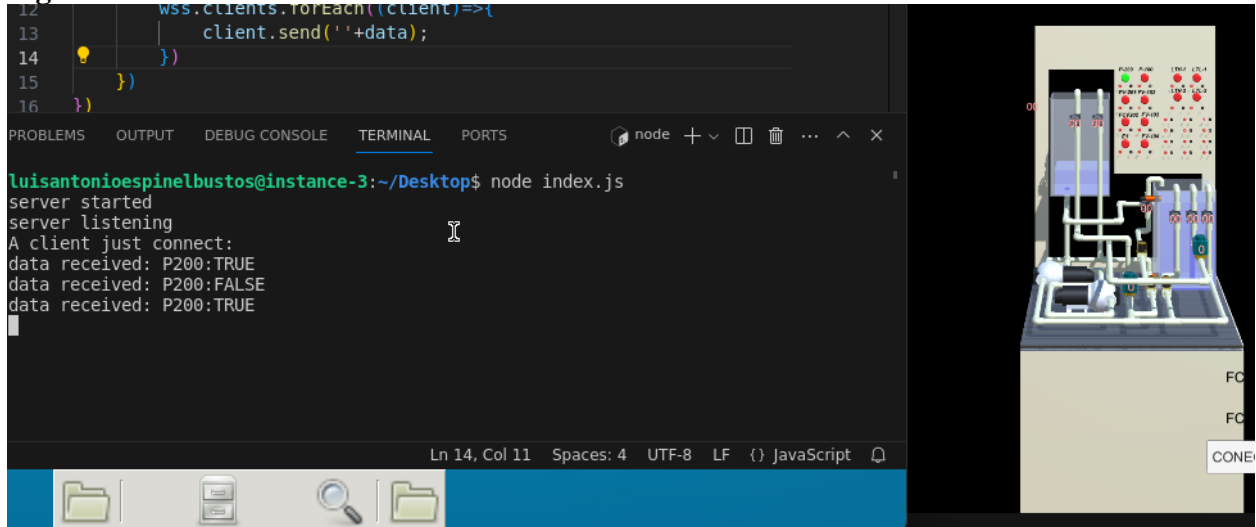
7 referencias
public void enviar(string datico)
{
    ws.Send(datico);
}
1 referencia

```

Podemos comprobar que se envió el mensaje revisando la salida de la consola del websocket. Los mensajes se estructuraron de la siguiente forma, la primera parte del mensaje hace

referencia a la variable a modificar y la segunda parte es el valor, estos irán separados por un “:” el cual nos permite hacer un Split a la hora de manejar la información.

Figura 63. *Tabla de variables*

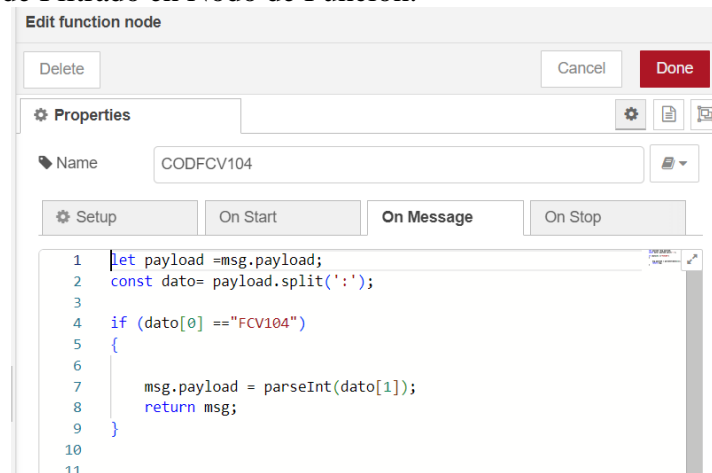


The image shows a code editor window with a JavaScript file named `index.js`. The code defines a WebSocket server that listens for connections and sends data back to clients. The terminal output shows the server starting and listening, followed by three client connections with data received: `P200:TRUE`, `P200:FALSE`, and `P200:TRUE`. To the right of the terminal is a 3D model of an industrial plant with various pipes, tanks, and control panels. The model is labeled with 'FC' and 'CONEX'.

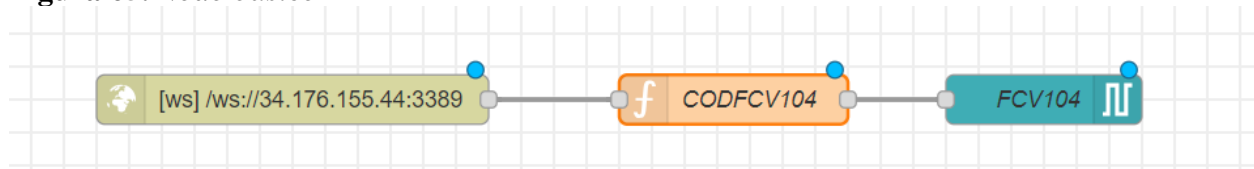
```
12 wss.clients.forEach((client)=>{
13   client.send(''+data);
14 })
15 })
16 }
```

```
luisantonioespinelbustos@instance-3:~/Desktop$ node index.js
server started
server listening
A client just connect:
data received: P200:TRUE
data received: P200:FALSE
data received: P200:TRUE
```

Con el nodo de websocket in podemos recibir los mensajes que llegaron del cliente y con el websocket out se puede enviar a el servidor, teniendo el nodo se deber realizar un filtrado para obtener la información útil, esto se puede desde un nodo llamado función el cual se puede programar funciones en JavaScript como se observa en la siguiente figura:

Figura 64. Proceso de Filtrado en Nodo de Función.

La función separa los datos y los retorna con la información útil, cada nodo ira conectado a el nodo de salido que está configurado en específico con esa variable, la función al comprobar el nombre de variable no permitirá pasar si no el dato correspondiente a la misma.

Figura 65. *Nodo básico*

Replicando esto, pero con el nodo de entrada enviamos información atreves del websocket, esta vez tendremos que manejar la información en unity, para esto usamos la misma estructura.

Figura 66. *Envío de información mediante el websocket*

```
Debug.Log("Message Received from " + ((WebSocket)sender).Url + ", Data : " + e.Data);
data2 = e.Data;

bandera = false;
string[] words;
words = e.Data.Split(':');
Debug.Log(words[0]);
```

Ahora contamos con un string con dos componentes una la cual será el nombre de la variable y la otra será la información útil, esto se filtra con if y se le asignan a una variable guardada en un objeto el cual se puede acceder desde los diferentes scripts solo citándolo.

Figura 67. Código para acceso a datos compartidos entre scripts.

```
if (words[0] == "FV102")
{
    if (words[1] == "TRUE")
    {
        fv102 = true;
    }
    else
    {
        fv102 = false;
    }
}
```

Para verificar el correcto funcionamiento de la versión final del gemelo digital, primero, se debe conectar las salidas analógicas con la variable que se quiere controlar y conectar las entradas analógicas y digitales que se quieren observar, teniendo en cuenta la tabla de variables.

Figura 68. Tablero de conexiones del PLC y la planta piloto.



Para que el PLC lea o escriba correctamente las variables analógicas en la configuración de TIA portal se deben añadir los módulos correspondientes de AQ y AI, a su vez, definir las memorias por las cuales se pueden acceder a ellas.

Figura 69. Configuración de variables analógicas.

Módulo	Slot	Dirección I	Dirección Q
PLC_1	1		
DI 14/DQ 10_1	1 1	0...1	0...1
AI 2_1	1 2	64...67	
AQ 1x12BIT_1	1 3		80...81
HSC_1	1 16	1000...10...	
HSC_2	1 17	1004...10...	
HSC_3	1 18	1008...10...	
HSC_4	1 19	1012...10...	
HSC_5	1 20	1016...10...	
HSC_6	1 21	1020...10...	
Pulse_1	1 32		1000...10...
Pulse_2	1 33		1002...10...
Interfaz PROFINET_1	1 X1		
	2		
	3		

Propiedades de AQ 1x12BIT_1 [AQ1 signal board]

Canal0

Dirección de canal: QW80

Tipo de salida analógica: Tensión

Rango de tensión: +/-10 V

Valor sustitutivo para canal en caso de transición de RUN a S...: 0.000 V

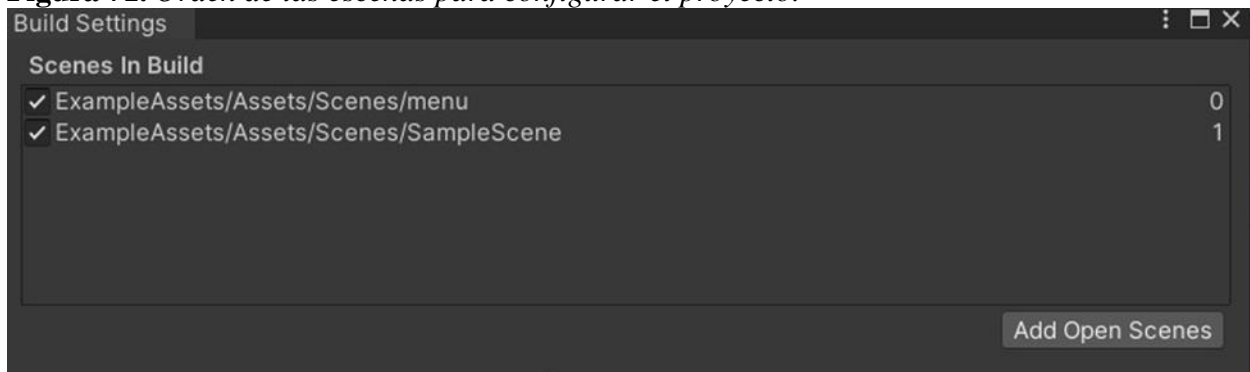
Luego de realizar la configuración de las variables analógicas se tienen que normalizar y escalar los datos para su adecuada visualización y control.

Figura 70. Ejemplo de la normalización de datos

4.2.2 Modificación del proyecto de Unity.

Haciendo pruebas de modificación del aplicativo en otros dispositivos se encontró que al abrir el proyecto, genera errores de compatibilidad y accesibilidad a los assets, por lo que después de varias pruebas se halló el método para poder mejorar el proyecto, los pasos para llevar a cabo la modificación del proyecto son los siguientes:

- Crear un proyecto vacío de ARcore 3d.
- Seleccionar la versión 2022.3.12f1
- Luego, se añade la carpeta assets a la carpeta assets del proyecto actual.
- Asegurarse de configurar el orden ejecuciones en escenas.

Figura 71. Orden de las escenas para configurar el proyecto.

De esta manera el proyecto se podrá modificar y compilar desde cualquier editor de unity permitiendo que se puedan realizar mejoras a futuro o actualizaciones.

4.2.3 Pruebas de estabilidad.

Comprobando la latencia en la llegada de paquetes desde la aplicación móvil hasta node-red se obtuvieron resultados de una media de 270 ms de latencia a la hora de recibir paquetes de la aplicación y de 170 ms al envío del servidor a la aplicación, para obtener estos resultados se realizaron mas de 5 pruebas en diferentes sitios del país siendo algunos casos de ellos Málaga, San Gil, Bogotá y Bucaramanga.

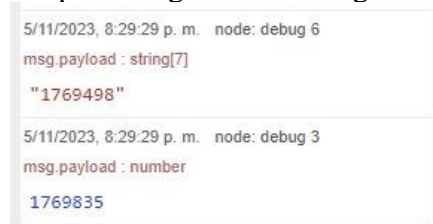
Figura 72. Probando la aplicación en San Gil, Santander.



Para realizar la prueba se tomó el tiempo actual de ejecución del envío y se envió por el web socket y la hora de llegada se toma el tiempo en que llego para esto se usa una función en node-red que a través de la función date now se obtiene el tiempo actual, teniendo estos dos valores se realiza una resta de los tiempos en milisegundos y con la diferencia que obtenemos se sabe la

latencia en milisegundos como se observara en la siguiente figura, estas pruebas se realizaron varias veces en las diferentes ubicaciones.

Figura 73. Tiempo de envío y tiempo de llegada en milisegundos.



```
5/11/2023, 8:29:29 p. m. node: debug 6
msg.payload : string[7]
"1769498"
5/11/2023, 8:29:29 p. m. node: debug 3
msg.payload : number
1769835
```

En el siguiente vídeo se puede observar la prueba final del funcionamiento del gemelo digital, donde se comprueba que se puede controlar, en este caso en específico: una válvula proporcional, el nivel del tanque, una bomba y un sensor de nivel. Para visualizar el vídeo ingrese al siguiente enlace [VID_20231027_153104.mp4](#) [22]

4.3. Resultados

Tabla 3. *Resultados esperados*

Resultado	Indicador	Objetivo Relacionado
Aplicativo móvil	Software	Objetivo específico 4
Servidor Web Server	Software	Objetivo específico 3
Gemelo digital	Modelo 3d	Objetivo específico 1 y 2
Flujo de datos en Node-Red	Software	Objetivo específico 3

4.3.1 *Aplicativo móvil.*

Como resultado del aplicativo móvil se entrega un APK que se puede ejecutar en dispositivos Android, la app cuenta con la configuración de ARcore integrada, la cual permite posicionar modelos de realidad aumentada en el entorno real. La evidencia del desarrollo se encuentra desde la página 51 hasta la página 57, el modelo que se utilizó para la realidad aumentada se puede encontrar desde la página 43 hasta la página 47. Para obtener accesos a la APK puede ser descargado desde `gemelo.apk` [23]

Teniendo en cuenta el resultado esperado del objetivo 4, se puede decir, que este se cumplió en su totalidad.

4.3.2 *Servidor web server.*

Como resultado se entrega una máquina virtual en los servicios en la nube de Google Cloud, la máquina ejecuta un sistema operativo basado en Linux el cual permite ejecutar programas de node.js. Mediante el siguiente enlace se suministran los códigos necesarios para ejecutar el servidor `servidornodejs.rar`. [24]

Los pasos para replicar la puesta en marcha del servidor se encuentran desde la página 66 hasta la página 71.

Teniendo en cuenta el resultado esperado del objetivo 3, se puede decir, que este se cumplió en su totalidad.

4.3.2 *Gemelo digital*

Como resultado se entrega el prefab, este es el diseño de cada una de las piezas y el modelo 3D de la planta de automatización completado en Unity. En el siguiente enlace, se podrá hacer visualización del prefab `Update your browser to use Google Drive, Docs, Sheets, Sites, Slides, and Forms - Google Drive Help` [25]

De igual forma, se entrega el desarrollo total de la aplicación en Unity, este incluye los script, los objetos y demás cosas en general `My project (2).rar` [26]

Teniendo en cuenta los resultados esperados del objetivo 2 y 1, se puede decir, que este se cumplió en su totalidad.

4.3.2. *Flujo de datos en Node-RED*

Como resultado se entrega un archivo tipo JSON, este se podrá ejecutar en la aplicación de Node-RED. `node-red.json` [27]

Teniendo en cuenta el resultado esperado del objetivo 3, se puede decir, que este se cumplió en su totalidad.

4.4. Discusión de resultados

Al momento de llevar a cabo el desarrollo de los objetivos propuesto se logró analizar que se pudo haber tomado mejores decisiones en cuanto a el software de diseño utilizado, este software (Blender) se eligió por ser un software libre, sin embargo, al poner en balance las horas de trabajo que conllevó adquirir los conocimientos necesarios para crear cada una de las piezas, se debería plantear otros aplicativos, evidentemente existen beneficios a la hora de intentar realizar cualquier modelo 3D debido a que el software altamente reconocido y utilizado, por lo que se deja en debate si es factible aprender a manejar un software sofisticado que cuenta con poca información a pesar de ser reconocido.

Otro punto de debate es el uso de ARcore y unity, con esto se logró implementar soluciones a ciertas dificultades de programación, no obstante, en el proceso de búsqueda se encontró una serie de programas más intuitivos y que no requieren un alto conocimiento o mucha capacidad de cómputos, a su vez, es necesario destacar que Unity sigue siendo usado por una gran cantidad de programadores de juegos, en consecuencia, la información es más asequible.

5. Conclusiones y Trabajos Futuros

5.1. Conclusiones

En base a la planta física se realizó un modelo 3D, donde se pueden observar características de la planta, además permitir modificar información de los respectivos actuadores de la planta. Para el correcto funcionamiento de la planta es necesario la normalización y escalado de datos ya que se necesita garantizar que los valores de entrada y salida sean los correspondientes a los que se desean observar y controlar, esta información se podrá encontrar en el datasheet de cada dispositivo correspondiente.

La creación del servidor en la nube habilita la posibilidad de acceder y tener control de la planta de automatización desde cualquier ubicación que cuente con internet, permitiendo que los estudiantes que no se encuentren presencialmente en el laboratorio, puedan vivir una experiencia similar a las prácticas directamente realizadas con la planta física, esto permite asimilar la información de forma más eficiente.

Tras realizar las pruebas de la comunicación del gemelo con la planta real se puede concluir que la latencia entre ellos dependerá del sitio donde se acceda a el servidor ya que este servidor en específico se encuentra hosteado en la región southamerica-west1-a la cual se encuentra en la región de Santiago de Chile, a pesar de esto la latencia no superara los 300 ms desde que se acceda de cualquier región en Suramérica, teniendo en cuenta esto si se llegan a presentar variaciones en la respuesta se pueden deber a factores como lo serian la capacidad de procesamiento del dispositivo que esta ejecutando el aplicativo, el acceso a la red o la interferencia de otras aplicaciones.

La interfaz intuitiva de la aplicación móvil facilita interacción con el gemelo digital. Se realizaron pruebas de funcionamiento para asegurar que a través del gemelo digital se logra controlar y supervisar dispositivos de la planta de automatización como válvulas, sensores de nivel y bombas de forma eficiente mediante.

5.2. Trabajos futuros

Una forma de mejorar el gemelo digital sería implementarlo con realidad aumentada, esto pasaría a ser el desarrollo completo de un aula virtual facilitando a modalidades de educación híbridas el tomar clases asistidas por tecnologías interactivas y completas, donde los estudiantes o cualquier usuario podría experimentar una inmersión total en el entorno 3D del aula virtual, tomando como base el laboratorio de automatización, el cual cuenta con bancos para el desarrollo de practicas basadas en los Programmable Logic Controller (PLC).

6. Recomendaciones

- Se recomienda que los equipos de la universidad cuenten con los suficientes módulos de extensión de salidas y entradas analógicas para garantizar el correcto funcionamiento del gemelo digital con todos los sensores.
- Se recomienda colocar un algún sistema de seguridad para acceso a la aplicación con el fin de hacer el entramado de la información, ya sea con una base de datos, un inicio de sesión o un acta de seguridad.
- Es necesario realizar ajustes en la planta de automatización, ya que algunos de los equipos no operan de manera adecuada
- Los usuarios deben consultar la tabla de variables definidas como referencia (figura 61), para poder acceder a las memorias, esto con el fin de que funcione correctamente la aplicación.

Referencias

- [1] Sáez Martínez, P. J. (2019, octubre 8). El pasado, presente y futuro de la Realidad Aumentada. Onirix. <https://www.onirix.com/es/el-pasado-presente-y-futuro-de-la-realidad-aumentada/>
- [2] Lledó, G. L., Lledó, A. L., & Carreres, A. L. (2022). Tendencias globales en el uso de la realidad aumentada en la educación: estructura intelectual, social y conceptual. *Revista de Investigación Educativa*, 40(2), 475-493.
- [3] C. R. Bello, "La realidad aumentada: lo que debemos conocer," *Tecnología Investigación y Academia*, vol. 5, no. 2, pp. 257-261, 2017.
- [4] El poder de los gemelos digitales para mejorar el diseño, la atención al cliente y la eficiencia. (2019, octubre 23). *Revista Española de Electrónica* | Todas las noticias de electrónica actualizadas a diario; *Revista Española de Electrónica*. <https://www.redeweb.com/articulos/el-poder-de-los-gemelos-digitales-para-mejorar-el-diseno-la-atencion-al-cliente-y-la-eficiencia/>

[5] Orive, D., López, A., Estévez, E., Orive, A., & Marcos, M. (2021). Desarrollo de gemelos digitales para la simulación e integración de activos de fabricación en la industria 4.0. In XLII Jornadas de

[6] Vivas, H. L., Muñoz Abbate, H., Cambarieri, M. G., Petroff, M., & García Martínez, N. (2014). Arquitectura de Software con websocket para aplicaciones web multiplataforma.

[7] Node-RED. (2023, 22 de octubre). Recursos. Recuperado de <https://nodered.org/about/resources/>

[8] E. Ollora Zaballa, "Implementación de una plataforma de análisis y visualización de datos IoT multi-protocolo en tiempo real sobre Node-Red," 2021.

[9] Tudela Peñarrubia, J. J. (2017). Diseño e implementación con Node.js de una aplicación web para el seguimiento y evaluación del aprendizaje.

[10] Haro, E., Guarda, T., Peñaherrera, A. O. Z., & Quiña, G. N. (2019). Desarrollo backend para aplicaciones web, servicios web restful: Node. js vs spring boot. Revista Ibérica de Sistemas e Tecnologías de Informação, (E17), 309-321.

[11] brandemia_. (2022, marzo 17). Ubuntu desvela su nueva identidad para 2022. ¿Qué cambios trae? Brandemia. <https://brandemia.org/nueva-identidad-de-ubuntu-y-logo-2022>

[12] Ubuntu: la polifacética distribución de Linux con soporte a largo plazo. (2023, septiembre 14). IONOS Digital Guide; IONOS. <https://www.ionos.es/digitalguide/servidores/know-how/ubuntu-un-sistema-para-todos-basado-en-linux/>

[13] Martínez Juliá, P. (2006). Software libre, Linux y Ubuntu. Eubacteria, nº 17 (2006).

[14] A. Ajenjo Jurado, "Desarrollo de una aplicación móvil en Unity 3D con la API ARCore de Realidad Aumentada," 2022.

[15] "Decreto 1412 de 2017," Gestor Normativo. [En línea]. Disponible en: <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=83247>.

[16] Moreno, W. Y. R., & Fagua, A. L. F. (2013). Laboratorios remotos y virtuales: una herramienta para el desarrollo de prácticas en ingeniería. *Revista Ciencia, Innovación y Tecnología*, 1, 71-80. doi: 10.18273/revcyt.v1i1.104

[17] Valderrama Serrato, J. N. (2018). Laboratorio remoto de automatización con aplicaciones de realidad aumentada

[18] Daquilema, M. Á., Checa, D. S., & Martín, C. A. (2020). Diseño de un ambiente de realidad aumentada para supervisión, mantenimiento y control de un proceso industrial (Doctoral dissertation, ESPOL. FIEC).

[20] Flórez Farfán, Y. G. (2020). Diseño de una aplicación piloto de realidad aumentada para el apoyo en el mantenimiento instrumental de una planta de aglomeración de cobre.

[21] Martínez Herrera, D. (2022). Estrategia virtual enfocada a la industria 4.0 sobre una planta piloto industrial. Universidad Santo Tomás.

[22] VID_20231027_153104.mp4. (s. d.). Google Docs.
<https://drive.google.com/file/d/13ZUCoBku8edTSTsfGfhO-1VRXhCmcZbI/view>

[23] gemelo.apk. (s. d.). Google Docs.
<https://drive.google.com/file/d/1SuG4UyO51HcV2kW7TETZ4eDYVqsnoX5D/view>

[24] servidornodejs.rar. (s. d.). Google Docs.
<https://drive.google.com/file/d/1PQlOcQcjoPInXvQBRa4zGU7QjQ1DyiJ/view?usp=sharing>

[25] Update your browser to use Google Drive, Docs, Sheets, Sites, Slides, and Forms - Google Drive Help. (s. d.). Personal Cloud Storage & ; File Sharing Platform - Google.
https://drive.google.com/drive/folders/1zIAE5DG9LdpmhbjbO-VxJxGRh-wlS_7B

[26] My project (2).rar. (s. d.). Google Docs. <https://drive.google.com/file/d/1E4W862d0-xlOb31hEZcViK9fueRyjjudh/view?usp=sharing>

[27] node-red.json. (s. d.). Google Docs. <https://drive.google.com/file/d/1C9DWjT3d-J-hQi13LZT8EVMMzj5jgB4B/view>