Gemelos digitales híbridos para la enseñanza de robótica. Un entorno distribuido en tiempo real

Alumno: Néstor Adrián Balich

Director: Gustavo Rossi

Co-Director: Carlos Neil

Trabajo presentado para

obtener el título de Doctorado en Informática

(noviembre, 2025)

Resumen

Esta tesis presenta el diseño, desarrollo e implementación de un entorno híbrido de simulación educativa basado en un modelo de gemelo digital, orientado a la enseñanza de programación y robótica en carreras técnicas y no técnicas. La solución se materializa en el simulador ESTER, una plataforma 3D multiusuario desarrollada en Unity, capaz de controlar hasta 30 robots virtuales sincronizados con robots físicos mediante una arquitectura bidireccional.

El modelo híbrido implementado integra componentes simulados y reales de forma complementaria, permitiendo experiencias educativas con alto grado de realismo, autonomía y retroalimentación sensorial.

El entorno fue aplicado en contextos universitarios presenciales y virtuales, integrando programación textual en múltiples lenguajes y programación visual basada en bloques (mediante ATOM+ para TurboWarp). Su diseño pedagógico se fundamenta en el construccionismo, el aprendizaje activo, el aprendizaje significativo y la Educación Basada en Competencias (EBC).

En este marco, se incorporan los resultados de aprendizaje como referencia para definir las capacidades observables y medibles que alcanzan los estudiantes, y el alineamiento constructivo como estrategia para asegurar la coherencia entre los resultados de aprendizaje, las actividades de enseñanza y las herramientas de evaluación (Neil, 2024). La validación se realizó mediante estudios de caso, rúbricas de desempeño, encuestas de percepción y métricas técnicas.

Los resultados muestran una mejora significativa en la comprensión de estructuras de control, la motivación estudiantil y la transferencia de conocimiento. El simulador ha sido reconocido por su innovación y escalabilidad, recibiendo premios académicos y financiamiento internacional. Esta investigación aporta un modelo replicable y sostenible para la enseñanza de robótica y programación en entornos híbridos, posicionando a los gemelos digitales como herramienta clave en la educación del siglo XXI.

Palabras claves

Simulación educativa; robótica; gemelo digital; programación visual; aprendizaje activo; educación basada en competencias; Unity 3D; modelos híbridos; educación superior.

Dedicatoria

Esta dedicatoria es para quienes han sido mi soporte incondicional. Quiero dedicar la presente tesis, que culmina una etapa de intenso trabajo a lo largo de los años, a mi esposa Mercedes y mis hijos Berenice y Franco, por ser mi fuente constante de inspiración y mi motor en los momentos más exigentes.

Sin su amor, paciencia y apoyo diario, este recorrido no habría sido posible. A mi familia, por su comprensión incondicional durante todas las horas dedicadas a esta investigación.

También a la Universidad Abierta Interamericana (UAI) y al Centro de Altos Estudios en Tecnología Informática (CAETI), por brindarme el espacio de investigación y la posibilidad de utilizar el simulador en el dictado de clase, integrando la teoría con la práctica.

En especial, a mis padres Ester y Juan, a mi abuela Teresa y a mi tía María, quienes sembraron en mí el deseo de aprender, la curiosidad y la perseverancia para alcanzar mis metas.

Reconozco a quienes creyeron en este proyecto desde el inicio, cuando solo era una idea. Su confianza fue un motor adicional “este trabajo también es suyo”.

Gracias, por último, a mis tutores y revisores, cuya ayuda y dedicación incansable contribuyeron significativamente a la calidad final de este trabajo.

Reconocimientos

Esta tesis doctoral fue posible gracias al apoyo, la colaboración y la confianza de numerosas personas e instituciones que, de diferentes maneras, contribuyeron a su realización.

Agradezco profundamente a mi director de tesis y codirector, por su orientación crítica, su confianza permanente y su visión para ayudarme a consolidar este trabajo en el marco de la formación doctoral.

A la Universidad Abierta Interamericana, por brindarme el marco institucional para llevar adelante este proyecto y por haber distinguido el uso de nuestro simulador ESTER (Entorno Simulado de Trabajo Educativo en Robótica) con el Premio a las Buenas Prácticas Educativas 2024, fortaleciendo su aplicación en la Maestría en Tecnología Educativa y en la carrera de Ingeniería en Sistemas Informáticos.

Al equipo del Laboratorio de Robótica Física e Inteligencia Artificial, por acompañar el crecimiento del simulador desde sus primeras versiones hasta su consolidación como herramienta pedagógica innovadora.

Al comité organizador del Congreso Argentino de Ciencias de la Computación 2024, por reconocer el valor del proyecto con el premio a la mejor exposición del track de Innovación en Educación Informática, validando así su impacto académico.

A AWS Cloud Credit for Research, por la adjudicación del grant cloud, que permitió escalar el simulador ESTER a un entorno distribuido, accesible y multiusuario.

A mis colegas, colaboradores, en especial a Berenice Lourdes Balich y Franco Adrián Balich mis hijos, por su compromiso, talento y dedicación en cada etapa de desarrollo, desde el diseño técnico hasta la validación pedagógica.

A todos los estudiantes que participaron en las pruebas y experiencias con el simulador, por aportar sus ideas, su entusiasmo y su tiempo para poner a prueba este entorno híbrido.

Índice General

Contenido

CAPÍTULO 1 1

Introducción 1

1.1 Problema de investigación preliminar 2

1.2 La pregunta que guía este trabajo es: 2

1. 3 Vinculación del problema con el ámbito laboral 2

1.4 Solución al Problema Identificado 3

1.5 Enfoque pedagógico basado en competencias 4

1.6 Resumen de problemas y soluciones 5

1.7 Objetivo del trabajo 5

1.7.1 Objetivo general: 5

1.7.2 Objetivos específicos: 6

1.8 Transferencia de los resultados obtenidos. 6

1.9 Contribuciones principales 7

1.10 Estructura general del trabajo final / tesis 8

CAPÍTULO 2 10

Estado del arte 10

2.1. Simuladores de robótica existentes 10

2.2. Modelos educativos basados en gemelos digitales 11

Origen del término "Digital Twin" 12

Evolución del concepto 12

2.3. Programación visual en contextos formativos 13

2.4. Integración hardware-software en entornos educativos 14

2.5. Modelos híbridos en gemelos digitales: tendencias y desafíos 14

2.6. Limitaciones de los enfoques actuales 15

CAPÍTULO 3 17

Diseño del entorno híbrido 17

3.1. Evolución de la plataforma: de un entorno básico a uno multiusuario 17

3.2. Arquitectura técnica: motor de simulación, API, sockets y backend 21

3.3. Infraestructura para clases con múltiples alumnos 24

3.4. Diseño y fabricación del robot físico experimental (TWIN ROBOT) 25

3.5. Incorporación del modelo híbrido de gemelo digital 27

3.6. Sincronización con el robot físico: arquitectura bidireccional 28

3.7. Procesos en tiempo real: sensado, respuesta y control 29

3.8. Seguridad, escalabilidad y rendimiento del sistema 31

3.8.1 Seguridad 31

3.8.2 Escalabilidad 32

3.8.3 Rendimiento 32

3.8.4 Compatibilidad 32

3.8.4 Proyección de validación futura: 32

CAPÍTULO 4 34

Modelo educativo y experiencia didáctica 34

4.1. Fundamentación pedagógica: construccionismo, aprendizaje activo y significativo 34

4.1.1 Constructivismo 34

4.1.2 Aprendizaje activo 35

4.1.3 Aprendizaje significativo 35

4.1.4 Educación basada en competencias (EBC) en el entorno ESTER 36

4.2. Experiencias en carreras técnicas y no técnicas 38

4.2.1 Carreras técnicas 39

4.2.2 Carreras no técnicas y posgrados 39

4.3. Programación visual con bloques integrada al entorno 40
4.3.1 Características de ATOM+ 41
4.3.2 Integración con el simulador 43
4.3.3 Impacto pedagógico observado 43
4.4. Actividades prácticas con el robot físico 44
4.4.1 Diseño y ejecución de actividades 44
4.4.2 Justificación pedagógica 45
4.4.3 Impacto observado 45
4.5. Estudio de caso: aplicación en Ingeniería en Sistemas 46
4.5.1 Contexto 46
4.5.2 Actividades implementadas 46
4.5.3 Resultados observados 47
4.5.4 Escalabilidad del sistema con múltiples robots físicos 47
Análisis de los resultados 48
Gráfico de latencia vs número de robots 49
Conclusión 50
4.6. Estudio de caso: aplicación en posgrados en Tecnología Educativa 50
4.6.1 Modalidad 51
4.6.2 Actividades destacadas 51
4.6.3 Resultados 52
4.6.4 Reflexión pedagógica 52
4.7. Impacto pedagógico: aprendizajes observados y percepción estudiantil 53
4.7.1 Diseño de rubricas de evaluación 53
4.7.2 Aprendizajes observados 54
4.7.3 Percepción estudiantil (encuestas anónimas) 54
4.7.4 Reflexión pedagógica 55
4.8 Conclusión 55

CAPÍTULO 5 57
Desarrollo de modelo híbrido de gemelo digital 57
5.1. Fundamentos del modelo híbrido propuesto 57
5.2. Interacción simulador ↔ robot físico: arquitectura y flujos 61
5.3. Integración de datos reales y simulados 62
5.4. Toma de decisiones compartida en tiempo real 65
5.5. Escenarios de predominancia: simulación vs. robot físico 66
5.6. Validación funcional del modelo con actividades concretas 67
CAPÍTULO 6 68
Evaluación y resultados 68
6.1. Estrategia metodológica de evaluación 68
6.2. Indicadores técnicos y pedagógicos de desempeño 69
6.2.1 Indicadores técnicos 69
6.2.2 Indicadores pedagógicos 69
6.3. Resultados cualitativos y cuantitativos 69
6.3.2 Resultados pedagógicos 70
6.4. Comparación con modelos educativos tradicionales 71
6.5. Retroalimentación de docentes y estudiantes 71
6.6. Limitaciones observadas 72
6.7. Conclusión 72
CAPÍTULO 7 74
Trabajos relacionados y producción científica vinculada 74
7.1 Introducción 74
7.2 Antecedentes y desarrollos relacionados 76
a) Simuladores educativos en robótica 76
b) Gemelos digitales y educación 78

c) Programación visual en contextos formativos 78
7.3 Originalidad y aportes del entorno ESTER 79
7.4 Producción científica generada 79
7.5 Convergencia con tendencias emergentes 81
7.6 Síntesis del aporte diferencial 82
7.7 Desafíos y líneas futuras 83
7.8 Conclusión 84
CAPÍTULO 8 85
Aplicación pedagógica del modelo de gemelo digital híbrido 85
8.1 Introducción 85
8.2 Fundamentación teórica 86
8.3 Diseño pedagógico de la clase 87
8.3.1 Matriz de tributación (simplificada) 87
8.4 Rúbrica analítica y lineamientos de aprendizaje 88
8.4.1 Lineamientos para el desarrollo de actividades de aprendizaje 88
8.4.2 Validación de la rúbrica mediante reglas y meta-reglas 91
8.5 Micro TP (trabajo práctico focalizado) y trabajo integrador 92
8.6 Secuencia didáctica analítica 94
8.6.1 Modulo 1 95
TP .1.1 Movimiento básico del robot 95
8.6.2 Modulo 2 95
TP. 1.2 Lectura de sensores 95
8.6.3 Modulo 3 96
TP. 1.3 Sincronización física 96
8.6.4 Modulo 4 97
TP. 1.4 Colaboración y optimización 97
8.6.5 Modulo 5 97

TP. 1.5 Trabajo integrador final 98
8.7 Lista de verificación 98
8.8 Conclusión 100
CAPÍTULO 9 102
Conclusiones 102
9.1. Reflexión general 102
9.2. Logros alcanzados 102
9.2.1 – Evolución del simulador ESTER (Versiones 1.0 a 5.0) 103
9.3. Impacto educativo y tecnológico 105
9.3.1 – Impacto educativo y tecnológico del modelo híbrido ESTER + TWIN ROBOT 106
9.3.2. Limitaciones y líneas abiertas 107
9.3.3 – Limitaciones identificadas y líneas futuras de investigación 108
9.4. Contribución al campo 108
9.5. Validación del enfoque por competencias 110
9.6. Conclusiones 111
CAPÍTULO 10 113
Trabajos futuros 113
10.1. Evolución del robot físico TWIN ROBOT 113
10.2. Ampliación de ATOM+ con nuevas capacidades 113
10.3. Incorporación de analítica de aprendizaje (Learning Analytics) 114
10.4. Expansión hacia entornos inmersivos: Realidad Extendida (XR) 114
10.5. Adaptación a nuevos niveles y áreas de aplicación 114
10.6. Consolidación de una red educativa colaborativa 115
10.7. Expansión hacia arquitecturas distribuidas: Proyecto ESTER GRID Robotics 115
10.8. Publicaciones futuras y diseminación 117
10.9. Conclusión 117

Referencias 118
Anexos 128
Anexo A – Arquitectura del simulador ESTER 128
A.1. Motor gráfico y lenguaje base 128
A.2. Componentes del entorno virtual 128
A.3. Comunicación API – Backend – Cliente 128
A.4. Estructura del backend de control 129
A.5. Mecanismo de sincronización con el robot físico 129
A.6. Requerimientos mínimos y recomendados 130
A.7. Escalabilidad y uso en el aula 130
A.8. Herramientas auxiliares 130
Anexo B - Programación visual con ATOM+ 131
B.1. Descripción general de ATOM+ 131
B.2. Interfaz gráfica 131
B.4. Ejemplos de programas creados con ATOM+ 135
B.5. Integración con el simulador ESTER 136
B.6. Valor pedagógico 136
Anexo C – Protocolos de conexión con TWIN ROBOT 137
C.1. Objetivo del protocolo 137
C.2. Arquitectura de conexión 137
C.3. Formato del mensaje enviado al robot 137
C.4. Mensaje recibido desde el robot físico 138
C.5. Flujo de sincronización y lógica de prioridad 138
C.6. Comandos disponibles (comunicación básica) 139
C.7. Configuración del robot físico (ESP32/Arduino) 139
C.8. Seguridad y control de errores 139

Anexo D – TWIN ROBOTS 140
F.1. Introducción 140
F.2. Versión 1 – Detalle de impresión y armado 141
3. Modulo de robot de futbol de dos ruedas 144
F.4. Módulo de robot omnidireccional de fútbol robot 146
F.5. Versión 2 – otras impresiones adicionales 148
F.6. Consideraciones de mejora en V2 148
F.7. Aplicabilidad educativa 149
Anexo E – Evaluación educativa: encuestas y rúbricas aplicadas 150
Anexo F – Pantallas del simulador 153

Glosario Técnico

API (Application Programming Interface).

Conjunto de funciones y protocolos que permiten la comunicación entre sistemas. En ESTER se usa para enviar comandos a robots simulados o físicos mediante JSON. Garantiza interoperabilidad y un canal estandarizado. Es la base del control remoto del gemelo digital.

ATOM+.

Módulo de programación visual por bloques integrado a TurboWarp. Facilita el aprendizaje de algoritmos sin sintaxis textual y permite programar robots en tiempo real. Conecta directamente con robots físicos y simulados. Favorece accesibilidad en entornos educativos.

Backend.

Conjunto de procesos en servidor que manejan lógica, sockets, sincronización y asignación de robots. Coordina el uso multiusuario y mantiene la coherencia del modelo híbrido. Procesa datos del robot real y virtual. Es la capa central del sistema ESTER.

Bidireccionalidad.

Capacidad de enviar y recibir datos en ambos sentidos de forma continua. En ESTER permite que cambios en el robot físico afecten a la simulación y viceversa. Mantiene coherencia entre gemelos digitales. Es esencial para el funcionamiento en tiempo real.

Ciberfísico (CPS).

Sistema que integra componentes computacionales con elementos físicos conectados en tiempo real. El modelo ESTER + TWIN ROBOT es un CPS educativo. Combina simulación 3D, sensores reales y control distribuido. Facilita aprendizaje experimental seguro.

Colisión.

Evento donde un robot impacta un objeto físico o virtual. En el modelo híbrido, los datos del robot físico tienen prioridad para garantizar realismo. La simulación replica inmediatamente ese estado. Se usa para retroalimentación y detección de errores.

Digital Twin (Gemelo Digital).

Representación virtual dinámica de un sistema físico conectada por datos en tiempo real. Permite monitoreo, análisis y control sin riesgo. En educación favorece experimentación guiada. En ESTER sincroniza comportamiento físico y simulado del robot.

Fog/Edge Computing (Computación en el borde).

Paradigma donde el procesamiento se realiza cerca de los dispositivos físicos —en routers, gateways o mini-servidores locales— en lugar de depender exclusivamente de la nube. Reduce la latencia, mejora la respuesta en tiempo real y disminuye el tráfico de red. En entornos educativos con robótica distribuida, permite ejecutar sincronización, filtrado de sensores y control del TWIN ROBOT directamente en el "borde", beneficiando la operación simultánea de múltiples robots en ESTER.

Gemelo Digital Híbrido.

Integra de forma simultánea datos físicos y simulados manteniendo coherencia. Cuando el robot real está conectado, sus datos predominan en la simulación. Permite operación continua entre ambos entornos. Eleva el realismo del aprendizaje práctico.

Hardware Embebido.

Conjunto de placas, sensores y actuadores diseñados para funciones específicas dentro del robot. Utiliza Arduino, ESP32 y WeMos, entre otros. Procesa comandos y datos para el gemelo digital. Es la base física del TWIN ROBOT.

IoT (Internet of Things).

Interconexión de dispositivos físicos a Internet para transmitir datos y ejecutar acciones remotas. Se usa para conectar robots y sensores en ESTER. Reduce cables y permite trabajo distribuido.

Lidar.

Sensor basado en láser para medir distancias y generar mapas. Puede integrarse al robot real o a la simulación. Mejora la percepción del entorno en tareas de navegación. Provee datos precisos para el gemelo digital.

Mirror (Unity Mirror Networking).

Biblioteca de Unity para sincronización de múltiples clientes en tiempo real. Replica estados, posiciones y acciones entre estudiantes. Permite escenarios colaborativos en ESTER. Es esencial para el modo multiusuario.

MQTT.

Protocolo de mensajería liviano para IoT con baja latencia. Transmite datos de sensores del TWIN ROBOT al servidor. Reduce consumo de ancho de banda. Facilita integración en entornos educativos distribuidos.

Nodo (Client Node).

Instancia individual del simulador asignada a cada estudiante. Controla un robot y sincroniza su estado con el servidor. Opera de forma independiente en entornos multiusuario. Mantiene comunicación constante con el backend.

Programación Visual.

Método de codificación mediante bloques gráficos. Reduce errores y facilita la enseñanza de algoritmos. En ESTER está representada por ATOM+. Permite crear soluciones rápidas sin experiencia previa en programación.

Robot Simulado.

Entidad virtual en Unity que reproduce la física y sensores del robot real. Responde a los mismos comandos enviados por el usuario. Se sincroniza con el robot físico en el modelo híbrido. Permite prácticas sin riesgo.

Robot Físico (TWIN ROBOT).

Robot impreso en 3D con sensores reales y tracción diferencial. Replica en el mundo físico los comandos enviados al simulador. Se conecta por Wi-Fi o Bluetooth y envía datos al gemelo digital. Funciona integrado con ESTER.

Sensores Virtuales.

Recreaciones digitales de sensores físicos como ultrasonido, color o infrarrojo. Permiten pruebas completas sin hardware real. Siguen la misma lógica que los sensores del robot físico. Son esenciales en actividades educativas previas al contacto real.

Sim-to-Real (S2R).

Transferencia de algoritmos desde simulación hacia hardware real. El modelo híbrido reduce discrepancias entre simulación y práctica. Permite verificar comportamientos antes de usar el robot físico. Mejora la confiabilidad del aprendizaje.

Simulación Multiusuario.

Capacidad de operar múltiples robots y estudiantes en simultáneo. En ESTER pueden coexistir hasta 30 nodos conectados. Cada robot mantiene estado independiente. Habilita prácticas colaborativas y ejercicios grupales.

Socket TCP.

Canal confiable orientado a conexión para enviar comandos y recibir datos. Asegura entrega ordenada y estable. Se utiliza para comunicación directa con el robot físico. Complementa otros canales del modelo híbrido.

Swarming (Robótica en enjambre).

Enfoque donde múltiples robots simples colaboran como un "enjambre", coordinándose mediante reglas locales sin un control centralizado. Permite comportamientos colectivos como exploración distribuida, formación de grupos, búsqueda cooperativa o cobertura de áreas extensas. En el contexto de ESTER-GRID, el swarming habilita actividades donde los estudiantes programan y observan la interacción emergente entre decenas de robots simulados o físicos trabajando juntos.

WebSocket.

Protocolo de comunicación bidireccional en tiempo real. Mantiene conexión persistente entre simulador, API y robots. Permite baja latencia y sincronización continua. Es fundamental para el modelo distribuido de ESTER.

UDP (User Datagram Protocol).

Protocolo de comunicación de baja latencia que envía paquetes sin necesidad de confirmación ni conexión persistente. Es más rápido que TCP porque evita comprobaciones de entrega y orden. Se usa en ESTER para transmisiones en tiempo real de audio, video o datos de sensores donde la velocidad es más importante que la fiabilidad perfecta. Ideal para robótica, streaming y control distribuido.

Unity 3D.

Motor de desarrollo utilizado para crear el simulador. Ofrece físicas realistas, renderizado 3D y compatibilidad multiplataforma. Soporta networking y sensores virtuales. Es el entorno base del gemelo digital.

WebGL.

Tecnología que ejecuta el simulador en el navegador sin instalaciones. Permite acceso desde cualquier dispositivo. Facilita despliegue educativo masivo. Mantiene buen rendimiento en entornos livianos.

CAPÍTULO 1

Introducción

La integración de tecnologías digitales emergentes en el ámbito educativo ha generado una transformación significativa en las estrategias de enseñanza, particularmente en áreas como la robótica educativa, donde la teoría y la práctica se interrelacionan de forma estrecha. En este contexto, el concepto de gemelo digital (digital twin), inicialmente desarrollado para aplicaciones industriales, comienza a destacarse en entornos educativos por su capacidad para conectar en tiempo real el mundo físico con entornos virtuales. Por otra parte, la falta de estándares y marcos de implementación universalizados limita la adopción masiva de gemelos digitales en (Sharma et al., 2022) específicamente orientado a la enseñanza de la robótica, que combina las fortalezas de un entorno simulado multiusuario con robots físicos reales, conectados mediante una arquitectura de comunicación y sincronización en tiempo real. A través de esta integración, se busca superar las limitaciones propias de los entornos exclusivamente virtuales o físicos, impulsando la participación, motivación y desarrollo de habilidades prácticas en estudiantes con perfiles académicos diversos (David et al., 2018; Bachmann et al., 2024).

A pesar de los importantes avances en entornos de simulación robótica educativa, persiste una clara brecha al trasladar el aprendizaje virtual hacia el contexto físico real (Ouyang & Xu, 2024; Chen et al., 2023; Tiryaki & Adigüzel, 2021). Los estudiantes enfrentan dificultades significativas cuando deben aplicar en robots físicos los algoritmos previamente validados en simuladores, debido a discrepancias en el comportamiento de sensores, motores y condiciones ambientales no previstas en la simulación. Esta problemática limita la adquisición efectiva de habilidades prácticas y disminuye la confianza de los estudiantes en sus propias competencias.

A través del desarrollo de este entorno híbrido se pretende evaluar el impacto educativo del gemelo digital en la comprensión de conceptos fundamentales, en la mejora del rendimiento académico y en la adquisición de competencias esenciales del siglo XXI.

El modelo híbrido propuesto busca promover la autonomía y la motivación intrínseca de los estudiantes (Ryan & Deci, 2000) mediante un enfoque activo de aprendizaje basado en proyectos (Thomas, 2000; Bell, 2010), integrando la filosofía construccionista de “aprender creando” (Resnick, 2013) en el contexto de la educación técnica argentina (Jacinto et al., 2025).

En los siguientes capítulos se desarrollan en profundidad la fundamentación científica, la arquitectura técnica implementada, las experiencias educativas llevadas a cabo, así como los resultados obtenidos y su análisis detallado.

1.1 Problema de investigación preliminar

El aprendizaje de robótica en contextos educativos presenta desafíos persistentes: acceso desigual al hardware, tiempos limitados de uso en clase, mantenimiento técnico de los dispositivos, barreras de entrada para los docentes y restricciones presupuestarias (Szczurek et al., 2023). Si bien los simuladores han brindado una respuesta parcial a estos obstáculos, a menudo presentan una desconexión con el mundo físico, lo que genera experiencias de aprendizaje fragmentadas (Filipescu et al., 2024).

Por otra parte, aunque el concepto de gemelo digital ha demostrado gran efectividad en ámbitos industriales, su adopción en la educación en especial enfocado a la enseñanza de la robótica aún incipiente (Kartashova et al., 2020). Existen pocos modelos que articulen simultáneamente el control del robot real con su representación virtual en tiempo real (Mazumder et al., 2023).

El modelo híbrido propuesto promueve la autonomía y la motivación intrínseca del estudiante, entendidas como factores esenciales para el aprendizaje significativo (Ryan & Deci, 2000), dentro del contexto de transformación de la educación técnica argentina (Jacinto et al., 2025) y del enfoque construccionista propuesto por (Resnick, 2013).

1.2 La pregunta que guía este trabajo es:

¿Cómo diseñar e implementar un modelo híbrido de gemelo digital que potencie el aprendizaje de robótica mediante la integración simultánea de entornos simulados y físicos?

1. 3 Vinculación del problema con el ámbito laboral

La problemática abordada guarda una relación directa con la práctica docente en carreras universitarias de ingeniería, tecnología educativa y disciplinas vinculadas a la programación y al desarrollo de sistemas. La carencia de herramientas que permitan prácticas distribuidas, multiusuario y en tiempo real limita las posibilidades de una formación integral, especialmente en instituciones con numerosos estudiantes o con recursos físicos (Szczurek et al., 2023;Piaggio et al., 2000).

La solución aquí propuesta surge de una experiencia concreta en el desarrollo de actividades formativas con estudiantes de grado y posgrado, en el marco de proyectos educativos reales relacionados con robótica, programación visual y sistemas embebidos.

1.4 Solución al Problema Identificado

Como respuesta a la problemática planteada, se desarrolla una arquitectura técnica de gemelo digital híbrido para robótica, implementada en el entorno denominado ESTER. Este entorno incorpora herramientas de control remoto, modelado virtual, y simulación distribuida "tecnologías alineadas con la visión de la Educación 4.0 y 5.0 (Balich et al., 2023; Mazumder et al., 2023;Filipescu et al., 2024).

Funcionalidades principales del entorno híbrido ESTER

- **Simulación multiusuario en 3D (Unity)**
    - Permite que hasta 30 estudiantes controlen sus robots virtuales en paralelo.
- **Sincronización con robots físicos (TWIN ROBOT)**
    - Arquitectura de comunicación bidireccional en tiempo real entre simulador y hardware real.
- **Modelo híbrido de gemelo digital**
    - Integración complementaria de datos simulados y reales.
    - Predominancia del robot físico cuando está conectado, garantizando coherencia.
- **Programación textual y visual (ATOM+ en TurboWarp)**
    - Soporte para múltiples lenguajes de programación.
    - Interfaz accesible para estudiantes sin perfil informático.
- **Retroalimentación sensorial y autonomía**
    - Respuesta inmediata del sistema al estado real o simulado.
    - Experiencias inmersivas con alto realismo educativo.
- **Validación educativa**
    - Uso de rúbricas, encuestas de percepción y métricas técnicas para evaluar competencias.
- **Escalabilidad y reconocimiento**

- Aplicación en grado y posgrado, adaptable a distintas disciplinas.
- Premios académicos y financiamiento internacional.

El entorno permite que cada estudiante controle su propio robot virtual, mientras que un robot físico refleje su estado, sensores y movimientos, en sincronía. El modelo admite que, en caso de conexión del robot real, sus datos prevalezcan sobre los simulados para enriquecer la experiencia educativa sin perder coherencia interna.

1.5 Enfoque pedagógico basado en competencias

Este trabajo se enmarca en el enfoque de Educación Basada en Competencias (EBC), ampliamente adoptado en carreras de ingeniería, educación tecnológica y enseñanza aplicada. La EBC propone un cambio en la lógica educativa tradicional: se orienta al desarrollo de capacidades demostrables y transferibles que permitan resolver problemas en contextos reales o simulados (Impagliazzo & Xu, 2024).

Las competencias se entienden como la integración de conocimientos (saber), habilidades (saber hacer) y actitudes (saber ser), evidenciadas en el desempeño del estudiante ante tareas concretas. En consecuencia, el foco se desplaza de la simple transmisión de contenidos hacia la capacidad de actuar con eficacia ante situaciones complejas o contextualizadas (Tobón, 2006).

Desde esta perspectiva, el desarrollo del simulador ESTER y del modelo híbrido de gemelo digital responde a la necesidad de formar competencias relevantes en programación, robótica, pensamiento computacional y trabajo colaborativo (Impagliazzo & Xu, 2024 ; Moon et al., 2020). El diseño del entorno fomenta la integración entre los conocimientos y habilidades a través de metodologías activas como la programación basada en bloques y los entornos de prueba remota (Bachmann et al., 2024) permitiendo:

- Realizar prácticas de programación con autonomía.
- Resolver problemas combinando simulación y robótica real.
- Comparar desempeños reales y virtuales para evaluar decisiones técnicas.
- Construir conocimiento activamente en un entorno seguro, progresivo y con retroalimentación inmediata.

El diseño didáctico del entorno se fundamenta en esta orientación competencial, desde la programación visual con nuestros propios módulos para TurboWarp que hemos llamado ATOM+ hasta las prácticas multiusuario y la validación en robots reales. Asimismo, se incorporan herramientas específicas de evaluación por competencias, como rúbricas, desafíos con desempeños observables y retroalimentación dinámica.

De este modo, la tesis se inscribe en una corriente contemporánea que busca que la tecnología no solo medie el aprendizaje, sino que también lo oriente hacia la acción, la autonomía y la construcción de saberes aplicables (Tobón, 2006).

1.6 Resumen de problemas y soluciones

Tabla 1: *Listado de problemas y propuesta de solución.*

Problema	Solución propuesta
Acceso desigual a hardware físico en robótica educativa	Integración de simulación multiusuario con representación física compartida
Falta de sincronización entre simuladores y robots reales	Modelo híbrido de gemelo digital con comunicación bidireccional en tiempo real
Limitaciones en la práctica distribuida y remota	Plataforma que permite acceso asincrónico y en red a entornos robóticos
Barreras de entrada en programación	Módulo visual con bloques ATOM+ basado en principios construccionistas

1.7 Objetivo del trabajo

1.7.1 Objetivo general:

- Este estudio se plantea como objetivo general implementar y evaluar un modelo híbrido de gemelo digital para la enseñanza de robótica que garantice una transferencia directa y efectiva del conocimiento entre simuladores y robots físicos.

1.7.2 Objetivos específicos:

- Desarrollar una plataforma educativa que vincule en tiempo real robots simulados y físicos en entornos distribuidos para posibilitar prácticas multiusuario sincronizadas y de baja latencia, escalables a clases numerosas y replicables en grado y posgrado.
- Evaluar la eficacia pedagógica del modelo híbrido en contextos educativos técnicos y no técnicos, utilizando diversas herramientas y estrategias de programación para evidenciar mejoras en la comprensión de estructuras de control, la transferencia simulación a real y la motivación estudiantil frente a enfoques tradicionales.
- Facilitar el acceso y aprendizaje de robótica a estudiantes con diferentes niveles de conocimiento mediante una interfaz amigable de programación visual (ATOM+) para democratizar la participación, reducir barreras de entrada y promover aprendizajes significativos en perfiles no informáticos.

1.8 Transferencia de los resultados obtenidos.

Los resultados fueron aplicados en distintos niveles educativos, destacando su replicabilidad y adaptación a otros contextos técnicos, como por ejemplo el laboratorio de robótica física e inteligencia artificial, la materia robótica de tercer año de ingeniería en sistemas y en la maestría en tecnología educativa y en la maestría en tecnología informática:

- Clases en carreras de grado (Ingeniería en Sistemas) donde los alumnos utilizan el entorno híbrido.
- Cursos de posgrado en Tecnología Educativa en los que docentes de distintas disciplinas programaron y controlaron robots reales desde un entorno virtual.
- Incorporación del simulador y los twin robots a proyectos institucionales de investigación y extensión universitaria.

Además, el modelo es adaptable a otras disciplinas que utilizan dispositivos físicos simulables como el Internet de las Cosas (IoT), la domótica o la automatización, lo que permite su extrapolación a diferentes contextos formativos. Asimismo, presenta un carácter escalable, lo que favorece su aplicación en propuestas de educación tecnológica a nivel nacional.

Desde los primeros trabajos sobre robótica en la nube (Cloud Robotic), se propuso aprovechar la potencia de Internet para extender las capacidades de procesamiento y aprendizaje de los robots mediante arquitecturas distribuidas. El trabajo presentado por Balich (2017) uno de los primeros modelos conceptuales que integraban robótica, IoT e inteligencia artificial como un sistema interconectado capaz de compartir información, recursos y experiencias entre dispositivos físicos y virtuales Figura 1.

Estos planteos iniciales constituyeron un antecedente tecnológico y conceptual clave para el desarrollo posterior del simulador ESTER, en tanto introdujeron la noción de colaboración remota y procesamiento distribuido que más tarde se consolidaría en el modelo híbrido de gemelo digital.

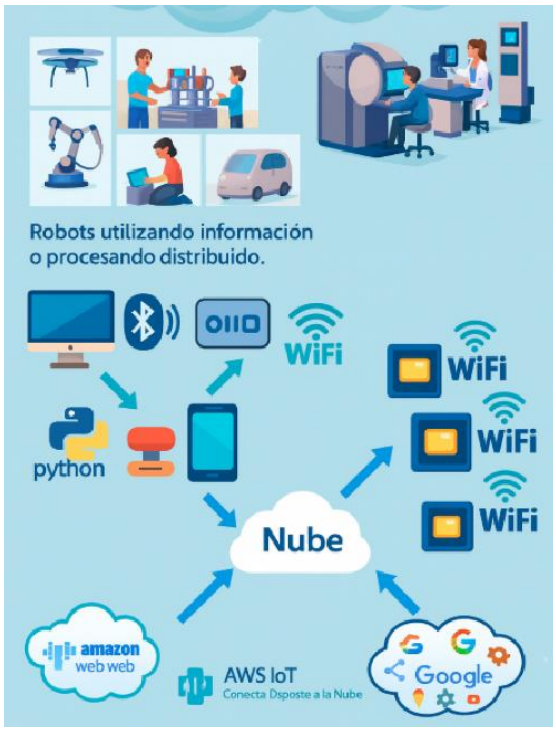


Figura 1. *Representación de robots autónomos, industriales y médicos que emplean procesamiento distribuido e integración en la nube mediante IoT y servicios web. Adaptado (Balich, 2017)*

1.9 Contribuciones principales

- Este trabajo ofrece múltiples contribuciones al campo de la tecnología educativa y la robótica aplicada. Propuesta de una arquitectura funcional de gemelo digital híbrido, que resuelve la brecha entre simuladores y robots físicos mediante comunicación bidireccional en tiempo real. Aunque estudios previos habían señalado la necesidad de

integrar ambas dimensiones (Marah & Challenger, 2024) esta tesis aporta una implementación concreta y validada en un entorno educativo.

- **Desarrollo de un entorno multiusuario sincronizado** en el simulador ESTER, que posibilita la interacción simultánea de hasta 30 estudiantes con robots virtuales y un robot físico. De esta forma se supera la limitación de modelos previos centrados en experiencias individuales (Mazumder et al., 2023).
- **Creación de una herramienta de programación visual personalizada (ATOM+)**, que facilita el acceso a la robótica en carreras no informáticas. Mientras que investigaciones anteriores destacaban el potencial de la programación por bloques para democratizar el aprendizaje (Moon et al., 2020; Coşkunserçe, 2023), aquí se presenta una solución propia integrada al simulador.
- **Validación pedagógica mediante estudios de caso, rúbricas y encuestas**, lo que permite medir el impacto tanto en la adquisición de competencias técnicas como en la motivación estudiantil. Esto responde a la necesidad de contar con evidencias empíricas que habían sido señaladas como faltantes en la literatura (Impagliazzo & Xu, 2024).
- **Diseño de un modelo replicable y adaptable**, aplicable a otras instituciones y disciplinas técnicas, consolidando la escalabilidad y sostenibilidad de la propuesta.

1.10 Estructura general del trabajo final / tesis

El trabajo final está estructurado en los siguientes capítulos:

Capítulo 1: Introducción.

Se plantea el problema a resolver, la pregunta guía, vinculación con el ámbito laboral, el enfoque pedagógico y objetivo y contribuciones esperadas.

Capítulo 2: Estado del arte.

Se revisan conceptos clave vinculados a la robótica educativa, la programación visual y los modelos de gemelo digital, así como antecedentes relevantes en el ámbito técnico y pedagógico.

Capítulo 3: Diseño del entorno híbrido.

Se detalla la arquitectura técnica, los componentes del sistema, el diseño de la interfaz y los procesos involucrados en la implementación del entorno ESTER.

Capítulo 4: Modelo educativo y experiencia didáctica.

Se presenta el enfoque pedagógico adoptado, se describen las experiencias formativas realizadas en contextos reales (grado y posgrado) y se analizan los aprendizajes observados.

Capítulo 5: Desarrollo del modelo híbrido de gemelo digital.

Se explican los flujos de datos, los procesos de sincronización entre simulación y robot físico, y los casos de uso específicos que validan la funcionalidad del sistema.

Capítulo 6: Evaluación y resultados.

Se presentan los datos obtenidos en las distintas experiencias de implementación, tanto cualitativos como cuantitativos, y se realiza un análisis de impacto educativo y técnico.

Capítulo 7: Trabajos relacionados y producción científica vinculada.

Se comparan las soluciones existentes con la propuesta desarrollada, y se detallan las publicaciones y ponencias que respaldan la evolución del trabajo.

Capítulo 8: **Conclusiones.**

Se sintetizan los principales hallazgos del proyecto, se reflexiona sobre los objetivos alcanzados y las limitaciones encontradas.

Capítulo 9: Trabajos futuros.

Se proponen líneas de investigación y desarrollo a partir de los resultados obtenidos, incluyendo mejoras técnicas, nuevas aplicaciones pedagógicas y posibilidades de escalabilidad.

Anexos: Se incluyen documentos complementarios como diagramas técnicos, capturas del entorno desarrollado, instrumentos de evaluación utilizados, entrevistas, encuestas y tablas de resultados.

Referencias: Se listan todas las fuentes bibliográficas, normativas y científicas utilizadas a lo largo del trabajo.

CAPÍTULO 2

Estado del arte

2.1. Simuladores de robótica existentes

La simulación ha sido una herramienta clave en la enseñanza de robótica, especialmente para superar desafíos asociados al uso de hardware físico, como costos elevados, mantenimiento, disponibilidad limitada y barreras de acceso técnico. Estas plataformas se utilizan ampliamente debido a su capacidad para modelar dinámicas físicas precisas, sensores, y entornos personalizables, facilitando así la transferencia de políticas de control desde las simulaciones hasta los robots reales presentan una revisión integral de los simuladores físicos empleados en robótica, identificando características esenciales como la fidelidad del motor de física, la compatibilidad con lenguajes de programación, y el soporte para simulación multiagente. La aleatorización de dominios y el enfoque sim-to-real (del simulador a la realidad) son estrategias emergentes para reducir la brecha que aún presentan estos (Muratore et al., 2022; Collins et al., 2021)

La simulación, cuando se implementa con fines pedagógicos, no solo proporciona un espacio seguro para la experimentación, sino que también favorece el aprendizaje activo y contextualizado (López-Nicolás et al., 2014).

La robótica educativa se consolida como un entorno ideal para el aprendizaje activo, integrando resolución de problemas, diseño y colaboración (Eguchi, 2014; Bers, 2008; Mubin et al., 2013). Este enfoque se articula con los principios del *Project-Based Learning* (Blumenfeld et al., 1991; Krajcik & Blumenfeld, 2006) y el ciclo iterativo del *Design Thinking* (Brown, 2008), los cuales guían el diseño progresivo del entorno híbrido ESTER.

Características de los simuladores evaluados: Cyberbotics Ltd (2025)

- **Gazebo**: muy utilizado en universidades junto con ROS (Robot Operating System): permite simulaciones físicas avanzadas, aunque con una configuración inicial compleja Open Robotic, (2025).
- **Webots**: combina facilidad de uso con simulaciones físicas realistas, útil para prototipado rápido Cyberbotics Ltd (2025).
- **V-REP / CoppeliaSim**: destaca por su integración con sistemas de control y conexión con robots reales; ideal para experimentación en IA (Coppelia Robotics AG, 2025).

- **Unity**: originalmente diseñado para videojuegos, ha sido aprovechado para simuladores educativos visuales y multiplataforma (Unity Technologies., 2025).
- **CoderZ, Virtual Robotics Toolkit, Tinkercad Circuits**: plataformas orientadas a niveles básicos con programación visual, aunque limitadas en flexibilidad y profundidad técnica. (goCoderZ, 2025;Cogmation Robotics, 2025; Autodesk, 2025).

Estos entornos permiten reproducir situaciones realistas y desarrollar competencias en control, lógica y toma de decisiones. No obstante, la mayoría funcionan como sistemas cerrados, sin una sincronización en tiempo real con robots físicos, lo que restringe su aplicabilidad en contextos híbridos.

2.2. Modelos educativos basados en gemelos digitales

El concepto de gemelo digital (DT) surge en la industria aeroespacial e ingeniería avanzada a principios de los años 2000. Tal como lo expone Grieves (2024) y más recientemente(Tao et al., 2019), un gemelo digital es una representación digital dinámica de un objeto, proceso o sistema físico, con una conexión continua de datos que permite monitoreo, simulación, predicción y control. Sus raíces teóricas se remontan a ideas previas sobre simulación y representación digital de sistemas físicos y su aplicación en educación representa un campo en expansión, donde la sincronización en tiempo real y la integración con entornos de simulación y control se consideran fundamentales (Sharma et al., 2022) .

La literatura actual distingue entre simulación convencional y modelos de gemelo digital con flujos de datos en tiempo real, señalando que estos últimos permiten la actualización dinámica del modelo virtual y una mayor capacidad de predicción y control (Mustafee et al., 2023).

Los gemelos digitales se diferencian de otras metodologías por su capacidad de reflejar el estado actual de un sistema físico mediante una réplica virtual conectada por canales de comunicación bidireccionales (Dihan et al., 2024). Esta característica los vuelve especialmente valiosos en escenarios de enseñanza técnica donde se requiere observar, predecir y validar comportamientos reales.

Se describen tres tipos de gemelos digitales: de instancia, de prototipo y de rendimiento, cada uno con diferentes niveles de integración y objetivos educativos o industriales (Botín-Sanabria et al., 2022).

Origen del término "Digital Twin"

- 2002: El término Digital Twin fue introducido formalmente por (Grieves, 2005) en una presentación sobre el ciclo de vida del producto (Product Lifecycle Management, PLM) en la Universidad de Michigan. En ese contexto, propuso una arquitectura donde cada producto físico tendría un equivalente digital que permitiría simular, analizar y optimizar su comportamiento a lo largo de todo su ciclo de vida.
- 2010: La NASA adoptó el concepto para mejorar el mantenimiento y simulación de naves espaciales, especialmente en misiones tripuladas, donde era crucial tener una réplica digital precisa para probar escenarios sin riesgos.

Evolución del concepto

Aunque inicialmente centrado en la industria aeroespacial y manufactura, el concepto se ha expandido a muchos otros campos, como:

- Ciudades inteligentes
- Salud personalizada
- Energía
- Robótica y educación tecnológica
- Internet de las Cosas (IoT), que ha sido clave en su desarrollo técnico
- Educación

En educación, su aplicación es aún emergente, con experiencias puntuales en:

- Formación profesional e ingeniería de sistemas (simulación de robots y dispositivos IoT
- Laboratorios remotos de electrónica o automatización (mediante IoT y tableros virtuales)
- Entrenamiento con realidad aumentada o realidad mixta

Sin embargo, los modelos aplicados específicamente a la enseñanza de robótica son escasos, y en su mayoría corresponden a:

- Unidireccionales (solo visualización de datos reales).
- Asincrónicos (carga de datos luego de la ejecución).
- No colaborativos ni accesibles, o dependientes de infraestructura compleja.

La tesis propone un modelo híbrido que busca superar estas limitaciones, integrando datos en tiempo real, accesibilidad y aplicabilidad pedagógica.

2.3. Programación visual en contextos formativos

La programación visual, particularmente mediante bloques, ha demostrado ser una estrategia eficaz para introducir conceptos de lógica computacional, pensamiento algorítmico y resolución de problemas, sobre todo en niveles iniciales y en carreras no técnicas (Feijoo-Almonacid & Rodriguez-Garavito, 2021). Estudios comparativos realizados sobre la misma, muestran que los entornos de programación visual favorecen el aprendizaje de estudiantes con baja experiencia técnica, incrementa la motivación y reducen los errores sintácticos, en comparación con lenguajes de programación textual (Sun et al., 2024). Siendo las plataformas más relevantes Scratch, Blockly, MakeCode y entornos específicos como CoderZ o mBlock permiten a los estudiantes sin formación técnica, construir algoritmos de forma intuitiva. Su efectividad ha sido ampliamente documentada para fomentar el pensamiento computacional, especialmente en poblaciones no técnicas.

En el ámbito universitario, su uso ha sido limitado por prejuicios sobre la necesidad de utilizar lenguajes de programación tradicionales desde el inicio. No obstante, en contextos híbridos (Balich N. A. et al., 2025) o en carreras no informáticas, la programación visual permite:

- Reducir barreras iniciales.
- Estimular la exploración y la creatividad.
- Disminuir la frustración en las etapas tempranas del aprendizaje.

Este enfoque pedagógico es clave para ampliar el acceso a la robótica educativa. Herramientas como ATOM+ (desarrollada en esta tesis) incorporan estos principios en un entorno visual robusto, accesible y conectado en tiempo real tanto con simuladores como con robots físicos (Balich et al., 2025).

2.4. Integración hardware-software en entornos educativos

El vínculo entre hardware físico (sensores, actuadores) y software de control es esencial en el aprendizaje de robótica. No obstante, las limitaciones técnicas, económicas y logísticas dificultan su integración plena en contextos educativos.

Los gemelos digitales ofrecen una solución a esta disyuntiva, permitiendo representar sistemas físicos complejos en entornos virtuales, recolectando datos reales para validar decisiones de diseño y comportamiento (Sharma et al., 2022). Esta integración no solo mejora la accesibilidad, sino que habilita prácticas pedagógicas basadas en datos reales y retroalimentación inmediata (Botín-Sanabria et al., 2022).

Además, los entornos híbridos basados en datos, como el modelo propuesto en esta tesis, incorporan sensores reales para actualizar el entorno virtual en tiempo real (Dihan et al., 2024).

Entre las limitaciones más comunes se destacan:

- Dependencia del hardware físico, propenso a fallas o disponibilidad limitada.
- Dificultad para reproducir condiciones de prueba de forma consistente.
- Acceso desigual de los estudiantes a las experiencias prácticas.

La simulación ha aliviado parcialmente estas limitaciones, pero a menudo se sacrifica realismo o interactividad. Los gemelos digitales aportan una nueva capa de integración: permiten no solo simular, sino también sincronizar datos reales y simulados, logrando entornos más ricos, dinámicos y colaborativos.

El modelo híbrido propuesto en esta tesis responde a esta necesidad, facilitando la conexión entre entornos simulados en Unity, APIs de control y robots reales con sensores activos, todo dentro de una experiencia educativa coherente.

2.5. Modelos híbridos en gemelos digitales: tendencias y desafíos

La evolución hacia modelos híbridos, que combinan simulación con datos en tiempo real, es una tendencia creciente en la literatura especializada. Estos modelos permiten no solo observar el

sistema real, sino también intervenir sobre él, mejorando su rendimiento y adaptabilidad (Mustafee et al., 2023).

Los desafíos más relevantes identificados incluyen la latencia en la comunicación, la sincronización de estados entre los entornos físico y virtual, y la creación de interfaces accesibles para usuarios no expertos (Mustafee et al., 2023). Por otra parte, la falta de estándares y marcos de implementación universalizados limita la adopción masiva de gemelos digitales, particularmente en el ámbito educativo. Esta problemática ha sido señalada por Sharma (2022), quienes destacan la necesidad de soluciones específicas orientadas a la enseñanza de la robótica que integren entornos simulados multiusuario con robots físicos reales.

Esto se logra mediante:

- Integración de algoritmos de *machine learning* para predicción y optimización.
- Comunicación bidireccional mediante tecnologías IoT.
- Retroalimentación física (estados, eventos, colisiones) desde el entorno real hacia el simulador.

Los desafíos principales incluyen:

- Latencia y sincronización de datos en tiempo real.
- Seguridad y robustez de la comunicación.
- Interfaces accesibles para usuarios no técnicos.

Pocos desarrollos han abordado esta problemática en entornos educativos, y aún menos en escenarios multiusuario. Esta tesis contribuye a ese campo mediante la creación de una arquitectura hibrida, distribuida, de bajo costo y diseñada para contextos reales de enseñanza colaborativa.

2.6. Limitaciones de los enfoques actuales

A partir del análisis de literatura y experiencias previas, se destacan las siguientes limitaciones en el uso de gemelos digitales en educación:

- Dificultades para aplicar estas tecnologías en carreras no informáticas o niveles introductorios (Feijoo-Almonacid & Rodriguez-Garavito, 2021; Botín-Sanabria et al., 2022).
- Ausencia de sincronización en tiempo real entre simuladores y dispositivos reales.
- Baja escalabilidad en propuestas pedagógicas multiusuario.

- Alta dependencia de hardware costoso y complejo.
- Escasa integración entre herramientas de simulación, programación y robótica física.

En este capítulo se ha revisado exhaustivamente los conceptos fundamentales que sustentan el modelo híbrido propuesto, desde los principios del construccionismo y el aprendizaje significativo hasta las más recientes investigaciones sobre gemelos digitales aplicados en contextos educativos (Grieves, 2024; Singh et al., 2021).La confluencia de estas perspectivas proporciona una sólida base científica que justifica el desarrollo de un entorno híbrido para la enseñanza de la robótica, abordando no solo las necesidades técnicas y pedagógicas sino también los desafíos actuales en educación tecnológica, tales como la motivación estudiantil, la adquisición de competencias clave y la reducción de la brecha entre teoría y práctica (Zhou, Zhang, et al., 2022; Kafai & Resnick, 2012). El marco teórico presentado constituye, por tanto, el fundamento esencial para el desarrollo metodológico y práctico descrito en los siguientes capítulos.

CAPÍTULO 3

Diseño del entorno híbrido

3.1. Evolución de la plataforma: de un entorno básico a uno multiusuario

El diseño del entorno híbrido partió de una necesidad didáctica concreta: ofrecer a los estudiantes un espacio accesible y seguro para practicar programación y robótica sin depender exclusivamente de hardware físico, cuya disponibilidad suele ser limitada en aulas real (Stein et al., 2025).Este diseño responde a los principios del *Project-Based Learning* contemporáneo, que prioriza la autenticidad de las tareas, la reflexión continua y la evaluación formativa (Larmer & Mergendoller, 2010; Condliffe et al., 2017). Además, se consideran los factores emocionales y de interés que influyen en la persistencia del estudiante frente al desafío tecnológico (Ortony et al., 2022; Swarat et al., 2012). Las primeras versiones del simulador **ESTER** que se llamó oRobby, desarrolladas entre 2020 y 2022, marcaron el inicio de un entorno virtual de programación robótica basado en la interacción remota entre múltiples usuarios.

En su versión inicial ESTER V1 (Figura 2), el sistema permitía que varios estudiantes se conectaran simultáneamente al mismo entorno y enviaran órdenes concurrentes a un único robot virtual compartido. Este modelo, aunque limitado en complejidad, constituyó una base experimental clave para el desarrollo posterior de las versiones multi-robot e integradas en la nube, que dieron lugar al modelo híbrido de gemelo digital implementado en 2025 (Balich et al., 2023). Las capturas de pantalla correspondientes a la interfaz principal, los menús y escenarios tridimensionales del simulador ESTER se presentan en el (Anexo F).

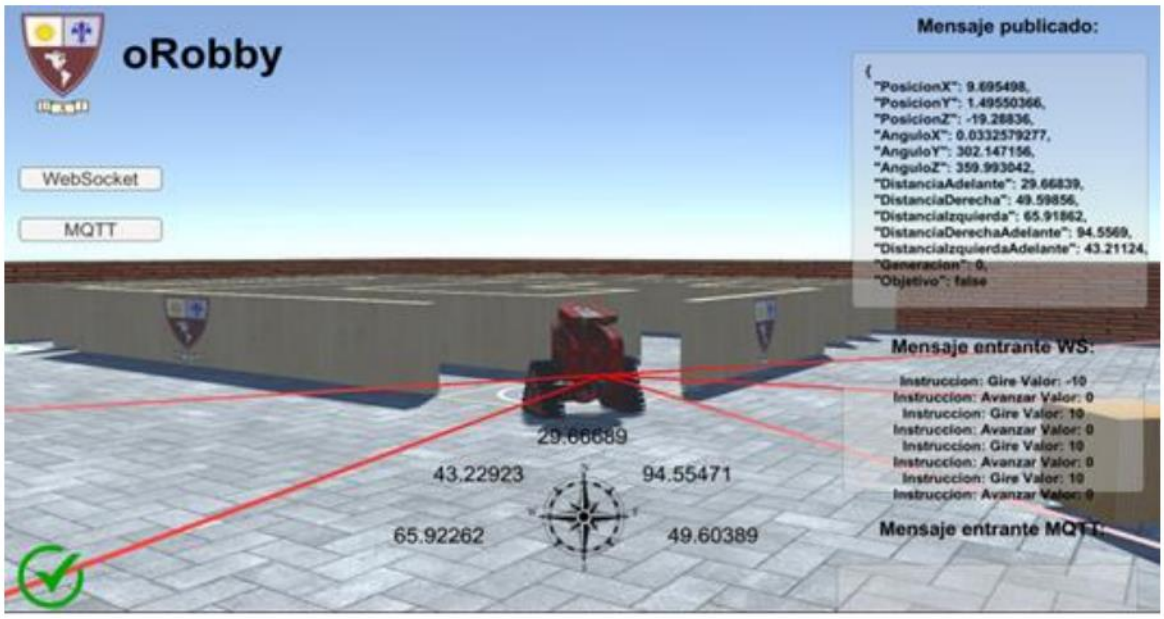


Figura 2. *oRobby inicio del simulador ESTER versión 1 con un único robot controlado concurrentemente por múltiples usuarios.*

Sin embargo, la expansión del uso en asignaturas como Robótica y paradigmas tecnológicos II reveló nuevas demandas:

- integración de múltiples usuarios y robots en simultáneo
- programación en diferentes lenguajes
- capacidad de sincronizar la simulación con robots físicos reales
- Incorporar lógica de interacción colaborativa entre robots.
- Sincronizar con robots físicos reales, integrando sensores y actuadores externos.

Estas demandas se complementan con las observadas en la literatura sobre robótica educativa, que destacan la necesidad de entornos inclusivos, escalables y adaptados a los contextos reales de enseñanza (Acker et al., 2023; Ananias & Gaspar, 2022).

En la versión ESTER V1, los alumnos podían conectarse concurrentemente al sistema y enviar comandos a su robot virtual (uno por alumno), ya que el prototipo mínimo viable (MPV) instanciaba 30 robots fijos, que respondía a las órdenes en tiempo real figura 3.

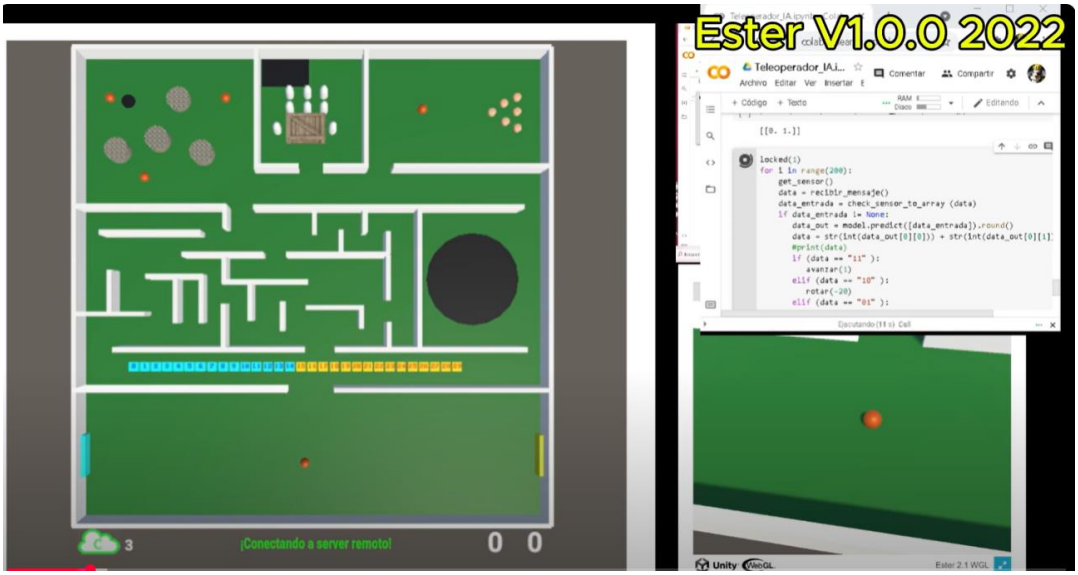


Figura 3. *Simulador ESTER versión 1 con 30 robots.*

El simulador evolucionó hacia la versión ESTER 2.3, que incorporó nuevos entornos tridimensionales y múltiples desafíos educativos diseñados para el desarrollo del pensamiento computacional y la lógica algorítmica.

Esta etapa incluyó prácticas como dibujar trayectorias, seguir líneas, rescatar objetos por color o desplazarse por laberintos, con una interfaz que permitía a cada estudiante experimentar el control de su propio robot en un entorno compartido.

La arquitectura seguía siendo centralizada, pero el modelo de programación y los escenarios 3D comenzaron a integrar principios de aprendizaje basado en proyectos (ABP) y simulación colaborativa, ampliando las posibilidades de aplicación del simulador en contextos educativos reales (figura 4).

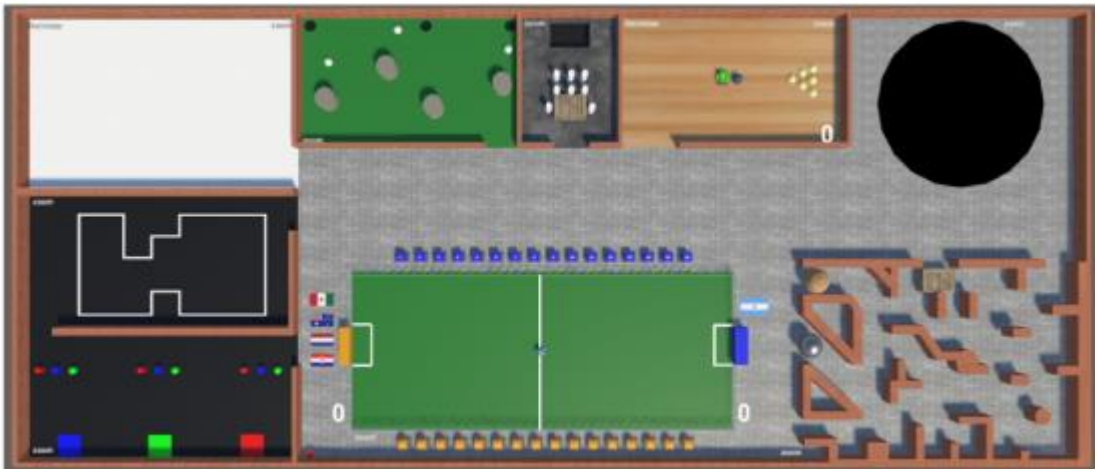


Figura 4. *Simulador ESTER versión 2.3 con múltiples desafíos: dibujar, sigue línea, rescate por color, golf, tesoro, bowling, fútbol y laberinto.*

De esta manera, el simulador evolucionó hacia una arquitectura multiusuario basada en modelos de gemelo digital, permitiendo hasta 30 robots virtuales independientes, uno por estudiante, y habilitando una comunicación bidireccional con robots físicos reales (Balich et al., 2024). Esta transición responde al enfoque pedagógico actual de la educación STEM, que demanda experiencias auténticas y colaborativas para potenciar habilidades técnicas y computacionales (Ouyang & Xu, 2024) -

Como señalan (Acosta et al., 2022),la validación empírica de simuladores robóticos frente a comportamientos físicos reales es fundamental para garantizar que los algoritmos desarrollados en entornos virtuales se transfieran de manera confiable a robots reales. Por ello, el entorno híbrido diseñado incorpora un módulo de sincronización que permite validar movimientos y sensores del robot físico contra su réplica virtual en Unity.

Parte de este proceso de evolución y consolidación del entorno multiusuario fue documentado en el video en *NeoroboticTV (Balich, 2024)* donde se presentan las principales etapas de desarrollo del simulador ESTER, la interacción en red de los robots virtuales y la incorporación del módulo ATOM+ en su versión 4.2.

La versión ESTER 4.2 (figura 5) consolidó todas las mejoras incorporadas en las etapas anteriores, integrando entornos tridimensionales más complejos, conectividad en red estable y una interfaz completamente inmersiva.

Esta edición permitió el desarrollo de múltiples desafíos simultáneos, como reciclaje, laberinto, fútbol, rescate, bowling y dibujo de trayectorias, con robots configurables que podían ser controlados desde distintos dispositivos conectados al servidor cloud.

El resultado fue un simulador robusto, estable y multiusuario, preparado para la integración con sistemas externos y la sincronización en tiempo real con robots físicos, lo que dio origen al modelo híbrido de gemelo digital presentado en el siguiente capítulo.



Figura 5. *Simulador ESTER versión 4.2 con múltiples escenarios y desafíos simultáneos*

3.2. Arquitectura técnica: motor de simulación, API, sockets y backend

El sistema se organiza en tres capas principales: simulación, comunicación e integración robótica. Esta estructura responde a la necesidad de generar un entorno flexible y robusto, tal como lo plantean las arquitecturas para automatización industrial con requisitos en tiempo real (Rodriguez & Castro, 2023).

3.2.1 Capa de simulación (cliente visual)

El motor gráfico seleccionado fue Unity 3D, por su alto rendimiento en visualización, su interfaz amigable y su compatibilidad con múltiples plataformas (Singh et al., 2025). La simulación contempla tanto visión en primera como en tercera persona, además de modos seguidor y cámara global, lo que facilita la comprensión de los fenómenos robóticos desde distintas perspectivas. Los robots animados reproducen un modelo físico realista de tracción diferencial de dos ruedas, complementado con sensores virtuales (ultrasónico, lidar, bumpers, color), cuya física y colisiones son gestionadas por el motor PhysX (Havok Physics), garantizando precisión en la representación de fenómenos físicos.

La elección de Unity frente a alternativas como Gazebo responde a varios factores diferenciales:

- **Rendimiento visual y latencia:** Unity ofrece tiempos de renderizado más bajos y mayor estabilidad gráfica en entornos multiusuario, mientras que Gazebo suele presentar latencias

mayores en simulaciones complejas, lo que puede afectar la fluidez en actividades educativas sin hardware especializado.

- **Curva de aprendizaje:** Unity proporciona una interfaz más accesible y una comunidad global de desarrolladores de videojuegos y simuladores, lo que permite capacitar a estudiantes y docentes en menor tiempo. En cambio, Gazebo exige mayor conocimiento técnico en ROS y Linux, lo que constituye una barrera de entrada para carreras no informáticas.
- **Multiplataforma y escalabilidad:** Unity permite desplegar el simulador en Windows, Mac, Linux y navegadores web, facilitando la distribución remota en aulas híbridas y multiusuario. Gazebo, en cambio, está más restringido a entornos de investigación y depende fuertemente de ROS para su integración.
- **Costo y accesibilidad:** Unity ofrece una versión gratuita con todas las funcionalidades necesarias para proyectos académicos, mientras que Gazebo requiere infraestructura y soporte adicional para integrarse en entornos distribuidos de gran escala.

Unity fue seleccionado por su equilibrio entre realismo gráfico, accesibilidad y escalabilidad educativa, mientras que Gazebo se reserva como opción robusta para proyectos de investigación avanzada en robótica. Esta decisión permitió implementar múltiples escenarios realistas para cada desafío pedagógico, con soporte multiplataforma y facilidad de adopción tanto para estudiantes con perfil técnico como no técnico.

3.2.2 Capa de comunicación

La capa de comunicación emplea una arquitectura híbrida, combinando API RESTful (para control de comandos del robot en formato JSON), WebSocket (para transferencia de datos en tiempo real), y el paquete Mirror para sincronización multiusuario en Unity. Esta última permite la replicación simultánea de robots y escenas, lo cual es esencial en entornos educativos distribuidos (Schäfer et al., 2022).

Además, se implementa un servidor central que administra los ID de cada estudiante y establece un canal bidireccional de datos. Esta arquitectura sigue modelos de colaboración educativa remota en tiempo real y está alineada con la necesidad de sincronización baja latencia en sistemas educativos con múltiples participantes (Stein et al., 2025; Rodriguez & Castro, 2023).

- Mediante el protocolo API REST se gestiona la comunicación con el entorno de programación mediante mensajes en json.
- Mediante WebSocket se establece la conexión entre la API y el simulador.

- Mediante el paquete Mirror para Unity se establece la sincronización de robots y escenas en cada instancia del simulador para cada estudiante.
- Servidor central que gestiona el ID de cada alumno/robot.
- Canal bidireccional que permite enviar comandos y recibir datos sensados (posición, velocidad, contacto con obstáculos, etc.).

3.2.2.1. Mecanismo de comunicación: mensajes vs. memoria compartida

La arquitectura de comunicación implementada en el entorno híbrido no utiliza ningún tipo de memoria compartida entre procesos, ni entre los robots simulados ni entre los robots físicos. Todo el intercambio de información se realiza mediante un modelo de mensajería distribuida, orientado a eventos y basado en protocolos de red. Este enfoque garantiza independencia entre los componentes, evita condiciones de carrera y facilita la escalabilidad para múltiples usuarios y múltiples robots físicos.

La comunicación se estructura exclusivamente a través de mensajes serializados en formato JSON, que viajan mediante distintos canales según el tipo de interacción (Anexo A):

- API REST: utilizada para el envío de órdenes del estudiante hacia el simulador o el robot físico.
- WebSocket: canal permanente para sincronizar en tiempo real el estado de los robots y del entorno (dos canales uno propio del simulador y otro de Mirror en unity).
- TCP/Socket: utilizado para la conexión directa con el robot físico, permitiendo la transmisión de datos sensoriales y comandos con baja latencia.

Cada robot —físico o simulado— mantiene únicamente su estado local, sin acceder a memoria compartida ni a estructuras centralizadas. El servidor actúa como autoridad global del sistema, recibiendo mensajes, resolviendo diferencias entre los datos provenientes del gemelo físico y del gemelo simulado, y redistribuyendo la información al resto de los clientes conectados.

Este enfoque de comunicación puramente basado en mensajes presenta varias ventajas:

- Reduce el acoplamiento entre componentes.
- Facilita la incorporación de nuevos robots físicos sin modificar el núcleo del simulador.
- Evita bloqueos o inconsistencias asociadas a la memoria compartida.
- Mejora la trazabilidad y el registro de las interacciones.
- Permite escalar el sistema a múltiples estudiantes y múltiples dispositivos en paralelo.

En síntesis, el entorno híbrido utiliza un mecanismo 100% orientado a mensajes, diseñado para garantizar robustez, coherencia y operación distribuida en tiempo real, sin mecanismos de memoria compartida en ninguna de las capas del sistema.

3.2.3 Capa de integración robótica

El sistema permite la conexión con robots físicos mediante sockets TCP, lo cual habilita el envío de comandos y recepción de datos desde sensores reales. En caso de discrepancias entre lo simulado y lo real, por ejemplo, una colisión no detectada en el entorno virtual pero sí en el físico, el sistema prioriza la información sensada físicamente. Esta estrategia se fundamenta en los principios de consistencia y fidelidad de los gemelos digitales en sistemas ciberfísicos.

La posibilidad de trabajar en espejo, donde lo físico y lo virtual reflejan sus estados, constituye una implementación práctica de modelos híbridos basados en datos en tiempo real, tal como se ha conceptualizado en la literatura (Mustafee et al., 2023). En el simulador ESTER, esta lógica se materializa mediante una arquitectura de comunicación que integra los siguientes componentes:

- **Conexión con robots físicos mediante sockets TCP**, lo que garantiza la transmisión confiable de datos en tiempo real.
- **Intercambio bidireccional de información**, a través del cual cada robot físico envía su estado actual y recibe los comandos generados desde el simulador.
- **Mecanismo de resolución de discrepancias entre simulación y realidad**, priorizando en todo momento los datos captados por los sensores físicos (por ejemplo, una colisión real no anticipada en el entorno virtual).
- **Soporte para multipresencia educativa**, que posibilita que cada estudiante se conecte desde su dispositivo, visualice y programe su robot virtual y, en caso de contar con un robot físico vinculado a su ID, experimente la modalidad híbrida; en ausencia de este, la interacción se mantiene en modo exclusivamente simulado.

De esta manera, la arquitectura asegura fidelidad en la representación de los fenómenos, flexibilidad en su aplicación pedagógica y escalabilidad en escenarios multiusuario.

3.3. Infraestructura para clases con múltiples alumnos

El entorno fue diseñado para adaptarse a contextos de enseñanza reales con alta densidad de usuarios. Se implementó un servidor en la nube para sesiones remotas, además de una opción de red local para clases presenciales sin conexión externa. Esto se ajusta a recomendaciones de arquitectura escalable y robusta en entornos educativos automatizados (Rodriguez & Castro, 2023)
.

Se definieron mecanismos de mapeo automático entre estudiantes y robots, junto con un repositorio de prácticas diferenciadas por niveles. Además, se priorizó la compatibilidad con equipos

de bajo rendimiento “notebooks básicas, tablets, móviles”, siguiendo las pautas de accesibilidad tecnológica en educación STEM (Refaat et al., 2025; Jacob et al., 2024) .

Las pruebas en carreras de grado y posgrado demostraron que esta arquitectura soluciona problemas históricos en la enseñanza de robótica, como la escasez de kits, los tiempos de espera y la baja personalización. Estas limitaciones ya habían sido identificadas como barreras recurrentes en estudios sobre simuladores educativos para robótica (Acker et al., 2023; Stein et al., 2025; Balich, Ocampo, et al., 2024)

- Servidor central en la nube para coordinar sesiones y robots (modo remoto).
- Opción de ejecución en red local (LAN) para clases presenciales sin conexión externa.
- Mapeo automático por ID de alumno ↔ robot asignado.
- Repositorio docente de prácticas y actividades por niveles (básico, medio, avanzado).
- Compatibilidad con equipos de bajo rendimiento: el simulador puede ejecutarse en notebooks básicas, tablet, mobiles, computadoras.

3.4. Diseño y fabricación del robot físico experimental (TWIN ROBOT)

Como parte central del modelo híbrido de enseñanza, se diseñó un robot físico denominado TWIN ROBOT, con el objetivo de replicar el comportamiento del robot simulado en el entorno 3D, posibilitando una experiencia educativa realista, coherente y accesible. El uso de robots físicos de bajo costo en educación ha demostrado ser una estrategia efectiva para reducir la brecha tecnológica (Balich et al., 2025), facilitar la apropiación de contenidos STEM y promover la motivación en los estudiantes (Refaat et al., 2025; Ananias & Gaspar, 2022).

3.4.1 Características del TWIN ROBOT

El diseño del robot incluye

- Chasis impreso en 3D, liviano y resistente.
- Tracción diferencial y sensores frontales y traseros.
- Microcontrolador intercambiable (Arduino, ESP32, WeMos, Raspberry Pi).
- Sensores ultrasónicos, giróscopos, encoders, y compatibilidad con Lidar.
- Batería recargable con autonomía media de 1 a 2 horas.
- Conectividad vía Wi-Fi y Bluetooth.

Estas decisiones responden a criterios de escalabilidad, accesibilidad y eficiencia en el uso de recursos, siguiendo modelos exitosos de plataformas educativas asequibles (Refaat et al., 2025; Jacob et al., 2024).

3.4.2 Objetivos de diseño:

El TWIN ROBOT fue diseñado para:

- Ser replicable en diversas instituciones educativas.
- Requerir bajo consumo de filamento y tiempos reducidos de impresión.
- Permitir armado sencillo sin herramientas especializadas.
- Compartir el mismo conjunto de comandos entre el robot real y el simulado.

Este diseño permite sincronizar los movimientos, sensores y eventos entre el entorno físico y el digital, lo que es esencial para cumplir con los criterios de un gemelo digital funcional y educativo (Singh et al., 2025).

Para una descripción detallada de las versiones, tiempos de armado y piezas del TWIN ROBOT V1 (figura 6) y V1b (fibura 6), ver **Anexo D**.

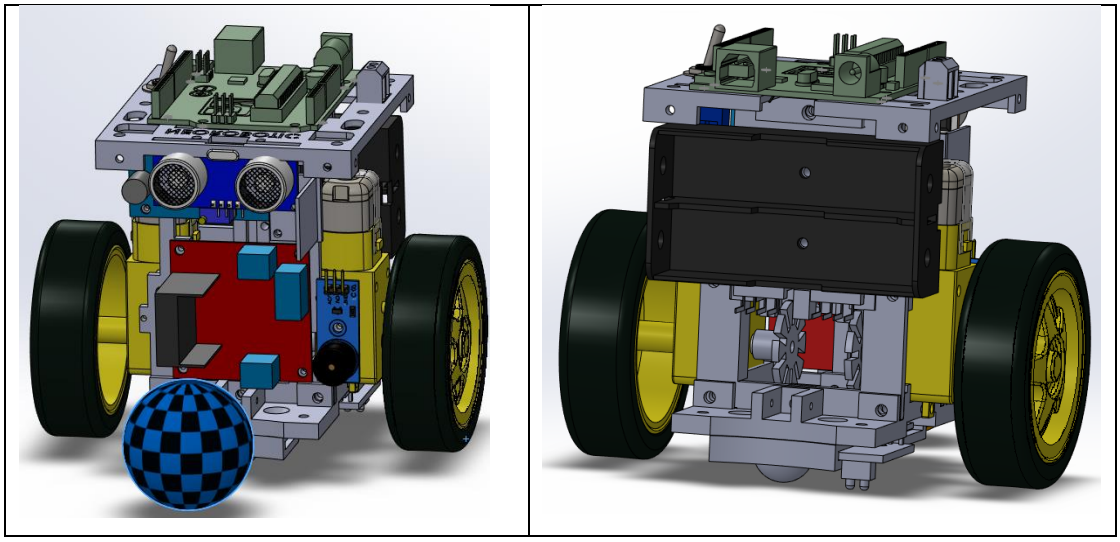


Figura 6. *Robot Twin Board Robot V1 diseño CAD*

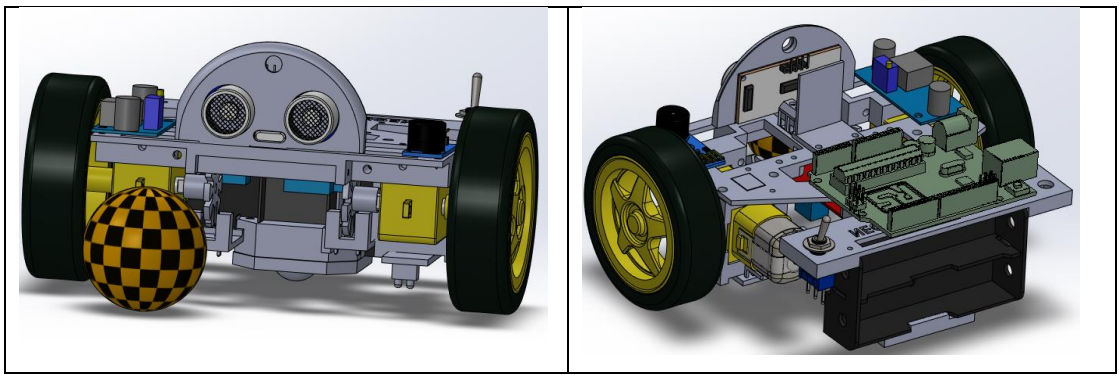


Figura 7. *Robot Twin Board Robot V1b diseño CAD*

3.5. Incorporación del modelo híbrido de gemelo digital

El modelo de gemelo digital híbrido implementado en el simulador ESTER se basa en una sincronización constante entre el robot simulado en Unity y su par físico, el TWIN ROBOT. Ambos reciben los mismos comandos, y los estados sensados del entorno físico se integran al simulador en tiempo real. Esta integración promueve una comprensión más profunda del comportamiento de los sistemas ciberfísicos por parte de los estudiantes (Mustafee et al., 2023; Radanliev et al., 2022).

Los estudiantes pueden observar cómo variables físicas (colisiones, ángulos, valores de sensores) impactan directamente en la lógica del sistema. Este modelo permite escenarios en los que se programan comportamientos en un entorno virtual que luego se validan y ejecutan en un entorno físico, aumentando la motivación y reduciendo la carga cognitiva (Tarng et al., 2024) .

- Robot simulado visualizado en un entorno 3D.
- Robot físico real (TWIN ROBOT), que puede conectarse opcionalmente a la sesión.
- Una arquitectura de comunicación que permite que ambos compartan información en tiempo real y se mantengan sincronizados.

Este modelo híbrido rompe con la lógica tradicional, en la que los entornos físico y virtual se consideran mutuamente excluyentes. En ESTER, la programación se realiza desde una interfaz unificada, y tanto el robot virtual como el físico ejecutan las mismas órdenes, salvo que una condición externa (por ejemplo, la detección de un obstáculo físico) genere una diferencia de estado.

La incorporación de este modelo híbrido permite que:

- El simulador reciba datos sensados desde el robot físico y los integre al entorno virtual.
- El robot físico ejecute comandos programados en el simulador.
- Ambos reflejen estados equivalentes (posición, rotación, colisiones) con mínima latencia.
- Se generen escenarios educativos donde lo simulado y lo real coexisten y se complementan.

Este enfoque fomenta en los estudiantes una visión sistémica de la robótica, donde no solo programan, sino que comprenden cómo el entorno, los sensores y la física afectan al sistema de control.

3.6. Sincronización con el robot físico: arquitectura bidireccional

Uno de los elementos más innovadores del simulador ESTER es su capacidad para sincronizarse con uno o más robots físicos reales en tiempo real (Balich et al., 2025), mediante una arquitectura bidireccional basada en sockets.

3.6.1 Flujo de datos:

- El robot físico transmite periódicamente su estado (posición, velocidad, detección de obstáculos, ángulos, valores de sensores).
- El simulador actualiza la visualización 3D en función de estos datos, si la sincronización está habilitada.
- El estudiante envía comandos desde la API, que se ejecutan tanto en el robot físico como en el virtual.

3.6.2 Funcionalidades clave:

- **Prioridad de datos físicos:** ante eventos del mundo real (como colisiones, giros, mapeo lidar), se prioriza el dato físico.
- **Control en espejo:** los movimientos se reflejan en tiempo real, garantizando continuidad y coherencia visual.
- **Desincronización controlada:** en caso de desconexión del robot físico, el sistema regresa automáticamente al modo simulado, evitando interrupciones pedagógicas.

Esta capacidad se alinea con lo que (Mustafee et al., 2023) definen como "gemelos digitales operativos", y es fundamental para la fiabilidad del sistema, especialmente en contextos educativos donde la precisión didáctica es crucial.

Además, se utiliza el paquete Mirror para Unity para garantizar la sincronización de múltiples instancias del simulador. Esta solución, adaptada originalmente de entornos multijugador en

videojuegos, ha sido aplicada exitosamente en simuladores educativos con alta concurrencia(Stein et al., 2025). Asimismo, se adaptaron algoritmos específicos para el control simultáneo de 30 robots y la actualización de los 30 entornos simulados por cliente (Balich et al., 2024)

3.7. Procesos en tiempo real: sensado, respuesta y control

Para mantener la coherencia en la ejecución simultánea de múltiples robots (simulados y reales), se aplican mecanismos como:

- Reducción de latencia en el envío de comandos al sistema de control (físico o simulado).
- Estabilización del rendimiento gráfico entre 30 y 60 FPS, incluso en equipos modestos.
- Sincronización simultánea de escenas de múltiples usuarios (Balich N. A., Balich F. A., et al., 2025).
- Recuperación automática ante pérdida de conexión o paquetes, usando el paquete Mirror.

El sistema implementa una lógica semi-autónoma: los comandos no se ejecutan de forma inmediata, sino que se procesan en una cola de eventos que permite priorizar y rechazar instrucciones repetidas o paquetes erróneos. Esto permite escenarios colaborativos, facilitando la gestión de interacciones complejas sin comprometer la estabilidad, a una elevada velocidad que hace que el estudiante los reciba como instantáneos.

Estos elementos permiten construir una experiencia educativa rica, fluida y confiable, incluso en dispositivos con recursos limitados. Se trata de una estrategia validada en arquitecturas robustas para automatización industrial en entornos educativos (Rodriguez & Castro, 2023).

En 2024 se realizó una prueba de rendimiento masivo del simulador ESTER 4.0 durante el Congreso CIITI, donde participaron simultáneamente 40 usuarios conectados a un mismo entorno en la nube. El objetivo fue evaluar la estabilidad de las conexiones, la latencia del servidor, la capacidad de respuesta en tiempo real y la sincronización entre instancias de cliente (figura 8).

Los resultados demostraron un funcionamiento estable y una comunicación fluida entre todos los usuarios, validando el diseño de la arquitectura multiusuario distribuida que serviría de base para las versiones posteriores integradas con gemelos digitales.

Este experimento permitió además analizar el comportamiento del sistema bajo carga intensiva y optimizar el motor de red para futuras aplicaciones educativas con mayor concurrencia.



Figura 8. *Prueba de la versión ESTER 4.0 durante el CIITI 2024 con un pico de hasta 50 usuarios conectados en simultáneo.*

Con la versión ESTER 2.x, se incorporó la visión en primera persona como una nueva modalidad de interacción dentro de los escenarios tridimensionales.

Esta innovación permitió que cada usuario pudiera observar el entorno desde la perspectiva del robot, incrementando el realismo de la simulación y favoreciendo la comprensión espacial durante las prácticas de control y navegación.

La función se aplicó por primera vez en el desafío de fútbol robótico, donde los estudiantes debían coordinar movimientos y estrategias basadas en lo que su robot “veía” dentro del entorno. Esta característica representó un avance hacia experiencias inmersivas y centradas en la percepción del agente, acercando el simulador a los principios de los entornos de realidad mixta y gemelos digitales interactivos (figura 9).



Figura 9. *Vista en primera persona del desafío de fútbol en el simulador ESTER 2.0.*

Resultados preliminares de pruebas de rendimiento:

- Latencia promedio entre envío de comando y ejecución en robot físico: 40–60 ms en red local.
- Desfase en la sincronización de datos entre robot y simulador: <100 ms percibido.
- Simulación fluida testeada hasta con 30 robots simultáneos: estable en equipos con GPU integrada.

Los tiempos de latencia observados en pruebas reales (40–60 ms en red local) y la estabilidad con 30 usuarios simultáneos confirman la viabilidad técnica del modelo híbrido, incluso en instituciones con limitaciones presupuestarias (Acker et al., 2023).

3.8. Seguridad, escalabilidad y rendimiento del sistema

El sistema fue diseñado bajo una visión integral que contempla:

3.8.1 Seguridad

- Validación de autenticación por ID único.
- Filtro de comandos simultáneos con buffer de control.
- Protección contra sobrecarga de red o eventos duplicados.

3.8.2 Escalabilidad

- Soporte para hasta 30 robots por sesión.
- Expansión a múltiples aulas mediante servidores por grupo.
- Integración con plataformas de aprendizaje compatibles con aplicaciones web.

Este enfoque modular ha sido recomendado en estudios sobre arquitectura robusta en entornos automatizados (Rodríguez & Castro, 2023).

3.8.3 Rendimiento

- Optimización de texturas, colliders y scripts para reducir el consumo de recursos.
- Compatibilidad con dispositivos móviles y tabletas.

3.8.4 Compatibilidad

- Optimización gráfica para asegurar funcionamiento fluido entre 30 y 60 FPS.
- Soporte para compilación en WebGL, Linux y Windows.
- Interfaz web adaptable a pantallas móviles y táctiles.

Estas características hacen posible su implementación en contextos escolares con recursos limitados, promoviendo la democratización del acceso a tecnologías emergentes (Stein et al., 2025; Singh et al., 2025).

3.8.4 Proyección de validación futura:

Se prevén futuras pruebas comparativas entre trayectorias simuladas y reales, análisis de latencia y pruebas de recuperación ante fallas. Estas pruebas permitirán mejorar la precisión, robustez y eficiencia del sistema, generando evidencia empírica que podrá utilizarse en publicaciones académicas y para su futura adopción institucional (Ouyang & Xu, 2024 ; Jacob et al., 2024).

- **Comparación entre trayectorias** físicas y simuladas: medir el error acumulado y la desviación en rutas preprogramadas.
- **Evaluación de tiempos de respuesta**: enviar una misma orden a ambos entornos y cronometrar ejecución.
- **Robustez ante errores**: inducir desconexiones y evaluar recuperación del sistema.

En este capítulo se presentó el diseño integral del entorno híbrido de gemelo digital orientado a la enseñanza de robótica, detallando las decisiones técnicas y tecnológicas adoptadas para garantizar

una interacción fluida y eficaz entre los componentes físicos y virtuales. La selección cuidadosa de plataformas, sensores, actuadores, protocolos de comunicación (API REST, WebSocket, MQTT) y el uso de herramientas avanzadas como Unity y SolidWorks permitieron alcanzar una solución robusta y escalable, capaz de soportar múltiples usuarios simultáneamente con baja latencia. Este diseño asegura no solo la fidelidad en la simulación del comportamiento robótico, sino también la flexibilidad pedagógica para adaptarse a diversos contextos educativos y niveles de aprendizaje, aspectos fundamentales que serán validados en la experiencia práctica descrita en los capítulos siguientes.

CAPÍTULO 4

Modelo educativo y experiencia didáctica

4.1. Fundamentación pedagógica: construccionismo, aprendizaje activo y significativo

El entorno ESTER y el modelo híbrido de gemelo digital se fundamentan en teorías educativas sólidas que han marcado la evolución de la pedagogía en el siglo XX y XXI. Su diseño responde a la necesidad de articular la tecnología educativa con enfoques que promueven la construcción activa del conocimiento, tales como el construccionismo, el aprendizaje activo, el aprendizaje significativo y la educación basada en competencias (EBC). Estos enfoques posibilitan que los estudiantes desarrollen competencias técnicas, cognitivas y actitudinales en entornos híbridos accesibles, escalables y orientados a la práctica. La evidencia empírica indica que los estudiantes logran mayor confianza y comprensión cuando los simuladores responden de manera coherente al comportamiento del robot físico (Acosta et al., 2022). Esta característica justifica la inclusión de pruebas prácticas en el modelo educativo híbrido de ESTER, donde cada algoritmo desarrollado puede validarse sin ambigüedad entre el mundo virtual y el físico.

También se plantea un modelo pedagógico basado en la rúbrica analítica que se estructura a partir del enfoque de competencias propuesto por Neil (2024) y se articula con los principios del *Project-Based Learning* (Bell, 2010; Thomas, 2000), donde la evaluación se concibe como un proceso formativo continuo. En este contexto, la robótica educativa fomenta la autonomía y la autorregulación (Ryan & Deci, 2000; Eguchi, 2014), con resultados positivos comprobados en entornos STEM (Condliffe et al., 2017) .

4.1.1 Constructivismo

El constructivismo, formulado por Jean Piaget en su teoría de la epistemología genética (PIAGET, 1979), plantea que el conocimiento se construye activamente a partir de la interacción entre el sujeto y el entorno. Esta perspectiva dio origen al construccionismo, propuesto posteriormente por Seymour Papert, quien extendió las ideas de Piaget hacia el uso de la tecnología y la inteligencia artificial como mediadores del aprendizaje (Saldarriaga et al., 2020).

Desde esta visión, el aprendizaje resulta más significativo cuando los estudiantes construyen objetos o artefactos que representan sus ideas, transformando la comprensión abstracta en una experiencia tangible.

En el ámbito de la robótica educativa, este enfoque adquiere especial relevancia, ya que los alumnos diseñan, programan y observan cómo sus creaciones cobran vida en robots virtuales y físicos. Diversos estudios confirman que la construcción de artefactos funcionales fortalece la motivación, el compromiso y la creatividad (Chen et al., 2023; Hu et al., 2024; Kuo et al., 2021).

El enfoque construccionista se ve además potenciado por el uso de entornos visuales accesibles como **ATOM+**, que permiten diseñar, probar, equivocarse y corregir de manera autónoma. De este modo, se favorece la autoeficacia y la apropiación de los saberes en procesos significativos de aprendizaje (Valls Pou et al., 2022; Valls Pou et al., 2022).

4.1.2 Aprendizaje activo

El aprendizaje activo coloca al estudiante en el centro del proceso formativo, destacando que "hacer" es tan importante como "saber". Tal como lo señalaron (Bonwell & Eison, 1991), las experiencias de aprendizaje deben involucrar al estudiante en actividades de análisis, síntesis y evaluación. En este sentido, ESTER promueve prácticas de programación repetibles, ajustables y orientadas a la resolución de problemas reales.

La interactividad del entorno, sumada a la posibilidad de experimentar en simultáneo con robots simulados y físicos, favorece la construcción activa del conocimiento (Fuller et al., 2020; Zhong et al., 2023). Asimismo, experiencias prácticas de programación y robótica integradas en proyectos han demostrado mejorar la motivación y el pensamiento creativo, al permitir a los estudiantes apropiarse de su proceso de aprendizaje(Kuo et al., 2021; Güleryüz & Dilber, 2021) .

4.1.3 Aprendizaje significativo

El modelo de aprendizaje significativo, desarrollado por Ausubel (1963) y ampliado por Novak (2010) con la teoría de los mapas conceptuales, plantea que los nuevos conocimientos se incorporan de manera efectiva cuando se relacionan con los saberes previos del estudiante. En concordancia con estas teorías, el diseño de ESTER estructura la progresión del aprendizaje en niveles que

conectan la teoría con la práctica, permitiendo que la experiencia simulada se complemente con la validación física.

Diversos estudios han demostrado que la integración de laboratorios virtuales y robótica educativa facilita la comprensión, reduce la carga cognitiva y mejora la motivación, especialmente en estudiantes sin formación técnica previa (Li & Liang, 2024; Panasiuk et al., 2021; Zhong et al., 2023). En este sentido, la programación visual, la retroalimentación inmediata y los escenarios auténticos del entorno híbrido favorecen la incorporación duradera del conocimiento y su transferencia a nuevas situaciones.

4.1.4 Educación basada en competencias (EBC) en el entorno ESTER

El enfoque de educación basada en competencias (EBC) se centra en el desarrollo de capacidades integradas y transferibles que permitan actuar de manera eficaz en contextos reales o simulados. Este paradigma, ampliamente difundido en la educación superior, se fundamenta en la noción de alineamiento constructivo propuesta por (Biggs, 1991), que plantea la necesidad de coherencia entre los resultados de aprendizaje, las actividades de enseñanza y las estrategias de evaluación.

El simulador ESTER fue diseñado para promover el desarrollo integral de competencias técnicas, cognitivas, actitudinales y metacognitivas en programación y robótica. Desde el plano técnico, facilita la adquisición de habilidades en control de robots, gestión de sensores y comunicación entre sistemas físicos y virtuales; en el cognitivo, estimula el razonamiento lógico, la resolución de problemas y la transferencia de conocimientos entre distintos lenguajes y entornos de simulación; en el actitudinal, fomenta la autonomía, la colaboración y la perseverancia ante la complejidad de las tareas; y en el metacognitivo, favorece la reflexión sobre las propias estrategias de aprendizaje y la optimización del desempeño a partir del análisis de resultados. De este modo, ESTER trasciende la función instrumental del software para convertirse en un mediador pedagógico coherente con el enfoque de educación basada en competencias y el principio del alineamiento constructivo (Biggs, 1991; Tobón, 2006).

Cada práctica en ESTER permite observar desempeños concretos, tales como:

- Programar estructuras de control funcionales.
- Resolver problemas de navegación autónoma o interacción robótica.
- Adaptar un algoritmo ante condiciones reales o simuladas imprevistas.

- Colaborar eficazmente en tareas distribuidas.

La inclusión de **ATOM+** (figura 10), facilita la participación de estudiantes con diversos perfiles académicos, sin disminuir los estándares de exigencia. Este diseño pedagógico se alinea con las demandas del siglo XXI, promoviendo pensamiento computacional, creatividad, colaboración y resolución autónoma de problemas (Valls Pou et al., 2022; S. Calderón et al., 2024; Tiryaki & Adigüzel, 2021) .

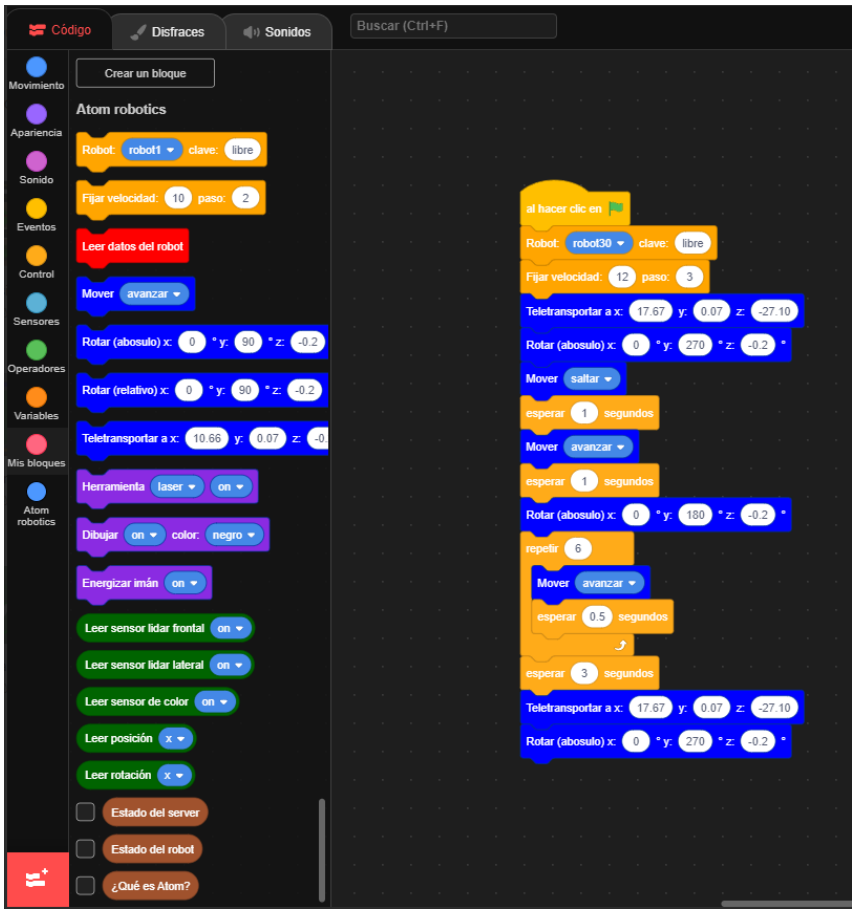


Figura 10., *Primera versión de la interfaz de programación visual ATOM integrada al simulador ESTER 2.3.*

En la versión ESTER 4.1, el entorno de programación visual evolucionó hacia una interfaz más robusta, intuitiva y completamente integrada en el simulador 3D.

El nuevo módulo ATOM+ incorporó un conjunto ampliado de bloques lógicos (figura 11), funciones de control condicional y estructuras de repetición, permitiendo a los estudiantes crear comportamientos más complejos sin necesidad de programación textual.

Además, se mejoró la interfaz de usuario con retroalimentación visual y sincronización directa entre los bloques de código y las acciones del robot dentro del entorno, lo que favoreció un aprendizaje más inmediato y experimental.

Esta versión consolidó el enfoque construccionista de la propuesta, al situar al alumno como diseñador activo de soluciones robóticas en un entorno accesible y visualmente motivador.

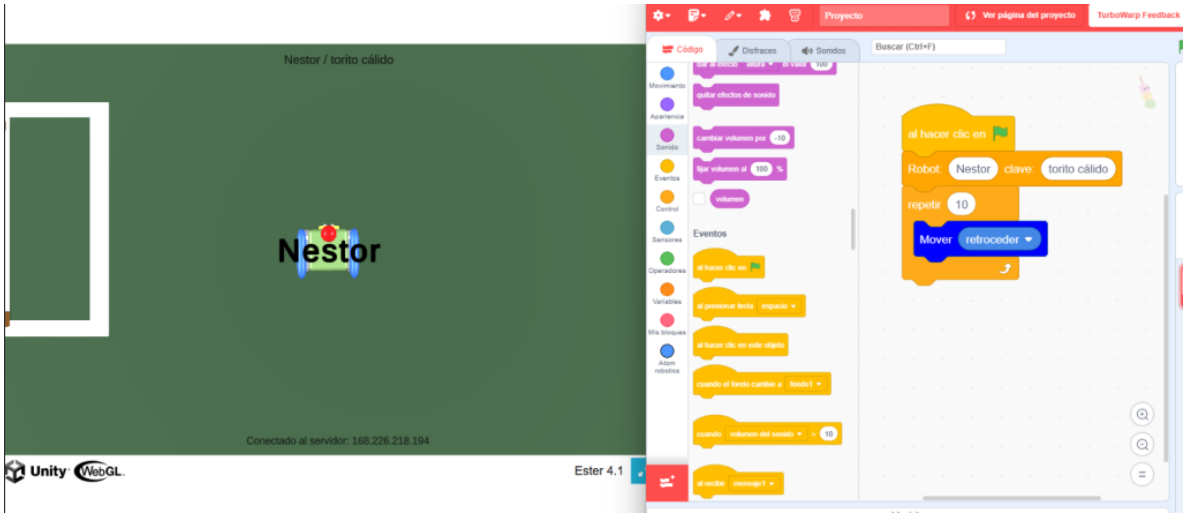


Figura 11. *Entorno de programación visual ATOM+ integrado al simulador ESTER, ejemplo de programa mover robot 10 pasos.*

4.2. Experiencias en carreras técnicas y no técnicas

El modelo híbrido de gemelo digital implementado en el entorno ESTER fue validado en contextos académicos diversos ya mencionado, que abarcaron tanto carreras técnicas universitarias como trayectos formativos de posgrado y programas destinados a estudiantes sin perfil informático. Esta diversidad de aplicaciones refleja la versatilidad de la propuesta y su capacidad de adaptarse a las demandas de diferentes públicos.

En términos pedagógicos, la implementación responde a enfoques activos y constructivistas que promueven el aprendizaje a través de la práctica situada y el diseño de proyectos (Bonwell & Eison, 1991; Papert, 1980). En el entorno del simulador ESTER, estos principios se concretan mediante experiencias donde los estudiantes programan, prueban y corrigen el comportamiento de robots virtuales y físicos en escenarios realistas, convirtiéndose en protagonistas de su propio proceso de aprendizaje.

Asimismo, el modelo se inscribe en la tradición del aprendizaje significativo (Ausubel, 1962; Novak, 2010), en tanto los nuevos conocimientos se integran con los saberes previos de los estudiantes y se transfieren a contextos auténticos de resolución de problemas. Esta dinámica favorece la comprensión profunda de los conceptos de programación y control, al tiempo que desarrolla la capacidad de aplicar el conocimiento en situaciones reales, consolidando así una formación orientada a la acción, la reflexión y la mejora continua.

4.2.1 Carreras técnicas

En la carrera de Ingeniería en Sistemas Informáticos de la Universidad Abierta Interamericana, el entorno se aplicó en la asignatura *Robótica* (tercer año), donde los estudiantes ya contaban con experiencias previas en hardware físico. La incorporación del simulador ESTER y de su modelo híbrido ofreció ventajas significativas:

- **Prácticas simultáneas en grupos numerosos**, resolviendo limitaciones comunes como la escasez de kits robóticos o la rotura de componentes.
- **Repetición autónoma de actividades fuera del aula**, lo que fortaleció la autoorganización y el aprendizaje autónomo.
- **Introducción de contenidos avanzados**, como sensores virtuales, redes neuronales y estructuras colaborativas entre robots, que resultaban difíciles de abordar solo con hardware físico.
- **Evaluación por competencias**, gracias al registro de proyectos y desempeños individuales, lo que permitió aplicar rúbricas integrales.

Estas prácticas se alinean con propuestas contemporáneas que resaltan el valor de los entornos híbridos y escalables para la enseñanza de disciplinas STEM, en los que la simulación se combina con la experimentación real para desarrollar pensamiento computacional, resolución de problemas y creatividad técnica (Fuller et al., 2020; Singh et al., 2025; Rodriguez & Castro, 2023)

4.2.2 Carreras no técnicas y posgrados

La flexibilidad del entorno también posibilitó su implementación en programas de formación docente y posgrado, donde muchos participantes no contaban con experiencia previa en

programación. Un caso relevante fue la aplicación en la Maestría en Tecnología Educativa (MTE) de la Universidad Abierta Interamericana, en la asignatura *Programación, robótica e impresión 3D*.

Gracias al módulo visual ATOM+, diseñado con bloques gráficos intuitivos, se lograron los siguientes resultados:

- **Alta motivación inicial**, incluso en docentes sin conocimientos técnicos previos.
- **Aprendizaje progresivo de estructuras de control y sensores**, con reducción de la carga cognitiva al emplear una interfaz visual (Zhong et al., 2023;Güleryüz & Dilber, 2021).
- **Comprensión significativa de principios de robótica**, validada mediante la observación directa del comportamiento del robot físico y su réplica virtual (Calderón et al., 2024; Hu et al., 2024).

Estas experiencias corroboran lo señalado por la literatura, que indica que los entornos virtuales y los laboratorios híbridos, cuando se diseñan con accesibilidad y retroalimentación inmediata, permiten que estudiantes no técnicos se apropien de conceptos complejos de manera eficaz (Li & Liang, 2024; Panasiuk et al., 2021; Valls Pou et al., 2022).

Desde una perspectiva educativa, esta implementación responde a la visión sobre la importancia de conectar lo nuevo con lo previo, y al enfoque (Biggs, 1996; Biggs, 2001) respecto al alineamiento constructivo, dado que los objetivos de aprendizaje, las actividades prácticas y las evaluaciones estuvieron estrechamente vinculados.

El entorno híbrido ESTER se consolida como una herramienta inclusiva que democratiza el acceso a la robótica y la programación, integrando estudiantes de perfiles técnicos y no técnicos en experiencias de aprendizaje activas, significativas y orientadas al desarrollo de competencias transferibles al ámbito profesional.

4.3. Programación visual con bloques integrada al entorno

Una de las barreras más comunes en la enseñanza de la robótica educativa es la dificultad inicial que enfrentan estudiantes sin experiencia previa en programación textual. La introducción abrupta a lenguajes formales puede generar ansiedad, baja motivación y abandono temprano. Con el propósito de superar este obstáculo, se desarrolló el módulo ATOM+ (Balich & Balich,

2025), basado en *Scratch* y adaptado en TurboWarp, que permite la programación visual mediante bloques gráficos.

La elección de la programación visual se fundamenta en teorías constructivistas y construccionistas que destacan la importancia de aprender a través de la creación de artefactos significativos. En esta línea, (Papert, 1980) sostiene que el conocimiento se construye de forma más efectiva cuando los estudiantes participan activamente en el diseño y la experimentación, mientras que (PIAGET, 1979)ya había señalado el valor del aprendizaje activo en etapas de desarrollo cognitivo.

Asimismo, la lógica de los bloques se vincula con el enfoque del aprendizaje significativo (Novak, 2010) **,** al facilitar que los nuevos conceptos de control y lógica algorítmica se relacionen con saberes previos de los estudiantes, reduciendo la carga cognitiva inicial y permitiendo una progresión gradual hacia lenguajes textuales como Python.

4.3.1 Características de ATOM+

El módulo ATOM+ fue diseñado específicamente para integrarse con el simulador ESTER, ofreciendo un entorno inclusivo y accesible, en el que estudiantes de distintos perfiles formativos puedan participar activamente. Sus características principales son:

- **Paleta de bloques personalizada**, con comandos específicos para movimiento, sensores, estructuras de control y comunicación con el entorno virtual.
- **Interfaz en español**, pensada para reducir barreras idiomáticas y culturales.
- **Ejecución online**, sin necesidad de instalar bibliotecas ni software adicional, lo que facilita la accesibilidad remota.
- **Alternancia entre bloques nativos de Scratch y bloques propios de ATOM+**, promoviendo una transición gradual hacia lenguajes textuales.

Estas características se alinean con propuestas de aprendizaje basado en proyectos (Bonwell & Eison, 1991; Tobón, 2006) y con la literatura reciente en robótica educativa, que resalta el valor de la programación visual para promover el pensamiento computacional, la creatividad y la resolución de problemas (Valls Pou et al., 2022; Güleryüz & Dilber, 2021; Calderón et al., 2024). La integración de ATOM+ al simulador ESTER, así como las experiencias iniciales de uso con docentes y estudiantes, fueron registradas en el video documental Simulador ESTER: evolución del proyecto 2021–2024

(Balich, 2024), donde se observan los primeros ejercicios de programación visual en entornos multipresenciales. Este material complementa las evidencias empíricas presentadas en este capítulo y refuerza la dimensión pedagógica del desarrollo.

En 2025, el entorno de programación visual ATOM+ incorporó un conjunto de bloques específicos para la gestión de gemelos digitales (figura 12), extendiendo sus capacidades más allá del control de robots simulados.

Estos nuevos módulos permitieron a los estudiantes interactuar directamente con los datos del robot físico, enviar comandos en tiempo real y visualizar la respuesta inmediata en el entorno 3D del simulador ESTER.

La integración de bloques de comunicación MQTT y WebSocket, junto con el manejo de variables de estado y sincronización, convirtió a ATOM+ en una plataforma visual de desarrollo para sistemas ciberfísicos educativos.

Este avance permitió vincular los principios del aprendizaje basado en proyectos con la programación visual y la robótica real, consolidando un entorno híbrido que unifica los planos virtual y físico en la enseñanza de la algoritmia y el control robótico.

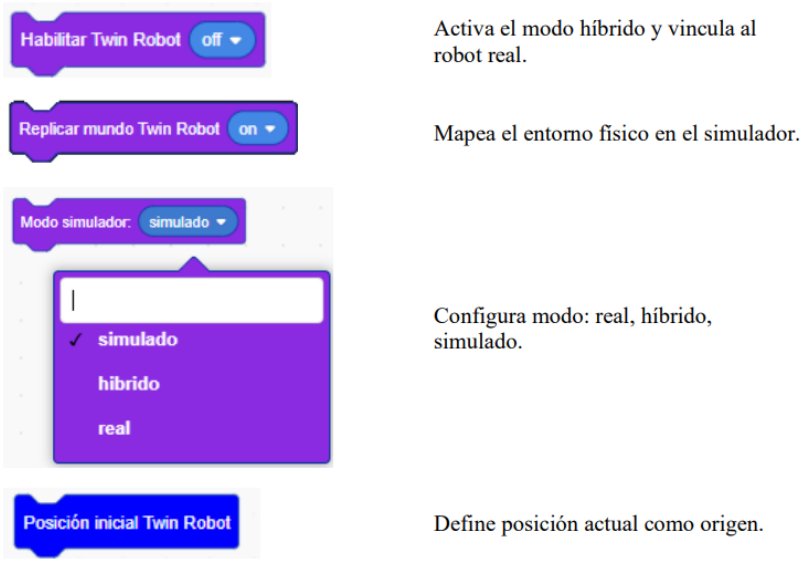


Figura 12. *Entorno ATOM+ con bloques de programación para la gestión de gemelos digitales y comunicación en tiempo real con el simulador ESTER.*

4.3.2 Integración con el simulador

Cada bloque de ATOM+ genera una instrucción que se comunica vía WebSocket con el backend del simulador ESTER. El servidor, desarrollado en Python, interpreta los comandos y actualiza en tiempo real el entorno virtual de Unity.

Este mecanismo permite que:

- El estudiante observe de inmediato el efecto de su código en el robot simulado.
- En modo híbrido, los comandos también se ejecuten sobre el robot físico (TWIN ROBOT), garantizando coherencia entre simulación y realidad.

Diversos estudios han señalado la importancia de este tipo de integración entre programación visual y entornos físicos para fortalecer la motivación, la creatividad y la transferencia de conocimientos (Chen et al., 2023; Kuo et al., 2021; Zhong et al., 2023).

4.3.3 Impacto pedagógico observado

La implementación de ATOM+ en el entorno ESTER evidenció múltiples beneficios pedagógicos:

- Reducción de la ansiedad inicial asociada a la programación textual.
- Incremento de la disposición a experimentar con estructuras complejas como bucles anidados, condicionales y sensores.
- Fortalecimiento del trabajo colaborativo, mediante la asignación de roles diferenciados en el diseño, prueba y depuración de programas.
- Desarrollo de competencias del siglo XXI, tales como pensamiento computacional, creatividad, resolución de problemas y colaboración.

Estas evidencias coinciden con investigaciones que han documentado el impacto positivo de los entornos de programación visual en estudiantes novatos y en docentes en formación, al promover la accesibilidad, la motivación y la construcción de aprendizaje (Tiryaki & Adigüzel, 2021Hu et al., 2024; Tiryaki & Adigüzel, 2021; Güleryüz & Dilber, 2021).

En este sentido, ATOM+ no constituye únicamente una interfaz gráfica, sino una estrategia pedagógica inclusiva coherente con el paradigma del alineamiento constructivo y con los modelos constructivistas que sitúan al estudiante como protagonista en la creación de conocimiento aplicable y transferible. Esta coherencia se evidencia en que los resultados de aprendizaje se expresan a través de la construcción de programas funcionales, las actividades de enseñanza se

orientan al diseño y experimentación de soluciones reales, y las evaluaciones se basan en el desempeño observable dentro del entorno de simulación. De este modo, ATOM+ materializa el principio de que el aprendizaje ocurre cuando el estudiante actúa, reflexiona y aplica el conocimiento en contextos auténticos, en consonancia con la propuesta (Biggs, 1999).

4.4. Actividades prácticas con el robot físico

La incorporación del TWIN ROBOT como gemelo físico del entorno simulado posibilitó la realización de experiencias prácticas que fortalecieron la integración entre programación, lógica de control y manipulación de sensores. Desde la perspectiva del construccionismo (Papert, 1980), la interacción con un artefacto tangible convierte el aprendizaje en un proceso activo y significativo, ya que los estudiantes pueden observar, analizar y corregir sus creaciones en un objeto real que refleja sus ideas.

Este enfoque se encuentra alineado también con la teoría del aprendizaje significativo de Ausubel (1968), en tanto que el contacto con situaciones concretas facilita la integración de nuevos conocimientos a la estructura cognitiva del estudiante. La posibilidad de comprobar hipótesis en un robot físico complementa el aprendizaje en el simulador, reduciendo la distancia entre lo teórico y lo práctico, y potenciando la transferencia de saberes a escenarios reales.

4.4.1 Diseño y ejecución de actividades

Las actividades prácticas se organizaron de acuerdo con un esquema de progresión gradual, propio de las metodologías activas y del aprendizaje basado en proyectos (ABP) (Bonwell & Eison, 1991; Tobón, 2006). Se inició con desafíos de control básico y se avanzó hacia escenarios colaborativos, autónomos e incluso asistidos por técnicas de inteligencia artificial. Entre los ejemplos más relevantes se incluyen:

- Seguimiento de línea negra con sensores infrarrojos.
- Evasión de obstáculos mediante sensores ultrasónicos.
- Navegación autónoma en laberintos, con comparación entre trayectorias simuladas y reales.
- Simulación de entornos colaborativos, como fábricas educativas con robots que ejecutan tareas secuenciales.

- Implementación de un control por red neuronal, entrenada en el entorno virtual y validada posteriormente en el robot físico.

Estas prácticas se fundamentan en el principio de que “aprender haciendo” promueve una comprensión más profunda y duradera (PIAGET, 1979; Saldarriaga et al., 2020).

4.4.2 Justificación pedagógica

El valor pedagógico de estas actividades radica en la posibilidad de comparar y contrastar el comportamiento del robot físico con el de su gemelo digital. Esta experiencia en espejo facilita el desarrollo de pensamiento crítico y de habilidades analíticas, al permitir que los estudiantes validen hipótesis, detecten discrepancias y ajusten algoritmos. La integración entre simulación y realidad potencia la transferencia del conocimiento y fortalece la autonomía del aprendizaje (Panasiuk et al., 2021).

Además, la organización del trabajo grupal mediante rotación de roles (programador, operador, evaluador) responde a la necesidad de desarrollar competencias sociales y colaborativas, identificadas como esenciales en la formación STEAM (Güleryüz & Dilber, 2021; . Calderón et al., 2024; Valls Pou et al., 2022). Estas dinámicas fomentan la comunicación técnica, la coordinación y el liderazgo compartido, en coherencia con el paradigma de la educación basada en competencias (Biggs, 1999; Tobón, 2006).

4.4.3 Impacto observado

Los resultados observados durante las clases utilizando el simulador ESTER (Balich et al., 2024)i indican un incremento notable en la motivación, la creatividad y la retención del conocimiento cuando los estudiantes interactúan con un robot físico que refleja sus programas. La experiencia tangible refuerza la percepción de logro y confianza, aspectos ya señalados como claves para el aprendizaje autónomo y significativo (Ausubel, 1963; Novak, 2010).

La evidencia empírica obtenida en base a las encuestas realizadas a alumnos coincide con hallazgos previos que destacan cómo la robótica educativa, híbrida al combinar simuladores y robots reales, eleva el compromiso estudiantil, reduce la ansiedad frente al error y favorece la apropiación de competencias técnicas y cognitivas (Chen et al., 2023; Tiryaki & Adigüzel, 2021 ; Hu et al., 2024) .

4.5. Estudio de caso: aplicación en Ingeniería en Sistemas

La implementación del entorno híbrido ESTER, junto con el robot físico TWIN ROBOT, se aplicó en la asignatura *Robótica* de la carrera de Ingeniería en Sistemas Informáticos de la Universidad Abierta Interamericana. Esta experiencia permitió articular simulación avanzada, programación visual y robótica física en una propuesta coherente con los objetivos de formación profesional en ingeniería.

Desde la perspectiva pedagógica, el estudio de caso se fundamenta en el aprendizaje significativo de Ausubel al vincular conceptos abstractos de programación y control con experiencias prácticas observables en el robot. Asimismo, responde al enfoque del construccionismo, dado que los estudiantes construyen conocimiento al diseñar algoritmos que se materializan en artefactos tangibles. El modelo se enmarca también en el paradigma del alineamiento constructivo.

4.5.1 Contexto

- **Materia:** Robótica (tercer año de la carrera).
- **Modalidad:** mixta, con clases teóricas presenciales y prácticas virtuales y físicas.
- **Cohortes:** entre 25 y 30 estudiantes por comisión.
- **Implementación progresiva del simulador ESTER**, como complemento y reemplazo de kits físicos tradicionales.

Este diseño responde a la necesidad de incorporar metodologías activas y tecnológicas que favorezcan el desarrollo de competencias del siglo XXI, tales como el pensamiento computacional, la resolución de problemas y la colaboración interdisciplinaria (Bonwell & Eison, 1991) .

4.5.2 Actividades implementadas

- Durante el cuatrimestre, se desarrollaron prácticas semanales orientadas a:
- El diseño de algoritmos de control y navegación.
- El uso de sensores simulados y reales (ultrasónicos, infrarrojos, bumpers).
- La programación de conductas de evasión de zonas y colaboración entre robots.
- La integración de estructuras algorítmicas avanzadas, como bucles y condicionales.

La posibilidad de trabajar en modo híbrido permitió validar en condiciones reales los programas desarrollados previamente en el simulador, consolidando el vínculo entre lo digital y lo físico. Este enfoque se corresponde con el aprendizaje basado en proyectos (ABP), donde los estudiantes resuelven problemas auténticos y generan productos funcionales como evidencia de su desempeño. Esto ha demostrado ser una estrategia eficaz para fortalecer el pensamiento computacional, el diseño de soluciones y la creatividad técnica en entornos formativos (Zhong et al., 2023; . A. R. Calderón, 2024; Singh et al., 2025).

4.5.3 Resultados observados

- Aumento del tiempo efectivo de práctica individual y grupal.
- Reducción de fallas técnicas, al permitir pruebas iniciales en el simulador antes de pasar al hardware.
- Mejora en la comprensión de los sistemas de control, sensores y estructuras algorítmicas.
- Mayor confianza del estudiantado al presentar proyectos funcionales, gracias a la posibilidad de ensayo y error sin consecuencias en el hardware físico.

Estos resultados coinciden con estudios que han demostrado cómo la integración de robots programables y entornos híbridos puede potenciar el aprendizaje, reducir la ansiedad ante el error y aumentar el compromiso estudiantil (Chen et al., 2023; Kuo et al., 2021; Tiryaki & Adigüzel, 2021).

La experiencia también evidencia la relevancia del alineamiento constructivo (Biggs, 1991), dado que los logros de los estudiantes se mantuvieron en clara correspondencia con los objetivos de aprendizaje y las actividades de enseñanza previstas en el diseño pedagógico. Esta coherencia se tradujo en una mejora observable en la aplicación práctica de los conocimientos, validando el principio de que el aprendizaje significativo ocurre cuando el estudiante actúa, reflexiona y aplica lo aprendido en contextos auténticos.

En este sentido, la posibilidad de programar una sola vez y ejecutar en dos entornos “virtual y físico” se consolidó como un factor clave para la apropiación y transferencia del conocimiento, favoreciendo la comprensión profunda de los conceptos y la retroalimentación inmediata del propio desempeño, tal como lo señalan experiencias educativas afines en la literatura (Fuller et al., 2020; Singh et al., 2021);

4.5.4 Escalabilidad del sistema con múltiples robots físicos

El análisis de la escalabilidad del sistema es un aspecto crítico para entornos de robótica colaborativa, donde múltiples robots físicos deben interactuar de manera simultánea con el

simulador y entre ellos. Para evaluar este comportamiento, se realizaron pruebas incrementando progresivamente el número de robots conectados al simulador Ester, midiendo la latencia de respuesta, la sincronización y la consistencia de los datos enviados y recibidos.

Los resultados obtenidos se resumen en la siguiente tabla:

Tabla 2: *Latencia por conexión de robots.*

Número de robots	Latencia promedio	Sincronización	Modelo híbrido ESTER
1 a 5	30 a 100 ms	Efectiva	No se aprecia desincronización.
5 a 10	30 a 100 ms	Efectiva	No se aprecia desincronización.
10 a 15	30 a 100 ms	Efectiva	Desincronización por API (TCP).
15 a 20	100 a 300 ms	Leve	Desincronización por API (TCP).
20 a 25	100 a 400 ms	Moderada	Desincronización por API (TCP).
25 a 30	100 a 500 ms	Moderada	Desincronización por API (TCP) y WebSocket.
Más de 30	1000 a 2000 ms	Fuerte	Desincronización por API (TCP) y WebSocket.

Análisis de los resultados

Durante las pruebas se observó que, cuando los alumnos ejecutaban programas independientes para cada robot en distintos entornos colaborativos (Collab), los robots no siempre comenzaban sincronizados en desafíos colectivos como la carrera. Esto se debe a que la comunicación mediante API permite que un robot comience su ejecución antes que otro, independientemente del orden de ejecución del programa.

Para resolver esta situación, los alumnos implementaron un único programa Python utilizando hilos (threading), asignando un hilo independiente para cada algoritmo de robot. Una alternativa para futuras versiones es establecer comunicación directa entre robots para garantizar sincronización.

Hasta 15 robots, no se observaron grandes retrasos en el procesamiento individual. El encolamiento de mensajes en el servidor de Mirror para Unity permite un movimiento fluido de los robots, y la

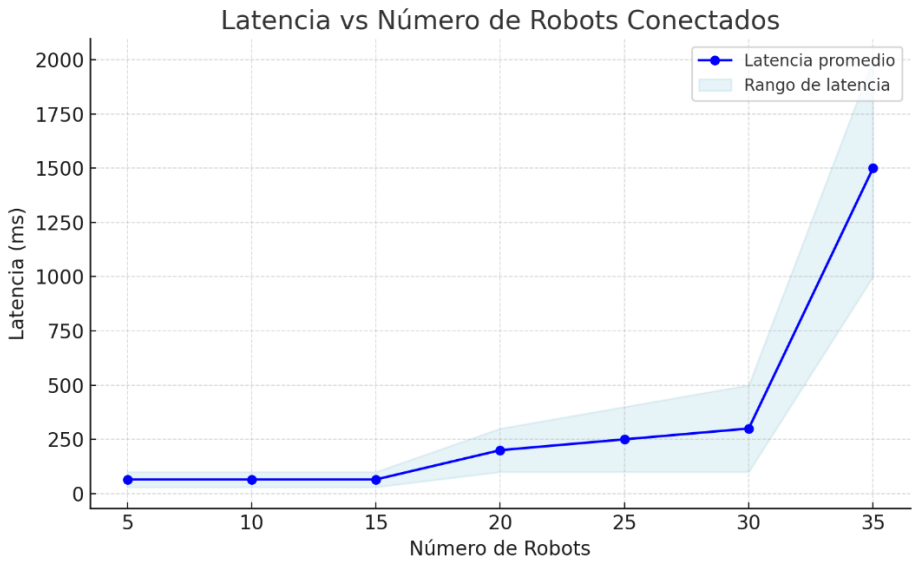
combinación TCP sobre WebSocket con timestamp garantiza que no se pierdan órdenes de control ni se altere la secuencia de ejecución.

Al conectar más de 20 robots, la latencia aumenta y comienza a impactar también sobre el servidor Socket, con retrasos de hasta 2 segundos en actividades individuales. Estas pruebas se realizaron inicialmente mediante conexión a Internet, lo que genera dependencia de la red de cada alumno. Posteriormente, las pruebas se repitieron en clase, con todos los alumnos conectados desde la misma red utilizando el cliente web y de escritorio Windows. La aplicación de escritorio resultó más estable, recuperando la conexión ante errores de manera más eficiente. En modalidad intranet con servidor interno, entorno de desarrollo local en Python o Atom+ local, los retardos disminuyeron significativamente, manteniendo la latencia de 10 a 100 ms incluso con hasta 30 robots conectados.

Gráfico de latencia vs número de robots

A continuación, se presenta un gráfico simulado que ilustra la relación entre el número de robots conectados y la latencia promedio observada:

Gráfico 1: *Latencia por conexión de robots.*



Como puede observarse, la latencia se mantiene baja y estable hasta 15 robots, comienza a aumentar entre 15 y 25 robots, y alcanza valores más elevados cuando se supera el umbral de 30 robots, confirmando los datos de la tabla anterior.

Conclusión

El sistema Ester es capaz de soportar hasta 20 robots físicos de manera eficiente, manteniendo latencias bajas y sincronización adecuada. Con más de 20 robots, la latencia y la desincronización aumentan, pero existen estrategias claras para mejorar la escalabilidad: optimización del envío de mensajes, distribución del dispatcher en múltiples instancias y segmentación de la comunicación por grupos de robots. Además, la modalidad intranet y el uso de clientes de escritorio contribuyen a minimizar los efectos de la latencia y mejorar la estabilidad general del sistema.

4.6. Estudio de caso: aplicación en posgrados en Tecnología Educativa

En el marco de la Maestría en Tecnología Educativa de la Universidad Abierta Interamericana, se implementó el entorno híbrido ESTER junto con el módulo de programación visual ATOM+ durante la asignatura *Programación, robótica e impresión 3D*. Este estudio de caso permitió explorar el impacto de los entornos híbridos en la formación de docentes provenientes de disciplinas no técnicas, muchos de ellos sin experiencia previa en programación.

Desde la perspectiva teórica, esta experiencia se fundamenta en la teoría del aprendizaje significativo de Ausubel (1968), revisitada en estudios contemporáneos (Bryce & Blown, 2023), que sostienen que los nuevos conocimientos adquieren sentido cuando se integran con los saberes previos del aprendiz. En esta línea, los conceptos de control y programación se vincularon con las experiencias previas de los docentes participantes dentro de sus propios contextos escolares. Asimismo, se inscribe en el construccionismo de Papert, al situar a los estudiantes como creadores activos de proyectos didácticos funcionales mediante la experimentación y la resolución de problemas. Finalmente, responde al paradigma del alineamiento constructivo propuesto por (J. Biggs, 1999) , delineado los objetivos, actividades y evaluaciones se diseñaron de manera coherente para promover competencias observables en escenarios auténticos de aprendizaje. os objetivos de aprendizaje se formularon en función de competencias observables vinculadas al diseño, programación y control de robots; las actividades promovieron la aplicación práctica de esos saberes en entornos simulados y reales; y las evaluaciones se centraron en evidenciar el desempeño y la transferencia del conocimiento mediante proyectos funcionales y reflexiones sobre el proceso. Esta articulación permitió garantizar que cada componente del proceso formativo contribuyera de manera integrada al logro de aprendizajes significativos y aplicables.

4.6.1 Modalidad

El curso se desarrolló en modalidad mixta, con una duración intensiva de ocho clases. Las características principales fueron:

- **Acceso remoto al simulador mediante navegador web**, sin necesidad de instalación adicional.
- **Uso de ATOM+ para la programación visual**, con bloques gráficos intuitivos adaptados al español.
- **Combinación de actividades virtuales y presenciales**, lo que permitió a los participantes experimentar tanto en entornos simulados como en interacciones con el robot físico.

Esta organización favoreció la inclusión de docentes con diferentes trayectorias académicas, en coherencia con la propuesta de educación basada en competencias (Tobón, 2006, 2013), donde lo relevante es la capacidad de aplicar conocimientos en contextos variados y transferibles.

La combinación de bloques gráficos y simulación en tiempo real permitió a los participantes involucrarse activamente, incluso sin experiencia previa en robótica o programación. Esta estrategia pedagógica se sustenta en la evidencia sobre los beneficios de los laboratorios virtuales y entornos accesibles para promover la motivación, el compromiso y la autoeficacia (Li & Liang, 2024; Panasiuk et al., 2021) .

4.6.2 Actividades destacadas

Las prácticas se diseñaron con un enfoque de aprendizaje basado en proyectos (ABP) y aprendizaje activo (Bonwell & Eison, 1991), con énfasis en la resolución de problemas y en la creación de propuestas didácticas contextualizadas. Entre las principales actividades se incluyeron:

- Programación de trayectorias simples mediante bloques.
- Verificación visual y funcional del código en el simulador.
- Exploración de sensores y estructuras de control condicional.
- Desarrollo de mini proyectos educativos, contextualizados en las escuelas donde los docentes ejercen.

Estas actividades estimularon la creatividad, la reflexión didáctica y la integración de tecnologías accesibles en entornos educativos reales. Experiencias similares han demostrado que la robótica

educativa gamificada y los entornos interactivos promueven altos niveles de motivación y pensamiento creativo (Chen et al., 2023; Tiryaki & Adigüzel, 2021; Kuo et al., 2021).

4.6.3 Resultados

Los resultados observados durante la experiencia reflejan un alto impacto pedagógico:

- **Participación sostenida** en todas las instancias prácticas.
- **Alta motivación inicial y final**, incluso en docentes sin antecedentes técnicos.
- **Desarrollo de propuestas innovadoras** para aplicar la robótica en distintos niveles educativos.
- **Valoración positiva del entorno**, especialmente en términos de accesibilidad, comprensión y aplicabilidad.

Los resultados obtenidos coinciden con hallazgos recientes que destacan el rol transformador de los entornos simulador en educación con la retroalimentación de los alumnos (Anexo E), especialmente cuando se aplican en carreras de formación docente o en programas de posgrado (Zhong et al., 2023; Güleryüz & Dilber, 2021; Calderón, 2024).

Este caso refuerza la premisa de que el acceso a plataformas tecnológicas intuitivas, como ATOM+, integradas con simulación y robótica física, democratiza el aprendizaje de la programación y la robótica, reduciendo las brechas entre perfiles técnicos y no técnicos, y promoviendo la alfabetización digital en contextos educativos diversos (Valls Pou et al., 2022; Fuller et al., 2020).

4.6.4 Reflexión pedagógica

Este estudio de caso refuerza la premisa de que plataformas accesibles e inclusivas, como ESTER y ATOM+, democratizan el aprendizaje de la programación y la robótica en la educación superior, reduciendo las brechas entre perfiles técnicos y no técnicos. Asimismo, se muestra que la robótica educativa, cuando es integrada en un marco de aprendizaje significativo, activo y basado en competencias, puede convertirse en un recurso innovador para la didáctica en contextos diversos.

4.7. Impacto pedagógico: aprendizajes observados y percepción estudiantil

La implementación del entorno híbrido ESTER, combinado con el módulo ATOM+ y el robot físico TWIN ROBOT, fue acompañada por un proceso sistemático de observación y recolección de datos cualitativos y cuantitativos. El objetivo fue evaluar los aprendizajes logrados y la percepción estudiantil en diferentes contextos formativos, tanto de grado como de posgrado.

Este diseño responde a los principios del alineamiento constructivo (Biggs, 1999), en la medida en que los objetivos de aprendizaje, las actividades y las evaluaciones estuvieron estrechamente vinculados, permitiendo evidenciar de manera objetiva el desarrollo de competencias. Asimismo, se apoya en el enfoque de educación basada en competencias, al considerar dimensiones técnicas, cognitivas, actitudinales y metacognitivas como parte integral del desempeño del estudiante.

4.7.1 Diseño de rubricas de evaluación

Se desarrollaron rúbricas específicas (capitulo 8) para medir el desempeño en el entorno híbrido, contemplando las siguientes dimensiones:

- **Técnicas:** funcionamiento del código, precisión en la ejecución y desempeño del robot.
- **Cognitivas:** aplicación de lógica, resolución de problemas y transferencia de conocimientos.
- **Actitudinales:** autonomía, colaboración y disposición hacia el trabajo grupal.
- **Metacognitivas:** capacidad de reflexión, diagnóstico de errores y mejora continua.

Este esquema se alinea con la tradición (Bryce & Blown, 2023) significativo de Ausubel (Bryce & Blown, 2023) porque prioriza la comprensión y la integración de nuevos conocimientos con las estructuras cognitivas previas del estudiante, en lugar de la simple memorización de contenidos. En este enfoque, la evaluación no se limita a medir resultados finales, sino que indaga en cómo el alumno establece relaciones conceptuales, aplica lo aprendido y reconstruye su conocimiento en contextos auténticos. Asimismo, se vincula con la visión (Novak, 2010), quien enfatiza la importancia de hacer visibles los procesos de construcción de significados, permitiendo que los estudiantes reflexionen sobre sus propias representaciones mentales y las modifiquen a través de la práctica y la retroalimentación.

Este enfoque evaluativo se alinea con las tendencias actuales en formación por competencias, en donde el rendimiento del estudiante se mide no solo en función del resultado técnico, sino de su

proceso de aprendizaje y adaptación a entornos dinámicos (O. A. R. Calderón, 2024; Güleryüz & Dilber, 2021). Resultados completos disponibles en Anexo E).

4.7.2 Aprendizajes observados

A lo largo de las experiencias en grado y posgrado se documentaron aprendizajes significativos en varios niveles:

- **Comprensión sólida de estructuras algorítmicas**, como condicionales, bucles y sensores.
- **Incremento de la autonomía y la confianza**, al resolver problemas robóticos de forma progresiva y con retroalimentación inmediata.
- **Transferencia exitosa del código** entre el entorno simulado y el robot físico, evidenciando apropiación del modelo de gemelo digital.
- **Aplicación creativa de los bloques de ATOM+**, con propuestas originales y adaptadas a los contextos educativos de los estudiantes.

Estos hallazgos confirman lo planteado por (Saldarriaga et al., 2020) respecto al poder de aprender construyendo, y coinciden con investigaciones recientes que destacan cómo los entornos de programación visual reducen la carga cognitiva y promueven aprendizajes significativos incluso en estudiantes sin experiencia técnica (Zhong et al., 2023; Li & Liang, 2024).

4.7.3 Percepción estudiantil (encuestas anónimas)

Las encuestas anónimas realizadas en contextos de grado y posgrado revelaron una percepción altamente positiva del entorno híbrido:

- El **92%** consideró útil la posibilidad de alternar entre robot virtual y físico.
- El **87%** destacó que la herramienta le permitió aprender a su propio ritmo.
- El **95%** valoró la interfaz visual de ATOM+ como facilitadora del aprendizaje.
- El **90%** expresó interés en continuar explorando robótica e inteligencia artificial.

Estos indicadores confirman lo señalado por la literatura sobre la motivación y el compromiso que generan los entornos interactivos y accesibles, especialmente cuando integran gamificación y retroalimentación inmediata (Chen et al., 2023; Tiryaki & Adigüzel, 2021; Calderón, 2024).

4.7.4 Reflexión pedagógica

Los resultados observados evidencian que el entorno híbrido ESTER potencia no solo el aprendizaje técnico y conceptual, sino también el desarrollo de competencias del siglo XXI, como el pensamiento computacional, la creatividad, la colaboración y la resolución autónoma de problemas (Kuo et al., 2021; Valls Pou et al., 2022; Hu et al., 2024) .

La combinación de programación visual, retroalimentación inmediata y experiencias prácticas con robots físicos confirma lo planteado por las teorías del aprendizaje activo (Bonwell & Eison, 1991) y del aprendizaje significativo (Ausubel, 1968; Novak, 1998), en tanto que sitúa a los estudiantes como protagonistas de su proceso formativo.

En síntesis, la percepción positiva del alumnado y los aprendizajes observados validan el modelo híbrido como una estrategia pedagógica innovadora, inclusiva y replicable en distintos contextos académicos, con potencial para transformar la enseñanza de la programación y la robótica.

4.8 Conclusión

La implementación del entorno ESTER, en combinación con el modelo híbrido de gemelo digital y la plataforma de programación visual ATOM+, permitió articular tecnología educativa de vanguardia con fundamentos pedagógicos sólidos. Este enfoque generó una experiencia formativa integral, en la que la robótica, la programación y el pensamiento computacional dejaron de ser herramientas exclusivas de carreras técnicas para convertirse en recursos inclusivos, accesibles y adaptables a múltiples perfiles académicos.

Desde la perspectiva teórica, el modelo se sustenta en el construccionismo (Saldarriaga et al., 2020), que enfatiza el valor de aprender construyendo artefactos significativos; en el aprendizaje significativo de Ausubel (Bryce & Blown, 2023) y las contribuciones (Novak, 2010), que destacan la importancia de vincular nuevos conocimientos con estructuras previas; y en el alineamiento constructivo (Biggs, 1999), que propone la coherencia entre objetivos, actividades y evaluación. Asimismo, la propuesta responde al paradigma de la educación basada en competencias (EBC) (Tobón, 2006), al centrarse en el desarrollo de habilidades técnicas, cognitivas, actitudinales y metacognitivas transferibles a contextos profesionales reales.

Los resultados empíricos obtenidos en las experiencias de grado y posgrado validan esta integración (anexo E): se observaron mejoras significativas en la comprensión de estructuras algorítmicas, un incremento en la autonomía y la motivación estudiantil, y un fortalecimiento de competencias del siglo XXI, como la creatividad, la colaboración y la resolución de problemas (Kuo et al., 2021; Valls Pou et al., 2022; Hu et al., 2024).Estas evidencias coinciden con investigaciones que destacan el impacto positivo de los entornos híbridos y de la robótica educativa en la democratización del acceso al conocimiento tecnológico (Fuller et al., 2020; Chen et al., 2023) .

Además, la posibilidad de trabajar en espejo "donde lo físico y lo virtual reflejan sus estados" demostró ser una estrategia eficaz para superar limitaciones materiales y logísticas, permitiendo a los estudiantes repetir experiencias prácticas sin depender de hardware costoso ni de recursos físicos limitados (Panasiuk et al., 2021; Zhang et al., 2023). Esto refuerza la idea de que los laboratorios híbridos constituyen una solución sostenible y escalable para la educación superior.

El modelo híbrido de gemelo digital implementado en ESTER constituye una propuesta educativa innovadora y replicable, que integra teorías pedagógicas consolidadas con tecnologías emergentes. Al situar a los estudiantes como protagonistas de su aprendizaje, la propuesta demuestra que la robótica educativa puede transformarse en una herramienta democratizadora del conocimiento, capaz de motivar y formar a las nuevas generaciones en competencias clave del siglo XXI: pensamiento crítico, resolución de problemas, colaboración y adaptabilidad en entornos tecnolóGGüleryüz & Dilber, 2021; Zhong et al., 2023) ; Güleryüz & Dilber, 2021; Tiryaki & Adigüzel, 2021).

CAPÍTULO 5

Desarrollo de modelo híbrido de gemelo digital

5.1. Fundamentos del modelo híbrido propuesto

La relevancia de este estudio radica en proporcionar una solución tecnológica y educativa innovadora que supera las limitaciones inherentes a métodos exclusivamente virtuales o puramente físicos. El modelo híbrido propuesto no solo mejora el aprendizaje práctico y significativo en robótica, sino que también favorece el desarrollo del pensamiento computacional, la resolución autónoma de problemas y la adaptación a entornos tecnológicos cambiantes. Este enfoque responde directamente a las tendencias internacionales en educación superior y a las competencias profesionales más demandadas en la actualidad (Singh et al., 2021;Zhou et al., 2022).

La propuesta de un modelo híbrido de gemelo digital en el simulador ESTER puede ser profundamente fortalecida desde el enfoque de la Ingeniería de Abstracciones (Abstraction Engineering), una disciplina emergente que proporciona herramientas teóricas y metodológicas para construir y manipular representaciones (abstracciones) útiles para sistemas complejos y adaptativos como los gemelos digitales. Tal lo expuesto por Bencomo (2024), las abstracciones deben ser diseñadas no solo para representar estados estáticos, sino también para adaptarse a entornos dinámicos, incorporando datos en tiempo real, contextos de incertidumbre y múltiples niveles de análisis. Esto se alinea con la estructura propuesta en ESTER, donde la representación del entorno simulado y físico es constantemente ajustada en base a datos del robot real.

El modelo híbrido puede entenderse como una instancia clara de "abstracciones dinámicas de ingeniería" que combinan elementos diseñados por expertos (como la estructura del entorno simulado en Unity) con datos descubiertos o inferidos automáticamente (como las mediciones en tiempo real del robot físico), tal como lo plantea la anatomía de la abstracción en Abstraction Engineering. Además, el objetivo pedagógico de reducir la brecha entre simulación y realidad se beneficia directamente del uso de abstracciones configurables por distintos actores educativos (docentes, estudiantes, sistema), lo que permite una adaptación contextual, característica esencial según el enfoque de Bencomo (2024).

A diferencia de los modelos unidireccionales, en los que los datos fluyen únicamente del entorno físico a la digital, el enfoque híbrido propuesto en ESTER establece una retroalimentación continua que permite:

Reflejar las acciones del robot físico en su gemelo simulado.

- Modificar el comportamiento del robot simulado en función de datos críticos del robot físico.
- Sincronizar comportamientos, alertas y decisiones en ambos entornos (Grieves, 2024; Kartashova et al., 2020).

El propósito de este modelo incluye:

- Reducir el costo y el riesgo de pruebas físicas mediante la validación previa en simulación (Kurniawan et al., 2018)
- Extender el acceso a prácticas reales en contextos educativos con limitaciones materiales.
- Facilitar la transición desde entornos visuales de aprendizaje hacia programación en robots reales.
- Optimizar el proceso de aprendizaje mediante experiencias auténticas en sistemas ciberfísicos (Selim et al., 2024).

Este modelo se sustenta en tres pilares teóricos:

- **CPS (Cyber-Physical Systems),** donde lo físico y lo digital se integran como un solo sistema (Singh et al., 2021)
- **Robótica distribuida y cooperativa**, aprovechando arquitecturas en red que permiten la interacción entre múltiples agentes (W. Chen et al., 2018)
- **Construccionismo**, que enfatiza el aprendizaje activo mediante la creación de artefactos significativos (Kafai & Resnick, 2012).

La versión 4.2 del simulador ESTER representó un punto de inflexión hacia la creación del modelo híbrido de gemelo digital de la versión 5.0, al incorporar la interoperabilidad con dispositivos externos (13).
Esta etapa permitió el control remoto del entorno simulado desde teléfonos celulares y tablets, estableciendo la primera conexión directa entre interfaces móviles y el sistema central en la nube. El experimento demostró la viabilidad de integrar hardware físico dentro del ecosistema del simulador, sentando las bases para la posterior conexión con el Twin Robot.
De esta forma, el simulador evolucionó desde un entorno puramente virtual hacia un sistema ciberfísico educativo, donde las acciones ejecutadas desde un dispositivo externo impactan directamente sobre la simulación en tiempo real.



Figura 13. *Control remoto del simulador ESTER 4.2 desde un teléfono celular conectado al entorno cloud.*

El primer prototipo del Twin Robot fue desarrollado en base a diseños de robots propios iniciados en 2024 desde la versión 1 (figura 14) hasta la versión 2 de 2025, robot económico diferencial (figura 16) con soporte a teléfono celular hasta la configuración avanzada omnidireccional (figura 16), como parte de las pruebas iniciales para establecer la conexión entre el entorno virtual del simulador ESTER y un robot físico real.

Este modelo experimental, construido con componentes modulares y piezas impresas en 3D, permitió validar los conceptos de sincronización básica y transmisión de comandos entre ambos sistemas.
Su diseño incluyó una base de tracción diferencial, un sistema de servomotores controlados por una placa ESP32. Wemos D1, Arduino y Raspberry PI y una estructura adaptable para incorporar distintos sensores, lo que facilitó la evolución hacia versiones más avanzadas del modelo híbrido.

El prototipo cumplió una función clave como plataforma de prueba y validación del flujo de datos, sirviendo de puente entre el simulador y la futura arquitectura bidireccional de comunicación en tiempo real.

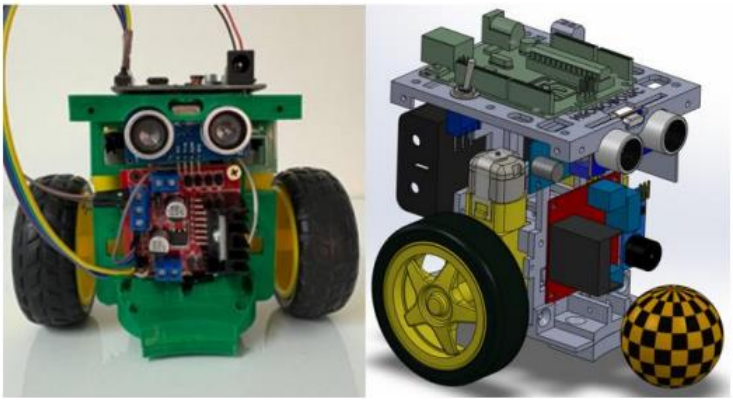


Figura 14. *Modelo de robot real de versión 1.*

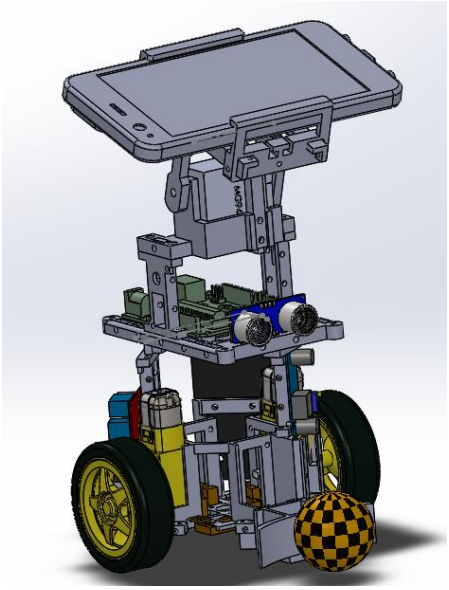


Figura 15. *Modelo versión 2 configuración diferencial de robot físico para el simulador ESTER.*

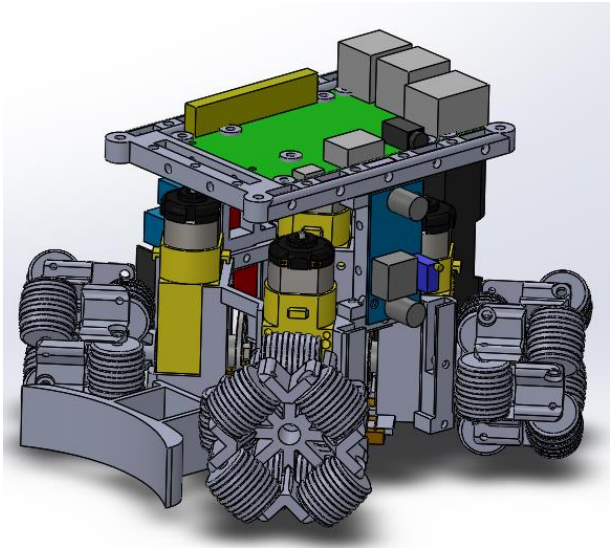


Figura 16. *Modelo versión 2 configuración omnidireccional de robot físico para el simulador ESTER.*

5.2. Interacción simulador ↔ robot físico: arquitectura y flujos

La interacción entre el simulador y el robot físico constituye el núcleo operativo del modelo híbrido propuesto, ya que permite la convergencia entre el entorno digital ESTER y los dispositivos robóticos reales, asegurando coherencia en tiempo real entre ambos planos.

La arquitectura del entorno ESTER se basa en un sistema de comunicación modular y robusto que integra REST, WebSocket y MQTT para garantizar sincronización baja en latencia, control bidireccional y actualización continua del estado del robot y su gemelo digital (Figura 17).

Este enfoque responde a los principios de diseño de gemelos digitales aplicados a sistemas autónomos y ciber-físicos, donde la consistencia temporal y la retroalimentación inmediata son esenciales para la toma de decisiones y la evaluación del comportamiento del sistema ((Zhang et al., 2023; Grieves, 2024).

De esta manera, el modelo adopta una arquitectura orientada a eventos y basada en flujos de datos continuos, alineada con estándares industriales emergentes para DT distribuidos y robótica educativa de alto rendimiento.

- API REST: empleada para el envío de comandos discretos y configuraciones iniciales.
- WebSocket: canal persistente y bidireccional para el intercambio de datos de sensores, control de movimiento y eventos en tiempo real.
- MQTT (fase experimental): protocolo liviano orientado a aplicaciones con alta concurrencia o entornos de red con latencias elevadas.

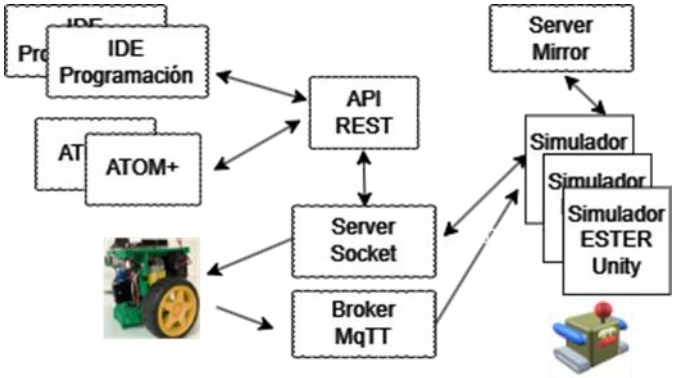


Figura 17. *Modelo de comunicación híbrida entre el Twin Robot y el simulador ESTER*

Flujo de comunicación:

1. El código del estudiante envía comandos por API REST o WebSocket.
2. El simulador ejecuta el movimiento virtualmente en Unity.
3. El mismo comando se transmite al TWIN ROBOT, ejecutando la acción en el mundo físico.
4. El robot físico retorna sus datos (posición, sensores).
5. El simulador compara y ajusta su representación, priorizando los datos reales si están disponibles.

Características clave de la arquitectura:

- **Baja latencia** (<100 ms en red local, validado en pruebas).
- **Capacidad multiusuario**: chequeada hasta 30 robots simulados conectados.
- **Capacidad robot físicos**: chequeada hasta 5 robots físicos conectados.
- **Tolerancia a fallos:** si el robot se desconecta, la simulación continúa.

Este enfoque responde a modelos recientes de interacción hombre-robot a través de entornos de realidad mixta y simulación escalable, como los propuestos por (Ostanin et al., 2021).
(Ver Anexo A: Esquema de arquitectura de interacción simulador-robot físico).
La arquitectura híbrida de comunicación entre robot y simulador es un ejemplo de lo que (Bencomo et al., 2024) define como ingeniería de modelos con adaptabilidad temporal, donde las abstracciones (como la posición del robot o la lectura del sensor) se actualizan no solo con datos sino también con eventos inesperados del entorno, permitiendo una razonabilidad bajo incertidumbre, en referencia a las preguntas 5 y 6 (¿Cómo pueden las abstracciones apoyar el razonamiento bajo incertidumbre y cambios? y ¿Cómo se puede diseñar una abstracción de manera confiable?). Asimismo, la coexistencia de API REST, WebSocket y MQTT puede ser vista como una estrategia de composición de abstracciones, coherente con los retos de modularidad y composición de modelos heterogéneos discutidos por los autores.

5.3. Integración de datos reales y simulados

El desafío de integrar coherentemente datos de sensores virtuales y reales es abordado mediante un módulo de fusión de estado, que se inspira en las arquitecturas compuestas de gemelos digitales descritas por (Singh et al., 2021;Schleich et al., 2017). En este módulo, la información real toma prioridad siempre que esté disponible, mientras que la simulación predice o interpola valores en caso de desconexión.

Ejemplos típicos de integración incluyen:

- Corrección automática de distancias medidas por sensores ultrasónicos.
- Ajustes en la visualización en caso de errores de encoder.
- Fallback automático a simulación cuando se interrumpe la conexión con el hardware.

Este diseño se alinea con propuestas académicas que integran múltiples subgemelos (sub-DT) con sincronización dinámica para componentes individuales (Grieves, 2024).

El modelo híbrido de gemelo digital permite la sincronización dinámica de información entre el robot físico y su réplica virtual, logrando una representación coherente y en tiempo real de su comportamiento.

Los datos recolectados por los sensores del robot (distancia, velocidad, orientación y trayectorias) se integran continuamente al simulador ESTER, donde son visualizados y analizados por los usuarios. Simultáneamente, las acciones generadas desde el entorno simulado se ejecutan sobre el robot real, manteniendo un ciclo de comunicación permanente que garantiza la fidelidad del gemelo digital.

Durante las pruebas de validación realizadas en 2025, se implementaron ejercicios de sincronización multi-robot, en los que distintos agentes (reales y virtuales) realizaron movimientos coordinados en tiempo real, como el trazado de trayectorias y figuras geométricas compartidas (figura 18, 19) . Esta experiencia demostró la estabilidad del sistema y su potencial para escenarios colaborativos, donde varios estudiantes pueden trabajar simultáneamente en la programación y control de robots interconectados.

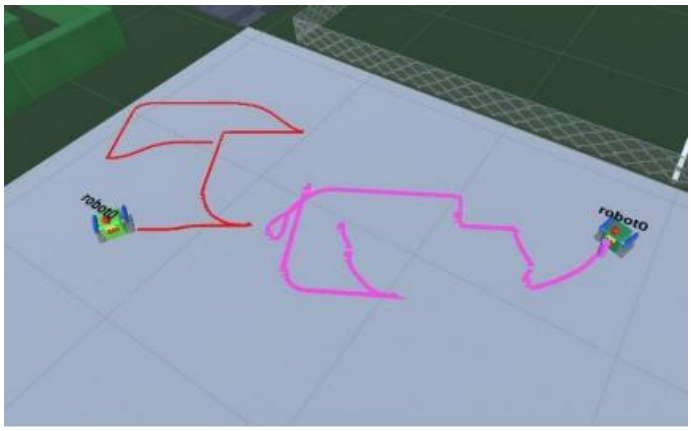


Figura 18. *Sincronización y actualización simultánea entre robots físicos y virtuales en tiempo real durante la prueba de trazado de trayectorias.*

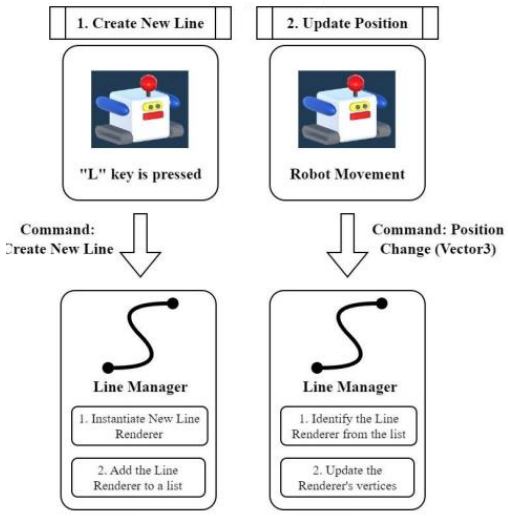


Figura 19. *Sincronización final del modelo híbrido mediante el módulo Line Manager, con control distribuido entre el simulador ESTER.*

Ejemplos de integración:

- Si el sensor ultrasónico virtual detecta un obstáculo a 20 cm pero el sensor real reporta 18 cm, el entorno se actualiza con el valor real.
- Si la posición simulada indica 3 cm de avance, pero el encoder real registra 2 cm, se corrige la visualización.
- Si el robot físico pierde conexión, se activa el modo de simulación, cuando el robot se conecta se revalida con los sensores del robot.

Este sistema garantiza que el estudiante visualice una representación precisa y coherente del entorno físico, o una predicción informada en caso de desconexión.

El componente físico del modelo híbrido de gemelo digital se materializa en el Twin Robot, desarrollado en el Laboratorio de Robótica Física e Inteligencia Artificial (LRFIA–UAI). Este robot actúa como la contraparte real del simulador ESTER, y fue diseñado para replicar sus movimientos y sensores en tiempo real mediante una arquitectura modular, accesible y de bajo costo.
El sistema incorpora un sensor LiDAR de 360° (figura 20), bumpers frontales, giróscopo de inclinación (figura 21) entre otros sensores y actuadores.

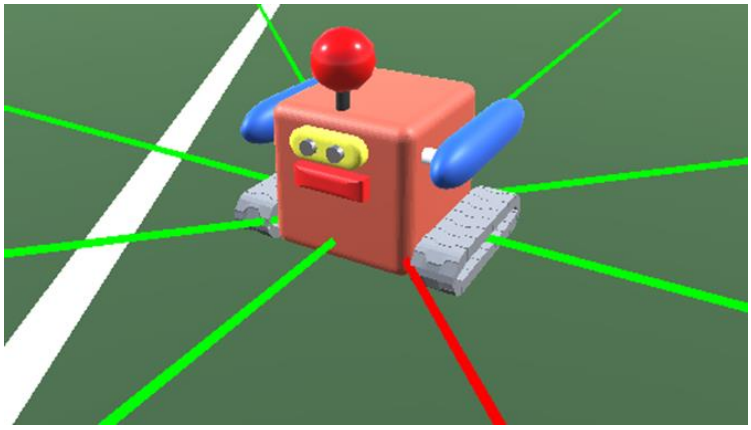


Figura 20. *Creación de sensor LiDAR 360° simulado.*

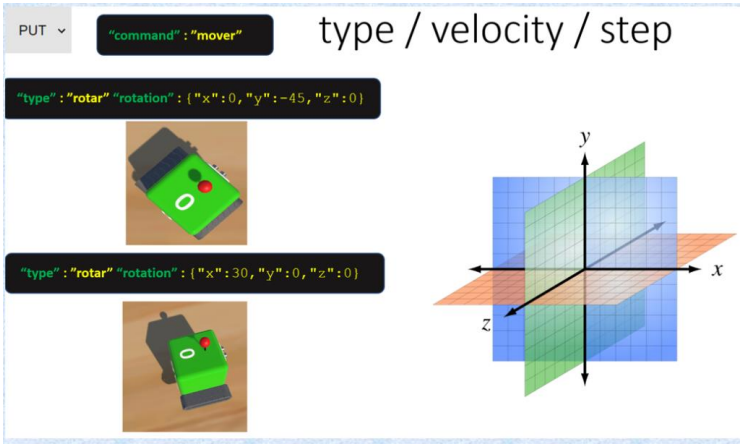


Figura 21. *Sensor de inclinación virtual.*

Esta configuración permite que el robot envíe datos sobre su entorno y posición al simulador, mientras recibe instrucciones de movimiento desde el entorno virtual, estableciendo un ciclo continuo de comunicación bidireccional que constituye la base funcional del modelo híbrido.

5.4. Toma de decisiones compartida en tiempo real

En línea con el concepto de sistemas inteligentes distribuidos, ESTER distribuye la toma de decisiones entre el código del alumno, el robot físico y el simulador. Esta estructura permite gestionar escenarios en los que la realidad supera la programación prevista, educando al estudiante en la naturaleza impredecible de los entornos físicos (Grieves, 2024; Zhang et al., 2023).

La dinámica de decisiones entre estudiante, robot y simulador representa un sistema de múltiples actores y niveles de abstracción, lo cual plantea desafíos sobre la alineación de perspectivas, otro de los ejes teóricos del Papert (Bencomo et al., 2024) en referencia a las preguntas 4 (¿Cómo podemos evaluar las abstracciones?). La capacidad de arbitraje por parte del simulador puede interpretarse como una estrategia de reconciliación entre abstracciones en conflicto, aportando robustez al sistema.

Ejemplos prácticos:

- Si el sensor físico detecta un obstáculo que el software no preveía, el robot se detiene y la simulación lo refleja.
- Si el giro físico es menor al previsto, la simulación se ajusta, enseñando sobre errores mecánicos reales.

Este enfoque está respaldado por estudios que muestran cómo el control mixto mejora el aprendizaje del pensamiento computacional (Liljaniemi & Paavilainen, 2020).

5.5. Escenarios de predominancia: simulación vs. robot físico

La arquitectura híbrida de ESTER permite alternar entre tres modos principales:

- Simulación pura: cuando no hay hardware disponible.
- Modo híbrido priorizando datos físicos, recomendado para prácticas avanzadas.
- Modo supervisado, utilizado en instancias evaluativas para asegurar validez técnica.

Esta flexibilidad es coherente con los modelos que conciben múltiples DT conectados en niveles jerárquicos, cada uno con diferente granularidad y fidelidad (Zhou et al., 2022).

La flexibilidad para alternar entre simulación pura, operación híbrida o predominio físico está directamente vinculada con el concepto de abstracciones adaptativas según contexto de uso, como lo señala la pregunta de investigación en referencia a las preguntas 4 (¿Cómo podemos diseñar mejores abstracciones?). Este enfoque permite seleccionar y configurar las abstracciones apropiadas según el nivel de experiencia del usuario, las condiciones de red o la disponibilidad de hardware, facilitando una experiencia educativa escalable.

5.6. Validación funcional del modelo con actividades concretas

El uso de pruebas prácticas para evaluar precisión, sincronización y latencia en el sistema puede interpretarse como parte del proceso de validación de abstracciones, en especial aquellas descubiertas automáticamente a partir del comportamiento del robot (por ejemplo, ajustes de trayectoria o lectura de sensores en entornos reales). De acuerdo con Selim (2024), tales evaluaciones son cruciales para mantener la confianza en sistemas ciberfísicos adaptativos, como los gemelos digitales educativos.

La validación del sistema se realizó mediante actividades didácticas que combinan robótica física y simulación. Los resultados fueron medidos en términos de precisión, latencia y sincronización, coincidiendo con métricas. En referencia a la rúbrica analítica, se estructura a partir del enfoque de competencias propuesto por Neil (2024) y se articula con los principios del *Project-Based Learning* (Bell, 2010; Thomas, 2000), donde la evaluación se concibe como un proceso formativo continuo. En este contexto, la robótica educativa fomenta la autonomía y la autorregulación (Ryan & Deci, 2000; Eguchi, 2014), con resultados positivos comprobados en entornos STEM (Condliffe et al., 2017).

Tabla 3: *Actividad y resultado.*

Actividad	Resultado	Observaciones
Seguimiento de línea	Precisión 94%	Desvío promedio entre trayectorias: ±2 cm
Evitación de obstáculos	97% de coincidencia	Latencia < 90 ms
ATOM+ en robot físico	Ejecución fluida	Transición sin interrupciones entre programación visual y ejecución real

Estos resultados confirman los principios establecidos por la literatura en DT aplicada a educación: sincronización, realismo, adaptabilidad y continuidad educativa (Grieves, 2024; Kartashova et al., 2020; Liljaniemi & Paavilainen, 2020).

- Fidelidad en el comportamiento entre entornos.
- Sincronización en tiempo real.
- Continuidad educativa incluso ante fallas técnicas.

CAPÍTULO 6

Evaluación y resultados

6.1. Estrategia metodológica de evaluación

La validación del simulador ESTER y del modelo híbrido de gemelo digital se abordó mediante una estrategia de evaluación mixta, que combinó:

- **Evaluación técnica:** mediciones objetivas de latencia, sincronización, precisión de replicación entre el robot físico y su contraparte simulada.
- **Evaluación pedagógica:** encuestas a estudiantes y docentes, análisis de desempeño en actividades de programación y robótica, y percepción del aprendizaje alcanzado.

Esta doble perspectiva permitió evaluar no solo la funcionalidad técnica del sistema, sino también su impacto real en los procesos de enseñanza y aprendizaje.

Las evaluaciones se realizaron en:

- **Carreras de grado**: Ingeniería en Sistemas (asignatura Robótica).
- **Carreras de posgrado**: Maestría en Tecnología Educativa (asignatura Programación, Robótica e Impresión 3D).
- **Curso y talleres de extensión** dirigidos a docentes en formación continua.

Como parte del enfoque pedagógico, se diseñó e implementó una rúbrica de evaluación por competencias, que permitió valorar de manera integral el desempeño de los estudiantes en el entorno híbrido. Esta rúbrica contempla dimensiones técnicas, cognitivas, pedagógicas y actitudinales, y se estructura en una escala de cuatro niveles de logro. Su uso facilitó:

- La evaluación formativa.
- La autoevaluación por parte de los estudiantes.
- La comparación de resultados entre prácticas realizadas en simulador y con el robot físico.

La rúbrica considera dimensiones técnicas, cognitivas, pedagógicas y actitudinales, y se estructura en una escala de cuatro niveles de logro. Su aplicación permitió no solo una evaluación más justa y transparente, sino también facilitar la autoevaluación, la retroalimentación formativa y la comparación entre prácticas en simulador y robot físico.

Este instrumento fue utilizado tanto en actividades de grado como en posgrado, adaptándose a diferentes perfiles de estudiantes. Se incluye el modelo completo en el capítulo 8, junto con ejemplos de uso y criterios observables para cada nivel.

6.2. Indicadores técnicos y pedagógicos de desempeño

Para evaluar el desempeño del entorno híbrido, se definieron indicadores específicos en dos ejes principales:

6.2.1 Indicadores técnicos

- **Latencia promedio** en la ejecución de comandos entre el simulador y el robot físico.
- **Precisión en trayectorias** programadas (comparación entre el entorno virtual y el movimiento real).
- **Coincidencia de eventos críticos**: detección de obstáculos, giros, colisiones.
- **Sincronización multiusuario** en prácticas colaborativas.

6.2.2 Indicadores pedagógicos

- **Facilidad de uso** de la plataforma combinada (simulador + robot físico).
- **Motivación estudiantil** para participar en actividades de programación.
- **Percepción de aprendizaje** de conceptos algorítmicos y de robótica.
- **Nivel de colaboración entre pares** en desafíos multiusuario.
- **Transferencia de conocimientos** entre el entorno simulado y el entorno físico.

6.3. Resultados cualitativos y cuantitativos

Tabla 4: *Resultados técnicos*

Prueba	Resultado	Observaciones

Latencia API REST ↔ robot físico	40–60 ms promedio	En red local cableada.
Latencia WebSocket ↔ robot físico	< 100 ms	En red Wi-Fi estándar, con leves fluctuaciones.
Precisión en seguimiento de trayectorias programadas	±2 cm de desvío promedio	En laberintos y pistas de seguimiento de línea.
Tiempo de reconexión automática ante caída de red	< 5 segundos	Se activa modo reconexión.
Fluidez en entorno multiusuario (30 robots)	Estable a 30 FPS	Equipos con GPU integrada mínima.

Estos resultados muestran que el modelo híbrido de ESTER cumple condiciones de operación en tiempo real adecuadas para el contexto educativo.

6.3.2 Resultados pedagógicos

Las encuestas aplicadas a estudiantes de diversas cohortes y talleres (N = 80) arrojaron los siguientes resultados:

Tabla 5: *Resultados pedagógicos*

Indicador	% respuestas positivas (Bueno + Muy Bueno + Excelente)
Facilidad de uso de simulador	85%
Utilidad para aprender algoritmia y programación	88%
Comprensión de conceptos de robótica colaborativa	89%
Motivación y participación activa	86%
Percepción de integración entre entorno real y simulado	92%
Transferencia de conocimientos a nuevas tareas	84%

Estos datos respaldan que el entorno no solo es técnicamente funcional, sino que también propicia un aprendizaje significativo.

Adicionalmente, se evaluó la alineación del entorno con el enfoque de Educación Basada en Competencias (EBC). Las prácticas realizadas en ESTER permitieron a los estudiantes demostrar competencias observables en situaciones que combinan programación, simulación y control de

robots reales. Las rúbricas (capitulo 8) contemplan dimensiones técnicas, cognitivas, actitudinales y pedagógicas, lo cual permitió una evaluación holística del aprendizaje.

Las rúbricas de evaluación incluidas en el capítulo 8 contemplan dimensiones técnicas, cognitivas, actitudinales y pedagógicas, lo que permitió una evaluación holística del desempeño estudiantil. La correlación entre los resultados obtenidos y los niveles esperados en la rúbrica confirma que el entorno es adecuado no solo para enseñar contenidos, sino para formar capacidades transferibles y contextualizadas, en línea con los principios de la EBC.

6.4. Comparación con modelos educativos tradicionales

Se comparó el modelo híbrido de ESTER con enfoques tradicionales centrados exclusivamente en kits físicos o simuladores desconectados. Los resultados mostraron ventajas sustanciales del enfoque propuesto:

Tabla 6: *Comparación modelos educativos*

Criterio	Simuladores tradicionales	Kits físicos convencionales	Modelo híbrido ESTER
Accesibilidad	Alta	Limitada (costo, disponibilidad)	Alta
Prácticas simultáneas	Limitadas	Limitadas	Hasta 30 alumnos
Repetición y ensayo	Ilimitado	Limitado (batería, daño)	Ilimitado
Transferencia a hardware real	Indirecta	Directa	Directa y sincronizada
Colaboración multiusuario	Baja	Moderada	Alta
Robustez ante fallas	Media	Baja	Media (modo desconexión)

La combinación de simulación y robótica física permite maximizar los recursos disponibles, potenciar la participación estudiantil y fomentar aprendizajes autónomos y colaborativos.

6.5. Retroalimentación de docentes y estudiantes

Los comentarios cualitativos recogidos en encuestas abiertas, entrevistas y talleres reflejan el impacto positivo del modelo:

Estudiantes:

- “Pude ver cómo mi programa realmente controlaba un robot físico sin tener miedo de romperlo.”
- “La programación visual con ATOM+ me ayudó a entender rápido lo que antes me costaba mucho.”
- “La conexión entre el simulador y el robot real me pareció mágica. Me motivó a seguir investigando.”

Docentes:

- “ESTER resolvió el problema de la falta de kits robóticos en las aulas y potenció las clases prácticas.”
- “La flexibilidad del modelo híbrido permite adaptarlo a distintos niveles de conocimientos.”
- “La estabilidad en red y la baja latencia son puntos muy fuertes del entorno desarrollado.

Estos testimonios refuerzan el valor práctico, pedagógico y escalable de la solución.

6.6. Limitaciones observadas

A pesar de los resultados positivos, se identificaron algunas limitaciones:

- Dependencia de la red Wi-Fi: en entornos saturados, pueden ocurrir desincronizaciones esporádicas.
- Curva de aprendizaje en ATOM+ avanzado: estudiantes con menor experiencia tecnológica requieren más tiempo de adaptación.
- Limitaciones físicas del prototipo robótico: el TWIN ROBOT presenta limitaciones de precisión en superficies irregulares o sucias.

Estas limitaciones abren líneas claras de trabajo futuro:

- Incorporación de mecanismos automáticos de corrección de errores.
- Optimización del hardware del robot físico.
- Ampliación de la robustez del sistema para redes inestables.

6.7. Conclusión

La validación técnica y pedagógica realizada demuestra que el modelo híbrido de gemelo digital implementado en ESTER es:

- **Viable**: se adapta a diferentes contextos educativos.
- **Eficaz**: cumple con los requisitos de sincronización, fidelidad y estabilidad.
- **Transferible**: permite replicar prácticas exitosas en otras instituciones.

Su adopción mejora la calidad del aprendizaje en robótica educativa, aportando una solución innovadora, flexible y centrada en el desarrollo de competencias reales.

Este capítulo abordó en profundidad la fundamentación pedagógica que sostiene el uso del modelo híbrido de gemelo digital en contextos educativos, destacando los enfoques de construccionismo, aprendizaje activo y aprendizaje significativo, además del paradigma de educación basada en competencias. Las experiencias prácticas, tanto en carreras técnicas como en trayectos formativos no técnicos y de posgrado, confirmaron que el simulador ESTER y el robot físico TWIN ROBOT, junto a la programación visual de ATOM+, son herramientas eficaces que favorecen la motivación, la participación y la comprensión profunda de conceptos complejos en robótica y programación. Los resultados de las experiencias realizadas muestran claramente que el modelo híbrido no solo facilita un aprendizaje autónomo y colaborativo, sino que también permite integrar conocimientos teóricos y prácticos en un entorno seguro, flexible y accesible. Así, ESTER emerge como una plataforma educativa robusta y escalable, alineada con las demandas actuales del mundo profesional y las tendencias internacionales en educación superior tecnológica.

CAPÍTULO 7

Trabajos relacionados y producción científica vinculada

7.1 Introducción

La presente investigación se inscribe en el campo emergente de los gemelos digitales aplicados a la educación en robótica, integrando desarrollos tecnológicos con fundamentos pedagógicos. El análisis de la literatura reciente evidencia un crecimiento sostenido entre 2018 y 2025, con un marcado predominio de experiencias en educación superior y un avance incipiente hacia modelos híbridos físico-virtuales.

El concepto de gemelo digital (digital twin) tiene sus orígenes en los programas de la NASA durante la década de 1960, cuando se emplearon réplicas físicas y virtuales de las naves espaciales para realizar simulaciones de contingencia, como se evidenció en las misiones Apolo, particularmente en el caso del Apolo 13 (N.A.S.A., 2021 ;Siemens, 2020) Estas simulaciones permitían reproducir en tierra el comportamiento de los sistemas reales en el espacio, constituyendo un antecedente directo de la filosofía de los gemelos digitales. En la misma línea, la Agencia Espacial Europea (E.S.A., 2022), 2022) señala que este enfoque marcó el inicio de los modelos vivos o living models, base conceptual de las aplicaciones modernas de digital twins.

Posteriormente (Gelernter, 1991) profundizó la idea con su obra Mirror Worlds, donde propuso la creación de réplicas digitales que reflejaran en tiempo real el funcionamiento de sistemas físicos complejos. Este pensamiento visionario sentó las bases teóricas de los entornos virtuales interconectados que hoy se asocian al gemelo digital (Zhou, Leng, et al., 2022).

Finalmente, el término “digital twin” fue formalizado por Michael Grieves (2002) en el ámbito industrial, dentro del marco del Product Lifecycle Management (PLM), como una representación virtual que replica el ciclo de vida de un producto físico (Grieves, 2014; Grieves & Vickers, 2016).

Su propuesta fue posteriormente adoptada por la NASA y otros sectores manufactureros, consolidando la definición contemporánea del gemelo digital como la integración dinámica entre el mundo físico y el digital (I.B.M., 2022 ; Systémes, 2023).

A partir de 2018, la literatura científica muestra un crecimiento sostenido de publicaciones que exploran su aplicación en educación, con especial énfasis en la enseñanza de la robótica y disciplinas STEM. En particular, el Mapeo Sistemático de la Literatura (MSL) (Balich et al., 2025) recientemente identificó 52 trabajos relevantes en el período 2018–2025. El estudio confirmó un pico de producción en 2023 y 2024, con un 70% de experiencias situadas en la educación universitaria (figura 22).

El MSL permitió identificar cinco tendencias relevantes:

- Crecimiento sostenido de la producción científica, con especial concentración en 2023–2024, asociado a la expansión de la Industria 4.0/Educación 4.0.
- Predominio de enfoques activos, como el aprendizaje basado en proyectos, el aprendizaje colaborativo y la gamificación, en línea con teorías constructivistas y construccionistas (Saldarriaga et al., 2020; Bonwell & Eison, 1991) .
- Uso recurrente de plataformas como Unity, Gazebo y ROS, junto con la exploración de XR e IoT cloud, lo que refleja una convergencia tecnológica hacia entornos inmersivos.
- Concentración en contextos universitarios, con menor presencia en niveles primarios y secundarios, lo cual constituye un desafío para la democratización del acceso.
- Brechas metodológicas y de acceso, vinculadas a la falta de estudios longitudinales, la escasa integración curricular y la necesidad de soluciones *low-cost*.

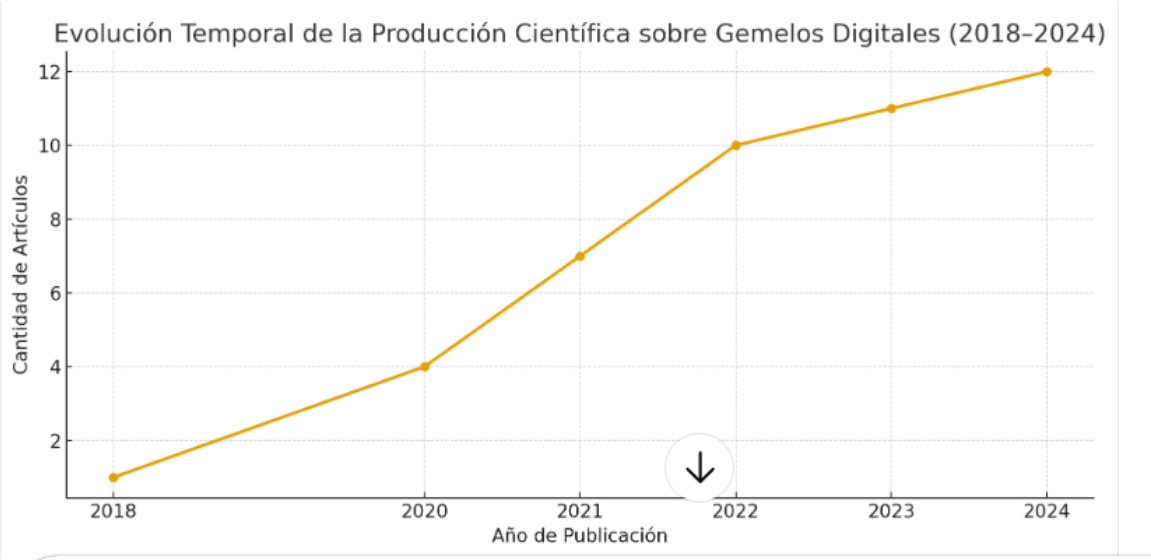


Figura 22. *Presenta la evolución de la producción científica vinculada a los gemelos digitales en educación entre 2018 y 2025, mostrando un crecimiento progresivo y alineado con las tendencias (Balich et al., 2025).*

7.2 Antecedentes y desarrollos relacionados

a) Simuladores educativos en robótica

Las plataformas de simulación como Gazebo, Webots, Tinkercad Circuits y CoppeliaSim han sido ampliamente utilizadas en la enseñanza de robótica. Si bien ofrecen motores de física avanzados y flexibilidad en la modelización de robots móviles, presentan limitaciones pedagógicas:

- complejidad técnica que dificulta su acceso a estudiantes novatos;
- interfaces poco amigables para usuarios sin formación informática;
- escasa adaptación a escenarios colaborativos multiusuario en tiempo real.

En contraste, Unity 3D ha comenzado a incorporarse en propuestas educativas por su potencia visual, baja latencia y compatibilidad multiplataforma. Sin embargo, la mayoría de los desarrollos reportados en la literatura se enfocan en la visualización y carecen de integración con hardware físico o de herramientas pedagógicas visuales integradas (Singh et al., 2025) .

El entorno ESTER supera estas limitaciones mediante una arquitectura bidireccional, que conecta robots simulados y físicos en un espacio multiusuario, validado pedagógicamente en experiencias de grado y posgrado.

De acuerdo con el MSL realizada por Balich (2025), las plataformas más utilizadas en educación robótica son Unity, Gazebo y Webots, seguidas de Tinkercad Circuits y CoppeliaSim.

Tabla 7, *sintetiza las diferencias entre simuladores tradicionales y el entorno híbrido ESTER.*

Criterio	Simuladores tradicionales (Gazebo, Webots, CoppeliaSim, Tinkercad)	Entorno híbrido ESTER
Accesibilidad	Requieren instalación compleja, bibliotecas externas y conocimientos avanzados.	Acceso mediante Unity 3D y cliente web; interfaz intuitiva y adaptable a distintos perfiles.
Orientación pedagógica	Enfoque técnico; escasa integración de teorías educativas y competencias.	Diseño fundamentado en construccionismo, aprendizaje significativo y EBC.
Interfaz de usuario	Generalmente textual o con configuraciones avanzadas poco amigables.	Interfaz visual amigable con soporte para programación visual (ATOM+).
Integración con hardware físico	Limitada o nula; algunos soportan conexión indirecta vía ROS.	Arquitectura bidireccional con sincronización en tiempo real de robots físicos (Twin Robot).
Entornos colaborativos	Orientados a simulación individual o por escenarios aislados.	Soporte multiusuario y multipresencia con hasta 30 robots en simultáneo.
Latencia y visualización	Alta carga computacional, interfaces menos inmersivas.	Baja latencia, gráficos inmersivos en Unity y cámaras en primera/tercera persona.
Escalabilidad	Dependencia de recursos locales; limitada para grandes grupos.	Ejecutable en red local y nube; adaptable a clases presenciales y remotas.
Evaluación pedagógica	Sin instrumentos integrados para medir aprendizajes.	Validación mediante rúbricas competenciales, encuestas y métricas técnicas.
Modelo educativo	Simulación unidireccional o centrada en sensores.	Modelo híbrido bidireccional, con coherencia entre simulación y robot real.
Innovación diferencial	Enfoque centrado en la simulación técnica.	Integración completa: simulación + hardware +

		programación visual + pedagogía.

b) Gemelos digitales y educación

El concepto de *digital twin*, originado en la industria para manufactura, mantenimiento predictivo y *smart cities* (Tao et al., 2019), ha comenzado a trasladarse al ámbito educativo. No obstante, en robótica educativa su implementación aún es incipiente: las experiencias reportadas suelen restringirse a la visualización de datos sensoriales o a la simulación unidireccional.

El modelo híbrido implementado en ESTER avanza sobre esta frontera al proponer un gemelo digital bidireccional. En caso de discrepancias, la información proveniente del robot físico se prioriza frente a la simulación, garantizando fidelidad. Además, el sistema permite la participación simultánea de hasta 30 usuarios en un mismo escenario distribuido, lo que representa un aporte diferencial frente a trabajos existentes (Mustafee et al., 2023; Mazumder et al., 2023).

c) Programación visual en contextos formativos

Entornos como Scratch, Blockly y Snap han democratizado el acceso a la programación en la educación inicial y en propuestas STEAM. Sin embargo, su integración con simulación tridimensional multiusuario y robots físicos sigue siendo limitada.

El módulo ATOM+, desarrollado en el marco de esta tesis, constituye una innovación al ofrecer:

- bloques personalizados vinculados directamente al simulador 3D,
- retroalimentación inmediata en robots virtuales y físicos,
- accesibilidad para estudiantes sin formación técnica,
- transición gradual hacia lenguajes textuales como Python.

Este diseño se enmarca en el construccionismo de Papert (Saldarriaga et al., 2020) y en el alineamiento constructivo (Biggs, 1999), posicionando al estudiante como protagonista de su aprendizaje y garantizando coherencia entre objetivos, actividades y evaluación.

7.3 Originalidad y aportes del entorno ESTER

El desarrollo del simulador ESTER representa un aporte original en múltiples dimensiones:

- **Técnica:** integración de Unity 3D, WebSocket, API REST y comunicación en tiempo real con robots físicos.
- **Pedagógica:** combinación de programación visual (ATOM+) con un modelo de gemelo digital accesible a carreras técnicas y no técnicas.
- **Arquitectónica:** soporte multiusuario, multipresencia y sincronización simultánea con robots virtuales y físicos.
- **Escalabilidad:** ejecución en red local o en nube, con posibilidad de expansión en aulas paralelas.
- **Modelo híbrido:** bidireccionalidad en tiempo real entre simulador y robot, con lógica de control sincronizada y tolerancia a fallos.

En comparación con otros desarrollos educativos y comerciales, ESTER ofrece un enfoque único de integración total entre simulación, hardware, programación visual, multiusuario y aprendizaje activo.

7.4 Producción científica generada

Durante el desarrollo de esta tesis se difundieron resultados en congresos, revistas y talleres, lo que permitió validar los avances y recibir retroalimentación de la comunidad académica. Esta producción fue reconocida con premios institucionales y financiamiento internacional.

Tabla 8: *Resume la producción científica vinculada al simulador ESTER desde 2009 hasta 2025.*

Año	Título / Trabajo	Evento / Publicación	Observaciones / Reconocimientos
2025	*Sincronización multi-robot en tiempo real para el simulador ESTER y gemelo digital*	Congreso Argentino de Ciencias de la Computación (Balich et al., 2025)	Presenta sincronización multi-robot y bidireccionalidad en tiempo real.
2025	*ATOM+: Integration of Visual Programming and*	Jornada Argentina de Informática e Investigación	Introduce ATOM+ como herramienta

	Digital Twins in the ESTER Simulator	Operativa Jornada Argentina de Informática e Investigación Operativa (Balich & Balich, 2025)	visual integrada al modelo híbrido.
2025	*Implementación de Robot Físico con Gemelo Digital para el Simulador Ester*	Congreso de Tecnología en Educación y Educación en Tecnología *(Balich et al., 2025)*	Creación de robot físico para el simulador Ester con gemelos digitales.
2025	*Gemelos Digitales en la Educación en Robótica: Un Mapeo Sistemático de la Literatura*	Congreso Nacional de Ingeniería Informática / Sistemas de Información (Balich et al., 2025)	Mapeo sistemático de la literatura con eje en gemelos digitales en educación.
2024	*Simulador de robótica 3D multirobot para enseñanza de programación*	Congreso Nacional de Ingeniería Informática / Sistemas de Información (Balich & Balich, 2024)	Validación con estudiantes de grado; entorno multiusuario.
2024	*Desarrollo de simulador de robótica cloud multipresencia para enseñanza de programación*	Congreso Argentino de Ciencias de la Computación	Extiende ESTER al ámbito cloud; multipresencia y escalabilidad.
2024	Exposición del trabajo	Congreso Argentino de Ciencias de la Computación – Workshop Innovación Educativa	Premio a la presentación destacada.
2024	Dictado Materia programación robótica e impresión 3D, de la Maestría en Tecnología Educativa	Universidad Abierta Interamericana	Premio a *Buenas Prácticas Educativas*.
2023	*Simulador de robótica en la nube para prácticas colaborativas*	Congreso Nacional de Ingeniería Informática / Sistemas de Información (Balich et al., 2023)	Extensión cloud y prácticas multiusuario.
2023	*Simulador 3D de robótica distribuido en Unity para enseñanza de programación e IA*	Revista Abierta de Informática Aplicada (Balich et al., 2023)	Publicación científica indexada.

2023	*Taller docente sobre simulación con ESTER*	Congreso Nacional de Ingeniería Informática / Sistemas de Información	Transferencia didáctica y capacitación docente.
2017	*Cloud Robotic*	*Revista Abierta de Informática Aplicada* (Balich, 2017)	Primer acercamiento al concepto de robótica en la nube.
2014	*Interface Natural Emotiva Niño-Robot*	Workshop de Investigadores en Ciencias de la Computación (Balich et al., 2014)	Investigación en interacción humano-robot.
2009	*Construcción de robots autónomos colaborativos*	Tesis de Maestría – Universidad Abierta Interamericana. (Balich, 2009)	Antecedente directo de la línea de investigación en robótica educativa.

Producción previa:

- ***Cloud Robotic*** – Revista Abierta de informática e Investigación Aplicada (2017) – Universidad Abierta Interamericana.
- ***Interfaz Natural Emotiva Niño-Robot*** – Workshop de Investigadores en Ciencias de la Computación (2014).
- ***Construcción de robots autónomos colaborativos*** – Tesis maestría en sistema informáticos, Universidad Abierta interamericana (2009).

Reconocimientos:

- **AWS Cloud Credit for Research** (2023): subsidio de USD 11.000 otorgado a ESTER para infraestructura cloud.

7.5 Convergencia con tendencias emergentes

El entorno **ESTER** converge con líneas de desarrollo contemporáneo:

- **Educación Basada en Competencias (EBC):** permite diseñar actividades orientadas a habilidades prácticas y medibles.
- **IoT y Cloud Robotics:** habilita la conexión de múltiples dispositivos físicos a entornos virtuales distribuidos.

- **XR y metaverso educativo:** sienta bases para futuras aplicaciones inmersivas.
- **Inteligencia artificial aplicada a la docencia:** abre posibilidades en analítica de aprendizaje y tutoría automatizada.

7.6 Síntesis del aporte diferencial

El simulador **ESTER** constituye un aporte innovador, replicable y validado, que integra teoría pedagógica y tecnología emergente. A lo largo de esta tesis se demostró que:

- los entornos híbridos bidireccionales son técnicamente viables y pedagógicamente eficaces;
- la programación visual integrada (ATOM+) amplía el acceso a perfiles no técnicos;
- la sincronización simultánea de robots virtuales y físicos potencia la transferencia de conocimientos.

La propuesta confirma el valor del construccionismo (Papert), el aprendizaje significativo (Ausubel, 1962; Novak, 2010), el alineamiento constructivo (Biggs, 1999)y la educación basada en competencias (Tobón, 2006).

El MSL realizado (Balich et al., 2025) demostró que los enfoques pedagógicos más frecuentes en la literatura son el aprendizaje basado en proyectos (≈40%), el aprendizaje colaborativo (≈20%) y la gamificación (≈20%). Sin embargo, la mayoría de las experiencias se apoyan en evaluaciones de percepción o test pre/post, con escaso uso de rúbricas pedagógicas robustas. En este punto, el simulador ESTER constituye un aporte diferencial al integrar un modelo evaluativo sistemático.

Tabla 9: *Enfoques pedagógicos más frecuentes en la literatura (2018–2025)*

Enfoque pedagógico	Frecuencia relativa	Observaciones
Aprendizaje Basado en Proyectos (PBL)	Alto (≈40%)	Favorece resolución de problemas reales
Aprendizaje colaborativo	Medio (≈20%)	Enfocado en multipresencia y entornos distribuidos

Gamificación	Medio (≈20%)	Motiva, pero limitado en profundidad curricular
Evaluación con encuestas y pre/post tests	Alto	Bajo rigor metodológico, escasez de estudios longitudinales
Learning Analytics	Bajo (<10%)	Identificada como brecha pendiente

Como líneas futuras, se prevé:

- extender ATOM+ con bloques específicos para gemelos digitales,
- profundizar la integración de realidad mixta e IA,
- aplicar el modelo en primaria y secundaria,
- realizar evaluaciones longitudinales para medir impacto sostenido.

Los resultados completos de percepción estudiantil y rúbricas aplicadas se presentan en el capitulo 8, complementando los hallazgos descritos.

7.7 Desafíos y líneas futuras

Se recomienda ampliar con lo que surge del MSL:

El MSL (Balich et al., 2025) identificó como principales desafíos:

- **Accesibilidad y costos:** necesidad de modelos abiertos y *low-cost* aplicables en contextos con recursos limitados.
- **Integración curricular:** las experiencias suelen ser aisladas, con escasa articulación longitudinal en planes de estudio.
- **Evaluación longitudinal:** falta de estudios que midan impacto sostenido en el tiempo.
- **Analítica de aprendizaje e IA:** escasa incorporación para personalizar trayectorias y generar datos de desempeño ricos.
- **Expansión a niveles iniciales y secundarios:** todavía ausente en la mayoría de los casos, pese a su potencial democratizador.

Estos desafíos orientan las líneas futuras de investigación y consolidan la relevancia de la propuesta del simulador ESTER en el campo.

7.8 Conclusión

El análisis de trabajos relacionados y la producción científica vinculada muestra que los gemelos digitales en educación, y particularmente en la enseñanza de robótica, constituyen un campo en expansión acelerada desde 2018. El Mapeo Sistemático de la Literatura (Balich et al., 2025) confirmó esta tendencia y evidenció tanto avances "como el crecimiento de experiencias en educación superior y el uso de plataformas como Unity, Gazebo y ROS" como brechas metodológicas persistentes, entre ellas la escasa integración curricular, la falta de estudios longitudinales y la limitada aplicación en niveles iniciales.

En este escenario, la propuesta del simulador ESTER se distingue por su originalidad técnica y pedagógica: integra simulación multiusuario, conexión bidireccional con robots físicos y un modelo educativo fundamentado en el construccionismo, el aprendizaje significativo y la educación basada en competencias. La producción científica generada y los reconocimientos obtenidos refuerzan la validez del modelo y su potencial de transferencia.

CAPÍTULO 8

Aplicación pedagógica del modelo de gemelo digital híbrido

8.1 Introducción

El presente capítulo ejemplifica la aplicación pedagógica del modelo híbrido de gemelo digital implementado en el simulador ESTER, integrando entornos virtuales y físicos en tiempo real. El objetivo es demostrar cómo la articulación entre simulación 3D y robótica física puede potenciar la enseñanza basada en competencias como lo plantean (Tobón, 2006) y el aprendizaje significativo (Ausubel, 1963).

El enfoque adopta los lineamientos del desarrollo de competencias conducido por rúbricas, propuesto (Neil, 2024) y los principios del alineamiento constructivo (Biggs, 1999), donde los resultados de aprendizaje, las estrategias de enseñanza y los criterios de evaluación se encuentran coherentemente integrados. Este modelo se apoya también en la perspectiva construccionista de Papert como lo expone (Folk, 1981) en su artículo, que promueve el aprendizaje a través de la creación activa de artefactos digitales.

Tomando como base los lineamientos y resultados obtenidos en los Trabajos Prácticos (TP) desarrollados a lo largo de los diferentes módulos del simulador ESTER, se propone el diseño de una secuencia didáctica integral orientada al desarrollo de una competencia específica del programa de estudios. Esta secuencia cuenta con el nivel de detalle necesario para ser implementada en el entorno híbrido físico-virtual, asegurando la coherencia entre los objetivos formativos, las estrategias de enseñanza y los criterios de evaluación establecidos.

En la construcción de estos instrumentos pedagógicos “matrices de tributación, rúbricas analíticas y secuencias didácticas“ se retoman los lineamientos metodológicos del *Taller de definición, desarrollo y evaluación de competencias* (Bertossi, 2024), adaptados al contexto híbrido del simulador ESTER para facilitar la observación del desempeño y la evaluación formativa:

- **Matriz de tributación (simplificada):**
  Vinculación entre competencias, resultados de aprendizaje y módulos del simulador ESTER.

- Niveles de tributación.

- **Matriz de competencia (simplificada):**

  Correspondencia entre competencias, niveles de dominio y criterios de evaluación.
  - Niveles de dominio.
- **Programa del módulo “Uso del simulador ESTER”:**

  Definición del resultado de aprendizaje y su relación con los microtrabajos prácticos (MicroTP) y el trabajo integrador (TPi).
- **Rúbrica analítica:**
  - Lineamientos para el desarrollo de las actividades de aprendizaje.
  - Lista de verificación derivada de la rúbrica para la evaluación formativa.
- **Proceso de desarrollo de competencias conducido por rúbricas:**
  - Microtrabajos prácticos (MicroTP1 a MicroTP5).
  - Trabajo Práctico Integrador (TPi).
- **Secuencia didáctica:**
  - Plantilla de secuencia didáctica analítica.
  - Secuencia didáctica completa para la implementación en aula o laboratorio, integrando las dimensiones virtual y física del gemelo digital.

Estos componentes conforman el andamiaje pedagógico que permite operacionalizar el modelo híbrido de gemelo digital educativo, facilitando la observación del desempeño, la retroalimentación formativa y la mejora continua del aprendizaje en robótica.

8.2 Fundamentación teórica

La enseñanza basada en competencias se centra en el desarrollo integrado de saberes, habilidades y actitudes orientadas a la acción efectiva en contextos reales o simulados (Tobón, 2006).

El alineamiento constructivo (Biggs, 2001) promueve la coherencia entre los resultados de aprendizaje (RA), las actividades de enseñanza y los criterios de evaluación, asegurando que todo el proceso formativo responda a un mismo objetivo.

En su artículo (Neil et al., 2023) sostienen que las rúbricas pueden conducir el desarrollo de las competencias, no solo medirlo. Por consiguiente, cada descriptor analítico de la rúbrica se transforma en un microTP (trabajo práctico focalizado) trabajo práctico que aborda un aspecto

puntual de la competencia, reduciendo la carga cognitiva y permitiendo avanzar progresivamente hacia la integración de saberes en un trabajo final. En este sentido, la integración del modelo híbrido en entornos educativos técnicos responde a la necesidad de fomentar la motivación intrínseca, la autonomía y la relación social como condiciones esenciales para el aprendizaje significativo (Ryan & Deci, 2000).

En línea con este enfoque, se propone una estructura analítica de rúbrica compuesta por descriptores A+ y B, que funcionan como criterios verificables y observables. Gracias a esta estructuración, las rúbricas operan tanto como instrumentos de evaluación como guías de aprendizaje. Además, el modelo se adapta naturalmente al entorno del simulador ESTER, donde las evidencias del desempeño (movimiento, sincronización, integración de sensores, colaboración) son observables y cuantificables..

8.3 Diseño pedagógico de la clase

8.3.1 Matriz de tributación (simplificada)

La matriz de tributación constituye un instrumento de alineamiento curricular que permite visualizar la relación entre las competencias de egreso y las actividades formativas que contribuyen a su desarrollo.

En el contexto de esta investigación, la matriz de tributación se emplea para vincular las competencias específicas de la asignatura Robótica con los módulos de uso del simulador ESTER, los cuales integran tanto la programación visual mediante ATOM+ como la interacción en tiempo real con el robot físico TWIN.

Utilizando los lineamientos establecidos en los trabajos prácticos desarrollados en los diferentes módulos, se propone una secuencia didáctica conducida por rúbricas que integra la simulación con el entorno físico. Este modelo permite evidenciar el proceso de aprendizaje y las competencias alcanzadas a través de matrices, listas de verificación y Micro TP (trabajo práctico focalizado) progresivos

La (Tabla 10) muestra la relación entre las competencias específicas de la asignatura Robótica, los módulos del simulador ESTER y los resultados esperados en cada instancia. Esta matriz permite visualizar la correspondencia entre competencias, actividades y resultados de aprendizaje, garantizando coherencia curricular.

Tabla 10: *muestra esta relación entre las competencias de egreso, los módulos de aplicación en el simulador ESTER y los resultados esperados en cada instancia.*

EXTRACTO DE LA MATRIZ DE TRIBUTACIÓN		
TIPO DE COMPETENCIA	**DESCRIPCIÓN**	
Tecnológicas	**CG1**	Identificar, formular y resolver problemas de ingeniería.
	CG2	Concebir, diseñar y desarrollar proyectos de ingeniería.
	CG3	Gestionar-planificar, ejecutar y controlar-proyectos de ingeniería (sistemas).
	CG4	Utilizar de manera efectiva las técnicas y herramientas de la ingeniería.
	CG5	Contribuir a la generación de desarrollos tecnológicos.
Sociales, políticas y actitudinales	**CG6**	Desempeñarse de manera efectiva en equipos de trabajo.
	CG7	Comunicarse con efectividad.
	CG8	Actuar con ética, responsabilidad profesional y compromiso social.
	CG9	Aprender en forma continua y autónoma.

Tabla 11: *Niveles de la matriz de tribulación*

Nivel de tribulación	Significado	Descripción
A – Alta	Contribución Alta	La asignatura desarrolla la competencia de manera central: los estudiantes la practican, aplican y evidencian en proyectos o actividades integradoras.
M – Media	Contribución Media	La asignatura fortalece parcialmente la competencia: se trabaja como apoyo o en un contexto acotado (por ejemplo, una práctica o módulo específico).
I – Incipiente	Contribución Incipiente	La competencia se introduce o sensibiliza, pero no se desarrolla plenamente (nivel inicial o de familiarización).
B – Básica *(a veces usada en lugar de I)*	Contribución Básica	Similar a "incipiente", indica una introducción conceptual o un abordaje indirecto.

8.4 Rúbrica analítica y lineamientos de aprendizaje

8.4.1 Lineamientos para el desarrollo de actividades de aprendizaje

La rúbrica analítica constituye el instrumento pedagógico central del modelo híbrido, desde el cual se derivan los demás artefactos de evaluación y enseñanza: lineamientos para actividades de aprendizaje, listas de verificación. Es importante también considerar los criterios relacionados con la creatividad, la resolución de problemas y la integración tecnológica que destacan el potencial de la robótica para la construcción de conocimiento interdisciplinario.

En coherencia con el enfoque de desarrollo de competencias conducido por rúbricas propuesto (Neil, 2024) y sus reglas y meta-reglas de validación, la rúbrica no solo mide el desempeño, sino que genera las pautas de acción y evaluación que guían la progresión de los estudiantes.

Tabla 11: *Ponderación de criterios y descriptores de evaluación para el diseño de las rubricas.*

Criterios de evaluación	Ponderación (%)	Descriptores	Ponderación (%)	∑
Producción escrita	5	Lenguaje simbólico	70	**100**
		Prolijidad	30	
Análisis	85	Dominio	6	**100**
		Derivada	20	
		Valores críticos	4	
		Intervalos de crecim./decrecim.	20	
		Extremos relativos	10	
		Intervalos de concavidad	20	
		Puntos de inflexión	10	
		Asíntotas	10	
Consistencia	10	Verificación de resultados	50	**100**
		Consistencia del proceso	50	
∑	**100**			

Se agrego una nueva columna a la matriz convencional (Tabla 13) llamada “reglas” en base a las reglas y metareglas propuestas (Neil, 2024) en su artículo.

Esta columna permite trazar la correspondencia entre cada criterio y las reglas de diseño de rúbricas analíticas (Neil, 2024).

Las Reglas R1–R7 aseguran la validez interna del instrumento (coherencia, observabilidad, consistencia y especificidad), mientras que las Meta-reglas M1–M7 refuerzan la validez externa (alineamiento constructivo, equilibrio semántico, aplicabilidad y evaluación 360°+).

Esta integración explicita la validación pedagógica y técnica de la rúbrica y respalda su reproducibilidad en contextos híbridos de aprendizaje con robots físicos y digitales.

Tabla 13: rúbrica analítica y lineamientos de aprendizaje.

Rúbrica: Análisis de funciones			
Criterios de evaluación	**Descriptor**	**Contexto**	
		Criterio analítico A+	**Criterio analítico B**
	Lenguaje simbólico	*La comunicación en programación debe ser clara, estructurada y coherente.*	
		Desarrolla programas con estructuras lógicas bien definidas, empleando sintaxis precisa y un uso adecuado y consistente de variables, condicionales y bucles, sin errores de ejecución.	Emplea estructuras lógicas que permiten ejecutar el programa con funcionalidad básica.

Producción técnica	**Prolijidad y organización del código**	*El código debe reflejar orden y claridad en la lógica de ejecución.*	
		Organiza el programa de manera clara y estructurada en ATOM+, utilizando nombres de variables y funciones descriptivos, comentarios explicativos en cada sección clave y una disposición visual coherente que facilita la lectura del código.	Presenta un código organizado, con estructura funcional básica y uso limitado de comentarios, aplicados solo en secciones principales o aisladas.
	Control y sincronización físico-virtual	*El sistema debe responder de forma coherente y fluida.*	
		Integra de forma completa y funcional los sensores, actuadores y la comunicación en tiempo real con el robot físico TWIN a través de WebSocket/MQTT, validando el sistema con baja latencia y comportamiento estable durante toda la ejecución.	Sincroniza el entorno físico y virtual, aunque presenta interrupciones ocasionales o latencia perceptible en al menos un componente del sistema.
	Interpretación de resultados	*Los resultados deben reflejar comprensión del proceso y evaluación crítica.*	
		Analiza en detalle las diferencias entre los datos del robot real y del simulador, explicando sus causas con argumentos técnicos y aplicando correcciones que eliminan las discrepancias, con validación de los resultados finales.	Compara los resultados entre el robot físico y el simulador, identificando al menos una discrepancia relevante, pero sin explicar sus causas ni aplicar ajustes correctivos..
	Resolución de problemas.	*Capacidad para abordar desafíos imprevistos y ajustar la lógica del sistema.*	
		Identifica de forma precisa las fallas funcionales del sistema, documenta el proceso de diagnóstico y aplica correcciones efectivas, validando su impacto mediante pruebas verificables.	Identifica al menos un problema funcional en el sistema y aplica una solución que permite continuar la ejecución, pero sin registrar el procedimiento completo ni verificar la efectividad del cambio.
	Colaboración y comunicación técnica	Trabajo grupal y distribución de tareas en entorno híbrido.	
		Coordina activamente el trabajo grupal, dejando registro claro y completo de la distribución de tareas, la comunicación sostenida y las contribuciones individuales durante todo el proceso.	Cumple con las tareas asignadas dentro del grupo y comunica avances relevantes, pero no registra de forma completa ni organizada su participación ni la de sus compañeros.
	Consistencia del proceso	Coherencia entre diseño, implementación y evaluación.	
		Documenta de forma completa y organizada el desarrollo del sistema, incluyendo diseño, pruebas e integración final, con resultados que se corresponden explícitamente con los objetivos propuestos	Evidencia el diseño y desarrollo del sistema con resultados funcionales parciales.

Nota: La rúbrica fue diseñada con enfoque analítico y validación binaria (“se cumple / no se cumple”), aplicando las Reglas R1–R7 y Meta-reglas M1–M7 de Neil (2024). La columna “Reglas aplicadas” documenta la correspondencia entre cada criterio de evaluación y las reglas de diseño que aseguran su coherencia y observabilidad.

Tabla 14: *Reglas y Meta-reglas de validación de rúbricas analíticas*

Código	Tipo	Descripción resumida	Aplicación en la rúbrica del modelo híbrido ESTER
R1	Regla de correspondencia	Cada criterio de evaluación debe vincularse con un resultado de aprendizaje.	En la rúbrica, cada criterio está alineado con las competencias de la matriz (Tabla 10).
R2	Regla de observabilidad	Los descriptores se expresan como proposiciones observables y verificables.	Se utilizan verbos operativos ("implementa", "configura", "documenta").
R3	Regla de diferenciación	Cada criterio analítico A^+ y B debe medir dimensiones distintas y no superpuestas.	Los descriptores A^+ y B se redactan con comportamientos exclusivos.
R4	Regla de claridad	Los enunciados deben ser inequívocos, precisos y de fácil comprensión.	Se revisó el lenguaje técnico y pedagógico de la Tabla 11.
R5	Regla de evidencia	Toda afirmación debe sustentarse en evidencias o productos verificables.	Los criterios refieren a entregables del entorno ESTER (código, simulaciones, logs).
R6	Regla de consistencia	Los criterios deben mantener coherencia lógica y semántica en toda la rúbrica.	Se aplicó formato y estructura uniforme entre criterios.
R7	Regla de independencia	Ningún criterio debe depender del cumplimiento de otro.	Cada dimensión (análisis, control, comunicación) se evalúa por separado.
M1	Meta-regla de alineamiento	Asegura la relación entre competencias, resultados y evaluación.	La rúbrica deriva directamente de la matriz de competencias.
M2	Meta-regla de progresión	Define la secuencia gradual de niveles de dominio.	Los descriptores se agrupan en niveles 1-3.
M3	Meta-regla de derivación	Permite generar instrumentos derivados (listas, lineamientos).	De la rúbrica surgen las Tablas 12 y 13.
M4	Meta-regla de equilibrio	Garantiza el balance entre dimensiones técnicas, cognitivas y actitudinales.	La rúbrica integra desempeño técnico, análisis y colaboración.
M5	Meta-regla de comprensibilidad	Asegura el uso de lenguaje inclusivo y comprensible.	Se evitó jerga técnica excesiva y se mantuvo claridad semántica.
M6	Meta-regla de evaluación 360°	Habilita auto, co y heteroevaluación.	Los Micro TP incluyen evaluación entre pares y docente.
M7	Meta-regla de adaptabilidad	La rúbrica debe poder aplicarse en entornos físicos y virtuales.	Adaptada al modelo híbrido físico-digital de ESTER.

8.4.2 Validación de la rúbrica mediante reglas y meta-reglas

La rúbrica principal y los instrumentos pedagógicos derivados fueron diseñados y validados aplicando las siete reglas (R1–R7) y siete meta-reglas (M1–M7) propuestas (Neil, 2024) en el *Marco conceptual para la definición, desarrollo y evaluación de competencias*.

Las **reglas (R1–R7)** aseguran la coherencia interna del instrumento:

- **R1:** cada criterio de evaluación se vincula con un resultado de aprendizaje.

- **R2:** los descriptores se formulan como proposiciones observables y verificables.
- **R3–R5:** los criterios analíticos A^+ y B definen evidencias contrastables, sin superposición y con cuantificadores claros.
- **R6–R7:** los criterios deben ser inequívocos y evaluar componentes distintos del desempeño.

Las **meta-reglas (M1–M7)** garantizan la consistencia global:

- **M1:** alineamiento constructivo entre competencias, resultados y evaluación.
- **M2:** progresión semántica equilibrada de los niveles de dominio.
- **M3:** posibilidad de derivar lineamientos y listas de verificación.
- **M4–M5:** equilibrio de ponderaciones y lenguaje comprensible.
- **M6:** habilitación de evaluación 360 ° (auto, co y heteroevaluación).
- **M7:** adaptabilidad de la rúbrica a entornos virtuales e híbridos como ESTER.

Esta validación metodológica asegura que la estructura analítica de la rúbrica conserve su validez pedagógica y técnica, y que el proceso de evaluación por competencias sea trazable, equitativo y reproducible en el entorno híbrido del gemelo digital

8.5 Micro TP (trabajo práctico focalizado) y trabajo integrador

Los Micro Trabajos Prácticos (Micro-TP) constituyen una serie de instancias formativas progresivas diseñadas para desarrollar competencias técnicas y pedagógicas vinculadas al modelo híbrido de gemelo digital. En esta tesis, los Micro TP no se presentan como actividades aisladas, sino como módulos de una única secuencia didáctica general, donde cada etapa introduce nuevos niveles de complejidad técnica, autonomía del estudiante y articulación con el robot físico.

Cada módulo está alineado con:

- Los Resultados de Aprendizaje planteados en el capítulo.
- La Matriz de Competencias (8.3.2).
- Los Lineamientos de aprendizaje (8.4), que continúan vigentes en cada actividad.
- La Rúbrica analítica completa, que se traslada al Anexo E, pero se cita en este capítulo para mantener coherencia metodológica.

La culminación de los Micro TP es el Trabajo Práctico Integrador (Módulo 5), donde los estudiantes deben aplicar de forma autónoma y colaborativa todos los conceptos adquiridos, resolviendo un

desafío auténtico con impacto real en el simulador y, de forma opcional, en el robot físico TWIN ROBOT.

Tabla 15: 5 *Módulos en 5 encuentros breves con actividades asincrónica.*

Unidad	Robótica Educativa								**Duración**
Tema	Programación y control de robots en el simulador ESTER mediante un modelo híbrido de gemelo digital.								
Resultado de aprendizaje	[Aplica] + [bloques de programación en ATOM+] + [para controlar desplazamientos secuenciales y respuestas a sensores del robot simulado en ESTER] / [ajustando velocidad, dirección y decisiones según consignas del entorno] / [verificando y corrigiendo los resultados en tiempo real].								
	Estrategias didácticas						**Aprendizaje colaborativo**	**Presentación oral multimedia**	
	Evaluación diagnóstica	**Aula invertida**	**Método expositivo dialogado**	**Resolución de problemas**	**Aprendizaje basado en problemas**	**Auto/co-evaluación**			
Módulo 1									
Secuencia	**1→**	**2→**	**3→**	**4→**	**5→**	**6→**			
Sincrónico			x						
Asincrónico	x	x		x	x	x			
Duración (Min.)	30	30	30	30	30	30			180
Total									180
Módulo 2									
Secuencia			**1→**	**2→**	**3→**	**4→**	**5→**		
Sincrónico			x	x					
Asincrónico					x	x	x		
Duración (Min.)			30	30	30	30	30		150
Módulo 3									
Secuencia			**1→**	**2→**	**3→**	**4→**			

Sincrónico			x						
Asincrónico				x	x	x			
Duración (Min.)			30	30	30	30			120
Módulo 4									
Secuencia	**1→**	**2→**	**3→**	**4→**		**5→**			
Sincrónico			x						
Asincrónico	x	x		x		x			
Duración (Min.)	30	30	30	30		30			150
Trabajo final integrador									
Secuencia	**1→**	**2→**	**3→**	**4→**		**5→**	**6→**	**7→**	
Sincrónico			x					x	
Asincrónico	x	x		x		x	x		
Duración (Min.)	30	30	30	30		30	30	30	210

8.6 Secuencia didáctica analítica

Secuencia didáctica general para el desarrollo de competencias.

El proceso formativo se organiza mediante una única secuencia didáctica general que articula los Micro TP (trabajos prácticos focalizados) y el TP integrador. Cada Micro TP no posee una secuencia propia: se implementa a través de un lineamiento específico derivado de la rúbrica principal, que enfoca uno o más descriptores y asegura la continuidad pedagógica dentro del itinerario global.

Como actividad asincrónica

Los estudiantes desarrollarán el trabajo práctico centrado en lógica de ejecución secuencial en ATOM+, logrando desplazamientos precisos y controlados.

8.6.1 Modulo 1

En el encuentro sincrónico

a) El docente evaluará las preguntas y comentarios y se centrará en la explicación de los principales problemas encontrados.
   1. El docente desarrollará el tema en general.
   2. Se abrirá un espacio de consultas e intercambio entre los estudiantes y el docente.

b) El docente presentará el trabajo práctico asociado a la unidad.

TP .1.1 Movimiento básico del robot

Entregar:

- TP. 1.1 Movimiento básico del robot.
- Algoritmo implementado (bloques de movimiento).
- Video de robot funcionando en el simulador (prueba de funcionamiento).
- Descripción de la lógica de funcionamiento (introducción) y planilla de autoevaluación.

8.6.2 Modulo 2

En el encuentro sincrónico

a) El docente evaluará las preguntas y comentarios y se centrará en la explicación de los principales problemas encontrados.

- El docente desarrollará el tema en general.
- Se abrirá un espacio de consultas e intercambio entre los estudiantes y el docente.

TP. 1.2 Lectura de sensores

Entregar:

TP. 1.2 Lectura de sensores

- Algoritmo implementado (bloques de movimiento en ATOM+ o en python google collab).
- Video de robot funcionando en el simulador (prueba de funcionamiento).
- Descripción de la lógica de funcionamiento (introducción, detección del problema, ajuste) y planilla de autoevaluación.

Consideraciones generales

- El docente los evaluará los trabajos prácticos mediante comentarios.
- Se permitirá más de una entrega.
- Los trabajos se entregarán en archivos editables y se enviarán por email.
- Los trabajos prácticos estarán aprobados cuando ya no haya más comentarios del docente.
- Una vez aprobados los TP, se subirán a la plataforma ULTRA.

8.6.3 Modulo 3

En el encuentro sincrónico

a) El docente evaluará las preguntas y comentarios y se centrará en la explicación de los principales problemas encontrados.

- El docente desarrollará el tema en general.
- Se abrirá un espacio de consultas e intercambio entre los estudiantes y el docente.

TP. 1.3 Sincronización física

Entregar:

- TP. 1.3 Sincronización física.
- Algoritmo implementado (bloques de movimiento en ATOM+ o en python con google collab).
- Video de robot funcionando en el simulador (prueba de funcionamiento).
- Descripción de la lógica de funcionamiento (introducción, Conexión, Comando de gemelo digital), planilla de comparación de tiempos del robot físico y su gemelo digital, y planilla de autoevaluación.

Consideraciones generales

- El docente los evaluará los trabajos prácticos mediante comentarios.
- Se permitirá más de una entrega.
- Los trabajos se entregarán en archivos editables y se enviarán por email.
- Los trabajos prácticos estarán aprobados cuando ya no haya más comentarios del docente.
- Una vez aprobados los TP, se subirán a la plataforma ULTRA.

8.6.4 Modulo 4

En el encuentro sincrónico

a) El docente evaluará las preguntas y comentarios y se centrará en la explicación de los principales problemas encontrados.

- El docente desarrollará el tema en general.
- Se abrirá un espacio de consultas e intercambio entre los estudiantes y el docente.

TP. 1.4 Colaboración y optimización

Entregar:

- TP. 1.4 Colaboración y optimización.
- Algoritmo implementado (bloques de movimiento en ATOM+ o en python con google collab).
- Video de robot funcionando en el simulador (prueba de funcionamiento).
- Descripción de la lógica de funcionamiento (introducción, propuesta del problema colaborativo a resolver), planilla de autoevaluación.

Consideraciones generales

- El docente los evaluará los trabajos prácticos mediante comentarios.
- Se permitirá más de una entrega.
- Los trabajos se entregarán en archivos editables y se enviarán por email.
- Los trabajos prácticos estarán aprobados cuando ya no haya más comentarios del docente.
- Una vez aprobados los TP, se subirán a la plataforma ULTRA.

8.6.5 Modulo 5

En el encuentro sincrónico

a) El docente evaluará las preguntas y comentarios y se centrará en la explicación de los principales problemas encontrados.

- El docente desarrollará el tema en general.
- Se abrirá un espacio de consultas e intercambio entre los estudiantes y el docente.

TP. 1.5 Trabajo integrador final

Entregar:

- TP. 1.5 Trabajo integrador final
- Algoritmo implementado (bloques de movimiento en ATOM+ o en python con google collab).
- Video de los robots funcionando en el simulador (prueba de funcionamiento).
- Presentación escrita y en video del equipo exponiendo el trabajo realizado.
- Descripción de la lógica de funcionamiento (introducción, propuesta del problema colaborativo a resolver, testeo), planilla de autoevaluación.

Consideraciones generales

- El docente los evaluará los trabajos prácticos mediante comentarios.
- Se permitirá más de una entrega.
- Los trabajos se entregarán en archivos editables y se enviarán por email.
- Los trabajos prácticos estarán aprobados cuando ya no haya más comentarios del docente.
- Una vez aprobados los TP, se subirán a la plataforma ULTRA.

8.7 Lista de verificación

Las listas de verificación permiten a los estudiantes y docentes realizar un seguimiento del cumplimiento de cada criterio analítico, promoviendo la autoevaluación y la coevaluación formativa.

Tabla 16: *Lista de verificación derivada de la rúbrica (cinco encuentros),*

Perspectiva de análisis	Descriptor	Criterios analíticos	S/N	Comentarios
1. Producción técnica básica	**Lenguaje lógico y simbólico**	Cada instrucción se representa con la sintaxis y semántica correctas dentro de ATOM+, empleando bloques coherentes para el control del robot.	☐	
	Prolijidad y organización	Los programas se desarrollan de forma ordenada, documentando cada bloque y secuencia. Se incluyen comentarios, etiquetas y recursos visuales que facilitan la lectura del código.	☐	
2. Dominio y alcance funcional (Gemelo simulado)	**Definición del dominio y alcance**	Se identifican los límites de operación del robot (espacio virtual, sensores activos, motores controlados). Se evitan errores de configuración o conexión.	☐	
	Integración de sensores y controladores	Se aplican correctamente las reglas de lectura y respuesta de sensores, ajustando la lógica del movimiento según las condiciones del entorno.	☐	
3. Validación y respuesta del sistema híbrido	**Valores críticos y validaciones**	Se registran los puntos de error o latencia y se analizan sus causas. Se verifica que el robot responda coherentemente en ambos entornos (físico y simulado).	☐	
	Intervalos de respuesta	Se comprueba el comportamiento en diferentes condiciones de distancia, velocidad y sincronización. Se mide la estabilidad de la comunicación.	☐	

	Ajuste de parámetros	Se modifican tiempos, intensidades o secuencias según los resultados observados. Se registran las modificaciones y su impacto.	☐	
4. Optimización y mejora iterativa	**Eficiencia y mejora continua**	Se propone al menos una mejora respecto de la versión inicial del programa, optimizando la lógica de ejecución y reduciendo redundancias.	☐	
5. Trabajo final – Integración, colaboración y presentación	**Comunicación y coevaluación**	Se documentan los aportes de cada integrante y se aplican mecanismos de coevaluación de acuerdo con la rúbrica del equipo.	☐	
	Verificación final de resultados	Se verifica la coherencia entre el comportamiento del robot físico y su gemelo digital. Los resultados se documentan y presentan en un informe o video.	☐	
	Síntesis y presentación del proyecto	Se representan visualmente las secuencias, la arquitectura del código y los resultados alcanzados. Se identifica la versión final del programa y su rendimiento.	☐	

8.8 Conclusión

La implementación del modelo híbrido de gemelo digital en el entorno ESTER permitió articular los espacios de simulación y experimentación física dentro de un marco pedagógico basado en competencias. A través del uso de rúbricas analíticas, listas de verificación y Micro TP (trabajo práctico focalizado), se logró operacionalizar la evaluación formativa, transformando los criterios en instrumentos de guía y autoevaluación. Este enfoque coincide con la propuesta (Neil, 2024), quien sostiene que las rúbricas pueden conducir el desarrollo de las competencias cuando son integradas desde el diseño curricular y no solo utilizadas como herramientas de medición.

Desde una perspectiva teórica, el modelo se sustenta en el constructivismo (PIAGET, 1979) y por (Papert, 1980) al situar a los estudiantes en un rol activo de experimentación, resolución de problemas y creación de artefactos digitales funcionales. Asimismo, el alineamiento constructivo propuesto (Biggs, 1999) se evidencia en la coherencia entre los resultados de aprendizaje, las actividades de programación en el simulador y los criterios de evaluación definidos en la rúbrica.

Cada componente del entorno "desde la simulación 3D hasta la sincronización con el robot físico" actúa como evidencia observable del aprendizaje significativo, tal como lo plantea Ausubel enunciado en trabajo de (Bryce & Blown, 2023), al vincular los nuevos conceptos con los saberes previos del estudiante en contextos auténticos.

Los resultados observados durante la aplicación de los micro TP y el trabajo integrador demuestran un incremento en la comprensión de los principios algorítmicos, la capacidad de depuración de código, y la transferencia de conocimiento entre el entorno virtual y el físico. Además, la incorporación de la colaboración entre pares y la coevaluación promueve una cultura de aprendizaje reflexivo, en línea con los postulados (Tobón, 2006) sobre la formación integral y el desarrollo de competencias transversales.

En síntesis, el capítulo evidencia cómo el modelo híbrido de gemelo digital potencia la enseñanza de la robótica al integrar teoría, práctica y evaluación en un ciclo continuo de aprendizaje y mejora. Este enfoque representa un avance hacia un paradigma educativo basado en la evidencia del desempeño, donde la tecnología no sustituye la acción humana, sino que amplifica la capacidad del estudiante para observar, reflexionar y reconstruir su propio conocimiento. La articulación entre el simulador ESTER y el robot TWIN constituye, por tanto, una plataforma de innovación pedagógica con potencial para ser replicada en otros ámbitos STEM y programas universitarios orientados al aprendizaje activo y la experimentación remota.

CAPÍTULO 9

Conclusiones

9.1. Reflexión general

La presente investigación ha demostrado que la integración de un modelo híbrido de gemelo digital en la enseñanza de la robótica resulta no solo factible, sino altamente efectiva para la formación en competencias técnicas, cognitivas y actitudinales. El simulador ESTER, junto al robot físico TWIN ROBOT, ha materializado un entorno distribuido, multiusuario y bidireccional, que permite sincronizar estados y comportamientos en tiempo real. Esta bidireccionalidad, con latencias inferiores a 100 ms, garantiza coherencia entre la simulación y la experiencia física, lo cual representa un avance significativo frente a enfoques unidireccionales reportados en la literatura (Mustafee et al., 2023; Mazumder et al., 2023) .

El modelo propuesto articula tecnologías emergentes (Unity 3D, WebSocket, API REST, sensores virtuales y físicos) con fundamentos pedagógicos sólidos (Papert, 1980; Ausubel, 1963; Tobón, 2006). De este modo, se genera una experiencia formativa integral que trasciende las limitaciones de hardware, fomenta el aprendizaje activo y significativo, habilitando las prácticas colaborativas en tiempo real.

9.2. Logros alcanzados

- La evolución del trabajo desarrollado se refleja en una serie de hitos técnicos y pedagógicos que consolidan la madurez del modelo híbrido propuesto:

- Evolución del simulador ESTER: desde sus primeras versiones (ESTER 1.0 y 2.0), concebidas como entornos de simulación para un único robot en escenarios simples, el simulador transitó hacia ESTER 3.0, que introdujo la capacidad de trabajar en red y habilitó el control simultáneo de múltiples robots virtuales. Con ESTER 4.0 y 4.1 se incorporaron la sincronización en la nube y el soporte para hasta 30 usuarios en paralelo, resolviendo limitaciones de hardware y acceso. Finalmente, la versión actual, ESTER 5.0, integra multipresencia, arquitectura distribuida, sincronización bidireccional en tiempo real con robots físicos y visualización inmersiva en primera y tercera persona. Esta trayectoria

evidencia un proceso iterativo de mejora que acompañó las necesidades pedagógicas detectadas en las distintas etapas de validación.

- Desarrollo del TWIN ROBOT: en paralelo a la evolución del simulador, se diseñaron y construyeron dos versiones del robot físico. El TWIN ROBOT V1 permitió validar la comunicación bidireccional con el simulador en tiempo real, mientras que el V2 optimizó el diseño mecánico mediante impresión 3D, reduciendo tiempos de armado y costos de fabricación, y manteniendo compatibilidad con placas de control ampliamente accesibles (Arduino, ESP32). Su bajo costo y facilidad de replicación lo hacen escalable para su uso en aulas universitarias y de posgrado con múltiples estudiantes.
- Implementación de ATOM+: el desarrollo de un entorno de programación visual con bloques personalizados, basado en TurboWarp, permitió ampliar significativamente la inclusión. ATOM+ facilitó que estudiantes sin experiencia previa en programación accedieran a la robótica educativa, mientras que aquellos con mayor dominio pudieron realizar la transición hacia lenguajes textuales como Python. Este enfoque híbrido redujo la brecha de entrada y potenció la motivación estudiantil, en línea con el paradigma del construccionismo (Papert, 1980).
- Validación empírica y reconocimiento: la propuesta fue aplicada en carreras técnicas (Ingeniería en Sistemas), carreras no técnicas y trayectos de posgrado (Maestría en Tecnología Educativa), alcanzando a más de 250 estudiantes. Los resultados obtenidos evidencian mejoras en la comprensión de estructuras algorítmicas, en la motivación y en el desarrollo de competencias transversales. Además, los avances fueron reconocidos en congresos y distinguidos con premios académicos como el Premio a las Buenas Prácticas Educativas UAI 2024, consolidando el impacto científico y educativo del proyecto.

9.2.1 – Evolución del simulador ESTER (Versiones 1.0 a 5.0)

La evolución del simulador ESTER entre los años 2021 y 2025 refleja un proceso de desarrollo progresivo que combina innovación tecnológica, validación pedagógica y aplicación en contextos reales de enseñanza universitaria.

Desde sus primeras versiones, el simulador se concibió como un entorno 3D para la enseñanza de robótica y programación, orientado a facilitar prácticas seguras, accesibles y colaborativas en carreras técnicas y no técnicas.

Entre 2021 y 2023, el proyecto incorporó la arquitectura multiusuario distribuida, permitiendo el control simultáneo de múltiples robots virtuales conectados a través de red local o en la nube. En 2024, se añadió el módulo ATOM+, que integra la programación visual mediante bloques inspirada en *Scratch* y adaptada en *TurboWarp*, facilitando la participación de estudiantes sin experiencia previa en lenguajes textuales y promoviendo la enseñanza por proyectos en entornos multipresenciales.

Finalmente, durante 2025, se alcanzó una etapa clave con la incorporación del modelo híbrido de gemelo digital, que sincroniza en tiempo real los robots simulados con los robots físicos TWIN ROBOT. Esta implementación permitió que los datos sensoriales, las colisiones y los movimientos del robot real se reflejaran directamente en el entorno virtual, consolidando la integración entre el mundo físico y el digital.

El video documental publicado por NeoroboticTV (Balich, 2024) resume visualmente esta evolución, mostrando el progreso del proyecto desde sus primeras versiones hasta la fase previa al gemelo digital. Este registro audiovisual complementa la evidencia técnica y educativa recopilada, ofreciendo una visión integral de cómo ESTER pasó de ser un simulador de robótica convencional a constituirse en un entorno híbrido distribuido en tiempo real.

En la (Tabla 17) se sintetizan las principales versiones del simulador, sus características técnicas y los aportes pedagógicos asociados a cada etapa (figura 23).

Tabla 17: *evolución del simulador Ester*

Versión	Características principales	Aportes pedagógicos
ESTER 1.0 (2018)	Primer prototipo de simulador 3D para un único robot; escenarios básicos en Unity.	Introducción de prácticas iniciales de programación y control.
ESTER 2.0 (2019)	Mejora gráfica y física; incorporación de sensores virtuales simples.	Posibilidad de replicar ejercicios de robótica física en un entorno seguro y accesible.
ESTER 3.0 (2021)	Extensión multiusuario en red local; soporte para control simultáneo de múltiples robots virtuales.	Fomento del trabajo colaborativo y actividades de programación distribuida.
ESTER 4.0–4.1 (2023)	Integración con la nube; comunicación distribuida; soporte hasta 30 usuarios en paralelo; sincronización básica con hardware físico.	Democratización del acceso remoto a prácticas de robótica y resolución de la limitación de kits físicos en el aula.
ESTER 5.0 (2025)	Multipresencia; arquitectura híbrida bidireccional; sincronización en tiempo real con TWIN ROBOT; soporte XR y escenarios avanzados.	Consolidación de un modelo híbrido accesible, escalable y validado pedagógicamente en grado y posgrado.

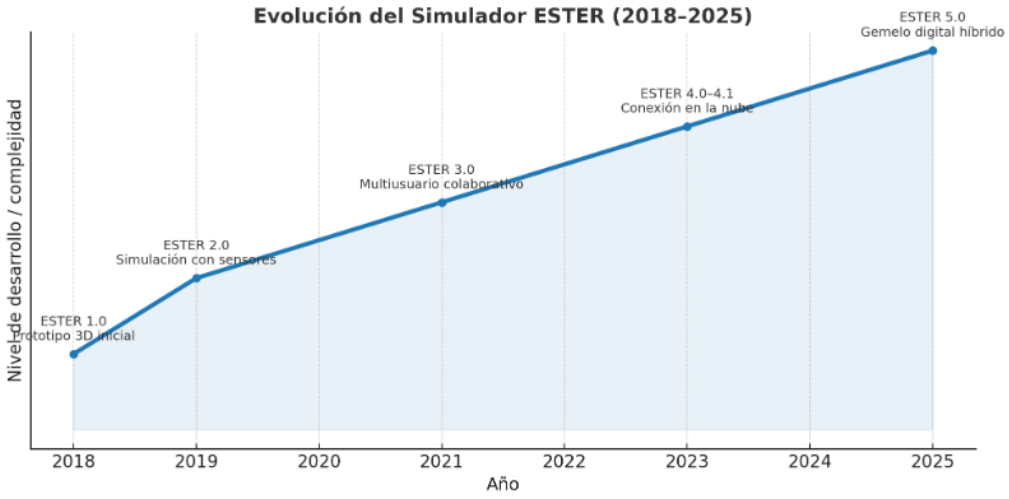


Figura 23. Evolución del simulador ESTER (2018–2025)

9.3. Impacto educativo y tecnológico

El impacto de esta tesis trasciende la construcción de un sistema funcional, al consolidarse como una propuesta educativa integral y validada. En el plano pedagógico, los aportes principales son:

- Democratización del acceso: el modelo permite realizar prácticas de robótica incluso en ausencia de hardware físico, eliminando la dependencia de kits costosos y reduciendo barreras de entrada para instituciones con recursos limitados.
- Articulación teoría–práctica: los entornos simulados dialogan en tiempo real con el mundo físico, posibilitando que cada algoritmo diseñado se valide en condiciones auténticas. Esto potencia aprendizajes significativos y disminuye la brecha entre la comprensión conceptual y la aplicación práctica.
- Fomento de la participación y la creatividad: la interactividad del entorno, junto con la inclusión de ATOM+, impulsa la exploración, el ensayo y error, y el desarrollo de soluciones originales, en línea con las pedagogías activas y el construccionismo.
- Ampliación de la cobertura disciplinar: al ser aplicable en carreras no informáticas y en trayectos de formación docente, la propuesta abre oportunidades de alfabetización digital y robótica en ámbitos tradicionalmente alejados de la programación.

Desde el punto de vista tecnológico, los resultados evidencian que es posible construir sistemas robustos, escalables y eficientes que:

- operan en tiempo real con múltiples usuarios y dispositivos conectados;
- mantienen latencias reducidas (<100 ms), asegurando la coherencia entre lo simulado y lo físico;
- funcionan en red local o en la nube, garantizando accesibilidad en diversos contextos;
- no requieren software propietario ni infraestructura costosa, lo que fortalece su replicabilidad y sostenibilidad.

En conjunto, el impacto educativo y tecnológico del modelo híbrido validado en esta tesis demuestra que los gemelos digitales en educación no solo constituyen una herramienta de simulación, sino también un recurso transformador capaz de ampliar el acceso, motivar a los estudiantes y formar competencias clave del siglo XXI, tales como el pensamiento computacional, la resolución de problemas y la colaboración interdisciplinaria.

Los resultados empíricos obtenidos mediante encuestas (Anexo C) corroboran el impacto del entorno híbrido.

9.3.1 – Impacto educativo y tecnológico del modelo híbrido ESTER + TWIN ROBOT

El desarrollo y la aplicación del modelo de gemelo digital híbrido ESTER + TWIN ROBOT fueron evaluados rigurosamente para determinar su contribución en los ámbitos pedagógico y tecnológico. En la (Tabla 18) presenta un resumen de los principales impactos observados en estas dos dimensiones, destacando cómo el sistema aborda desafíos clave en la enseñanza de la robótica y el potencial de su arquitectura distribuida.

Tabla 18: *Impacto del modelo hibrido*

Dimensión	Impactos observados
Educativo	• Democratiza el acceso a prácticas de robótica, incluso sin hardware físico. • Articula teoría y práctica con validación en entornos reales. • Potencia la motivación, la creatividad y la resolución de problemas. • Amplía la cobertura a carreras no técnicas y formación docente. • Favorece la inclusión de estudiantes con distintos niveles de experiencia en programación.
Tecnológico	• Sistema robusto y escalable con múltiples usuarios simultáneos. • Baja latencia (<100 ms) que asegura coherencia entre simulador y robot. • Ejecución en red local y en la nube, con soporte multiusuario y multipresencia. • Independencia de software propietario, lo que favorece replicabilidad y sostenibilidad. • Integración fluida de programación visual (ATOM+) y lenguajes textuales (Python).

9.3.2. Limitaciones y líneas abiertas

Si bien los resultados de esta investigación son alentadores, se identifican limitaciones que abren nuevas oportunidades de mejora e investigación:

- Accesibilidad y costos: aunque el TWIN ROBOT fue diseñado como una solución low”cost, su replicación a gran escala todavía requiere recursos financieros y soporte técnico especializado. Esto limita su adopción en instituciones con restricciones presupuestarias. Una línea futura será el diseño de kits aún más económicos, modulares y de código abierto, que reduzcan la dependencia de insumos específicos y favorezcan la sostenibilidad (Zhou, Zhang, et al., 2022) .
- Integración curricular: la validación del modelo se realizó en asignaturas y experiencias puntuales de grado y posgrado. No obstante, aún resta consolidar una integración longitudinal en los planes de estudio que permita dar continuidad a las experiencias a lo largo de distintas materias y niveles. El desafío consiste en articular los resultados de aprendizaje, las metodologías y la evaluación bajo un marco de alineamiento constructivo (Biggs, 1999).
- Evaluación longitudinal: la evidencia recogida proviene de cohortes acotadas y de evaluaciones en períodos breves. No se dispone todavía de estudios longitudinales que analicen el impacto sostenido en la adquisición de competencias a lo largo del tiempo. Futuras investigaciones deberían incorporar diseños experimentales o cuasi-experimentales con seguimiento extendido (Mustafee et al., 2023) .
- Expansión a otros niveles educativos: la propuesta se validó principalmente en educación superior y posgrado. Su aplicación en primaria y secundaria se presenta como una línea abierta crucial para democratizar el acceso temprano a la robótica y al pensamiento computacional (Papert, 1991; Valls Pou et al., 2022).
- Analítica de aprendizaje e inteligencia artificial: la integración de técnicas de learning analytics y sistemas de tutoría inteligente podría enriquecer la experiencia formativa, personalizando trayectorias de aprendizaje y ofreciendo retroalimentación adaptativa. El modelo híbrido abre la posibilidad de recolectar datos ricos de desempeño, lo que constituye una oportunidad para avanzar hacia ecosistemas educativos más inteligentes (Singh et al., 2025) .

Las limitaciones observadas no constituyen obstáculos insalvables, sino más bien puntos de partida para el desarrollo futuro. La superación de estos desafíos permitirá consolidar al modelo híbrido ESTER–TWIN ROBOT como una propuesta integral, replicable y sostenible en distintos contextos educativos y profesionales.

9.3.3 – Limitaciones identificadas y líneas futuras de investigación

Aunque el modelo híbrido ESTER + TWIN ROBOT demostró un impacto positivo considerable, es fundamental reconocer las limitaciones inherentes a su fase actual de desarrollo y aplicación. La (Tabla 19) detalla las principales restricciones identificadas (desde desafíos de accesibilidad hasta áreas de mejora en la integración curricular y la evaluación) y propone las líneas futuras de investigación que buscan superar estas barreras para consolidar y expandir la propuesta.

Tabla 19: limitaciones futuras

Limitación	Descripción	Línea futura propuesta
Accesibilidad y costos	El TWIN ROBOT es low-cost, pero su replicación a gran escala aún requiere financiamiento y soporte técnico.	Diseñar kits más económicos, modulares y de código abierto para favorecer la sostenibilidad y la adopción masiva.
Integración curricular	La validación se aplicó en experiencias puntuales, sin continuidad en planes de estudio completos.	Incorporar el modelo en trayectorias longitudinales y articular objetivos, actividades y evaluación bajo alineamiento constructivo.
Evaluación longitudinal	Los estudios realizados son de corto plazo y no permiten medir impacto sostenido.	Desarrollar investigaciones longitudinales y diseños experimentales con seguimiento extendido de cohortes.
Expansión a otros niveles educativos	Aplicación validada en educación superior y posgrado, con escasa presencia en niveles iniciales y medios.	Extender la propuesta a primaria y secundaria para democratizar el acceso temprano a la robótica y al pensamiento computacional.
Analítica de aprendizaje e IA	Actualmente no se explota el potencial de datos generados por el entorno.	Integrar técnicas de learning analytics y sistemas de tutoría inteligente para personalizar la experiencia formativa y enriquecer la retroalimentación.

9.4. Contribución al campo

El desarrollo del simulador ESTER y del robot físico TWIN ROBOT constituye una contribución original en la intersección entre la robótica educativa, los gemelos digitales y la tecnología educativa. A diferencia de los enfoques reportados en la literatura, que en su mayoría se limitan a la visualización de datos o a la simulación unidireccional, el modelo propuesto implementa una arquitectura híbrida,

bidireccional y multiusuario, validada en contextos reales de enseñanza universitaria y de posgrado (Mustafee et al., 2023; Mazumder et al., 2023) .

Las contribuciones principales pueden organizarse en tres dimensiones:

1. **Técnica**
    a. Desarrollo de un sistema basado en Unity 3D con comunicación en tiempo real mediante WebSocket y API REST.
    b. Sincronización fluida entre simulador y robots físicos con latencia inferior a 100 ms.
    c. Integración de sensores virtuales y físicos que garantizan coherencia en la representación híbrida.
    d. Arquitectura distribuida y escalable, ejecutable tanto en red local como en la nube.
2. **Pedagógica**
    a. Diseño de un entorno que articula el construccionismo (Papert, 1980), el aprendizaje significativo Ausubel (Bryce & Blown, 2023; Novak, 2010), el alineamiento constructivo (Biggs, 1999), y la educación basada en competencias (Tobón, 2006).
    b. Inclusión de estudiantes con distintos niveles de experiencia en programación gracias a la implementación de ATOM+, que facilita la transición de la programación visual a la textual.
    c. Generación de prácticas accesibles, motivadoras y contextualizadas en problemas reales, que fortalecen competencias técnicas, cognitivas y actitudinales.
3. **Científica**
    a. Aporte al campo emergente de los gemelos digitales en educación, un área donde la evidencia empírica todavía es incipiente y fragmentada (Balich, Guanipa Márquez, et al., 2025).
    b. Generación de producción académica validada en congresos y revistas indexadas, que documenta la aplicabilidad del modelo en contextos diversos.
    c. Propuesta replicable y adaptable a otras disciplinas STEM y a contextos no técnicos, lo que amplía el alcance del paradigma de los gemelos digitales más allá de la ingeniería.

La presente tesis demuestra que los gemelos digitales híbridos pueden superar la brecha entre la simulación virtual y la práctica física, constituyéndose en un recurso pedagógico transformador. El

entorno ESTER no solo aporta un avance técnico en términos de sincronización y multipresencia, sino que también ofrece un marco pedagógico robusto y validado, lo que lo convierte en una referencia para el diseño de futuros ecosistemas híbridos en educación.

9.5. Validación del enfoque por competencias

La validación del modelo híbrido se llevó a cabo bajo el marco de la Educación Basada en Competencias (EBC), lo que permitió evaluar aprendizajes más allá de la simple ejecución técnica de programas. Este enfoque reconoce que la formación profesional exige la integración de conocimientos, habilidades y actitudes en contextos auténticos (Tobón, 2006).

Para operacionalizar esta perspectiva, se diseñó una rúbrica específica de evaluación competencial, que contempló cuatro dimensiones:

- **Técnica**: dominio en la programación, implementación y ejecución de algoritmos en simulador y robot físico.
- **Cognitiva**: capacidad de análisis, resolución de problemas y aplicación de lógica en escenarios híbridos.
- **Actitudinal**: participación, colaboración con pares y compromiso con las tareas.
- **Metacognitiva**: reflexión crítica sobre el proceso, diagnóstico de errores y mejora continua.

La rúbrica fue aplicada en distintos contextos de grado y posgrado, y permitió implementar evaluación formativa, autoevaluación y coevaluación, brindando una retroalimentación integral. Este procedimiento se alinea con el principio de alineamiento constructivo (Biggs, 1999), al garantizar coherencia entre los resultados de aprendizaje esperados, las actividades prácticas y los criterios de evaluación.

Los resultados empíricos confirmaron que los estudiantes desarrollaron competencias integrales, evidenciadas tanto en su desempeño técnico como en su capacidad de reflexión y colaboración. Asimismo, se observó que la estructura de la rúbrica facilitó la motivación y la autoeficacia, al ofrecer una guía clara sobre los niveles de logro alcanzables.

En este sentido, la validación por competencias no solo demostró la efectividad del modelo híbrido en términos de aprendizaje técnico, sino que también aseguró su pertinencia pedagógica y su

alineación con los lineamientos institucionales de la Universidad Abierta Interamericana (UAI, 2025), que promueven la formación integral y transferible a escenarios profesionales reales.

9.6. Conclusiones

La presente tesis ha demostrado que los gemelos digitales híbridos constituyen una herramienta pedagógica y tecnológica transformadora para la enseñanza de la robótica y la programación. A través del desarrollo del simulador ESTER y del robot físico TWIN ROBOT, se comprobó que es posible integrar en un mismo entorno la simulación virtual, la interacción física y la programación visual y textual, generando experiencias formativas de alto impacto.

Los hallazgos más relevantes pueden sintetizarse en los siguientes puntos:

- **Baja latencia y coherencia híbrida**: la arquitectura distribuida garantiza tiempos de respuesta inferiores a 100 ms, lo que asegura consistencia entre el comportamiento simulado y el físico, superando la limitación de modelos unidireccionales reportados en la literatura (Mustafee et al., 2023; Mazumder et al., 2023) .
- **Democratización del acceso**: el modelo reduce la dependencia de hardware costoso, habilitando la realización de prácticas en escenarios con recursos limitados, tanto en red local como en la nube.
- **Impacto pedagógico comprobado**: la propuesta favorece aprendizajes activos, significativos y colaborativos, en consonancia con teorías como el construccionismo (Papert, 1980;Saldarriaga et al., 2020) el aprendizaje significativo (Bryce & Blown, 2023; Novak, 2010) y el alineamiento constructivo (Biggs, 1999).
- **Competencias del siglo XXI**: los estudiantes desarrollaron pensamiento computacional, resolución de problemas, creatividad y colaboración, competencias transversales reconocidas como esenciales en entornos STEM (Tobón, 2006 ; Valls Pou et al., 2022).
- **Validación empírica y científica**: la implementación alcanzó a más de 250 estudiantes en carreras técnicas y no técnicas, y fue reconocida en congresos, así como con premios institucionales y financiamiento internacional.

Más allá de los logros alcanzados, se identificaron líneas abiertas de investigación:

- consolidar planes curriculares longitudinales que integren el modelo híbrido de forma sostenida;

- extender la propuesta a niveles primarios y secundarios;
- avanzar hacia la incorporación de analítica de aprendizaje e inteligencia artificial, para personalizar la enseñanza y optimizar la retroalimentación;
- continuar reduciendo los costos de replicación para ampliar su adopción en contextos con recursos limitados.

En conclusión, el modelo híbrido desarrollado en esta tesis representa una contribución significativa al campo de la tecnología educativa y la robótica aplicada, al demostrar que es posible diseñar, implementar y validar un ecosistema educativo que combine accesibilidad, escalabilidad, rigor pedagógico y robustez tecnológica. Su originalidad radica en articular simulación, hardware físico, programación visual y textual, y evaluación por competencias en un mismo entorno, ofreciendo una solución replicable y sostenible.

Este trabajo abre el camino hacia una nueva generación de plataformas híbridas para la educación en ingeniería y ciencias aplicadas, en las que los gemelos digitales se consolidan como instrumentos clave para democratizar la robótica, ampliar la alfabetización digital y formar profesionales capaces de enfrentar los desafíos de la sociedad digital del siglo XXI.

La principal contribución de esta tesis doctoral consiste en la formulación, implementación y validación de un modelo híbrido de gemelo digital para la enseñanza de robótica, que integra la dimensión pedagógica con la tecnológica, aportando una arquitectura escalable y replicable en contextos educativos diversos

CAPÍTULO 10

Trabajos futuros

El desarrollo del simulador ESTER y del robot físico TWIN ROBOT ha permitido demostrar la viabilidad y pertinencia pedagógica de los gemelos digitales híbridos en la enseñanza de robótica y programación. No obstante, como toda investigación doctoral, los resultados obtenidos abren nuevas preguntas y desafíos que podrán guiar futuras líneas de trabajo.

10.1. Evolución del robot físico TWIN ROBOT

El prototipo TWIN ROBOT cumplió satisfactoriamente con su función educativa y de sincronización en tiempo real con el entorno simulado. No obstante, se identifican oportunidades de mejora técnica:

- Incorporación de sensores adicionales (LiDAR, cámara, sensor de color).
- Optimización de motores, encoders y estabilidad de tracción.
- Diseño modular que permita añadir brazos manipuladores y nuevos periféricos.
- Reducción del tamaño y consumo energético, favoreciendo su uso en escuelas con restricciones de espacio o autonomía eléctrica.

10.2. Ampliación de ATOM+ con nuevas capacidades

El entorno de programación visual **ATOM+** se consolidó como una herramienta eficaz de acceso inicial a la programación. En sus futuras versiones podrían incorporarse:

- Bloques orientados a inteligencia artificial educativa (entrenamiento básico de modelos, redes neuronales simples, detección de imágenes).
- Integración de condiciones contextuales del entorno (hora, clima, sensores remotos).
- Generación automática de informes docentes (registro de bloques utilizados, errores frecuentes, evolución del estudiante).
- Secuencias didácticas gamificadas y adaptadas a distintos niveles o grupos de estudiantes.

Con estas mejoras, ATOM+ podrá evolucionar hacia un ecosistema de programación inclusivo, inteligente y adaptativo, alineado con los avances de la educación digital.

10.3. Incorporación de analítica de aprendizaje (Learning Analytics)

El registro detallado de las interacciones en el entorno híbrido abre la posibilidad de integrar herramientas de analítica de aprendizaje que permitan:

- Detectar patrones de comportamiento y errores recurrentes.
- Identificar estudiantes con dificultades específicas en lógica computacional o estructuras de control.
- Diseñar paneles visuales (dashboards) para docentes, con seguimiento automático del progreso.
- Evaluar el impacto del entorno híbrido en el desarrollo de la autonomía, la creatividad y el pensamiento computacional.
- La integración de Learning Analytics e inteligencia artificial representa una línea estratégica para personalizar la enseñanza y enriquecer la retroalimentación formativa.

10.4. Expansión hacia entornos inmersivos: Realidad Extendida (XR)

El avance de las tecnologías inmersivas abre la posibilidad de extender el simulador hacia la realidad extendida (XR):

- Desarrollo de versiones compatibles con realidad virtual (VR) y aumentada (AR).
- Interacción mediante gestos, comandos de voz o dispositivos hápticos.
- Laboratorios híbridos donde estudiantes remotos manipulen robots reales a través de realidad mixta.
- Experiencias 360° para fortalecer la percepción espacial y la contextualización de las prácticas.
- La integración de XR permitirá acercar a los estudiantes a un metaverso educativo orientado a la robótica.

10.5. Adaptación a nuevos niveles y áreas de aplicación

El modelo híbrido puede expandirse a distintos niveles educativos y disciplinas:

- **Nivel medio y técnico**: prácticas alineadas con contenidos de programación, electrónica y automatización.
- **Educación inicial y primaria**: robots simplificados y bloques visuales intuitivos que introduzcan el pensamiento computacional de manera lúdica.
- **Disciplinas no tecnológicas**: aplicaciones en medicina (robots de asistencia), automatización industrial y ciudades inteligentes (smart cities).

De este modo, el alcance de ESTER y TWIN ROBOT puede trascender el campo de la informática y convertirse en un recurso interdisciplinario.

10.6. Consolidación de una red educativa colaborativa

Una línea estratégica es la creación de una red interinstitucional que utilice ESTER como entorno común de enseñanza, lo cual habilitaría:

- Intercambio de buenas prácticas, proyectos y algoritmos.
- Proyectos de aula espejo entre universidades o escuelas de distintos países.
- Concursos de programación colaborativa entre instituciones educativas.
- Desarrollo de módulos de autoevaluación compartidos por docentes de diversas regiones.

La construcción de una red de este tipo fortalecería la comunidad académica en robótica educativa híbrida y consolidaría a ESTER como referente.

10.7. Expansión hacia arquitecturas distribuidas: Proyecto ESTER GRID Robotics

Se plantea la evolución del simulador ESTER 5.0 a un centrando en la algoritmia dsitribuida con gemelo digital para el mundo física que llamaremos ESTER GRID Robotics (ESTER 6.0), abre una nueva línea de investigación centrada en la coordinación distribuida de múltiples robots físicos y simulados dentro del mismo entorno educativo. Esta evolución natural del simulador busca superar el paradigma de un robot por estudiante y avanzar hacia modelos de robótica colaborativa, enjambres educativos y prácticas multiagente en tiempo real.

El proyecto ESTER GRID introduce capacidades que representan un salto conceptual respecto a la arquitectura actual:

- Orquestación de múltiples robots físicos en simultáneo, cada uno con su propio gemelo digital sincronizado.
- Distribución de cargas de procesamiento, permitiendo que diferentes nodos (PCs de alumnos, servidores, placas remotas) participen de la simulación compartida.
- Asignación dinámica de robots a estudiantes, con gestión automática de colisiones, prioridades y recursos.
- Protocolos optimizados para baja latencia, implementados mediante WebSocket, UDP y mecanismos híbridos.
- Modelos de comportamiento colaborativo, donde los robots puedan coordinarse para tareas de búsqueda, desplazamiento, navegación compartida o construcción de patrones.
- Escenarios multiusuario, en los cuales grupos de estudiantes programan simultáneamente robots que interactúan entre sí dentro del mismo plano simulado o físico.

Desde la perspectiva pedagógica, ESTER GRID permitirá:

- Practicar conceptos avanzados de robótica colaborativa, navegación colectiva y swarming.
- Entrenar habilidades de coordinación, negociación y resolución de conflictos entre programas.
- Explorar problemas aplicados a logística, ciudades inteligentes, coordinación y fabricación distribuida.
- Integrar nuevas actividades gamificadas basadas en competencia o cooperación entre varios robots.

En términos de investigación, esta línea futura abre la puerta a:

- Modelos de sincronización más complejos entre múltiples gemelos digitales.
- Análisis comparado entre interacciones reales y simuladas con varios robots simultáneos.
- Estudio de patrones emergentes en grupos de robots educativos.
- Implementación de arquitecturas tipo fog/edge computing aplicadas a laboratorios remotos.

10.8. Publicaciones futuras y diseminación

A partir de los resultados obtenidos, se proyecta:

- Publicación de artículos científicos en revistas de alto impacto.
- Participación en congresos internacionales sobre gemelos digitales, robótica educativa, inteligencia artificial y simulación.
- Generación de material abierto (repositorios, guías, tutoriales, videos) que facilite la replicación del modelo en otras instituciones.

Estas acciones contribuirán a posicionar la propuesta a nivel global y a fortalecer la transferencia científica y pedagógica.

10.9. Conclusión

El proyecto desarrollado no constituye un producto cerrado, sino un modelo en expansión, adaptable a diferentes niveles formativos, contextos socioeconómicos y disciplinas. Las líneas de trabajo aquí delineadas proyectan a ESTER y TWIN ROBOT como una plataforma de referencia en educación tecnológica, capaz de evolucionar técnica y pedagógicamente.

En perspectiva, el futuro del modelo se orienta hacia entornos más inteligentes, inclusivos e inmersivos, donde los gemelos digitales híbridos se consoliden como una herramienta clave para democratizar la robótica educativa, fortalecer la alfabetización digital y formar profesionales preparados para los desafíos de la sociedad digital del siglo XXI.

Referencias

Acker, J., Rogers, I., Guerra-Zubiaga, D., Tanveer, M. H., & Moghadam, A. A. A. (2023). Low-cost digital twin approach and tools to support industry and academia: A case study connecting high-schools with high degree education. *Machines*, *11*(9), 860. https://doi.org/https://www.mdpi.com/2075-1702/11/9/860

Acosta, B., Yang, W., & Posa, M. (2022). Validating Robotics Simulators on Real-World Impacts. *IEEE Robotics and Automation Letters*, *7*(3), 6471–6478. https://doi.org/10.1109/LRA.2022.3174367.

Ananias, E., & Gaspar, P. D. (2022). A Low-Cost Collaborative Robot for Science and Education Purposes to Foster the Industry 4.0 Implementation. *Applied System Innovation*, *5*(4), 72. https://doi.org/10.3390/asi5040072

Ausubel, D. P. (1962). A subsumption theory of meaningful verbal learning and retention. *The Journal of General Psychology*, *66*(2), 213–224. https://doi.org/10.1080/00221309.1962.9711837

Ausubel, D. P. (1963). *The Psychology of Meaningful Verbal Learning: An Introduction to School Learning*. Grune & Stratton. worldcat.org+3Google Libros+3Archivo de Internet+3

Autodesk. (2025). *Tinkercad Circuits [Plataforma de modelado y simulación de circuitos]*. https://www.tinkercad.com/

Bachmann, J. E. C., Silveira, I. F., & Martins, V. F. (2024). Digital Twins for Education: A Literature Review. In *Simpósio Brasileiro de Informática na Educação (SBIE* (pp. 722–736). https://doi.org/https://sol.sbc.org.br/index.php/sbie/article/view/31280

Balich, , N. A., Balich, B. L., & Regal, V. J. (2025). Implementación de Robot Físico con Gemelo Digital para el Simulador Ester . *Congreso de Tecnología En Educación y Educación En Tecnología*. https://sedici.unlp.edu.ar/handle/10915/185505

Balich, , N. A., Balich, F. A., & Balich, B. L. (2024). Simulador de robótica 3D multirobot para enseñanza de programación. *Congreso Nacional de Ingeniería Informática / Sistemas de Información*.

Balich, B. L., Ocampo Orozco, T. A., Balich, F. A., & Balich, N. A. (2023). *Simulador de robótica en la nube para prácticas de programación colaborativa*. Congreso Nacional de Ingeniería Informática / Sistemas de Información.

https://dspaceapi.uai.edu.ar/server/api/core/bitstreams/f271e269-6acf-4ed0-8fe2-c0d6967a40b8/content

Balich, N. (2017). Cloud Robotic. *Revista Abierta de Informática Aplicada*, *1*(1).

Balich, N. A. (2009). *CONSTRUCCIÓN DE ROBOTS AUTÓNOMOS COLABORATIVOS*.

Balich, N. A. (2024). Simulador ESTER: evolución del proyecto 2021–2024 [Video]. *YouTube*.

Balich N. A., & Balich B. L. (2025). ATOM+ Integración de Programación Visual y Gemelos Digitales en el Simulador ESTER para la Enseñanza de Programación y Robótica. *Jornada Argentina de Informática e Investigación Operativa*, *Simposio Argentino de Educación en Informática*. https://54jaiio.sadio.org.ar/trabajos-aceptados/

Balich, N. A., Balich, F. A., Balich, B. L., Yuan, H., & Maghi Scheck, M. (2025). Sincronización multi-robot en tiempo real para el simulador ESTER y gemelo digital. *Congreso Argentino de Ciencias de La Computación*. https://conaiisi2025.frc.utn.edu.ar/trabajos/

Balich N. A., Balich F. A., Ocampo T., & Balich B. L. (2024). Desarrollo de simulador de robótica cloud multipresencia para enseñanza de programación. *Congreso Argentino de Ciencias de La Computación*.

Balich, N. A., Cena, A., & Fransoy, A. (2014). Interface natural emotiva niño-robot. *Workshop de Investigadores En Ciencias de La Computación*. https://sedici.unlp.edu.ar/handle/10915/41579

Balich N. A., Guanipa Márquez C. A., Balich B. L., & Rossi Gustavo. (2025). Gemelos Digitales en la Educación en Robótica: Un Mapeo Sistemático de la Literatura. *Congreso Nacional de Ingeniería Informática / Sistemas de Información*.

Balich, N., Ocampo, T., & Balich, B. (2023). Simulador 3D de robótica distribuido en Unity para enseñanza de programación e inteligencia artificial. *Revista Abierta de Informática Aplicada*, *7*, 24–36. https://www.researchgate.net/publication/374119176_Simulador_3D_de_robotica_distribuido_en_Unity_para_ensenanza_de_programacion_e_inteligencia_artificial

Bell, S. (2010). Project-based learning for the 21st century: Skills for the future. *The Clearing House*, *83*(2), 39–43. https://doi.org/10.1080/00098650903505415

Bencomo, N., Cabot, J., Chechik, M., Cheng, B. H. C., Combemale, B., Wąsowski, A., & Zschaler, S. (2024). *Abstraction Engineering*. https://doi.org/10.48550/arXiv.2408.14074

Bertossi, V. (2024). Taller de definición, desarrollo y evaluación de competencias: Trabajo práctico integrador. *RIIZIC*.

Biggs, J. (1991). What the student does: Teaching for enhanced learning. *Higher Education Research & Development*, *18*(1), 57–75. https://doi.org/10.1080/0729436990180105

Biggs, J. (1996). Enhancing teaching through constructive alignment. *Higher Education*, *32*(3), 347–364. https://link.springer.com/article/10.1007/bf00138871

Biggs, J. (1999). *Teaching for quality learning at university: What the student does*. Open University Press.

Biggs, J. B. (1999). *Teaching for Quality Learning at University*. Open University Press. letr.org.uk

Biggs, J. B. (2001). What the Student Does: Teaching for Enhanced Learning. *Higher Education Research & Development*, *18*(1), 57–75. https://www.tru.ca/__shared/assets/Constructive_Alignment36087.pdf?utm_source=chatgpt.com

Bonwell, C. C., & Eison, J. A. (1991). Active Learning: Creating Excitement in the Classroom. *ASHE-ERIC Higher Education Report*, *1*. https://eric.ed.gov/?id=ED336049

Botín-Sanabria, D. M., Mihaita, A. S., Peimbert-García, R. E., Ramírez-Moreno, M. A., Ramírez-Mendoza, R. A., & Lozoya-Santos, J. D. J. (2022). Digital twin technology challenges and applications: A comprehensive review. *Remote Sensing*, *14*(6), 1335. https://doi.org/10.3390/rs14061335

Bryce, T. G. K., & Blown, E. J. (2023). Ausubel's meaningful learning re-visited. *Current Psychology*, *43*(5), 4579–4598. https://link.springer.com/article/10.1007/s12144-023-04440-4

Calderón, O. A. R. (2024). Fundamentos teóricos del uso de la robótica educativa. *Ciencia Latina: Revista Multidisciplinar*, *8*(1), 6364–6375. https://doi.org/https://dialnet.unirioja.es/servlet/articulo?codigo=9430312

Calderón, S., Silva, O., Bournissen, J. M., & Tumino, M. C. (2024). Impacto de la implementación de un robot programable en la enseñanza de la programación en el nivel medio: Un estudio de caso. *RIEE | Revista Internacional de Estudios en Educación*, *24*(1), 19–40. https://doi.org/https://riee.um.edu.mx/index.php/RIEE/article/view/323

Chen, T. I., Lin, S. K., & Chung, H. C. (2023). Gamified educational robots lead an increase in motivation and creativity in STEM education. *Journal of Baltic Science Education*, *22*(3), 427–438. https://doi.org/https://www.ceeol.com/search/article-detail?id=1127378

Chen, W., Yaguchi, Y., Naruse, K., Watanobe, Y., Nakamura, K., & Ogawa, J. (2018). A study of robotic cooperation in cloud robotics: Architecture and challenges. *IEEE Access*, *6*, 36662–36682. https://doi.org/https://ieeexplore.ieee.org/abstract/document/8403209

Cogmation Robotics. (2025). *Virtual Robotics Toolkit [Simulador robótico]*. https://www.virtualroboticstoolkit.com/?utm_source=chatgpt.com

Condliffe, B., Quint, J., Visher, M. G., Bangser, M. R., Drohojowska, S., Saco, L., & Nelson, E. (2017). *Project-based learning: A literature review*. MDRC.

Coppelia Robotics AG. (2025). *CoppeliaSim [Simulador robótico]*. https://www.coppeliarobotics.com/?utm_source=chatgpt.com

Coşkunserçe, O. (2023). Comparing the use of block-based and robot programming in introductory programming education: Effects on perceptions of programming self-efficacy. *Computer Applications in Engineering Education*, *31*(5), 1234–1255. https://doi.org/10.1002/cae.22637

Cyberbotics Ltd. (2025). *Webots [Simulador robótico]*. https://cyberbotics.com/?utm_source=chatgpt.com

David, J., Lobov, A., & Lanz, M. (2018). Learning experiences involving digital twins. *2018 44th Annual Conference of the IEEE Industrial Electronics Society (IECON*, 3681–3686. https://ieeexplore.ieee.org/document/8591460

Dihan, M. S., Akash, A. I., Tasneem, Z., Das, P., Das, S. K., Islam, M. R., & Hasan, M. M. (2024). Digital twin: Data exploration, architecture, implementation and future. *Heliyon*, *10*(5). https://doi.org/10.1016/j.heliyon.2024.e26247

Eguchi, A. (2014). Robotics as a learning tool for educational transformation. *Proceeding of 4th International Workshop Teaching Robotics, Teaching with Robotics & 5th International Conference Robotics in Education Padova (Italy*, *1*, 1–6. https://www.terecop.eu/TRTWR-RIE2014/files/00_WFr1/00_WFr1_04.pdf

E.S.A. (2022). *European Space Agency*. EO Portal. https://www.eoportal.org/other-space-activities/digital-twins?utm_source=chatgpt.com

Feijoo-Almonacid, A., & Rodriguez-Garavito, C. H. (2021). Hardware-software platform for the development of STEM skills. *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, *17*(2), 170–177. https://doi.org/10.1109/RITA.2022.3155791

Filipescu, A., Simion, G., Ionescu, D., & Filipescu, A. (2024). IoT-Cloud, VPN, and Digital Twin-Based Remote Monitoring and Control of a Multifunctional Robotic Cell in the Context of AI, Industry, and Education 4.0 and 5.0. *Sensors*, *24*(23), 7451. https://doi.org/https://www.mdpi.com/1424-8220/24/23/7451

Folk, M. (1981). Review of" Mindstorms: Children, Computers, and Powerful Ideas by Seymour Papert". *ACM SIGCUE Outlook*, *15*(1), 23–24. https://doi.org/10.1145/1045071.1045074

Fuller, A., Fan, Z., Day, C., & Barlow, C. (2020). Digital twin: Enabling technologies, challenges and open research. *IEEE Access*, *8*, 108952–108971. https://doi.org/10.1109/ACCESS.2020.2998358

Gelernter, D. (1991). *Mirror Worlds: Or the day software puts the universe in a shoebox... How it will happen and what it will mean*. Oxford University Press. https://global.oup.com/academic/product/mirror-worlds-9780195079067?cc=ar&lang=en&

goCoderZ. (2025). *CoderZ [Plataforma de robótica educativa]*. https://gocoderz.com/?utm_source=chatgpt.com

Grieves, M. (2014). *Digital Twin: Manufacturing Excellence through Virtual Factory Replication*. ResearchGate.

Grieves, M. (2024). Digital Twins and Their Role in Reengineering Engineering Education. In *Digital Twin* (pp. 237–261). Springer. https://doi.org/10.1007/978-3-031-67778-6_11

Grieves, M., & Vickers, J. (2016). Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems. In *Transdisciplinary Perspectives on Complex Systems* (pp. 85–113). Springer. https://link.springer.com/chapter/10.1007/978-3-319-38756-7_4

Grieves M W. (2005). Product lifecycle management: the new paradigm for enterprises. *International Journal of Product Development*, *2*(1–2), 71–84. https://www.inderscienceonline.com/doi/abs/10.1504/IJPD.2005.006669

Güleryüz, H., & Dilber, R. (2021). STEM activities with robotic coding; The effect on awareness of teacher candidates regarding its use in science lessons. *International Journal of Engineering Technologies and Management Research*, *8*(11), 37–54. https://doi.org/10.29121/ijetmr.v8.i11.2021.1029

Hu, C. C., Yang, Y. F., Cheng, Y. W., & Chen, N. S. (2024). Integrating educational robot and low-cost self-made toys to enhance STEM learning performance for primary school students. *Behaviour & Information Technology*, *43*(8), 1614–1635. https://doi.org/10.1080/0144929X.2023.2222308

I.B.M. (2022). *Point to Ponder: The History of Digital Twin*. IBM Data Science Community. https://community.ibm.com/community/user/blogs/moloy-de1/2022/12/07/point-to-ponder?utm_source=chatgpt.com

Impagliazzo, J., & Xu, X. (2024). A competency-based transformation in computing and engineering education in the digital era. *Frontiers of Digital Education*, *1*(1), 97–108. https://doi.org/10.1007/s44366-024-0025-8

Jacinto, C., Graciana, R., Vinacur, T., & Alegre, S. (2025). Educación técnico profesional en Argentina: desafíos para la articulación del sistema de formación. In *Banco Interamericano de Desarrollo División de Educación Septiembre 2025*. https://publications.iadb.org/publications/spanish/document/Educacion-tecnico-profesional-en-Argentina-desafios-para-la-articulacion-del-sistema-de-formacion.pdf

Jacob, D. R. H., Hafez, ashraf mohamed, & elawa, mohamed T. (2024). Design and Implementation of a Low-Cost Educa-tional Robot for Engineering Curriculums. *Journal of Engineering - Research*, *8*(6, Article 8). https://doi.org/10.70259/engJER.2024.861831

Kafai, Y. B., & Resnick, M. (2012). *Constructionism in practice: Designing, thinking, and learning in a digital world* (Y. B. Kafai & M. Resnick, Eds.). Routledge.

Kartashova, L. A., Gurzhii, A. M., Zaichuk, V. O., Sorochan, T. M., & Zhuravlev, F. M. (2020). Digital twin of an educational institution: An innovative concept of blended learning. *Proceedings of the Symposium on Advances in Educational Technology (AET*. https://www.scitepress.org/Papers/2020/109311/109311.pdf

Kuo, H. C., Yang, Y. T. C., Chen, J. S., Hou, T. W., & Ho, M. T. (2021). The impact of design thinking PBL robot course on college students' learning motivation and creative thinking. *IEEE Transactions on Education*, *65*(2), 124–131. https://doi.org/10.1109/TE.2021.3090801

Kurniawan, O., Lee, N. T. S., Datta, S., Sockalingam, N., & Leong, P. K. (2018). Effectiveness of Physical Robot Versus Robot Simulator in Teaching Introductory Programming. *2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE*, 486–493. https://doi.org/10.1109/TALE.2018.8615190.

Larmer, J., & Mergendoller, J. R. (2010). *The main course, not dessert: How are students reaching 21st century goals with 21st century project-based learning?* Buck Institute for Education.

Li, J., & Liang, W. (2024). Effectiveness of virtual laboratory in engineering education: A meta-analysis. *PLOS ONE*, *19*(12), 316269. https://doi.org/10.1371/journal.pone.0316269

Liljaniemi, A., & Paavilainen, H. (2020). Using digital twin technology in engineering education–course concept to explore benefits and barriers. *Open Engineering*, *10*(1), 377–385. https://doi.org/10.1515/eng-2020-0040/html

López-Nicolás, G., Romeo, A., & Guerrero, J. J. (2014). Active learning in robotics based on simulation tools. *Computer Applications in Engineering Education*, *22*(3), 509–515. https://doi.org/10.1002/cae.20576

Marah, H., & Challenger, M. (2024). Adaptive hybrid reasoning for agent-based digital twins of distributed multi-robot systems. *Simulation*, *100*(9), 931–957. https://doi.org/10.1177/00375497231226436

Mazumder, A., Sahed, M. F., Tasneem, Z., Das, P., Badal, F. R., Ali, M. F., & Islam, M. R. (2023). Towards next generation digital twin in robotics: Trends, scopes, challenges, and future. *Heliyon*, *9*(2). https://doi.org/https://www.cell.com/heliyon/fulltext/S2405-8440(23)00566-2

Moon, H., Cheon, J., & Lee, J. (2020). Teaching block-based programming: A systematic review of current approaches and outcomes. *Society for Information Technology & Teacher Education International Conference*, 73–78. https://www.learntechlib.org/noaccess/215732/

Muratore, F., Ramos, F., Turk, G., Yu, W., Gienger, M., & Peters, J. (2022). Robot Learning From Randomized Simulations: A Review. In *Frontiers in Robotics and AI* (Vol. 9). Frontiers Media S.A. https://doi.org/10.3389/frobt.2022.799893

Mustafee, N., Harper, A., & Viana, J. (2023). Hybrid models with real-time data: Characterising real-time simulation and digital twins. *Proceedings of the Simulation Workshop 2023*, 23. https://doi.org/10.36819/SW23.031

N.A.S.A. (2021). *Digital Twins and Living Models at NASA*. NASA Technical Reports Server (NTRS. https://ntrs.nasa.gov/citations/20210023699?utm_source=chatgpt.com

Neil, C. (2024). *Marco conceptual para la definición, desarrollo y evaluación de competencias. Editorial UAI*. https://uai.edu.ar/Competencias#Libro

Neil, C., Battaglia, N., & De Vincenzi, M. (2023). *Desarrollo de competencias conducido por rúbricas*.

Novak, J. D. (2010). *Learning, creating, and using knowledge: Concept maps as facilitative tools in schools and corporations*. Routledge. https://www.taylorfrancis.com/books/mono/10.4324/9780203862001/learning-creating-using-knowledge-joseph-novak

Open Robotic. (2025). *Gazebo [Simulador robótico]*. https://gazebosim.org/home

Ortony, A., Clore, G. L., & Collins, A. (2022). *The cognitive structure of emotions*. https://www.cambridge.org/core/books/cognitive-structure-of-emotions/33FBA9FA0A8A86143DD86D84088F289B

Ostanin, M., Yagfarov, R., Devitt, D., Akhmetzyanov, A., & Klimchik, A. (2021). Multi robots interactive control using mixed reality. *International Journal of Production Research*, *59*(23), 7126–7138. https://doi.org/10.1080/00207543.2020.1834640

Ouyang, F., & Xu, W. (2024). The effects of educational robotics in STEM education: a multilevel meta-analysis. *International Journal of STEM Education*, *11*(1), 7. https://doi.org/10.1186/s40594-024-00469-4

Panasiuk, O., Akimova, L., Kuznietsova, O., & Panasiuk, I. (2021). Virtual laboratories for engineering education. *2021 11th International Conference on Advanced Computer Information Technologies (ACIT*, 637–641. https://doi.org/10.1109/ACIT52158.2021.9548567

Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books. worrydream.com

PIAGET, J. (1979). *Psicología y Epistemología* (1a edición (1970). Buenos Aires, Emecé.

Piaggio, M., Sgorbissa, A., & Zaccaria, R. (2000). A programming environment for real-time control of distributed multiple robotic systems. *Advanced Robotics*, *14*(1), 75–86. https://doi.org/10.1163/156855300741456

Radanliev, P., Roure, D., Nicolescu, R., Huth, M., & Santos, O. (2022). Digital twins: artificial intelligence and the IoT cyber-physical systems in Industry 4.0. *International Journal of Intelligent Robotics and Applications*, *6*(1), 171–185. https://doi.org/10.1007/s41315-021-00180-5

Refaat, D., Elawa, M. T., & Hafez, A. M. (2025). Design and Development of a Low-Cost Educational Robot: A Scalable and Affordable Software Learning Platform. In Tanta University (Ed.), *Journal of Engineering Research (JER)*. https://digitalcommons.aaru.edu.jo/cgi/viewcontent.cgi?article=1907&context=erjeng

Resnick, M. (2013). *Learn to code, code to learn. EdSurge*. https://www.edsurge.com/news/2013-05-08-learn-to-code-code-to-learn

Rodriguez, S. L. C., & Castro, D. M. (2023). Flexible, scalable, and robust architecture for industrial automation applications with real-time requirements. *2023 IEEE 6th Colombian Conference on Automatic Control (CCAC*, 1–6. https://ieeexplore.ieee.org/abstract/document/10333842

Ryan, R. M., & Deci, E. L. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, *55*(1), 68–78. https://doi.org/10.1037/0003-066X.55.1.68

Saldarriaga, G. E., Lasso, M. A. M., Sánchez, E. R. V, Badilla-Saxe, E., Quinn, M., Alzati, E. C., & Solórzano, C. M. V. (2020). *Constructores de conocimiento: Papert y su visión* (Vol. 1). Corporación Universitaria para el desarrollo de Internet. https://books.google.com.ar/books?hl=es&lr=&id=nZrpDwAAQBAJ&oi=fnd&pg=PA51&dq=Se

ymour+Papert+(1980,+1991)+y+basado+en+el+constructivismo+de+Jean+Piaget+(1970)&ots=jt_7lw4hvk&sig=8r3XCwL2L94DTEQpQJMtUyOHHnw&redir_esc=y#v=onepage&q&f=false

Schäfer, A., Reis, G., & Stricker, D. (2022). A survey on synchronous augmented, virtual, andmixed reality remote collaboration systems. *ACM Computing Surveys*, *55*(6), 1–27. https://doi.org/10.1145/3533376

Schleich, B., Anwer, N., Mathieu, L., & Wartzack, S. (2017). *Shaping the digital twin for design and production engineering* (Vol. 66, Issue 1, p. 01513846). Manufacturing Technology. https://doi.org/https://hal.science/hal-01513846

Selim, A., Ali, I., Saracevic, M., & Ristevski, B. (2024). Application of the digital twin model in higher education. *Multimedia Tools and Applications*, 1–18. https://doi.org/10.1007/s11042-024-20014-3

Sharma, A., Kosasih, E., Zhang, J., Brintrup, A., & Calinescu, A. (2022). Digital twins: State of the art theory and practice, challenges, and open research questions. *Journal of Industrial Information Integration*, *30*, 100383. https://doi.org/10.1016/j.jii.2022.100383

Siemens. (2020). *Apollo 13: The first digital twin*. Siemens Simcenter Blog. https://blogs.sw.siemens.com/simcenter/apollo-13-the-first-digital-twin/?utm_source=chatgpt.com

Singh, M., Fuenmayor, E., Hinchy, E. P., Qiao, Y., Murray, N., & Devine, D. (2021). Digital twin: Origin to future. *Applied System Innovation*, *4*(2), 36. https://doi.org/https://www.mdpi.com/2571-5577/4/2/36

Singh, M., Kapukotuwa, J., Gouveia, E. L. S., Fuenmayor, E., Qiao, Y., Murray, N., & Devine, D. (2025). Comparative Study of Digital Twin Developed in Unity and Gazebo. *Electronics*, *14*(2), 276. https://doi.org/10.3390/electronics14020276

Stein, G., Jean, D., Kittani, S., Deweese, M., & Lédeczi, Á. (2025). A Novice-Friendly and Accessible Networked Educational Robotics Simulation Platform. *Education Sciences*, *15*(2), 198. https://doi.org/10.3390/educsci15020198

Sun, D., Looi, C. K., Li, Y., Zhu, C., & Cheng, M. (2024). Block-based versus text-based programming: A comparison of learners' programming behaviors, computational thinking skills and attitudes toward programming. *Educational Technology Research and Development*, *72*(2), 1067–1089. https://doi.org/10.1007/s11423-023-10328-8

Swarat, S., Ortony, A., & Revelle, W. (2012). Activity matters: Understanding student interest in school science. *Journal of Research in Science Teaching*, *49*(4), 515–537. https://doi.org/10.1002/tea.21010

Systémes, D. (2023). *Virtual Twin as a Service: How We Got Here*. https://www.3ds.com/consulting-services-value-engagement/virtual-twin-service/history-and-evolution?utm_source=chatgpt.com

Szczurek, K. A., Prades, R. M., Matheson, E., Rodriguez-Nogueira, J., & Castro, M. (2023). Accessible Educational Resources for Teaching and Learning Robotics. *Robotics*, *10*(1), 38. https://doi.org/https://www.mdpi.com/2218-6581/10/1/38

Tao, F., Sui, F., Liu, A., Qi, Q., Zhang, M., Song, B., & Nee, A. Y. (2019). Digital twin-driven product design framework. *International Journal of Production Research*, *57*(12), 3935–3953. https://books.google.com.ar/books?hl=es&lr=&id=PvKGDwAAQBAJ&oi=fnd&pg=PP1&dq=(Tao+et+al.,+2019)+twin+digital&ots=hmYpOHLghk&sig=gdYC4w5tQ7fiEjz4SBcWmhqLPgs&redir_esc=y#v=onepage&q=(Tao%20et%20al.%2C%202019)%20twin%20digital&f=false

Tarng, W., Wu, Y. J., Ye, L. Y., Tang, C. W., Lu, Y. C., Wang, T. L., & Li, C. L. (2024). Application of virtual reality in developing the digital twin for an integrated robot learning system. *Electronics*, *13*(14), 2848. https://doi.org/https://www.mdpi.com/2079-9292/13/14/2848

Thomas, J. W. (2000). *A review of research on project-based learning*. https://tecfa.unige.ch/proj/eteach-net/Thomas_researchreview_PBL.pdf

Tiryaki, A., & Adigüzel, S. (2021). The effect of STEM-based robotic applications on the creativity and attitude of students. *Journal of Science Learning*, *4*(3), 288–297. https://doi.org/https://eric.ed.gov/?id=EJ1307654

Tobón, S. T. (2006). *Competencias, calidad y educación superior*. https://books.google.com/books?hl=es&lr=&id=jW7G7qRhry4C&oi=fnd&pg=PA7&dq=Tob%C3%B3n,+S.+(2006).+Modelo+de+competencias+para+el+siglo+XXI+&ots=ivL-ZGL073&sig=Yv14BoczeFoMEltaKVbo8D7l7ZA

Unity Technologies. (2025). *Unity Real-Time Development Platform [Motor de desarrollo multiplataforma]*. https://unity.com/

Valls Pou, A., Canaleta, X., & Fonseca, D. (2022). Computational thinking and educational robotics integrated into project-based learning. *Sensors*, *22*(10), 3746. https://doi.org/10.3390/s22103746

Zhang, Y., Fang, L., Deng, H., Qi, Z., & Liang, H. (2023). Recent advances and future perspectives of digital twins. *2023 IEEE International Conference on Control, Electronics and Computer Technology (ICCECT*, 1563–1566. https://doi.org/10.1109/ICCECT57938.2023.10140652

Zhong, B., Zheng, J., & Zhan, Z. (2023). An exploration of combining virtual and physical robots in robotics education. *Interactive Learning Environments*, *31*(1), 370–382. https://doi.org/10.1080/10494820.2020.1786409

Zhou, J., Leng, J., & Yan, D. (2022). Digital Twin: Origin to Future. *Encyclopedia*, *2*(3), 932–947. https://encyclopedia.pub/entry/11253?utm_source=chatgpt.com

Zhou, J., Zhang, S., & Gu, M. (2022). Revisiting digital twins: Origins, fundamentals, and practices. *Frontiers of Engineering Management*, *9*(4), 668–676. https://doi.org/10.1007/s42524-022-0216-2

Anexos

Anexo A – Arquitectura del simulador ESTER

A.1. Motor gráfico y lenguaje base

- Motor de simulación: Unity 3D, versión 2022.3.x LTS
- Lenguaje de desarrollo principal: C#
- Motor de física: NVIDIA PhysX (nativo de Unity)
- Plataforma de distribución: WebGL / Standalone (.exe)
- Sistema operativo compatible: Windows 10+ / Linux (experimental)

A.2. Componentes del entorno virtual

- Modelos 3D personalizados: robots, obstáculos, mapas de prueba.
- Robots simulados: hasta 30 instancias individuales con mapeo por ID.
- Sensores virtuales:
  - Sensor ultrasónico (distancia frontal).
  - Sensor infrarrojo (detección de línea).
  - Lidar simulado (modo experimental).
  - Giroscopio virtual.

A.3. Comunicación API – Backend – Cliente

- Arquitectura general

  Alumno (cliente)

  ↕

Web/API Socket ↔ Servidor central ↔ Motor Unity ↔ Visualización

↕

Robot físico (TWIN ROBOT)

Tecnologías involucradas

Componente	Tecnología
Backend API	Python + Flask
Comunicación WebSocket	Socket.IO
Servidor en la nube	AWS / HyperV
Autenticación usuario	ID por alumno
Autenticación robot	automática dinámica clave

A.4. Estructura del backend de control

- API REST para autenticación, envío de comandos y consulta de logs.
- WebSocket bidireccional para control en tiempo real.
- Gestor de sesiones: asocia ID de alumno a instancia de robot simulado y robot físico si está disponible.

A.5. Mecanismo de sincronización con el robot físico

- Comunicación vía sockets TCP, en red local o externa.
- Formato de mensaje estandarizado:

```
json
{
 "robot_id": "R23",
 "comando": "avanzar",
 "velocidad": 80,
 "timestamp": "2025”03”15T13:20:05.004Z"
}
```

- Priorización del dato físico: si el robot real envía un estado nuevo, el entorno lo refleja y sobreescribe el estado virtual.
- Failover automático: si el robot se desconecta, la simulación continúa sin errores.

A.6. Requerimientos mínimos y recomendados

Recurso	Mínimo	Recomendado
CPU	Intel i3	Intel i5/i7
RAM	4 GB	8 GB+
GPU	Integrada (HD 4000)	Dedicada (GTX 1050 o superior)
Sistema operativo	Windows 10 / Linux	Windows 10 64 bits
Conectividad	Wi”Fi / LAN	LAN estable
Navegador (modo WebGL)	Chrome / Edge	Chrome actualizado

A.7. Escalabilidad y uso en el aula

- Robots simulados dinámicos, se crean con cada alumno, testados hasta 30 robots simulados en simultáneo.
- Cada alumno se conecta con su ID único, lo que permite:
    - Visualización separada del entorno.
    - Logs personalizados por alumno.
    - Evaluación individual de desempeño.
    - Modo local LAN (sin internet).
    - Modo remoto en la nube (docentes distribuidos).

A.8. Herramientas auxiliares

- ATOM+: entorno de programación visual basado en TurboWarp.
- 6 escenarios fijos
- 1 escenario para gemelos digitales

Anexo B - Programación visual con ATOM+

B.1. Descripción general de ATOM+

ATOM+ es una interfaz de programación visual desarrollada específicamente para el simulador ESTER, basada en TurboWarp, una extensión de Scratch. Se diseñó con el objetivo de facilitar el acceso a la robótica y al pensamiento computacional, especialmente en contextos no informáticos o de formación docente.

B.2. Interfaz gráfica

La interfaz de ATOM+ contiene:

- Área de bloques personalizados (comandos de robótica, sensores, movimientos).
- Bloques nativos de scratch para programación.
- Consola de eventos para visualizar respuestas del simulador.
- Conexión por API (sin necesidad de instalar bibliotecas o paquetes externos).
- Paleta de bloques organizada por categorías: movimiento, sensores, control, operadores, funciones.

Link Atom+ módulos implementados sobre turbowarp: [Atom+]

Captura de pantalla 1:

Vista general de ATOM+ con bloques de navegación y sensores ultrasónicos (figura 24).

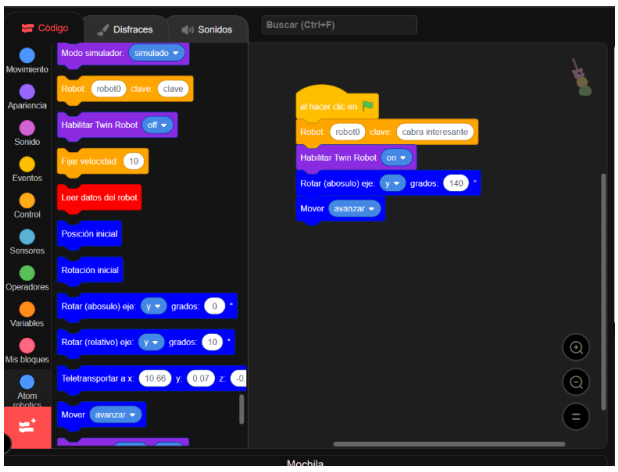


Figura 24. *Pantalla principal de ATOM*

Captura de pantalla 2:

Ejecución del robot en simulador desde bloques programados por el alumno (figura 25).

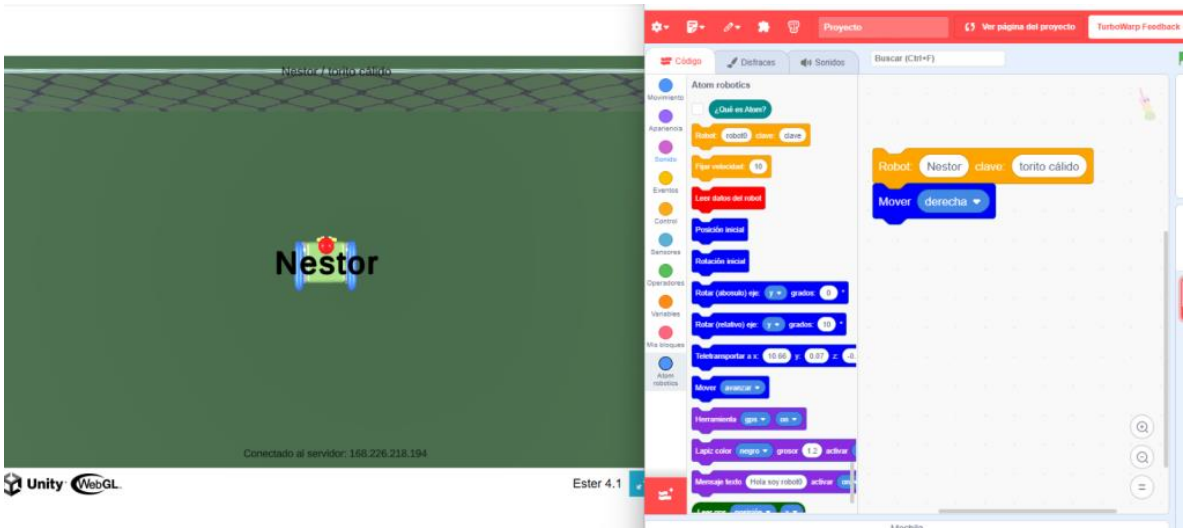


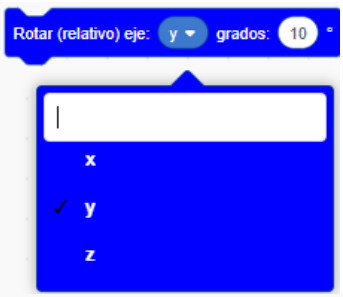
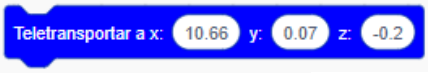
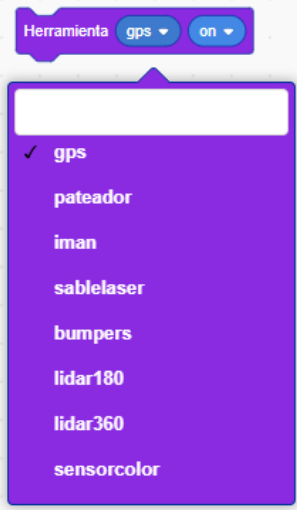
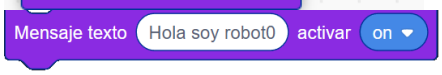
Figura 25. *Programa en ATOM+ controlando robot.*

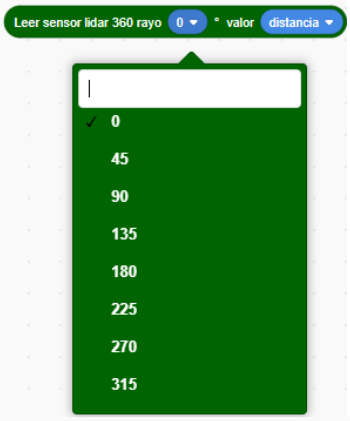
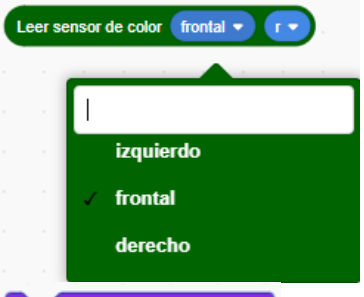
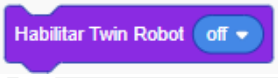
B.3. Conjunto de bloques personalizados

Los bloques específicos que se desarrollaron para el entorno ESTER incluyen (Tabla 20):

Tabla 20: Listado de módulos ATOM+.

Nro.	Módulo	Descripción
1	Atom robotics ¿Qué es Atom?	Muestra un texto con los datos generales del conjunto de módulos ATOM+.
2	Robot: robot0 clave: clave	Permite elegir el robot y la clave.
3	Fijar velocidad: 10	Fija la velocidad en vueltas por minuto de las ruedas
4	Leer datos del robot	Actualiza los datos del robot.
5	Posición inicial	Posiciona el robot en su posición inicial que obtuvo al comenzar la simulación.
6	Rotación inicial	Rota el robot a su rotación original que obtuvo al comenzar la simulación.
7	Rotar (abosulo) eje: y grados: 0 ° x ✓ y z	Rotación absoluta en X, Y, Z.

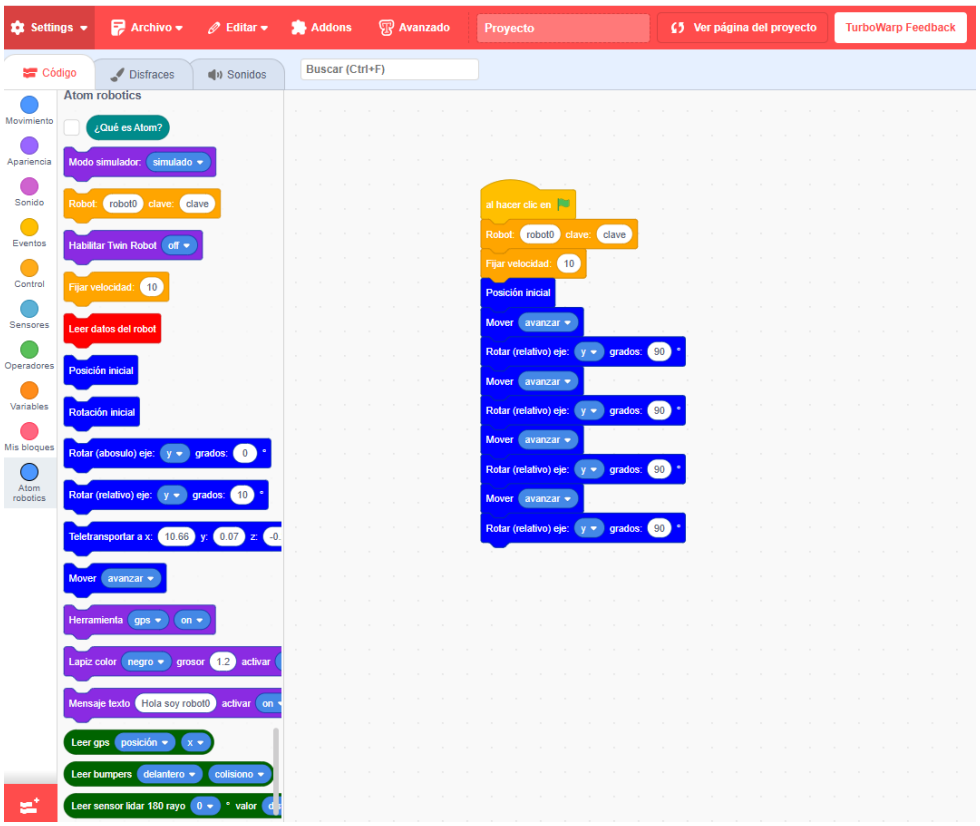
8		Rotación relativa en X, Y, Z.
9		Posiciona el robot en X, Y, Z.
10		Mueve el robot en el simulador o gemelo físico.
11		Activar herramientas como GPS, imán, etc.
12		Permite dibujar con color en el suelo simulado.
13		Envía un mensaje de texto sobre el robot.

14		Lee posición o rotación GPS en X, Y, Z.
15		Detecta colisión en bumpers (sensores de choque) del robot.
16		Lectura del sensor Lidar 180.
17		Lectura del sensor Lidar 360.
18		Lectura RGB u objeto detectado.
19		Activa el modo híbrido y vincula al robot real.
20		Mapea el entorno físico en el simulador.

21		Configura modo: real, híbrido, simulado.
22		Define posición actual como origen.

B.4. Ejemplos de programas creados con ATOM+

Ejemplo 1: autómata secuencial



B.5. Integración con el simulador ESTER

- Cada bloque ejecuta una instrucción que se envía a una API que lo envía al simulador vía WebSocket.
- Los resultados se ven reflejados inmediatamente en el robot virtual.
- En modo híbrido, los mismos bloques se reflejan también en el TWIN ROBOT físico, si está conectado.
- Las respuestas del robot (posición, colisiones, sensores) pueden retroalimentar visualmente al alumno.

B.6. Valor pedagógico

- Reducen la barrea de entrada para estudiantes sin experiencia en código.
- Permiten interactuar desde programación visual a código textual con cualquier lenguaje que implemente conexión a API.
- Favorecen el trabajo colaborativo y por proyectos.
- Han sido validadas en carreras técnicas y en maestrías con docentes de nivel inicial, primario y secundario (ver Capítulo 4).

Anexo C – Protocolos de conexión con TWIN ROBOT

C.1. Objetivo del protocolo

El protocolo de conexión entre el simulador ESTER y el robot físico TWIN ROBOT fue diseñado para garantizar una comunicación bidireccional en tiempo real, segura y eficiente, que permita sincronizar los estados del robot virtual y físico bajo un modelo de gemelo digital híbrido.

Este anexo documenta el formato de mensajes, el flujo de datos, los mecanismos de sincronización y los criterios de priorización entre los entornos simulado y real.

C.2. Arquitectura de conexión

- Tipo de comunicación: socket TCP personalizado (cliente”servidor).
- Ubicación del servidor: embebido en el simulador Unity (lado cliente).
- Puerto estándar utilizado: 8804 (configurable).
- Tasa de envío: 10–30 Hz (dependiendo de la capacidad de red).
- Buffer de respaldo: 100 ms.
- Diagrama de flujo de conexión:

  [Alumno]

  ↓

  [Simulador Unity] ←→ [Robot Físico: ESP32 / Arduino con WiFi]

  ↑

  [API WebSocket / TurboWarp / Python Script]

C.3. Formato del mensaje enviado al robot

Los comandos se envían en formato JSON, codificados como string plano vía socket TCP:

json

```
{
 "user_id": "alumno",
 "object_id": "robot0",
 "object_key": "conejo veloz",
 "comando": "mover",
 "tipo": "avanzar",
 "velocidad": 80,
 "rotación": “{X:0,y:0,z:0}”,
```

"posición": “{X:0,y:0,z:0}”,

"timestamp": "2025”05”01T14:03:45.120Z"

}

Campo	Tipo	Descripción
user_id	String	Nombre del alumno
object_id	String	Nombre del robot, único
object_key	String	Clave del robot, temporal por cada sesión
Comando	String	Comandos que pueden enviarse al robot por ejemplo: mover
Tipo	String	Tipo de comando, por ejemplo: avanzar

C.4. Mensaje recibido desde el robot físico

El robot físico envía periódicamente su estado sensado:

json

```
{
 "robot_id": "R001",
 "estado": {
  "posicion": [120, 56],
  "angulo": 90,
  "distancia": 27,
  "infrarrojo": 1
 },
 "conexion": "OK",
 "timestamp": "2025”05”01T14:03:45.320Z"
}
```

C.5. Flujo de sincronización y lógica de prioridad

- El alumno envía un comando → se ejecuta en simulador y robot físico.
- El robot físico responde con su estado real.
- Si el dato físico está disponible y es válido, sobrescribe el estado simulado.
- Si el robot se desconecta o no responde:
  - Se activa modo “solo simulador” automáticamente.
  - El entorno se mantiene operativo sin errores para el alumno.

- Cuando el robot vuelve a conectarse:
    - El sistema reinicia la sincronización desde el último estado válido.

C.6. Comandos disponibles (comunicación básica)

Comando	Descripción
avanzar	Mueve el robot hacia adelante
retroceder	Mueve el robot hacia atrás
rotar_izq	Gira el robot hacia la izquierda
rotar_derecha	Gira el robot hacia la derecha
detener	Detiene todos los motores
leer_sensor_ultrasonido	Devuelve distancia e infrarrojo
Leer_giroscopo	Lee los datos del giroscopio
Leer_encoder	Lee el numero relativo de vueltas del encoder de cada rueda (derecha:0,izquierda:0)

C.7. Configuración del robot físico (ESP32/Arduino)

El robot físico debe contener:

- **Bibliotecas de WiFi y Socket TCP** para la comunicación con el servidor.
- **Funciones para interpretar comandos en formato JSON,** recibiendo instrucciones de movimiento y lectura de sensores.
- **Sensores conectados a pines predefinidos** (por ejemplo: D2 = sensor IR, D3 = ultrasonido, D4 = giróscopo).
- **Control de motores mediante PWM** a través del motor shield correspondiente.
- **Buffer de seguridad** para almacenar temporalmente comandos no ejecutados y evitar pérdida de acciones en caso de latencia o reconexión.

C.8. Seguridad y control de errores

- Validación del formato JSON en cada mensaje recibido.
- Control de duplicados y “paquetes fantasmas”.
- Modo silencioso en caso de error de red.
- Consola del simulador con log de cada evento (ver Anexo A).

Anexo D – TWIN ROBOTS

Este anexo presenta las vistas gráficas, despiece y componentes principales del robot físico TWIN ROBOT, desarrollado como gemelo del simulador ESTER. Las imágenes documentan su diseño modular, impresión 3D y ensamble, y complementan la descripción técnica incluida en el Capítulo 5.



Figura 26. *Desafío reciclaje con robots reales*

F.1. Introducción

Este anexo presenta la versión de **TWIN ROBOT**, este modelo tiene una arquitectura modular y para un robot con tracción diferencial.

F.2. Versión 1 – Detalle de impresión y armado

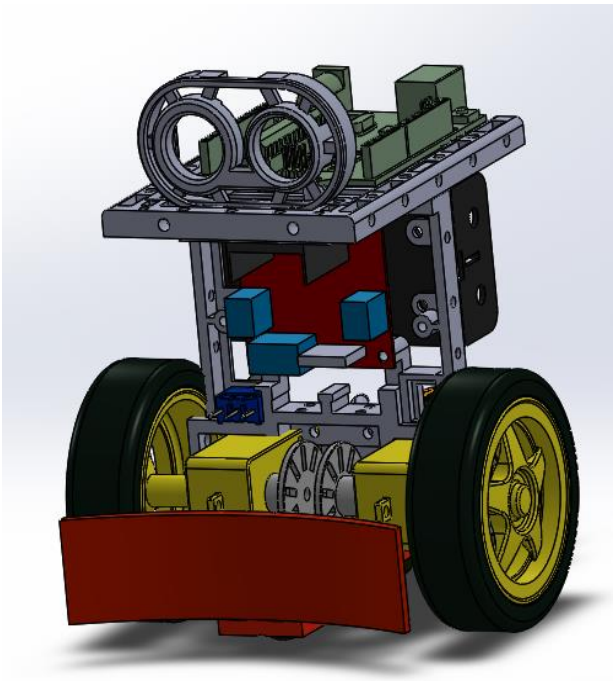


Figura 27. *Robot V1 diseño CAD*

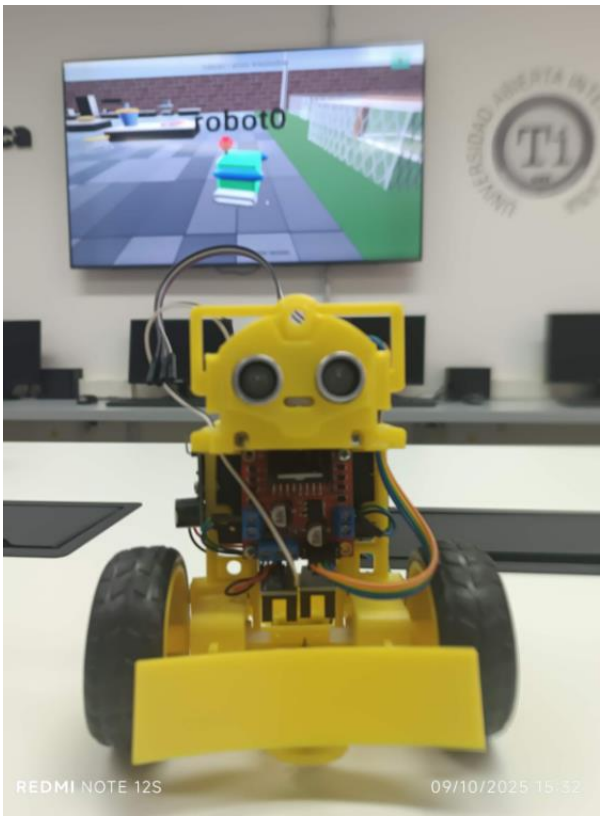
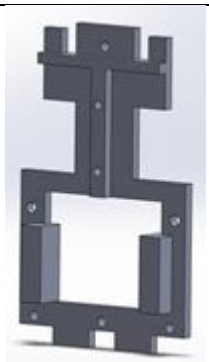
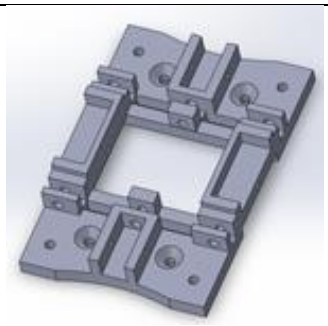
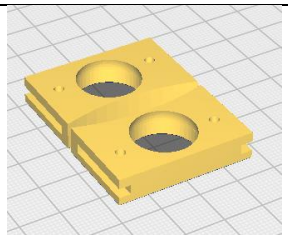
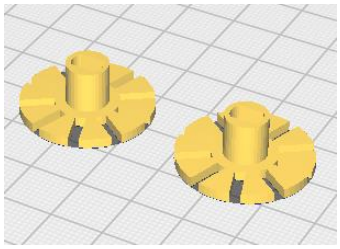
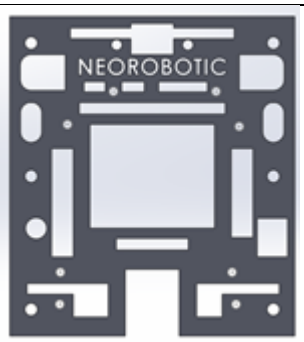
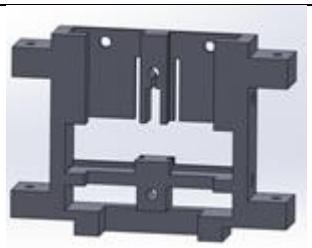


Figura 28. *Robot V1 simulado real*

Pieza 3D	Imagen	Tiempo (hh:mm:ss)
lateral plaquetas		1:13:00
enfriamiento y extracción		0:10:00
lateral portapilas		1:12:00
enfriamiento y extracción		0:10:00
porta bolitas		2:54:00
enfriamiento y extracción		0:10:00
tapa bolitas		0:48:00
enfriamiento y extracción		0:10:00

rueda dentada		0:30:00
enfriamiento y extracción		0:10:00
porta placa		2:34:00
enfriamiento y extracción		0:10:00
unión central		0:51:00
enfriamiento y extracción		0:10:00
Tiempo total de impresión		**11:12:00**
Armado y post”impresión		1:40:00
Total general en horas		**12:52:00**

3. Modulo de robot de futbol de dos ruedas

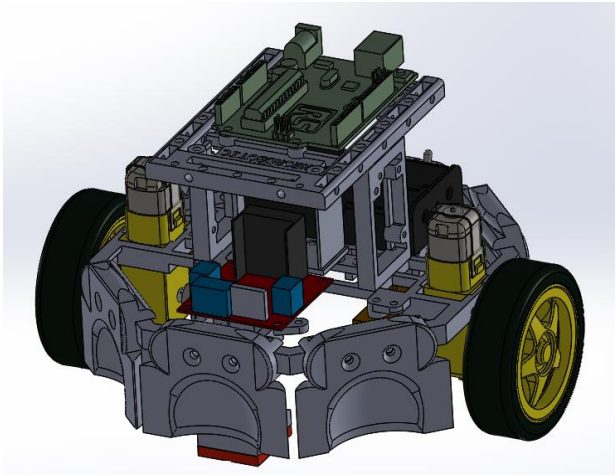
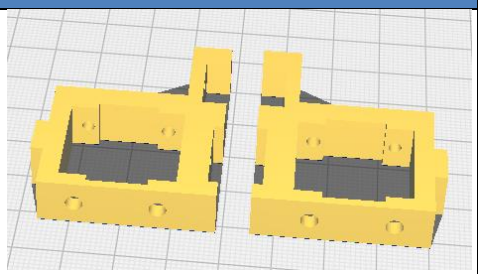
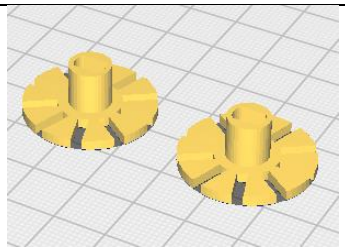
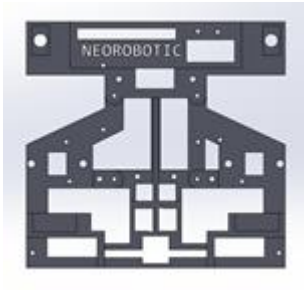
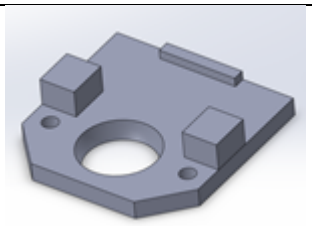
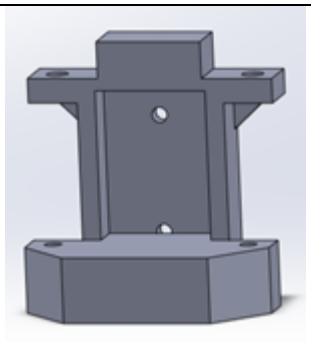


Figura 29. *Robot V1b diseño CAD*

Este módulo permite al robot adaptarse a las competencias de futbol de robot de la robocup de acuerdo a las reglas internacionales.

Pieza 3D	Imagen	Tiempo (hh:mm:ss)
porta motor		1:21:00
ruedas dentadas		0:30:00
enfriamiento y extracción		0:10:00

porta placa v2		3:31:00
enfriamiento y extracción		0:10:00
porta ultrasonido		0:35:00
enfriamiento y extracción		0:10:00
tapa bolita		0:34:00
enfriamiento y extracción		0:10:00
columna		1:12:00
enfriamiento y extracción		0:10:00
Tiempo total de impresión		**8:23:00**
Armado y post”impresión		0:40:00
Total general en horas		**9:03:00**

F.4. Módulo de robot omnidireccional de fútbol robot

Este módulo permite que el robot de se desplace en todas direcciones, permitiendo representar en el simulador dos tipos de robot: 1 Twin robot tracción omnidireccional y 2 Twin robot con tracción diferencial .

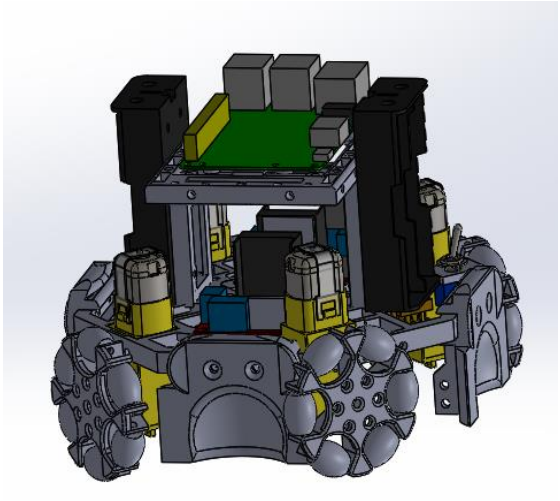


Figura 30. *Robot V2 diseño CAD*

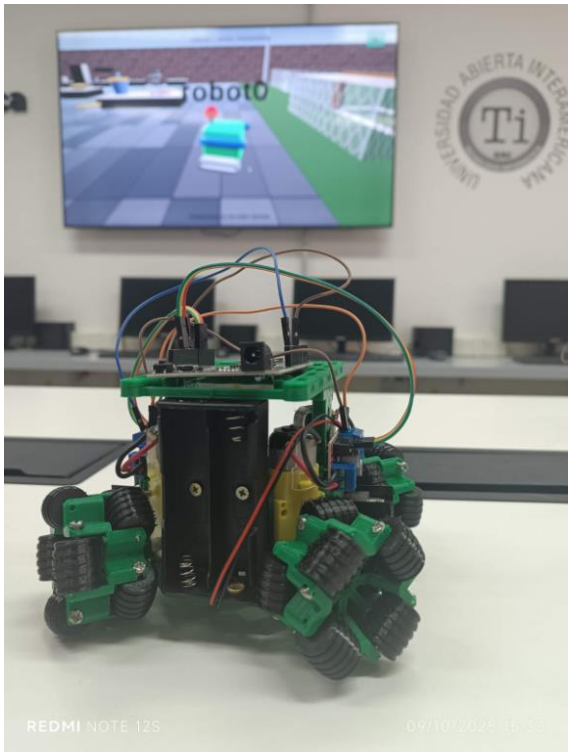


Figura 31.*Robot V2 real*

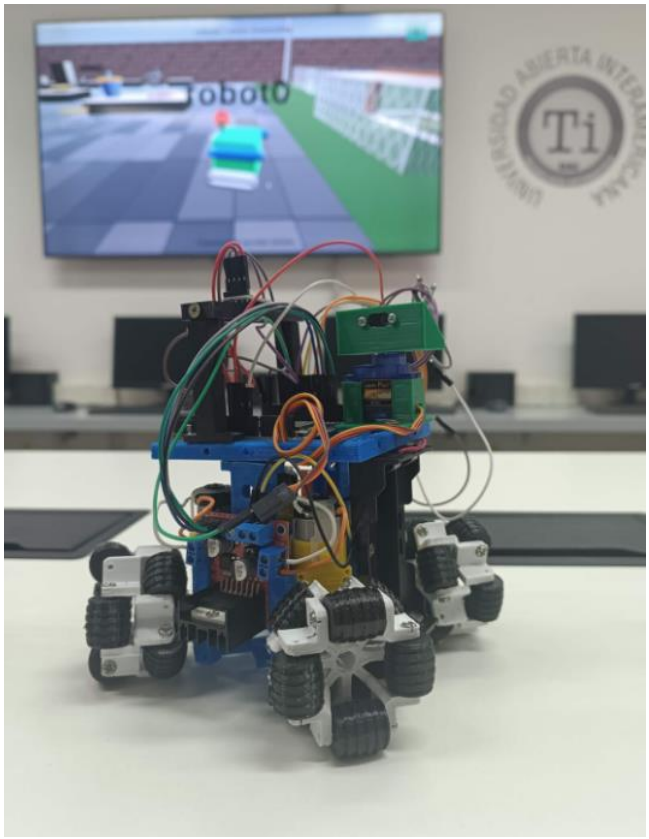
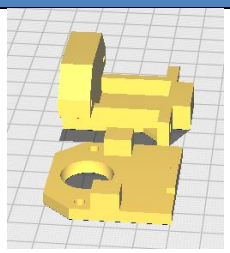
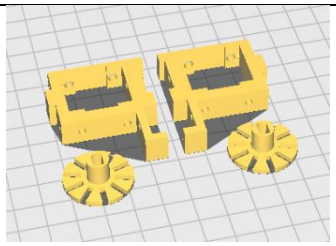
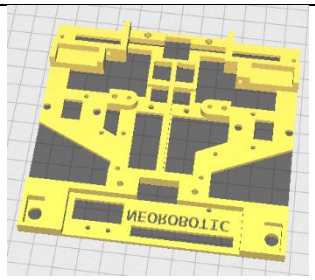
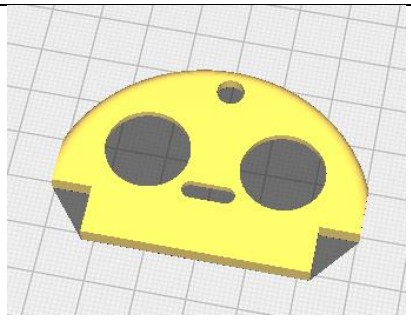
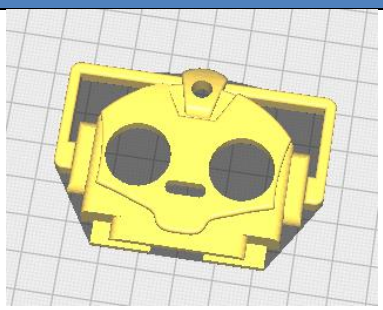


Figura 32. *Robot V2 real*

Pieza 3D	Imagen	Tiempo (hh:mm:ss)
Columna bolita y tapa		1:47:00
enfriamiento y extracción		0:10:00
2 porta motor y 2 ruedas dentadas		1:49:00
enfriamiento y extracción		0:10:00

Porta placa		3:31:00
enfriamiento y extracción		0:10:00
Porta ultrasonido simple		0:33:00
enfriamiento y extracción		0:10:00
Total de impresión		**8:20:00**
Armado y post”impresión		0:40:00
Total general V2		**9:00:00**

F.5. Versión 2 – otras impresiones adicionales

Pieza 3D	Imagen	Tiempo (hh:mm:ss)
Porta ultrasonido Ciberman		**1:57:00**

F.6. Consideraciones de mejora en V2

- **Reducción del tiempo total de fabricación**: de 14:35 a 9:04 hs.
- **Optimización del diseño de piezas**: menos piezas con mayor funcionalidad integrada.

- **Reducción del uso de filamento** (no detallado aquí, pero mensurable en futuras versiones).
- **Mejor ensamblaje**: armado más rápido, reducción de fricción mecánica.

F.7. Aplicabilidad educativa

La reducción de tiempos de impresión y ensamblado permite:

- Mayor escalabilidad en aulas con alta cantidad de alumnos.
- Uso más intensivo de los robots físicos en talleres de corta duración.
- Rápida reposición de piezas en caso de rotura o desgaste.
- Adaptabilidad a distintos tipos de impresoras 3D (FDM domésticas).

Anexo E – Evaluación educativa: encuestas y rúbricas aplicadas

Este anexo presenta los resultados completos de las encuestas realizadas a estudiantes de grado y posgrado que utilizaron el entorno híbrido del simulador **ESTER** junto con el robot físico **TWIN ROBOT**.

Las encuestas fueron anónimas y tuvieron como propósito recoger información sobre:

- Nivel de motivación y compromiso con las actividades propuestas.
- Utilidad percibida del simulador y del robot físico en la comprensión de algoritmos y estructuras de control.
- Valoración de la interfaz de programación visual **ATOM+**.
- Grado de transferencia de conocimientos entre el entorno simulado y la práctica física.
- Opinión general sobre la experiencia de aprendizaje híbrida.

Los resultados se organizaron en tablas y gráficos que incluyen:

- **Distribución porcentual de respuestas** en escalas Likert (Muy de acuerdo – De acuerdo – En desacuerdo – Muy en desacuerdo).
- **Comparación entre cohortes de grado y posgrado**.
- **Comentarios abiertos** que reflejan percepciones cualitativas del estudiantado.

Estos datos complementan el análisis presentado en los **Capítulos 4 y 7**, ofreciendo evidencia empírica que respalda la validez pedagógica del modelo híbrido implementado.

Tabla 21: *Encuesta de percepción estudiantil – Cloud Robotic (2023)*

Indicador	Excelente (5)	Muy bueno (4)	Bueno (3)	Aceptable (2)	Insuficiente (1)
1 ¿Qué tan fácil fue para ti entender y utilizar el simulador 3D de robótica?	13	5	1		
2 ¿Qué tan útil consideras que fue el simulador para aprender algoritmia y programación?	16	2		1	
3 ¿En qué medida crees que el simulador 3D de robótica te ayudó a comprender los conceptos de programación para robots colaborativos?	14	5			
4 ¿Cómo calificarías la calidad de las actividades y ejercicios propuestos en el simulador?	14	5			

5 ¿Qué tan motivado te sentiste al utilizar el simulador 3D de robótica para aprender programación y algoritmia aplicada a un entorno cloud?	14	3	1	1	
6 ¿En qué medida consideras que el simulador te ayudó a visualizar y comprender los resultados de tus programas de forma interactiva?	12	5	2		
7 ¿Recomendarías el uso de este simulador 3D de robótica a otros estudiantes que deseen aprender programación para robots colaborativos?	15	3	1		
8 ¿Consideras que el simulador 3D de robótica aceleró tu velocidad de aprendizaje en algoritmia y programación para robots colaborativos en comparación con otros métodos tradicionales de enseñanza?	11	8			
9 ¿En qué medida consideras que el simulador 3D de robótica abarcó en manera adecuada los contenidos de la materia de algoritmia y programación para robots colaborativos?	14	3	2		
10 ¿En qué medida consideras que el simulador 3D de robótica fomentó el trabajo colaborativo entre los alumnos en el aprendizaje de algoritmia y programación para robots colaborativos?	12	5	2		

Tabla 22: *Encuesta de percepción estudiantil – Maestría MTE (2024)*

Indicador	Excelente (5)	Muy bueno (4)	Bueno (3)	Aceptable (2)	Insuficiente (1)
1 ¿Qué tan fácil fue para ti entender y utilizar el simulador 3D de robótica?	7	6	8	5	1
2 ¿Qué tan útil consideras que fue el simulador para aprender algoritmia y programación?	10	7	5	2	3
3 ¿En qué medida crees que el simulador 3D de robótica te ayudó a comprender los conceptos de programación para robots colaborativos?	12	4	3	4	4
4 ¿Cómo calificarías la calidad de las actividades y ejercicios propuestos en el simulador?	11	8	3	2	3
5 ¿Qué tan motivado te sentiste al utilizar el simulador 3D de robótica para aprender programación y algoritmia aplicada a un entorno cloud?	11	7	4	2	3
6 ¿En qué medida consideras que el simulador te ayudó a visualizar y comprender los resultados de tus programas de forma interactiva?	11	6	4	3	3
7 ¿Recomendarías el uso de este simulador 3D de robótica a otros estudiantes que deseen aprender programación para robots colaborativos?	14	6	2		6

8 ¿Consideras que el simulador 3D de robótica aceleró tu velocidad de aprendizaje en algoritmia y programación para robots colaborativos en comparación con otros métodos tradicionales de enseñanza?	14	4	3	3	3
9 ¿En qué medida consideras que el simulador 3D de robótica abarcó en manera adecuada los contenidos de la materia de algoritmia y programación para robots colaborativos?	12	7	3	2	3
10 ¿En qué medida consideras que el simulador 3D de robótica fomentó el trabajo colaborativo entre los alumnos en el aprendizaje de algoritmia y programación para robots colaborativos?	11	7	4	1	4

En este anexo se presentan los resultados complementarios derivados de su aplicación, incluyendo las encuestas de percepción estudiantil, que permitieron triangular la evidencia empírica obtenida mediante la rúbrica con la valoración directa del alumnado.

Anexo F – Pantallas del simulador

Link de simulador 4.2: [Unity WebGL Player | Ester 4.2]



Figura 33. *Login de alumno*

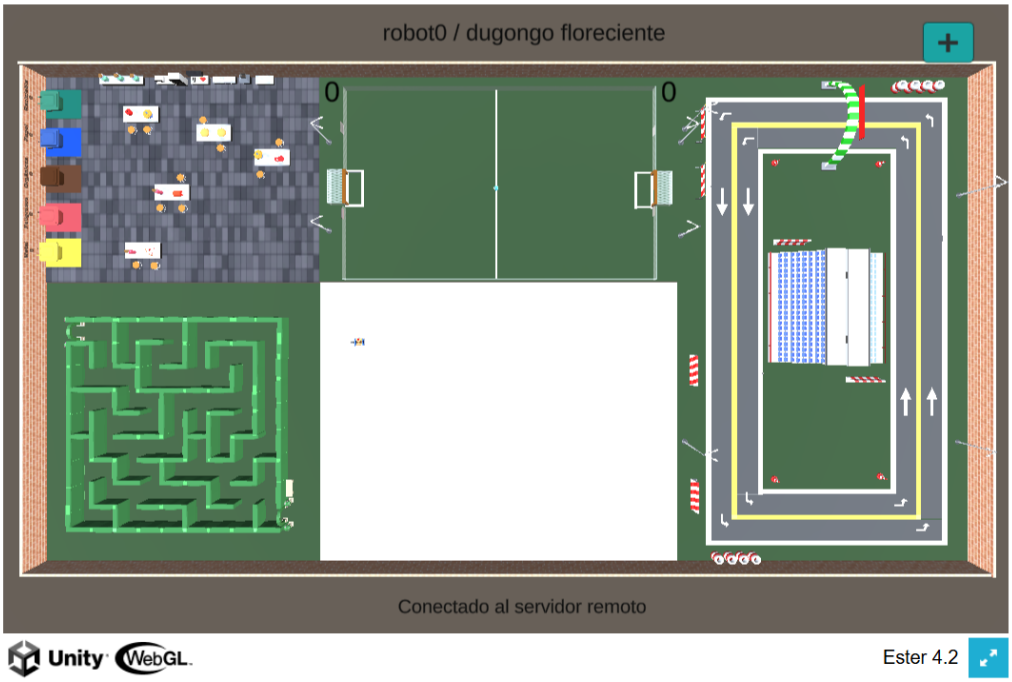


Figura 34. *Pantalla principal, vista cámara superior*

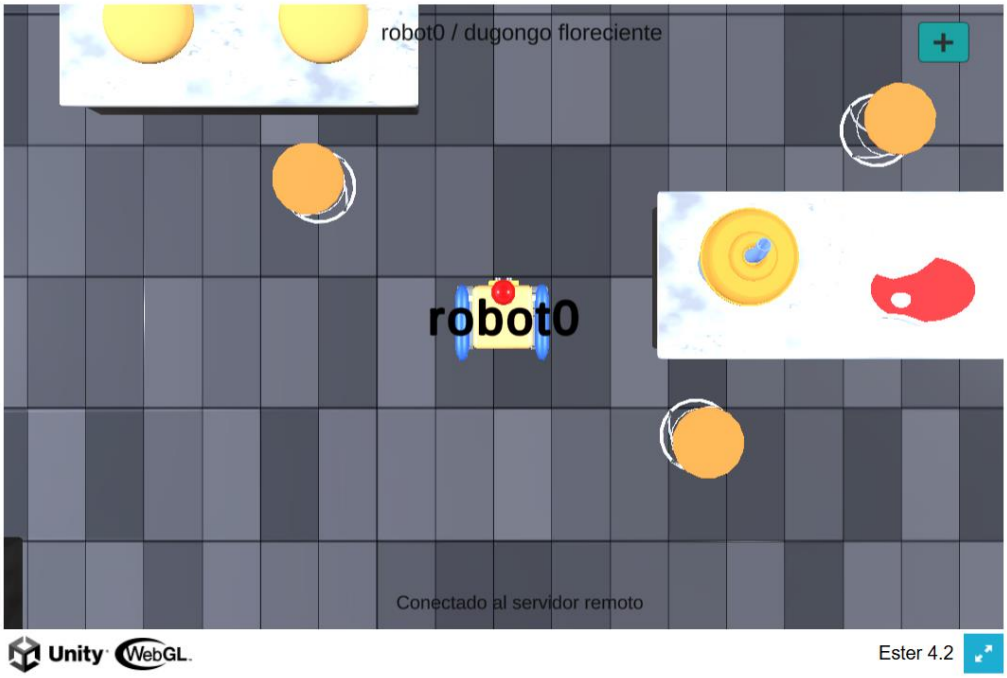


Figura 35. *Modo cámara "C" seguidora robot*

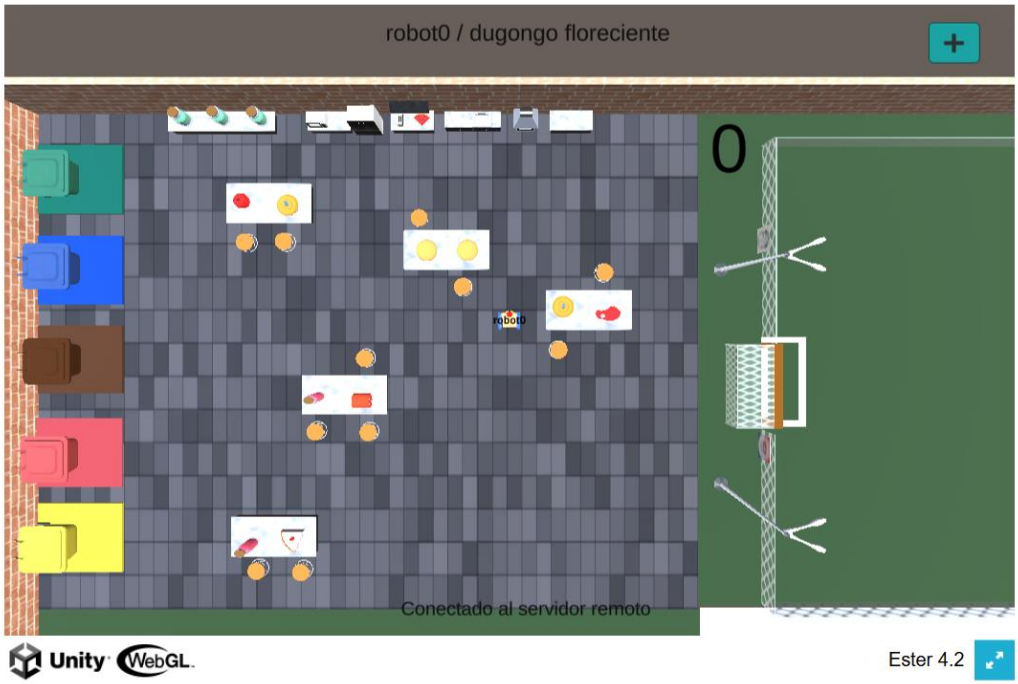


Figura 36, *Modo desafío reciclaje*

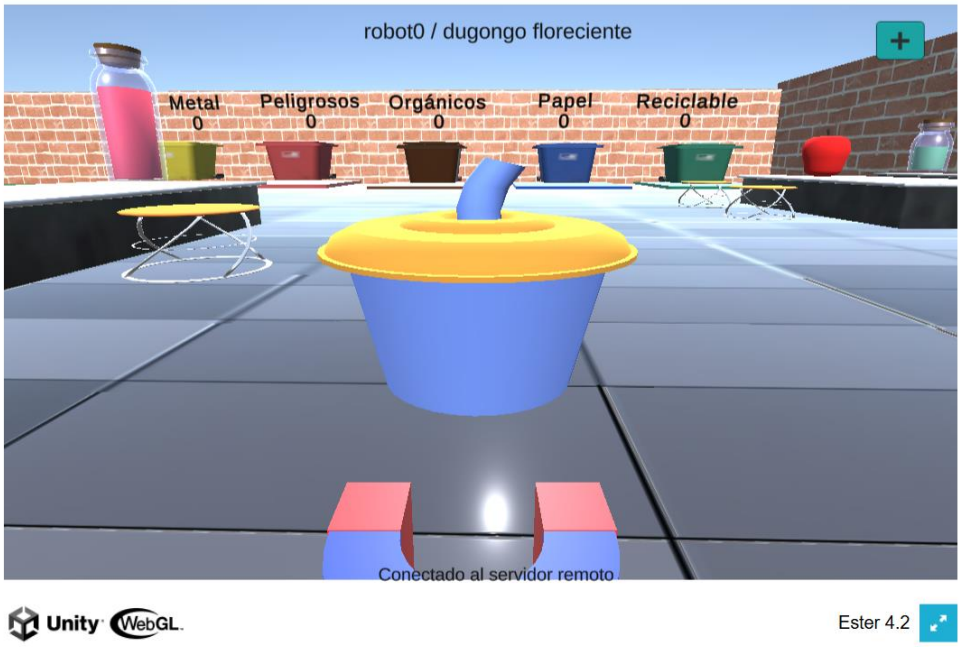


Figura 37. *Modo primera persona imán activado lleva el vaso de papel a contenedor de papel*



Figura 38. *Vista seguidor de robot con mensajes entre alumnos activado*



Figura 39. *Desafío de fútbol con pateador activado*

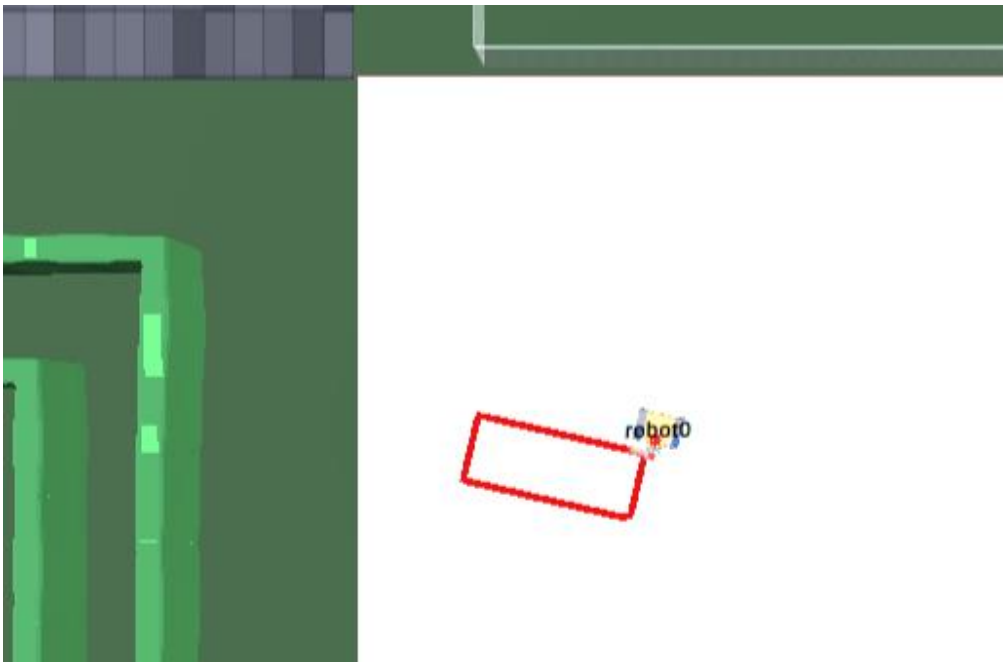


Figura 40. *Modo dibujar activado*

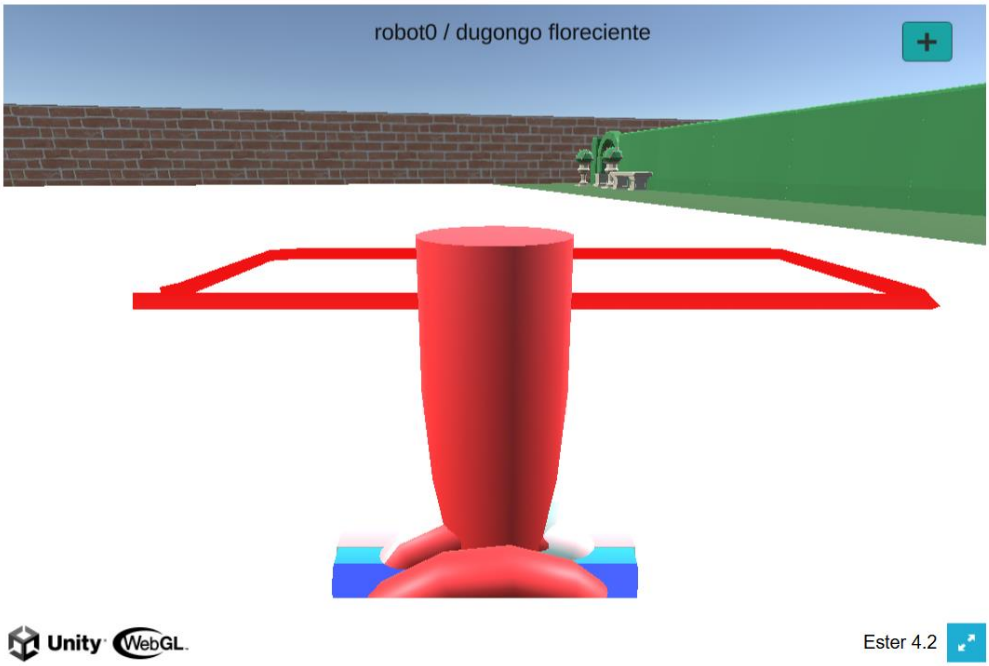


Figura 41. *Modo dibujar activado, cámara en primera persona*

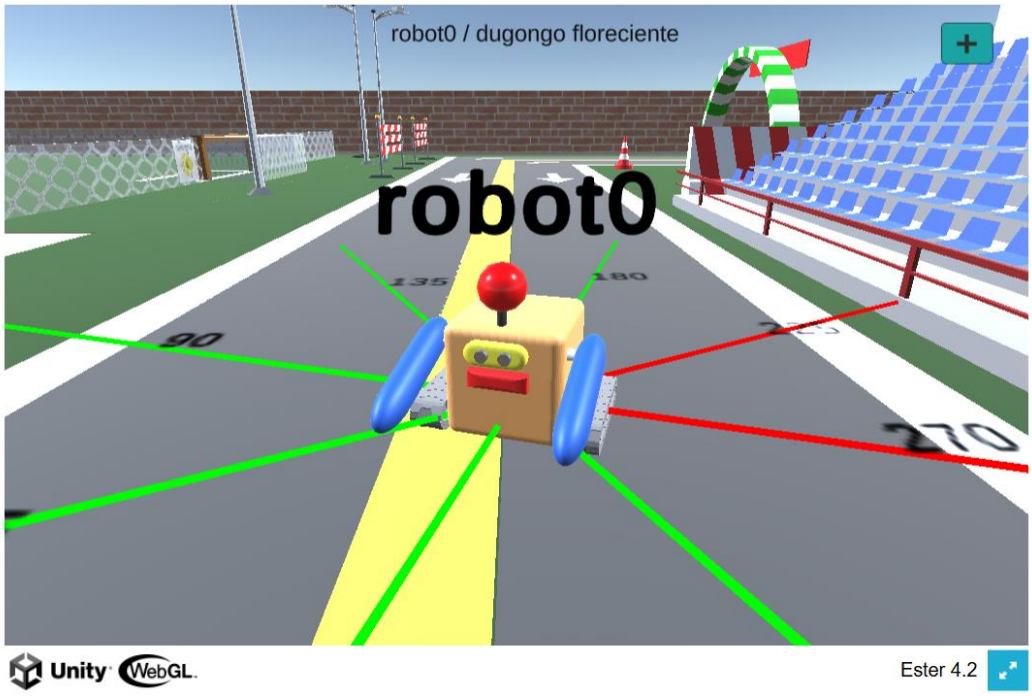


Figura 42. Lidar 360 activado en desafío carrera detectando colisión en 225° y 270°

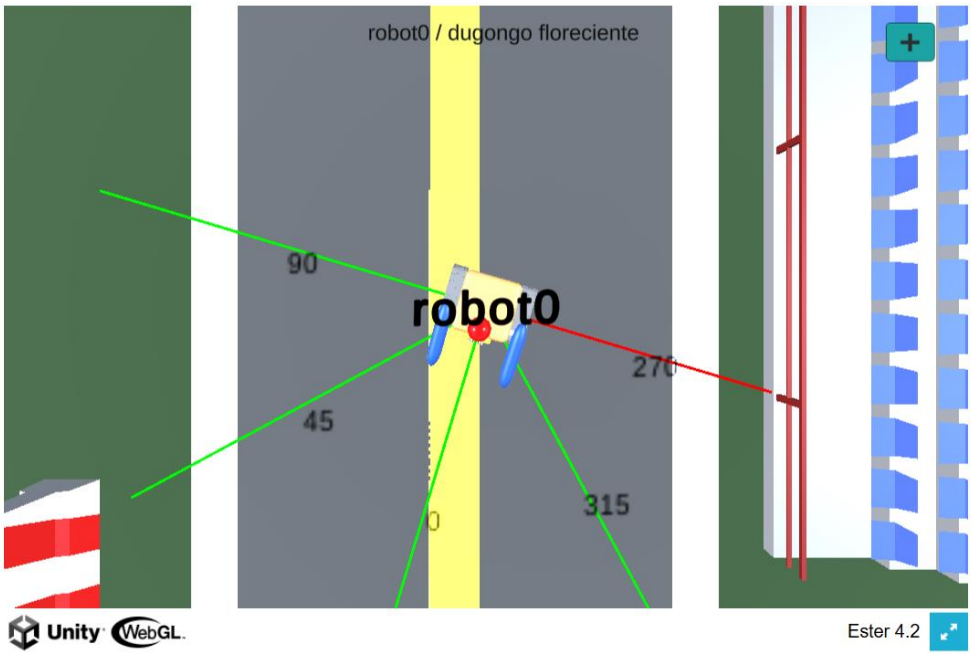


Figura 43. *Lidar 180° activado detectando*

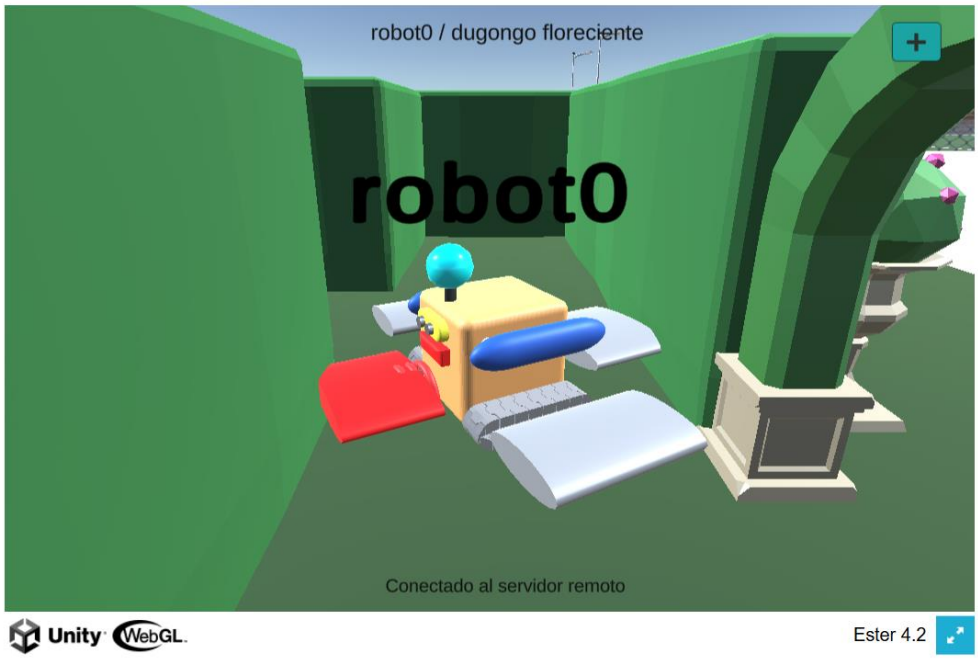


Figura 44. *Modo visión lateral con bumper activado detectando choque frontal en laberinto*

Desafío con control celular activado, en este modo manual aparece un circulo rojo en el robot, pues los ejercicios solo se hacen en forma autónoma, este modo es para que el alumno pueda racionalizar el algoritmo antes de programarlo.

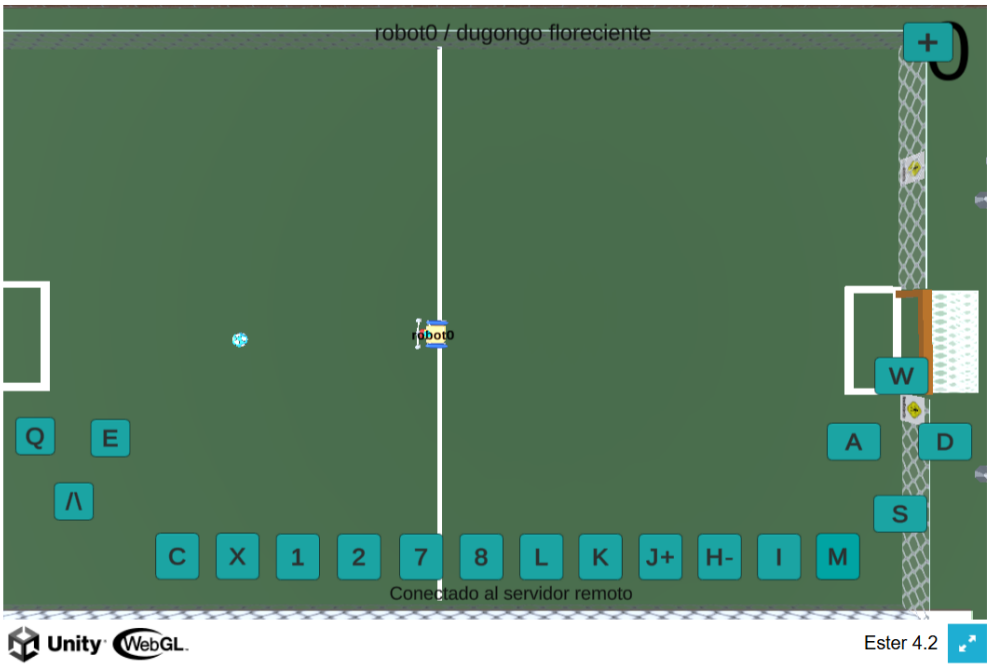


Figura 45. *Modo control manual por celular activado*