



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA
CARRERA DE COMPUTACIÓN**

**DESARROLLO DE UNA INTERFAZ ACCESIBLE PARA LA INTERACCIÓN ENTRE
OBJETOS VIRTUALES Y DISPOSITIVOS PORTÁTILES EN PERSONAS CON DIS-
CAPACIDAD VISUAL, MEDIANTE DETECCIÓN DE MOVIMIENTO DE MANOS
CON SENSOR LEAP MOTION**

Trabajo de titulación previo a la obtención del
título de Ingeniero en Ciencias de la Computación

AUTOR: DANIEL BRIAN YANZA CÁCERES

TUTOR: ING. PATRICIO FERNANDO URGILÉS ORTIZ, PhD.

Cuenca - Ecuador

2025

**CERTIFICADO DE RESPONSABILIDAD
Y AUTORÍA DEL TRABAJO DE TITULACIÓN**

Yo, Daniel Brian Yanza Cáceres con documento de identificación N° 1752773521 manifiesto que:

Soy el autor y responsable del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 21 de julio del 2025

Atentamente,

Daniel Brian Yanza Cáceres
1752773521

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Yo, Daniel Brian Yanza Cáceres con documento de identificación No. 1752773521, expreso mi voluntad y por medio del presente documento cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del Proyecto técnico: “Desarrollo de una interfaz accesible para la interacción entre objetos virtuales y dispositivos portátiles en personas con discapacidad visual, mediante detección de movimiento de manos con sensor Leap Motion”, el cual ha sido desarrollado para optar por el título de: Ingeniero en Ciencias de la Computación, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribo este documento en el momento que hago la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 21 de julio del 2025

Atentamente,

Daniel Brian Yanza Cáceres

1752773521

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Patricio Fernando Urgilés Ortiz con documento de identificación N° 0102137593, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el Proyecto técnico: DESARROLLO DE UNA INTERFAZ ACCESIBLE PARA LA INTERACCIÓN ENTRE OBJETOS VIRTUALES Y DISPOSITIVOS PORTÁTILES EN PERSONAS CON DISCAPACIDAD VISUAL, MEDIANTE DETECCIÓN DE MOVIMIENTO DE MANOS CON SENSOR LEAP MOTION, realizado por Daniel Brian Yanza Cáceres con documento de identificación N° 1752773521, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 21 de julio del 2025

Atentamente,

Ing. Patricio Fernando Urgilés Ortiz, PhD.

0102137593

Dedicatoria

Dedico este trabajo a todas las personas que me guiaron hasta este momento. En especial a mi familia, por brindarme su apoyo incondicional, tanto emocional como, en cierta medida, económico, permitiéndome continuar con mi formación.

A mi padre, quien confió en mí lo suficiente como para nunca dejar de apoyarme, incluso en momentos de dificultad académica.

A mi madre, quien en muchas ocasiones fue mi chofer personal y el desahogo de mis emociones y frustraciones frente a los desafíos de la vida educativa y social.

A mis hermanitos, quienes se mostraban intrigados por cada uno de mis inventos y me inspiraron a profundizar aún más en el conocimiento.

A mi hermana, quien me introdujo en círculos educativos que ampliaron mi visión. Con quien compartí madrugadas de estudio, y quien, con sinceridad, me enseñó la importancia de equilibrar mi vida académica, social y personal.

A mis profesores, por su paciencia a lo largo de todos los ciclos, por permitirme equivocarme una y otra vez para aprender, tal como ellos también lo hicieron alguna vez.

Y a mis amigos más cercanos, quienes me mantuvieron motivado a lo largo de la carrera, fomentando una competencia sana que me impulsó a dar siempre lo mejor de mí.

1. Resumen

El presente proyecto se desarrolló con el objetivo de ampliar las posibilidades de interacción y aprendizaje en entornos físicos y digitales para personas con discapacidad visual. Muchas de las soluciones tecnológicas empleadas en la actualidad que están dirigidas a este público, son diseñadas principalmente para usuarios con capacidades visuales plenas o con una pérdida mínima, lo que limita la accesibilidad y participación de quienes presentan una condición de ceguera total o severa. Frente a esta problemática, se propuso un sistema que emplea tecnologías de captura de movimiento y retroalimentación física como medio de interacción alternativo.

La solución se basa en el uso del sensor **Leap Motion**, que permite captar con precisión los movimientos de las manos del usuario. Esta información es procesada a través de la plataforma de desarrollo de videojuegos Unity, donde las coordenadas captadas se transforman en señales que luego son enviadas a un microcontrolador **ESP32 con modulo Wi-Fi**, el cual se encarga de generar estímulos físicos a través de motores. De este modo, el usuario puede interactuar con un entorno virtual y recibir retroalimentación táctil en tiempo real. La comunicación entre **Unity** y el ESP32 se realiza mediante conexión inalámbrica por Wi-Fi. Para asegurar una comunicación estable y continua, se implementó la asignación de direcciones IP estáticas basadas en las **direcciones MAC** de cada dispositivo, evitando así desconexiones por asignación dinámica incorrecta con los protocolos de **DHCP**. Este sistema busca ofrecer una alternativa intuitiva y accesible que combine percepción táctil con control gestual, permitiendo a personas con discapacidad visual desenvolverse mejor en entornos educativos o de entretenimiento digital. Además, su arquitectura modular y de bajo costo lo hace replicable y adaptable a distintas necesidades, no solo al público objetivo, pero ayudando a fomentar la noción del desarrollo de tecnologías inclusivas en el ámbito educativo y recreativo. *Palabras clave: Unity – ESP32 – Wi-Fi – DHCP – MAC address – Leap Motion – Discapacidad visual*

2. Abstract

This project was developed with the objective of expanding interaction and learning possibilities in both physical and digital environments for individuals with visual impairments. A system was proposed that employs motion capture technologies and physical feedback as an alternative interaction method. The solution is based on the use of the **Leap Motion** sensor, which accurately captures the user's hand movements. This data is processed using the **Unity** game development platform, where the captured coordinates are transformed into signals that are sent to an **ESP32 microcontroller with Wi-Fi** capabilities. The ESP32 then generates physical stimuli through motors, enabling the user to interact with a virtual environment and receive real-time tactile feedback. Communication between Unity and the ESP32 is established wirelessly via Wi-Fi, based on each device's MAC address, to avoid incorrect dynamic address

assignments through **DHCP** protocols.

This system aims to provide an intuitive and accessible alternative that combines tactile perception with gesture control, allowing visually impaired individuals to engage more effectively in educational or digital entertainment environments.

Índice

1. Resumen	1
2. Abstract	2
3. Introducción	6
4. Problema	6
4.1. Antecedentes	6
4.2. Importancia	8
4.3. Grupo Objetivo (Beneficiarios)	9
5. Objetivos	10
5.1. Objetivo General	10
5.2. Objetivos Específicos	10
6. Fundamentos Teóricos	11
6.1. Software de Acceso Libre	11
6.2. Captura de datos tridimensionales	12
6.3. Sistemas hápticos y auditivos en entornos virtuales	14
7. Marco Metodológico	17
7.1. Topología de Red del Sistema	17
7.2. Arquitectura en Capas	17
7.2.1. Capa de Datos	19
7.2.2. Capa de Procesamiento	19
7.2.3. Capa de Comunicación	23
7.3. Capa de Retroalimentación	24
7.3.1. Programación del ESP32	24
7.4. Análisis final sobre la implementación en capas	27
7.5. Resumen de la arquitectura del sistema desarrollado	28

8. Pruebas y Resultados	29
8.1. Investigación teórica	29
8.2. Configuración de librerías Ultraleap	29
8.3. Extracción de coordenadas	30
8.4. Funciones de interacción	30
8.5. Construcción de entornos virtuales	31
8.6. Verificación de métodos	31
8.7. Comunicación con hardware externo	32
8.8. Envío de datos	33
8.9. Interfaz para usuarios con discapacidad visual	33
8.10. Métodos de retroalimentación auditiva	34
8.11. Validación con usuarios	35
8.12. Documentación final	36
9. Conclusión final	37
10. Recomendaciones finales	38
11. Glosario	39
12. ANEXO	41
12.1. Imágenes de pruebas en SONVA	41
12.2. Plataforma de Código Fuente	41
12.3. Estado de desarrollo según actividades planteadas	42
12.4. Carta De Consentimiento Informado	43
Anexo E: Consentimiento Informado	43

3. Introducción

La interacción con entornos tridimensionales en el contexto de digital representa un desafío persistente, especialmente en la manipulación de objetos virtuales sin contacto físico. Aunque las tecnologías de captura de movimiento han avanzado, aún presentan limitaciones en tanto a la precisión y naturalidad de la detección de gestos manuales. Para las personas ciegas de nacimiento, comprender formas tridimensionales es aún más ambiguo en complejidad y entendimiento, ya que su percepción del espacio depende del tacto y la representación mental. Por ello, se requiere un sistema que, mediante visión por computadora y sensores, como Leap Motion, facilite una interacción intuitiva con entornos virtuales permitiendo la mejora de detección de las partes del cuerpo y la simulación del contacto.

4. Problema

4.1. Antecedentes

La necesidad de expandir el acceso a la educación es indiscutible, pero esta expansión debe incluir a toda la población, sin excepción. Las personas con discapacidad, particularmente aquellas con discapacidad visual, enfrentan barreras estructurales y sociales que dificultan su acceso equitativo a procesos educativos. A menudo se tiene la percepción errónea de que la educación de este grupo requiere esfuerzos desproporcionados o cuidados especiales, lo cual refuerza prácticas de exclusión. Como consecuencia, los sistemas educativos suelen marginar a esta comunidad, tanto en términos de inversión como de incorporación de tecnologías accesibles.

En Ecuador, según datos estadísticos oficiales, existen 56.644 personas con discapacidad visual, de las cuales 33.386 son hombres y 23.258 son mujeres (CONADIS, 2025). Esta cifra resalta la existencia prominente de ecuatorianos que están a la espera de tecnologías inclusivas para igualar el terreno educativo.

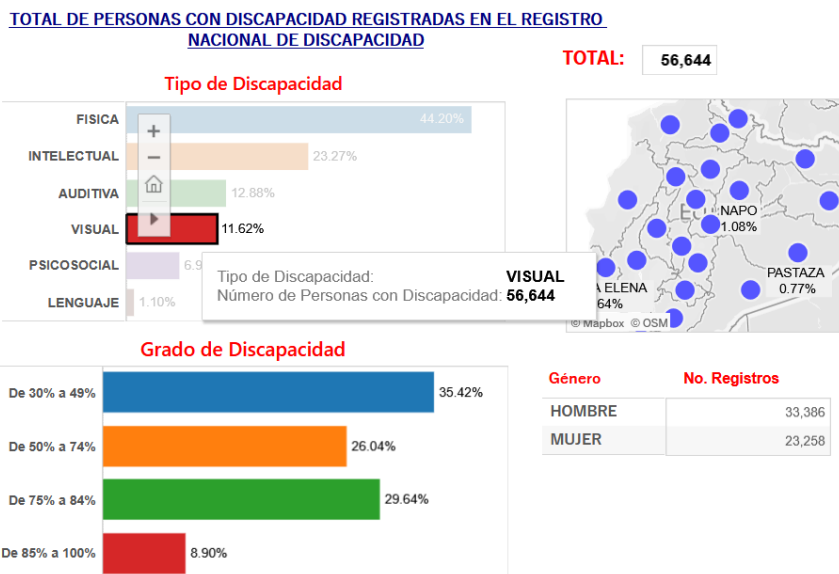


Figura 1: Estadísticas de discapacidad en Ecuador.

Fuente: CONADIS, 2025

Cuando se centra en discapacidades visuales, el aprendizaje basado en estímulos visuales y la interacción con objetos físicos se convierte en un desafío significativo. Informes médicos y educativos han documentado que incluso personas que recuperan parcialmente la visión mantienen conceptos erróneos sobre nociones espaciales y geográficas, lo que evidencia que la simple restauración visual no garantiza una comprensión adecuada del entorno. La interacción con entornos gráficos o digitales en estos casos es prácticamente nula, salvo que existan tecnologías especializadas de apoyo.

En 2011, investigaciones lideradas por el Dr. Pawan Sinha, a través del Project Prakash, mostraron que, al recuperar la vista, los participantes inicialmente no podían reconocer visualmente los objetos. Sin embargo, tras aproximadamente una semana de experiencia visual, lograban asociar visión y tacto de manera efectiva. (Christopher G. Thomas, 2025)

Esto respalda la hipótesis de que la relación entre visión y tacto es un proceso aprendido y no completamente innato. Sinha explica que, si bien la estructura general del sistema visual está determinada genéticamente, su conectividad más específica depende de la experiencia sensorial. Este hallazgo también revela que el cerebro humano conserva cierto grado de plasticidad incluso en la adolescencia, permitiendo adaptaciones funcionales significativas tras la exposición a nuevos estímulos visuales.

4.2. Importancia

Actualmente, existen tecnologías diseñadas para facilitar el aprendizaje y la interacción en personas ciegas o con baja visión, como:

- Pantallas braille dinámicas
- Dispositivos hápticos, como el *HaptiRead*, que permite leer texto mediante vibraciones en patrones braille sobre la piel.
- Lectores de pantalla (como *JAWS*)
- Plataformas de accesibilidad como **VoiceOver** de Apple o **TalkBack** en Android

Sin embargo, estos dispositivos suelen tener un costo elevado, lo que los hace inaccesibles para muchas instituciones educativas, especialmente aquellas con una alta diversidad de estudiantes o con recursos económicos limitados.

En este contexto, el presente proyecto se enfoca en contribuir a la resolución del Objetivo de Desarrollo Sostenible (ODS) 4: Educación de Calidad, promoviendo el acceso equitativo a oportunidades de aprendizaje inclusivo para personas con discapacidad visual. El uso de tecnologías alternativas de bajo costo ofrece una vía prometedora para construir herramientas accesibles que no dependan exclusivamente de la visión.



Figura 2: ODS 4. Educación de Calidad

Fuente: Naciones Unidas Ecuador, 2025

El uso de tecnologías alternativas, como los sensores de movimiento (*Leap Motion*) y micro-controladores programables (ESP32), permite el desarrollo de prototipos de bajo costo orientados a contribuir al **ODS 4: Educación de Calidad**. (Ultraleap, 2024b)

A partir de estas plataformas se pueden crear entornos interactivos con retroalimentación táctil que sustituyan parcialmente la dependencia de estímulos visuales, abriendo nuevas oportunidades para una educación verdaderamente inclusiva.

Con respaldo investigativo del **Project Prakash**, se evidencia la importancia de diseñar tecnologías accesibles que no dependan únicamente del canal visual, sino que integren otros sentidos —especialmente el tacto— para apoyar el aprendizaje en estudiantes con discapacidad visual severa. (Christopher G. Thomas, 2025)

Aunque estas tecnologías ya existen, en Ecuador su implementación es aún muy limitada y, en muchas provincias, prácticamente nula. Por ello, este proyecto busca promover e inspirar la adopción de prácticas inclusivas a través del desarrollo de un sistema accesible que sirva como prototipo funcional. La intención es demostrar que es posible construir soluciones educativas inclusivas de bajo costo, con el potencial de ser replicadas y escaladas en diferentes contextos educativos del país.

4.3. Grupo Objetivo (Beneficiarios)

El sistema desarrollado está dirigido a usuarios con discapacidad visual del Ecuador, sean de nacimiento o posterior desarrollo, que requieren una interacción avanzada con entornos 3D, donde la información tridimensional de movimiento y gestos toma papel protagónico para el desarrollo de sistemas.

A través de ejercicios guiados, podrán familiarizarse con formas geométricas básicas y posteriormente con objetos más complejos, facilitando su adaptación al entorno físico y mejorando su independencia; Estudios de neuroimagen han demostrado que, en personas con ceguera adquirida, la estimulación sensorial activa —como la lectura en Braille o la práctica musical— puede prevenir la inactividad funcional de regiones cerebrales previamente dedicadas a la visión. (Sadato et al., 1996) evidenciaron que la corteza visual se activa durante la lectura en Braille, lo cual respalda la necesidad de intervenciones tempranas para evitar una reorganización cerebral pasiva o limitada.

Para expandir el alcance más allá de los grupos directamente vinculados a la investigación en

discapacidades visuales, los potenciales usuarios de esta propuesta incluyen a:

- Desarrolladores de aplicaciones de realidad virtual
- Investigadores en interacción humano-computadora
- Diseñadores de experiencias inmersivas
- Profesionales en simulación digital

Asimismo, el sistema tendrá un impacto significativo en el ámbito educativo de personas con discapacidad visual, ya sea congénita o adquirida, al permitirles explorar y reconocer objetos tridimensionales de forma virtual. Esto promueve una experiencia de aprendizaje más inclusiva, multisensorial y accesible.

5. Objetivos

5.1. Objetivo General

Desarrollar una Interfaz Accesible para la Interacción entre Objetos Virtuales y Dispositivos Portátiles en Personas con Discapacidad Visual, Mediante Detección de Movimiento de Manos con sensor Leap Motion.

5.2. Objetivos Específicos

- Investigar técnicas de visión por computadora y captura de movimiento aplicadas a la interacción en entornos tridimensionales utilizando el sensor Leap Motion.
- Diseñar mecanismos que permitan ubicar y reconocer los movimientos en el espacio tridimensional, extrayendo sus coordenadas en los ejes x , y y z .
- Desarrollar una interfaz accesible que facilite la exploración y el reconocimiento de formas tridimensionales mediante Leap Motion y un dispositivo físico.
- Diseñar un proceso de validación para evaluar el desempeño del sistema a través de pruebas con usuarios, midiendo su eficacia en la interpretación de formas tridimensionales y

la interacción con entornos virtuales.

6. Fundamentos Teóricos

6.1. Software de Acceso Libre

El desarrollo de interfaces virtuales accesibles puede abordarse de manera efectiva mediante herramientas gratuitas y de código abierto. A continuación, se describen las principales plataformas utilizadas en este proyecto:

a) Unity

Unity es una herramienta de desarrollo multiplataforma ampliamente utilizada para crear entornos interactivos en 3D. Su capacidad de integrar datos tridimensionales y modelos 3D permite diseñar entornos virtuales adaptables a usuarios con discapacidad visual (Ultraleap, 2024a).

Gracias a su compatibilidad con dispositivos de entrada como Leap Motion, Unity permite crear interfaces naturales basadas en el movimiento de las manos. Además, ofrece mecanismos de comunicación con hardware externo, ya sea por puertos seriales (USB), vía Bluetooth o Wi-Fi, lo que abre la puerta de para implementar tecnologías variadas centradas en dar una respuesta física según la interacción virtual como: retroalimentación táctil o auditiva (Ultraleap, 2024b).

b) Blender y Maya

El diseño y creación de los modelos 3D se realiza principalmente con herramientas como Blender (software libre) y Maya (software comercial con licencia académica). Estas plataformas permiten generar geometrías personalizadas que luego se exportan e integran fácilmente en Unity.

Blender, en particular, destaca por su compatibilidad con formatos estándar como ‘.fbx’ y ‘.obj’, lo cual facilita el flujo de trabajo entre el modelado y la programación de interacción. Los modelos pueden ser actualizados y modificados según las necesidades del usuario, permitiendo una personalización progresiva de los elementos con los que se interactuará. (Studio, 2022)

c) Arduino IDE

El uso del entorno de desarrollo Arduino IDE permite programar microcontroladores como el ESP32 para gestionar la retroalimentación física (por ejemplo, motores de vibración, luces o sensores). Su compatibilidad con Unity a través de comunicación serial o inalámbrica (Bluetooth/Wi-Fi) facilita la integración del entorno virtual con estímulos reales.

Esta herramienta es esencial para implementar sistemas de salida multisensorial, especialmente para personas con discapacidad visual, ya que permite construir prototipos funcionales que complementan la experiencia virtual con respuestas físicas tangibles.

6.2. Captura de datos tridimensionales

La captura de información tridimensional es un componente fundamental en el diseño de interfaces para entornos virtuales accesibles. Este proceso se basa en el uso de sensores especializados capaces de registrar, con alta precisión y en tiempo real, la posición, orientación y movimiento de objetos o partes del cuerpo humano en un espacio 3D.



Figura 3: Dispositivo Leap Motion de Ultraleap
Fuente: Group, 2025

Uno de los dispositivos más relevantes en esta área es el **Leap Motion** de *Ultraleap*, un sensor óptico diseñado para el seguimiento preciso de manos y dedos. Este dispositivo opera dentro de un volumen definido en forma de semiesfera, con un radio aproximado de un metro, capturando el movimiento sin necesidad de contacto físico ni controladores adicionales.

Los datos recopilados son procesados mediante algoritmos propietarios, desarrollados por la

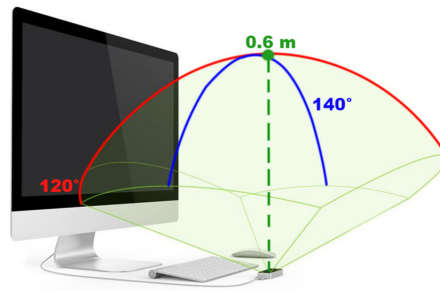


Figura 4: Zona de detección del Leap Motion

Fuente: Tripathy y Mohanty, 2023

empresa, que interpretan los gestos capturados y los traducen en acciones dentro del entorno virtual. Esto permite una interacción más natural, fluida e intuitiva, especialmente útil para personas con discapacidad visual, al no requerir interfaces físicas complejas.

Entre las principales ventajas del Leap Motion se destacan:

- Alta precisión en la detección de movimientos de manos y dedos.
- Baja latencia en la respuesta del sistema.
- Facilidad de uso, incluso para usuarios sin experiencia técnica.
- Compatibilidad con múltiples plataformas de desarrollo, como Unity y Unreal Engine.

Al permitir la manipulación, transformación y envío de datos capturados hacia sistemas de terceros, sea mediante la nube, conexión inalámbrica o una conexión física. Ambas plataformas ofrecen soporte para distintos lenguajes de programación y facilitan la integración con hardware externo.

Las especificaciones de cada Cámara Leap Motion son:

- **Leap Motion Controller (V1):** tiene un rango de detección efectivo de aproximadamente **25 cm a 60 cm**, con un máximo teórico de hasta **80 cm** desde el sensor.
- **Leap Motion Controller 2:** mejora la precisión y la estabilidad, extendiendo el rango efectivo hasta **110 cm**, con un campo de visión más amplio (casi $160^\circ \times 160^\circ$).

Esta tecnología tiene aplicaciones que van desde:

- Videojuegos y simulaciones interactivas

- Sistemas de control por gestos
- Aplicaciones accesibles para personas con discapacidad visual

6.3. Sistemas hápticos y auditivos en entornos virtuales

Los sistemas hápticos y auditivos permiten enriquecer la experiencia sensorial de los usuarios en entornos virtuales, especialmente en aplicaciones diseñadas para personas con discapacidad visual. El uso de vibraciones, sonidos espaciales y retroalimentación táctil permite proporcionar información contextual sin depender del sentido de la vista.



Figura 5: Dispositivos auditivos y vibratorios
Fuente: AliExpress, 2025

a) Retroalimentación auditiva

La retroalimentación auditiva va más allá de simples efectos de sonido; se enfoca en ofrecer descripciones auditivas detalladas del entorno. A través de mensajes verbales generados por síntesis de voz o sonidos específicos.

b) Módulo ESP32

El módulo ESP32 es un microcontrolador de bajo costo y alto rendimiento desarrollado por Espressif Systems. Cuenta con conectividad Wi-Fi y Bluetooth integrada, lo que lo convierte en una solución ideal para sistemas embebidos que requieren comunicación inalámbrica eficiente. Gracias a su arquitectura de doble núcleo, capacidades multitarea y múltiples interfaces de entrada/salida (GPIO, PWM, ADC, I2C, SPI, entre otros), el ESP32 es ampliamente utilizado

en aplicaciones de Internet de las Cosas (IoT), automatización y sistemas interactivos.

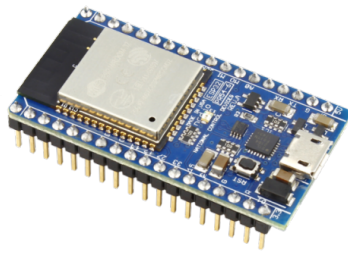


Figura 6: Modulo ESP32
Fuente: Electronics-Lab, 2025

En entornos virtuales accesibles, el ESP32 se emplea como intermediario entre el software de control y los actuadores físicos, como motores vibratorios o sistemas de audio. Su capacidad para recibir comandos inalámbricos en tiempo real permite sincronizar estímulos hápticos y auditivos de manera eficiente, sin depender de conexiones físicas que puedan limitar la movilidad del usuario. Además, su bajo consumo energético y tamaño compacto facilitan su integración en dispositivos portátiles o vestibles.

El uso del ESP32 en este tipo de sistemas habilita una retroalimentación inmediata y precisa, mejorando la inmersión del usuario y permitiendo una experiencia más intuitiva y natural dentro del entorno simulado.

Permitiendo una personalización sensorial con la posibilidad de ajustar los parámetros auditivos y hápticos según las preferencias y necesidades individuales del usuario. Esto no solo mejora la accesibilidad, sino que también favorece la adaptación y el confort durante el uso prolongado del sistema.

c) Retroalimentación háptica

Complementando el canal auditivo, los sistemas hápticos utilizan vibraciones de distinta frecuencia e intensidad para transmitir información adicional. Estos estímulos táctiles pueden emplearse para señalar colisiones, límites del entorno, o confirmar interacciones exitosas (por ejemplo, al tocar un objeto o completar una tarea).

La combinación de ambos sistemas permite crear experiencias multisensoriales coherentes, facilitando una navegación fluida y natural dentro del entorno virtual. (Unity Technologies, 2024a)

En este contexto, se toma como referencia el proyecto titulado **“Guante Multifunción con Entorno Virtual Inmersivo y Comunicación Háptica para Personas con Discapacidad Visual”**, desarrollado por un compañero dentro del mismo proceso de titulación. Dicho proyecto consiste en dos guantes, cada uno equipado con un microcontrolador **ESP32** y tres motores hápticos distribuidos en puntos clave de la mano.

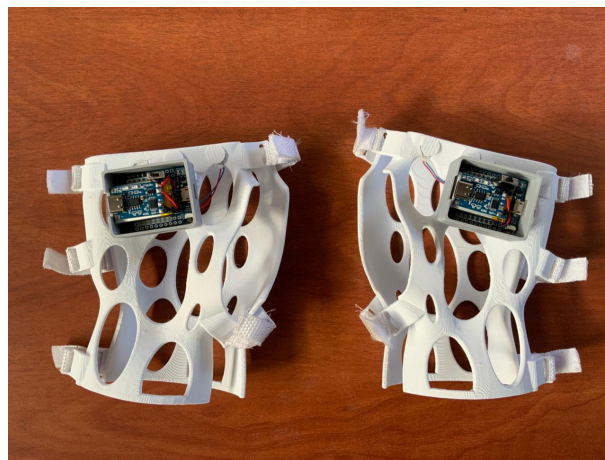


Figura 7: Guante Multifunción
Fuente: Ponce, 2025

El guante es capaz de recibir información desde un entorno virtual (Unity) —mediante comunicación inalámbrica— y activar los motores hápticos según el tipo de objeto o situación detectada. Este tipo de retroalimentación no visual resulta esencial para que las personas con discapacidad visual puedan interpretar y navegar entornos tridimensionales de manera intuitiva y autónoma.

Este desarrollo guarda relación directa con el presente proyecto, ya que ambos comparten el objetivo de mejorar la accesibilidad mediante tecnologías inmersivas y periféricos personalizados. (Ponce, 2025)

7. Marco Metodológico

La propuesta se fundamentó en una arquitectura modular organizada por capas, lo que permitió dividir de forma clara y estructurada tanto la metodología como el desarrollo del sistema. Para gestionar eficazmente el proceso, se empleó la metodología ágil Scrum, lo que permitió trabajar en ciclos incrementales que facilitaron el control, seguimiento y mejora continua del proyecto.

7.1. Topología de Red del Sistema

El sistema propuesto se implementa de forma completamente local, aprovechando el entorno de desarrollo Unity para gestionar los procesos de captura de datos, el procesamiento y la interacción con dispositivos físicos.

Esto no solo permitió un despliegue, eficaz y ligero. También reduce la latencia, en caso de estar desplegado en la nube, con principal foco en la respuesta de comunicación con los ESP32

Topología del sistema:

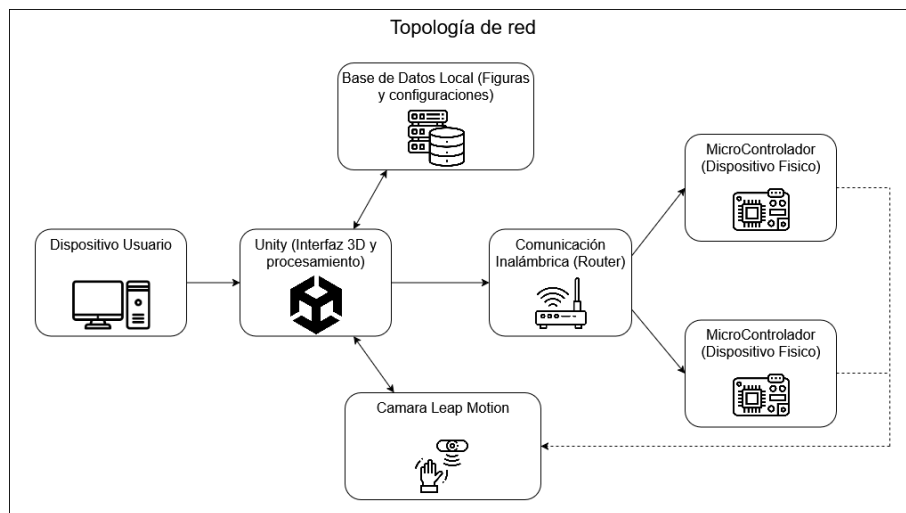


Figura 8: Topología de red local de la solución desarrollada

7.2. Arquitectura en Capas

El siguiente proyecto, implementó la estructura basada en capas, las cuales son:

- **Capa de Datos:** Donde la entrada de datos de la Cámara leap motion serán recopilados.

- **Capa de Procesamiento:** Donde decisiones y procesos se ejecutan de manera interna, en transformación, detección, carga y manipulación de los datos obtenidos.
- **Capa de comunicación:** Encargado de transportar los datos necesarios, en su formato requerido; a los dispositivos que lo requieran.
- **Capa de Retroalimentación Física:** Activación de dispositivos físicos/hápticos; en base de ello, realizar pruebas y consultas con usuarios del grupo objetivo.

Diagrama de desarrollo:

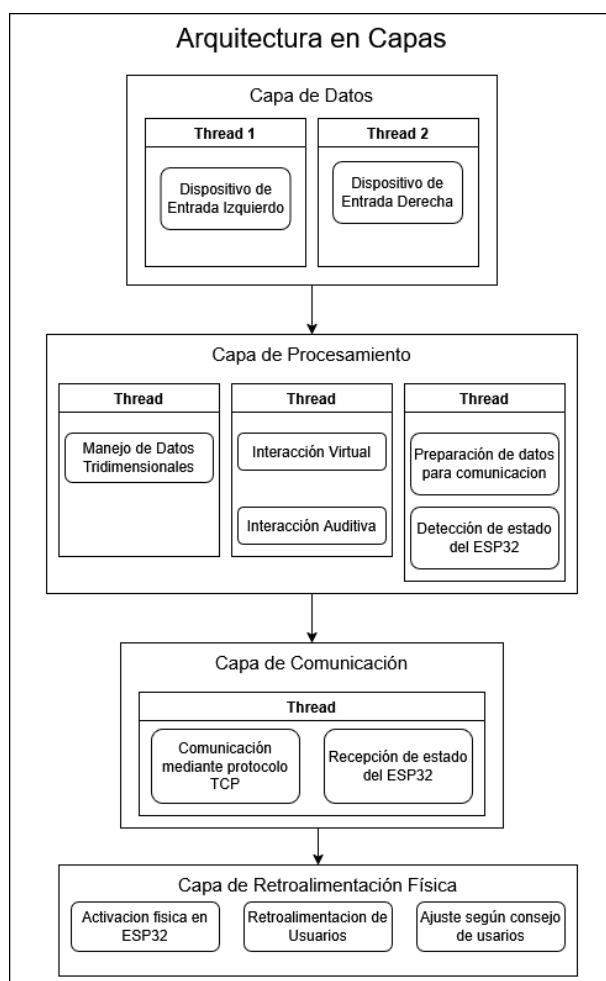


Figura 9: Arquitectura en capas del sistema propuesto

- **Dispositivos de Entrada:** Sensor Leap Motion, encargado de capturar el movimiento de las manos del usuario en tiempo real.
- **Manejo de datos Tridimensionales:** Análisis de las coordenadas tridimensionales obtenidas y su transformación para la decisión de acciones en la aplicación.

- **Interacción Virtual:** Entorno virtual encargado de interpretar las coordenadas recibidas y manipular los objetos virtuales correspondientes, además permite visualizar las acciones del usuario debido a sus librerías reconocidas por Leap Motion; finalizando con audios descriptivos de los objetos a interactuar.
- **Interacción Físico:** Generación de señales hacia el dispositivo físico ESP32 en forma de vibración, que reflejan la interacción dentro del entorno virtual.
- **Prueba y Ajuste:** Etapa de mejora continua que consideró la retroalimentación de usuarios no videntes para actualizar métodos de procesamiento, comunicación con el entorno virtual y retroalimentación física.

7.2.1. Capa de Datos

La **Capa de Datos** se encarga del registro de las coordenadas tridimensionales de las manos, captadas mediante el sensor *Leap Motion*. Este dispositivo se conecta directamente al computador a través de un puerto USB o USB-C. Para su correcto funcionamiento, es imprescindible la instalación del **driver oficial de Ultraleap**, encargado del procesamiento y envío de datos de seguimiento (tracking).

En el entorno de desarrollo *Unity*, también se requiere la instalación del **addon oficial de Ultraleap** y la activación del soporte para el *Leap Motion Desktop Tracking*. En este proyecto se utilizó la versión más reciente disponible hasta la fecha de implementación, **v2.3.1**.

Las dificultades en esta capa fueron mínimas, principalmente gracias a la documentación extensa y bien estructurada proporcionada por Ultraleap para la integración del sensor con Unity.

7.2.2. Capa de Procesamiento

Esta capa se divide en tres secciones fundamentales: procesamiento de coordenadas, procesamiento de figuras y procesamiento de configuración.

A) Procesamiento de Coordenadas

En esta sección, se prioriza la transformación y uso de los datos captados por el dispositivo Leap Motion. A través del driver y el addon oficial de Ultraleap, se logró la visualización e impresión en consola de las coordenadas de las manos, lo cual sirvió como validación inicial del sistema.

El mayor reto fue comprender el flujo de datos desde el dispositivo hacia Unity, ya que no se contaba con experiencia previa en este software. Los tutoriales disponibles fueron esenciales para entender la instalación de los addons, la manipulación de la interfaz y la interpretación de los datos.

Una complicación importante fue la representación visual de las cápsulas de los dedos: al principio eran invisibles, debido a que el componente 'LeapProvider' asignado a cada cápsula no tenía escala visible ni documentación clara sobre cómo manipularla. Además, cada mano tenía su propia escala, lo que dificultaba la calibración. Parte del problema se debió a que la documentación oficial se encontraba en transición entre Unity 5 y Unity 6, lo que generó inconsistencias en el material de referencia, con cambios en el nombre de funciones como la llamada de los dedos, cambio de Finger a finger.

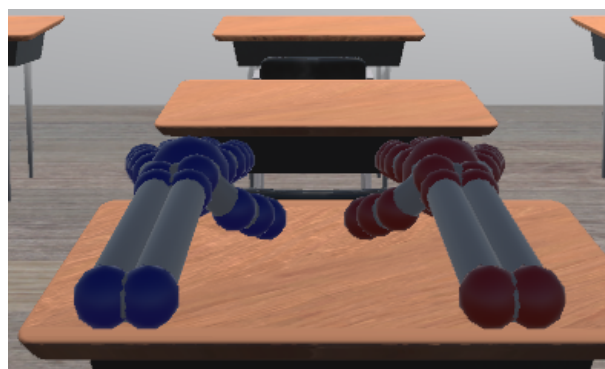


Figura 10: Error de escala de Capsule Hands

Luego de resolver estas complicaciones, surgió un nuevo desafío relacionado con el procesamiento de figuras: se registraban toques fantasma. Esto se debía a que el **collider** utilizado no era el más adecuado. Inicialmente se emplearon componentes como el **BoxCollider** o **SphereCollider**, cuya detección es de forma general y elíptica, lo cual no se ajustaba a la geometría real de modelos con extremos puntiagudos, como la pirámide o el cono.

La solución fue reemplazar estos por un `MeshCollider` (Unity Technologies, 2024b), que permite una detección precisa sobre la superficie real del objeto 3D. Tras este cambio, el sistema logró registrar correctamente el contacto de cada mano con las figuras, obteniendo resultados mucho más precisos y coherentes.

B) Procesamiento hacia los ESP32

Como etapa final del procesamiento, fue necesario enviar la información a los microcontroladores ESP32. Tal como se explicó anteriormente, se sustituyó el método de comunicación por puerto serial por uno inalámbrico mediante WiFi, utilizando un esquema de comunicación TCP simple para mantener la conexión entre Unity y los ESP32. Una vez establecida la conexión, se definió el formato de los datos a enviar como un arreglo de tres elementos:

$$[x, y, z]$$

[IdMano][IdDedo][Potencia de interacción]

Este formato no solo es ligero y fácil de interpretar en el ESP32, sino que también permite distinguir entre distintos niveles de interacción según la proximidad del dedo a la figura. En particular, si el dedo está dentro de la figura se envía una señal de vibración fuerte, mientras que en el borde se genera una vibración más suave. Esta estructura de datos también facilitó la inversión de los niveles de intensidad, mejorando la adaptabilidad del sistema según la petición de cada usuario en cada actividad.

C) Procesamiento de Figuras

Esta parte del sistema se encarga de la instancia y referencia de los modelos 3D utilizados durante la interacción. Las figuras fueron modeladas previamente en Blender y exportadas en formato **.fbx**. Al iniciar la escena, se ejecuta el método **LoadFigures()**, el cual realiza una carga dinámica de todos los prefabs contenidos en la carpeta **Resources/Models**.

Cada figura es identificada por su nombre archivo.

dificultad_nombreFigura
easy_cubo

A partir de esta nomenclatura, el sistema separa tanto el nivel de dificultad como el tipo de figura, y agrupa estos datos en un diccionario llamado **figureMap**, que clasifica las figuras por dificultad: **easy**, **medium**, **hard** y una categoría adicional llamada **All** que incluye todas las figuras.

Durante la carga, cada figura es asociada a un objeto **FigureData**, el cual contiene su nombre, ruta, escala y rotación inicial. Algunos modelos requieren ajustes específicos en la escala (por ejemplo, la pirámide usa una escala mayor) y orientación (como el rombo que necesita una rotación de 45° sobre el eje Z). Esto se configura dinámicamente durante el ciclo de carga, permitiendo que las figuras se instancien correctamente sin necesidad de realizar modificaciones manuales sobre cada modelo.

Finalmente, se proporciona el método

```
GetFiguresByDifficulty(string difficulty)
```

El cual se llama para recuperar la lista de figuras según el nivel de dificultad seleccionado por el usuario.

Este mecanismo también permite la reutilización de la misma escena de presentación, ya que las figuras se cargan según el contexto del menú anterior. Esto ahorra recursos y mejora la mantenibilidad del sistema.

D) Procesamiento de Configuración

En esta sección se definen y controlan los parámetros del sistema. Se incluyen los siguientes elementos clave:

- Configuración de los **joints** o articulaciones a rastrear (por ejemplo, solo las puntas de los dedos índice, medio y pulgar en el modo por defecto).
- Selección de los niveles de dificultad: **easy**, **normal**, **hard** y **all**, donde este último incluye todas las figuras.
- Carga de los audios correspondientes al nombre de cada figura, que se reproducen al momento de seleccionarlas.

E) Manejo de Hilos en Unity

Al tener que segmentar cada uno de estos procedimientos, se optó por el uso de hilos en Unity. Durante el desarrollo surgieron múltiples errores provocados por la sobrescritura del hilo principal del motor, lo que generaba comportamientos inestables y cierres inesperados. Para resolver este problema, se implementó un proceso iterativo de depuración basado en prueba y error, lo que permitió aislar la lógica intensiva en hilos secundarios sin comprometer el rendimiento ni la estabilidad del hilo principal de Unity.

7.2.3. Capa de Comunicación

Esta capa es responsable de la transmisión de datos hacia los microcontroladores ESP32, los cuales se distinguen internamente por sus direcciones IP estáticas: 192.168.10.48 para el dispositivo izquierdo y 192.168.10.49 para el derecho. La comunicación se realiza a través de una red local creada mediante un router intermedio.

A) Routing del sistema

Para garantizar una conexión persistente y confiable desde Unity hacia los microcontroladores, se utilizó un router *Trendnet* configurado con asignación de IP estática basada en la dirección MAC de cada ESP32. De este modo, cada dispositivo mantiene su IP asignada en cada reinicio: 192.168.10.48 e 192.168.10.49. Cualquier otro dispositivo se conecta mediante el protocolo DHCP.

Es fundamental que tanto el computador que ejecuta Unity como los microcontroladores estén conectados a la misma red local para asegurar la detección y comunicación constante entre ellos. Las dificultades encontradas al intentar mantener una conexión estable por medio de comunicación serial (como bloqueos de puertos y velocidades de transferencia inconsistentes) justificaron el cambio a una solución inalámbrica, la cual no solo mejoró la estabilidad del sistema, sino que también incrementó la comodidad y escalabilidad del entorno de trabajo.

7.3. Capa de Retroalimentación

Esta capa resume la programación del ESP32, las pruebas realizadas con usuarios y los ajustes efectuados en función de su comodidad.

7.3.1. Programación del ESP32

Para programar el ESP32 se utilizó el entorno Arduino IDE, apoyándose principalmente en recursos audiovisuales para implementar la comunicación vía Wi-Fi. Dentro de esta programación se destaca el siguiente componente:

A) Descifrado de coordenadas

En esta fase, se registra primero la ID de la mano. Si dicha ID coincide con la del dispositivo, se extraen secuencialmente los valores correspondientes al identificador del dedo (I_{dDedo}) y al nivel de potencia. Dado que el ESP32 no cuenta con pines PWM capaces de generar voltajes entre 0 y 3.7 V (voltaje típico de la batería), sino únicamente salidas digitales (0 V o 5 V), se optó por simular diferentes niveles de intensidad mediante la duración del *delay* en la activación de las salidas.



Figura 11: MicroMotor vibratorio
Fuente: Ecuador, 2021

Con el objetivo de ampliar las capacidades de retroalimentación física, se configuraron tres salidas digitales, cada una conectada a un micromotor vibratorio, estos a su vez están adheridas dentro del guante multifuncional (Ponce, 2025).

La intensidad de vibración se controló por software ajustando la duración de los pulsos: un pulso de 100 ms activa el motor a máxima potencia, mientras que uno de 30 ms genera una vibración de menor intensidad, debido a que el tiempo no es suficiente para alcanzar el pico de corriente eléctrica.

Esta estrategia permitió implementar una lógica sencilla pero eficaz de intensidad variable, adaptable al contexto de la interacción: cuando un dedo se detecta tocando el borde de una figura, se aplica una vibración máxima; en cambio, si el dedo se encuentra dentro del volumen interno, se genera una vibración atenuada, simulando diferentes niveles de profundidad o presión percibida por el usuario.

B) Componentes Físicos del Sistema:



Figura 12: Componentes físicos de sistema desarrollado

C) Retroalimentación de Usuarios

Se llevó a cabo una evaluación integral del sistema en la Sociedad de No Videntes del Azuay (SONVA), con la participación de usuarios con distintos grados de discapacidad visual, tanto

congénita como adquirida. Esto permitió obtener un espectro más amplio y diverso de observaciones y recomendaciones por parte de quienes probaron el sistema.

Con el apoyo de cinco usuarios, y luego de leerles y obtener su consentimiento mediante la firma del documento correspondiente (Carta de Consentimiento), se procedió a realizar la simulación. Esta integró desde el despliegue del ejecutable hasta la conexión con el router dedicado y el circuito compuesto por los dispositivos ESP32 y los micromotores incorporados en el guante multifuncional (Ponce, 2025). Todo el proceso se realizó con el acompañamiento de un cuidador vidente por participante.

Cada usuario interactuó con el sistema durante aproximadamente 50 minutos. Durante estas sesiones, se mantuvieron rondas continuas de retroalimentación y diálogo, facilitadas tanto por mi persona como por mi compañero, quien además es el desarrollador principal del guante (Ponce, 2025).

Como parte de esta fase, se aplicó una encuesta posterior al uso del sistema, con el fin de cumplir el último objetivo planteado del proyecto. Esta encuesta fue diseñada para evaluar el sistema a través de las siguientes métricas clave:

- Facilidad de uso
- Utilidad percibida
- Comodidad durante la interacción
- Precisión del sistema
- Percepción de autonomía durante el uso
- Experiencia general
- Claridad del entorno
- Experiencia previa (como contexto del usuario)

Se utilizó una escala de valoración del 1 al 5, donde 1 representa una experiencia deficiente y 5 una experiencia excelente.

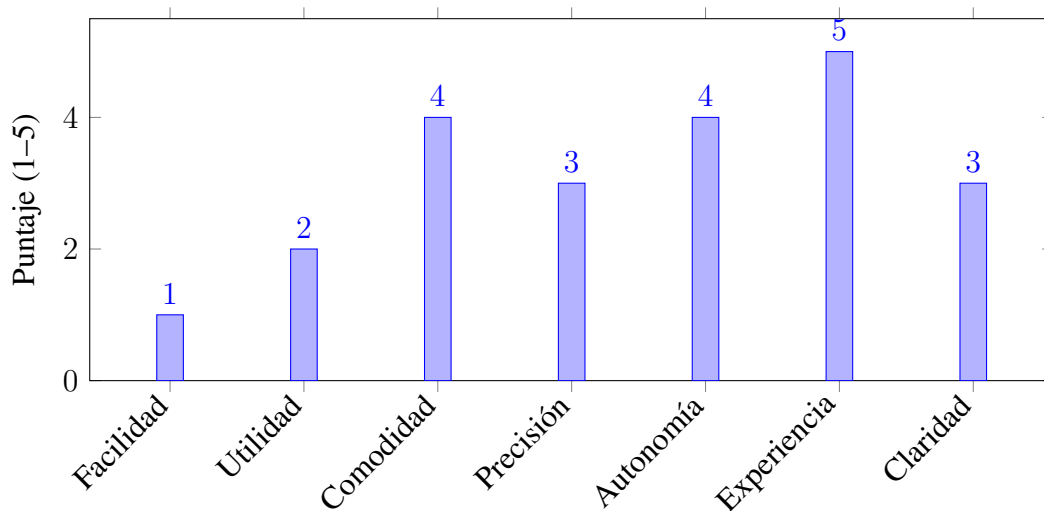


Figura 13: Ejemplo no factible: Simulación de resultados con experiencia normal (escala 1 a 5)

Durante esta fase, se mantuvo una comunicación directa con los usuarios, lo que permitió obtener retroalimentación cualitativa adicional. Gracias a estas interacciones, fue posible identificar áreas de mejora, realizar ajustes técnicos y refinar aspectos tanto de la interfaz como de la retroalimentación física del sistema. Esta validación fue esencial no solo para evaluar el desempeño técnico del sistema, sino también para asegurar que resultara inmersivo, útil y comprensible para personas con discapacidad visual.

7.4. Análisis final sobre la implementación en capas

La implementación de una estructura en capas fue fundamental para optimizar y organizar el flujo de información desde la captura de datos tridimensionales con Leap Motion hasta la generación de retroalimentación física en el usuario. Esta arquitectura permitió una separación clara de responsabilidades entre la adquisición de datos, el procesamiento lógico y la comunicación con los actuadores, facilitando el mantenimiento y la escalabilidad del sistema.

Gracias a esta segmentación, fue posible reemplazar la comunicación serial por una inalámbrica sin reestructurar por completo el sistema. Cada capa pudo adaptarse de forma independiente, lo que evidenció una mayor flexibilidad y robustez en el desarrollo. Esta estructura también ayudó a establecer patrones reutilizables, a mejorar la estabilidad del sistema y a reducir las limitaciones físicas asociadas con soluciones cableadas.

Además, la organización por capas permitió definir con claridad los puntos de entrada y salida de datos, asegurando una interpretación precisa de la información proveniente del sensor y su adecuada traducción en estímulos hápticos. Esta modularidad fue clave para implementar funciones como el aviso por colisión con zonas internas de figuras virtuales, mejorando la experiencia inmersiva.

Por lo tanto, la estructura en capas no solo facilitó el desarrollo técnico, sino que también potenció la capacidad del sistema para simular una interacción realista, accesible y libre de barreras físicas, abriendo nuevas posibilidades en la exploración de interfaces para personas con discapacidad visual.

7.5. Resumen de la arquitectura del sistema desarrollado

Resumen de sistema desarrollado previamente explicado:

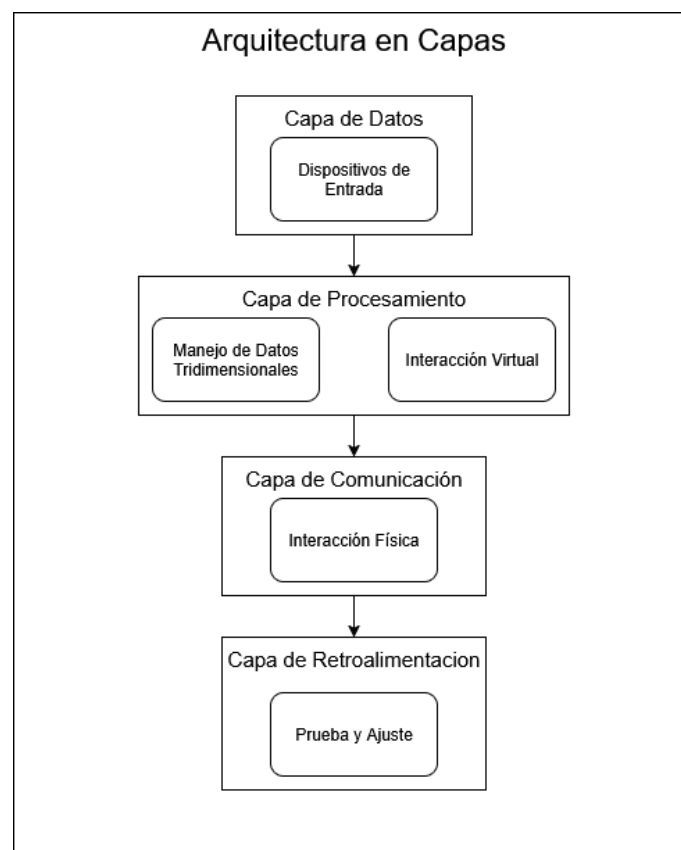


Figura 14: Arquitectura en capas resumido del sistema propuesto

8. Pruebas y Resultados

8.1. Investigación teórica

Investigación sobre la aplicación de teoremas y fundamentos necesarios para su implementación en Unity.

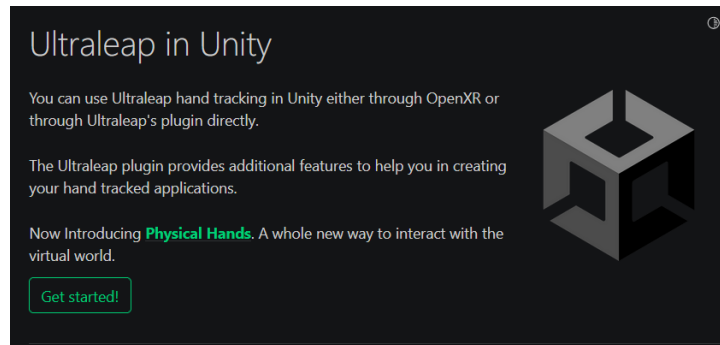


Figura 15: Actividad de investigación
Fuente: Ultraleap, 2024a

8.2. Configuración de librerías Ultraleap

Aplicación de métodos y librerías proporcionadas por Ultraleap para controlar el sensor Leap Motion desde Unity.

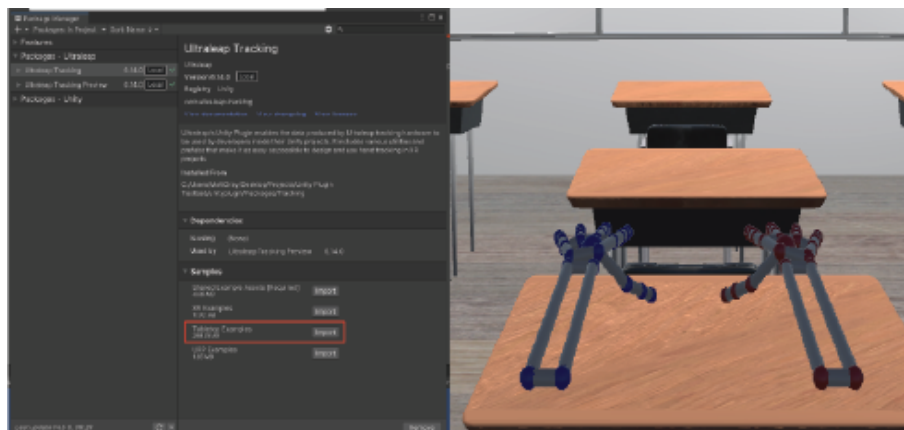


Figura 16: Actividad de configuración

8.3. Extracción de coordenadas

Desarrollo de funciones para la extracción de coordenadas medidas por el sensor Leap Motion en tiempo real.



Figura 17: Extracción de coordenadas

Permitió el correcto adquisición y transformación de la coordenadas adquiridas a coordenadas de Unity.

8.4. Funciones de interacción

Creación de funciones lógicas basadas en las coordenadas obtenidas, orientadas a interpretar la interacción física del usuario.

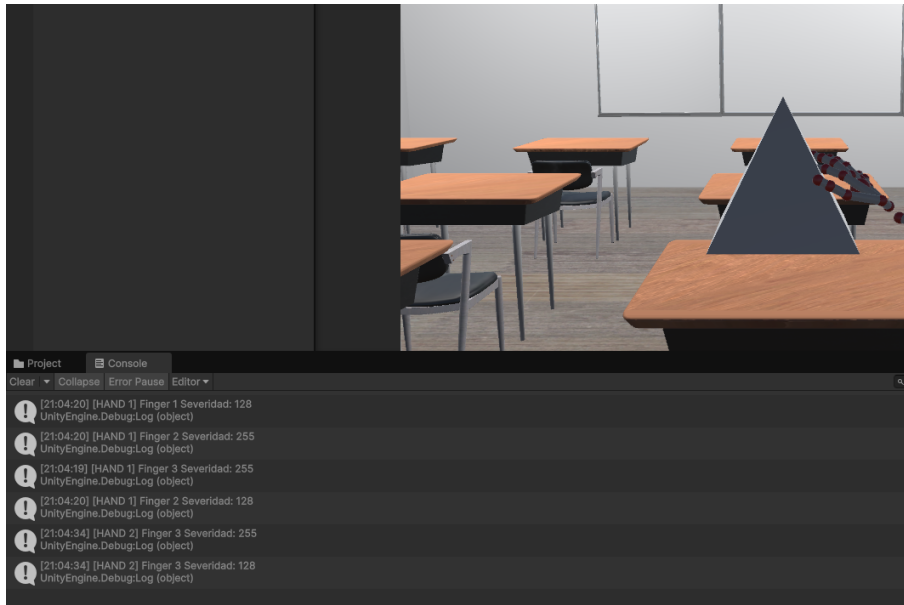


Figura 18: Interacción con los objetos

8.5. Construcción de entornos virtuales

Se diseñaron y crearon entornos personalizados que ejemplifican la interacción del usuario con objetos virtuales, favoreciendo la comprensión del sistema y su usabilidad.

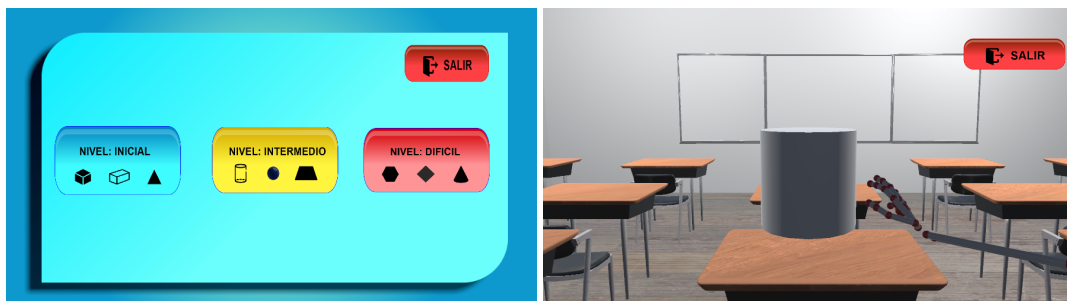


Figura 19: Elección de entorno personalizado

El resultado fue un diseño sencillo y comprensible. Con éxito en aislar la lógica de navegación y carga de escenas; de modo que estos no interrumpan el hilo principal de Unity.

8.6. Verificación de métodos

Se realizó la vinculación y validación de los métodos utilizados para la extracción de coordenadas y la manipulación del espacio interactivo, asegurando su funcionamiento dentro del entorno virtual.

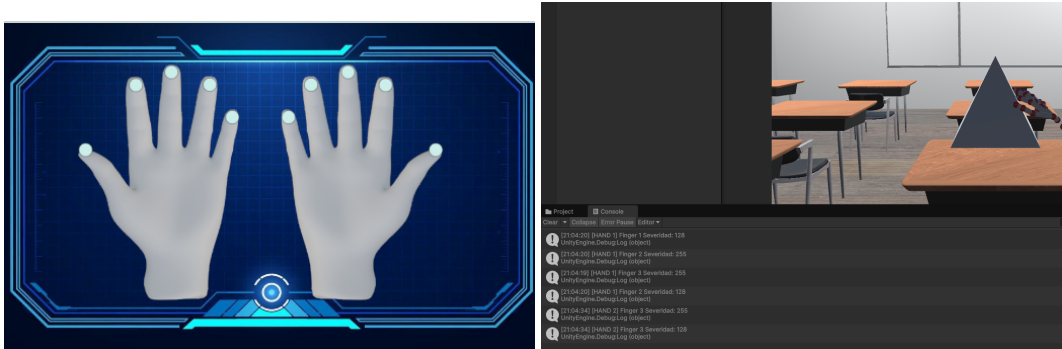


Figura 20: Verificación de métodos

Este proceso derivó en una reconfiguración consistente y persistente dentro del proyecto, lo que permitió minimizar los errores de formato de datos y reducir significativamente los crasheos durante la ejecución.

8.7. Comunicación con hardware externo

Se desarrolló la comunicación entre Unity y un dispositivo físico (ESP32), utilizando los datos procesados generados por el usuario para controlar funciones específicas del hardware.

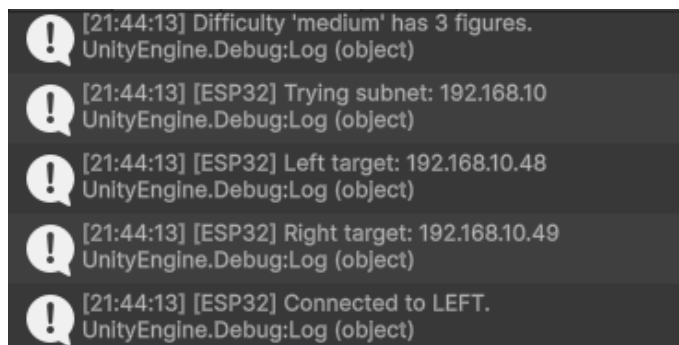


Figura 21: Comunicación con hardware

Se logró establecer una comunicación estable y segura, en la cual los datos pueden ser transmitidos correctamente al dispositivo correspondiente, de acuerdo con los requerimientos funcionales.

8.8. Envío de datos

Distribución de datos procesados a través de la comunicación Wi-Fi previamente establecida con el microcontrolador.

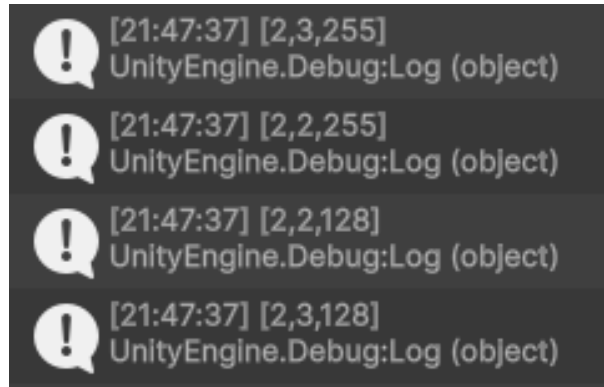


Figura 22: Envío de datos

Se realizó la transmisión del arreglo de datos correspondiente a la activación de los micromotores hápticos del guante, utilizando el canal de comunicación establecido con el dispositivo ESP32 (Ponce, 2025).

8.9. Interfaz para usuarios con discapacidad visual

Diseño e implementación de una interfaz accesible, considerando que los usuarios principales son personas con discapacidad visual.

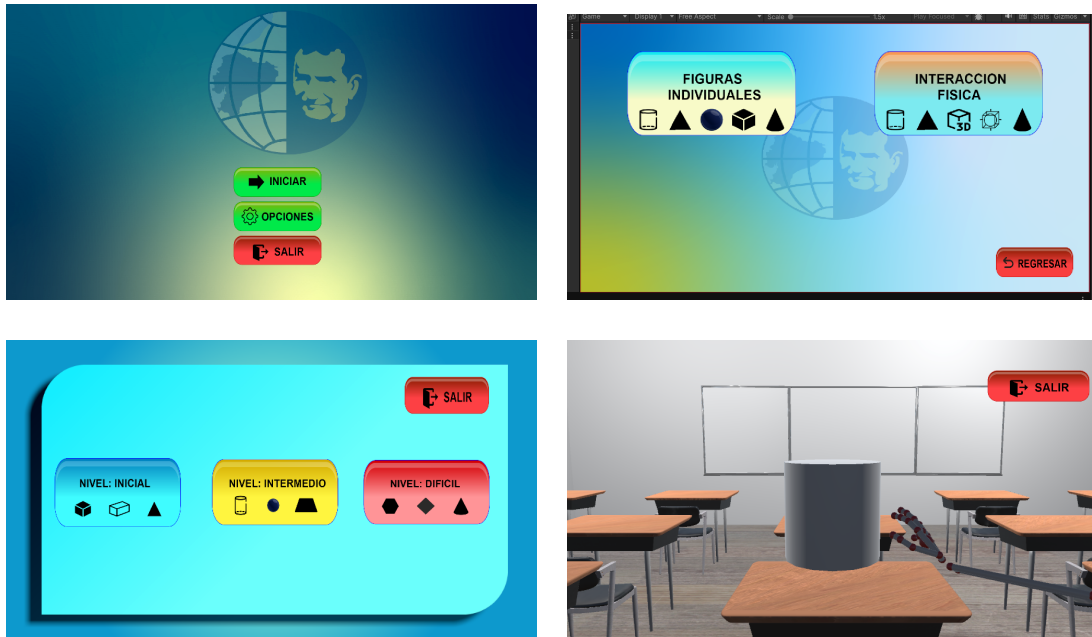


Figura 23: Interfaz de usuario

La interfaz fue desarrollada utilizando sprites diseñados en Adobe Illustrator, con una estructura de navegación lógica y transiciones de escena lineales. Estas decisiones facilitaron la interacción y orientación de los usuarios dentro del entorno virtual, reduciendo la complejidad y mejorando la experiencia de uso.

8.10. Métodos de retroalimentación auditiva

Expansión de la interfaz virtual mediante la incorporación de descripciones auditivas para los objetos seleccionados, con el objetivo de mejorar la accesibilidad para usuarios con discapacidad visual.



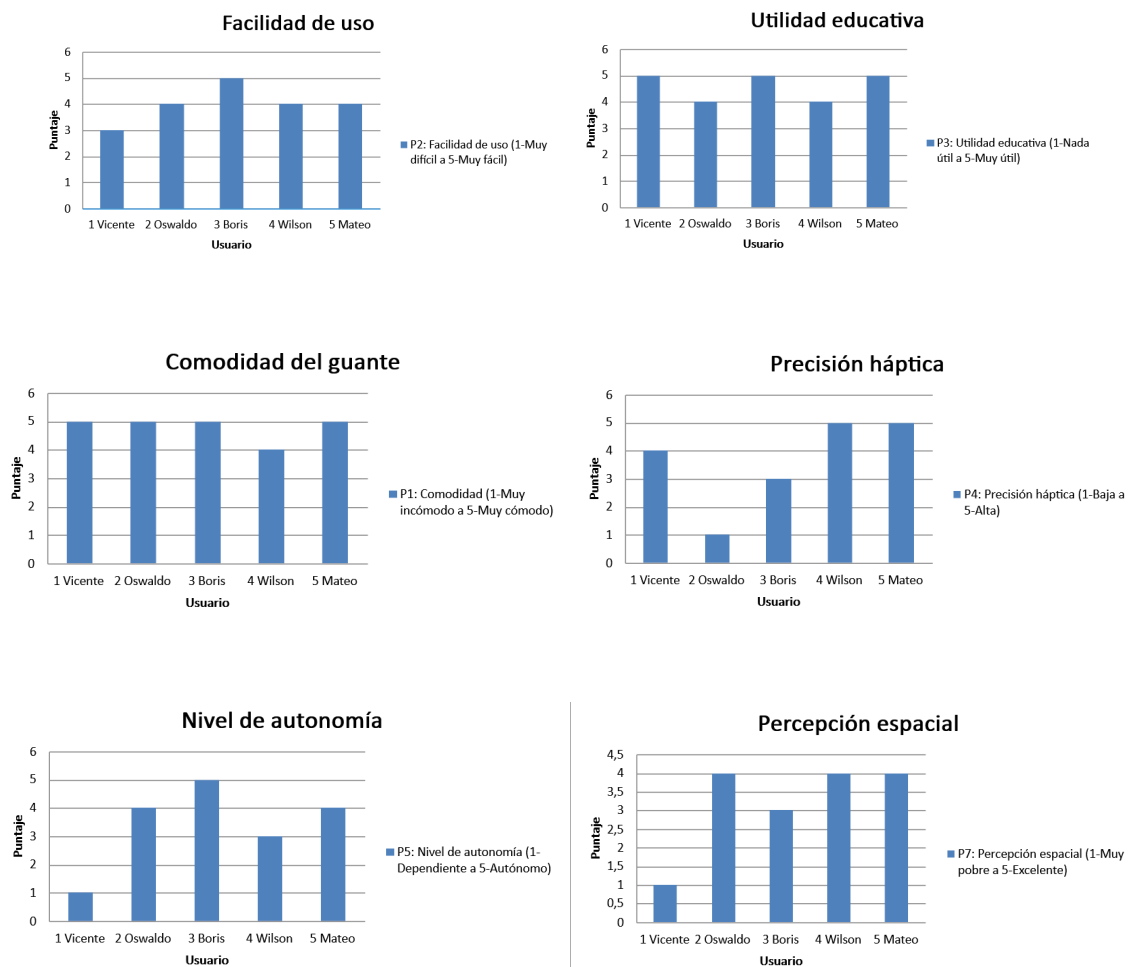
Figura 24: Retroalimentación auditiva

El resultado fue una implementación funcional que permite la reproducción de audios descriptivos de manera sencilla. Se utilizó el componente **AudioSource** de Unity, lo que facilita futuras modificaciones o ampliaciones del sistema de retroalimentación auditiva.

8.11. Validación con usuarios

Los resultados obtenidos del ejercicio en SONVA, reflejan una recepción mayoritariamente positiva y en algunas preguntas con respuesta neutral, lo cual refuerza la viabilidad y pertinencia del sistema propuesto. En general, los participantes valoraron favorablemente aspectos como la facilidad de uso, la utilidad práctica del sistema, la comodidad durante la interacción, la precisión en las respuestas y, especialmente, la autonomía percibida durante la experiencia de uso.

Las gráficas presentadas a continuación muestran los resultados de dicha encuesta.



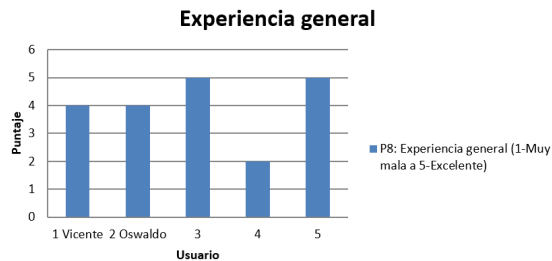


Figura 25: Resultados de la validación por categorías

Estos resultados de las encuestas sugiere que el proyecto responde a una necesidad real dentro de la comunidad con discapacidad visual, abriendo oportunidades para su adopción y mejora continua. Entre los comentarios más relevantes, los usuarios propusieron ciertas mejoras para futuras versiones del sistema, tales como:

- Incorporación de una retroalimentación auditiva más personalizada y clara.
- Reubicación de los sensores hápticos vibratorios hacia la punta de los dedos para una mayor sensibilidad y precisión.
- Aumento de potencia de la estimulación física.
- Aumento de rango visual para así obtener un rango de movimiento mas estático y cómodo.

El involucramiento directo de los usuarios en esta etapa confirma la importancia de un enfoque centrado en el ser humano, donde la inclusión y la accesibilidad no son complementos, sino pilares fundamentales del desarrollo tecnológico.

8.12. Documentación final

El repositorio contiene un archivo **README.md** estructurado en formato Markdown, que documenta detalladamente el proceso de instalación, configuración y ejecución del sistema. Dentro del entorno de desarrollo Unity, se estableció una estructura de carpetas organizada y jerárquica, segmentada según la funcionalidad de los elementos. Esta división incluye:

- **Materiales:** Archivos visuales aplicados a los modelos 3D.

- **Sprites 2D:** Elementos gráficos utilizados para la interfaz de usuario.
- **Scripts:** Código fuente en C# para la lógica del sistema.
- **Audios:** Archivos de sonido para retroalimentación auditiva.
- **Escenas:** Organización de los objetos en jerarquías (GameObjects), representando los distintos estados o niveles de la aplicación.

Esta estructura modular favorece la mantenibilidad del sistema y permite que los diferentes componentes puedan ser fácilmente modificados, reemplazados o ampliados sin afectar el resto del proyecto.

9. Conclusión final

El proceso de validación con usuarios finales permitió comprobar que el sistema propuesto cumple adecuadamente con los objetivos planteados, especialmente en términos de facilidad de uso, utilidad, precisión y comodidad. Los resultados obtenidos en las encuestas reflejan una valoración mayoritariamente positiva, destacando la experiencia general y la percepción de autonomía que brinda la interfaz.

El diseño centrado en el usuario y el enfoque iterativo aplicado durante el desarrollo fueron fundamentales para alcanzar un prototipo funcional, accesible y coherente con las necesidades reales del grupo objetivo. La inclusión de retroalimentación directa permitió realizar ajustes significativos que mejoraron tanto la interacción como la percepción del sistema. Asimismo, la documentación generada, junto con las pruebas sistemáticas realizadas, garantiza la reproducibilidad del proyecto y sienta las bases para su implementación en escenarios reales o su extensión a otros contextos educativos y tecnológicos.

Gracias a la culminación del proyecto, este puede posicionarse como un punto de partida sólido para contribuir de forma profesional, ética y abierta al cumplimiento del Objetivo de Desarrollo Sostenible (ODS) 4, el cual promueve una educación inclusiva, equitativa y de calidad para todos. En este sentido, la solución desarrollada no solo representa una herramienta de apoyo, sino también un símbolo del compromiso con la accesibilidad y la equidad tecnológica.

En definitiva, este trabajo resalta la importancia de involucrar activamente a los usuarios en todas las fases del desarrollo, no solo como destinatarios finales, sino como actores clave que orientan las decisiones de diseño, retroalimentan las funcionalidades y validan los avances del sistema. Este enfoque participativo y humano fortalece la pertinencia del producto final y reafirma el valor de diseñar tecnología al servicio de todos.

10. Recomendaciones finales

Durante el desarrollo de las etapas finales del sistema, se consideró la posibilidad de modificar los puntos de contacto donde se emplean los motores hápticos. Gracias a la retroalimentación de los usuarios, se identificó que sería valioso experimentar con diferentes ubicaciones para evaluar la experiencia y la percepción del usuario. Esta experimentación podría medirse utilizando los resultados de la misma encuesta aplicada en la fase de validación.

En cuanto al hardware, se recomienda considerar la integración de un módulo más potente, que permita un control más preciso y rápido de los actuadores hápticos. Esto contribuiría a una mejor experiencia de usuario, reduciendo la latencia en la activación de los estímulos táctiles.

Respecto a la capa de comunicación, se sugiere explorar el uso de tecnología Bluetooth como alternativa a la conexión Wi-Fi, con el fin de simplificar el sistema y reducir la necesidad de hardware adicional, como el router dedicado.

En lo relacionado con las funcionalidades del sistema, la sección de retroalimentación auditiva podría ampliarse para incluir sugerencias expresadas por algunos usuarios durante las pruebas.

Entre estas destacan:

- Instrucciones sonoras que permitan una navegación más autónoma dentro del entorno virtual.
- Control de la interfaz mediante comandos de voz, reduciendo así la dependencia de un asistente vidente.
- Configuraciones dinámicas que permitan ajustar en tiempo real las propiedades de las figuras virtuales, como su escala y rotación.

11. Glosario

Leap Motion: Dispositivo de entrada que permite el seguimiento preciso de las manos y dedos en 3D, utilizado para interacciones naturales en entornos virtuales.

ESP32: Microcontrolador de bajo consumo y alto rendimiento con capacidades WiFi y Bluetooth, utilizado comúnmente en proyectos de IoT y sistemas embebidos.

Unity: Motor de desarrollo de videojuegos y entornos interactivos en tiempo real, utilizado en este caso para la creación de interfaces accesibles.

Hilos (Threads): Unidad de ejecución dentro de un proceso que permite la realización de tareas en paralelo.

Serial: Comunicación por puerto serie, común en microcontroladores para enviar y recibir datos a través de cables.

Comunicación inalámbrica: Transmisión de datos sin cables, usando WiFi para mejorar la libertad de movimiento del usuario.

Sistema háptico: Tecnología que proporciona retroalimentación táctil mediante vibraciones u otras sensaciones físicas, simulando el sentido del tacto.

Retroalimentación auditiva: Información sonora proporcionada al usuario, como voz sintética o efectos, para comunicar estados del sistema.

Entorno virtual: Espacio simulado digitalmente donde el usuario puede interactuar con objetos tridimensionales.

Accesibilidad: Principio de diseño centrado en permitir el uso de sistemas a personas con discapacidades.

Interfaz accesible: Sistema de interacción pensado para ser comprendido y utilizado por personas con discapacidades, con soporte auditivo y/o háptico.

Capas (estructura en capas): Modelo de arquitectura de software donde los componentes están organizados en niveles.

Interacción multisensorial: Integración de múltiples sentidos (tacto, oído, vista) para lograr una experiencia más inmersiva.

Prueba y error (debugging): Método de desarrollo donde se identifican y corrigen errores del sistema iterativamente.

Experiencia de usuario (UX): Percepción y respuesta de una persona al usar un sistema, influenciada por su facilidad de uso.

Sistema tridimensional: Sistema que maneja coordenadas en tres dimensiones (X, Y, Z) para representar objetos o interacciones espaciales.

Movimiento libre: Capacidad del usuario para interactuar sin estar atado por cables u obstáculos.

Entornos educativos: Espacios virtuales con fines pedagógicos, especialmente útiles para usuarios con discapacidad.

Arduino: Plataforma de hardware y software de código abierto usada para prototipos electrónicos, similar al ESP32.

Feedback: Retroalimentación sensorial (visual, auditiva, táctil) que indica el resultado de una acción del usuario.

Microcontrolador: Chip programable que ejecuta tareas específicas dentro de un sistema embebido.

Desarrollo iterativo: Modelo de desarrollo basado en ciclos sucesivos de diseño, prueba y mejora.

Experiencia accesible: Experiencia de usuario diseñada especialmente para personas con necesidades particulares.

12. ANEXO

12.1. Imágenes de pruebas en SONVA

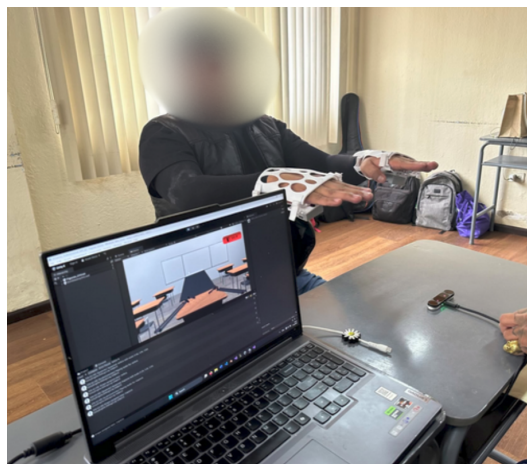


Figura 26: Prueba física con usuario

12.2. Plataforma de Código Fuente

El código fuente completo está alojado en el repositorio privado de GitHub en la siguiente URL:

Código fuente disponible en el repositorio de GitHub:

https://github.com/Fernandourgiles/Proyecto_Tridimensional_UPS_65.git

Nota: Este repositorio es privado, por lo que sólo los usuarios con permiso explícito pueden acceder a su contenido. Si desea colaborar o revisar el proyecto, debe

solicitar acceso al propietario.

12.3. Estado de desarrollo según actividades planteadas

Cuadro 1: Actividades del Proyecto (Abril - Julio)

Código	Actividad
AC1	Investigación de aplicación de teoremas a utilizar en Unity
AC2	Aplicación de métodos y librerías Ultraleap para controlar el sensor Leap Motion en Unity.
AC3	Extracción de las coordenadas medidas por el Leap Motion.
AC4	Creación de funciones ligados a las coordenadas Leap Motion
AC5	Creación de entornos propios para ejemplificar la interacción virtual.
AC6	Vinculación y verificación de los métodos utilizados para extracción de coordenada y manipulación del espacio.
AC7	Comunicación con un dispositivo externo según los datos registrados desde Unity.
AC8	Distribución de datos mediante la comunicación serial establecida.
AC9	Creación de interfaz general tomando en cuenta que los usuarios primarios son no videntes.
AC10	Creación de la interfaz virtual para renderizado del entorno con que se interactuara.
AC11	Comunicación con usuarios potenciales para ajustes, limpieza, sugerencias.
AC12	Documentación del código final.

Resumen de fin de cada actividad propuesta

Objetivo	Actividad	Fecha Fin
OE1	AC1	26/04/2025
OE1	AC2	26/04/2025
OE1	AC3	28/04/2025
OE2	AC4	09/05/2025
OE2	AC5	13/05/2025
OE2	AC6	13/05/2025
OE2	AC7	23/05/2025
OE3	AC8	23/05/2025
OE3	AC9	09/06/2025
OE3	AC10	20/06/2025
OE4	AC11	14/07/2025 (Retraso en pruebas usuario)
OE4	AC12	14/07/2025 (Retraso en documentación final)
HORAS TOTALES		240h

12.4. Carta De Consentimiento Informado

CONSENTIMIENTO INFORMADO

21 de julio de 2025

Proyecto: “Guante Multifunción con Entorno Virtual Inmersivo y Comunicación Háptica para Personas con Discapacidad Visual”

Investigador Principal: Ronny Joel Morocho Ponce (Estudiante de Ingeniería Biomédica)

Entidades: Universidad Politécnica Salesiana (UPS), GiB (Grupo de Investigación en Ingeniería Biomédica), GiMaT (Grupo de Investigación en Nuevos Materiales y Procesos de Transformación)

Colaboradores: SONVA (Sociedad de No Videntes del Azuay)

1. PROPÓSITO DE LA INVESTIGACIÓN

Usted está invitado(a) a participar en una prueba de usabilidad del guante háptico multifunción, desarrollado para mejorar la autonomía de personas con discapacidad visual. El objetivo es evaluar su eficacia en navegación espacial e interacción con entornos virtuales.

2. PROCEDIMIENTOS

- **Duración:** 1 sesión de 30–45 minutos.
- **Actividades:**
 - Uso del guante háptico en un entorno virtual (simulación de aula).
 - Realización de tareas como identificar obstáculos y manipular objetos virtuales.
 - Encuesta verbal de satisfacción (preguntas sobre comodidad y facilidad de uso).

- **Lugar:** Laboratorios de la UPS (Campus Cuenca) o sede de SONVA, según su preferencia.

3. BENEFICIOS

- **Para usted:** Acceso gratuito a tecnología innovadora y retroalimentación personalizada sobre su experiencia.
- **Para la comunidad:** Contribuirá al desarrollo de herramientas que mejorarán la inclusión educativa y social de personas con discapacidad visual.

4. RIESGOS O MOLESTIAS

- **Mínimos:** Fatiga muscular leve por movimientos repetitivos.
- **Precauciones:**
 - Pausas durante la sesión si manifiesta incomodidad.
 - Ajuste del guante para evitar presión excesiva.

5. CONFIDENCIALIDAD

Toda la información recopilada (audio, notas, datos de desempeño) será anónima. Los datos se almacenarán en servidores cifrados de la UPS y solo el equipo investigador tendrá acceso. Los resultados se publicarán sin identificar a participantes individuales.

6. VOLUNTARIEDAD

Su participación es totalmente voluntaria. Puede retirarse en cualquier momento sin explicación ni consecuencias. Si se retira, sus datos serán destruidos inmediatamente.

7. COMPENSACIÓN

No habrá compensación económica, pero recibirá:

- Un informe de resultados del estudio.
- Prioridad en futuros talleres de tecnología accesible organizados por UPS/SONVA.

8. CONTACTO

Investigador Principal: Ronny Joel Morocho Ponce

Teléfono: 0969033515

Correo: rmorechop3@est.ups.edu.ec

Tutor Académico: Ing. Patricio Urgiles Ortiz PhD.

Correo: furgiles@ups.edu.ec

SONVA: Representante: Jose Tenezaca

DECLARACIÓN DE CONSENTIMIENTO

“He leído (o me han leído) toda la información anterior. Comprendo el propósito, procedimientos, beneficios y riesgos de este estudio. Todas mis dudas han sido resueltas. Autorizo mi participación voluntaria y el uso anónimo de mis datos con fines científicos.”

Participante

Testigo (Representante SONVA)

Nombre: _____

Nombre: _____

Firma o Huella Dactilar: _____

Firma: _____

Fecha (Participante): _____

Fecha (Testigo): _____

“Su voz impulsa la innovación inclusiva”

Universidad Politécnica Salesiana · GiIB · GiMaT · SONVA

Referencias

- AliExpress. (2025). Aparatos de audio utilizados en el sistema de salida sonora [Accedido en julio de 2025].
- Christopher G. Thomas, P. (2025). Project Prakash Enlightens Our Understanding of Vision [Última actualización: 2 de enero de 2025. Basado en investigaciones de Ostrovsky et al. (2006, 2009) y Held et al. (2005)]. <https://www.nei.nih.gov/about/goals-and-accomplishments/nei-research-initiatives/international-vision-research/project-prakash-enlightens-our-understanding-vision>
- CONADIS. (2025). Estadísticas de Discapacidad [Accedido en julio de 2025]. <https://www.consejodiscapacidades.gob.ec/estadisticas-de-discapacidad/>
- Ecuador, N. (2021). Motor de vibración tipo pancake [Imagen de componente electrónico, accedido en julio de 2025].
- Electronics-Lab. (2025). Uso de la funcionalidad BLE del ESP32 en proyectos integrados [Accedido en julio de 2025].
- Group, L. (2025). Documentación de Leap Motion para Interact SDK [Accedido en julio de 2025].
- Naciones Unidas Ecuador. (2025). ODS 4: Educación de Calidad - Objetivos de Desarrollo Sostenible [Accedido en julio de 2025].
- Ponce, R. J. M. (2025). Guante Multifunción con Entorno Virtual Inmersivo y Comunicación Háptica para Personas con Discapacidad Visual [Trabajo en desarrollo, no publicado].
- Sadato, N., Pascual-Leone, A., Grafman, J., Ibañez, V., Deiber, M., Dold, G., & Hallett, M. (1996). Activation of the primary visual cortex by Braille reading in blind subjects. *Nature*, 380(6574), 526-528. <https://doi.org/10.1038/380526a0>
- Studio, G. (2022). Importing FBX models the right way - Unity Tutorial (Rotation & Scale Fix) [YouTube video, accessed on 2025-07-18].

Tripathy, P. K., & Mohanty, S. (2023). Leap Motion Controller with its range of visibility [Imagen en ResearchGate, accedido en julio de 2025].

Ultraleap. (2024a). Getting started with Unity. <https://docs.ultraleap.com/xr-and-tabletop/xr/unity/getting-started/index.html>

Ultraleap. (2024b). Ultraleap Plugin Features. <https://docs.ultraleap.com/xr-and-tabletop/xr/unity/plugin/index.html>

Unity Technologies. (2024a). AudioSource Class (UnityEngine) [Unity Scripting API. Accessed: 2025-07-18].

Unity Technologies. (2024b). MeshCollider [Unity Scripting API. Accessed: 2025-07-18].