

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

**CARRERA:
INGENIERÍA ELECTRÓNICA**

**Trabajo de titulación previo a la obtención del título de:
INGENIERO ELECTRÓNICO**

**TEMA:
DESARROLLO DE UN JUEGO DIDÁCTICO MEDIANTE SENSORES
LEAP MOTION PARA ESTIMULAR EL APRENDIZAJE DEL LENGUAJE
BÁSICO DE SEÑAS ECUATORIANO**

**AUTOR:
MATEO ALEJANDRO PARREÑO ALVAREZ**

**TUTORA:
CARMEN JOHANNA CELI SÁNCHEZ**

Quito, marzo del 2018

CESIÓN DE DERECHOS DE AUTOR

Yo Mateo Alejandro Parreño Alvarez, con documento de identificación N°0502786247, manifiesto mi voluntad y cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del trabajo de titulación intitulado: “DESARROLLO DE UN JUEGO DIDÁCTICO MEDIANTE SENSORES LEAP MOTION PARA ESTIMULAR EL APRENDIZAJE DEL LENGUAJE BÁSICO DE SEÑAS ECUATORIANO”, mismo que ha sido desarrollado para optar por el título de Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



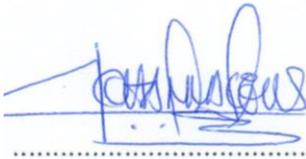
.....
Mateo Alejandro Parreño Alvarez
C.I. 0502786247

Quito, marzo 2018.

DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTORA

Yo declaro que bajo mi dirección y asesoría fue desarrollado el Trabajo de Titulación, “DESARROLLO DE UN JUEGO DIDÁCTICO MEDIANTE SENSORES LEAP MOTION PARA ESTIMULAR EL APRENDIZAJE DEL LENGUAJE BÁSICO DE SEÑAS ECUATORIANO”, realizado por Mateo Alejandro Parreño Alvarez obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerados como trabajo final de titulación.

Quito, marzo 2018.



.....
Carmen Johanna Celi Sánchez

C.I. 1717437808

DEDICATORIA

Dedicado a Dios, a mí familia por ser parte fundamental en mi vida, brindarme su apoyo en momentos difíciles e impulsarme a superar adversidades; en especial quiero dedicar este proyecto a mis padres Raúl Parreño y Jessy Alvarez quienes con su esfuerzo, sacrificio y consejos supieron guiarme por el camino correcto para cumplir mi sueño; a mis hermanos Richard y Jessy quienes me han brindado su amistad, carisma y apoyo incondicional, a mis tíos y primos que me dieron ánimos y me enseñaron a no rendirme por más complicada que sea la situación y a mis abuelitos por la confianza que han depositado en mí.

AGRADECIMIENTO

A mi tutora Ingeniera Johanna Celi, Mgtr., por su valiosa labor en las tutorías y dirección del presente proyecto.

A la Universidad Politécnica Salesiana sede Quito, que con sus instalaciones y metodología ha hecho posible que pueda lograr una meta profesional en el interminable camino por alcanzar el conocimiento y por el apoyo económico para el viaje a la ponencia del artículo científico “ Teaching-Learning of Basic Language of Signs through Didactic Games” en el ICETC 2017 que se desarrolló en Barcelona, España el cual me ofreció una fuente adicional de conocimientos técnico-científicos y compartir experiencias con jóvenes de otras universidades y profesionales

Un agradecimiento especial al Ingeniero Víctor Andaluz PhD, Ingeniero David Rivas, Mgtr., Ingeniero Washington Quevedo y a los integrantes del Laboratorio de Investigación Automatización, Robótica y Sistemas Inteligentes ARSI de la Universidad de las Fuerzas Armadas ESPE - Sede Latacunga, por su valiosa colaboración que hizo posible el desarrollo del presente proyecto de titulación y quienes han hecho lo posible por incluirme en el fantástico proceso de investigación con paciencia y dedicación.

Un agradecimiento a Germanía Núñez por la facilidad para realizar las pruebas de este proyecto en el Colegio “Primero de Abril” de la ciudad de Latacunga.

ÍNDICE DE CONTENIDO

CESIÓN DE DERECHOS DE AUTOR.....	i
DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTORA	ii
DEDICATORIA	iii
AGRADECIMIENTO.....	iv
ÍNDICE DE CONTENIDO.....	v
RESUMEN.....	x
ABSTRACT	xi
INTRODUCCIÓN	xii
CAPÍTULO 1.....	1
ANTECEDENTES	1
1.1. Planteamiento del problema.....	1
1.2. Justificación del proyecto	1
1.3. Objetivos.....	2
1.4. Objetivo general.....	2
1.4.1. Objetivos específicos	2
1.5. Propuesta de solución	2
1.6. Alcance	2
CAPÍTULO 2.....	4
MARCO CONCEPTUAL	4
2.1. Lenguaje de señas ecuatoriano	4
2.2. Psicología del color en el aprendizaje.....	5
2.2.1. Los colores en el aprendizaje	5
2.3. Tecnología gestual	5
2.4. Leap motion	6
2.4.1. Cámaras.....	8
2.4.2. Iluminación infrarroja	8
2.4.3. El microcontrolador	9
2.5. Unity3D	9
2.6. Matlab	9
2.7. Estado del arte.....	10
CAPÍTULO 3.....	12
DISEÑO E IMPLEMENTACIÓN	12
3.1. Procesamiento de datos.....	13

3.2.	Comunicación bilateral	14
3.3.	Extracción de características.....	17
3.4.	Clasificador	21
3.4.1.	Configuraciones estáticas.....	21
3.4.2.	Configuraciones dinámicas	24
3.5.	Interfaz de juego	26
3.5.1.	Fuga de letras	27
3.5.2.	Tres en raya	32
3.5.3.	Multijugador.....	35
CAPÍTULO 4	41
PRUEBAS	41
4.1.	Distancia de interacción.....	41
4.2.	Reconocimiento de señas.....	44
4.3.	Tiempo de respuesta	46
4.4.	Test de usabilidad	48
4.4.1.	Escala cuestionario sus.....	49
4.4.2.	Puntuación de sus.....	49
4.4.3.	Experimentación	49
CONCLUSIONES	54
RECOMENDACIONES	56
BIBLIOGRAFÍA	57
ANEXO		

ÍNDICE DE FIGURAS

Figura 2.1 Leap Motion.....	6
Figura 2.2 Zona de cobertura del dispositivo.....	7
Figura 2.3 Partes del sensor Leap Motion.....	7
Figura 3.1 Desarrollo del funcionamiento del sistema.....	12
Figura 3.2 Visualizador SDK Leap Motion.....	13
Figura 3.3 Rastreo de manos con Leap Motion.....	14
Figura 3.4 Lectura del rastreo de manos.....	14
Figura 3.5 Interfaz de la memoria compartida.....	15
Figura 3.6 Script de Unity3D Memoria compartida.....	16
Figura 3.7 Script de Matlab Memoria compartida.....	16
Figura 3.8 Reordenamiento de vector.....	18
Figura 3.9 Señas del lenguaje básico de señas configuración.....	19
Figura 3.10 Imagen estroboscópica de seña dinámica.....	21
Figura 3.11 Cuadrícula de distancia global.....	25
Figura 3.12 Menú de inicio del juego.....	27
Figura 3.13 Juego Fuga de letras.....	28
Figura 3.14 Desarrollo juego fuga de letras.....	28
Figura 3.15 Script PE_Matlab_D.....	29
Figura 3.16 Script Respuesta.....	29
Figura 3.17 Script BasePalabrasImágenes.....	30
Figura 3.18 Script Comparador.....	30
Figura 3.19 Script ControladorScenes.....	31
Figura 3.20 Script Timer.....	31
Figura 3.21 Juego Tres en raya.....	32
Figura 3.22 Desarrollo juego Tres en raya.....	32
Figura 3.23 Script PE_Matlab.....	33
Figura 3.24 Script Respuesta.....	33
Figura 3.25 Script ControladorScenes.....	34
Figura 3.26 Script Timer2.....	34
Figura 3.27 Script TicTacToe.....	35
Figura 3.28 Servidor y host.....	36
Figura 3.29 Diagrama de jugador y clientes en modo multijugador.....	37

Figura 3.30 Modulo Network Manager	38
Figura 3.31 Modulo ManagerHUB	39
Figura 3.32 Modulo Network Identity	39
Figura 3.33 Información inherente a la conexión de red	40
Figura 3.34 Modulo Network Transform.....	40
Figura 4.1 Ejes de coordenadas de Leap Motion	42
Figura 4.2 Interacción en el eje X	42
Figura 4.3 Interacción eje Y	43
Figura 4.4 Interacción en el eje Z.....	43
Figura 4.5 Tiempo de respuesta del sistema	48
Figura 4.6 Prueba del juego fuga de letras	50
Figura 4.7 Prueba del juego tres en raya	51
Figura 4.8 Resultado del juego fuga de letras	51
Figura 4.9 Resultado del juego tres en raya	52

ÍNDICE DE TABLAS

Tabla 3.1 Matriz Rmxn	17
Tabla 3.2 Valores de correlación validos de la letra B	20
Tabla 3.3 Valores de correlación no validos de la letra C	20
Tabla 3.4 Sintonización de configuración estática.....	23
Tabla 3.5 Sintonización de configuración estática.....	24
Tabla 3.6 Sintonización configuración dinámica.....	26
Tabla 4.1 Distancia de interacción de la letra B.....	44
Tabla 4.2 Prueba de reconocimiento de letras de configuración estáticas.....	45
Tabla 4.3 Resultados del reconocimiento de letras de configuración dinámicas.....	46
Tabla 4.4 Prueba de reconocimiento de números de configuración estáticas.....	46
Tabla 4.5 Resultados del tiempo de respuesta	47
Tabla 4.6 Cuestionario de resultados	53

RESUMEN

En este proyecto de titulación se presenta el desarrollo de un juego didáctico para la enseñanza-aprendizaje del lenguaje básico de señas ecuatoriano, para personas con deficiencia auditiva sin descartar que el sistema puede ser utilizado por niños, jóvenes y adultos que tengan o no esta discapacidad y con interés en aprender el lenguaje básico de señas ecuatoriano. El dispositivo de entrada es un Leap Motion, que detecta las señales gestuales a través del reconocimiento de los huesos y articulaciones de cada dedo de la mano en el software Unity3D. Para el procesamiento de las señales gestuales se cuenta con un sistema de clasificación implementado en MatLab para los dos tipos de configuración de la señal: estática y dinámica; se usa correlación si la configuración es estática (sin movimiento) y el algoritmo DTW para configuraciones dinámicas (con movimiento). La comunicación entre Unity3D y MatLab es bidireccional para que los dos entornos puedan leer y escribir datos simultáneamente, en tiempo real utilizando memoria compartida. Esta aplicación desarrolla un uso didáctico con dos juegos, implementados en Unity3D para un solo jugador y multijugador. La interfaz del juego consta de un menú interactivo que permite seleccionar dos juegos los cuales son; fuga de letras que presenta una imagen y una palabra, el objetivo es completar las letras que faltan en la palabra y el juego tres en raya; es un juego por turnos, se debe poner una X o un O, intentando hacer una línea vertical, horizontal o diagonal.

ABSTRACT

In this project of degree the development of a didactic game for the teaching-learning of the basic language of Ecuadorian signs, for people with hearing impairment is presented without discarding that the system can be used by children, youth and adults that have or not this disability and with interest in learning the basic language of Ecuadorian signs. The input device is a Leap Motion, which detects the gestural signals through the recognition of the bones and joints of each finger in the Unity3D software. For the processing of gestural signals, there is a classification system implemented in MatLab for the two types of signal configuration: static and dynamic; Correlation is used if the configuration is static (without movement) and the DTW algorithm for dynamic configurations (with movement). The communication between Unity3D and MatLab is bidirectional so that both environments can read and write data simultaneously in real time using shared memory. This application develops a didactic use with two games, implemented in Unity3D for a single player and multiplayer. The interface of the game consists of an interactive menu that allows to select two games which are; flight of letters that presents an image and a word, the objective is to complete the missing letters in the word and the game three in a row; It is a turn-based game, you must put an X or an O, trying to make a vertical, horizontal or diagonal line.

INTRODUCCIÓN

Los modelos educativos han evolucionado a lo largo de los años, adaptándose a la realidad y necesidad de la sociedad, involucrando así a la tecnología en los procesos de enseñanza aprendizaje a través de aplicaciones multimedia, acceso a información por medio de internet, clases virtuales, canales de comunicación, y más. (Virvou, Katsionis, & Manos , 2005) El creciente mundo tecnológico crea una relación con la educación permitiendo complementar los modelos educativos por medio de la adquisición, producción, y almacenamiento del conocimiento con el objetivo de mejorar el proceso de enseñanza aprendizaje. (Vannatta, Beyerbach, & Walsh, 2001). La incorporación de las Tecnologías de la Información y la Comunicación, TIC, en la educación ha contribuido a potencializar los sistemas educativos brindando mayor flexibilidad y eficiencia a las metodologías de enseñanza aprendizaje, las TIC tienen una influencia cada vez mayor en la formación académica permitiendo el desarrollo de ambientes de integración docente-estudiantes las cuales estimulan la creatividad respetando el ritmo de aprendizaje y capacidades de los estudiante, además de fortalecer el trabajo en grupo (Paula, Israel, & Núñez Govín, 2005). En el proyecto de titulación abarca los siguientes capítulos:

En el capítulo 1, se describe el planteamiento del problema, justificación del proyecto, objetivos, justificación, propuesta de solución y alcance.

En el capítulo 2, se detalla la documentación técnica que fue necesaria para el desarrollo del presente proyecto técnico de titulación para comprender de mejor manera cada etapa del proyecto.

En el capítulo 3, se describe el desarrollo del juego didáctico usando sensores Leap Motion el mismo que permite en tiempo real la interacción e inmersión del usuario con el entorno virtual. La aplicación puede ser subdividida en cinco subprocesos: tratamiento de datos, comunicación bilateral, base de datos, comparador e interfaz de juego.

En el capítulo 4, se detalla las pruebas de la aplicación del juego las cuales se realizaron según los siguientes aspectos: distancia de interacción, reconocimiento de letras, tiempo de respuesta y test de usabilidad del sistema

CAPÍTULO 1

ANTECEDENTES

1.1. Planteamiento del problema

El aprendizaje del lenguaje de señas puede resultar tedioso y presentar ciertas dificultades para los niños en centros educativos, sobre todo al no estar familiarizados con este lenguaje en el país.

La mayoría de niños con deficiencias auditivas provienen de familia oyentes, por ello están relativamente privados del desarrollo de lenguaje y que viven en un ambiente comunicativo menos eficiente, en comparación con los niños oyentes. Esta privación lingüística tiene consecuencias en su desarrollo emocional y social. (INEC, s.f.)

En Ecuador habitan un aproximado de 16 millones de personas. Datos del Ministerio de Salud Pública (MSP) indican que durante 2014 en el país habitan cerca de 400 mil personas con discapacidad. De ellas, 48 mil asumen discapacidad auditiva profunda o total (el 0.3% de la población del país) y 5510 (0.03%) tienen discapacidad del lenguaje, en este grupo estarían las personas que podrían requerir lengua de señas. (CORDICOM, 2015)

1.2. Justificación del proyecto

El lenguaje de señas ayuda a personas con deficiencia auditiva en especial a niños a conocer ideas, sentimientos usando gestos, adicional un juego didáctico al usuario a la práctica sin necesidad de agobiarlo incrementaría su habilidad en el nuevo lenguaje.

Este proyecto propone un sistema basado en un juego didáctico usando sensores gestuales Leap Motion que permitan la interacción de uno y dos usuarios con una interfaz gráfica amigable en Unity3D que ayude y facilite el aprendizaje del lenguaje de señas ecuatoriano a niños con discapacidad auditiva.

Para alcanzar todo esto se utiliza un sensor gestual que permite reconocer el movimiento de las manos y dedos, donde todas las señales que envía el dispositivo serán analizadas y procesadas para interpretar cada una de las posiciones, acciones y

movimientos que se realizaron, las cuales entraran a la interfaz gráfica amigable del juego que puede ser de una o dos personas.

1.3. Objetivos

1.4. Objetivo general

Desarrollar un juego didáctico mediante sensores Leap Motion para estimular el aprendizaje del lenguaje básico de señas ecuatoriano.

1.4.1. Objetivos específicos

- Estudiar el lenguaje de señas que se utilizan en los centros de educación especial del Ecuador para aplicarlo en juegos didácticos.
- Diseñar el sistema en el entorno grafico de programación Unity3D con el tipo de sensores gestuales adecuado al juego didáctico y al procesador para que sea amigable con el usuario.
- Implementar el hardware con el procesador y los sensores para juegos interactivos de 1 y 2 personas.
- Implementar el software para juegos interactivos de 1 y 2 personas con los algoritmos e interfaz amigable con el usuario.
- Validar el juego y verificar el resultado a través de pruebas.

1.5. Propuesta de solución

En el presente proyecto se detalla la aplicación de un sistema basado en un juego didáctico usando sensores gestuales Leap Motion que permitan la interacción entre dos usuarios con una interfaz amigable que ayude y facilite el aprendizaje del lenguaje de señas a niños con discapacidad auditiva.

Los desarrolladores de Leap Motion aseguran que su sistema es doscientas veces más preciso que cualquier otro que podamos encontrar en el mercado, entendemos que apuntan hacia Kinect, y su precio es accesible. Una ventaja es su pequeño tamaño dispositivo donde están los sensores, se conecta vía USB, y no requiere de ningún tipo de calibrado o configuración especial.

1.6. Alcance

El juego es apto para personas que tengan completa movilidad en las manos y todos sus dedos, que no tengan dificultad en realizar el abecedario del lenguaje de señas

incluyendo los números del 1 al 9. El sistema está diseñado para personas con discapacidades auditivas las cuales no poseen problemas de motricidad fina, con el fin de enseñar lúdicamente el lenguaje básico de señas ecuatoriano.

La limitación se encuentra en la distancia de detección debida que el área de detección se estableció como origen frente al sensor Leap Motion a 30cm y a la altura 25cm del touch pad de la computadora portátil y la zona de interacción puede tener variaciones de hasta ± 25 cm en el eje Y (Horizontal), ± 10 cm en el eje Z (Vertical) y 10 cm en el eje X.

CAPÍTULO 2

MARCO CONCEPTUAL

2.1. Lenguaje de señas ecuatoriano

Según (Vicepresidencia, 2014) en el entorno de la discapacidad es tan diverso, que existe algunas discapacidades que no se puede apreciar tal es el caso de la auditiva. Sólo en el momento que nos encontramos con una persona que usa sus manos para comunicarse mediante señas, nos damos cuenta que tiene una discapacidad auditiva.

De acuerdo con (Sordos Ecuatorianos, s.f.) en Ecuador existen dos diccionarios de Lenguaje de Señas que fue publicado por organización de personas con deficiencia auditiva.

El primer, Lenguaje de Señas Guía Básica sobre una Comunicación Especial - Tomo I, el cual se publicó en el año 1987 por la Asociación de Personas Sordas de Pichincha APSOP (en el cual se llamaba Sociedad de Sordos Adultos “Fray Luis Ponce de León”). Este diccionario fue desarrollado mediante la investigación del Proyecto Mano a Mano de la APSOP y financiado por la Fundación Interamericana con sede en los Estados Unidos. El diccionario contiene una recopilación de señas básicas de las ciudades de Quito y Guayaquil.

El segundo fue titulado Diccionario Oficial de la Lengua de Señas Ecuatoriana fue publicado en el año 2012 por la Federación Nacional de Personas Sordas del Ecuador FENASEC. Este diccionario fue el resultado del trabajo cumplido con la colaboración de la USAID y la Vicepresidencia de la República del Ecuador. Este diccionario recopila aproximadamente 4.000 señas y está dividido en dos tomos.

En junio del 2014 el Consejo Nacional de Igualdad de Discapacidades CONADIS en colaboración con FENASEC el Diccionario Virtual de Lengua de Señas Ecuatoriana “Gabriel Román”. Este trabajo, dentro del convenio CONADIS- Universidad Tecnológica Indoamérica recoge alrededor de 5000 palabras del Diccionario Oficial de la Lengua de Señas Ecuatoriana, contiene gráficos y videos explicativos, a través de este se puede observar forma correcta de realizar una seña. (Sordos Ecuatorianos, s.f.)

2.2. Psicología del color en el aprendizaje

El color se percibe en cada objeto que se observa en la vida e incluso en los sueños. Las consideraciones de colores emergen regularmente de la toma de decisiones, el color estimula la vista y apoya el proceso de aislamiento de información, además sirve para la inspiración de la creatividad e incentiva la creación de nuevas ideas.

La percepción del color puede tener múltiples significados, de ahí a que sea importante asociar el color adecuado para potenciar o transmitir una sensación o una idea, ya sea para la creación de un logotipo, una página o web o cualquier otra creación. (Elliot, 2014)

2.2.1. Los colores en el aprendizaje

Las características de los colores estimulan y benefician la retención de la información por más tiempo cuando el color se usa como material educativo. (Elliot, 2014)

- Los colores cálidos: rojo, naranja y amarillo estimulan e incrementan la actividad cerebral.
- Los colores fríos: verde, azul y violeta inducen a la relajación.

2.3. Tecnología gestual

La tecnología gestual en (NMformación, 2015) fue alrededor de pocos años empezó el desarrollo de tecnologías que es orientado a identificar los movimientos físicos de personas y puede realizar acciones en función en base a eso, accediendo a un alto nivel de interacción que nunca se ha realizado.

De igual forma que existen códigos universales para saludar o pedir la hora, el uso de tecnología masiva de tabletas y smartphones origina un avance en el lenguaje gestual. Estos nuevos hábitos se emplean en comandos en diferentes dispositivos digitales y se usan para desechar las barreras de la comunicación. En la comunicación no verbal, el mensaje se puede expresar utilizando gestos, que va creciendo al compás de las nuevas tecnologías. Obviamente, la civilización digital inventa nuevos signos, tendencia que deben ser analizados. (J.M.Nieves, 2014)

En la actualidad los que predominan el mercado de la tecnología gestual es Microsoft y su Kinect, diferentes expertos podrían atreverse a decir que ellos serán los que

controlen esta tecnología en próximos años. Pero por otra parte los especialistas, de la tecnología gestual se verá como algo de Apple y no de Microsoft en el futuro. (Verbal, 2017)

2.4. Leap motion

En (Oleaga, 2014) describe como un pequeño dispositivo que permite controlar el ordenador a base de gestos que se realizan en el aire, se puede usar con los dedos como con toda la mano. Crea un trazo de una imagen virtual de las manos y articulaciones desde la muñeca, y rastrea todos los movimientos. Al ser usado se tendrá la sensación de estar en esa típica escena de película de ciencia ficción donde los protagonistas interactúan con el computador haciendo movimientos en el aire. La gran diferencia es que en ellas siempre hay alguna referencia flotando en el aire, con este sistema se tiene que hacer gestos con la pantalla como única referencia.

Se encuentra acostumbrado a la entrada del computador en dos dimensiones, como el ratón o una pantalla táctil, así que el Leap motion descubre la tercera dimensión, añadiendo mayor profundidad. En la Figura 2.1 se observa el sensor gestual Leap Motion

Figura 2.1 Leap Motion

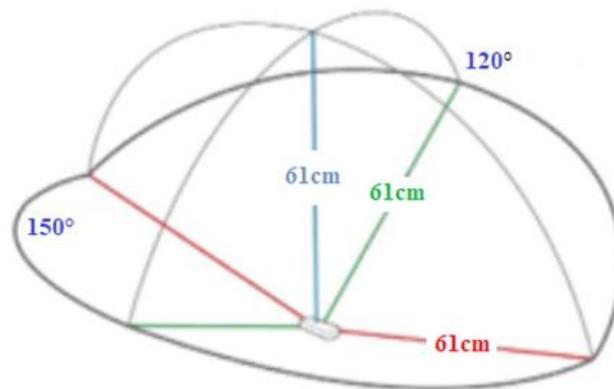


Dispositivo Leap Motion, Fuente: (Belda, 2015)

De acuerdo con (Belda, 2015) este dispositivo posee pequeñas dimensiones en comparación a otros dispositivos que se encuentran en el mercado sus medidas son 75 mm de largo, 25 mm de ancho y 11 mm de alto.

La zona de cobertura del dispositivo, como se puede ver en la imagen 2, es una zona semiesfera de 61 cm de radio. La cual depende del ángulo de visión de los lentes de las cámaras y de la intensidad máxima que puede entregar la conexión USB a los LEDs. A su vez, el ángulo de visión depende de la distancia focal y del tamaño del sensor, ver Figura 2.2

Figura 2.2 Zona de cobertura del dispositivo



Zona de detección del sensor, Fuente: (Belda, 2015)

Sus componentes de funcionamiento se basan de tres partes principales con son dos cámaras, tres LEDs infrarrojos y un microcontrolador como se muestra en la Figura 2.3. A continuación vamos a ver cada parte con un poco más en de detalle: (Belda, 2015)

Figura 2.3 Partes del sensor Leap Motion



Partes principales del sensor Leap Motion, Fuente: Belda, 2015

2.4.1. Cámaras

Las cámaras, son una de las partes más importantes del funcionamiento del dispositivo, debido a que realizan la captura de imágenes y su buen desempeño llevara al correcto funcionamiento del resto de sistema.

Cada cámara que posee el dispositivo cuenta con un sensor monocromático, sensible a la luz infrarroja, con una longitud de onda de 850 nm. Estos sensores pueden trabajar a una velocidad de hasta 200 fps, dependiendo de las características la computadora al que se conecte el dispositivo. (Belda, 2015)

- La digitalización de los píxeles en un sensor CMOS se produce dentro de cada celda, por lo que no es necesario un chip externo como ocurriría en el caso de utilizar sensores CCD. Esto se traduce en mayor velocidad para capturar imágenes y en menor espacio para albergar los sensores.
- Estos sensores son más económicos que los sensores CCD.
- En este tipo de sensor no se produce el fenómeno blooming, al contrario que en los sensores CCD. Este fenómeno se produce cuando una celda se satura de luz y hace que las celdas de alrededor también se saturen.
- La lectura simultánea de celdas en los CMOS es mayor que en los CCD.
- El consumo eléctrico de los CMOS es menor que el de los CCD.

2.4.2. Iluminación infrarroja

El dispositivo posee 3 LEDs que se encuentran distribuidos en los extremos y uno en la mitad, la cual ilumina la zona de cobertura por inundación. Trabajan en el espectro de luz infrarroja a una longitud de onda de 850nm la cual es la misma a la que son sensibles los sensores ópticos. Varían su consumo eléctrico y por tanto la iluminación dependiendo de la luz que haya en la zona de cobertura para asegurar una misma resolución de imagen.

Como se puede observar en la imagen 3, los LEDs están separados por pequeñas barreras de plástico. De esta manera se asegura que la iluminación sea uniforme en toda la zona de cobertura. Además, se protege a los sensores ópticos de una posible saturación de luz, dado que de esta manera la luz infrarroja no les ilumina directamente. (Belda, 2015)

2.4.3. El microcontrolador

Se trata de un circuito integrado en este caso contiene el programa que controla todo el dispositivo para regular la iluminación y se encarga de recoger la información de los sensores para luego enviarla al controlador instalado en el computador. (Belda, 2015)

2.5. Unity3D

Unity3D es un motor de videojuegos multiplataforma creada por Unity3D Technologies, En particular, se encuentra disponible para Windows y OS X, y permite crear juegos para las plataformas: Windows, OS X, Linux, Xbox 360, PlayStation 3, PlayStation Vita, Wii, Wii U, iPad, iPhone, Android y Windows Phone y WebGL. El motor de juegos 3D proporcionan herramientas de navegación 3D estándar, así como también cualquier vista programable para crear una navegación atractiva a través del entorno virtual. Al usar un motor de juego, es posible crear otras interacciones, como la manipulación de objetos, la interacción de personajes que no juegan (NPC) con el jugador y / o el entorno. Se lleva a cabo análisis de motores de juegos anteriores y experimentamos en proyectos de diseño urbano con el motor de juegos Unity3D para visualización e interactividad. (Schmitz B. T., 2013)

2.6. Matlab

Matlab es un software matemático usado para realizar cálculos numéricos con vectores y matrices. Puede trabajar con números escalares, tanto reales como complejos. MATLAB combina un entorno de escritorio perfeccionado para el análisis iterativo y los procesos de diseño con un lenguaje de programación que expresa las matemáticas de matrices y arrays directamente. Una de las capacidades más atractivo de Matlab es la posibilidad de realizar una amplia variedad de gráficos en dos, tres dimensiones, así como usa diversos toolboxes para diferentes áreas. Las apps de MATLAB permiten ver cómo funcionan diferentes algoritmos con sus datos. Realiza iteraciones hasta obtener los resultados deseados y después genera automáticamente un programa de MATLAB para reproducir o automatizar su trabajo. (MathWorks, Inc, 2017)

2.7. Estado del arte

En la literatura se puede encontrar diferentes propuestas orientadas a la detección de señas aplicada al lenguaje de señas básico propio de cada región.

En (Potter, 2013) artículo presenta una revisión de la capacidad del sensor LeapMotion para realizar el rastreo del lenguaje de señas de Australia. Las pruebas mostraron que el controlador es capaz de proporcionar un seguimiento preciso de las manos y dedos, y para rastrear el movimiento. Esta detección pierde precisión cuando la mano se mueve a una posición que obstruye la capacidad del controlador para ver, como cuando la mano gira y es perpendicular al controlador. La detección también falla cuando los elementos individuales de las manos se unen, como dedo a dedo. En ambos casos, el controlador no puede leer o rastrear la mano. Existe el potencial para el uso de esta tecnología para reconocer lenguaje de señas de Australia, sin embargo, más lejos se requiere el desarrollo de la API de Leap Motion.

En (Mohandes, 2014) se propone el uso de Leap Motion para el reconocimiento de lenguaje de señas arábigo utilizando una etapa de extracción de características, procesamiento y clasificación. El traductor se basa en Hidden Modelos de Markov (HMM). El sistema fue entrenado para reconocer 40 signos de lenguaje de señas árabe estándar. Se presenta un esquema de go-stop para manejar secuencias de signos que construyen oraciones en tiempo real. La tasa de éxito de reconocimiento basada en la nueva metodología es por encima del 90 por ciento con rendimiento en tiempo real en una PC.

En (Chuan, 2014) se muestra el desarrollo de un sistema de detección de las 26 letras del alfabeto inglés en lenguaje de señas americano usando el sensor Leap Motion del tamaño de la palma mano que proporciona una solución mucho más portátil y económica que Cyblerglove o Microsoft kinect utilizado en estudios existentes. Aplicando the k-nearest neighbor and support vector machine El resultado del experimento muestra que la tasa promedio de clasificación más alta es del 72.78% y 79.83% fue logrado por el vecino más cercano y el vector de soporte k máquina respectivamente. También proporcionamos discusiones detalladas sobre configuración de parámetros en métodos de aprendizaje automático y precisión de letras del alfabeto específicas en este documento.

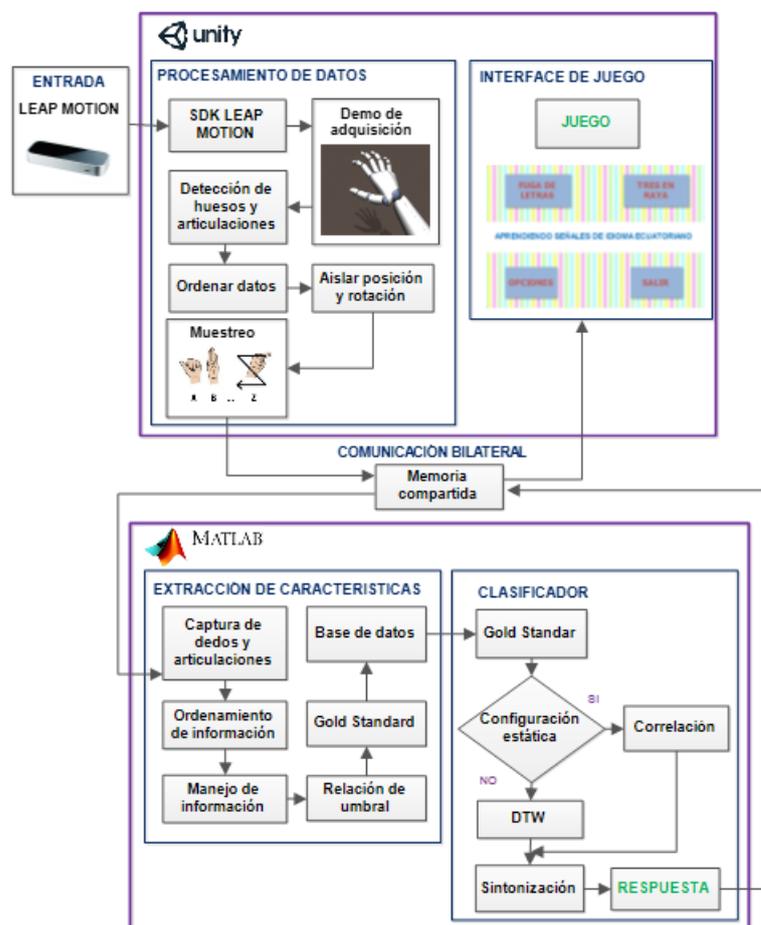
En (Jiménez, 2017) propone un sistema inteligente para traducir el lenguaje de señas a texto. Este enfoque consiste en hardware y software. El hardware está formado por sensores de flexión, contacto e inerciales montados en un guante de nylon de poliéster, el software consiste en un algoritmo de clasificación basado en los k vecinos más cercanos, árboles de decisión y los algoritmos dinámicos de deformación de tiempo. El sistema propuesto es capaz de reconocer los gestos estáticos y dinámicos. Este sistema puede aprender a clasificar los patrones de gestos específicos de cualquier persona. El sistema propuesto fue probado al traducir 61 letras, números y palabras del lenguaje de signos ecuatoriano. Los resultados experimentales demuestran que nuestro sistema tiene una precisión de clasificación del 91.55%. Este resultado es una mejora significativa en comparación con los resultados obtenidos en trabajos relacionados anteriores

CAPÍTULO 3 DISEÑO E IMPLEMENTACIÓN

En el presente capítulo se va desarrollar el juego didáctico usando sensores gestuales para el rastreo de los dedos de las manos a fin de que permitan la interacción entre uno o dos usuarios con una interfaz inmersiva que ayude y facilite el proceso de enseñanza-aprendizaje del lenguaje a personas con discapacidad auditiva. La Figura 3.1 muestra el esquema del desarrollado el mismo que permite en tiempo real la interacción e inmersión del usuario con el entorno virtual 3D.

La aplicación desarrollada puede ser subdividida en cinco subprocesos: Procesamiento de datos, comunicación bilateral, extracción de características, clasificador e interfaz de juego.

Figura 3.1 Desarrollo del funcionamiento del sistema



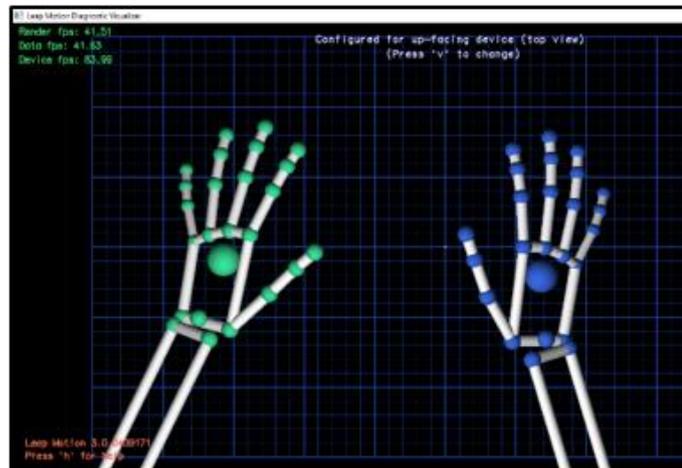
Esquema del desarrollo del sistema, Fuente: Mateo Parreño

3.1. Procesamiento de datos

Para poder implementar el sistema se realiza el tratamiento de datos se considera como entrada el sensado de la posición y orientación de los dedos de la mano del usuario, datos que son procesados en tiempo real a fin de interactuar con el entorno virtual. El rastreo de los dedos se realiza a través del sensor gestual Leap Motion a través de dos cámaras infrarrojas en conjunto con las interfaces de programación de aplicaciones (API) oficiales del dispositivo que además son compatibles con lenguajes de programación por ejemplo C++, C#, Java, Python, etc. (Leap Motion SDK Documentation, 2017). (Guna, 2014)

En la Figura 3.2 se presenta el visualizador del demo de SDK de Leap Motion en el entorno de Unity3D.

Figura 3.2 Visualizador SDK Leap Motion

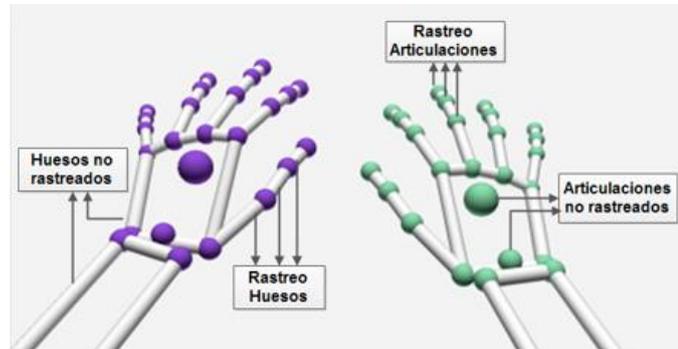


Visualización de las manos en el SDK de Leap Motion, Fuente: Mateo Parreño

El rastreo de la mano a través del dispositivo Leap Motion, instalado en el motor gráfico Unity3D, permite detectar el movimiento de la mano y localizar los hueso y articulaciones de cada uno de los dedos a fin de ordenar de la siguiente manera. Se realiza el rastreo de 3 huesos y 3 articulaciones por cada dedo de cada mano, dando lugar un total 60 puntos de interés; Además existen huesos y articulaciones que detecta el dispositivo pero que no se utilizan en el algoritmo de detección de señales.

En la Figura 3.3 se observar la selección de rastreo de los huesos y articulaciones mediante el Leap Motion.

Figura 3.3 Rastreo de manos con Leap Motion



Rastreo de los puntos de interés hueso y articulaciones, Fuente: Mateo Parreño

Para la toma de muestras se cuenta con una interfaz gráfica en Unity3D que permite visualizar y capturar la posición y rotación de huesos y articulaciones de los dedos de la mano involucrada mientras el usuario realiza una determinada señal. En la Figura 3.4 se observa la lectura del rastreo de las manos mediante de Unity3D

Figura 3.4 Lectura del rastreo de manos



Rastreo de las manos en Unity3D, Fuente: Mateo Parreño

3.2. Comunicación bilateral

El algoritmo de clasificación de señales gestuales se implementa en el software Matlab, en tiempo real. Matlab considera como entradas los datos de rastreo obtenidos desde el motor gráfico Unity3D y da como salida la identificación del

gentos realizado. La comunicación entre el motor gráfico Unity3D y Matlab es bidireccional, que los dos entornos puedan leer y escribir datos simultáneamente.

A fin de cumplir este requerimiento se implementa el uso de la memoria compartida. De acuerdo con (Andaluz, 2016). La Memoria Compartida (SM) es realizada por una Biblioteca de Enlace Dinámico o llamada .dll. Este método gestiona el espacio SM, proporciona permisos para las aplicaciones, etiqueta el espacio RAM de memoria, proporciona funciones para obtener información compartida y espacio libre cuando se cierra la aplicación.

En la Figura 3.5 se observa a la memoria compartida, se puede crear n número de memorias compartidas, con la opción de determinar tamaño, y tipo de datos de cada memoria.

Figura 3.5 Interfaz de la memoria compartida

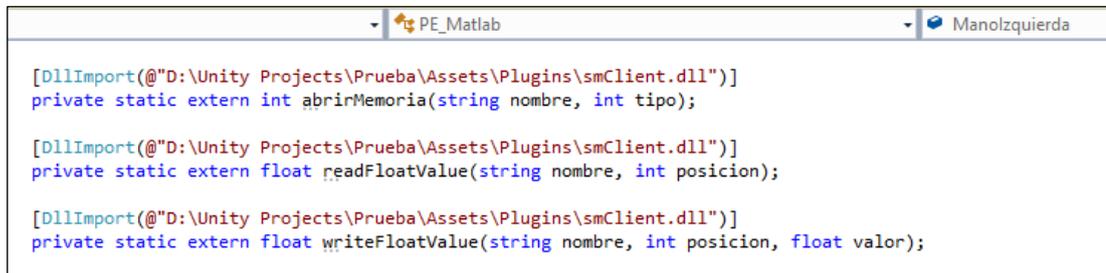


Creación de las memorias que se utiliza para la comunicación, Fuente: Mateo Parreño

Para comunicar Unity3D con Matlab es necesario crear un Script con la librería dynamic-link-library, dll, en el que se implementa el método de memoria compartida, SM, en RAM. En la Figura 3.6 ilustra la implementación de la memoria compartida, donde el dll gestiona el espacio SM, además de proporcionar permisos

para las aplicaciones, etiquetar el espacio de memoria, proporcionar funciones para modificar/obtener la información almacenada y liberar el espacio cuando la aplicación finaliza (Andaluz, 2016)

Figura 3.6 Script de Unity3D Memoria compartida



```
[DllImport(@"D:\Unity Projects\Prueba\Assets\Plugins\smClient.dll")]
private static extern int abrirMemoria(string nombre, int tipo);

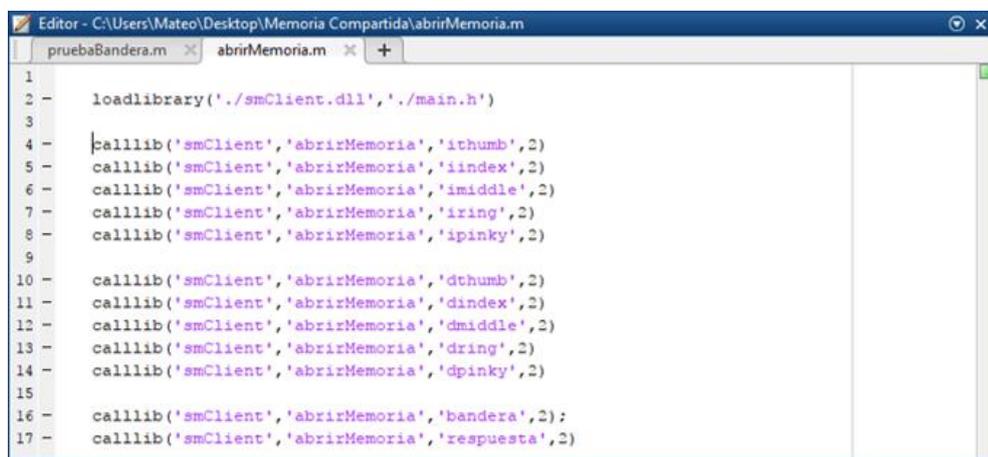
[DllImport(@"D:\Unity Projects\Prueba\Assets\Plugins\smClient.dll")]
private static extern float readFloatValue(string nombre, int posicion);

[DllImport(@"D:\Unity Projects\Prueba\Assets\Plugins\smClient.dll")]
private static extern float writeFloatValue(string nombre, int posicion, float valor);
```

Código del Script de Unity3D para hacer la comunicación, Fuente: Mateo Parreño

El uso del dll entre Matlab y Unity3D, el dll se puede instanciar desde una aplicación a través de un controlador, en el que se establecen los atributos de seguridad y herencia, permisos para leer/escribir en registros de memoria reservado, gestión del espacio de RAM y etiquetado. Las características de la dll permiten que la memoria generada pueda ser inicializado desde Matlab o Unity3D. A partir de este paso, ambas aplicaciones deben usar funciones para identificar espacios de memoria dedicados, modificar datos u obtenerlos. En la Figura 3.7 se observa el script en Matlab para el envío y recepción (Andaluz, 2016)

Figura 3.7 Script de Matlab Memoria compartida



```
1
2 - loadlibrary('./smClient.dll', './main.h')
3
4 - calllib('smClient', 'abrirMemoria', 'ithumb', 2)
5 - calllib('smClient', 'abrirMemoria', 'iindex', 2)
6 - calllib('smClient', 'abrirMemoria', 'imiddle', 2)
7 - calllib('smClient', 'abrirMemoria', 'iring', 2)
8 - calllib('smClient', 'abrirMemoria', 'ipinky', 2)
9
10 - calllib('smClient', 'abrirMemoria', 'dthumb', 2)
11 - calllib('smClient', 'abrirMemoria', 'dindex', 2)
12 - calllib('smClient', 'abrirMemoria', 'dmiddle', 2)
13 - calllib('smClient', 'abrirMemoria', 'dring', 2)
14 - calllib('smClient', 'abrirMemoria', 'dpinky', 2)
15
16 - calllib('smClient', 'abrirMemoria', 'bandera', 2);
17 - calllib('smClient', 'abrirMemoria', 'respuesta', 2)
```

Código del Script de Matlab para hacer la comunicación, Fuente: Mateo Parreño

3.3. Extracción de características

Los valores de las coordenadas de posición y rotación de las señales gestuales para referencia se obtienen a través del dispositivo Leap Motion. Los datos son agrupados en una matriz simétrica de $R \times n$ que contiene los valores de cada hueso de los dedos y articulaciones; donde $m=15$ representa los dedos: pulgar, índice, medio, anular y meñique, de los cuales cada uno se divide en 3 partes distal, medial y proximal los cuales están abreviados por las letras respectivamente; $n=12$ representa las coordenadas (X, Y, Z) en el espacio de las características de posición y rotación de huesos y articulaciones.

En la Tabla 3.1 se observa cómo se realiza el envío de datos mediante una matriz de $n \times m$.

Tabla 3.1 Matriz $R \times n$

Mano Derecha / Izquierda													
Dedos		Huesos						Articulaciones					
		Posición			Rotación			Posición			Rotación		
		x	y	z	x	y	z	x	y	z	x	y	z
Pulgar	ρ	1	2	3	4	5	6	7	8	9	10	11	12
	σ	2											
	τ	3											
Índice	ρ	4											
	σ	5											
	τ	6											
Medio	ρ	7											
	σ	8											
	τ	9											
Anular	ρ	10											
	σ	11											
	τ	12											
Meñique	ρ	13											
	σ	14											
	τ	15											

Matriz de datos de envío de la mano, Fuente: Mateo Parreño

Mediante la matriz de extracción de características se reordena los datos para crear un vector de 180 posiciones en el cual se va ordenando los valores de pulgar, índice, medio, anular y meñique, de los cuales cada uno se divide en 3 partes distal, medial y proximal en posición y rotación en los hueso y articulaciones.

En la Figura 3.8. se observa el vector reordenado a partir de la matriz.

Figura 3.8 Reordenamiento de vector



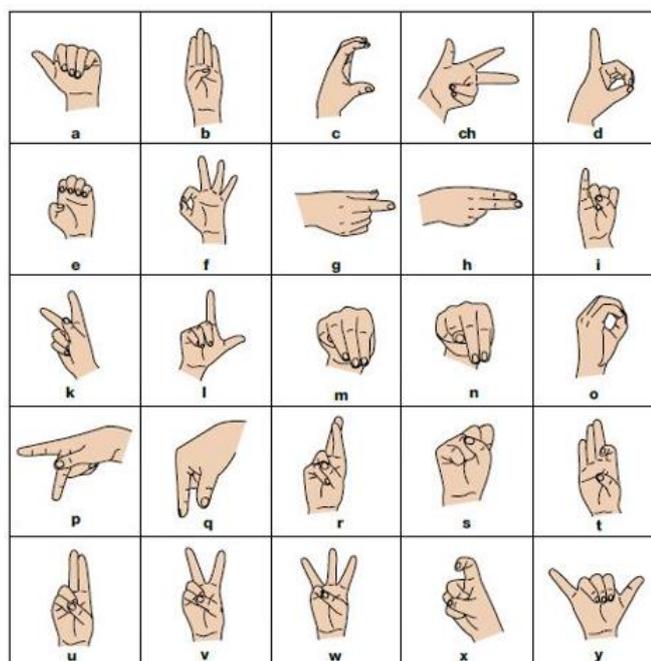
```
1
2 - loadlibrary('./smClient.dll','./main.h')
3 |
4 - calllib('smClient','abrirMemoria','ithumb',2)
5 - calllib('smClient','abrirMemoria','iindex',2)
6 - calllib('smClient','abrirMemoria','imiddle',2)
7 - calllib('smClient','abrirMemoria','iring',2)
8 - calllib('smClient','abrirMemoria','ipinky',2)
9
10 - calllib('smClient','abrirMemoria','dthumb',2)
11 - calllib('smClient','abrirMemoria','dindex',2)
12 - calllib('smClient','abrirMemoria','dmiddle',2)
13 - calllib('smClient','abrirMemoria','dring',2)
14 - calllib('smClient','abrirMemoria','dpinky',2)
15
```

Reordenamiento de datos en vector, Fuente: Mateo Parreño

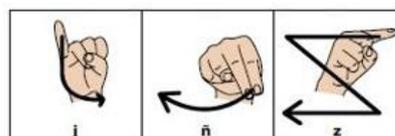
Para realizar la recolección de características de cada letra del alfabeto de señas, es necesario considerar que el lenguaje de señas se divide según su tipo de ejecución:

En la Figura 3.9 (b) se observa los signos dinámicos, son aquellos que poseen movimiento para expresar el gesto (j, ll, ñ, rr and z) y en la Figura 3.10(b) se observa los signos estáticos (restantes del alfabeto), las que carecen de movimiento para efectuar la señal. (CONADIS, 2016)

Figura 3.9 Señas del lenguaje básico de señas configuración



a) Configuración estática



b) Configuración dinámica

Configuración del lenguaje de señas estáticas y dinámicas, Fuente: Mateo Parreño

Para obtener la característica de referencia de letras de configuración estática, se recolecta 9 muestras, con las que se realiza la correlación entre sí. Cada resultado debe ser mayor a 0.8 para considerarse una representación válida de la letra, Tabla 3.2 muestra un ejemplo de la matriz con valores validos de correlación para la letra B.

Tabla 3.2 Valores de correlación válidos de la letra B

	B1	B2	B3	B4	B5	B6	B7	B8	B9
B1	1	0.9996	0.9994	0.9355	0.9355	0.9688	0.9851	0.9771	0.9862
B2	0.9996	1	0.9997	0.9349	0.9969	0.9718	0.9858	0.9790	0.9880
B3	0.9994	0.9997	1	0.9356	0.9977	0.9691	0.9859	0.9763	0.9869
B4	0.9355	0.9349	0.9356	1	0.9381	0.8992	0.9257	0.9039	0.9194
B5	0.9975	0.9969	0.9977	0.9381	1	0.9599	0.9886	0.9648	0.9807
B6	0.9688	0.9718	0.9691	0.8992	0.9599	1	0.9553	0.9948	0.9935
B7	0.9851	0.9858	0.9859	0.9257	0.9886	0.9553	1	0.9558	0.9703
B8	0.9771	0.9790	0.9763	0.9039	0.9648	0.9948	0.9558	1	0.9925
B9	0.9862	0.9880	0.9869	0.9194	0.9807	0.9935	0.9703	0.9925	1

La tabla de correlación mayores de 0.8 para hacer valido una muestra, Fuente:

Caso contrario se realiza la toma de muestras nuevamente debido que se considera muestras basura. Tabla 3.3 muestra un ejemplo de la matriz con valores no validos de correlación para la letra C.

Tabla 3.3 Valores de correlación no validos de la letra C

	C1	C2	C3	C4	C5	C6	C7	C8	C9
C1	1	0.9172	0.9993	0.7766	0.8050	0.9221	0.9990	0.9984	0.8262
C2	0.9172	1	0.9157	0.7416	0.7403	0.8840	0.9174	0.9177	0.7761
C3	0.9993	0.9157	1	0.7802	0.8083	0.9224	0.9994	0.9994	0.8278
C4	0.7766	0.7416	0.7802	1	0.7380	0.8599	0.7823	0.7743	0.8808
C5	0.8050	0.7403	0.8083	0.6380	1	0.7394	0.8079	0.8106	0.7005
C6	0.9221	0.8840	0.9224	0.8599	0.7394	1	0.9225	0.9226	0.7679
C7	0.9990	0.9174	0.9994	0.7823	0.8079	0.9225	1	0.9991	0.8286
C8	0.9984	0.9177	0.9994	0.7743	0.8106	0.9226	0.9991	1	0.8273
C9	0.8262	0.7761	0.8278	0.8808	0.7005	0.7679	0.8286	0.8273	1

La tabla de correlación menores de 0.8 que no son válidos, Fuente: Mateo Parreño

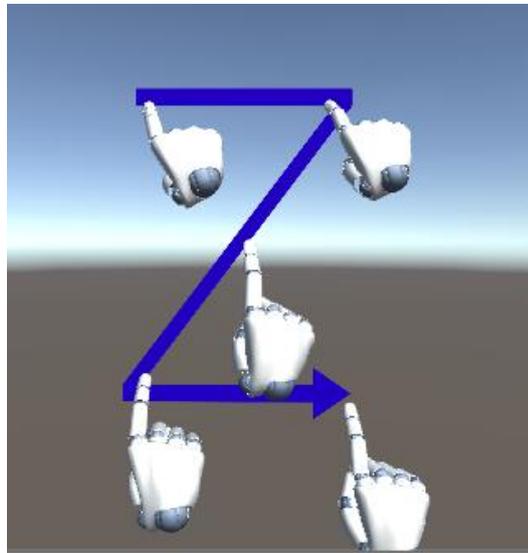
La fase final de la extracción de características consiste en aplicar el promedio GoldStandar con los valores de correlación de la Tabla 3.2. Este método es una tendencia al punto de referencia, debido a que es la mejor muestra y sirve como base de comparación, se obtiene mediante el promedio de las 9 muestras validas (Versi, 1992)

Para obtener la característica de referencia de letras de configuración dinámica, se realiza el muestreo de toda la trayectoria con un total de 5 muestras, la primera muestra va a ser el inicio y la quinta vá ser el final de la secuencia; para completar la

trayectoria de la letra Z, para al final obtener una matriz de 12 x 60 que contengan la 5 muestras en secuencia para crear la trayectoria final.

En la Figura 3.10 se observa estroboscopia de la seña dinámica de la letra Z.

Figura 3.10 Imagen estroboscópica de seña dinámica



Toma de muestra de 5 posiciones que trazan la letra z, Fuente: Mateo Parreño

3.4. Clasificador

En esta subsección la información entregada por la base de datos se analiza a fin de determinar si el gesto realizado con la mano del usuario es similar a la señal de referencia. Se implementan dos procesos en tiempo real de acuerdo con la configuración de cada letra como son: letras estáticas y letras dinámicas.

3.4.1. Configuraciones estáticas

El clasificador apropiado para las letras estáticas es la correlación, ya que determina la relación que existe entre dos variables.

a. Coeficiente de correlación

El coeficiente de correlación es un método estadístico muy utilizado para resumir datos de investigación poniendo énfasis en la relación, determina si existe relación y que tan fuerte es la relación entre dos variables. (Taylor, 1990)

La repuesta de correlación se determina según las siguientes condiciones:

- Si $r = 1$ es correlación positiva perfecta lo que supone una determinación absoluta entre las dos variables, en sentido directo: Existe una relación lineal perfecta con pendiente positiva).
- Si $r > 0$ existe correlación positiva: las dos variables se correlacionan en sentido directo. A valores altos de una le corresponden valores altos de la otra e igualmente con los valores bajos.
- Si $r = 0$ las variables están incorrelacionadas: no establece ningún sentido de relación entre las variables.
- Si $r < 0$ existe correlación negativa: Las dos variables se correlacionan en sentido inverso. A valores altos de una de ellas le suelen corresponder valores bajos de la otra y viceversa.
- Si $r = -1$ es correlación negativa perfecta: Lo que supone una determinación absoluta entre las dos variables, en sentido inverso: Existe una relación funcional perfecta entre ambas, una relación lineal de pendiente negativa.

b. Sintonización configuración estáticas

Para la prueba se sintonización identifica la letra que se realiza, la coincidencia de interés al definir un valor de relación entre el factor GoldStandar y la letra o número que realiza el usuario.

En la Tabla 3.4 se observa los valores de umbral de cada letra estática para que tenga una sintonización fina.

Tabla 3.4 Sintonización de configuración estática

Configuración Estática de Letras	Umbral
A	0.96
B	0.96
C	0.95
CH	0.93
D	0.98
E	0.95
F	0.90
G	0.95
H	0.85
I	0.90
K	0.92
L	0.93
M	0.92
N	0.93
O	0.92
P	0.87
Q	0.91
R	0.86
S	0.96
T	0.91
U	0.87
V	0.94
W	0.88
X	0.87
Y	0.92

Tabla de valores de la sintonización de las letras de configuración estáticas, Fuente:

En la Tabla 3.5 se observa los valores de umbral de cada número de configuración estática

Tabla 3.5 Sintonización de configuración estática

Configuración	Umbral
1	0.92
2	0.97
3	0.97
4	0.98
5	0.96
6	0.91
7	0.97
8	0.98
9	0.91

Tabla de valores de la sintonización de las letras de configuración estáticas, Fuente: Mateo Parreño

3.4.2. Configuraciones dinámicas

Hace referencia a las letras del alfabeto tiene movimiento por lo cual no es adecuado implementar la correlación entre gestos, ya que presenta dificultades al momento de comparar varios puntos que forman una trayectoria. A fin de resolver este problema se implementa el algoritmo de Alineamiento Temporal Dinámico DTW.

a. DTW (alineamiento temporal dinámico)

El algoritmo DTW nos entrega una medida de distancia que compara dos series temporales después de alinearlas de manera óptima, lo cual permite medir la similitud entre dos secuencias que pueden variar en el tiempo y el espacio.

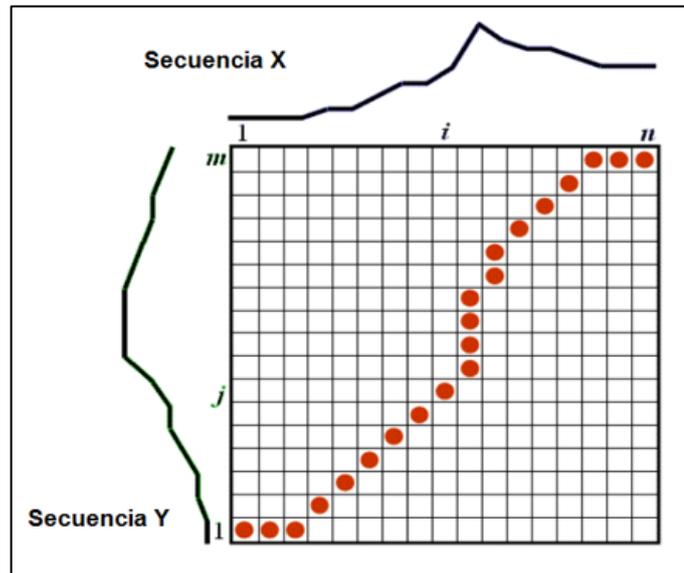
El algoritmo tiene como objetivo alinear dos secuencias de vectores de características mediante la deformación repetitiva del eje del tiempo hasta se encuentra una coincidencia óptima entre las dos secuencias. Este algoritmo realiza un mapeo lineal por piezas del tiempo eje para alinear ambas señales.

Considere dos secuencias de vectores de características en un espacio n-dimensional.

$$x = [x_1, x_2, \dots, x_n] \text{ and } y = [y_1, y_2, \dots, y_n] \quad \text{Ec.(3. 1)}$$

Las dos secuencias están alineadas en los lados de una cuadrícula, con una en la parte superior y otra en el lado izquierdo. Ambas las secuencias comienzan en la parte inferior izquierda de la cuadrícula, ver Figura 3.11

Figura 3.11 Cuadrícula de distancia global



Distancia global para la mejor secuencia, Fuente: (Dhingra, 2013)

En cada celda, se coloca una medida de distancia, que compara los elementos correspondientes de las dos secuencias. La distancia entre los dos puntos se calcula a través de la distancia euclidiana.

$$Dist(x, y) = |x, y| = [(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2]^{1/2} \quad \text{Ec.(3. 2)}$$

La mejor coincidencia o alineación entre estas dos secuencias es la ruta a través de la cuadrícula, que minimiza la total distancia entre ellos, que se denomina Distancia global, encontrando y pasando por todas las rutas posibles a través de la cuadrícula, cada uno calcula la distancia total. La distancia global es el mínimo de la suma de las distancias (distancia euclidiana) entre los elementos individuales en el camino dividido por la suma de la función de ponderación. (Dhingra, 2013)

DTW permite medir la similitud entre dos secuencias que pueden variar en el tiempo y el espacio. Al momento de realizar un gesto no siempre se hace un movimiento de

la misma duración y exacta trayectoria como la señal patrón, por lo que medir la homogeneidad de la información en las secuencias del gesto se hace útil en las configuraciones dinámicas (Mueen, 2016)

- Si el valor es pequeño, entonces existe gran similitud.
- Si el valor es grande, entonces no existe similitud.

La identificación de la letra se realiza mediante sintonización, la coincidencia de interés al definir un valor de relación entre el factor GoldStandard y la letra que realiza el usuario.

b. Sintonización configuración dinámicas

Para el caso de DTW la sintonización se realiza para cada configuración dinámica proporcionando una distancia de umbral en donde la señal será detectada sin mayor dificultad por el algoritmo.

En la Tabla 3.6 se observa los valores de umbral que se encuentra las letras de configuración dinámica.

Tabla 3.6 Sintonización configuración dinámica

Configuración Dinámica	Umbral
J	Entre 70 a 80
Ñ	Entre 42 a 45
Z	Entre 35 a 42

Tabla de valores de la sintonización de las letras de configuración dinámica, Fuente: Mateo Parreño

3.5. Interfaz de juego

El desarrollo de la interfaz de juego considera la interacción e inmersión entre el usuario y el entorno virtual 3D. La inmersión es un proceso psicológico que es provocado cuando la persona abandona su forma de percepción clara de su medio natural al colocar toda su atención en un objeto, narración o imagen que sumerge en un medio artificial, en este caso en un juego. La aplicación desarrollada considera un menú principal que permite acceder a los juegos y opciones de usuario, según

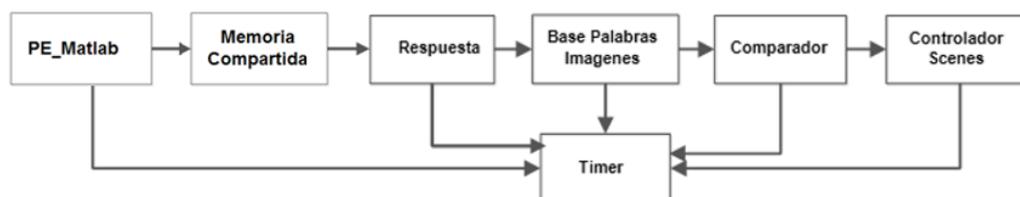
Figura 3.13 Juego Fuga de letras



Entorno grafico del juego de fuga de letras, Fuente: Mateo Parreño

Para el desarrollo del juego fuga de letras se crea los siguientes Script. En la Figura 3.14 se muestra el esquema del funcionamiento.

Figura 3.14 Desarrollo juego fuga de letras



Esquema de funcionamiento de los scripts del juego fuga de letras, Fuente: Mateo Parreño

- Script PE_Matlab_D

Se encarga de detectar la presencia de las manos derecha y/o mano izquierda y envía los datos a la memoria compartida como se observa en la Figura 3.15.

Figura 3.15 Script PE_Matlab_D

```
PE_Matlab_D Update()
//-----Thumb Derecha-----
DPosicionThumbBone1 = ManoDerecha.transform.GetChild(5).GetChild(0).position;
DPosicionThumbBone2 = ManoDerecha.transform.GetChild(5).GetChild(1).position;
DPosicionThumbBone3 = ManoDerecha.transform.GetChild(5).GetChild(2).position;
DRotacionThumbBone1 = ManoDerecha.transform.GetChild(5).GetChild(0).eulerAngles - new Vector3(360, 360, 360);
DRotacionThumbBone2 = ManoDerecha.transform.GetChild(5).GetChild(1).eulerAngles - new Vector3(360, 360, 360);
DRotacionThumbBone3 = ManoDerecha.transform.GetChild(5).GetChild(2).eulerAngles - new Vector3(360, 360, 360);

DPosicionThumbJoint3 = ManoDerecha.transform.GetChild(5).GetChild(3).position;
DRotacionThumbJoint3 = ManoDerecha.transform.GetChild(5).GetChild(3).eulerAngles - new Vector3(360, 360, 360);
DPosicionThumbJoint2 = ManoDerecha.transform.GetChild(5).GetChild(4).position;
DRotacionThumbJoint2 = ManoDerecha.transform.GetChild(5).GetChild(4).eulerAngles - new Vector3(360, 360, 360);
DPosicionThumbJoint1 = ManoDerecha.transform.GetChild(5).GetChild(5).position;
DRotacionThumbJoint1 = ManoDerecha.transform.GetChild(5).GetChild(5).eulerAngles - new Vector3(360, 360, 360);
// Bone Thumb
writeFloatValue("dthumb", 1, DPosicionThumbBone3.x);//Posicion
writeFloatValue("dthumb", 2, DPosicionThumbBone3.y);
writeFloatValue("dthumb", 3, DPosicionThumbBone3.z);
writeFloatValue("dthumb", 4, DRotacionThumbBone3.x);//Rotacion
writeFloatValue("dthumb", 5, DRotacionThumbBone3.y);
writeFloatValue("dthumb", 6, DRotacionThumbBone3.z);
```

Script de la detección de la mano derecha y envío de datos, Fuente: Mateo Parreño

- Script Respuesta

Lee los datos de ubicación en memoria compartida en la cuales Matlab escribe la letra o número detectado como se muestra en la Figura 3.16.

Figura 3.16 Script Respuesta

```
Respuesta
switch (numeroLetra)
{
    case 1:
        letra = "A";
        break;
    case 2:
        letra = "B";
        break;
    case 3:
        letra = "C";
        break;
    case 4:
        letra = "CH";
        break;
    case 5:
        letra = "D";
        break;
    case 6:
        letra = "E";
        break;
}
```

Script de respuesta que envía datos desde Matlab por la memoria compartida a Unity3D, Fuente: Mateo Parreño

- Script BasePalabrasImágenes

En este script está guardado la base de palabras y las imágenes que se va desplegar en cada intento, además de una función que elige las palabras aleatoriamente y la complementa con la función que elimina aleatoriamente las letras de las palabras que se presentaran como las faltantes como se muestra en la Figura 3.17

Figura 3.17 Script BasePalabrasImágenes



```
public class BasePalabrasImágenes : MonoBehaviour {
    public string[] basePalabras = new string[] { "",
        /*1*/"CARRO",
        /*2*/"ESCUELA",
        /*3*/"CUADERNO",
        /*4*/"BALLENA",
        /*5*/"PELOTA",
        /*6*/"MOCHILA",
        /*7*/"BICICLETA",
        /*8*/"GATO",
        /*9*/"PATINETA",
        /*10*/"CHOCOLATE",
        /*11*/"PANTALON",
        /*12*/"ABUELO",
        /*13*/"HELADO",
        /*14*/"ZAPATOS",
    };
}
```

Script del base de las palabras e imágenes del juego fuga de letras, Fuente: Mateo Parreño

- Script Comparador

Compara las letras ingresadas con las letras faltante de las palabras a completar además otorga un puntaje al completar cada palabra como se observa en la Figura 3.18

Figura 3.18 Script Comparador



```
public void unir()
{
    estado.text = "";
    for(int i = 0; i < palabraPorCompletar.Length; i++)
    {
        estado.text = estado.text + palabraPorCompletar[i];
    }
}

// Use this for initialization
void Start () {
    inicio = false;
    auxVerificar = true;
}

// Update is called once per frame
void Update () {
    letraDetectada = contenedor.gameObject.GetComponent<Respuesta>().letra;

    palabraVerdadera = gameObject.GetComponent<BasePalabrasImágenes>().palabraActual;
    palabraOriginal = palabraVerdadera.ToCharArray();

    palabraPorCompletar = gameObject.GetComponent<BasePalabrasImágenes>().palabraRecortada;
    //letraRecibida = contenedor2.GetComponent<AdminLetras>().letraRecibida;
    letraRecibida = Convert.ToChar(contenedor2.GetComponent<Respuesta>().letra);
}
```

Script del algoritmo de comparador, Fuente: Mateo Parreño

- Script ControladorScenes

Administra la interfaz de usuario para los botones de atrás y la selección de juegos como se muestra en la Figura 3.19

Figura 3.19 Script ControladorScenes



```
using UnityEngine.SceneManagement;

public class ControladorScenes : MonoBehaviour {

    // Use this for initialization
    void Start () {

    }

    // Update is called once per frame
    void Update () {

    }

    public void atras()
    {
        SceneManager.LoadScene("Home");
    }

    public void MissingLetters()
    {
        SceneManager.LoadScene("MissingLetters");
    }

    public void TicTacToe()
    {
        SceneManager.LoadScene("TicTacToe");
    }
}
```

Script de las funcione para controlar las Scenes, Fuente: Mateo Parreño

- Script Timer

El presente script se encarga de generar el contador de tiempo además se detiene en un tiempo límite especificado y tiene la función de iniciar y reiniciar el contador de tiempo como se observa en la Figura 3.20

Figura 3.20 Script Timer



```
public void activar()
{
    botonReintentar.SetActive(false);
    comenzar = true;
    indicadorFinal.text = "";
    indicadorFinal.gameObject.SetActive(false);
    segundos = 0;
    minutos = 0;
    gameObject.GetComponent<Comparador>().puntaje = 0;
}

public void desactivar()
{
    comenzar = false;
    indicadorFinal.text = "Puntaje Final: " + gameObject.GetComponent<Comparador>().puntaje;
    indicadorFinal.gameObject.SetActive(true);
    textoDinamico.text = "Reintentar";
    botonReintentar.SetActive(true);
}
```

Script el cual controla el tiempo en el juego, Fuente: Mateo Parreño

3.5.2. Tres en raya

Es un juego tradicional, se juega por turnos, el jugador debe poner una X o un O, intentando hacer una línea vertical, horizontal o diagonal. Sólo se puede elegir las casillas donde este libre, para llegar a una casilla se debe realizar el número que se encuentra ahí mediante el lenguaje de señas y se sobrepone la X, puede terminar con un ganador o bien en empate. En la Figura 3.21 se observa la interfaz del juego tres en raya.

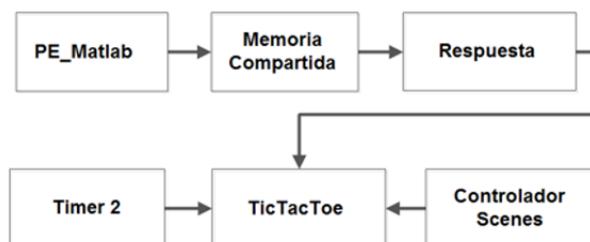
Figura 3.21 Juego Tres en raya



Interfaz gráfica del juego tres en raya, Fuente: Mateo Parreño

Para el desarrollo del juego tres en raya se crea los siguientes Script. En la Figura 3.22 se muestra el esquema del funcionamiento.

Figura 3.22 Desarrollo juego Tres en raya



Esquema de funcionamiento de los scripts del juego tres en raya, Fuente: Mateo Parreño

- Script PE_Matlab_I

Busca la mano izquierda, guardando los valores en la memoria compartida como se observa en la Figura 3.23

Figura 3.23 Script PE_Matlab_I



```
//-----Thumb Izquierda-----
IPosicionThumbBone1 = ManoIzquierda.transform.GetChild(5).GetChild(0).position;
IRotacionThumbBone1 = ManoIzquierda.transform.GetChild(5).GetChild(0).eulerAngles - new Vector3(360, 360, 360);
IPosicionThumbBone2 = ManoIzquierda.transform.GetChild(5).GetChild(1).position;
IRotacionThumbBone2 = ManoIzquierda.transform.GetChild(5).GetChild(1).eulerAngles - new Vector3(360, 360, 360);
IPosicionThumbBone3 = ManoIzquierda.transform.GetChild(5).GetChild(2).position;
IRotacionThumbBone3 = ManoIzquierda.transform.GetChild(5).GetChild(2).eulerAngles - new Vector3(360, 360, 360);

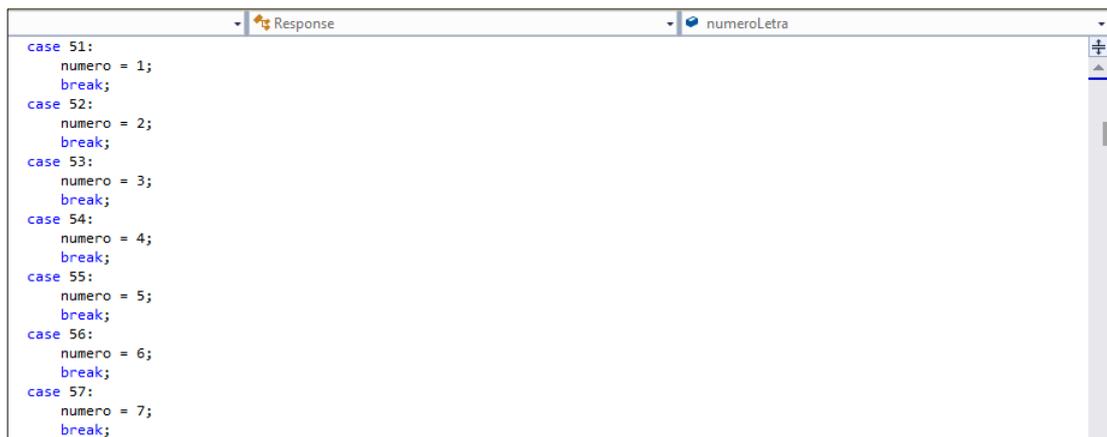
IPosicionThumbJoint3 = ManoIzquierda.transform.GetChild(5).GetChild(3).position;
IRotacionThumbJoint3 = ManoIzquierda.transform.GetChild(5).GetChild(3).eulerAngles - new Vector3(360, 360, 360);
IPosicionThumbJoint2 = ManoIzquierda.transform.GetChild(5).GetChild(4).position;
IRotacionThumbJoint2 = ManoIzquierda.transform.GetChild(5).GetChild(4).eulerAngles - new Vector3(360, 360, 360);
IPosicionThumbJoint1 = ManoIzquierda.transform.GetChild(5).GetChild(5).position;
IRotacionThumbJoint1 = ManoIzquierda.transform.GetChild(5).GetChild(5).eulerAngles - new Vector3(360, 360, 360);
// Bone Thumb Izquierda
writeFloatValue("ithumb", 1, IPosicionThumbBone3.x); // Posicion
writeFloatValue("ithumb", 2, IPosicionThumbBone3.y);
writeFloatValue("ithumb", 3, IPosicionThumbBone3.z);
writeFloatValue("ithumb", 4, IRotacionThumbBone3.x); // Rotacion
writeFloatValue("ithumb", 5, IRotacionThumbBone3.y);
writeFloatValue("ithumb", 6, IRotacionThumbBone3.z);
```

Script de la detección de la mano izquierda y envío de datos, Fuente: Mateo Parreño

- Script Response

Lee lo que responde Matlab que sea un número como se muestra en la Figura 3.24

Figura 3.24 Script Respuesta



```
case 51:
    numero = 1;
    break;
case 52:
    numero = 2;
    break;
case 53:
    numero = 3;
    break;
case 54:
    numero = 4;
    break;
case 55:
    numero = 5;
    break;
case 56:
    numero = 6;
    break;
case 57:
    numero = 7;
    break;
```

Script de respuesta que envía datos desde Matlab por la memoria compartida a Unity3D, Fuente: Mateo Parreño

- Script ControladorScenes

Controla el botón atrás y los botones del menú principal como observa en la Figura 3.25.

Figura 3.25 Script ControladorScenes

```
using UnityEngine.SceneManagement;

public class ControladorScenes : MonoBehaviour {

    // Use this for initialization
    void Start () {

    }

    // Update is called once per frame
    void Update () {

    }

    public void atras()
    {
        SceneManager.LoadScene("Home");
    }

    public void MissingLetters()
    {
        SceneManager.LoadScene("MissingLetters");
    }

    public void TicTacToe()
    {
        SceneManager.LoadScene("TicTacToe");
    }
}
```

Script de las funcione para controlar las Scenes, Fuente: Mateo Parreño

- **Script Timer2**

Permite ejecutar el conteo de tiempo para saber quién gano en un instante de tiempo como se observa en la Figura 3.26

Figura 3.26 Scrip Timer2

```
// Update is called once per frame
public void Update()
{
    if (comenzar == true)
    {
        if (minutos == minutoLimite)
        {
            desactivar();
        }
        segundos += Time.deltaTime;

        indicador.text = "" + Convert.ToInt16(minutos).ToString("00") + ":" + Convert.ToInt16(segundos).ToString("00");
        if ((int)segundos == 59)
        {
            segundos = 0;
            minutos++;
        }
    }
}
```

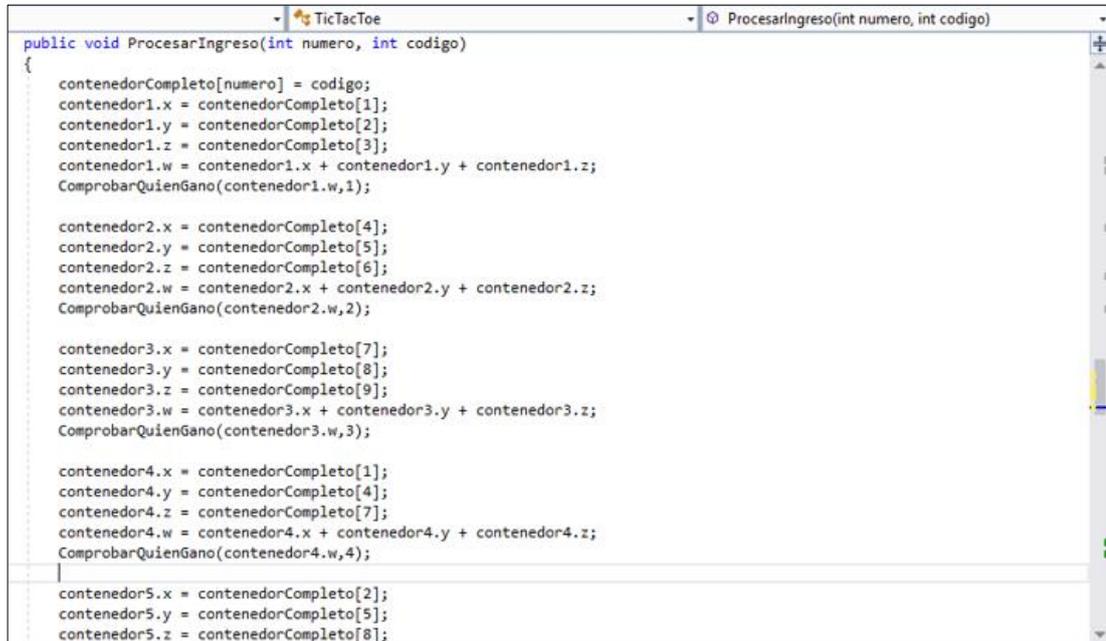
Script el cual controla el tiempo en el juego, Fuente: Mateo Parreño

- **Script TicTacToe**

Este script contiene funciones que permite: Borrar todas las marcas realizadas, reinicia el juego, alterna entre el player 1 y el player 2, administrar la operación de multijugador, ubicar la marca X o O dependiendo de casillas que el usuario elija,

administrar las imágenes que se muestran en la interfaz del usuario y ejecutar el algoritmo que determine el ganador o empate en el juego como se muestra en la Figura 3.27

Figura 3.27 Script TicTacToe



```
public void ProcesarIngreso(int numero, int codigo)
{
    contenedorCompleto[numero] = codigo;
    contenedor1.x = contenedorCompleto[1];
    contenedor1.y = contenedorCompleto[2];
    contenedor1.z = contenedorCompleto[3];
    contenedor1.w = contenedor1.x + contenedor1.y + contenedor1.z;
    ComprobarQuienGano(contenedor1.w,1);

    contenedor2.x = contenedorCompleto[4];
    contenedor2.y = contenedorCompleto[5];
    contenedor2.z = contenedorCompleto[6];
    contenedor2.w = contenedor2.x + contenedor2.y + contenedor2.z;
    ComprobarQuienGano(contenedor2.w,2);

    contenedor3.x = contenedorCompleto[7];
    contenedor3.y = contenedorCompleto[8];
    contenedor3.z = contenedorCompleto[9];
    contenedor3.w = contenedor3.x + contenedor3.y + contenedor3.z;
    ComprobarQuienGano(contenedor3.w,3);

    contenedor4.x = contenedorCompleto[1];
    contenedor4.y = contenedorCompleto[4];
    contenedor4.z = contenedorCompleto[7];
    contenedor4.w = contenedor4.x + contenedor4.y + contenedor4.z;
    ComprobarQuienGano(contenedor4.w,4);

    contenedor5.x = contenedorCompleto[2];
    contenedor5.y = contenedorCompleto[5];
    contenedor5.z = contenedorCompleto[8];
```

Script donde se encuentra el algoritmo y las funciones del juego tres en raya, Fuente: Mateo Parreño

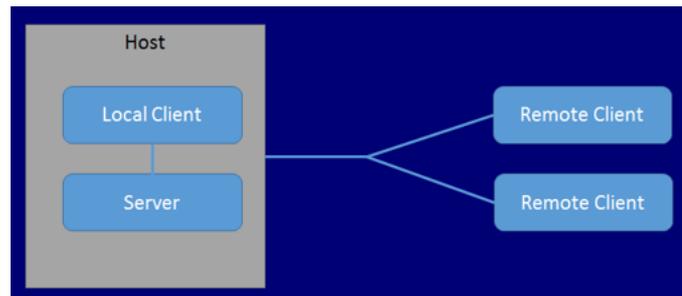
3.5.3. Multijugador

En el presente apartado se describe las características y el funcionamiento del multijugador en la interfaz de Unity3D.

- Servidor y Host

En el sistema de red de Unity, los juegos tienen un servidor y múltiples clientes. Cuando no hay un servidor dedicado, uno de los clientes cumple la función del servidor: llamamos a este cliente el "host" como se ejemplifica en la Figura 3.28.

Figura 3.28 Servidor y host



Esquema del funcionamiento del servidor y host en el juego, Fuente: Mateo Parreño.

El host es un servidor y un cliente en el mismo proceso. El host usa un tipo especial de cliente llamado LocalClient, mientras que otros clientes son RemoteClients. El LocalClient se comunica con el servidor (local) a través de llamadas de función directa y colas de mensajes, ya que está en el mismo proceso. En realidad, comparte la escena con el servidor. Los clientes remotos se comunican con el servidor a través de una conexión de red normal.

Uno de los objetivos del sistema de red es que el código para LocalClients y RemoteClients sea el mismo, de modo que los desarrolladores solo tengan que pensar en un tipo de cliente la mayor parte del tiempo.

- **Jugadores, jugadores locales y dedicados**

En el sistema de red, los objetos del jugador son especiales. Hay un objeto jugador asociado con cada persona presente en el juego, y los comandos se dirigen a ese objeto. Una persona no puede invocar un comando en el objeto de jugador de otra persona, solo el suyo. Entonces hay un concepto de "mi" jugador como objeto. Cuando se agrega un jugador y la asociación se realiza con una conexión, ese objeto de jugador se convierte en un objeto de "jugador local" en el cliente de ese jugador. Hay una propiedad isLocalPlayer que se establece en true y una retrollamada OnStartLocalPlayer () que se invoca en el objeto en el cliente. La Figura 3.29 muestra un diagrama de dos clientes y sus jugadores locales.

Figura 3.29 Diagrama de jugador y clientes en modo multijugador

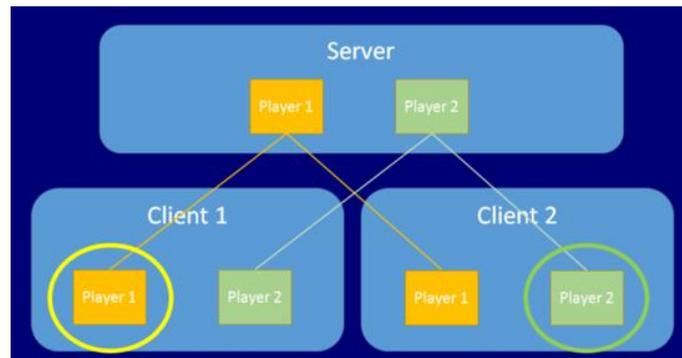


Diagrama explicativo de jugadores y clientes en modo multijugador Fuente: Mateo Parreño.

- **Módulo network manager**

Es un componente para administrar el estado de la red de un juego multijugador. En realidad, se implementa completamente con la API de alto nivel (HLAPI), por lo que todo lo que hace también está disponible, sin embargo, el componente Network Manager contiene una gran cantidad de funcionalidades útiles en un solo lugar, y hace que crear, ejecutar y depurar juegos multijugador sea lo más simple posible.

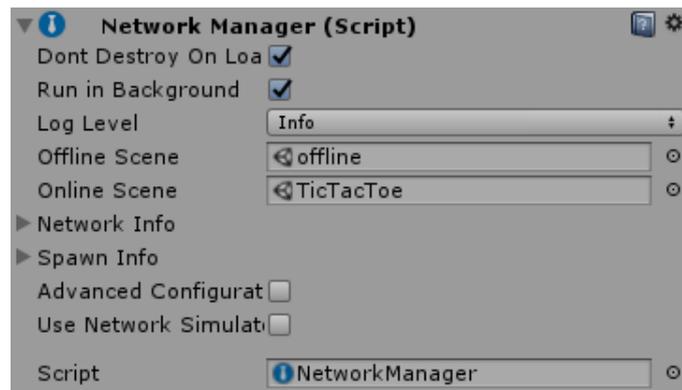
El módulo de Network Manager se puede usar completamente sin secuencias de comandos. Tiene controles Inspector en el Editor que permiten la configuración de todas sus características. El HUD de Network Manager proporciona una interfaz de usuario simple y predeterminada en tiempo de ejecución que permite que el juego en red sea controlado por el usuario. En el caso de aplicaciones específicas que demanden el uso de variables no contempladas por default, se puede acceder a la clase de NetworkManager y personalizar su comportamiento anulando cualquiera de los ganchos de funciones virtuales que proporciona.

Las características de Network Manager incluyen:

- Gestión del juego
- Gestión del desove
- Gestión de escena
- Información de depuración
- Casamentero
- Personalización

El módulo Network Manager se puede usar como el componente central de control de un juego multijugador. Para comenzar, se debe crear un GameObject vacío en la escena del juego o a su vez se puede elegir un GameObject conveniente que pueda alojar el componente de Network Manager. A continuación, agregue el componente NetworkManager desde el elemento de menú Network / NetworkManager. El componente de NetworkManager recién agregado se muestra en la Figura 3.30.

Figura 3.30 Modulo Network Manager



Módulo Network Manager en Unity3D, Fuente: Mateo Parreño

El Inspector para el Administrador de red en el Editor le permite configurar y controlar muchas cosas relacionadas con la red.

- **Módulo network manager hud**

Es otro componente que funciona con Network Manager. Le da una interfaz de usuario simple cuando el juego se ejecuta para controlar el estado de la red. Esto es bueno para comenzar con un proyecto de red, pero no pretende ser utilizado como la interfaz de usuario final para un juego. El NetworkManagerHUD se tal como se muestra en la Figura 2.31.

Figura 3.31 Modulo ManagerHUB

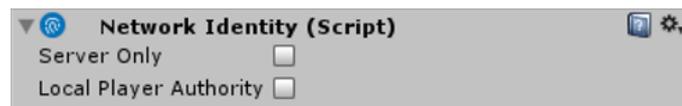


Interfaz que permite la conexión como servidor o cliente, Fuente: Mateo Parreño

- **Modulo network identity**

El componente NetworkIdentity es un módulo de Unity3D que está en el corazón del nuevo sistema de red. Se puede agregar a los objetos desde el elemento de menú AddComponents-> Network-> NetworkIdentity. Este componente controla la identidad de red de un objeto y lo hace consciente del sistema de red. La Figura 3.32 muestra cómo se ve el componente NetworkIdentity en un objeto.

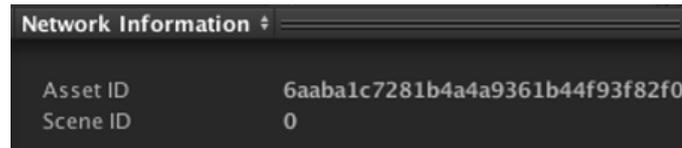
Figura 3.32 Modulo Network Identity



Módulo Network Identity en la interfaz Unity3D, Fuente: Mateo Parreño

Este componente contiene información de seguimiento, como qué ID de escena, ID de red e ID de elemento se ha asignado al objeto. El ID de escena es válido en todos los objetos de escena con un componente NetworkIdentity. El ID de red es el ID de esta instancia de objeto particular, puede haber múltiples objetos instanciados de un tipo de objeto particular y el ID de red se usa para identificar a qué objeto, por ejemplo, se debe aplicar una actualización de red. La identificación del activo se refiere a qué fuente de origen se creó la instancia del objeto. Esto se usa internamente cuando se genera un objeto prefabricado de objetos particular a través de la red. Esta información está expuesta en el panel de vista previa en la parte inferior del inspector como se muestra en la Figura 3.33

Figura 3.33 Información inherente a la conexión de red

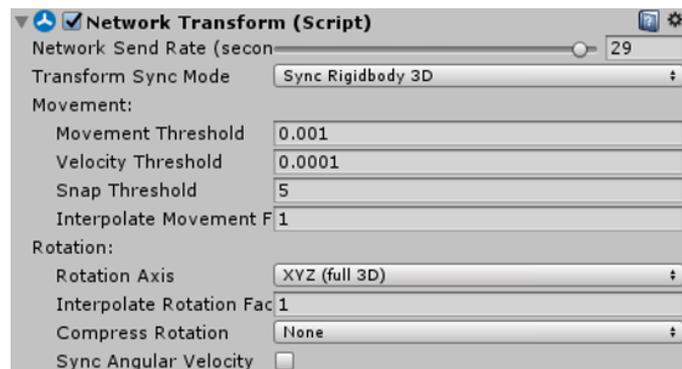


Información inherente a la conexión de red del módulo Network Identity, Fuente: Mateo Parreño

- **Modulo network transform**

El componente NetworkTransform sincroniza el movimiento de los objetos del juego a través de la red. Este componente tiene en cuenta la autoridad, por lo que los objetos LocalPlayer (que tienen autoridad local) sincronizan su posición del cliente al servidor y luego a otros clientes. Otros objetos (con autoridad del servidor) sincronizan su posición del servidor a los clientes. Para usar este componente, se debe agregarlo al objeto prefabricado del juego para el que desea sincronizar el movimiento o cambio de valores como se observa en la Figura 3.34. El componente requiere que los objetos del juego tengan una NetworkIdentity.

Figura 3.34 Modulo Network Transform



Módulo Network Transform en la interfaz de Unity3D, Fuente: Mateo Parreño.

CAPÍTULO 4

PRUEBAS

En esta sección se definen las especificaciones didácticas para diseñar cada uno de los componentes del juego, las mismas que están en función de proceso de enseñanza-aprendizaje que desean desarrollar en los para personas con deficiencia auditiva sin descartar que el sistema puede ser utilizado por niños, jóvenes y adultos que tengan o no esta discapacidad y con interés en aprender el lenguaje básico de señas ecuatoriano que interactúan en el juego con una interfaz amigable con el entorno que ayude y facilite el proceso de aprendizaje.

Combinar una interfaz gráfica de juego a través de los beneficios de la psicología del color en el proceso enseñanza-aprendizaje y dispositivo de entrada Leap Motion permiten interactuar con el entorno. (Elliot, 2014)

Las pruebas se realizaron según los siguientes aspectos:

- Distancia de interacción
- Reconocimiento de letras
- Tiempo de respuesta
- Test de usabilidad del sistema

4.1. Distancia de interacción

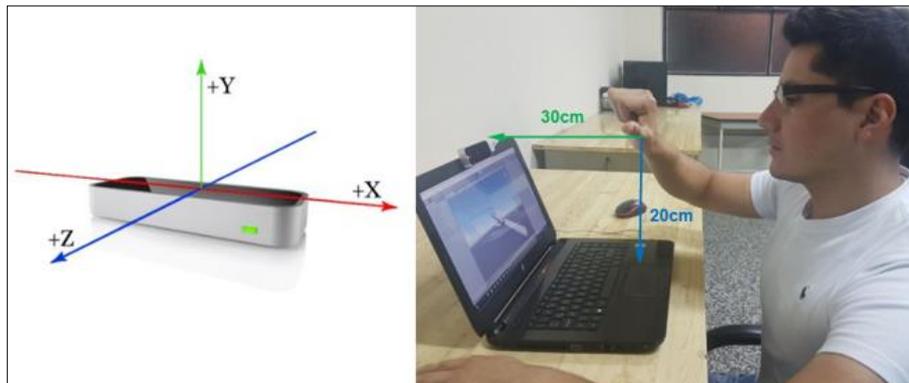
La prueba que se realizó para determinar la distancia de interacción, primero se ubicó el sensor Leap Motion en la parte superior de la pantalla del computador delante de cámara la cual el sensor tiene los ejes de coordenadas X, Y, Z. En la Figura 4.1 se observa los ejes de coordenadas que nos genera el Leap Motion.

Los datos técnicos del sensor indican que el área de cobertura es de 61cm para la detección de las manos, pero debido que el proyecto depende de la detección señas presenta problemas al momento del rastreo de las manos.

Las pruebas se han realizado con la letra A ya que es la letra más utilizada en el juego. En los ejes de coordenadas X, Y, Z a partir del punto de origen que se estableció frente al sensor Leap Motion a 30cm y a la altura 25cm del touchpad de la

computadora portátil. En la Figura 4.1 se muestran el punto de origen para determinar la distancia de interacción.

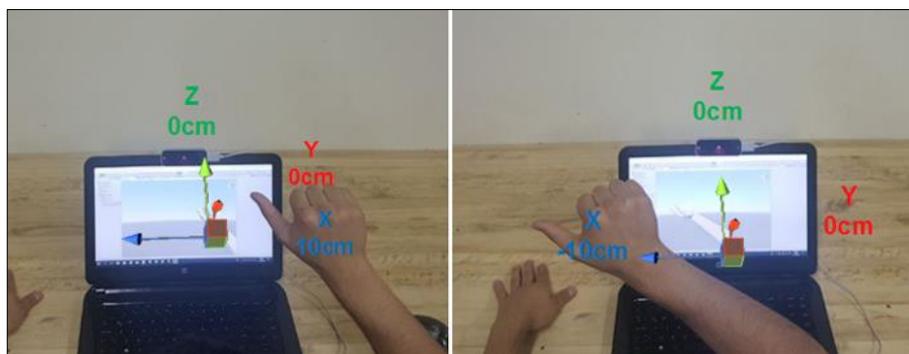
Figura 4.1 Ejes de coordenadas de Leap Motion



Eje de coordenadas y punto de origen de la interacción con el sensor Leap Motion, Fuente: Mateo Parreño

La prueba se realizó variando los valores de distancia sobre el eje X hasta que la señal de la letra A pierda presencia y manteniendo los ejes Y y Z en cero como se muestra la Figura 4.2.

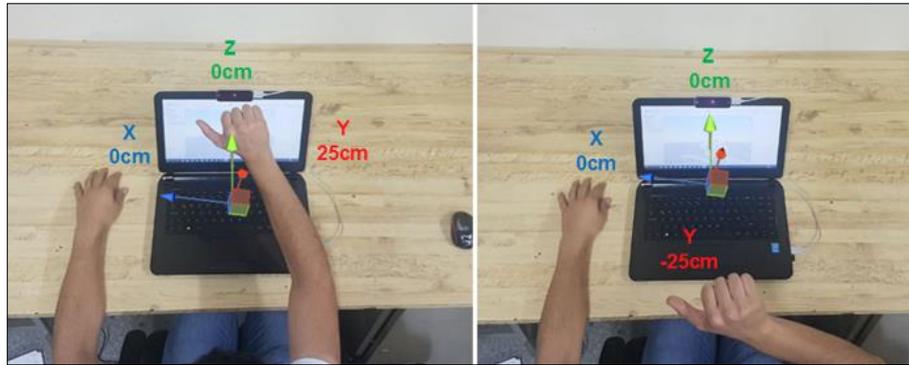
Figura 4.2 Interacción en el eje X



Distancia de interacción en el eje X del sensor Leap Motion, Fuente: Mateo Parreño

Se realiza de la misma forma la distancia de interacción en el eje Y variando la distancia hasta que la señal no sea reconocida y manteniendo los ejes X y Z en cero, pero se puede observar que el valor es de $\pm 25\text{cm}$ debido que tiene mayor detección en profundidad como se muestra en la Figura 4.3

Figura 4.3 Interacción eje Y



Distancia de interacción en el eje Y del sensor Leap Motion, Fuente: Mateo Parreño

La última prueba de distancia de interacción se realiza en el eje Z considerado como eje vertical y manteniendo en cero los ejes X e Y, los valores obtenidos son $\pm 10\text{cm}$ como se muestra en la Figura 4.4

Figura 4.4 Interacción en el eje Z



Distancia de interacción en el eje Z del sensor Leap Motion, Fuente: Mateo Parreño

Los valores obtenidos en la prueba de distancia de interacción se muestran en la Tabla 4.1 que indica que el sensor tiene una detección óptima para la ubicación de la mano al realizar las señas que corresponden a la posición Profundidad, Horizontal y Vertical respectivamente (Rivas, 2017), debido a los movimientos que realiza al contorno de la zona de interacción puede tener variaciones de hasta $\pm 25\text{ cm}$ en el eje Y (Profundidad), $\pm 10\text{ cm}$ en el eje Z (Vertical) y 10 cm en el eje X(Horizontal).

Tabla 4.1 Distancia de interacción de la letra B

DISTANCIA	EJE DE COORDENADAS		
	X (Y=0, Z=0)	Y (X=0, Z=0)	Z (X=0, Y=0)
-30cm	NO DETECTADO	NO DETECTADO	NO DETECTADO
-25cm	NO DETECTADO	DETECTADO	NO DETECTADO
-20cm	NO DETECTADO	DETECTADO	NO DETECTADO
-15cm	NO DETECTADO	DETECTADO	NO DETECTADO
-10cm	DETECTADO	DETECTADO	DETECTADO
-5cm	DETECTADO	DETECTADO	DETECTADO
0cm	DETECTADO	DETECTADO	DETECTADO
5cm	DETECTADO	DETECTADO	DETECTADO
10cm	DETECTADO	DETECTADO	DETECTADO
15cm	NO DETECTADO	DETECTADO	NO DETECTADO
20cm	NO DETECTADO	DETECTADO	NO DETECTADO
25cm	NO DETECTADO	DETECTADO	NO DETECTADO
30cm	NO DETECTADO	NO DETECTADO	NO DETECTADO

Tabla de valores de distancia de interacción de la letra B, Fuente: Mateo Parreño

4.2. Reconocimiento de señas

Para la prueba de reconocimiento de letras se realiza 20 repeticiones de cada letra del abecedario dividido en tres secciones: configuraciones dinámicas y estáticas (con la mano derecha) y números (con la mano izquierda) para determinar el grado de identificación al momento de ingresar los datos a la aplicación.

El valor obtenido en la prueba de reconocimiento de letras estáticas como se muestra en la Tabla 4.2 es de 86%, lo cual significa que tiene un rango adecuado para que el juego represente las señas en la interfaz de cada juego.

Tabla 4.2 Prueba de reconocimiento de letras de configuración estáticas

LETRA	REPETICIÓN																				PORCENTAJE	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20		
A	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	95%	
B	1	1	1	0	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	85%	
C	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1	0	1	80%	
CH	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	90%	
D	1	1	1	0	1	1	1	1	1	1	0	1	1	0	1	1	1	1	1	1	85%	
E	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	0	1	1	1	85%	
F	1	1	0	1	1	0	1	1	1	1	0	1	1	1	0	1	1	1	1	1	80%	
G	1	1	1	1	1	1	1	1	0	1	1	0	1	1	1	0	1	1	0	1	80%	
H	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	0	1	0	1	1	80%	
I	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	95%	
K	1	1	1	0	1	1	1	0	1	1	0	1	1	1	1	1	1	1	1	1	85%	
L	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	95%	
M	1	0	1	1	1	0	1	1	1	0	1	1	0	1	1	1	1	1	1	1	80%	
N	1	1	1	0	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	1	85%	
O	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	90%	
P	1	1	1	1	0	1	1	0	1	1	1	0	1	1	1	1	0	1	1	1	80%	
Q	1	1	1	0	1	1	0	1	0	1	1	1	0	1	1	1	1	0	1	1	75%	
R	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100%	
S	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	0	1	0	1	1	80%	
T	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	1	1	80%	
U	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	90%	
V	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	90%	
W	1	1	0	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	85%	
X	1	1	0	1	0	1	1	1	1	1	1	1	0	1	1	0	1	1	1	1	80%	
Y	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	95%	
																					PROMEDIO	86%

Tabla de valores obtenidos en la prueba de reconocimiento en las letras de configuración estática,

Fuente: Mateo Parreño

El promedio de reconocimiento de las letras de configuración dinámica es bajo con un promedio de 75% como se muestra en la Tabla 4.3 en comparación a la Tabla 4.2 de la configuración estática debido que la detección pierde precisión cuando la mano se mueve a una posición que obstruye la capacidad del controlador para detectarla, como en el caso de la letra j en la cual la mano debe girar por lo cual permanece un instante de tiempo perpendicular al controlador y su detección no es fiable en el 100% de los casos.

Tabla 4.3 Resultados del reconocimiento de letras de configuración dinámicas

LETRA	REPETICIÓN																				PORCENTAJE
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
J	1	1	0	0	1	0	1	0	1	1	1	1	0	1	1	1	0	1	1	1	70%
Ñ	1	0	1	1	1	0	1	0	1	0	1	1	1	0	1	1	1	0	1	1	70%
Z	1	1	1	1	0	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	85%
PROMEDIO																				75%	

Tabla de valores obtenidos en la prueba de reconocimiento en las letras en configuración dinámica,
Fuente: Mateo Parreño

Se realiza de la misma manera las repeticiones para los números como se muestra en la Tabla 4.4 obteniendo el mejor promedio con 89%, debido que estas señas se realizan exclusivamente con la mano izquierda y visualmente son más fáciles de realizar números que letras.

Tabla 4.4 Prueba de reconocimiento de números de configuración estáticas

NÚMERO	REPETICIÓN																				PORCENTAJE
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	90%
2	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	0	1	1	85%
3	1	1	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1	1	1	1	80%
4	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	90%
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	100%
6	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	90%
7	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	90%
8	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	0	1	1	1	85%
9	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	90%
PROMEDIO																				89%	

Tabla de valores obtenidos en la prueba de reconocimiento de números, Fuente: Mateo Parreño

4.3. Tiempo de respuesta

La prueba se realiza para tener un registro de los tiempos de respuesta desde que el usuario ingresa la seña con su mano hasta que el sistema retorna la letra o número realizado. Se realiza en las escenas de sintonización, juego fuga de letras y juego tres en raya. Los tiempos de retardo muestran una clara tendencia de baja latencia en la prueba de sintonización con un promedio de tiempo de 10.5ms, mientras que el tiempo de respuesta promedio de juego fuga de letras es 23.6ms. y el juego tres en raya tiene un tiempo de respuesta de 33.4ms como se muestra en la Tabla 4.5

Tabla 4.5 Resultados del tiempo de respuesta

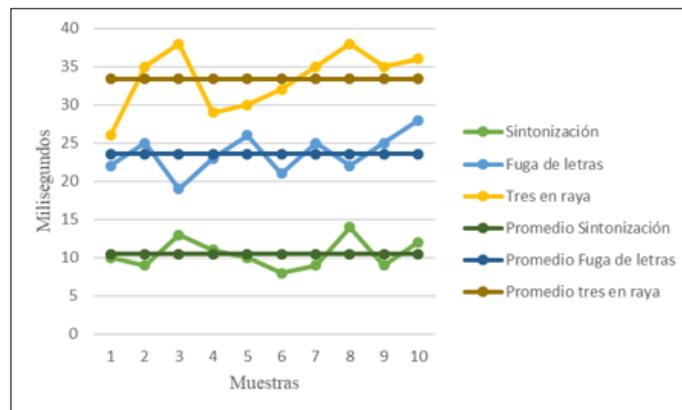
MUESTRA	TIEMPO DE RESPUESTA (ms)		
	Sintonización	Fuga de letras	Tres en raya
1	10	22	26
2	9	25	35
3	13	19	38
4	11	23	29
5	10	26	30
6	8	21	32
7	9	25	35
8	14	22	38
9	9	25	35
10	12	28	36
PROMEDIO	10,5	23,6	33,4

Tabla de valores obtenidos en la prueba de tiempo de respuesta en la sintonización y en el juego,

Fuente: Mateo Parreño

La Figura 4.5 muestra los valores obtenidos en la prueba de tiempo de respuesta de la Tabla 4.5 expresados en una gráfica de líneas 2D con marcadores, se puede observar que el tiempo de sintonización es más rápido y no sufre de retardos debido que sólo realiza el rastreo de las manos y el algoritmo de detección. En cambio, en la curva de tiempo del juego fuga de letras se nota un retardo promedio de 23.6ms debido a que incorpora un algoritmo de funcionamiento del juego adicional al módulo de rastreo y detección, y por último los tiempos del juego tres en raya tiene un retardo promedio de 33,4ms a causa de que tiene el módulo multijugador adicional al algoritmo de funcionamiento del juego y al módulo de rastreo y detección.

Figura 4.5 Tiempo de respuesta del sistema



Curvas de los tiempos de respuesta en las etapas de: sintonización, juego fuga de letras y tres en raya,
Fuente: Mateo Parreño

4.4. Test de usabilidad

El cuestionario SUS (System Usability Scale) fue desarrollado por John Brooke en 1986, en las oficinas de Digital Equipment Corporation (DEC) el objetivo del cuestionario SUS es proporcionar un test fácil de completar que cuente con un mínimo ítem, fácil de puntuar y que permitiera medir la usabilidad de programas, instrumentos y herramientas técnicas. (Schmitz B. T., 2013)

Las preguntas que consta en el test de usabilidad son las siguientes:

1. ¿Creó que me gustaría utilizar este sistema con frecuencia?
2. ¿Encontré el sistema innecesariamente complejo?
3. ¿Pensé que el sistema era fácil de usar?
4. ¿Creo que necesitaría el apoyo de una persona técnica para poder usar este sistema?
5. ¿Encontré que las diversas funciones en este sistema estaban bien integradas?
6. ¿Pensé que había demasiada inconsistencia en este sistema?
7. ¿Me imagino que la mayoría de las personas aprenderían a usar este sistema muy rápidamente?
8. ¿Encontré el sistema muy engorroso de usar?
9. ¿Me sentí muy seguro de usar el sistema?
10. ¿Necesitaba aprender muchas cosas antes de poder comenzar a usar este sistema?

4.4.1. Escala cuestionario sus

Es una escala de 1 a 5, de 1 "Totalmente en desacuerdo" a 5 "Totalmente de acuerdo". (Schmitz B. T., 2013)

1. Totalmente en desacuerdo
2. En desacuerdo
3. Ni de acuerdo ni en desacuerdo
4. De acuerdo
5. Totalmente de acuerdo.

Si el participante no responde a un ítem, asígnele un 3 (el centro de la escala de calificación).

4.4.2. Puntuación de sus

Para los ítems redactados positivamente (1, 3, 5, 7 y 9), la contribución de puntaje es la escala posición menos 1. Para los ítems de redacción negativa (2, 4, 6, 8 y 10), es 5 menos la posición de escala. Para obtener el puntaje total del SUS, se multiplica la suma de las contribuciones del puntaje del elemento por 2.5. Por lo tanto, los puntajes SUS van de 0 a 100 en incrementos de 2.5 puntos. (Schmitz B. T., 2013).

Se establece a través de este test el nivel de satisfacción de los usuarios al utilizar el juego. La aplicación desarrollada tiene un enfoque didáctico en el proceso enseñanza-aprendizaje, por ello la evaluación del uso de la aplicación involucra a personas con deficiencia auditiva sin descartar que el sistema puede ser utilizado por personas que estén interesadas en aprender el lenguaje básico de señas ecuatoriano. Se utiliza el método de evaluación SUS (Sauro, 2011) usando un único número en una escala estilo Likert. (Likert, 1932).

4.4.3. Experimentación

El experimento se realizó con dos sujetos que tienen deficiencia auditiva del colegio Primero de Abril de la ciudad de Latacunga los cuales cursan el bachillerato tienen conocimiento del lenguaje de señas y reciben educación inclusiva.

El sistema utilizado empleó dos computadoras que disponen de los programas de Unity3D y Matlab desarrollados en el presente trabajo de titulación.

En la primera sección se pone a prueba el juego fuga de letras que consiste en presentar una imagen y su correspondiente palabra incompleta, el objetivo es completar las letras que faltan en los espacios vacíos de la palabra, a través del lenguaje de señas como se observa en la Figura 4.6.

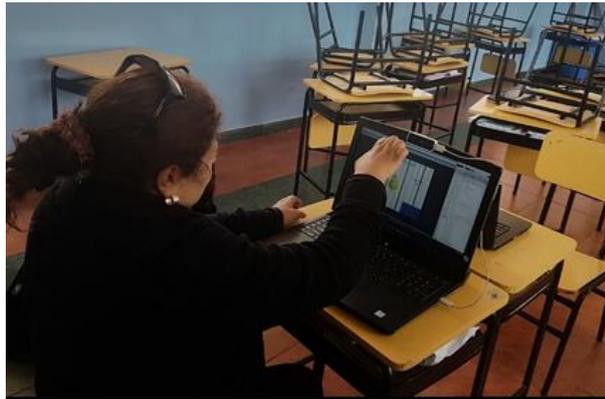
Figura 4.6 Prueba del juego fuga de letras



Prueba realizada en el Colegio Primero de Abril que tiene un programa de inclusión de personas con deficiencia auditiva, Fuente:

En la segunda sección de la prueba se utiliza el juego tres en raya que se basa en colocar una X o un O (la X es el símbolo del jugador 1 y O es el símbolo de jugador 2) en una matriz 3x3, intentando hacer una línea vertical, horizontal o diagonal con tres símbolos iguales. Sólo se pueden elegir las casillas vacías, para seleccionar una posición se debe realizar el número correspondiente a la casilla deseada mediante el lenguaje de señas y se ubica el símbolo X o O según corresponda el turno de cada jugador tal como se muestra en la Figura 4.7

Figura 4.7 Prueba del juego tres en raya



Prueba con personas con deficiencia auditiva del juego tres en raya, Fuente: Mateo Parreño

Después de las dos intervenciones del juego, el puntaje ganador de la primera sección es 30 puntos como se muestra en la Figura 4.8.

Figura 4.8 Resultado del juego fuga de letras



Resultados obtenidos al momento de jugar fuga de letras, Fuente: Mateo Parreño

El resultado de la segunda sección es el jugador 2 gana la partida como se observa que gana la partida de tres en raya como se observa en la Figura 4.9.

Figura 4.9 Resultado del juego tres en raya



Resultados obtenidos al momento de tres en raya, Fuente: Mateo Parreño

La tercera sección es realizar a los participantes el test de usabilidad. Se procede a evaluar mediante el test SUS a los sujetos de prueba. El cuestionario aplicado con los resultados obtenidos se muestra en la Tabla 4.6.

Tabla 4.6 Cuestionario de resultados

PREGUNTAS	PUNTUACIÓN			OPERACIÓN		
	USUARIO 1	USUARIO 2	USUARIO 3	USUARIO 1	USUARIO 2	USUARIO 3
1.¿Creó que me gustaría utilizar este sistema con frecuencia?	4	4	5	3	3	4
2.¿Encontré el sistema innecesariamente complejo?	1	1	2	4	4	3
3.¿Pensé que el sistema era fácil de usar?	3	5	4	2	4	3
4.¿Creo que necesitaría el apoyo de una persona técnica para poder usar este sistema?	2	1	1	3	4	4
5.¿Encontré que las diversas funciones en este sistema estaban bien integradas?	5	4	4	4	3	3
6.¿Pensé que había demasiada inconsistencia en este sistema?	2	2	2	3	3	3
7.¿Me imagino que la mayoría de las personas aprenderían a usar este sistema muy rápidamente?	4	5	5	3	4	4
8.¿Encontré el sistema muy engorroso de usar?	3	2	1	2	3	4
9.¿Me sentí muy seguro de usar el sistema?	4	5	5	3	4	4
10.¿Necesitaba aprender muchas cosas antes de poder comenzar a usar este sistema?	4	1	1	1	4	4
TOTAL				28	36	36
RESUSLTADO SUS				70	90	90
PROMEDIO SUS				83,33		

Tabla de valores obtenidos en el test de usabilidad, Fuente: Mateo Parreño

El número total obtenido según el tratamiento de resultados del test SUS indica que se debe multiplicar por un factor 2.5 dando como resultado tres valores 65, 90 y 90 dando lugar a un promedio de 81.67.

El resultado indica una clara oportunidad de mejorar en la interfaz de usuario, pero no implica que la versión no sea adecuada para pruebas a futuro. Si se quiere lograr un resultado mejor se puede intervenir en el módulo de interfaz de usuario utilizando psicología del color, añadir funciones al menú para que sea más amigable con el usuario.

CONCLUSIONES

A través del motor gráfico Unity3D y el sensor Leap Motion se logró desarrollar los juegos didácticos: fuga de letras y tres en raya, para estimular el proceso de aprendizaje del lenguaje básico de señas ecuatoriano.

Al momento de sintonizar las señas estáticas se usó el coeficiente de correlación, para determinar el significado de la seña, se considera el umbral requerido entre 0.85 y 0.98 con lo cual asegura un rango de detección fiable entre letras cuyas señas son similares como ocurre con las letras m y n.

El coeficiente de correlación se realizó con señas estáticas con una detección del 86% y en señas de números del 89%, y se realizó el algoritmo DTW para cada configuración dinámica proporcionando un rango de detección del 75% mediante las pruebas de detección realizadas. Estos datos reflejan que el sistema está reconociendo las letras y números del lenguaje de señas ecuatoriano sin mayor dificultad.

Para crear la letra patrón con la que se comparará la señal ingresada por el usuario al momento de la detección, se utilizó el promedio GoldStandar dando como resultado una representación válida de la seña cuando su valor es mayor o igual a 0.8.

El tiempo de detección en la etapa de sintonización con un valor de 10,5ms, siendo el menor en la aplicación; debido a que sólo realiza el rastreo de las manos y el algoritmo de detección, mientras que el segundo más rápido es el juego fuga de letras con un valor de 23,6ms porque se agrega un algoritmo de funcionamiento al módulo de rastreo y detección. El juego tres en raya tiene el tiempo de detección más alto con 33.3ms debido que adicionalmente al módulo de rastreo y detección implementa el algoritmo de funcionamiento y un módulo de multijugador.

A través del test de usabilidad SUS se obtuvo un valor de 81.67 el cual indica que el juego ofrece una experiencia del usuario aceptable, sin embargo, muestra también las mejoras que se deben implementar a nivel de interfaz de usuario para lograr un puntaje de al menos 90, medida en la cual se considera una aplicación con usabilidad amigable para el usuario.

Como resultado del presente trabajo de titulación se realizó un artículo académico con el tema “Teaching-Learning of Basic Language of Signs through Didactic Games” presentado en la "9th International Conference on Education Technology and Computers (ICETC 2017)" dando un enfoque al aprendizaje del lenguaje básico de señas ecuatoriano a través de un juego didáctico a personas con deficiencias auditivas.

RECOMENDACIONES

Se recomienda que la aplicación sea utilizada por usuarios que tengan completa movilidad en las manos y todos sus dedos, debido a que el clasificador depende del rastreo de los huesos de la mano y al no detectar todos sus dedos el umbral cambia radicalmente.

Es necesario utilizar la versión del driver 3.10 del sensor Leap Motion debido a los cambios de posición de referencia en la parte frontal de la mano, las últimas versiones hacen referencia a la parte posterior de la mano para un mejor manejo en aplicaciones de realidad virtual.

Se debe usar la aplicación por un periodo máximo de 4 horas debido a que el sensor LeapMotion tiende a calentarse, presenta lecturas erróneas y su latencia se incrementa

Se recomienda realizar las señas dentro de la zona de trabajo en la cual la detección es óptima, esta zona está determinada por una distancia de ± 25 cm en el eje Y (Profundidad), ± 10 cm en el eje Z (Vertical) y 10 cm en el eje X(Horizontal).

BIBLIOGRAFÍA

- Andaluz, V. H. (2016). Unity3D-MatLab simulator in real time for robotics applications. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics*, 246-263.
- Belda, J. (2015, 04 20). *Showleap Technologies*. Retrieved from Leap Motion Características Técnicas: <http://blog.showleap.com/2015/04/leap-motion-caracteristicas-tecnicas/>
- Chuan, C. H. (2014). American sign language recognition using leap motion sensor. *Machine Learning and Applications (ICMLA)*, 541-544.
- CONADIS. (2016, 09 15). *Diccionario de Lengua de señas Ecautoriana Gabriel Román*. Retrieved from <http://plataformaconadis.gob.ec/diccionario/>
- CORDICOM. (2015, 02 09). *Consejo de regulación y desarrollo de la información y comunicación*. Retrieved from <http://www.cordicom.gob.ec/las-personas-sordas-conquistan-el-espacio-en-la-television-ecuatoriana/>
- Dhingra, S. D. (2013). Isolated speech recognition using MFCC and DTW. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 2(8), 4085-4092.
- Elliot, A. J. (2014). Color psychology: Effects of perceiving color on psychological functioning in humans. . *Annual review of psychology*, 65, 95-120.
- Guna, J. J. (2014). n analysis of the precision and reliability of the leap motion sensor and its suitability for static and dynamic tracking. *Sensors*, 3702-3720.
- INEC. (n.d.). *Instituto Nacional de Estadísticas y Censos*. Retrieved from <http://www.ecuadorencifras.gob.ec/censo-de-poblacion-y-vivienda/>
- J.M.Nieves. (2014, 04 29). *ABC Tecnología*. Retrieved from La tecnología marca el lenguaje gestual: <http://www.abc.es/tecnologia/informatica-software/20140423/abci-tecnologia-gestos-smartphones-lenguaje-201404231337.html>

- Jiménez, L. A. (2017). Gesture Recognition and Machine Learning Applied to Sign Language Translation. *VII Latin American Congress on Biomedical Engineering CLAIB 2016.*, 233-236.
- Kristoufek, L. (2014). Detrending moving-average cross-correlation coefficient: Measuring cross-correlations between non-stationary series. *Physica A: Statistical Mechanics and its Applications*, 406, 169-175.
- Leap Motion SDK Documentation. (2017, 10 10). *Leap Motion SDK Documentation*. Retrieved from <https://developer.leapmotion.com/documentation/index.html>
- Likert, R. (1932). A technique for the measurement of attitudes. *Archives of psychology*.
- MathWorks, Inc. (2017). *MathWorks*. Retrieved from https://www.mathworks.com/programs/trials/trial_request.html?ref=ggl&seid=ppc_29850150442&q=matlab
- Mohandes, M. A. (2014). Arabic sign language recognition using the leap motion controller. *23rd International Symposium on*, 960-965.
- Mueen, A. &. (2016). Extracting optimal performance from dynamic time warping. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* , 2129-2130.
- NMformación. (2015, 07 15). *NMformación*. Retrieved from Tecnología gestual la evolución de lo táctil: <http://www.nmformacion.com/blog.asp?vcblog=1359&vcSeccion=1>
- Oleaga, J. (2014, 02 05). *ABC Tecnología*. Retrieved from <http://www.abc.es/tecnologia/informatica/20140204/abci-probamos-leap-motion-201402041411.html>
- Paula, N., Israel, A., & Núñez Govín, Y. (2005). Propuesta de clasificación de las herramientas-software para la gestión del conocimiento. *ACIMED*, 13(2), 1-1.
- Potter, L. E. (2013). The leap motion controller: a view on sign language. *roceedings of the 25th Australian computer-human interaction conference: augmentation, application, innovation, collaboration*, 175-178.

- Rivas, D. A. (2017). LeSigLa_EC: Learning Sign Language of Ecuador. *Springer, Cham.*, 170-179.
- Sauro, J. &. (2011). When designing usability questionnaires, does it hurt to be positive? *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2215-2224.
- Schmitz, B. T. (2013). Designing a mobile learning game to investigate the impact of role-playing on helping behaviour. *Springer*, 357-370.
- Schmitz, B. T. (2013). Designing a mobile learning game to investigate the impact of role-playing on helping behaviour. *In European Conference on Technology Enhanced Learning Springer*, 357-370.
- Sordos Ecuatorianos. (s.f.). *Lenguaje de Señas*. Obtenido de Diccionarios de Lengua de Señas Ecuatoriana : <http://www.sordosecuador.com/p/diccionarios-de-lengua-de-senas.html>
- Taylor, R. (1990). Interpretation of the correlation coefficient a basic review. *Journal of diagnostic medical sonography*, 6(1), 35-39.
- Vannatta, R., Beyerbach, B., & Walsh, C. (2001). From teaching technology to using technology to enhance student learning: Preservice teachers' changing perceptions of technology infusion. *ournal of Technology and Teacher Education*, 9(1), 105-127.
- Verbal, C. (2017, 01 10). *El Cuerpo no miente*. Retrieved from Invadirá tecnología gestual la vida cotidiana.: <http://www.elcuerponomiente.com/publicacion-48>
- Versi, E. (1992). Gold standard is an appropriate term. . *British Medical Journal*, 305(6846), 187.
- Vicepresidencia. (2014). *Vicepresidencia*. Retrieved from Lanzamiento diccionario de lengua de señas ecuatoriana: <http://www.vicepresidencia.gob.ec/lanzamiento-diccionario-de-lengua-de-senas-ecuatoriana/>
- Virvou, M., Katsionis, G., & Manos , K. (2005). Combining software games with education. *Educational Technology & Society*, 8(2), 54-65.

ANEXO:

MANUAL DE USUARIO

Elaborado por Mateo Parreño

El menú de la aplicación se encuentra tres botones los cuales son: fuga de letras, tres en raya y el botón salir.

Se siguen los siguientes pasos.

1. Se selecciona el primer juego fuga de letras



2. Para comenzar el juego se selecciona el botón iniciar



3. Se debe completar las letras que faltan en los espacios vacíos de la palabra, a través del lenguaje de señas.



4. El ganador es aquel usuario que tenga el mayor puntaje



5. Para regresar al menú principal se selecciona el botón atrás que se encuentra en la parte inferior izquierda

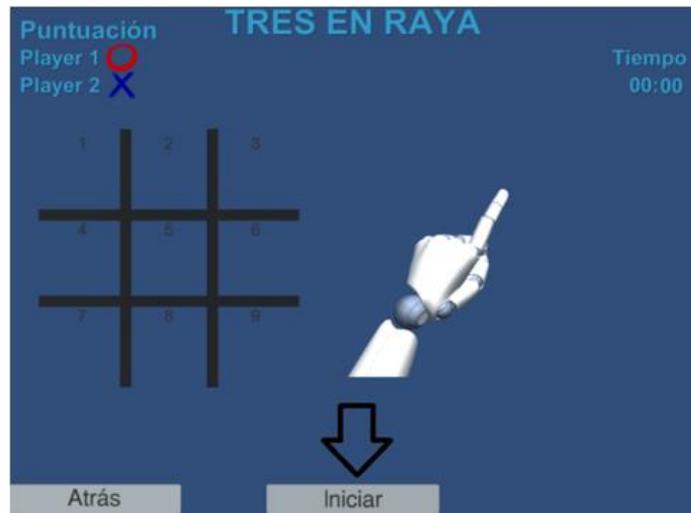


Ahora se realiza el segundo juego tres en raya que se basa en colocar una X o un O (la X es el símbolo del jugador 1 y O es el símbolo de jugador 2), intentando hacer una línea vertical, horizontal o diagonal con tres símbolos iguales realiza los siguientes pasos:

1. Para seleccionar el segundo juego en el menú se presiona el botón del juego tres en raya



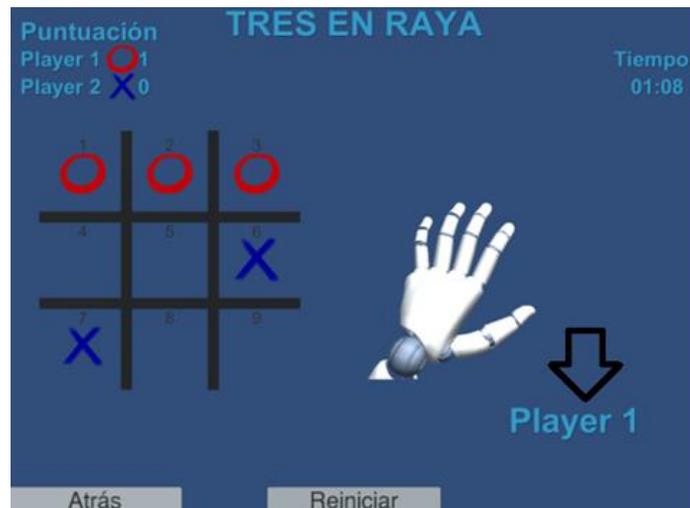
- Para comenzar el juego se selecciona el botón iniciar que se encuentra en la parte inferior de la interfaz grafica



- Se selecciona una casilla que este vacía para colocar una X o O



4. Se debe respetar el orden de cada jugador, en la parte inferior derecha se especifica el turno del jugador



5. El ganador es el jugador que más partidas haya ganado en tiempo que va de 3 a 5 minutos



6. Para regresar al menú principal se selecciona el botón atrás que se encuentra en la parte inferior izquierda



Para terminar la aplicación se selecciona el botón salir que se encuentra en la parte inferior derecha

