



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA E  
INSTRUMENTACIÓN**

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL  
TÍTULO DE INGENIERO EN ELECTRÓNICA E  
INSTRUMENTACIÓN**

**TEMA: SISTEMA DE REALIDAD VIRTUAL PARA  
REHABILITACIÓN MOTORA FINA EN NIÑOS MEDIANTE  
DISPOSITIVOS DE TRACKING ÓPTICO**

**AUTORAS:**

**ELSA ALEXANDRA CHUQUITARCO AGUAYO**

**JENNY PAOLA TIGSE CANDO**

**DIRECTOR: ING. MARCO PILATASIG**

**LATACUNGA**

**2017**



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA  
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN**

**CERTIFICACIÓN**

Certifico que el trabajo de titulación, **“SISTEMA DE REALIDAD VIRTUAL PARA REHABILITACIÓN MOTORA FINA EN NIÑOS MEDIANTE DISPOSITIVOS DE TRACKING ÓPTICO”** realizado por las señoritas **ELSA ALEXANDRA CHUQUITARCO AGUAYO Y JENNY PAOLA TIGSE CANDO**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de las Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar a las señoritas **ELSA ALEXANDRA CHUQUITARCO AGUAYO Y JENNY PAOLA TIGSE CANDO** para que lo sustenten públicamente.

Latacunga, 20 de octubre 2017

  
\_\_\_\_\_  
Ing. Marco Pilatasig  
**DIRECTOR**



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA  
CARRERA DE INGENIERÍA EN ELECTRÓNICA E INSTRUMENTACIÓN**

**AUTORÍA DE RESPONSABILIDAD**

Nosotras, **Elsa Alexandra Chuquitarco Aguayo**, con cedula de identidad N°1718921420 y **Jenny Paola Tigse Cando**, con cedula de identidad N°0504235664 declaro que el presente trabajo de titulación, “**SISTEMA DE REALIDAD VIRTUAL PARA REHABILITACIÓN MOTORA FINA EN NIÑOS MEDIANTE DISPOSITIVOS DE TRACKING ÓPTICO**” ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de nuestra autoría, en virtud de ello nos declaramos responsables del contenido, veracidad y alcance de la investigación mencionada.

Latacunqa, 20 de octubre del 2017

---

**Elsa Alexandra Chuquitarco Aguayo**

**C.C.: 1718921420**

---

**Jenny Paola Tigse Cando**

**C.C.: 0504235664**



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA  
CARRERA DE INGENIERÍA DE ELECTRÓNICA E INSTRUMENTACIÓN**

**AUTORIZACIÓN**

Nosotras, **ELSA ALEXANDRA CHUQUITARCO AGUAYO** y **JENNY PAOLA TIGSE CANDO**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la Biblioteca Virtual de la institución el presente trabajo de titulación **“SISTEMA DE REALIDAD VIRTUAL PARA REHABILITACIÓN MOTORA FINA EN NIÑOS MEDIANTE DISPOSITIVOS DE TRACKING ÓPTICO”** cuyo contenido, ideas y criterios son de nuestra autoría y responsabilidad.

Latacunga, 20 de octubre del 2017

---

Elsa Alexandra Chuquitarco Aguayo

C.C.: 1718921420

---

Jenny Paola Tigse Cando

C.C.: 0504235664

## DEDICATORIA

Dedicado a mi Dios por haberme permitido lograr las metas que me he propuesto. Y a mis padres, quienes son mi pilar fundamental en toda mi carrera y que con su amor, paciencia, y apoyo me guiaron por el buen camino para ser una persona de bien y convertirme en una buena profesional.

Alexandra Chuquitarco

## DEDICATORIA

Dedico este proyecto con mucho amor a mis padres, Raúl Tigse y Carmen Cando quienes han estado dándome sus consejos y apoyo incondicional durante toda mi vida y mi carrera universitaria, por su paciencia, amor, cariño y por ser los motores principales en mi vida para luchar diariamente a pesar de las dificultades que se presentan.

A mis dos grandes ángeles, mis abuelos Gerardo Cando y Carmen Amelia Caillagua, quienes desde el cielo me cuidan y me guían para continuar por el camino del bien.

A mi hermanita Lessly Tigse, quien llegó a mi vida a alegrarme con sus ocurrencias y para quien deseo ser el mayor ejemplo de lucha.

A toda mi familia, por ese cariño tan maravilloso que me han sabido brindar siempre y todas esas palabras de motivación.

Jenny Paola Tigse Cando

## AGRADECIMIENTO

Doy gracias a Dios por haberme guiado, protegido, por iluminar mi mente y haberme dado fuerzas y valor para poder culminar esta etapa de mi vida.

A mis padres Elsa Aguayo y Jorge Chuquitarco, quienes siempre han creído en mí y han sido y seguirán siendo mi apoyo incondicional durante toda mi vida, ya que me han demostrado su amor en las buenas y las malas. ¡Gracias por apoyarme, los amo!

A mi hermano Jorge Chuquitarco, por estar junto a mí y apoyarme siempre a pesar de las dificultades que hemos tenido.

A mi enamorado Rodrigo Barrionuevo, por apoyarme, motivarme y demostrarme que siempre puedo contar él.

A mis ingenieros Marco Pilatasig, Ivón Escobar y Edwin Pruna, quienes con sus aportaciones han hecho posible la culminación de este proyecto de investigación.

Y a todos mis amigos que entre bromas, risas y enojos estamos logrando culminar con éxito nuestra carrera universitaria.

Alexandra Chuquitarco

## AGRADECIMIENTO

Al culminar con este proyecto, quiero agradecer a mis padres, por su apoyo, comprensión, por esas palabras motivadoras que me ha permitido luchar ante cualquier obstáculo y por su infinito amor, son mi vida entera, los adoro.

A mi abuelito Gerardo Cando, quien antes de partir, supo dejar clavado en mi mente aquellas frases que en momentos difíciles me ha ayudado a levantarme y continuar esforzándome en la vida.

A mi abuelita Carmen Caillagua, quien a pesar de no haber tenido la oportunidad de conocerla me dejó el mejor regalo de la vida, un padre luchador.

A mis tíos William, Marisol y Walter Tigse, quienes han estado dándome consejos, apoyándome y enviando esas buenas vibras a pesar de la distancia. Al Ing. Marco Pilatasig, Ing. Ivón Escobar y al Ing. Edwin Pruna por sus asesoramientos y paciencia brindada, durante la ejecución de este proyecto de titulación.

A toda mi familia, por el apoyo que siempre me han brindado en todo momento y la confianza que me han tenido.

A todos mis amigos que durante estos años de Universidad he conocido y con los cuales he vivido grandes momentos que quedaran guardados en mi memoria para siempre, especialmente a Nataly, Rosita, Liseth, Fernanda, Ale, Washo, Carlos, Andy, Jefferson, Cristian con quienes he compartido muchos momentos de alegrías, tristezas y me han estado apoyando en todo momento.

Jenny Paola Tigse Cando



## ÍNDICE DE CONTENIDO

PORTADA .....	ii
CERTIFICACIÓN.....	ii
AUTORÍA DE RESPONSABILIDAD .....	iii
AUTORIZACIÓN .....	iv
DEDICATORIA .....	v
AGRADECIMIENTO .....	vii
ÍNDICE DE CONTENIDO .....	ix
ÍNDICE DE TABLAS .....	xiii
ÍNDICE DE FIGURAS.....	xiv
RESUMEN .....	xviii
ABSTRACT .....	xix

### CAPÍTULO I

<b>1. INTRODUCCIÓN .....</b>	<b>1</b>
1.1 Planteamiento del problema .....	1
1.2 Antecedentes .....	2
1.3 Justificación e Importancia .....	3
1.4 Objetivos .....	4
1.4.1 Objetivo General.....	4
1.4.2 Objetivos Específicos.....	4
1.5 Variables de la investigación .....	5
1.5.1 Variable Independiente.....	5
1.5.2 Variable Dependiente .....	5
1.6 Hipótesis .....	5

### CAPÍTULO II

<b>2. MARCO TEÓRICO .....</b>	<b>6</b>
2.1 Antecedentes Investigativos .....	6
2.2 Fundamentación Teórica.....	6
2.2.1 Realidad Virtual .....	7
2.2.2 Dispositivos de Tracking Ópticos.....	8
2.2.3 Motor Gráfico Unity .....	14
2.2.4 Motricidad Fina .....	16

## CAPÍTULO III

<b>3.</b>	<b>DIMENSIONAMIENTO DE HARDWARE Y SOFTWARE .....</b>	<b>22</b>
3.1	Análisis del Software Unity .....	22
3.2	Conexión de dispositivos de entrada y salida del sistema .....	23
3.3	Integración de dispositivos del sistema en Unity.....	25
3.3.1	Integración Oculus Rift .....	25
3.3.2	Integración LeapMotion .....	26
3.4	Integración de módulos de LeapMotion a Unity .....	27
3.4.1	Módulo Attachments .....	28
3.4.2	Módulo Detection Examples .....	29
3.4.3	Interaction Engine .....	30
3.4.4	Módulo de UI Input Module .....	31

## CAPÍTULO IV

<b>4.</b>	<b>DESARROLLO DE LA INTERFAZ .....</b>	<b>32</b>
4.1	Diseño de objetos 3D en Blender .....	33
4.2	Diseño de aplicaciones en el software Unity.....	35
4.2.1	Diseño del menú de ingreso de datos del usuario.....	35
4.2.2	Diseño del menú principal de los juegos .....	37
4.2.3	Diseño de la interfaz del cuarto.....	39
4.2.4	Diseño de la interfaz del Zoológico .....	42
4.2.5	Diseño de la interfaz Aula .....	46
4.3	Desarrollo de scripts para el sistema.....	47
4.3.1	Script para el menú ingreso de datos del usuario.....	48
4.3.2	Script para el menú principal de los juegos.....	48
4.3.3	Script para la aplicación agarre y lanzamiento de pelotas .....	49
4.3.4	Script para la aplicación clasificación de objetos por colores. ....	51
4.3.5	Script para la aplicación clasificación de objetos por formas.....	53
4.3.6	Script para la aplicación explota globos con pinzas. ....	54
4.3.7	Script para la aplicación atraviesa los aros.....	56
4.3.8	Script para la aplicación atraviesa los arcos.....	59
4.3.9	Script para la aplicación Pinta y escribe libremente. ....	60
4.4	Descripción general del sistema.....	61
4.4.1	Actividades de Coordinación Viso-Manual.....	62

4.4.2	Actividades de Coordinación Grafo Perceptiva .....	66
4.5	Movimientos de rehabilitación en el sistema.....	67
4.6	Descripción de los juegos.....	69

## **CAPÍTULO V**

<b>5.</b>	<b>ANÁLISIS DE RESULTADOS .....</b>	<b>76</b>
5.1	Consideraciones generales de los dispositivos de tracking óptico.....	76
5.2	Validación de los dispositivos mediante el test Box and Block.....	77
5.2.1	Toma de datos de las manos izquierda y derecha .....	77
5.3	Validación de Movimientos de Rehabilitación motora fina con el dispositivo LeapMotion. ....	79
5.4	Validación del sistema de realidad virtual en niños.....	81
5.5	Validación del sistema de realidad virtual mediante el Test de Usabilidad SEQ.....	83
5.6	Confirmación de la hipótesis planteada .....	85

## **CAPÍTULO VI**

<b>6.</b>	<b>CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>89</b>
6.1	Conclusiones.....	89
6.2	Recomendaciones.....	90

<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>91</b>
---	-----------

## **ANEXOS**

### **Anexo A: Manual de Usuario**

Ingreso de datos

Menú

Aplicaciones

### **Anexo B: Scripts de Programación**

Scripts de Unity

### **Anexo C: Manuales de Instalación**

Instalador Unity3D

Instalador Oculus Rift DK2 Runtime

Instalador LeapMotion

### **Anexo D: Cuestionario SEQ de Usuarios**

CuestionarioSEQ N°1

CuestionarioSEQ N°2

CuestionarioSEQ N°3

CuestionarioSEQ N°4

CuestionarioSEQ N°5

## ÍNDICE DE TABLAS

<b>Tabla 1.</b>	Comparación de motores gráficos .....	23
<b>Tabla 2.</b>	Comparación de Movimientos .....	67
<b>Tabla 3.</b>	Movimiento de la nave.....	73
<b>Tabla 4.</b>	Movimiento de la esfera.....	74
<b>Tabla 5.</b>	Características técnicas del LeapMotion .....	76
<b>Tabla 6.</b>	Características técnicas del Oculus Rift DK2 .....	76
<b>Tabla 7.</b>	Datos de cubos colocados mediante Test Box and Blocks físico virtual.....	78
<b>Tabla 8.</b>	Test SEQ con niños.....	84
<b>Tabla 9.</b>	Tiempo por día, Primer Usuario.....	85
<b>Tabla 10.</b>	Tiempo por día, Segundo Usuario .....	85
<b>Tabla 11.</b>	Tiempo por día, Tercer Usuario .....	86
<b>Tabla 12.</b>	Tiempo por día, Cuarto Usuario .....	86
<b>Tabla 13.</b>	Tiempo por día, Quinto Usuario.....	86

## ÍNDICE DE FIGURAS

<b>Figura 1.</b>	Realidad Virtual.....	7
<b>Figura 2.</b>	Hardware del dispositivo LeapMotion. ....	9
<b>Figura 3.</b>	Software del dispositivo LeapMotion.....	10
<b>Figura 4.</b>	Zona de cobertura LeapMotion. ....	10
<b>Figura 5.</b>	Mapa de mallado de puntos de LeapMotion. ....	11
<b>Figura 6.</b>	Sistema de coordenadas de LeapMotion.....	11
<b>Figura 7.</b>	Cuatro gestos que detecta el LeapMotion.....	12
<b>Figura 8.</b>	Dispositivo Oculus Rift DK2. ....	12
<b>Figura 9.</b>	Comunicación correcta a través del demo de Oculus. ....	14
<b>Figura 10.</b>	Editor de Unity3D.....	15
<b>Figura 11.</b>	Interfaz de la Asset Store de Unity.....	16
<b>Figura 12.</b>	Resumen de la psicomotricidad .....	17
<b>Figura 13.</b>	Actividades de motricidad .....	18
<b>Figura 14.</b>	Beneficios de la motricidad .....	19
<b>Figura 15.</b>	Clasificación de la motricidad fina .....	19
<b>Figura 16.</b>	Actividades de la Motricidad Manual.....	21
<b>Figura 17.</b>	Esquema del usuario y el sistema virtual .....	22
<b>Figura 18.</b>	Diagrama general del sistema .....	24
<b>Figura 19.</b>	Conexión cable USB LeapMotion-PC .....	24
<b>Figura 20.</b>	Conexión Casco HMD-PC .....	25
<b>Figura 21.</b>	Importación de Unity Package .....	26
<b>Figura 22.</b>	Ruta de acceso a la carpeta Scenes.....	27
<b>Figura 23.</b>	Prueba de funcionamiento LeapMotion en Unity .....	27
<b>Figura 24.</b>	Ejemplos de módulos de LeapMotion .....	28
<b>Figura 25.</b>	Detectors del módulo Attachments .....	29
<b>Figura 26.</b>	Detector Logic Gate en Unity .....	29
<b>Figura 27.</b>	Combinación de detectores .....	30
<b>Figura 28.</b>	Clasificación de componentes de interacción .....	30
<b>Figura 29.</b>	Manipulación de un botón estándar .....	31
<b>Figura 30.</b>	Pantalla inicial de Blender.....	33
<b>Figura 31.</b>	Figuras básicas y herramientas .....	34

<b>Figura 32.</b> Montaje de modelo 3D finalizado.....	34
<b>Figura 33.</b> Añadir un botón a la escena. ....	36
<b>Figura 34.</b> Ingreso de datos del usuario.....	36
<b>Figura 35.</b> Objetos 3D para la aplicación Menú principal .....	37
<b>Figura 36.</b> Plataforma de la base añadida en el entorno virtual. ....	37
<b>Figura 37.</b> Plataforma circular con rueda y letras.....	38
<b>Figura 38.</b> Entorno del Menú de juegos. ....	39
<b>Figura 39.</b> Objetos 3D como libros, cama, entre otros en la escena.....	39
<b>Figura 40.</b> Materiales de diferentes colores para los objetos del cuarto. ....	40
<b>Figura 41.</b> Interfaz del cuarto realista.....	40
<b>Figura 42.</b> Osos y pelotas agregados al entorno.....	41
<b>Figura 43.</b> Panel de visualización de datos para el usuario. ....	41
<b>Figura 44.</b> Panel de visualización del entorno clasificación por colores.....	42
<b>Figura 45.</b> Datos de la aplicación: Clasificación por formas.....	42
<b>Figura 46.</b> Primer ambiente del zoológico.....	43
<b>Figura 47.</b> Segundo ambiente del zoológico. ....	43
<b>Figura 48.</b> Tercer ambiente del zoológico. ....	44
<b>Figura 49.</b> Cuarto ambiente del zoológico.....	44
<b>Figura 50.</b> Unión de los cuatro ambientes del zoológico.....	44
<b>Figura 51.</b> Aplicación de explota globos con pinzas.....	45
<b>Figura 52.</b> Aplicación de atraviesa los aros.....	45
<b>Figura 53.</b> Aplicación de atraviesa los arcos.....	45
<b>Figura 54.</b> Decoración de la interfaz del aula.....	46
<b>Figura 55.</b> Botones de colores para la aplicación de pintar y escribir libremente. ....	46
<b>Figura 56.</b> Diagrama de flujo de la aplicación agarre y lanzamiento de pelotas.....	49
<b>Figura 57.</b> Diagrama de flujo de la aplicación Clasificación de objetos por colores.....	51
<b>Figura 58.</b> Asignación de Tag Blue al objeto denominado bolo. ....	52
<b>Figura 59.</b> Diagrama de flujo de la aplicación Clasificación de objetos por formas.....	53
<b>Figura 60.</b> Diagrama de flujo de la aplicación explota globos con pinzas. ..	55
<b>Figura 61.</b> Diagrama de flujo de la aplicación atraviesa los aros. ....	57

<b>Figura 62.</b> Detección de posición de la mano Derecha mediante el dedo índice.....	58
<b>Figura 63.</b> Diagrama de flujo de la aplicación atraviesa los arcos.....	59
<b>Figura 64.</b> Diagrama de flujo de la aplicación pintar y colorear libremente.	60
<b>Figura 65.</b> Esquema general del sistema de Realidad Virtual.....	62
<b>Figura 66.</b> Movimiento para el Agarre de pinza .....	63
<b>Figura 67.</b> Movimiento para el Agarre de gancho .....	64
<b>Figura 68.</b> Clasificación de objetos por la forma y color .....	64
<b>Figura 69.</b> Pinza dedo pulgar con los cuatro dedos de la mano .....	65
<b>Figura 70.</b> Movimiento de Flexión, neutral y extensión de la muñeca.....	65
<b>Figura 71.</b> Movimiento de aducción de la muñeca .....	66
<b>Figura 72.</b> Movimiento de pronación y supinación de la muñeca.....	66
<b>Figura 73.</b> Movimiento grafoperceptivo. ....	67
<b>Figura 74.</b> Menú Principal .....	69
<b>Figura 75.</b> Aplicación 1: Agarre y Lanzamiento.....	70
<b>Figura 76.</b> Aplicación 2: Clasificación de objetos por colores .....	71
<b>Figura 77.</b> Aplicación 3: Clasificación de objetos por formas .....	71
<b>Figura 78.</b> Colores en las yemas de los dedos .....	72
<b>Figura 79.</b> Aplicación 4: Explota globos con pinza .....	72
<b>Figura 80.</b> Posición de las manos .....	72
<b>Figura 81.</b> Aplicación 5: Atraviesa los aros .....	73
<b>Figura 82.</b> Aplicación 6: Atraviesa los arcos .....	74
<b>Figura 83.</b> Aplicación 7: Muestras de imágenes.....	75
<b>Figura 84.</b> Aplicación 7: Aprende a colorear .....	75
<b>Figura 85.</b> Movimientos mediante el Test Box and Block físico vs virtual ...	77
<b>Figura 86.</b> Error de mano derecha e izquierda con N=25 .....	79
<b>Figura 87.</b> Agarre de pelota .....	80
<b>Figura 88.</b> Lanzamiento de pelota.....	80
<b>Figura 89.</b> Instrucciones del sistema virtual al usuario.....	81
<b>Figura 90.</b> Ingreso de datos del usuario.....	81
<b>Figura 91.</b> Movimientos del usuario en el sistema virtual.....	82
<b>Figura 92.</b> Interacción del niño al menú principal.....	82
<b>Figura 93.</b> Movimientos del sistema de realidad virtual.....	83
<b>Figura 94.</b> Test Block and Block físico y virtual.....	87



<b>Figura 95.</b> Movimiento agarre de gancho.....	87
<b>Figura 96.</b> Movimiento de agarre de pinza digital.....	88

## RESUMEN

El trabajo de titulación tiene como propósito implementar un sistema de realidad virtual para la rehabilitación motora fina en niños con la utilización de dispositivos de tracking óptico. Este sistema fue desarrollado en el software Unity3D con la integración de dos dispositivos como: Oculus Rift DK2 y LeapMotion, que hacen posible que el sistema sea intuitivo, didáctico y beneficioso para el usuario mejorando la coordinación viso-manual. El sistema es complejo ya que posee varias aplicaciones basadas en ambientes como: una habitación, un zoológico y un aula de clases, en las cuales se estableció tareas que el usuario debe realizar para terminar cada aplicación. El usuario con el sistema realiza varios movimientos de rehabilitación motora fina como: agarre de pinza o gancho, flexión y extensión de muñeca, pronación, supinación, aducción, pinza con dedos y pinza bidigital. Adicionalmente, el usuario recibe una estimulación auditiva mediante sonidos agradables en las aplicaciones y otros sonidos que indican acciones correctas e incorrectas de las tareas. Además, este sistema permite ingresar datos del usuario como: nombre, edad, años, el cual sirve para guardar los tiempos de culminación de cada aplicación, estos datos serán evaluados por el especialista para determinar el mejoramiento de la rehabilitación en cada usuario. Finalmente, este sistema fue probado con varios niños que presentan capacidades diferentes y aplicado un test de usabilidad SEQ, dando como resultado una buena aceptación del sistema de realidad virtual implementado.

### **PALABRAS CLAVE:**

- **REALIDAD VIRTUAL**
- **REHABILITACIÓN MOTORA**
- **DISPOSITIVO GESTUAL**
- **DISPOSITIVO ÓPTICO**

## **ABSTRACT**

The titling work intended to implement a virtual reality system for fine motor rehabilitation in children with the use of optical tracking devices. This system was developed in Unity3D software with the integration of two devices such as: Oculus Rift DK2 and LeapMotion, which make it possible for the system to be intuitive, didactic and beneficial for the user, improving visual-manual coordination. The system is complex since it has several applications based on environments such as: a room, a zoo and a classroom, in which tasks were set up by the user to complete each application. The user with the system performs several movements of fine motor rehabilitation such as: claw or hook grip, flexion and wrist extension, pronation, supination, adduction, finger grip and two-wire gripper. In addition, the user receives auditory stimulation through pleasant sounds in the applications and other sounds that indicate correct and incorrect actions of the tasks. In addition, this system allows to enter user data such as: name, age, years, which serves to save the completion times of each application, these data will be evaluated by the specialist to determine the improvement of rehabilitation in each user. Finally, this system was tested with several children who presented different abilities and applied a SEQ usability test, resulting in a good acceptance of the virtual reality system implemented.

### **KEY WORDS:**

- **VIRTUAL REALITY**
- **MOTOR REHABILITATION**
- **LEAPMOTION**
- **OCULUS RIFTDK2**

# CAPÍTULO I

## 1. INTRODUCCIÓN

### 1.1 Planteamiento del problema

Actualmente, en el campo de la medicina se describen problemas de motricidad, especificando que la motricidad fina en niños involucra movimientos precisos y de alta coordinación, en la cual intervienen todos los músculos pequeños. Para lo cual, se necesita de un sistema que sea capaz de fortalecer dichos músculos y a su vez permita desarrollar el sistema nervioso central. En estos movimientos es de suma importancia la utilización simultánea del ojo, mano, dedos, al realizar actividades tales como: cortar, pintar, colorear, etc.

Entonces, debido a diferentes factores biológicos o psicológicos los niños sufren comúnmente problemas de motricidad fina, la cual afecta directamente a su condición física. Además, los niños debido a sus habilidades motoras débiles evitan realizar tareas complejas de movimiento, debido al miedo o a quedar en ridículo ante otras personas, esto involucra la ejecución de diferentes programas de rehabilitación motora física basados en ejercicios de forma manual y subjetiva para mejorar su desempeño. Sin embargo, la mayoría de estos programas de rehabilitación requieren de la presencia de un profesional experto en problemas de motricidad, lo cual significa una inversión de alto costo y de una larga duración, a su vez provocan frustraciones a los usuarios al no poder cumplir con las expectativas requeridas.

Además, existe un gran porcentaje de niños que sufren de problemas de motricidad, esto implica menor capacidad para explorar diferentes entornos, interactuar socialmente y poder comunicarse con otras personas. Es decir, estos trastornos hacen que los niños sean considerados menos

capaces de forma intelectual ante los demás grupos sociales, impidiéndoles demostrar sus conocimientos.

Por lo tanto, es importante tratar de solucionar a tiempo los diferentes problemas de motricidad fina de los niños, para que el proceso de desarrollo natural no se vea interrumpido, ni obstaculicen otros procesos de aprendizaje.

## **1.2 Antecedentes**

El uso de la Realidad Virtual dentro del campo de la Rehabilitación Motora en usuarios con Daño Cerebral Adquirido (DCA), es un hecho científicamente contrastado en los últimos años. Los prometedores y satisfactorios resultados que se están obteniendo en esta línea de investigación (tanto a nivel cognitivo, como a nivel motor), en conjunción con los procesos terapéuticos tradicionales, proporcionan el auge de un nuevo hito dentro de la rehabilitación tradicional conocido como Virtual Motor Rehabilitation cuyo acrónimo es (VMR), describiéndolo como Rehabilitación Virtual Motora (RVM). (Albiol Pérez, 2014)

Por lo tanto, este sistema proporciona nuevas características aplicables en la Rehabilitación Tradicional. Diversos estudios han mostrado las ventajas de estos sistemas en la Rehabilitación. La RVM incluye por lo general un enfoque lúdico que involucra a los usuarios y minimiza el grado de aburrimiento, incrementándose su motivación y adherencia al tratamiento. La inclusión de la realidad virtual en los protocolos de rehabilitación de niños, ha generado diversas expectativas acerca de esta intervención innovadora en el área, buscando mejorar las competencias individuales y el desempeño motor de los niños.

Diferentes estudios han mostrado que la realidad virtual permite al niño, acceder a experiencias que de otra forma no podría realizar. Estos ambientes interactivos disponibles, ofrecen a los niños la oportunidad de practicar y experimentar movimientos sin el temor a la vergüenza o el riesgo de lesión.

Esto puede reflejarse en una mejoría en las aptitudes motoras y quizás en el sentido personal de autosuficiencia. En un ambiente virtual el usuario puede ser rehabilitado en un ambiente seguro y controlado que permite simular actividades de la vida diaria, lo que permite realizar aplicaciones que evalúen este tipo de actividades. (Salcedo Maldonado M. , 2011) (Rojas Ortiz, 2011)

Además, la RVM aporta herramientas susceptibles de ser utilizadas en casa del usuario; esto permitiría a la creciente población de usuarios crónicos una mayor recuperación y mantenimiento de las habilidades motoras recuperadas, minimizando la necesidad de rehabilitación en centros clínicos especializados.

### **1.3 Justificación e Importancia**

Actualmente las investigaciones en el campo de la informática, están relacionadas a diferentes aplicaciones basadas en realidad virtual que atribuyen a la rama de la medicina en el ámbito de rehabilitación. Es decir, esto implica que el ser humano pueda ver y entender más allá de la realidad, ya que es una simulación del mundo real generada con ayuda de un software y hardware específico a través de una interfaz humano-computador, la cual permite ser asociada a la rehabilitación física de usuarios que sufren con dificultades de motricidad fina ya que es una herramienta de ayuda rápida y efectiva para mejorar las habilidades motoras.

Por lo tanto, este proyecto es importante debido a que permite obtener un sistema de realidad virtual que contienen una serie de ambientes simulados interactivos y multidimensionales que ayudarán al desempeño de los niños que sufren de problemas de motricidad fina, el cual se diferencia de todos los demás trabajos presentados hasta la actualidad mediante artículos en las literaturas publicadas tales como SCOPUS, IEEE Xplore Digital Library, ELSEVIER, entre otros, en virtud de esto se propone obtener un sistema basado en videojuegos que faciliten la rehabilitación motora fina de niños que sufren problemas de motricidad utilizando el dispositivo LeapMotion, el cual

permite obtener el movimiento de las dos manos y Oculus Rift para la visualización del entorno virtual, esto difiere de los demás trabajos los cuales utilizan dispositivo ópticos que simplemente se ejecuta movimientos con una sola mano.

Es importante indicar que este proyecto se encuentra enmarcado en el proyecto de investigación "VRChild", con el código 2016-PIC-0017, en el cual la Universidad de las Fuerzas Armadas ESPE Extensión Latacunga lidera la coordinación del mismo.

## **1.4 Objetivos**

### **1.4.1 Objetivo General**

- Implementar un sistema de realidad virtual para rehabilitación motora fina en niños mediante dispositivos de tracking óptico.

### **1.4.2 Objetivos Específicos**

- Investigar sobre la rehabilitación motora fina en niños.
- Investigar los distintos entornos de desarrollo de realidad virtual a través de la recopilación de información de fuentes científicas.
- Implementar la interfaz de realidad virtual para complementar la rehabilitación tradicional.
- Desarrollar algoritmos de análisis y detección de movimientos de motricidad fina en las manos.
- Realizar pruebas funcionales del sistema de realidad virtual.

## **1.5 Variables de la investigación**

### **1.5.1 Variable Independiente**

Dispositivos de Tracking Óptico.

### **1.5.2 Variable Dependiente**

Sistema de realidad virtual.

## **1.6 Hipótesis**

Los dispositivos de tracking óptico permitirán implementar un sistema de realidad virtual para la rehabilitación motora fina en niños.



## CAPÍTULO II

### 2. MARCO TEÓRICO

#### 2.1 Antecedentes Investigativos

Actualmente, existen varios trabajos relacionados con el tema de rehabilitación para motricidad fina y gruesa, a través de la realidad virtual ya que en el campo de la medicina al utilizar estas nuevas tecnologías ayudan a mejorar la forma de rehabilitación a usuarios y personas que sufren diversos problemas que en ocasiones han afectado en parte su sistema nervioso, los cuales permiten desarrollar adecuadamente la parte de motricidad.

Entre algunas investigaciones que se han desarrollado en la parte de rehabilitación está: (Francisco Moreno, 2013) el cual desarrolla un sistema de rehabilitación para miembros superiores basado en videojuegos que permiten realizar varias actividades físicas. Además, en (Vargas Herrera, 2017) realiza un sistema de espacios virtuales en tiempo real, ayudando a mejorar a usuarios sus diversas capacidades de coordinación ojo-mano.

Por lo tanto, la realidad virtual en aplicaciones ayudan a mejorar la rehabilitación de la motricidad permitiendo que una persona se sienta satisfactoriamente bien, ya que para la rehabilitación cuenta con diversos juegos los cuales contiene una interfaz amigable para el usuario, teniendo una interacción óptima y una percepción del mundo virtual tan eficiente que causa una sensación de transportación a varios mundos.

#### 2.2 Fundamentación Teórica

El Sistema de Realidad Virtual para Rehabilitación Motora Fina que se propone, requiere de hardware y software tales como: dispositivos ópticos y tecnologías de inmersión con otros mundos como es la realidad virtual. También, se requiere una previa investigación de la rehabilitación motora fina

y las actividades que implican el mejoramiento motriz en niños. Las descripciones se muestran a continuación.

### 2.2.1 Realidad Virtual

La Realidad virtual se define como un entorno de desarrollo a través de los sistemas informáticos en tiempo real, y a su vez estos permiten crear diversos mundos irreales mediante varias escenas en las cuales contienen un sin número de cuerpos.

La Realidad Virtual genera una simulación tridimensional de varios entornos de desarrollo en el computador, permitiendo al usuario tener acceso a estos mundos virtuales con una total inmersión al estar en el interior de este mundo, así como pueden interactuar con todos los objetos del mismo. Esto hace que el usuario se desconecte del mundo real por un momento y se introduzca en el mundo virtual completamente, obteniendo una sensación acorde al mundo que observa la persona, cabe indicar que también se desconecta todos sus sentidos porque se sumerge completamente en el nuevo mundo (Escartín, 2017), (Facultad de Informatica de Barcelona, 2017).

Se puede ejecutar varias actividades tales como: conducir, disparar, etc., dentro del entorno virtual al mismo tiempo que se manipula objetos para cumplir determinadas tareas, obteniendo una preparación y entrenamiento de diversas formas, mejorando los diversos movimientos de motricidad de las personas (Difementes.com, 2017).



**Figura 1.** Realidad Virtual

## 2.2.2 Dispositivos de Tracking Ópticos

Son aquellos dispositivos que utilizan para su interacción sensores ópticos como pueden ser fotodiodos, LDR o fotorresistores, infrarrojos que sirven como retroalimentación para una mayor precisión óptica, obteniendo un sistema de posicionamiento adecuado (Gerro Prinsloo, 2015)

**Sensores Ópticos:** Son aquellos que pueden detectar objetos en función de un haz de luz, el cual es interrumpido por el mismo objeto que se está detectando (Montufar, 2012).

**Tracking ópticos:** Son dispositivos basados en luces que permiten establecer un control de orientación, posición de un objeto en base a ciertas características específicas, para que un objeto que se encuentra en constante movimiento permanezca dentro de un campo de visión. Además, están constituidos de varios LED's infrarrojos y de sensores que son cámaras que se encuentran repartidas por todo el dispositivo capaces de detectar infrarrojos (Carmen Ortigueira, 2017) (Pérez & Reinoso, 2003).

### 2.2.2.1 LeapMotion

El LeapMotion es un dispositivo basado en sensores para un control de movimiento gestual, ya que detectan con gran precisión todos los movimientos que se realizan con las manos, los dedos e incluidos objetos que se encuentren alrededor del campo de visión. Por lo tanto, debido a su tecnología se puede controlar varias interfaces en cualquier ordenador, sin la necesidad de tocar las pantallas, simplemente con la ejecución de movimientos en el aire. Además, a este dispositivo se le puede utilizar con otros dispositivos de entrada para desarrollar aplicaciones de movimiento, teniendo como consideración que se utiliza en computadoras que tengan un sistema operativo Mac OS X y Windows 7/8 (Michán, 2012) (Garrido, 2013).

LeapMotion está constituido por hardware y software que le permite interpretar cada uno de los movimientos.

**Hardware:** El dispositivo es pequeño teniendo como dimensiones: 75 mm de largo, 25 mm de ancho y 11 mm de alto. Además, está compuesto por dos cámaras que contienen sensores monocromáticos sensibles a la luz infrarroja, las cuales tienen como función capturar todas las imágenes. También, cuenta con tres LEDs que trabajan con una longitud de onda de 850 nm dentro del espectro de luz infrarroja, de esta manera son quienes iluminan mediante una luz infrarroja la zona de cobertura de manera uniforme, de igual forma dispone de un microprocesador el cual mediante un programa controla el LeapMotion, permitiendo regular toda la iluminación del dispositivo, así como también guarda la información que es enviada por los sensores para posteriormente ser enviada al driver o controlador que esté instalado en algún ordenador, finalmente cuenta con un controlador USB 3.0 que permite reconocer al dispositivo en un computador (Belda, 2015).



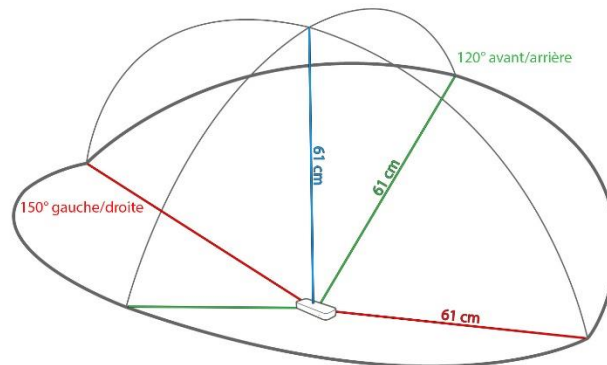
**Figura 2.** Hardware del dispositivo LeapMotion.

**Software:** Es en donde se recibe todos los datos del dispositivo, los cuales son examinados para adquirir toda la información de las manos, dedos y brazos. Ya que el software dispone de un modelo de una mano en forma interna, para posteriormente comparar con los datos que recibe del dispositivo logrando obtener una información de los movimientos lo más precisa posible (Almá, 2014).



**Figura 3.** Software del dispositivo LeapMotion.

**Zona de cobertura:** Es una semiesfera de radio de 61 cm, que depende del ángulo de visión que tienen los lentes de las dos cámaras, así como también de la intensidad máxima que entrega la conexión USB a los LEDs (Belda, 2015).



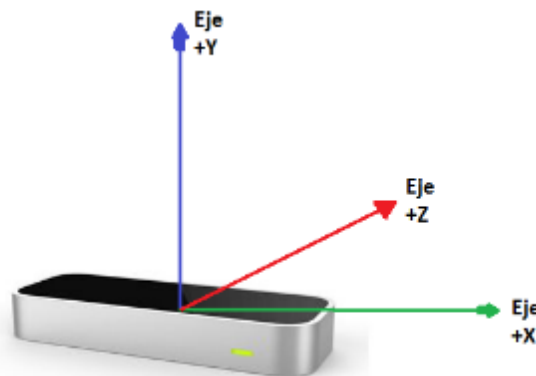
**Figura 4.** Zona de cobertura LeapMotion.  
**Fuente:** leap-motion, 2017

**Principio de funcionamiento:** Cuando se coloca las manos sobre el dispositivo estas son iluminadas, provocando que se refleje una luz en el dispositivo y a la vez incurra sobre las lentes de las dos cámaras, esto hace que los rayos se concentren en los sensores que posee cada cámara, por lo tanto, los datos que se recogen se almacenan en una matriz dentro de la memoria del controlador USB, a su vez se ajusta la resolución de cada imagen mediante el microcontrolador que posee el dispositivo y se envía al driver instalado en el computador. Los datos no son más que la representación de valores de intensidad luminosa que se le da a cada pixel de una imagen capturada, guardándola en un buffer. Estos valores se cuantifican a 8 bits para poder generar una imagen en escalas de grises (Belda, 2015).



**Figura 5.** Mapa de mallado de puntos de LeapMotion.

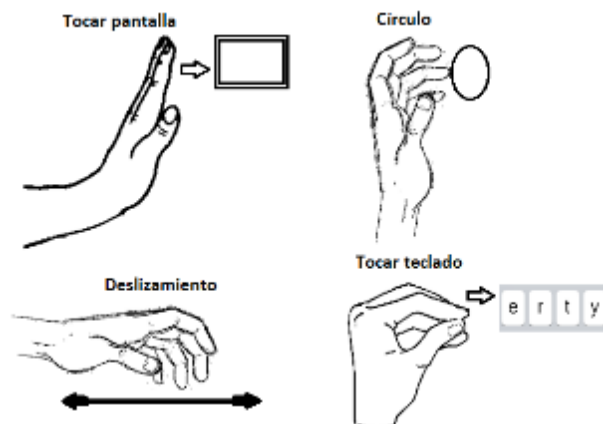
**Características de LeapMotion:** La información que proporciona el controlador se transmite con una frecuencia de hasta 100 Hz y con una precisión de hasta 0,01 mm en la detección de la posición de cada dedo y mano (Andrade).



**Figura 6.** Sistema de coordenadas de LeapMotion.

El SDK del dispositivo contiene clases que permiten la interpretación de los objetos capturados por el dispositivo, teniendo como unidad básica el Frame el cual contiene la información de: manos, dedos y gestos. Por lo tanto, el LeapMotion realiza un seguimiento predictivo debido a que utiliza un modelo de mano interno para interpretar todos los movimientos, esto ayuda a que cuando el controlador del dispositivo que provee la información no detecta alguna parte o la totalidad de un dedo, mano, todos estos caracteres son determinados a partir de datos anteriores para aproximar las posibles posiciones de las partes no visibles. Finalmente, el controlador puede detectar

cuatro gestos que son: un movimiento de círculo con un dedo, un gesto de deslizar, una acción de puntear sobre un plano vertical y un plano horizontal (Andrade).



**Figura 7.** Cuatro gestos que detecta el LeapMotion.

### 2.2.2.2 Oculus Rift DK2

Las Oculus Rift son gafas o cascos para realidad virtual ya que tiene la capacidad de proporcionar una inmersión totalmente real de entornos de escenas para la persona que usa las gafas. Además, la persona tiene la posibilidad de compartir todas las experiencias únicas y diferentes que adquirió con otras personas. Este dispositivo actualmente no solo es usado para el área de videojuegos, sino también está siendo desarrollado en aplicaciones que ayudan a mejorar el campo de la medicina, arqueología, educación, entretenimiento, cultura, entre otros (Gafas Oculus, 2017).



**Figura 8.** Dispositivo Oculus Rift DK2.

## Características del Oculus Rift DK2

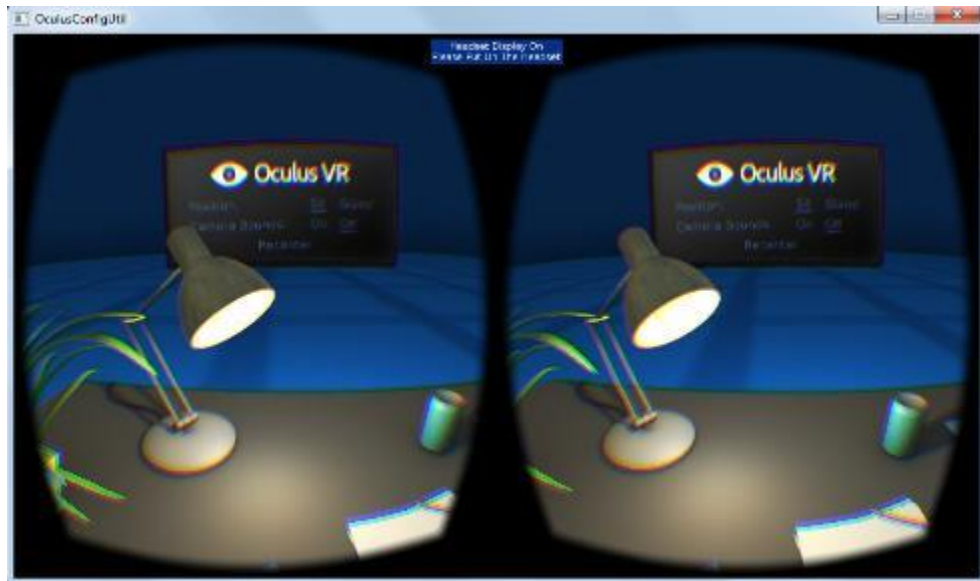
**Hardware:** Este dispositivo tiene un peso total de 380 gramos para ser utilizado como gafas sobre los ojos de cualquier persona ya que cuenta con un sistema de soporte para ajustar sobre la cabeza, también tiene una pantalla OLED de 5 pulgadas de baja persistencia, con una buena resolución de 960x1080 pixeles en cada ojo, logrando obtener un ángulo de visión de 100° en cada uno de ellos para tener una visión 3D en estéreo, la baja persistencia permite eliminar adecuadamente los desenfoques de movimientos y las posibles vibraciones que existan logrando obtener una imagen más estable e incrementando la sensación de presencia (ABC Tecnología, 2014), (Abubakra, Herrero, Fernández Villanueva, Martín Sanz, & Zabala Hidalgo, 2015)

También, permite el rastreo de movimiento de la cabeza con una latencia baja a través de una cámara externa con sensor CMO, la cual tiene la capacidad de capturar infrarrojos que se encuentran ubicados en las gafas para permitir un movimiento de 6 grados de libertad ya que utiliza una combinación de un giroscopio, un acelerómetro y un magnetómetro. Este dispositivo viene en una caja la cual consta también de dos pares de lentes de visión para ser intercambiados ya que permiten corregir las distrofias, cable USB, un adaptador HDMI, cable de alimentación con adaptadores. Además, el casco cuenta con un puerto UBS, un conector de audio y un dial a cada lado para ajustar la distancia de la pantalla (Abubakra, Herrero, Fernández Villanueva, Martín Sanz, & Zabala Hidalgo, 2015), (Blanchard Cruz, 2016)(Cruz, 2016).

**Software:** Es Para su utilización con un computador se debe instalar el runtime, el cual tiene una interfaz que sirve para configurar el tipo de lente que se utiliza, el espacio interpupilar, la altura de la persona y un demo de prueba para probar el dispositivo. Además, cuenta con una integración de motores gráficos que dan soporte para utilizar con varios programas como Unity, Unreal Engine, entre otros, así como también dispone de librerías de desarrollo de software (Blanchard Cruz, 2016).



Para la comunicación con un ordenador se realiza mediante el cable HDMI y USB 3.0. Sin embargo, para su funcionamiento es necesario que el ordenador cuente con un sistema operativo Windows 8, y en especial cuente con una buena tarjeta gráfica dedicada ya sea Nvidia GTX600 o una AMD Radeon de serie HD 7000 o superiores a esta, caso contrario si el computador no cuenta con estas especificaciones requeridas el Oculus Rift DK2 no funcionara y no podrá realizar ninguna aplicación (Gafas Oculus, 2017).



**Figura 9.** Comunicación correcta a través del demo de Oculus.

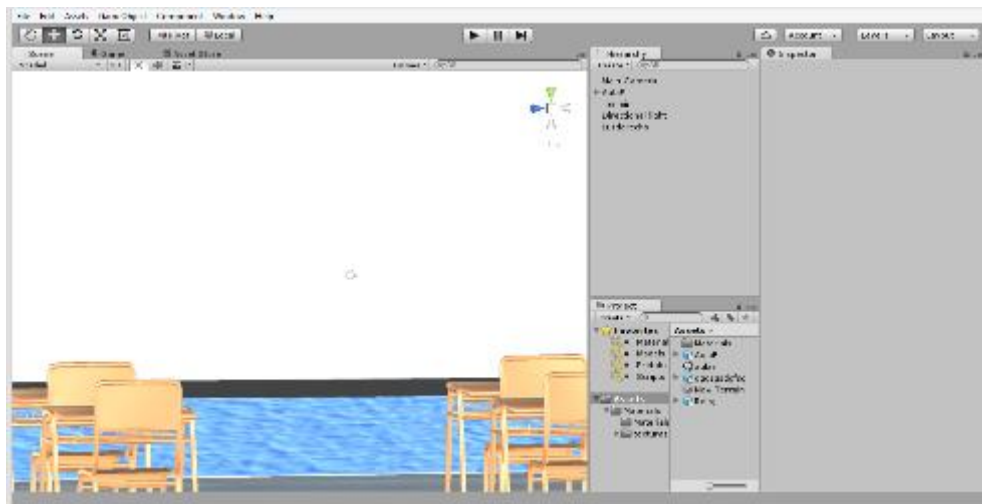
### 2.2.3 Motor Gráfico Unity

La plataforma de videojuegos Unity inició desde el año 2005 creado por la empresa Unity Technologies, el cual tiene como objetivo proporcionar un ambiente de RV (Realidad Virtual) o RA (Realidad Aumentada), es decir un entorno de desarrollo de juegos que tiene un conjunto completo de herramientas todas ellas totalmente intuitivas para la creación de contenido 2D y 3D multiplataforma. Está diseñada tanto para el uso de empresas como por desarrolladores independientes gracias a que se ha convertido en un gran motor de videojuegos totalmente integrado. (Hervás Torío, 2014).

Tiene una gran popularidad gracias a las licencias gratuitas y a la gran compatibilidad de la mayoría de dispositivos que existen en el mercado como por ejemplo: Windows, OS X, Linux, Xbox 360, PlayStation 3, Playstation Vita,

Wii, Wii U, iPad, iPhone, Android, Windows Phone 8 , Tizen, aplicaciones de la Windows Store, Mac ,reproductor web y navegadores de internet. De la misma forma, Unity está trabajando en compatibilidad para PlayStation 4 y Xbox One (Pérez Colado & Pérez Colado, 2014).

La función fundamental de todo motor de videojuegos en este caso de Unity es ofrecer un canal entre el usuario y la aplicación; está compuesto por un editor el cual ayuda al momento de diseñar una aplicación, es decir el entorno de desarrollo como se puede ver en la Figura 10, obteniendo así un desarrollo fácil, rápido e intuitivo a la hora de crear objetos. Sin embargo, al momento de realizar la animación se debe programar mediante scripts según desee él usuario (Fernández Arroyo, 2013).

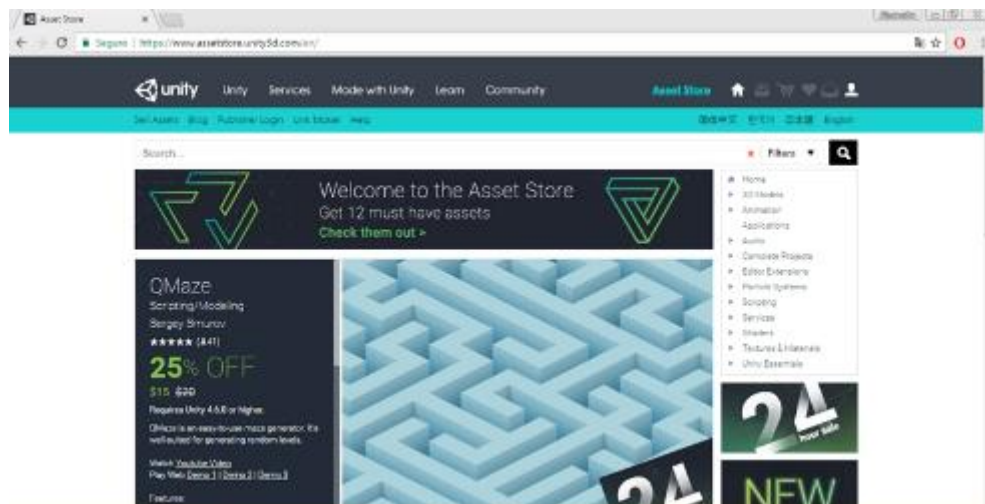


**Figura 10.** Editor de Unity3D

Unity tiene características similares a motores de videojuegos ya existentes, pero tiene una gran ventaja a la hora de la realización de diseños gráficos de objetos ya que permite la importación de gran cantidad de elementos desde diferentes programas de diseño como por ejemplo: Cinema 4D, Blender, 3DS Max, Adobe Photoshop, Adobe Fireworks, Cheetah3D, Modo, Maya, ZBrush, Softimage y Allegorithmic Substance.

Otra de sus ventajas que presenta unity es ofrecer a sus desarrolladores de forma gratuita o de pago diferentes recursos como proyectos, audio, modelos 3D, texturas, scripting, juegos, todos estos creados

por la comunidad de desarrolladores debido a que unity presenta un mercado interno conocido como Unity Asset Store, como se observa en la Figura 11, facilitando tareas a los desarrolladores. Por ello, la página oficial de Unity3D es una de las más completas y mejor estructuradas ya que existe diversa documentación facilitando un aprendizaje más sencillo y por cuenta propia (Soriano Maudes, 2012).

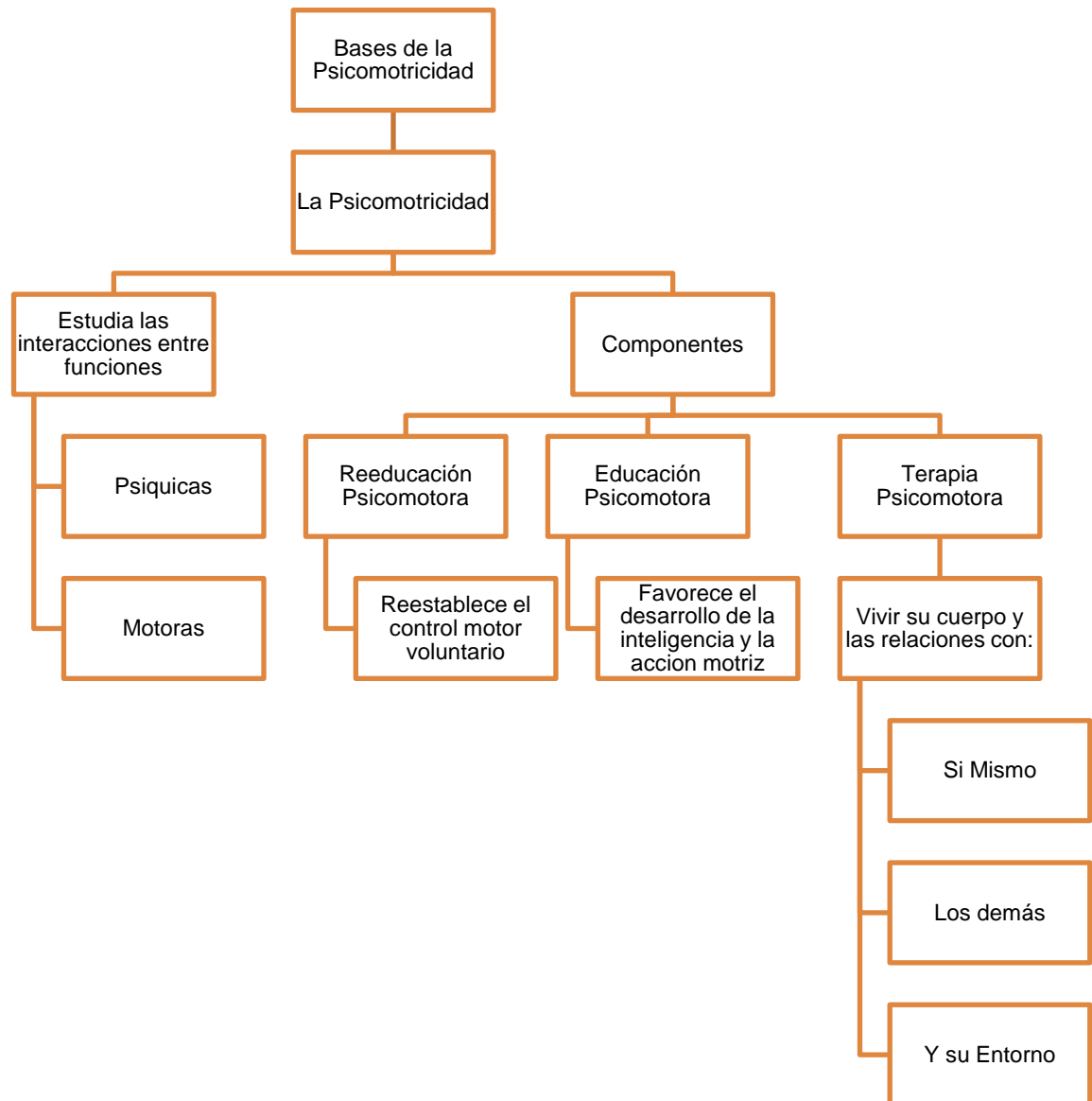


**Figura 11.** Interfaz de la Asset Store de Unity

Por lo tanto, Unity es uno de los motores de videojuegos que más acogida a tenido por los desarrolladores con un gran potencial a la hora de realizar aplicaciones de proyectos como lo son los videojuegos, cabe mencionar que más juegos son creados por esta plataforma por lo tanto más desarrolladores confían en sus herramientas y servicios logrando así ser un software de desarrollo de juegos dominante a nivel mundial.

#### **2.2.4 Motricidad Fina**

La motricidad humana evoluciona como consecuencia del desarrollo de la persona. Para empezar a hablar sobre la motricidad fina se debe conocer la diferencia entre los conceptos de psicomotricidad y motricidad. Por lo tanto, la psicomotricidad engloba todo lo referente a funciones motrices y funciones psíquicas(mente), mientras que la motricidad hace referencia a funciones motrices voluntarias (Berruezo Adelantado, 2008). A continuación se muestra un resumen de la psicomotricidad a través de un mapa conceptual, tal y como se muestra en la Figura 12.



**Figura 12.** Resumen de la psicomotricidad

*“La motricidad fina comprende todas aquellas actividades del niño que necesitan de una precisión y un elevado nivel de coordinación, es decir, una actividad armónica de partes que cooperan en una función, especialmente la cooperación de grupos musculares bajo la dirección cerebral”.* (Psicología de la educación psicomotriz, 1994)



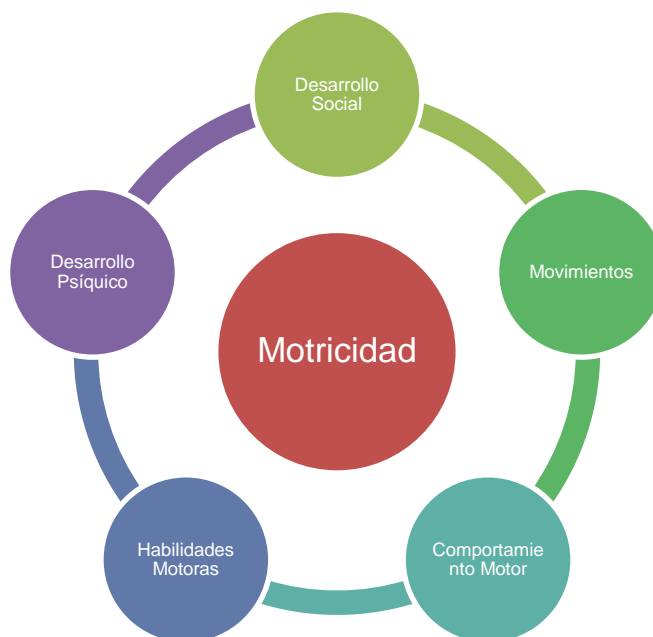
**Figura 13.** Actividades de motricidad

En otras palabras, la motricidad fina hace referencia a un conjunto de actividades habituales específicas que requieren una sincronización entre los grupos musculares de ejecución comandados por funciones cerebrales conscientes y voluntarias tales como parpadear, sonreír, tomar objetos, manipular objetos, utilizar herramientas, recortar, arrugar la frente, mover los dedos de los pies y manos, etc. Mismos que exigen un alto nivel de destreza y coordinación, los cuales llegan a desarrollarse con la práctica y el pasar del tiempo llegando así a un punto de madurez y desempeño óptimo de cada una de las actividades, generando un control de los diferentes niveles de motricidad existentes. (Cerón Méndez & Gutiérrez Botero, 2015)

Estas actividades son de vital importancia para que el individuo se pueda desarrollar con total normalidad en su vida cotidiana y relacionarse con su entorno. Para esto requiere en su mayoría la manipulación de objetos, expresiones corporales para el relacionamiento entre individuos así también la comunicación ya sea esta verbal o escrita, en tanto no sean atendidos estos problemas tendrá más dificultades de desarrollarse con normalidad y autónomamente, y a más dificultades que tenga el individuo en desarrollar con normalidad sus habilidades motoras, va a necesitar mucha más ayuda. Esto puede conllevar a problemas psicológicos y un deterioro aún mayor en sus habilidades. (Gil Madrona, Gómez Barreto, & Contreras Jordán, 2017)

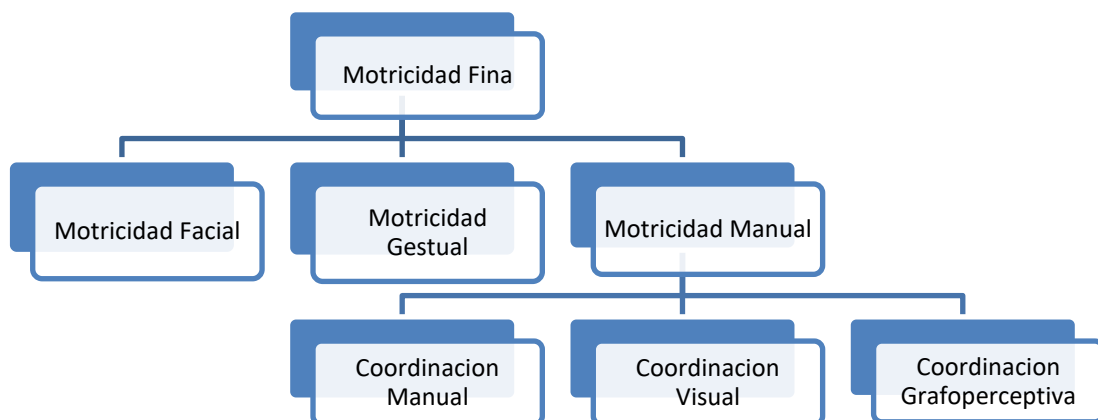
Implica un nivel elevado de maduración y un aprendizaje largo para la adquisición plena de cada uno de sus aspectos, ya que hay diferentes niveles

de dificultad y precisión”. La motricidad fina, micro-motricidad o motricidad de la pinza digital tiene relación con la habilidad motriz de las manos y los dedos. Es la movilidad de las manos centrada en tareas como el manejo de las cosas, orientada a la capacidad motora para la manipulación de los objetos, para la creación de nuevas figuras y formas, y el perfeccionamiento de la habilidad manual (Acaro Acaro, 2011).



**Figura 14.** Beneficios de la motricidad

La actividad motriz de la pinza digital y manos, forma parte de la educación psicomotriz del escolar. Su finalidad es la de adquirir destrezas y habilidades en los movimientos de las manos y dedos. A continuación se observa una clasificación de la motricidad fina. (Chuva Castillo, 2016)



**Figura 15.** Clasificación de la motricidad fina

### **Motricidad Fácil**

Son aquellas actividades que favorecen el aspecto motriz al interpretar los pensamientos y los estados de ánimo y así tener un dominio muscular y una flexibilidad, para lo cual es necesario comprender los gestos, las muecas, los guiños, las sonrisas, entre otros. El órgano que se debe desarrollar y ejercitar en esta etapa es la lengua con actividades de masticación, ejercicios de soplar, succión, leer, hablar, etc., ya que si se tiene alguna articulación imperfecta es debido a un mal desarrollo de la habilidad motriz generalmente que afecta a los órganos buco faciales. (Gandulfo & Young, 2011)

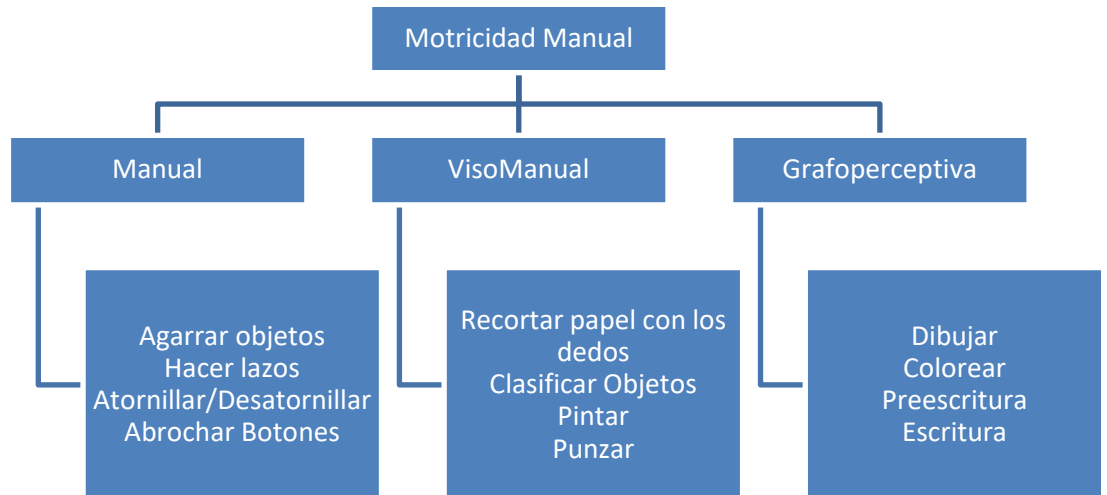
### **Motricidad Gestual**

Son todos los movimientos con los que se expresa una persona o se comunica, como son los gestos ya que son de una gran importancia para poder expresar los sentimientos, pensamientos o ideas y así desarrollar la parte comunicativa del ser humano, sin limitarnos a la expresión fácil. Favorece el aprendizaje al realizar:

- Situaciones con lenguaje gestual
- Situaciones de apoyo del lenguaje gestual al verbal
- Verbalización de una interpretación
- Imitación de personajes con dramatizaciones

### **Motricidad Manual**

Son aquellos movimientos en los que interviene la coordinación entre las manos y la visión, los pies y la visión o combinación de los mismos. Para esta etapa se hace referencia a actividades con movimientos de la mano, ya sea derecha o izquierda, sin ser necesario que se utilice las dos manos a la vez, también incluyen movimientos de la muñeca, el antebrazo, el brazo y el hombro. Esto favorece al desarrollo y destreza de una mayor flexibilidad y agilidad motriz. A continuación se muestra las actividades que se puede realizar en la coordinación manual, coordinación viso-manual y en la coordinación grafoperceptiva. (Rodríguez Sánchez, 2014)



**Figura 16.** Actividades de la Motricidad Manual

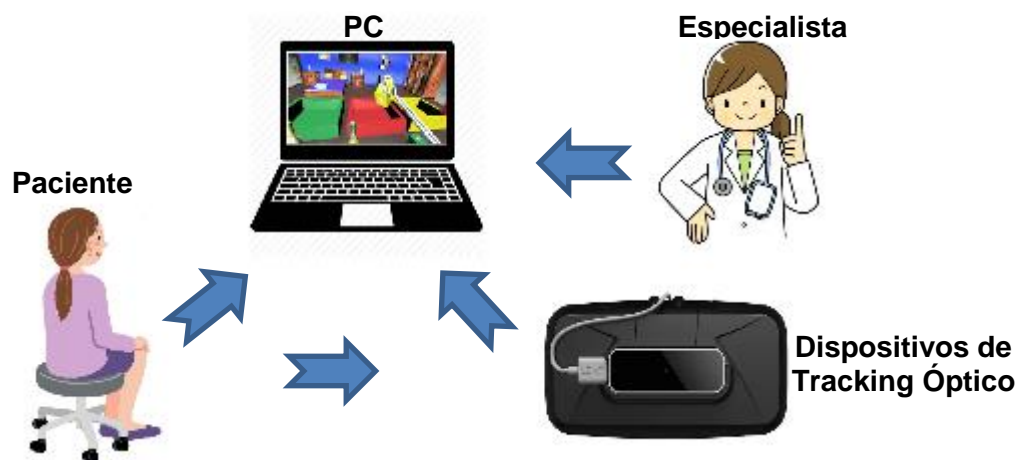
Por lo tanto, para conseguir un resultado adecuado en la rehabilitación de las manos es de suma importancia mantener un constante desarrollo en las habilidades motrices. Esto va a depender también en gran medida de la ayuda visual y táctil que se maneje. La motricidad fina en su forma más esencial conlleva un aprendizaje de un largo tiempo hasta llegar a un punto de maduración de la habilidad motriz, para lo cual es de suma importancia realizar la rehabilitación lo más coordinada y precisa posible de una manera constante, sólo así se podrá lograr el objetivo. Las habilidades motoras básicas que se busca mejorar son: agarre, traslado, alcanzar, soltar voluntariamente, manipulación en la mano y actividad bimanual.



## CAPÍTULO III

### 3. DIMENSIONAMIENTO DE HARDWARE Y SOFTWARE

En este capítulo se va a hablar sobre el dimensionamiento de hardware y software que necesita el sistema de realidad virtual, por lo tanto, para tener un buen dimensionamiento de los dispositivos es importante conocer principalmente cuáles son los elementos que conforman el sistema virtual, tal y como se muestra a continuación.



**Figura 17.** Esquema del usuario y el sistema virtual

En el esquema se observa que el sistema está conformado por el usuario, una computadora, el especialista y los dispositivos de tracking óptico (LeapMotion y Oculus Rift).

#### 3.1 Análisis del Software Unity

Unity es considerado uno de los motores gráficos de videojuegos más potentes en la actualidad, para determinar si su funcionamiento es correcto en este proyecto se planteó las siguientes preguntas: ¿Qué juego se planea hacer?, ¿Es un juego para celulares?, ¿Se va a realizar en 2D o 3D?, ¿Es compatible con Oculus Rift y LeapMotion?, ¿Qué licencia se necesita para el proyecto? ¿Qué características debe tener la computadora para un buen funcionamiento?, ¿Qué lenguaje de programación se utiliza? Para contestar estas preguntas se realizó una comparación entre motores gráficos mostrada en la tabla 1.

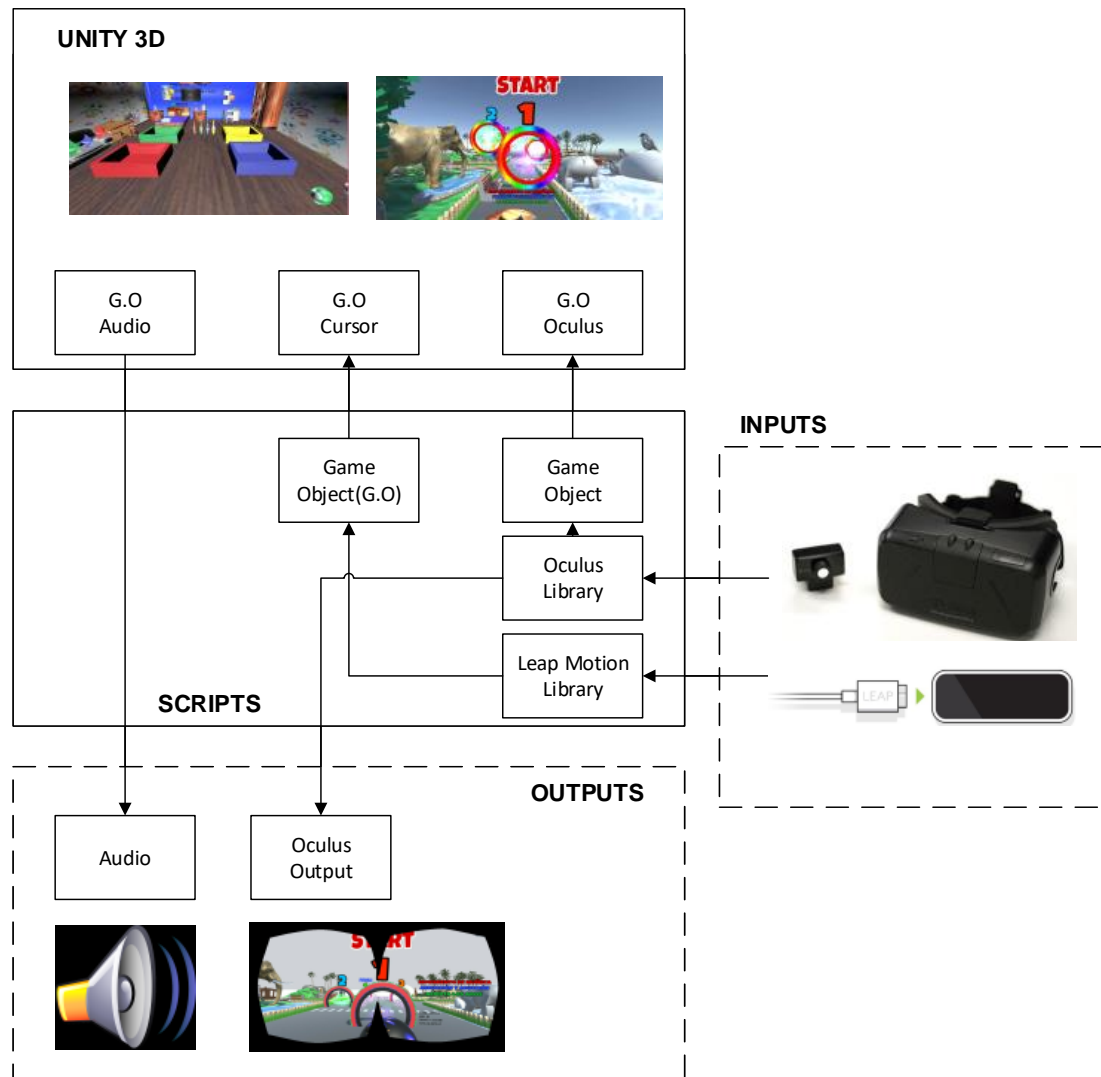
**Tabla 1.**  
**Comparación de motores gráficos**

Especificaciones	Motores Gráficos				
	Unity 5	Corona SDK	Gamemaker	Unreal Engine 4	CryEngine 5
<b>Plataformas</b>	La mayoría de las plataformas: iOS, Android, PC, PlayStain, etc	iOs, Android y Kinle Fire	iOs, Andrioid, PS3, PS4, Steam	La mayoría de las plataformas: iOs, Android, PS, Steam	PC, PS4, Xbox One
<b>Licencias</b>	Versión gratuita con limitaciones y de pago	Versión gratuita casi completa y de pago	Versión gratuita y de pago	Versión gratuita con acceso a todo SDK y de pago	Versión gratuita y de pago(lo que quiera o si se desea pagar)
<b>2D/3D</b>	Ambos	2D	2D	Ambos	3D
<b>Conocimientos de Programación</b>	Sí, existe mucha documentación en la página oficial	Sí	No, pero es útil	Si	Si
<b>Lenguaje de Programación</b>	C# y Java Script	Lua	GML	C++	C++
<b>Velocidad de desarrollo</b>	Medio	Alto	Alto	Medio	Lenta
<b>Potencia</b>	Alta	Media	Baja	Muy Alto	Alta
<b>LeapMotion</b>	Sí	Sí	Sí	Sí	Sí
<b>OculusRift DK2</b>	Sí	No	No	Sí	Sí

Realizando el análisis se observa que el motor gráfico Unity es la mejor opción para este proyecto de investigación debido a su tecnología, funcionalidad, la gran compatibilidad de plataformas y al conocimiento que se tiene en lenguaje de programación C#.

### 3.2 Conexión de dispositivos de entrada y salida del sistema

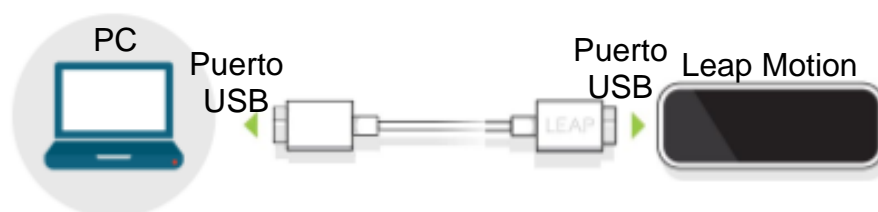
En esta sección se explicará las conexiones y configuraciones de los dispositivos de entrada y salida necesarios para el sistema de realidad virtual, para lo cual se ha realizado un diagrama general del sistema, tal y como se muestra en la Figura 18 y así proceder a desglosarlo de forma individual.



**Figura 18.** Diagrama general del sistema

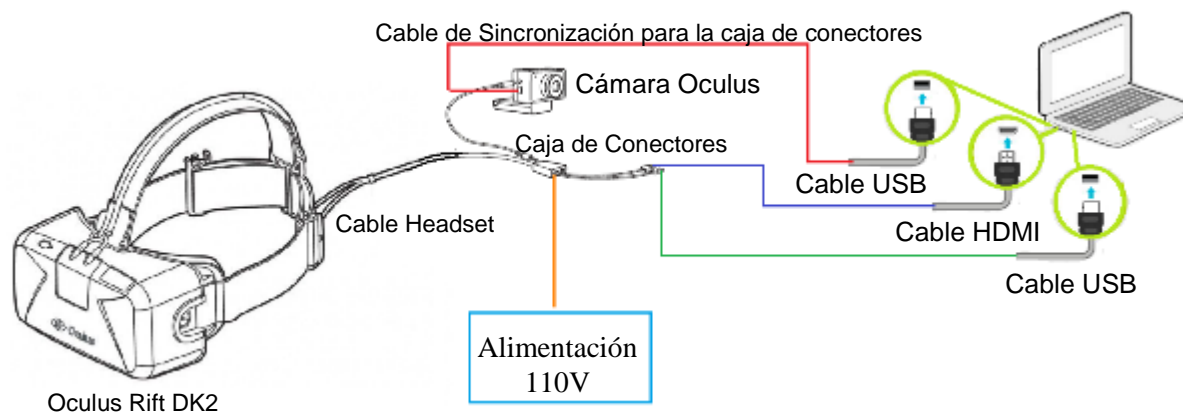
En este sistema de realidad virtual los dispositivos periféricos de entrada son las gafas Oculus Rift Dk2 (Casco HMD-Head-Mounted Display) y el dispositivo LeapMotion (Dispositivo de tracking óptico).

La entrada de datos de las diferentes posiciones de las manos que el usuario realiza en tiempo real, se realiza a través del controlador LeapMotion, el cual es conectado hacia la computadora a través de un puerto USB, como se muestra en la Figura 19.



**Figura 19.** Conexión cable USB LeapMotion-PC

El casco HMD (Oculus Rift) se conecta mediante un cable HDMI y USB a la computadora como se puede ver en la Figura 20, éste a través de los sensores que tiene incorporado capta el movimiento de la cabeza del usuario, estos movimientos son transferidos a la computadora por una cámara externa infrarroja mediante un cable USB al entorno virtual realizado en el software Unity y en los lentes del casco HMD se genera los entornos virtuales del sistema de realidad virtual.



**Figura 20.** Conexión Casco HMD-PC

### 3.3 Integración de dispositivos del sistema en Unity

En esta sección se detalla una explicación de la integración de los dispositivos Oculus Rift y LeapMotion con el motor gráfico Unity el cual es utilizado para el desarrollo de entornos virtuales necesarios en el sistema de realidad virtual para rehabilitación motora fina. La versión que se utilizará en este trabajo de investigación es Unity 5.3.2f1.

#### 3.3.1 Integración Oculus Rift

En primer lugar, es necesario instalar y configurar el runtime de Oculus Rift en la computadora, tal y como se detalla en el Anexo C. La versión que se utiliza es la V0.8.0.0 puesto que ésta versión brinda una estabilidad en la computadora utilizada. Para el desarrollo de este proyecto de investigación se cuenta con una computadora con tarjeta gráfica AMD RADEON, sistema operativo Windows 8.1 y un procesador Intel Core i7, debido que Oculus Rift

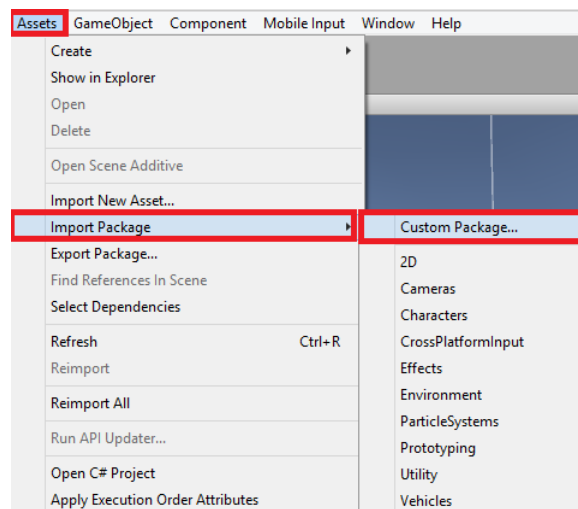
Dk2 necesita de estas especificaciones mínimas para tener un buen funcionamiento al momento de ser utilizado.

Desde la versión de Unity 5.1 la integración de Oculus Rift con Unity no necesita de Unity Packages para el uso en el entorno ya que lo realiza de forma automática. Entonces, para comprobar la integración dentro del entorno de Unity se debe acceder a la pestaña Inspector, la cual para su visualización se encuentra en la siguiente la ruta de acceso: Edit/ Project Settings/ Player.

Al aparecer la pestaña Inspector seleccionar en la sesión “Other Settings” la opción “Virtual Reality Supported”. Una vez terminado lo anterior, se debe ejecutar la escena y el usuario podrá observar en el HMD todo el entorno creado en Unity.

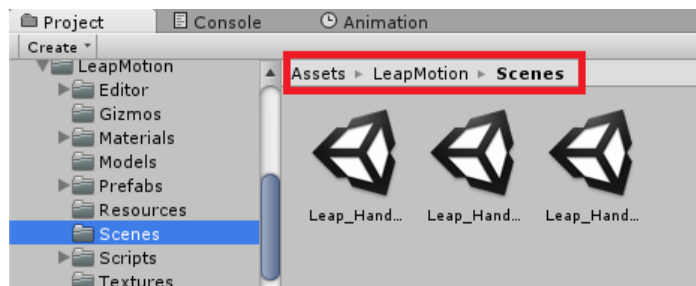
### 3.3.2 Integración LeapMotion

Para la conexión entre el dispositivo LeapMotion y el software Unity, es imprescindible instalar y configurar el runtime del controlador LeapMotion el cual esta detallado en el Anexo C. Es necesario descargar Unity Core Assets, que contiene plugins, scripts y prefabs y así lograr la integración del dispositivo LeapMotion con el entorno de desarrollo de Unity. Una vez descargado de la página de LeapMotion el Unity Package se debe importarlo tal y como se muestra en la Figura 21.



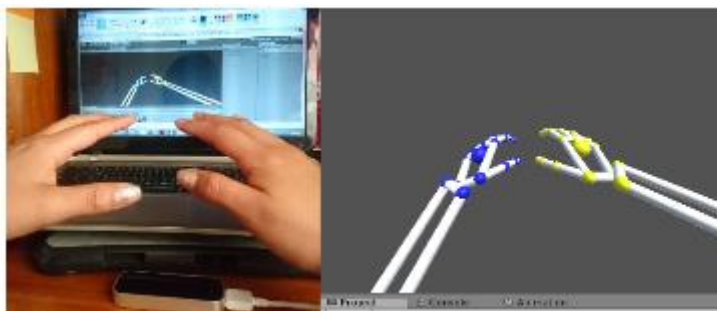
**Figura 21.** Importación de Unity Package

A continuación se debe verificar que el paquete este importado correctamente, para lo cual se debe acceder a la carpeta Scenes como se muestra en la Figura 22 y seleccionar la escena Leap\_Hands\_Demo\_VR que se utiliza para realidad virtual con las gafas Oculus Rift DK2.



**Figura 22.** Ruta de acceso a la carpeta Scenes

Para observar las manos dentro del entorno virtual se debe ejecutar Unity. Al acceder al entorno virtual se observa un modelo 3D de las manos del usuario el cual se moverá en tiempo real, como se observa en la Figura 23.



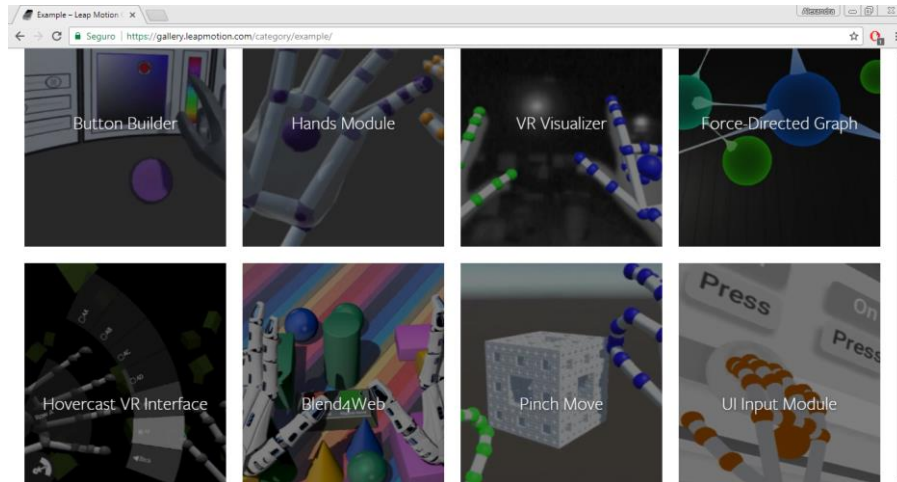
**Figura 23.** Prueba de funcionamiento LeapMotion en Unity

### 3.4 Integración de módulos de LeapMotion a Unity

En esta sección se detalla una explicación de la integración de los diferentes módulos de LeapMotion utilizados en este proyecto de investigación, los cuales serán descargados previamente desde la página oficial de LeapMotion y son:

- Attachments
- DetectionExamples
- InteractionEngine
- UIInputModule

Para descargar los diferentes módulos se accede a la dirección <https://gallery.leapmotion.com/category/example/> y se da clic en el módulo que se desee descargar, tal y como se muestra en la Figura 24.



**Figura 24.** Ejemplos de módulos de LeapMotion

Una vez descargados todos los módulos se procede a ejecutar cada uno de ellos, la integración estará terminada cuando en Unity se observe en la carpeta LeapMotionModules todos los módulos mencionados anteriormente.

### 3.4.1 Módulo Attachments

El módulo Attachments proporciona un modelo de mano la cual sirve para realizar las diferentes tareas dentro del entorno virtual, por ejemplo: controlar interfaces de entrada, colocar objetos, cambiar tamaño de objetos entre otros con los diferentes comandos de la mano. Este módulo permite separar la lógica y la física del juego, dando así la opción de utilizar los Prefabs de este módulo sin tener que construir unos nuevos.

El módulo Attachments contiene los siguientes Prefabs:

- HandAttachments: modelo principal de mano
- Transition: animación de Attachments

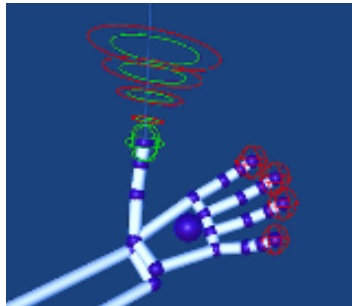
Y los siguientes Scripts:

- AttachmentController: activa y desactiva los Attachments
- CameraFollower: objeto gira para que siempre esté frente a la cámara

- HandAttachments: expone los Attachments a una transformación
- ITransition: base para las transiciones
- Transition: anima una transformación y el color de un objeto cuando se llama a la función AttachmentController.

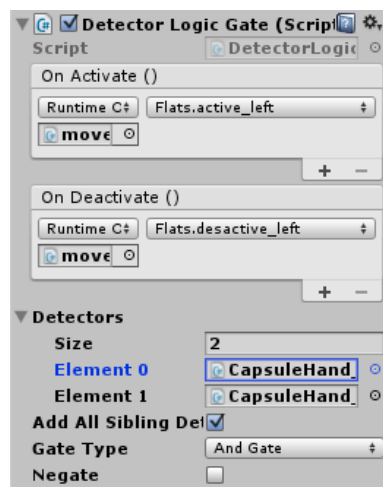
### 3.4.2 Módulo Detection Examples

Este módulo ayuda a detectar lo que está haciendo la mano, es decir puede detectar por ejemplo cuando los dedos de la mano están extendidos o recogidos, cuando la palma este apuntando hacia abajo, arriba, al frente o hacia atrás, cuando un dedo este apuntando hacia alguna dirección o si la mano o los dedos estén cerca de algún objeto.



**Figura 25.** Detectors del módulo Attachments

Estos detectores se pueden combinar con una compuerta lógica llamada Detector Logic Gate, el cual combina dos o más detectores para determinar su propio estado, como se muestra en la Figura 26.



**Figura 26.** Detector Logic Gate en Unity



Un ejemplo de la combinación de estos detectores sería la combinación de la dirección de la palma con la dirección del dedo índice, tal y como se observa en la Figura 27.

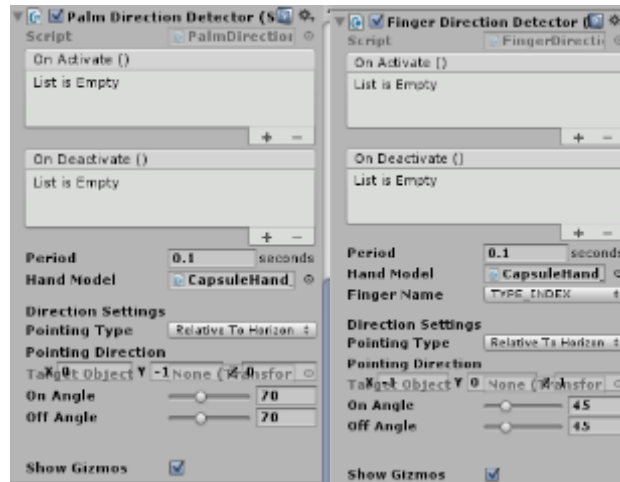


Figura 27. Combinación de detectores

### 3.4.3 Interaction Engine

Este módulo ayuda a la interacción entre el motor gráfico Unity y la física de las manos del mundo real con objetos creados dentro del entorno de Unity, se encarga de la Física del objeto en tiempo real, detección de agarre, colisión entre objetos, etc. Los componentes básicos de interacción se encuentran mostrados en la Figura 28 y están ordenados según el orden de jerarquía a los que deben estar expuestos.

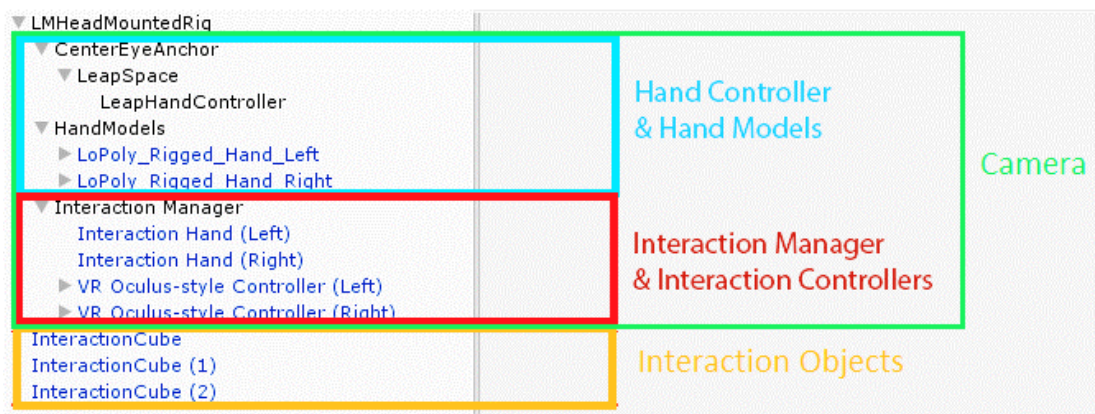


Figura 28. Clasificación de componentes de interacción

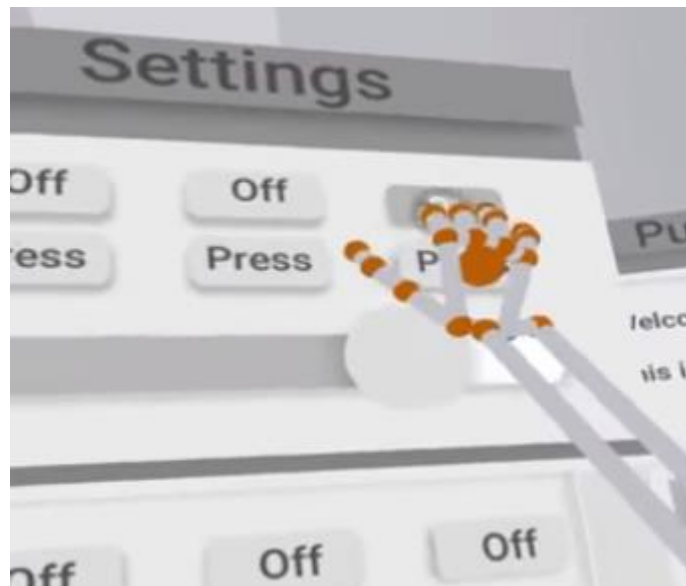
**Interaction Manager:** administra toda la lógica interna de Unity y hace posible las interacciones incluyendo la actualización de datos de cada mano y la actualización de datos de objetos de interacción.

**Interaction Controller:** realiza la interacción real con los objetos de interacción, es decir tocar, agarrar, acercar, etc.

**Interaction Objects:** los objetos pueden estar o no dentro de la plataforma de cámara del controlador de LeapMotion.

#### 3.4.4 Módulo de UI Input Module

El módulo de UI Input Module es un módulo de entrada de interfaz de usuario, es decir, permite manipular objetos estándares al dar clic en un botón, una tecla, un joystick, un mouse, una entrada personalizada, etc., tal y como se muestra en la Figura 29 y se basa en la lógica del sistema y su componente principal es el LeapEventSystem.



**Figura 29.** Manipulación de un botón estándar

Este módulo de entrada de datos es utilizado específicamente en desarrollos y diseños de menús para entornos virtuales ya sea en 2D o 3D.

## CAPÍTULO IV

### 4. DESARROLLO DE LA INTERFAZ

Para mejorar la motricidad fina en niños se vio la necesidad de implementar un sistema de rehabilitación virtual, el cual dispone de varias aplicaciones implementadas a base de juegos interactivos, los mismos que permitirán a los usuarios mejorar sus habilidades motoras y su concentración al ejecutar cada una de las aplicaciones. Para el desarrollo es indispensable conocer cómo se diseñó cada uno de los elementos que posee los entornos en 3D en el software Blender, los mismos que son agrupados para darles una mejor visión al ambiente y conjuntamente con la programación complementar las tareas para cada movimiento en Unity3D.

Para el desarrollo de scripts es importante analizar los algoritmos que existen en los módulos del LeapMotion, los cuales ayudan a definir posiciones, detectar movimientos, interacción de objetos, entre otros. Por lo tanto, para la ejecución de tareas en cada aplicación de este sistema, se ha analizado todos estos algoritmos y se ha creado nuevos algoritmos de programación para complementar las funciones en base a la temática de cada aplicación, estos algoritmos realizarán tareas de detección de colisiones entre objetos, identificación de objetos por colores o formas, detección de pinzas con dedos, detección de posiciones de manos, pinza bidigital con la mano derecha. Cabe indicar que estos algoritmos son colocados en varios GameObject, los mismos que analizan si se ha realizado las tareas especificadas en cada código de programación y en función de su acción correcta o incorrecta procede a ejecutar el siguiente algoritmo que permitirá en algunas aplicaciones la aparición y desaparición de objetos, cambio de la posición de objetos en X, Y, Z, entre otros.

Además, el sistema posee nueve aplicaciones las mismas que están divididas en cinco entornos diferentes y estos a su vez contienen varios entornos, estos son:

- Menú de Datos
  - Aplicación inicio: Ingreso de datos del usuario
- Menú juegos virtuales
  - Aplicación menú: Menú principal de los juegos
- Interfaz Cuarto
  - Aplicación 1: Agarre y lanzamiento de pelotas
  - Aplicación 2: Clasificación por colores.
  - Aplicación 3: Clasificación por formas
- Interfaz Zoológico
  - Aplicación 4: Explota globos con pinzas
  - Aplicación 5: Atraviesa los aros
  - Aplicación 6: Atraviesa los arcos
- Interfaz Aula
  - Aplicación 7: Pinta y escribe libremente

En este apartado se detalla con claridad cómo se desarrolló todo el sistema paso a paso para el buen funcionamiento.

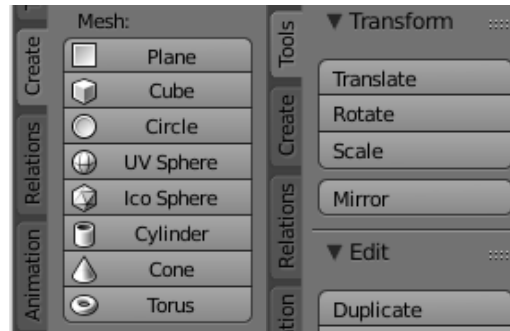
#### 4.1 Diseño de objetos 3D en Blender

Para empezar con el diseño 3D de cada aplicación se debe crear los modelos necesarios que van a ser visualizados en cada entorno del sistema, los cuales cuenta con varios elementos. Para ello, se debe descargar previamente el software Blender 2.78b. Una vez descargado e instalando se da doble clic en el icono y se abrirá una pantalla tal y como se muestra en la Figura 30.



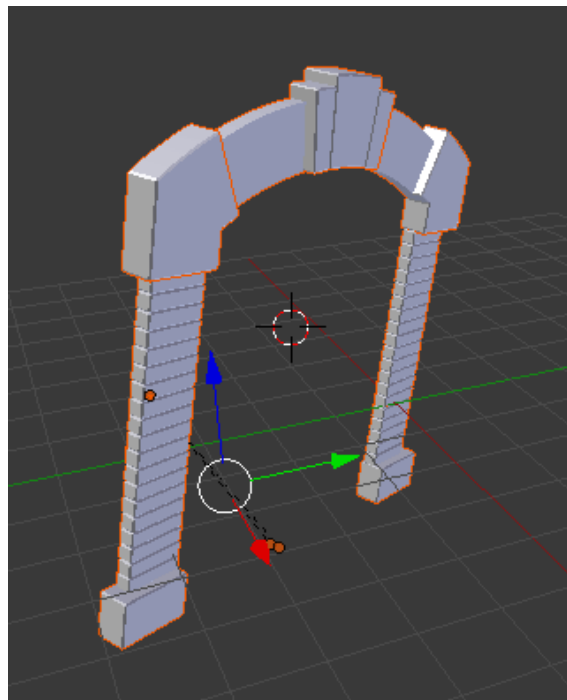
**Figura 30.** Pantalla inicial de Blender.

Utilizando las figuras básicas (círculo, triángulo, rectángulo, etc.) y las herramientas (rotación, escalado de objetos, duplicado, modo de edición, agrupación de objetos, etc.) que ofrece el software Blender permite crear modelos más complejos que ayudan al diseño de cada aplicación del sistema virtual.



**Figura 31.** Figuras básicas y herramientas

Los modelos 3D creados para este sistema son: estructuras de cada entrono, estructuras adicional para decoración, animales y objetos de interacción. Los cuales necesitan ser ensamblados mediante la utilización de objetos básicos, consolidando su unión mediante las funciones de agrupación y des-agrupación de objetos, como por ejemplo en la construcción del arco para el menú principal como se muestra en la Figura 32.



**Figura 32.** Montaje de modelo 3D finalizado.

Una vez finalizado cada modelo 3D se debe exportar a un formato compatible con Unity (.obj y .fbx), para esto se dirige a la pestaña File/Export y se selecciona el formato que dice .fbx u .obj. Teniendo en cuenta que el formato .fbx permite manipular el objeto por capas de modelado, mientras que el formato .obj solo permite manipular el objeto como una sola capa de modelado común. Después, se elige la ubicación en donde se pretende guardar el objeto y a su vez se da un nombre al archivo que se va a exportar.

## **4.2 Diseño de aplicaciones en el software Unity**

La Software Unity permite desarrollar múltiples video juegos ya sea en 2D y 3D, a la vez que se complementa con otro software en este caso Blender para la parte de diseño gráfico en los entornos. A continuación se detallara como se procedió a la creación de cada entorno virtual.

### **4.2.1 Diseño del menú de ingreso de datos del usuario**

En esta aplicación se crea un menú de ingreso de datos, el mismo que contiene títulos, botones, etiquetas y entradas de escritura. Para insertar cada elemento se debe seguir los siguientes pasos:

1. Ir a la pestaña GameObject/UI/Canvas. Este canvas es un rectángulo que contiene a todos los elementos que forman parte de la Interfaz de Usuario, además de estar renderizado en el espacio de pantalla para cuando se cambie de tamaño, este canvas cambie automáticamente.
2. Para añadir una imagen, se debe dar clic derecho sobre el Canvas y seleccionar UI/Raw Image. Este objeto de imagen será un hijo del canvas, por lo tanto la imagen debe ser arrastrar dentro de Texture.
3. Para insertar un botón hay que dar clic derecho sobre el Canvas/UI/Button. En el inspector en la opción Text se escribe el nombre que se va a visualizar en el botón, en este caso se asigna Guardar.

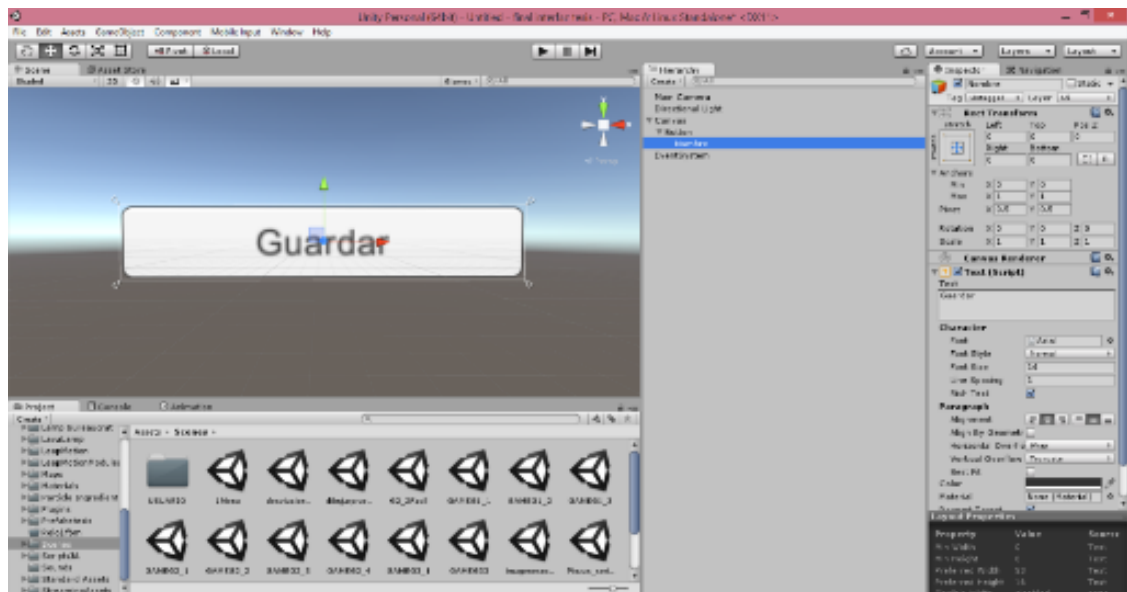


Figura 33. Añadir un botón a la escena.

Para insertar un campo de entrada de datos se debe seleccionar Canvas/UI/TextField. Luego, para escribir un texto, de igual manera se elige Canvas/UI/Text, y en la casilla texto se escribe títulos, subtítulos, nombres, etc., de igual manera en el inspector se puede elegir fuente, tamaño, entre otras características propias de escritura.

Mediante la implementación de varios botones, textos, imágenes, campos de entradas se obtuvo la primera aplicación del sistema de Realidad Virtual.



Figura 34. Ingreso de datos del usuario.

#### 4.2.2 Diseño del menú principal de los juegos

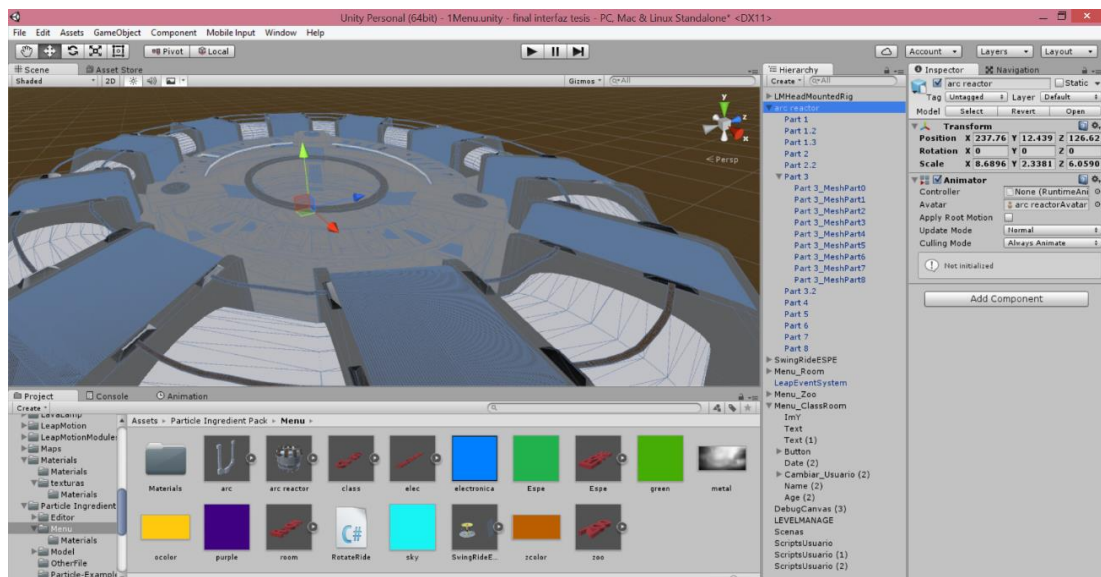
Este entorno se diseñó con la implementación de varios objetos previamente diseñados en el software Blender, los mismos que fueron importados a Unity para su correcta utilización. Para la creación del menú de juegos se realizó los siguientes procedimientos:

Se colocó todos los objetos de tipo prefabs a ser utilizados en la escena en una carpeta denominada Menú. Todos estos objetos no tienen ninguna textura son de color blanco y negro.



**Figura 35.** Objetos 3D para la aplicación Menú principal

Para utilizar cada objeto en Unity, se debe simplemente arrastrar el objeto a la escena y automáticamente se crea el GameObject que contiene todas las partes internas del objeto en la pestaña del Hierarchy.



**Figura 36.** Plataforma de la base añadida en el entorno virtual.



Para poner un color o textura a cada objeto se debe:

- **Color:** crear un material, este debe ser guardado en cualquier carpeta que el diseñador lo requiera, en este caso la carpeta se llama Materials, dar clic derecho en la carpeta y seleccionar Create/Material. El material se crea como una esfera y se le asigna un nombre como por ejemplo: Green y para seleccionar su color, en la pestaña del Inspector se da clic y se busca el color que se requiera.
- **Textura:** Se puede diseñar varias formas utilizando la herramienta Paint y guardarlo como .jpg o descargarse desde el internet texturas. Para agregar una textura, se debe seleccionar el objeto al que se requiere añadir, luego se arrastra la textura a la pestaña del Inspector y automáticamente se añade como un material.

Se colocó una plataforma circular como base y sobre este se agregó tres arcos y una rueda. Sobre la rueda se agregó varias letras formando las palabras **ELECTRONICA ESPE**, de igual manera sobre cada arco se colocó los nombres que identifica a cada entorno como es **ROOM** (Cuarto), **ZOO** (Zoológico), **CLASRRROM** (Clase). A todos estos objetos se les añadió un color adecuado.

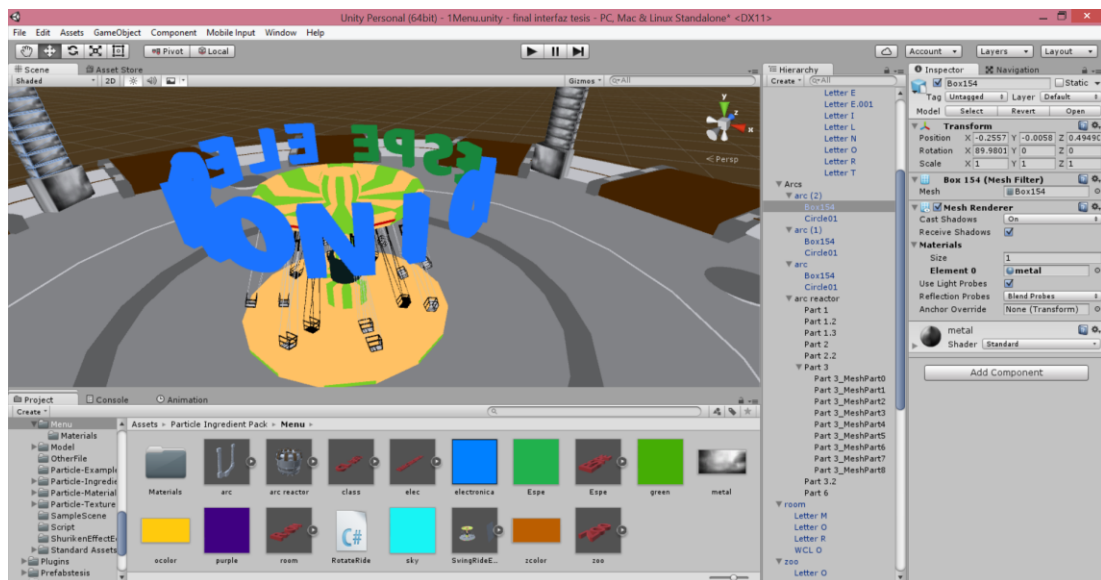
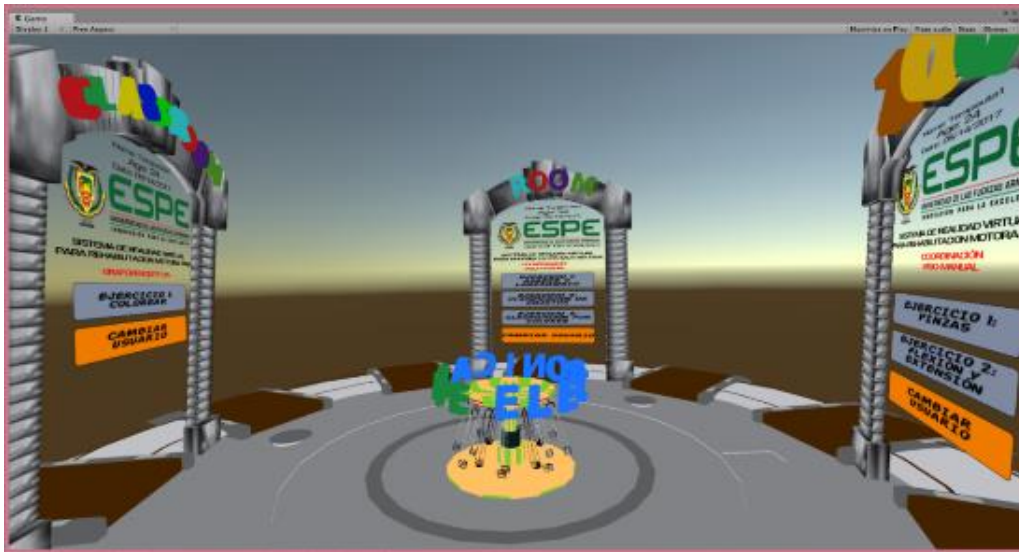


Figura 37. Plataforma circular con rueda y letras.

Para este Menú, en cada uno de los arcos se colocó diferentes botones, textos, imágenes que identifican los ejercicios que se pueden ejecutar en cada

uno. Al ejecutar el ambiente se observa todo el entorno ya agrupado, con todos los elementos y los colores llamativos para los niños.



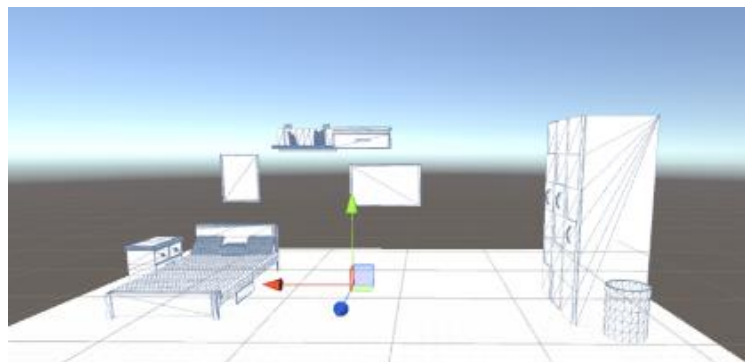
**Figura 38.** Entorno del Menú de juegos.

#### 4.2.3 Diseño de la interfaz del cuarto

La interfaz del cuarto servirá como escenario base para el desarrollo de tres juegos con tareas diferentes, para esto se utilizó varios elementos para su diseño que a continuación se detalla:

Como primer paso en una nueva escena se procedió a crear un plano que ejecutará la función del piso para el cuarto, para esto en la pestaña de Hierarchy en la parte de Create se da clic derecho y se selecciona 3DObject/Plane.

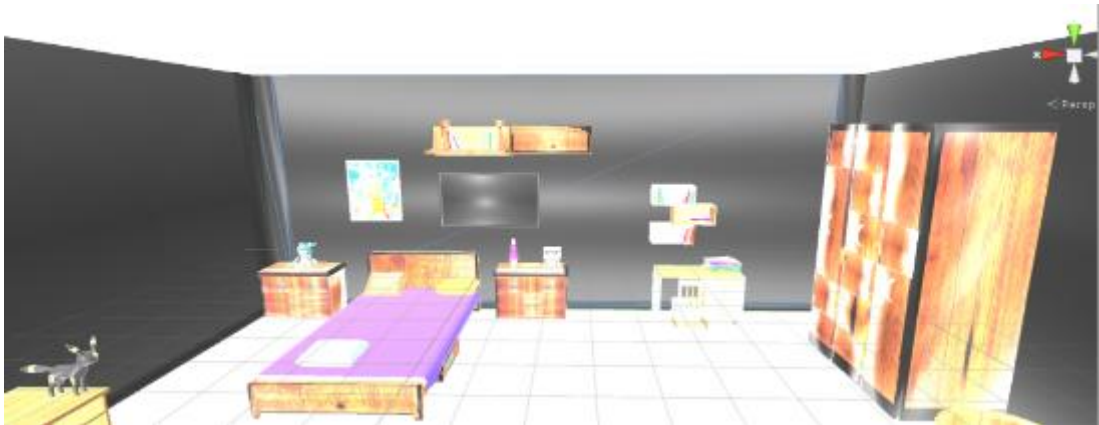
Después se procede añadir los objetos 3D como es: cama, almohada, libros, estanterías, armario, velador, mesa, juguetes, silla, lámpara, televisión, entre otros.



**Figura 39.** Objetos 3D como libros, cama, entre otros en la escena.

Para encerrar al cuarto se colocaron cubos que forman las paredes respectivas, estos se crean al dar clic derecho en Create/3DObject/Cube. También, se agrando el tamaño y se redujo el grosor a través de la utilización de las herramientas propias del Unity.

A cada uno de los objetos que forman el entorno del aula se le añadió diferentes colores para una mejor visualización y diferenciación de cada elemento.



**Figura 40.** Materiales de diferentes colores para los objetos del cuarto.

Para darle un enfoque más realista al ambiente, se agregó a las paredes una textura en forma de dibujos, estrellas, círculos de colores llamativos, del mismo modo al suelo se colocó una textura de madera.



**Figura 41.** Interfaz del cuarto realista.

Para la aplicación de agarre y lanzamiento de pelotas, al cuarto se le agregó una cesta con su respectiva textura, 18 osos y 18 pelotas.



**Figura 42.** Osos y pelotas agregados al entorno.

En el entorno 3D, para la visualización de letras, imágenes como 2D desde el Oculus Rift se utilizó un panel de tipo World Space, para esto se debe dar clic derecho en Create/UI/Panel. Seleccionar el panel y en la pestaña del inspector en la opción de Canvas elegir World Space.

En el panel se agrega textos para visualizar el nombre del usuario, edad, el conteo de los osos y el tiempo. También se crea un canvas 2D para visualización de varios datos por parte del especialista, además, se agregó un botón de retorno.



**Figura 43.** Panel de visualización de datos para el usuario.

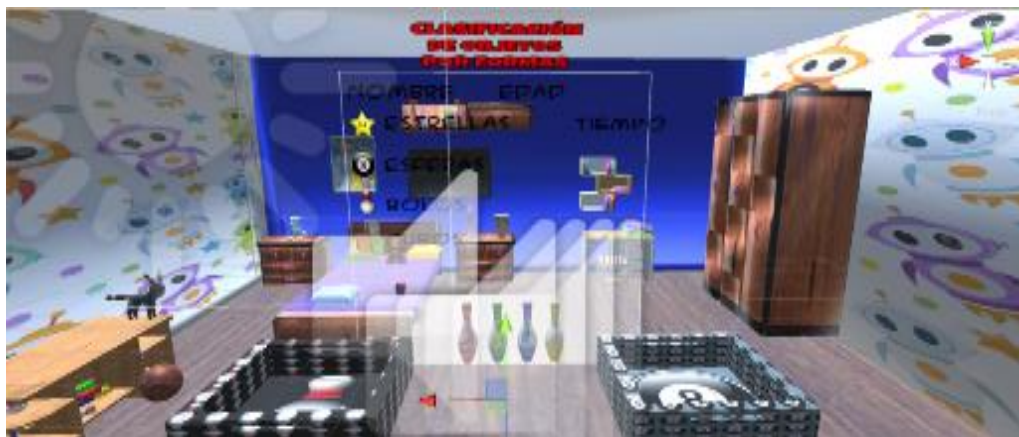
En la aplicación denominada clasificación por colores, se agregó al entorno cuatro cestas de color amarillo, verde, rojo y azul.

También en el ambiente se incorporó objetos como bolos, estrellas, esferas y cubos con sus respectivas texturas diseñadas en Paint y materiales de varios colores. Se agrega sus respectivas imágenes que identifican los colores, título de la aplicación, nombre del usuario, edad y tiempo.



**Figura 44.** Panel de visualización del entorno clasificación por colores.

Para la aplicación de clasificación por objetos se conservó los mismos elementos del entorno clasificación por colores, como adicional se creó texturas con figuras de bolos, estrellas, cubos, esferas y se colocó en cada pieza de las cestas. De igual manera se colocó un panel para la visualización de datos de la aplicación.



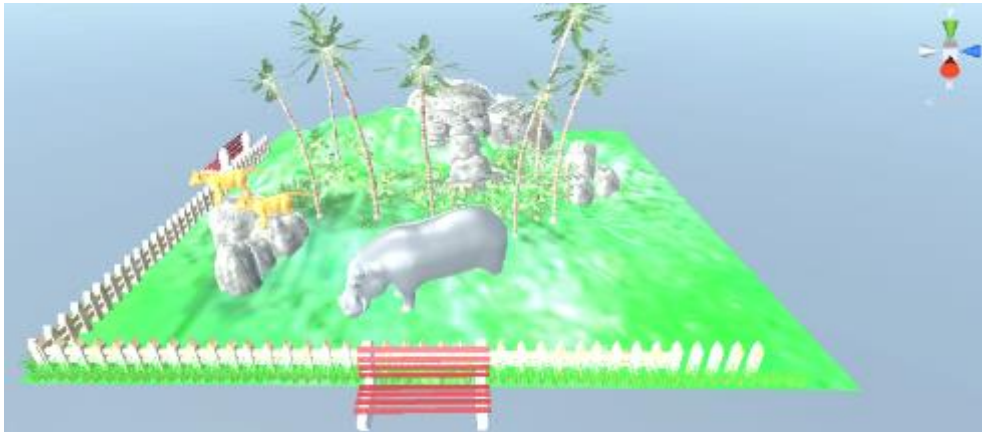
**Figura 45.** Datos de la aplicación: Clasificación por formas.

#### 4.2.4 Diseño de la interfaz del Zoológico

En cuanto a la interfaz del zoológico contiene un diseño diferente, porque es la agrupación de cuatro pequeños ambientes, compuestos de varios objetos de la naturaleza, a continuación se describe cada uno de ellos.

Para el primer ambiente se agrega un terreno, en el cual se puede desnivelar el mismo creando montañas mediante la opción propia del Terreno seleccionando Lower Terrain y se elige cualquier Brush, de igual forma se le agrega el color en la opción Paint Texture. Se colocó en el terreno objetos como palmas, piedras, hierba con su respectivo color y textura adecuada para ser más real.

Para completar este primer ambiente se agregó 3 animales, 2 tigres y 1 elefante, también se colocó una rejilla en dos de sus lados y unas bancas.



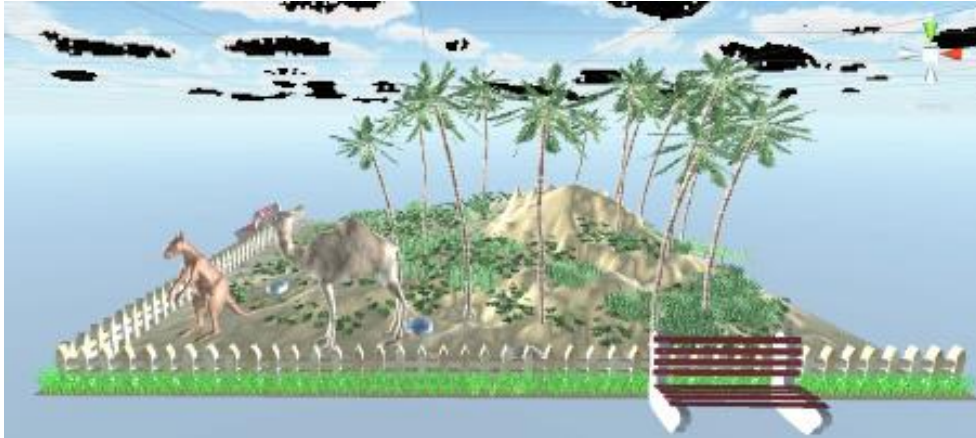
**Figura 46.** Primer ambiente del zoológico.

En el segundo ambiente se añadió un nuevo terreno, en el cual se agregó árboles, piedras, un lago, rejilla, bancas y 2 animales (elefante, cocodrilo).



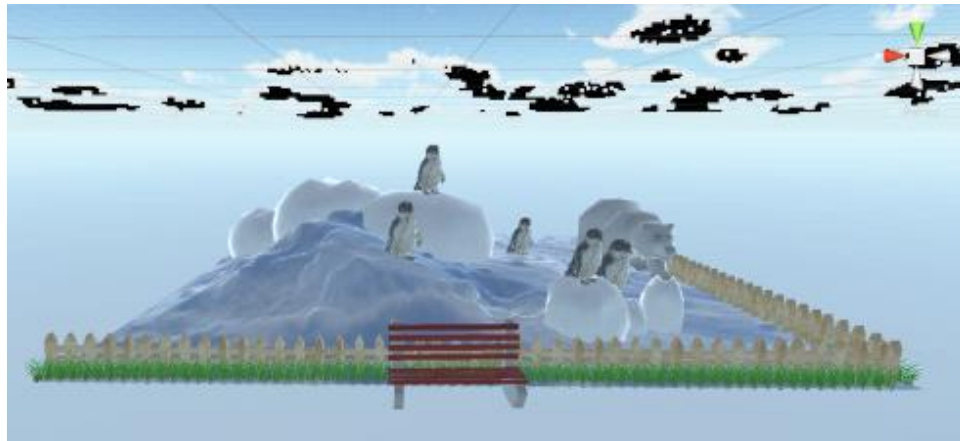
**Figura 47.** Segundo ambiente del zoológico.

Para el tercer ambiente se colocó al terreno una textura de desierto, algunas palmeras y poco césped, de igual forma se puso 2 animales (canguro, camello).



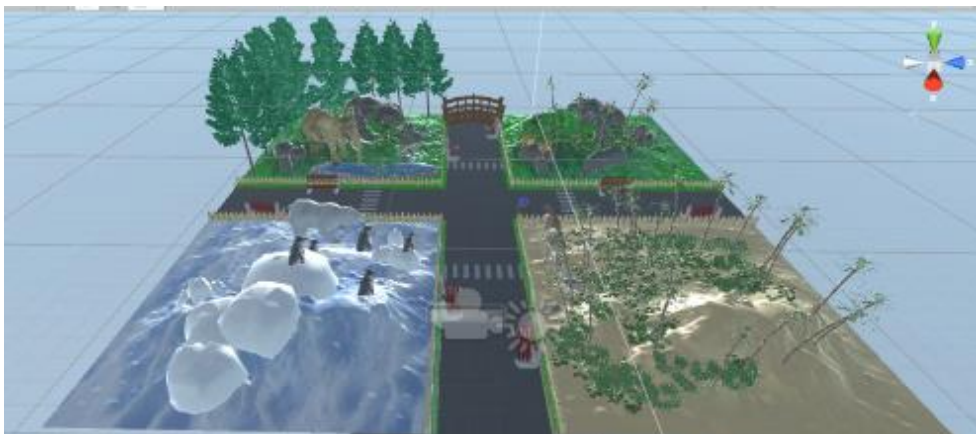
**Figura 48.** Tercer ambiente del zoológico.

En el cuarto ambiente al terreno se le añadió piedras de color blanco para simular el hielo, también se añadió varios pingüinos con su textura real y un oso polar.



**Figura 49.** Cuarto ambiente del zoológico.

La interfaz del zoológico es la agrupación de los cuatro ambientes previamente diseñados, unidos mediante un plano con textura de calle.



**Figura 50.** Unión de los cuatro ambientes del zoológico.

El zoológico es utilizado para dos aplicaciones que son:

- **Explota globos con pinzas:** En esta aplicación se agregan varios prebafs de globos de varios colores, también contiene un panel para visualizar las instrucciones del juego.

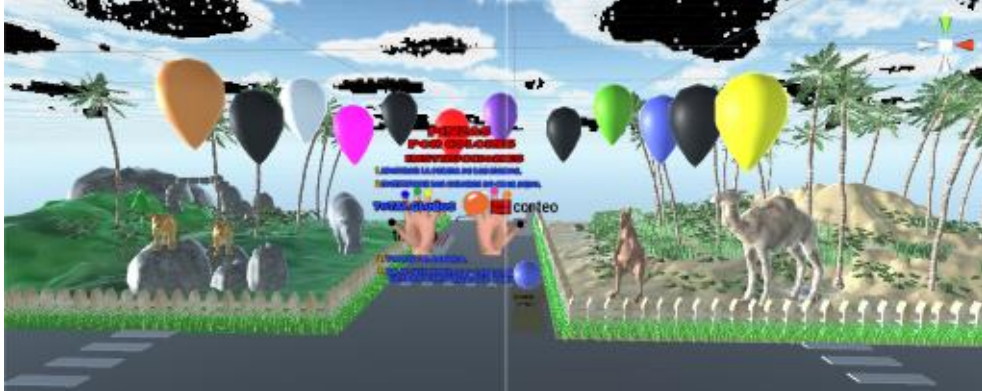


Figura 51. Aplicación de explota globos con pinzas.

- **Atraviesa los aros:** Se añade 4 aros con colores de arco iris, sobre cada uno de ellos se coloca un número y una nave espacial.



Figura 52. Aplicación de atraviesa los aros.

- **Atraviesa los arcos:** Se colocó 4 arcos y una esfera con una textura de luces.



Figura 53. Aplicación de atraviesa los arcos.



#### 4.2.5 Diseño de la interfaz Aula

Esta interfaz fue diseñada en base a una idea que muestre una actividad de la vida diaria de los niños como es asistir y observar un aula de clases, por lo tanto, se incorporó varios objetos como: pizarrón, mesas, sillas, estanterías, plantas, reloj, focos, ventiladores, estos elementos fueron importados a Unity y arrastrados a la escena de desarrollo, por su parte serán los encargados de brindar una imagen realista del entorno.

Por otra parte para realzar el entorno, se procedió a crear cubos para las paredes y a cada objeto se añadió un respectivo color, textura, de esta manera se obtuvo el siguiente entorno.



**Figura 54.** Decoración de la interfaz del aula.

De igual manera, para la aplicación de pintar y escribir libremente, se agregaron varios botones de diferentes colores con sus respectivos textos, también, se colocó una imagen que muestra el título del entorno y otros textos para visualizar el nombre, edad y tiempo de cada usuario.



**Figura 55.** Botones de colores para la aplicación de pintar y escribir libremente.

### 4.3 Desarrollo de scripts para el sistema

En este apartado, se describe cómo se desarrolla cada script mediante el lenguaje C# de Unity, este lenguaje es orientado a objetos utilizando la plataforma.NET. Se crean scripts para dar una función o acción que debe realizar un objeto, sin embargo, hay que destacar que los Game Object en Unity se pueden controlar mediante componentes propios que están adjuntos en cada uno de ellos. Por otro lado, estos componentes no permiten tener acceso o modificar continuamente ciertas características de los objetos, por lo que es imprescindible la utilización de scripts para acceder mediante código de programación a modificaciones de otros parámetros.

Por lo tanto, a continuación se muestra el diagrama de flujo que representa la acción o tarea que el usuario debe realizar en cada juego, del mismo modo, se detallará todos los comandos de mayor importancia que fueron utilizados para cumplir con las tareas especificadas en cada aplicación. También es importante indicar las librerías que se utilizaron en los scripts, estos son:

- using UnityEngine;
- using System.Collections;
- using UnityEngine.UI;
- using System.Text;
- using UnityEngine.VR;
- using UnityEngine.Events;
- using UnityEngine.EventSystems;
- using UnityEngine.Serialization;
- using System.Collections.Generic;
- using Leap.Unity;
- using Leap;
- using System;
- using System.IO;

### 4.3.1 Script para el menú ingreso de datos del usuario

Se crea un nuevo script dando clic derecho sobre la carpeta que se vaya a guardar todos los scripts de desarrollo de la aplicación, después se selecciona Create/C# Script y se le asigna un nombre.

Para comenzar con el desarrollo se debe primeramente crear las variables a ser utilizadas en este caso serán de tipo string y de tipo InputField, las mismas que servirán para grabar los datos ingresados en el campo de control del nombre y edad.

Por consiguiente, para guardar estos valores al presionar el botón GUARDAR, en el script se utiliza la función ApplyButton(), en la cual mediante el uso del comando PlayerPrefs.SetString() se establece el valor de un texto y se procede a guardar este texto en una variable, como se observa a continuación:

```
public void ApplyButton()
{
    name = enter.text;
    age = age1.text;
    PlayerPrefs.SetString("nombre", name);
    PlayerPrefs.SetString("Age", age);
}
```

### 4.3.2 Script para el menú principal de los juegos

En la aplicación se muestra los datos del usuario como es nombre, edad, estos datos fueron guardados en la aplicación anterior y para su utilización se crea un nuevo script y mediante el comando PlayerPrefs.GetString(), se recibe el dato guardado para ser utilizado en esta nueva aplicación, asignada a otra variable de tipo Text, tal y como se muestra a continuación.

```
textonombre.text = "Name: " + PlayerPrefs.GetString("nombre");
edad.text = "Age: " + PlayerPrefs.GetString("Age");
```

Para que al mover las manos dentro del entorno se realice la acción de presionar botones, se debe en la unidad del LeapMotion especificada a través de un prefabricado LMHeadMountedRig colocar el script Pinch Detector, este

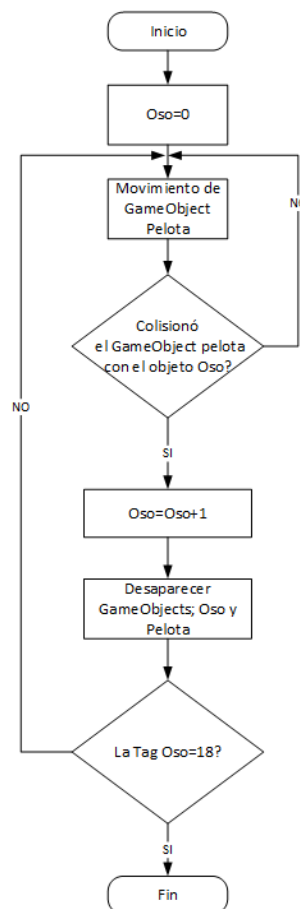
permitirá presionar botones dentro del menú, y se indicara su acción correcta mediante la visualización de un círculo dibujado con el comando DrawCircle.

Entonces, para que al presionar cualquier botón este se dirija a otra escena, se crea un script, en el cual se escribe métodos públicos que mediante el comando Application.LoadLevel() y el número de la aplicación requerida, automáticamente se dirigirán a ese otro entorno.

```
public void Reiniciar()
{
    Application.LoadLevel(2);
}
```

### 4.3.3 Script para la aplicación agarre y lanzamiento de pelotas

Antes de desarrollar el script, hay que tomar en cuenta cual será la tarea que va a realizar, para esto se muestra a continuación un diagrama de flujo que explica cómo se debe ejecutar correctamente la actividad de esta aplicación.



**Figura 56.** Diagrama de flujo: Agarre y lanzamiento de pelotas.

En esta aplicación, para que pueda existir interacción entre las manos del usuario con los diversos objetos es necesario agregar modelos de manos físicas con el BrushHand, estos modelos son especiales ya que son invisibles a la vista del usuario pero permite el movimiento de objetos dentro de un entorno.

Asimismo, para que los objetos puedan tener interacción, estos deben ser cuerpos físicos rígidos, por lo tanto a cada objeto se le agrega el componente Rigidbody para que tenga una masa, gravedad, etc. Por otro lado, se les añade un script denominado InteractionBehaviour para que en la interacción con la mano del usuario, este al cerrar correctamente el objeto quede agarrado y lo pueda trasladar por todo el entorno.

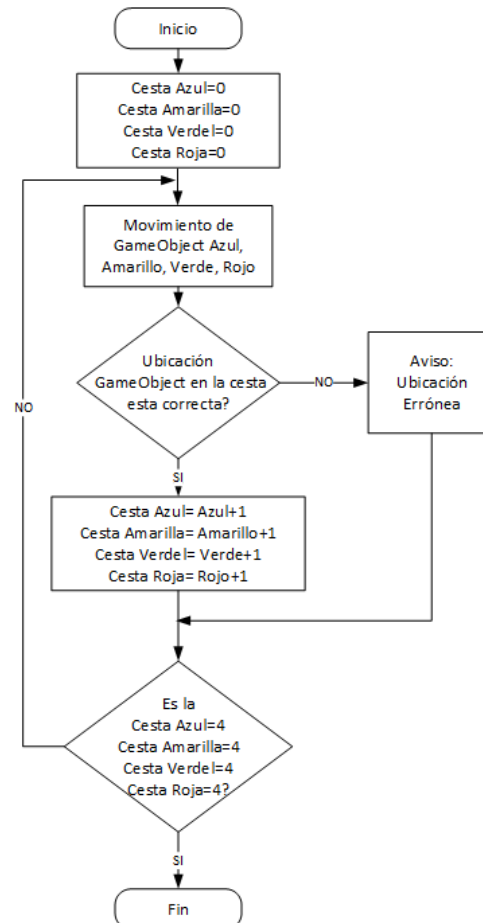
Al iniciar toda programación de los scripts, es indispensable declarar e inicializar todas las variables a ser utilizadas durante el desarrollo, al mismo tiempo que se debe establecer el grado de acceso de las mismas, es decir, cuando se utiliza la opción Public, este dato puede ser utilizado en otro script, pero cuando se requiere de una variable solo para trabajar en un solo script se indica con la palabra Private.

```
public GameObject esfera;
public int contador = 0;
public GameObject oso1, oso2, oso3, oso4, oso5;
public int bonus;
private float hora = 0;
private float minutos = 0;
private float segundos = 0.0f;
private string path1, tiempo1, theDate;
```

Mientras tanto, para detectar si las pelotas del entorno hacen colisión con los osos, a estos objetos se les asigna una TAG, denominada BEAR, la cual servirá para identificar si verdaderamente colisionaron entre ellos, para posteriormente desaparecer tanto al objeto oso como a la pelota del entorno virtual y aumentara las unidades cada vez que el objeto de interacción asignado colisione y cuando este contador sea igual a 18 se ejecutara la condición de finalización de juego y él envió directamente al menú principal.

#### 4.3.4 Script para la aplicación clasificación de objetos por colores.

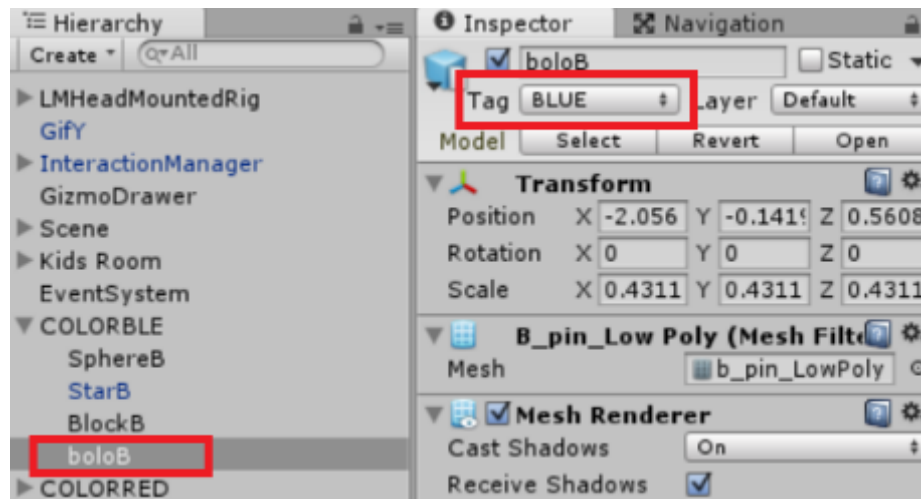
La tarea que debe ejecutar el usuario en esta aplicación esta descrita en el siguiente diagrama de flujo para su mayor comprensión, tal y como se muestra en la Figura 57.



**Figura 57.** Diagrama de flujo: Clasificación de objetos por colores.

Esta aplicación al igual que la anterior su función es de interacción de las manos del usuario con los objetos virtuales, para lo cual se utilizó nuevamente el script InteractionBehaviour en cada uno de los objetos virtuales como son: bolos, estrellas, esferas y cubos.

Para identificar a los objetos por colores se crearon varias Tag denominadas Blue, Yellow, Green y Red, estas fueron añadidas a cada objeto y al cubo base que contiene cada cesta de acuerdo a su respectivo color.



**Figura 58.** Asignación de Tag Blue al objeto.

Se utilizó la función `OnTriggerEnter()` para determinar si el objeto realiza una colisión con la cesta adecuada y poner una bandera en el valor de 1, al mismo tiempo que se activara un sonido indicando la acción correcta.

```
private void OnTriggerEnter(Collider other)
{
    if (other1.gameObject.CompareTag("BLUE"))
    {
        cont = 1;
        clap.Play();
    }
    else
        clap.Stop();
}
```

Además, para indicar la acción errónea de que un objeto sea colocado en una cesta que no corresponde a su color y el usuario lo pueda identificar, se agregó en cada cesta un script con dos funciones que con ayuda de las Tags asignadas a cada elemento se realiza una comparación y de acuerdo a su función se procederá activar un objeto que indicara que dicho elemento está ubicado en una cesta de diferente color, para activar el objeto se utilizará la función `OnTriggerEnter()` y para desactivar el objeto la función `OnTriggerExit()`.

```
private void OnTriggerEnter(Collider other1)
{
    if (other1.gameObject.CompareTag("BLUE"))
        faceB.SetActive(true);

    if (other1.gameObject.CompareTag("YELLOW"))
        faceB.SetActive(true);
}
```

La finalización del juego se dará cuando el contador de cada color sea igual a 4 de los colores Amarillo, Azul, Verde y Rojo, al mismo tiempo que activará varias partículas que indicarán la finalización del mismo y luego lo llevará de regreso al menú principal.

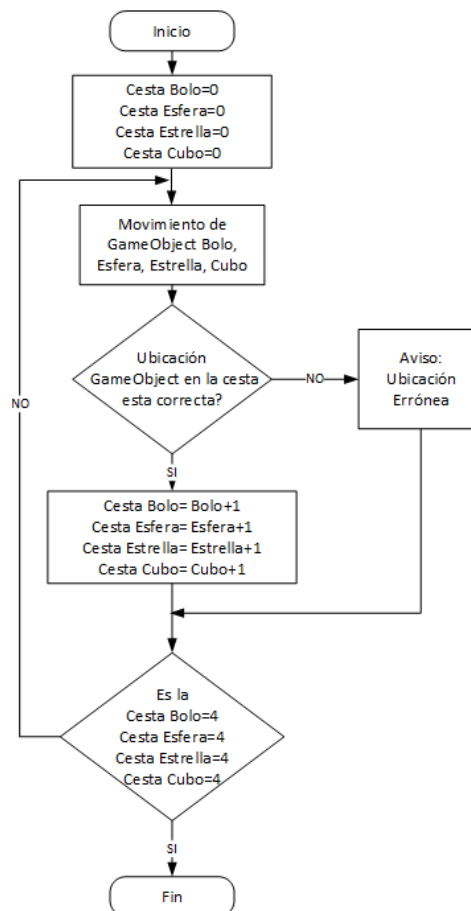
```

If (Total1.cont == 4 && Total11.cont == 4 && Total2.cont == 4 &&
Total3.cont == 4)
{
    Particle1.SetActive(true);
    Particle2.SetActive(true);
    Particle3.SetActive(true);
    if (!claps.isPlaying)
        Application.LoadLevel("1Menu");
}

```

#### 4.3.5 Script para la aplicación clasificación de objetos por formas.

La tercera aplicación tiene como escenario la habitación, en la cual el usuario realizará la acción de colocar cada objeto en la cesta correspondiente a la imagen correcta, a continuación se explica mediante un diagrama de flujo la tarea que se debe ejecutar.



**Figura 59.** Diagrama de flujo: Clasificación de objetos por formas.



El script denominado InteractionBehaviour es el que maneja el movimiento de agarre de objetos y se utilizara en esta aplicación en cada objeto. En esta aplicación también se hace uso de las Tags asignadas a cada objeto y a cada cubo que contiene cada cesta de acuerdo a la forma que posee las imágenes en la base de cada una.

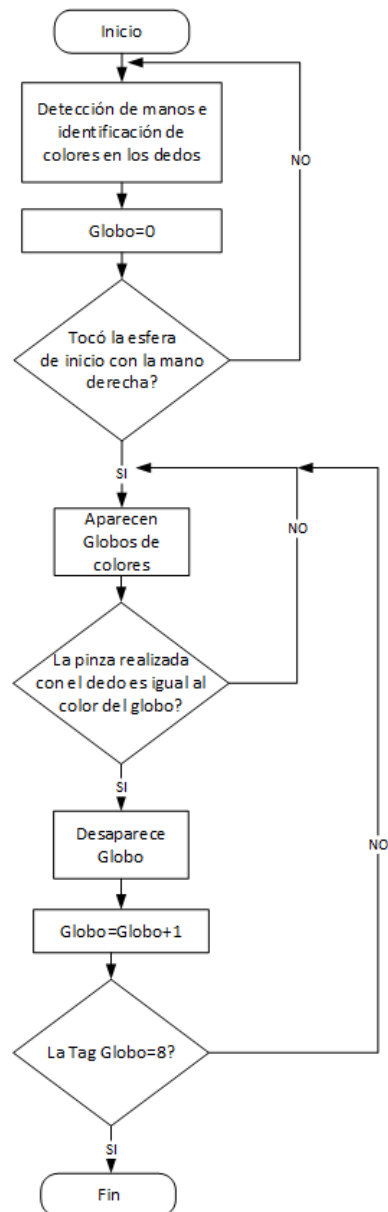
Del mismo modo para identificar que exista una colisión entre los objetos y la cesta correcta, se utilizó la función OnTriggerEnte(), la cual pondrá en 1 el valor de una variable si la acción es correcta, este valor será utilizado en otro script, para realizar la suma de todos los objetos y a su vez este valor pueda ser visualizado por el usuario y el especialista cuando se está ejecutando la aplicación.

```
void Update()
{
    countTotal = count1.count + count2.count + count3.count +
count4.count;
    CTBolo.text = "BOWLLING=" + countTotal.ToString();
    CTB.text = "BOWLLING=" + countTotal.ToString();
}
```

La aplicación finalizara cuando el contador de las cuatro cestas sean igual a cuatro, aparecerán partículas que indicaran la finalización del juego y en seguida se dirigirá a la aplicación de menú principal para la selección de otra aplicación.

#### **4.3.6 Script para la aplicación explota globos con pinzas.**

Para la siguiente aplicación, se utilizará la concentración total del usuario, puesto que debe identificar correctamente los colores que se encuentran en cada dedo y realizar de forma correcta la tarea especificada en el siguiente diagrama de flujo.



**Figura 60.** Diagrama de flujo: Revienta globos con pinzas.

Se iniciaría esta aplicación cuando el usuario toque la esfera que se encuentra en el entorno con la mano derecha, para esto se compara si existe una colisión entre la esfera y la mano derecha mediante el dedo índice. Si la acción es correcta se procederá a activar los globos.

```

if (other.gameObject.CompareTag("Habilitar"))
{
    iniciar = 1;
    IndiceeR.SetActive(false);
}

```

Se creó varios círculos colocados en cada dedo, a los cuales se les agregó diferentes colores, también se creó varias Tags para los dedos: Pulgar, Índice, Medio, Anular y Meñique de la mano izquierda y derecha. Cabe indicar

que la acción de pinzas está realizada mediante una colisión de los dedos con el dedo pulgar de cada mano, para esto se compara si la colisión de cada dedo y si es correcta se le asignará un valor.

```

    if (other.gameObject.CompareTag("Habilitar"))
    {
        fingerI = 50;
    }
    if (other.gameObject.CompareTag("Medio"))
    {
        fingerM = 51;
    }
    if (other.gameObject.CompareTag("Anular"))
    {
        fingerA = 52;
    }
    if (other.gameObject.CompareTag("Menique"))
    {
        fingerN = 53;
    }

```

Cuando se haya realizado todas las pinzas con todos los dedos, se comparará si el total de globos reventados es igual a 8, este indicara mediante la aparición de partículas que la tarea del juego se ha concluido correctamente.

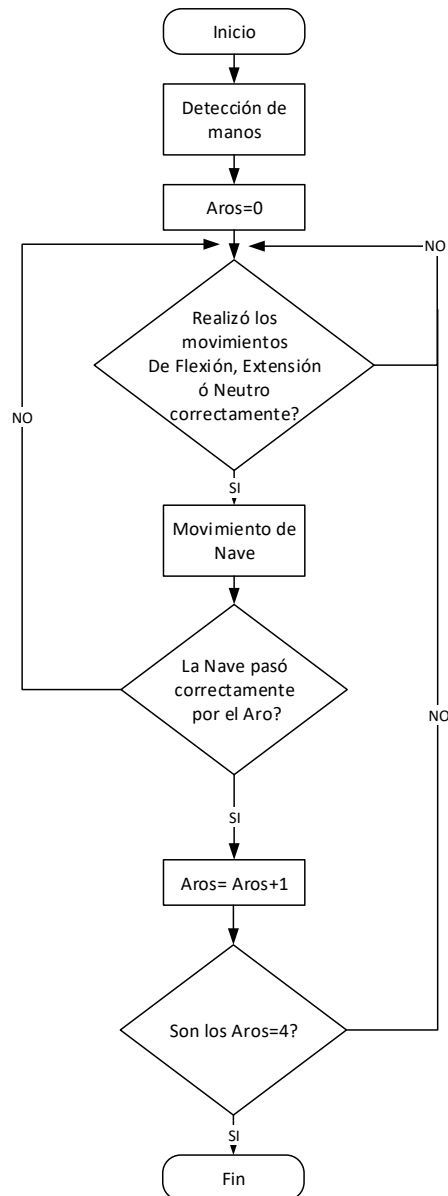
```

    if (total == 8)
    {
        Particle1.SetActive(true);
        Particle2.SetActive(true);
        Particle3.SetActive(true);
        Particle4.SetActive(true);
        if (!clap.isPlaying)
        {
            Application.LoadLevel("1Menu");
        }
    }

```

#### 4.3.7 Script para la aplicación atraviesa los aros.

La ejecución de la tarea se especifica en el siguiente diagrama de flujo, el mismo que indica que se debe realizar para que la nave pueda moverse por todo el entorno y llegue a cada lugar deseado.

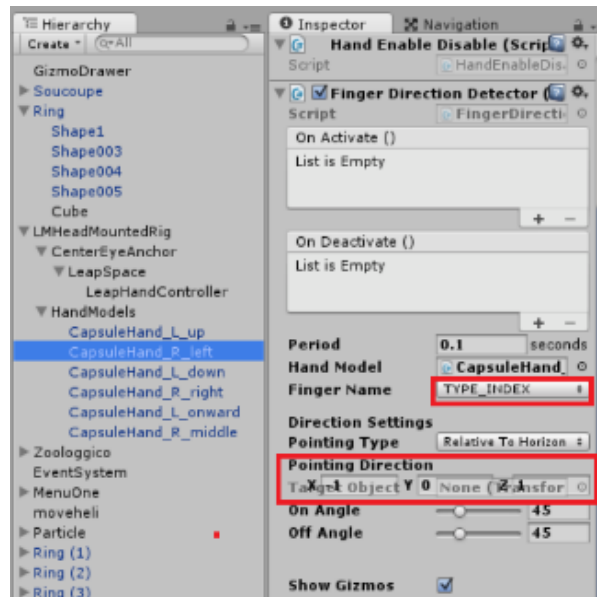


**Figura 61.** Diagrama de flujo: Atraviesa los aros.

Como ya se ha venido realizando anteriormente, en esta aplicación también se hace uso de Tags, las cuales fueron asignadas a cubos que se encuentran invisibles en cada aro para identificar que la nave ha pasado correctamente por estos, a las Tags se ha denominado Pos1, Pos2, Pos3 y Pos4, ya que existen cuatro aros por todo el entorno.

Para que con cada mano realice los movimientos de flexión y extensión, se les agrega varios scripts previamente diseñados por los módulos del LeapMotion, como es Extended Finger Detector(Detecta si los dedos o cualquier dedo se encuentra completamente extendido), Palm Direction

Detector (Se agrega un vector de 3D para establecer mediante valores booleanos en qué dirección se requiere la palma), Finger Direction Detector (Se selecciona el dedo que se requiere detectar y la posición del mismo en el plano utilizando un vector 3D), para que una misma mano pueda realizar varias acciones se utiliza el script Detector Logic Gate que mediante la activación de una compuerta AND determina la acción siempre y cuando la posición deseada cumpla con todas las especificaciones establecidas.



**Figura 62.** Detección de posición de la mano y dedo índice.

En cada mano según la detección se posición se elegirá una función determinada, la misma que pondrá en 1 varias banderas para ejecutar los movimientos de la nave.

```
//subir
public void active_up()
{
    Flat_up = 1;
}
public void desactive_up()
{
    Flat_up = 0;
}
```

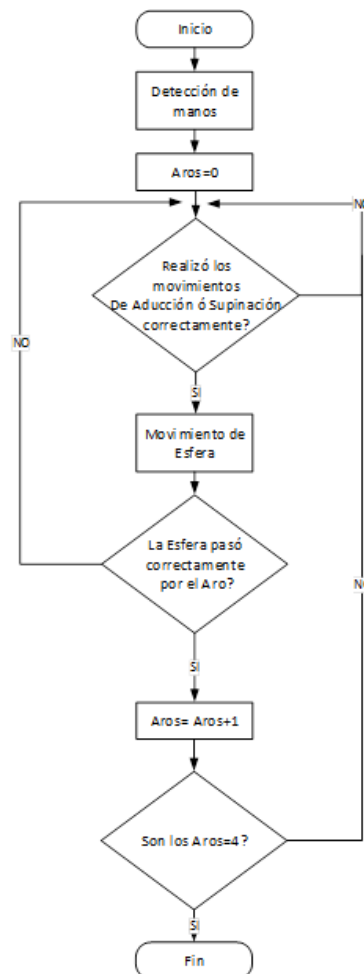
Con las banderas que estarán constantemente activándose y desactivándose según la posición correcta de las manos, se procederá a comparar si el parámetro es correcto y a ejecutar la acción de movimiento en el eje correspondiente.

Se finalizará cuando la suma de la variable TotalRing sea igual a cuatro, esto indicará que el usuario ha realizado correctamente los movimientos de sus manos.

```
TotalRing = variable1 + variable2 + variable3 + variable4;
Total.text = "Total de Aros= " + TotalRing.ToString();
```

#### 4.3.8 Script para la aplicación atraviesa los arcos.

En esta aplicación, la esfera se moverá por todo el camino establecido en el entorno y deberá cumplir con la tarea especificada en el siguiente diagrama de flujo para concluir el juego.



**Figura 63.** Diagrama de flujo: Atraviesa los arcos.

Para el seguimiento de la cámara según el movimiento del GameObject denominado esfera, se colocó un script de control de cámara en el LMHeadMountedRig, para este seguimiento se utilizó la instrucción Transform que da la posición en X, Y, Z del objeto.

Para cada movimiento se agrega el script Move, el cual contiene las funciones y este permitirá activar o desactivar la bandera especificada en cada función en relación a la posición de las manos que el usuario realice durante la ejecución del juego.

De acuerdo a las banderas activadas, la esfera se podrá mover hacia adelante, atrás, izquierda y derecha de todo el entorno. Para lo cual se va cambiando las posiciones en X, Y, Z y mediante un valor de velocidad especificado su movimiento será adecuado y no muy rápido.

#### 4.3.9 Script para la aplicación Pinta y escribe libremente.

Esta última aplicación contiene varias tareas que el usuario debe realizar según como lo escoja, para comprender de mejor manera la tarea se muestra a continuación el diagrama de flujo.

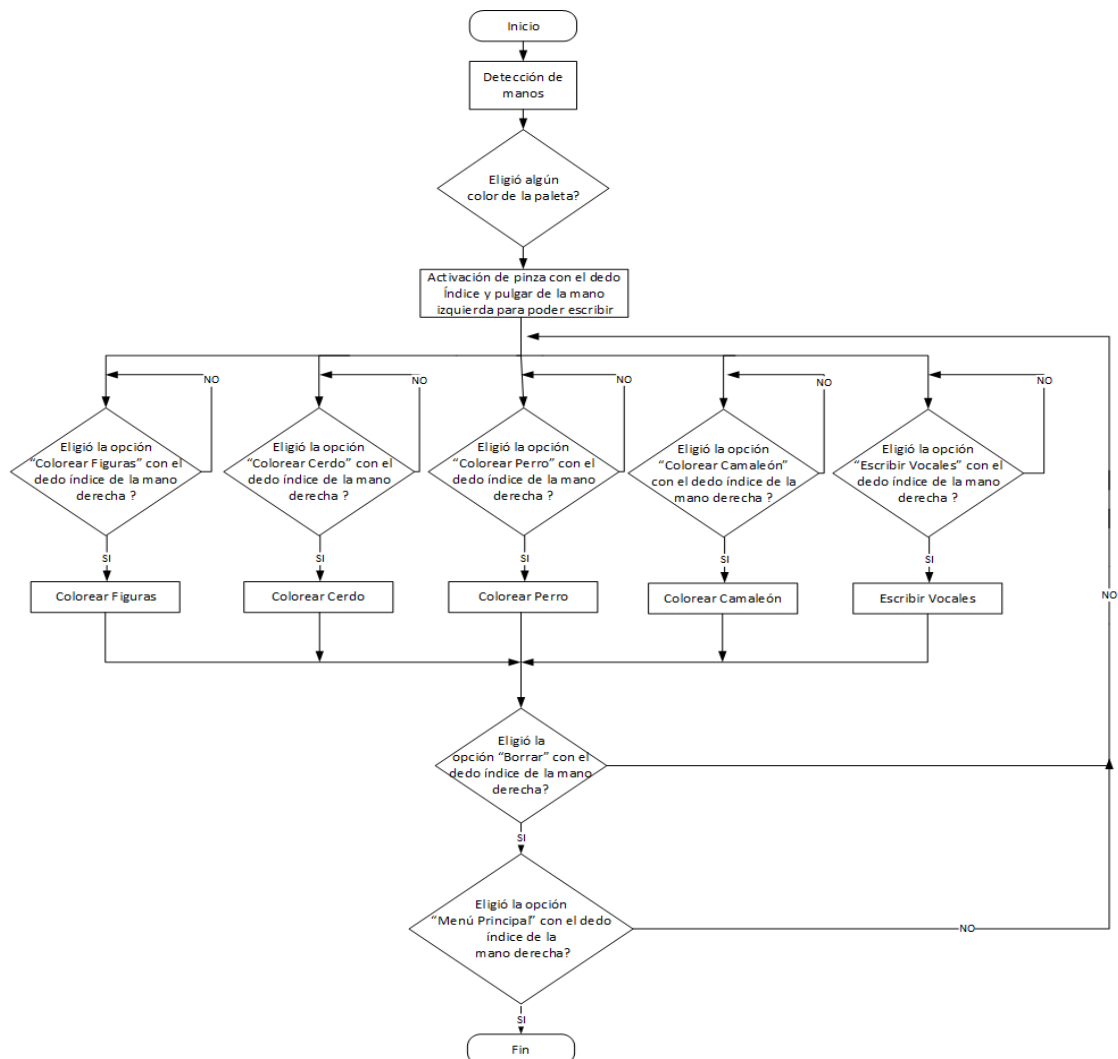


Figura 64. Diagrama de flujo: Pintar y colorear libremente.

Para activar la escritura con varios colores al seleccionarlos, se creó Tags con los nombres de los colores y se asignó a cada cubo. Estas Tags servirán para activar y desactivar el script Pinch Drawing de cada color al cual se les coloco como GameObject para su mejor manejo dentro de los scripts.

Como esta aplicación contiene varios botones en los cuales si se selecciona con el dedo índice aparecerán varias imágenes de animales o letras, estos elementos serán activados y desactivados mediante comparación de Tags.

Para borrar todo lo que el usuario haya realizado dentro del entorno, deberá seleccionar el botón borrar el mismo que mandara a resetear la escena y podrá seguir eligiendo más opciones. Cuando quiera finalizar la aplicación simplemente debe seleccionar el botón menú y enseguida se transportara al menú de inicio de juegos principal.

#### **4.4 Descripción general del sistema**

Esta sección se centra en las etapas que tiene el sistema de realidad virtual para rehabilitación motora fina, se observa dos niveles de dificultad y precisión para poder desarrollar habilidades motrices, como es: la Coordinación Viso-Manual y la Coordinación Grafo preceptiva. Estos diferentes niveles se centran en que el usuario logre realizar una actividad para así poder utilizar de forma simultánea e integrada las manos y la vista.

Con el esquema del sistema, como se muestra en la Figura 65 se explicará las diferentes aplicaciones elaboradas con el propósito de mejorar las habilidades motrices del usuario obteniendo así una coordinación visomotora.





**Figura 65.** Esquema general del sistema de Realidad Virtual

Este sistema de realidad virtual ayuda a la rehabilitación motora fina en niños a través de juegos interactivos motrices obteniendo así beneficios en la parte física y psicológica, ya que desempeña una gran influencia en el desarrollo integral del niño. El sistema se encuentra orientado a niños con patologías como: síndrome de Down, retraso mental y espasticidad motora fina, todas ellas mostrando en parte alguna debilidad motriz. El esquema mostrado anteriormente muestra las diferentes actividades convenientes para que el niño mejore en sus habilidades motrices finas.

#### 4.4.1 Actividades de Coordinación Viso-Manual

En el sistema se desarrolla un conjunto de juegos para desarrollar la parte de coordinación viso manual del niño, se debe tener claro que la coordinación viso manual se realiza con las manos y dedos siempre y cuando exista una coordinación con los ojos, es decir la vista. Las aplicaciones, es decir los juegos constan de diferentes actividades y se clasifican en:

- Aplicación de Agarre
  - Agarre y Lanzamiento
  - Clasificación de objetos por colores
  - Clasificación de objetos por su forma

- Aplicación de pinzas
  - Secuencia de pinzas con cada dedo de la mano
- Aplicación de movimientos de muñeca
  - Flexión y extensión de la muñeca
  - Supinación y pronación de la muñeca
  - Aducción de la muñeca

### Aplicación de Agarre

Dentro de este punto las aplicaciones de agarre tienen como objetivo principal desarrollar el agarre y lanzamiento de determinados objetos virtuales en el menor tiempo posible, logrando así una concentración y coordinación de los movimientos de la manos y la vista cuyo objeto es coordinar las funciones neurológicas, esqueléticas y musculares para producir movimientos precisos a la hora de lanzar el objeto y clasificar el objeto por colores o por formas. Este juego es apropiado para el desarrollo de la habilidad motriz fina gracias a los movimientos de agarrar, sostener, soltar y manipular objetos.

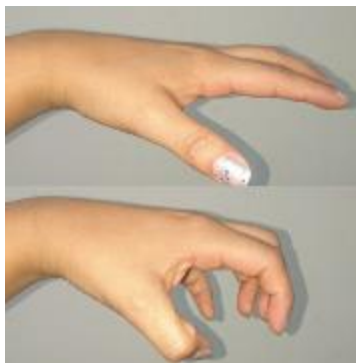
En el juego se consigue ejecutar dos tipos de agarre:

- Agarre de pinza con todos los dedos de la mano: se caracteriza por utilizar la punta del dedo pulgar en oposición a los cuatro dedos en forma de pinza.



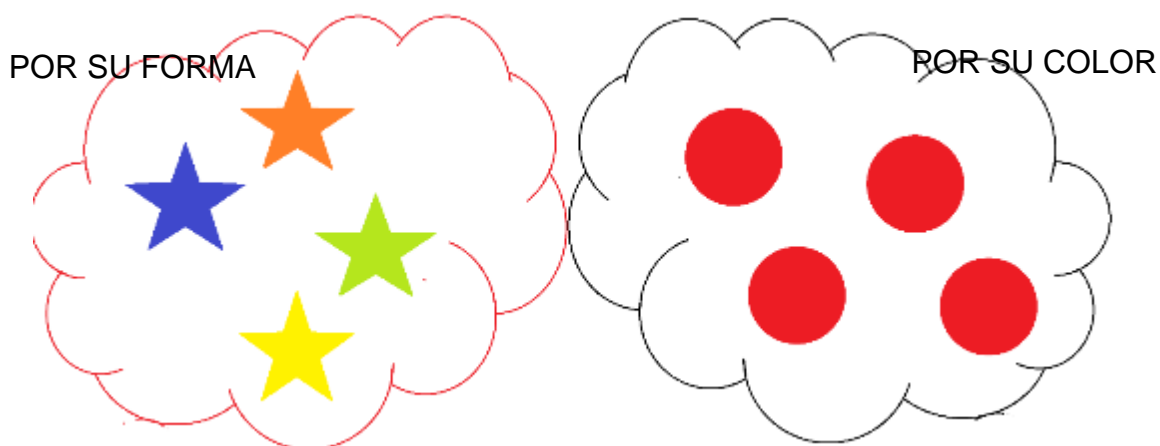
**Figura 66.** Movimiento para el Agarre de pinza

- Agarre de gancho: se caracteriza por utilizar el dedo pulgar en oposición a los cuatro dedos.



**Figura 67.** Movimiento para el Agarre de gancho

También, se consigue un desarrollo cognitivo con la clasificación de objetos por colores y la clasificación de objetos por la forma al tener colores llamativos y entornos intuitivos, en esta actividad necesita de una gran concentración al momento de buscar el objeto, agarrar, desplazar y soltar en el lugar que le corresponde todo esto conlleva a que el niño este concentrado, atento y vaya memorizando los colores y formas a clasificarse.



**Figura 68.** Clasificación de objetos por la forma y color

### **Aplicación de pinzas**

Esta aplicación tiene como objetivo realizar el ejercicio de pinzas en cada dedo, es decir juntar y separar los dedos logrando así estimular, desarrollar y adquirir destrezas y habilidades en los movimientos de los dedos con movimientos precisos y coordinados para así fortalecer el dominio de la pinza del dedo pulgar con los otros cuatro dedos de la mano, como se muestra en la Figura 69.



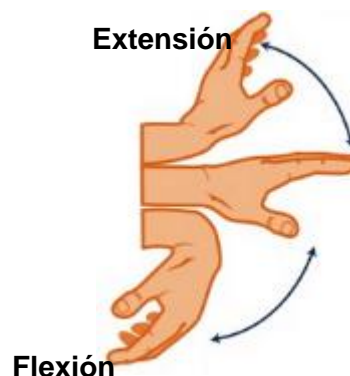
**Figura 69.** Pinza dedo pulgar con los cuatro dedos de la mano

Este ejercicio conlleva principalmente a la estimulación de la coordinación viso-manual perfeccionando movimientos finos de los dedos para llegar así al objetivo final que es la precisión en la escritura.

#### **Aplicación de movimientos de muñeca**

Para esta aplicación el objetivo principal es fortalecer la muñeca y así incrementar la fuerza muscular de la mano. También trata de que el usuario domine sus movimientos para así lograr el control de diversas coordinaciones motoras. Se basa principalmente en movimientos de muñeca como son:

- Extensión: se caracteriza por inclinar la muñeca hacia arriba apuntando los dedos aproximadamente hacia arriba.
- Flexión: se caracteriza por inclinar la muñeca hacia abajo apuntando los dedos aproximadamente hacia abajo.



**Figura 70.** Movimiento de Flexión, neutral y extensión de la muñeca

- Aducción: se caracteriza por desviar los dedos hacia el exterior.

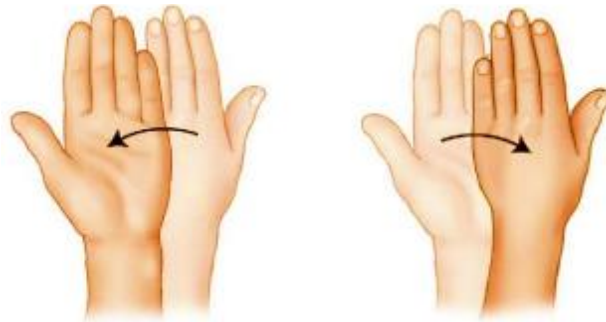


**Figura 71.** Movimiento de aducción de la muñeca

- Supinación: se caracteriza por rotar la muñeca hacia el exterior, dicho de otra manera mostrando la palma de la mano hacia arriba.
- Pronación: se caracteriza por rotar la muñeca hacia el interior, dicho de otra manera mostrar la palma de la mano hacia abajo.

**Supinación**

**Pronación**



**Figura 72.** Movimiento de pronación y supinación de la muñeca

#### 4.4.2 Actividades de Coordinación Grafo Perceptiva

Para esta aplicación el objetivo fundamental es completar y potenciar el desarrollo motriz del niño, así como desarrollar su área creativa e imaginativa al realizar formas y trazos para así lograr activar el lóbulo frontal. Por otra parte, también permite expresar sus emociones y sentimientos a través de los colores para así alcanzar un conocimiento y diferenciación de los diversos colores propuestos, por ende el niño desarrolla una organización de espacio y tiempo.



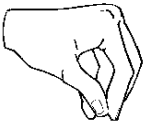
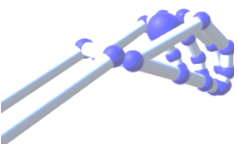

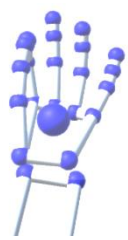
**Figura 73.** Movimiento grafoperceptivo.

Hay que tener en cuenta que la aplicación es prácticamente libre, para que de a poco vaya perfeccionando los movimientos de la mano, muñeca, antebrazo y especialmente de los dedos. Se realiza movimientos horizontales-verticales, oblicuos, bucles hasta llegar a la escritura de letras.

#### 4.5 Movimientos de rehabilitación en el sistema

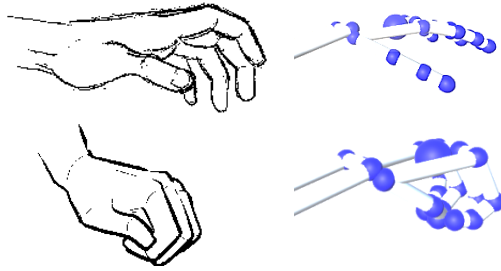
Con ayuda de una terapeuta profesional se selecciona los movimientos de rehabilitación para este sistema y se realiza una comparación entre los movimientos reales y los movimientos que detecta el dispositivo LeapMotion. Se observa en la Tabla 2 la comparación de movimientos.

**Tabla 2.**  
**Comparación de Movimientos**

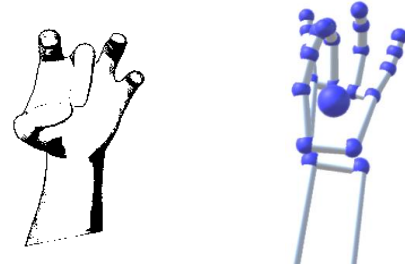
CUADRO COMPARATIVO DE MOVIMIENTOS			
TERAPEUTICOS	LEAPMOTION	TERAPEUTICOS	LEAPMOTION
Agarre de pinza		Pinza dedo pulgar con dedo índice	
			

CONTINÚA 

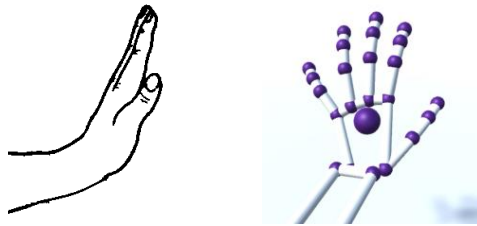
**Agarre de gancho**



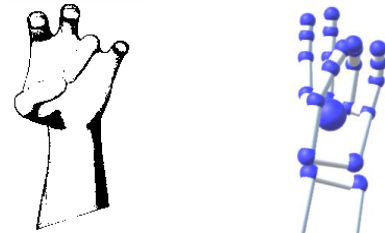
**Pinza dedo pulgar con dedo medio**



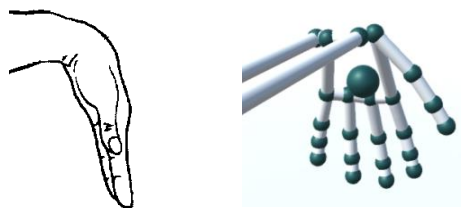
**Extensión de muñeca**



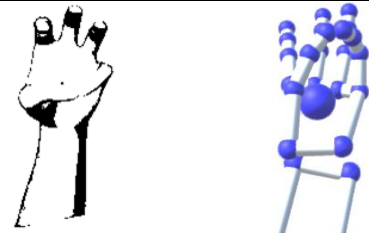
**Pinza dedo pulgar con dedo anular**



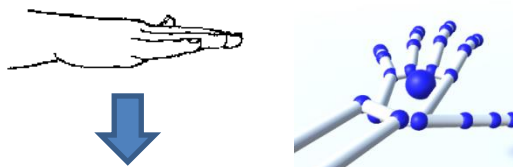
**Flexión de muñeca**



**Pinza dedo pulgar con dedo meñique**



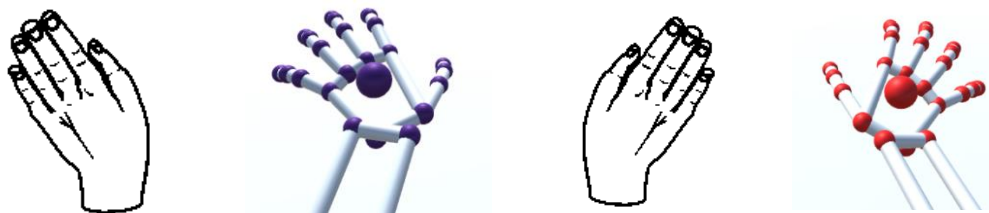
**Pronación**



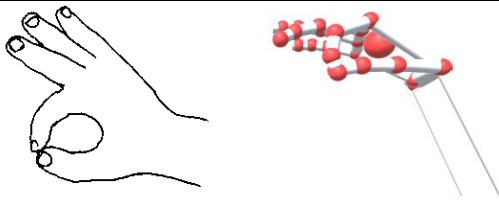
**Supinación**



**Abducción de las manos**



**Pinza para pintar o escribir**



Los movimientos de rehabilitación motora fina coinciden con los movimientos establecidos por el dispositivo LeapMotion, con anticipación es necesario realizar una programación específica de cada movimiento, para que dentro del entorno de Unity el dispositivo LeapMotion reconozca los movimientos establecidos que sirven para la rehabilitación.

Cada movimiento requiere de diferentes parámetros los cuales ayudan a la combinación del dispositivo y los ejercicios de rehabilitación, con el software que se utiliza para el sistema de realidad virtual para la rehabilitación motora fina. También, se considera necesario guardar los datos del usuario para almacenar los tiempos de finalización de los juegos virtuales para observar la mejoría en cada movimiento.

#### 4.6 Descripción de los juegos

En esta sección se detalla la descripción de cada juego y los movimientos físicos, los cuales llevarán al objetivo de cada juego para lograr una rehabilitación motora fina en niños. Este sistema cuenta con un menú interactivo principal, como se muestra en la Figura 74, el cual permite que el usuario seleccione el botón con el dedo índice de la mano derecha una de las seis aplicaciones. Una vez seleccionada la aplicación, inmediatamente el sistema procederá a llevar virtualmente al usuario a la aplicación deseada.



Figura 74. Menú Principal



El entorno virtual desde la primera a la tercera aplicación está basado en una habitación con colores y objetos llamativos para el usuario. La primera aplicación consta de dieciocho osos los cuales se encuentran alrededor de una cesta que contiene dieciocho pelotas. El usuario debe agarrar la pelota ya sea con la mano derecha o la mano izquierda y lanzarla hacia uno de los osos, una vez la pelota haya colisionado con el oso los dos objetos desaparecerán. Finalmente, cuando el usuario haya acertado los dieciocho osos se reproducirá un audio final y unas luces de colores como muestra de cumplir del objetivo del juego, los mismos que se repetirán al finalizar con éxito cualquier juego.



**Figura 75.** Aplicación 1: Agarre y Lanzamiento

La segunda aplicación está basada en la clasificación de objetos por colores, la habitación cuenta con 4 cestas de color azul, amarillo, verde y roja, y 4 objetos de cada color mencionado anteriormente. Una vez el usuario haya identificado los colores y los objetos situados en diferentes partes de la habitación, procederá a coger cada objeto y ubicarlo en la cesta correspondiente al color mostrando así un sonido de acierto, en caso de que el objeto se haya ubicado en la cesta errónea se indicará una cara triste y un sonido como muestra del error. Este error no desaparecerá hasta que el usuario nuevamente corrija la ubicación del objeto erróneo y lo desplace hacia la cesta correcta.



**Figura 76.** Aplicación 2: Clasificación de objetos por colores

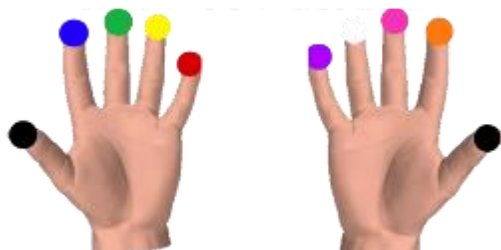
La tercera aplicación consiste en la clasificación de objetos por su forma, la habitación cuenta de cuatro cestas en las cuales cada cesta contiene la forma del objeto a clasificar, en este caso se tiene la cesta de bolos, la cesta de estrellas, la cesta de cubos y la cesta de esferas. Existe cuatro objetos de: bolos, estrellas, cubos y esferas los cuales cuando el usuario haya agarrado el objeto debe colocar dichos objetos en la cesta correspondiente. Esta aplicación es similar a la explicada anteriormente, por lo tanto también contiene un audio y una cara triste mostrando al usuario que el objeto está en una ubicación errónea.



**Figura 77.** Aplicación 3: Clasificación de objetos por formas

Desde la cuarta hasta la sexta aplicación el entorno está basado en un zoológico con animales que al usuario llama la atención debido a que no son muy usuales de visualizar en la vida real. La cuarta aplicación cuenta con ocho globos de colores y consiste en que el usuario al visualizar sus manos observará en las yemas de sus dedos círculos de colores, tal y como se

muestra en la Figura 78, similares a los globos, así el usuario debe realizar el movimiento de pinza de acuerdo al color del globo que aparezca.



**Figura 78.** Colores en las yemas de los dedos

La pinza se deberá realizar el dedo pulgar de color negro con los cuatro dedos faltantes. Una vez que el usuario haya realizado la pinza correctamente, el globo desaparecerá y se mostrará en el entorno un audio y un objeto animado para que el usuario sepa que realizó correctamente la pinza.



**Figura 79.** Aplicación 4: Explota globos con pinza

Para la quinta aplicación, se tiene cuatro aros en el aire alrededor del entorno virtual y una nave la cual será guiada por el usuario según el movimiento que realice ya sea con la mano derecha o la mano izquierda, Figura 80 , como se muestra en la Tabla 3.



**Figura 80.** Posición de las manos

**Tabla 3.**  
**Movimiento de la nave**

Mano	Movimiento Real	Movimiento de la Nave
Izquierda (Palma hacia abajo)	Extensión	Arriba
	Flexión	Abajo
	Pronación	Adelante
Derecha (Palma hacia mano izquierda)	Flexión	Derecha
	Extensión	Izquierda
	Pronación	Atrás

El objetivo del juego consiste en que el usuario mediante los movimientos que realice con la mano derecha o izquierda mueva a la nave y está pase por el aro obteniendo así un resultado final de cuatro aros. Cada que la nave pase por el aro se mostrará un audio como muestra que el movimiento hacia el aro ha sido correcto.



**Figura 81.** Aplicación 5: Atraviesa los aros

La sexta aplicación tiene el mismo objetivo que el juego anterior, pero con diferentes movimientos. Consiste en pasar la esfera por los cuatro aros los cuales se encuentran en el suelo del entorno virtual.



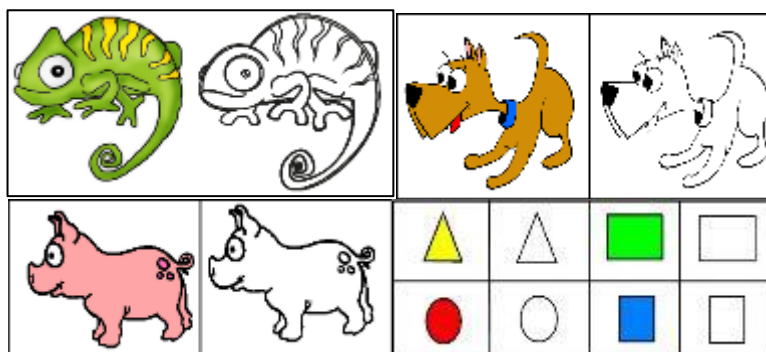
**Figura 82.** Aplicación 6: Atraviesa los arcos

Al pasar la esfera correctamente se mostrará un audio, los movimientos para esta aplicación se muestran en la Tabla 4.

**Tabla 4.**  
**Movimiento de la esfera**

Mano	Movimiento Real	Movimiento de la esfera
Izquierda	Aducción	Izquierda
	Supinación	Atrás
Derecha	Aducción	Derecha
	Supinación	Delante

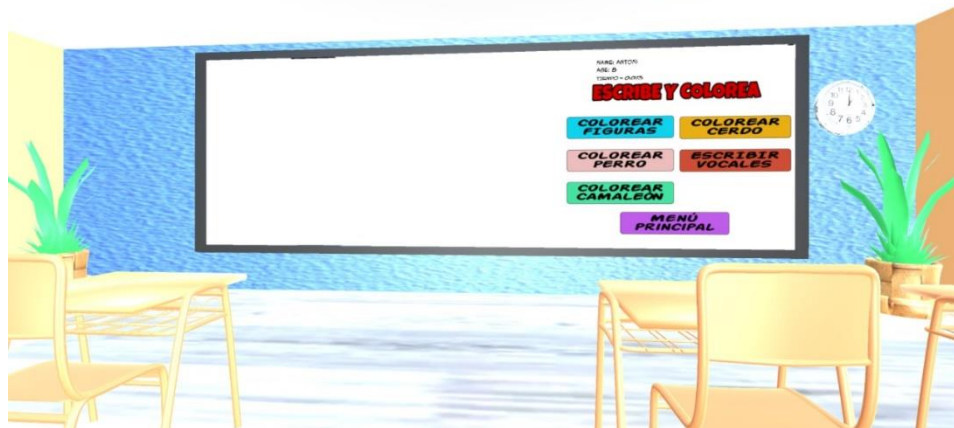
Finalmente, en la séptima aplicación se tiene un entorno en un aula la cual contiene: pizarrón, sillas, mesas, entre otros. Se observa en la parte derecha del pizarrón cinco botones: Colorear Camaleón, Colorear Perro, Colorear Cerdo, Colorear Figuras y Escribir vocales, los cuales al seleccionar con el dedo índice de la mano derecha se mostrará en el pizarrón muestras de imágenes a colorear, tal y como se observa en el Figura 83.





**Figura 83.** Aplicación 7: Muestras de imágenes

En esta aplicación se trata de incentivar al usuario a que dibuje, coloreé y escriba de forma libre. Para lo cual, al observar la palma de la mano izquierda se tiene una paleta de colores y en ella existe la opción de borrar lo que se realizó anteriormente.



**Figura 84.** Aplicación 7: Aprende a colorear

Sin embargo, si el usuario no ha seleccionado el color deseado no podrá colorear. Una vez seleccionado el color se procede a realizar la pinza (dedo pulgar con dedo índice) con la mano derecha y se podrá empezar a dibujar libremente.

## CAPÍTULO V

### 5. ANÁLISIS DE RESULTADOS

Para evaluar los resultados del sistema se acudió a la fundación de niños Especiales “San Miguel” de la ciudad de Salcedo, mediante la colaboración de las autoridades, especialistas y usuarios que acuden diariamente a esta institución se logró realizar y culminar con las pruebas que en este capítulo se detalla.

#### 5.1 Consideraciones generales de los dispositivos de tracking óptico

Los dispositivos de tracking óptico utilizados en las pruebas de funcionamiento del sistema virtual, poseen características técnicas que permitirán complementar conjuntamente los dos dispositivos en un sólo sistema virtual, como se muestra en la Tabla 5 y 6.

**Tabla 5.**  
**Características técnicas del LeapMotion**

Parámetros	Características
Dimensiones	75x25x11 (mm)
Cámara	CMOS Monocromático L:850nm
Velocidad de captura	200 fps
Zona de cobertura	150°
Sensor	Infrarrojo L:850nm
Distancia de trabajo	7-25 cm

**Tabla 6.**  
**Características técnicas del Oculus Rift DK2**

Parámetros	Características
Zona de cobertura	100°
Resolución	1920x1080 pixeles (960x1080 por ojo)
Peso	440g

Cabe destacar que el dispositivo LeapMotion, es un dispositivo de control gestual el cual permite detectar los movimientos de las manos en tiempo real, mientras el dispositivo Oculus Rift permitirá la inmersión del

usuario a un entorno virtual para realizar las actividades respectivas de rehabilitación motora fina.

## 5.2 Validación de los dispositivos mediante el test Box and Block

Para validar que los dispositivos de tracking óptico utilizados en las diferentes aplicaciones desarrolladas en el sistema de realidad virtual para movimientos de motricidad fina, se ejecutó un experimento con cinco usuarios mediante la evaluación del test físico Box and Block y un test virtual Box and Block, el mismo que consiste en medir cuantos cubos transporta de una caja a otra en un tiempo de 1 minuto, con la mano izquierda y derecha. Esto permitirá determinar un error entre el Box and Blocks físico y el virtual.

Cabe indicar que para el experimento a cada usuario se le ubica en frente del Box and Block y de acuerdo a la mano con la que va a ejecutar el movimiento se cambia la posición de la caja siempre permaneciendo los cubos cerca de la mano de trabajo para mayor comodidad, como se muestra en la Figura 85.



**Figura 85.** Movimientos mediante el Test Box and Block físico vs virtual

### 5.2.1 Toma de datos de las manos izquierda y derecha

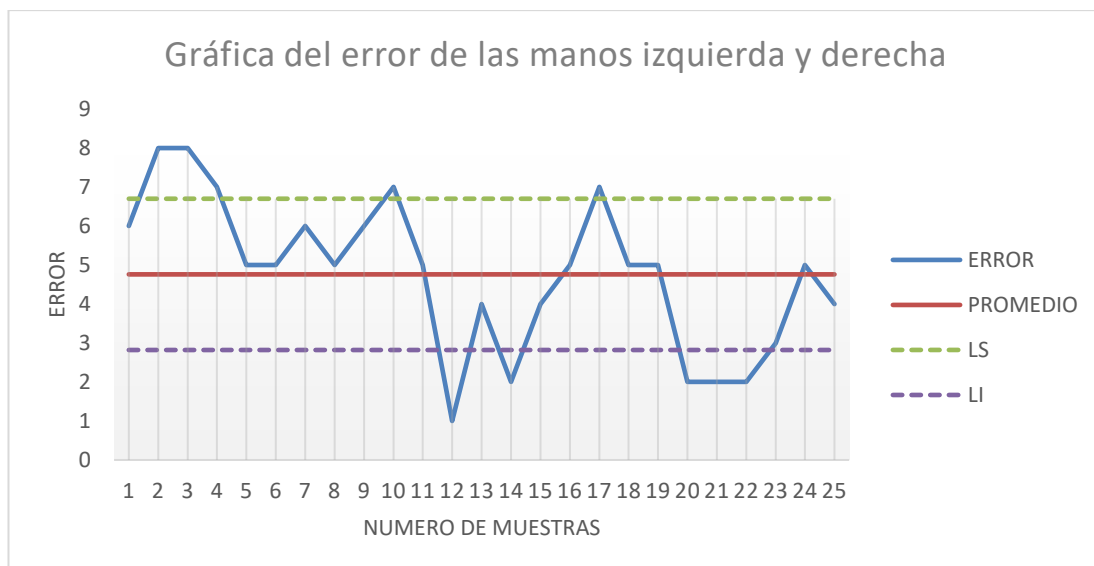
Para la adquisición de los datos, después que el usuario termina de realizar su tarea en 1 minuto, se procede a contar cuantos cubos han logrado



trasladar de una caja a otra pasando por el obstáculo, que dicho test posee, cabe indicar que se realizaron varias sesiones diferentes en 5 días, es decir, se tomaron cinco datos por cada usuario, dando un total de 25 muestras, las mismas que fueron analizadas logrando obtener los siguientes resultados como se muestra en la Tabla 7.

**Tabla 7.**  
**Datos de cubos colocados mediante Test Box and Blocks físico y virtual.**

N	Sistema Físico			Sistema Virtual			Valores			
	Izquierda	Derecha	Total Cubos	Izquierda	Derecha	Total Cubos	ERROR	VARIANZA	DESVIACION	
1	15	13	28	12	10	22	6	3,77	1,94	
2	18	16	34	14	12	26	8			
3	20	19	39	17	14	31	8			
4	19	17	36	15	14	29	7			
5	21	19	40	19	16	35	5			
6	13	15	28	11	12	23	5			
7	15	16	31	12	13	25	6			
8	15	17	32	12	15	27	5			
9	18	19	37	14	17	31	6			
10	22	18	40	15	18	33	7			
11	19	18	37	15	17	32	5			
12	19	16	35	16	18	34	1			
13	22	18	40	16	20	36	4			
14	25	20	45	21	22	43	2			
15	25	23	48	22	22	44	4			
16	17	19	36	14	17	31	5			
17	19	22	41	15	19	34	7			
18	19	24	43	17	21	38	5			
19	21	25	46	19	22	41	5			
20	22	26	48	22	24	46	2			
21	17	19	36	16	18	34	2			
22	19	20	39	17	20	37	2			
23	24	27	51	23	25	48	3			
24	26	27	53	22	26	48	5			
25	28	30	58	26	28	54	4			
	<b>PROMEDIO</b>						4,76			

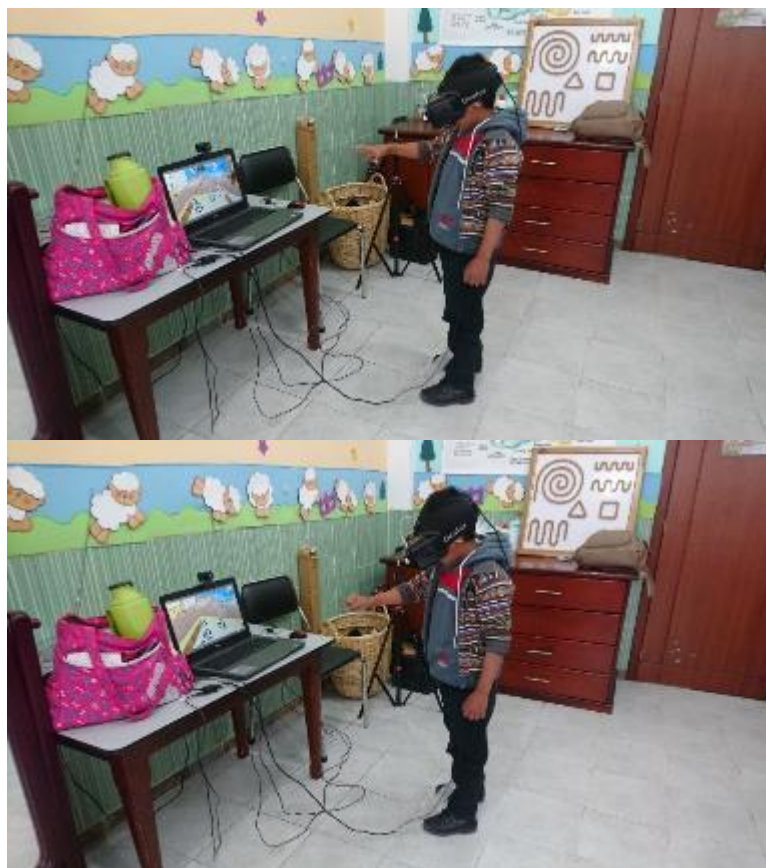


**Figura 86.** Error de mano derecha e izquierda con N=25

En base a los resultados obtenidos, se puede visualizar en la Figura 86 que la mayor cantidad de muestras están dentro de los rangos inferiores y superiores, teniendo un promedio de error de 4,76 y una desviación estándar de 1,94, los cuales son valores aceptables para utilizar los dispositivos en el sistema virtual.

### 5.3 Validación de Movimientos de Rehabilitación motora fina con el dispositivo LeapMotion.

La validación de los diversos movimientos que se realiza en el sistema para que sean adecuados, eficientes y en realidad ayuden a mejorar la parte motora fina de los niños, fue evaluado por una especialista en el área de terapia física, la misma que probó cada una de las aplicaciones y determinó que todo el sistema está basado en varios movimientos útiles y de rehabilitación para los niños que presentan algún déficit motor, estos movimientos son: agarre de pinza, flexión y extensión de muñeca, pronación, supinación, abducción, aducción y pinza bidigital. Además, indicó que cada una de las aplicaciones con ayuda del Oculus Rift DK2, los sonidos establecidos en las aplicaciones llevan a la persona que utiliza este sistema a estar inmerso en mundos diferentes. Cabe indicar que cada aplicación además ayuda a mejorar la parte de concentración mano-ojo en niños que presentan un déficit de concentración.



**Figura 87.** Agarre de pelota



**Figura 88.** Lanzamiento de pelota

#### 5.4 Validación del sistema de realidad virtual en niños

Se realizó las pruebas con varios niños que asisten a sus terapias diarias en el área de rehabilitación de la fundación “San Miguel”. Para la puesta en marcha del sistema, se utilizó un espacio libre dentro de la habitación, en el cual se colocó la cámara de forma fija sobre el computador, después se ubicó al niño frente de la cámara, se dio una explicación del uso de los dispositivos y los diversos movimientos que debe realizar para cumplir con los objetivos en cada juego, posteriormente se ingresó los datos personales como nombre, edad del niño para que se genere de forma automática un archivo .txt con su nombre, el mismo que servirá para después analizar los tiempos de ejecución de las aplicaciones. Luego, se le colocó el casco de realidad virtual y se procedió a la ejecución de las aplicaciones.



**Figura 89.** Instrucciones del sistema virtual al usuario



**Figura 90.** Ingreso de datos del usuario



**Figura 91.** Movimientos del usuario en el sistema virtual

Como primer punto, el niño entra a un entorno de menú que contiene varios paneles con diferentes botones de accesos, en el cual deberá moverse de forma tranquila hasta acercarse al botón que le llevará a la aplicación deseada y presionarlo con cualquiera de sus manos para que en cuestión de segundos trasladarse a un nuevo entorno.



**Figura 92.** Interacción del niño al menú principal

De acuerdo a la elección del niño, éste realizará varias tareas como puede ser agarre y lanzamiento de pelotas, clasificación de objetos por colores y formas, reventar globos con pinzas en los dedos, acertar aros mediante la flexión y extensión de muñeca, cruzar arcos con abducción de muñecas y como opción libre pintar figuras o escribir letras al realizar una pinza bidigital con su mano derecha.



a) Lanzamiento de pelotas y clasificación de objetos por colores



b) Visualización de colores en las manos y secuencia de pinzas por colores



c) Movimiento de flexión de la muñeca y Grafo perceptivo: Colorear a un perro

**Figura 93.** Movimientos del sistema de realidad virtual

## 5.5 Validación del sistema de realidad virtual mediante el Test de Usabilidad SEQ

El SEQ es un test que permite evaluar que tan eficiente es un sistema de realidad virtual en base a varias preguntas, el mismo que se aplica a 5 niños con patologías diferentes entre ellos: 2 niños con retraso mental leve y moderado, 1 niño con déficit de motricidad fina y 2 niños con Síndrome de Down leve y moderado. Estas preguntas fueron realizadas a los niños una vez concluido con la prueba de varias aplicaciones del sistema.

Una vez finalizado las pruebas a todos los usuarios, los datos se analizan uno por uno, los mismos que llevaron a determinar el valor promedio

y la desviación estándar de cada pregunta, como se puede observar en la Tabla 8. Para comprobar los resultados se basa en los estándares previamente establecidos por el Test SEQ, el mismo que da a conocer su validación cuando este posee una puntuación entre 40 y 65.

De acuerdo a los resultados obtenidos en forma general de los usuarios, se obtuvo un valor de 56.4, lo cual lleva a concluir que este Sistema de Realidad Virtual para Rehabilitación Motora Fina en niños tiene un nivel de usabilidad adecuado. También, se podría decir que este sistema es muy útil, eficiente y aceptable por los usuarios, ya que le encontraron como una actividad muy divertida.

**Tabla 8.**  
**Test SEQ con niños.**

Preguntas	Resultados (N=5)	
	Promedio	SD
P1. ¿Cuánto disfruto su experiencia con el sistema?	4.8	0.4
P2. ¿Cuánto sentiste estar en el ambiente del sistema?	4.4	0.8
P3. ¿Qué tan exitoso fue en el sistema?	3.8	0.75
P4. ¿Hasta qué punto fue capaz de controlar el sistema?	4	0.89
P5. ¿Qué tan real es el entorno virtual del sistema?	4.6	0.8
P6. ¿Está clara la información proporcionada por el sistema?	4.8	0.4
P7. ¿Ha sentido molestias durante su experiencia con el sistema?	1.2	0.4
P8. ¿Experimento mareos o nauseas durante su práctica con el sistema?	1.6	0.8
P9. ¿Experimento molestias oculares durante su práctica con el sistema?	1.4	0.48
P10. ¿Se sintió confundido o desorientada durante su experiencia con el sistema?	2.2	0.4
P11. ¿Cree usted que este sistema será útil para su rehabilitación?	4.4	0.49
P12. ¿Has encontrado la tarea difícil?	2.2	0.74
P13. ¿Encontraste que los dispositivos del sistema eran difíciles de usar?	1.8	0.97
<b>Total</b>	<b>56.4</b>	<b>0.37</b>

## 5.6 Confirmación de la hipótesis planteada

La comprobación de este sistema fue puesto a prueba con niños entre edades de 6 a 14 años y de diferentes patologías. A estos niños se le realizaron varias sesiones durante aproximadamente un mes y medio, en donde ellos comenzaron a probar el sistema juego por juego, el mismo que en base a los tiempos de finalización de cada aplicación se pudo evaluar su avance y mejora de su motricidad y concentración en cada uno de ellos.

Se estableció como prueba el análisis de cada juego por 5 días, como el sistema posee 7 aplicaciones o juegos, cada día se les iba probando un juego diferente de esta forma se logró realizar una tabla de ejecución de tiempos por juegos de cada usuario como se muestra en las siguientes tablas.

**Tabla 9.**  
**Tiempo por día, Primer Usuario**

Usuario 1		Tiempo (hh:mm:ss)				
Aplicación	Día 1	Día 2	Día 3	Día 4	Día 5	
Juego 1	00:05:00	00:04:15	00:03:50	00:03:18	00:02:58	
Juego 2	00:05:50	00:04:15	00:03:52	00:03:40	00:03:25	
Juego 3	00:06:00	00:06:00	00:05:35	00:04:50	00:03:45	
Juego 4	00:05:53	00:05:26	00:04:38	00:04:09	00:03:44	
Juego 5	00:08:13	00:06:00	00:05:06	00:04:58	00:04:45	
Juego 6	00:08:02	00:08:00	00:07:52	00:06:30	00:05:10	

**Tabla 10.**  
**Tiempo por día, Segundo Usuario**

Usuario 2		Tiempo(hh:mm:ss)				
Aplicación	Día 1	Día 2	Día 3	Día 4	Día 5	
Juego 1	00:09:11	00:09:07	00:08:43	00:08:12	00:07:36	
Juego 2	00:07:58	00:07:12	00:06:55	00:06:22	00:05:41	
Juego 3	00:07:38	00:07:11	00:06:48	00:06:17	00:06:03	
Juego 4	00:06:47	00:06:25	00:06:00	00:05:39	00:05:11	
Juego 5	00:09:52	00:09:47	00:09:28	00:09:02	00:08:55	
Juego 6	00:09:28	00:09:13	00:08:59	00:08:24	00:08:12	



**Tabla 11.**  
**Tiempo por día, Tercer Usuario**

<b>Usuario 3</b>		<b>Tiempo(hh:mm:ss)</b>				
<b>Aplicación</b>	<b>Día 1</b>	<b>Día 2</b>	<b>Día 3</b>	<b>Día 4</b>	<b>Día 5</b>	
<b>Juego 1</b>	00:06:13	00:05:32	00:04:57	00:04:33	00:04:20	
<b>Juego 2</b>	00:07:30	00:07:17	00:06:53	00:05:52	00:05:08	
<b>Juego 3</b>	00:07:09	00:06:59	00:06:35	00:05:47	00:04:59	
<b>Juego 4</b>	00:07:26	00:06:57	00:05:29	00:04:43	00:04:21	
<b>Juego 5</b>	00:07:10	00:06:48	00:06:07	00:05:28	00:05:03	
<b>Juego 6</b>	00:07:54	00:07:23	00:06:56	00:06:14	00:05:25	

**Tabla 12.**  
**Tiempo por día, Cuarto Usuario**

<b>Usuario 4</b>		<b>Tiempo(hh:mm:ss)</b>				
<b>Aplicación</b>	<b>Día 1</b>	<b>Día 2</b>	<b>Día 3</b>	<b>Día 4</b>	<b>Día 5</b>	
<b>Juego 1</b>	00:10:30	00:10:09	00:09:42	00:09:33	00:08:31	
<b>Juego 2</b>	00:11:05	00:10:29	00:10:01	00:09:42	00:08:59	
<b>Juego 3</b>	00:11:18	00:11:07	00:10:44	00:09:57	00:09:32	
<b>Juego 4</b>	00:10:20	00:10:00	00:09:12	00:08:18	00:07:35	
<b>Juego 5</b>	00:15:29	00:14:55	00:14:31	00:13:58	00:13:18	
<b>Juego 6</b>	00:14:42	00:14:22	00:13:56	00:13:41	00:12:58	

**Tabla 13.**  
**Tiempo por día, Quinto Usuario**

<b>Usuario 5</b>		<b>Tiempo(hh:mm:ss)</b>				
<b>Aplicación</b>	<b>Día 1</b>	<b>Día 2</b>	<b>Día 3</b>	<b>Día 4</b>	<b>Día 5</b>	
<b>Juego 1</b>	00:17:06	00:16:57	00:16:31	00:16:00	00:15:41	
<b>Juego 2</b>	00:15:25	00:15:02	00:14:55	00:14:33	00:14:21	
<b>Juego 3</b>	00:15:38	00:15:06	00:14:48	00:14:22	00:14:03	
<b>Juego 4</b>	00:14:57	00:14:20	00:13:54	00:13:26	00:12:58	
<b>Juego 5</b>	00:16:31	00:15:54	00:15:30	00:14:47	00:14:22	
<b>Juego 6</b>	00:15:52	00:15:32	00:14:49	00:14:08	00:13:49	

Como se puede observar en las tablas anteriores, los tiempos que los usuarios emplearon para completar cada uno de los juegos por cada día y efectuando un análisis en base a los datos obtenidos, se determina que con el pasar de los días los resultados fueron mejorando, es decir reduciendo su duración en cada aplicación al realizar el objetivo de cada juego.

Además, a través de los resultados obtenidos con el experimento utilizando el test box and block físico y el desarrollado de forma virtual, se afirma que los dos dispositivos utilizados en este sistema permiten realizar varios movimientos de rehabilitación motora fina.



**Figura 94.** Test Block and Block físico y virtual

Una vez comprobado que los dispositivos de tracking óptico permiten realizar movimientos y con la información recolectada por todo el sistema de realidad virtual, se lleva a concluir que el sistema es muy eficiente y es un complemento de la rehabilitación tradicional ya que ayuda a mejorar la parte motor de cada usuario con los diversos movimientos que se realizan en cada juego.



**Figura 95.** Movimiento agarre de gancho



**Figura 96.** Movimiento de agarre de pinza digital

## CAPÍTULO VI

### 6. CONCLUSIONES Y RECOMENDACIONES

#### 6.1 Conclusiones

- En base a la información obtenida durante la investigación del proyecto, se pudo determinar que la motricidad fina requiere de coordinación y precisión en algunos movimientos como son: agarrar, desplazar objetos, soltar, manipular objetos con la mano, tomando en cuenta que para mejorar la motricidad se necesita ejercitar los pequeños músculos con ejercicios de forma regular.
- Para el desarrollo de juegos virtuales existen varios software de desarrollo como Corona SDK, Gamemaker, CryEngie, Unity5, entre otros., pero en base a un análisis de parámetros que conforman cada software se ha logrado determinar que la plataforma Unity es la que cuenta con mejores prestaciones y tiene una alta compatibilidad con varias plataformas, y además es compatible con los dispositivos como Oculus y LeapMotion que fueron utilizados en el desarrollo de este proyecto.
- Las rehabilitaciones tradicionales que se realizan en los centros de salud, hospitales generales o centros de rehabilitación son muy rutinarias y aburridas, por lo que los usuarios en ocasiones se incomodan, motivo por el cual se creó un sistema de rehabilitación virtual complementario con varios entornos amigables, intuitivos e interactivos los cuales motivan a los usuarios a realizar sus terapias de rehabilitación para mejorar la habilidad de motricidad fina.
- Analizando los algoritmos necesarios para la detección de los diferentes movimientos de la mano tanto en LeapMotion como en Kinect, se observa que el dispositivo Kinect tiene limitaciones al definir

movimientos específicos como son: agarre de objetos, flexión, extensión, aducción y supinación de muñeca, entre otros; ya que éste detecta a cada extremidad como un solo objeto discriminando así las manos y dedos, para lo cual la mejor opción es LeapMotion debido a que el dispositivo proporciona la posición de la mano en tiempo real.

- En la Fundación “San miguel” de la ciudad de Salcedo se realizaron pruebas funcionales del sistema, con 5 niños con patologías de síndrome de Down, retraso mental y espasticidad motora fina, los mismos que al probar cada aplicación pudieron realizar varias destrezas manuales que involucran: agarre de pinza o gancho, flexión y extensión de muñeca, abducción, supinación y pinza bidigital, por lo que se obtuvo gran acogida del sistema de realidad virtual complementando a la rehabilitación tradicional.

## **6.2 Recomendaciones**

- Para la implementación de este sistema es necesario disponer de una computadora que cumpla con ciertas características como: disponer de un buen procesador, una tarjeta gráfica Nvidia GTX 970 o superiores a esta, para que la ejecución de la aplicación sea buena.
- Debido a que la detección de las manos se dan siempre y cuando éstas se encuentren al frente del dispositivo LeapMotion en un radio de 61cm, lo cual limita y puede llegar a cansar a los usuarios, por lo que sería recomendable utilizar otros elementos que detecten los movimientos en cualquier posición como son los guantes de realidad virtual.
- La rehabilitación virtual se debe emplear en varias sesiones por cada aplicación, manteniendo siempre un tiempo de descanso para que el usuario no presente molestias al utilizar por un largo período de tiempo las gafas Oculus Rift DK2.

## REFERENCIAS BIBLIOGRÁFICAS

ABC Tecnología. (12 de Mayo de 2014). Recuperado el: 5 de marzo del 2017. Obtenido de: <http://www.abc.es/tecnologia/informatica-hardware/20140326/abci-oculus-rift-funcionamiento-video-01403261225.html>

Abubakra, Y. A., Herrero, M. A., Fernández Villanueva, J., Martín Sanz, D., & Zabala Hidalgo, A. (19 de Junio de 2015). Recuperado el: 8 de abril del 2017. Obtenido de: <http://eprints.ucm.es/32965/1/TFG%20Producci%C3%B3n%20de%20una%20experiencia%20de%20horror%20inmersivo%20utilizando%20un%20entorno%20de%20desarrollo%20de%20videojuego.pdf>

Acaro Acaro, M. G. (2011). Programa de rehabilitación de la motricidad gruesa y fina en los niños y niñas con Síndrome de Down de 1 a 5 años de edad que asisten al Centro de Rehabilitación Angelitos de Luz, de la ciudad de Loja, durante el período octubre 2009 a mayo del 2010. Loja, Ecuador: Universidad Nacional de Loja.

Albiol Pérez, S. (Marzo de 2014). Rehabilitación Virtual Motora: una Evaluación al tratamiento de pacientes con Daño Cerebral Adquirido. Valencia.

Almá, A. (1 de Octubre de 2014). inLab FIB. Recuperado el: 3 de abril del 2017. Obtenido de: <http://inlab.fib.upc.edu/es/blog/interaccion-3d-con-leap-motion>

Andrade, F. J. (s.f.). Recuperado el: 25 de mayo del 2017. Obtenido de: <http://www.ridaa.unicen.edu.ar/xmlui/bitstream/handle/123456789/601/Tesis%20de%20Grado%20Andrade%20Facundo.pdf?sequence=1&isAllowed=y>

Belda, J. (20 de Abril de 2015). Showleap Technologies. Recuperado el: 18 de agosto del 2017. Obtenido de: <http://blog.showleap.com/2015/04/leap-motion-caracteristicas-tecnicas/>

Berruezo Adelantado, P. P. (2008). El contenido de la Psicomotricidad. Reflexiones para la delimitación de su ámbito teórico y práctico. *Revista Interuniversitaria de Formación del Profesorado*, pp. 19-34.

Blanchard Cruz, J. O. (3 de Marzo de 2016). Diseño e implementación de un prototipo de videojuego de Realidad Virtual. La Laguna.

Cerón Méndez, T., & Gutiérrez Botero, P. A. (Octubre de 2015). Desarrollo de un entorno didáctico como herramienta de apoyo para el aprendizaje en los niños y niñas de 4 a 6 años utilizando realidad aumentada. Cali, Colombia: Universidad de San Buenaventura.

Chuva Castillo, P. G. (Agosto de 2016). Desarrollo de la motricidad Fina a través de técnicas grafoplásticas en niños de 3 a 4 años de la Escuela Básica Federico González Suárez. Ecuador: Universidad Politécnica Salesiana.

Difementes.com. (2017). Recuperado el: 28 de abril del 2017. Obtenido de: <http://difementes.com/realidadvirtual/index.html>

Escartín, E. R. (2017). La realidad virtual, una tecnología educativa a nuestro alcance. Recuperado el: 14 de mayo del 2017. Obtenido de: [https://idus.us.es/xmlui/bitstream/handle/11441/45510/file\\_1.pdf?sequence=1&isAllowed=y](https://idus.us.es/xmlui/bitstream/handle/11441/45510/file_1.pdf?sequence=1&isAllowed=y)

Facultad de Informatica de Barcelona. (2017). Retro Informática. Recuperado el: 25 de junio del 2017. Obtenido de: <http://www.fib.upc.edu/retro-informatica/avui/realitatvirtual.html>

Fernández Arroyo, A. (Septiembre de 2013). Simulador de accidentes de tráfico mediante motor de videojuegos Unity. Universidad Politécnica de Cartagena.

Gafas Oculus. (15 de Agosto de 2015). Recuperado el: 6 de abril del 2017. Obtenido de: <http://www.gafasoculus.com/rift/>

Gandulfo, M., & Young, S. (24 de Noviembre de 2011). Juguete Terapéutico para desarrollar la motricidad fina en niños con

TEA(Trastorno de Espectro Autista) de 1 a 6 años. Universidad de Buenos Aires.

Garrido, R. (23 de Julio de 2013). Recuperado el: 5 de agosto del 2017. Obtenido de: <https://www.xatakawindows.com/hardware/leap-motion-el-dispositivo-para-controlar-windows-8-por-gestos-ya-esta-disponible>

Gerro Prinsloo, R. D. (2015). Book Edition.

Gil Madrona, P., Gómez Barreto, I. M., & Contreras Jordán, O. R. (21 de Julio de 2017). Habilidades motrices en la infancia y su desarrollo desde una educación física animada. ResearchGate, pp. 71-96.

Hervás Torío, A. I. (2014). Juego Serio para entrenamiento de personas con discapacidad física mediante tecnología Kinect. Valladolid: Universidad de Valladolid. Escuela Técnica Superior de Ingenieros de Telecomunicación.

Michán, M. (22 de Mayo de 2012). Recuperado el: 26 de mayo del 2017. Obtenido de: <https://www.applesfera.com/accesorios/leap-motion-el-sistema-de-control-gestual-mas-barato-y-preciso-que-kinect-que-dificilmente-triunfara>

Montufar, B. (13 de Enero de 2012). Recuperado el: 5 de agosto del 2017. Obtenido de: <https://es.scribd.com/doc/78113183/SENSORES-OPTICOS>

Pérez Colado, I. J., & Pérez Colado, V. M. (20 de Junio de 2014). Un Conjunto de herramientas para Unity orientado al desarrollo de videojuegos de acción-aventura y estilo retro con gráficos isométricos 3D. Madrid, España: Universidad Complutense de Madrid.

Pérez, C., & Reinoso, O. (23 de Mayo de 2003). Recuperado el: 12 de julio del 2017. Obtenido de: <http://www.disc.ua.es/controlvisual/ponencias/SVCPCMumh.pdf>

Psicología de la educación psicomotriz. (1994). En A. Mesonero Valhondo. Oviedo: Textos Universitarios Ediuno.



Rodríguez Sánchez, M. Y. (2014). Programa didáctico basado en el método Montessori para estimular la lectura y escritura en los niños de 5 años de edad de la I.E.E. "Rafael Narvaéz cadenillas" de la ciudad de Trujillo, en el año 2014. Trujillo, Perú: Universidad Nacional de Trujillo.

Rojas Ortiz, J. R. (Mayo de 2011). Beneficios terapéuticos de los videojuegos para el desarrollo físico en niños. Universidad Metropolitana Recinto de Cupey.

Salcedo Maldonado, M. (2011). Recuperado el: 2 de marzo del 2017. Obtenido de <http://www.bdigital.unal.edu.co/11142/1/598081.2011.pdf>

Soriano Maudes, P. (2012). Estudio de viabilidad del entorno de programación Unity3D para la Rehabilitación cognitiva contra el Alzheimer. Gandia: Universidad Politécnica de Valencia.

Vargas Herrera, D. (13 de Febrero de 2017). Sistema de Espacios Virtuales para Neuro-rehabilitación. Recuperado el: 12 de abril del 2017. Obtenido de <http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/10619/tesis.pdf?sequence=1>

# ANEXOS



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA  
CARRERA DE ELECTRÓNICA E INSTRUMENTACIÓN**

**CERTIFICACIÓN**

Se certifica que el presente trabajo fue desarrollado por las señoritas: **Elsa Alexandra Chuquitarco Aguayo y Jenny Paola Tigse Cando.**

En la ciudad de Latacunga, a los 20 días del mes de octubre del 2017.

---

**Ing. Marco Pilatasig  
DIRECTOR DEL PROYECTO**

Aprobado por:

---

**Ing. Franklin Silva Monteros  
DIRECTOR DE CARRERA**

---

**Dr. Rodrigo Vaca  
SECRETARIO ACADÉMICO**