



# UNIVERSIDAD DE ALMERÍA

ESCUELA POLITÉCNICA SUPERIOR Y FACULTAD DE  
CIENCIAS EXPERIMENTALES

INGENIERÍA INFORMÁTICA

**KiNEEt: Aplicación para el aprendizaje y rehabilitación en  
educación especial**

**El Alumno:**

**Juan Jesús Ojeda Castelo**

**Almería, Marzo de 2014**

**Director(es):**

**José Antonio Piedra Fernández**

**César Bernal Bravo**

*Usamos las TIC como vía de comunicación  
para personas que perciben el mundo de manera diferente.*

---

# Agradecimientos

A mi familia, por su apoyo incondicional, en especial a mis padres, por su comprensión en los momentos difíciles y estar a mi lado cuando más falta me hacía.

A José Antonio Piedra, por su guía y consejos durante la realización de este proyecto y lo más importante, su formación como persona no sólo como ingeniero, inculcándome que puedo mucho más de lo que aparento y que sólo necesito un pequeño empujón para despegar.

A César Bernal, por ser la llave que me permitió entrar en el CEEE Provincial Princesa Sofía y por hacerme entender el mundo de la pedagogía, que era totalmente desconocido para mí.

A Mercedes Lorena Pedrajas, por su colaboración como psicopedagoga en las primeras fases del proyecto y estar dispuesta a echar una mano cuando hiciera falta.

A Aida Sancho, por esas ilustraciones que le han dado vida a la aplicación.

A Diego Cangas, porque la parte 3D del proyecto no hubiera sido lo mismo sin sus avatares y animaciones.

A José Aguado, por estar siempre dispuesto a ayudarme en lo que hiciera falta del proyecto, aportando sus ingeniosas ideas para darle un toque de calidad al proyecto.

Al CEEE Provincial Princesa Sofía, porque cuando estaba allí rodeado de mis niños y de los excelentes profesionales que lo forman, dudaba si me encontraba en el centro o en mi casa. Con especial dedicación, a las docentes, Laura, Luisa y Rocío, por poner baldosas donde antes sólo había tierra y crear un camino donde todos somos iguales y el término de discapacidad sea efímero.



# Índice general

<b>Agradecimientos</b>	<b>5</b>
<b>Lista de figuras</b>	<b>11</b>
<b>Lista de tablas</b>	<b>15</b>
<b>1. Introducción</b>	<b>17</b>
1.1. Análisis del problema . . . . .	18
1.2. Objetivos . . . . .	19
1.3. Metodología . . . . .	20
1.4. Recursos . . . . .	25
1.5. Planificación . . . . .	25
1.6. Cronograma . . . . .	27
1.7. Estructura del documento . . . . .	27
<b>2. Estado del Arte</b>	<b>29</b>
2.1. Introducción . . . . .	29
2.2. Herramientas . . . . .	29
2.3. Aplicación KiNEEt . . . . .	34
2.4. Comparación entre las diferentes herramientas . . . . .	35
<b>3. Tecnología</b>	<b>39</b>
3.1. Kinect . . . . .	39
3.1.1. Componentes . . . . .	40
3.1.2. Campo de actuación . . . . .	41
3.1.3. Kinect SDK . . . . .	42
3.2. XNA . . . . .	44
3.2.1. Ciclo de vida de un juego . . . . .	45
3.2.2. La clase DrawableGameComponent . . . . .	46
3.3. Unity . . . . .	47
3.3.1. Assets . . . . .	47

3.3.2.	Escena . . . . .	48
3.3.3.	Game Object . . . . .	49
3.3.4.	Prefabs . . . . .	51
3.3.5.	Scripts . . . . .	51
3.3.6.	Descubriendo la interface . . . . .	51
3.3.7.	Programación con scripts . . . . .	57
3.4.	Principios recomendados entre TIC y discapacidad . . . . .	58
3.4.1.	Principios básicos . . . . .	59
<b>4.</b>	<b>Aplicación KiNEEt</b>	<b>63</b>
4.1.	Arquitectura . . . . .	63
4.2.	Proceso del Software . . . . .	65
4.2.1.	Modelo del proceso del software . . . . .	65
4.2.2.	Actividades del proceso de software . . . . .	66
4.2.3.	Iteración de procesos . . . . .	68
4.2.4.	Ciclos del desarrollo en espiral . . . . .	68
4.2.4.1.	Ciclo del módulo de actividades . . . . .	69
4.2.4.2.	Ciclo del módulo de coordinación . . . . .	75
4.2.4.3.	Ciclo del módulo de arte . . . . .	77
4.2.4.4.	Ciclo del módulo de comunicador . . . . .	80
4.3.	Implementación . . . . .	83
4.3.1.	Movimiento del cursor . . . . .	83
4.3.2.	El proceso de selección de actividades . . . . .	83
4.3.3.	Método de colisiones con bounding box . . . . .	84
4.3.4.	Obtener el skeleton más cercano . . . . .	84
4.3.5.	Proceso de reconocimiento de gestos . . . . .	85
4.3.6.	Base de Datos . . . . .	86
4.4.	Pruebas . . . . .	87
<b>5.</b>	<b>Resultados</b>	<b>89</b>
5.1.	Estudio de la aplicación . . . . .	89
5.1.1.	Evaluación Usabilidad . . . . .	91
5.1.2.	Evaluación Uso Educativo . . . . .	94
5.1.3.	Evaluación Período de tiempo . . . . .	97
5.1.4.	Desarrollo y Resultados en el CEEE Princesa Sofía . . . . .	97
5.2.	Estudio de los alumnos . . . . .	100
<b>6.</b>	<b>Conclusiones y Trabajos Futuros</b>	<b>105</b>
6.1.	Difusión . . . . .	105
6.2.	Conclusiones . . . . .	107
6.3.	Trabajos Futuros . . . . .	109

<b>Bibliografía</b>	<b>110</b>
<b>A. Manual de instalación</b>	<b>115</b>
A.1. Instalar SDK para Kinect . . . . .	115
A.2. Instalar XNA Framework Redistributable 4.0 . . . . .	118
A.3. Aplicación KiNEEt . . . . .	120
<b>B. Manual de usuario</b>	<b>125</b>
B.1. Elementos comunes . . . . .	126
B.1.1. Iconos estándar . . . . .	126
B.1.2. Pantalla de opciones . . . . .	127
B.2. Pantalla de Inicio . . . . .	128
B.3. Módulo Coordinación . . . . .	129
B.3.1. Configuración Coordinación . . . . .	129
B.3.2. Edición Coordinación . . . . .	130
B.3.3. Actividad de Coordinación . . . . .	131
B.4. Módulo Actividades . . . . .	132
B.4.1. Selección de actividades . . . . .	132
B.4.2. Configuración Números . . . . .	133
B.4.3. Actividad Números . . . . .	135
B.4.4. Configuración Formas Geométricas . . . . .	135
B.4.5. Actividad Formas Geométricas . . . . .	137
B.4.6. Configuración Grafomotricidad . . . . .	137
B.4.7. Actividad Grafomotricidad Horizontal . . . . .	138
B.4.8. Actividad Grafomotricidad Vertical . . . . .	139
B.4.9. Actividad Grafomotricidad Libre . . . . .	140
B.4.9.1. Edición Grafomotricidad Libre . . . . .	140
B.4.9.2. Realización de la Actividad . . . . .	141
B.4.10. Configuración Causa – Efecto . . . . .	142
B.4.11. Actividad Causa – Efecto . . . . .	144
B.5. Módulo Comunicador . . . . .	144
B.5.1. Configuración Comunicador . . . . .	144
B.5.2. Actividad Comunicador . . . . .	145
B.6. Módulo Arte . . . . .	157
B.6.1. Pantalla de Configuración Genérica para Avatares . . . . .	157
B.6.2. Actividad de Música con Platillos . . . . .	157
B.6.3. Actividad de Música con Piano . . . . .	158
B.6.4. Actividad de Música Conjunta . . . . .	159
B.6.5. Actividad de Danza Interactiva . . . . .	160
B.6.6. Actividad de Danza con Objetos . . . . .	161
B.6.7. Configuración Pintura . . . . .	162

B.6.8. Actividad Pintura . . . . .	163
C. Cuestionarios de la Aplicación	167
D. Cuestionario de las Capacidades del Alumnado	169
E. Diagrama de Casos de Uso	171
F. Diagrama de Flujo de Coordinación y Comunicador	173
G. Diagrama de Flujo de Actividades	175
H. Diagrama de Flujo de Arte	177
I. Diagrama de Clases	179

# Índice de figuras

2.1. Pictogram Room. ( <a href="http://fundacionorange.es">http://fundacionorange.es</a> ) . . . . .	30
2.2. Virtual Rehab. . . . .	31
2.3. Proyecto Teki. ( <a href="http://www.ilitia.com/Expertise/Teleasistencia_basada_en_Kinect_Teki">http://www.ilitia.com/Expertise/Teleasistencia_basada_en_Kinect_Teki</a> ) . . . . .	33
2.4. Neuro at Home. ( <a href="http://www.neuroathome.net/p/tratamientos.html">http://www.neuroathome.net/p/tratamientos.html</a> ) . . . . .	33
2.5. Tango:H. ( <a href="http://www.iter.es/index.php/mod.pags/mem.detalle/idpag.238/idmenu.1280/">http://www.iter.es/index.php/mod.pags/mem.detalle/idpag.238/idmenu.1280/</a> ) . . . . .	33
3.1. Componentes Kinect. (Abhijit, J.2012) . . . . .	41
3.2. Zona de actuación Kinect. (Catuhe, D.2012) . . . . .	42
3.3. Logo XNA . . . . .	44
3.4. Ciclo de vida de un videojuego XNA . . . . .	45
3.5. Logo Unity . . . . .	47
3.6. Menú Assets . . . . .	48
3.7. Crear nueva escena . . . . .	49
3.8. Crear Game Object desde el panel Hierarchy . . . . .	50
3.9. Crear Game Object desde el menú GameObject . . . . .	51
3.10. Tipos de layout . . . . .	52
3.11. Panel de Jerarquía . . . . .	53
3.12. Panel de Proyecto . . . . .	53
3.13. Panel Inspector . . . . .	55
3.14. Panel de Escena . . . . .	56
3.15. Panel de Juego . . . . .	56
3.16. Creación de un script en C# . . . . .	57
3.17. Script vacío en C# . . . . .	57
3.18. Diseño Universal e Inteligencia Ambiental . . . . .	61
4.1. Arquitectura del sistema . . . . .	64
4.2. Clase de la actividad de Números . . . . .	70
4.3. Clase de la actividad de Formas Geométricas . . . . .	71
4.4. Clase de la actividad de Causa-efecto . . . . .	73
4.5. Clase de la actividad de Grafomotricidad . . . . .	74
4.6. Clase de la actividad de Coordinación . . . . .	76

4.7. Clase de la actividad de Pintar . . . . .	79
4.8. Clase de la actividad del Comunicador . . . . .	81
4.9. Método getMinDepth() . . . . .	85
4.10. Elección del Skeleton más cercano . . . . .	85
4.11. Lectura de archivo XML . . . . .	87
5.1. Evaluación de Usabilidad . . . . .	91
5.2. Evaluación Uso educativo . . . . .	94
5.3. Evaluación Capacidades del Alumnado . . . . .	102
A.1. Descargar SDK Kinect . . . . .	115
A.2. Instalación SDK . . . . .	116
A.3. Instalación SDK . . . . .	117
A.4. Instalación SDK . . . . .	117
A.5. Descargar XNA . . . . .	118
A.6. Instalación XNA . . . . .	119
A.7. Instalación XNA . . . . .	119
A.8. Instalación XNA . . . . .	120
A.9. Copiar archivo . . . . .	121
A.10.Descomprimir archivo . . . . .	121
A.11.Instalar voz . . . . .	122
A.12.Carpeta KiNEEt . . . . .	122
A.13.Doble click en Kineet.exe . . . . .	123
B.1. Splash Screen . . . . .	125
B.2. Icono inicio . . . . .	126
B.3. Icono aceptar . . . . .	126
B.4. Icono reiniciar . . . . .	126
B.5. Icono salir . . . . .	127
B.6. Pantalla de opciones . . . . .	128
B.7. Pantalla de Inicio . . . . .	129
B.8. Pantalla de Configuración de Coordinación . . . . .	130
B.9. Pantalla de Edición de Coordinación . . . . .	131
B.10.Pantalla de Coordinación . . . . .	132
B.11.Pantalla de Selección de Actividades . . . . .	133
B.12.Pantalla de Configuración de Números . . . . .	134
B.13.Pantalla de la Actividad de Números . . . . .	135
B.14.Pantalla de Configuración de Formas Geométricas . . . . .	136
B.15.Pantalla de la Actividad de Formas Geométricas . . . . .	137
B.16.Pantalla de Configuración de Grafomotricidad . . . . .	138
B.17.Pantalla de Grafomotricidad Horizontal . . . . .	139

## ÍNDICE DE FIGURAS

---

B.18.Pantalla de Grafomotricidad Vertical . . . . .	140
B.19.Pantalla de Edición de Grafomotricidad Libre . . . . .	141
B.20.Pantalla de Grafomotricidad Libre . . . . .	142
B.21.Pantalla de Configuración de Causa-Efecto . . . . .	143
B.22.Pantalla de Configuración de Comunicador . . . . .	145
B.23.Zona de Pictogramas Seleccionados . . . . .	146
B.24.Pantalla de Identificación . . . . .	147
B.25.Pantalla de Peticiones . . . . .	147
B.26.Pantalla de Acciones . . . . .	148
B.27.Pantalla de Objetos . . . . .	148
B.28.Pantalla Final . . . . .	149
B.29.Pantalla de Configuración Genérica . . . . .	157
B.30.Pantalla de Actividad de Platos . . . . .	158
B.31.Pantalla de Actividad de Piano . . . . .	159
B.32.Pantalla de Actividad de Música Conjunta . . . . .	160
B.33.Pantalla de Actividad de Danza interactiva . . . . .	161
B.34.Pantalla de Actividad de Danza con objetos . . . . .	162
B.35.Pantalla de Actividad de Configuración de Pintura . . . . .	163
B.36.Pantalla de Actividad de Pintura . . . . .	164



# Índice de cuadros

1.1. Plan de trabajo . . . . .	26
2.1. Tabla comparativa de las diferentes herramientas . . . . .	36
3.1. Tabla Comparativa Microsoft SDK y OpenNI . . . . .	43
4.1. Tabla Casos de Uso del Módulo Actividades . . . . .	69
4.2. Tabla Casos de Uso del Módulo Coordinación . . . . .	76
4.3. Tabla Casos de Uso del Módulo Arte . . . . .	78
4.4. Tabla Casos de Uso del Módulo Comunicador . . . . .	81
5.1. Alumnos del CEEE Princesa Sofía usando KiNEEt . . . . .	100
B.1. Tabla de Gestos y Comandos por Voz reconocidos por el Comunicador	150
B.2. Tabla de Gestos . . . . .	153
B.3. Tabla de Alimentos . . . . .	154
B.4. Tabla de Bebidas . . . . .	155
B.5. Tabla de Juguetes . . . . .	156
C.1. Usabilidad . . . . .	167
C.2. Uso educativo . . . . .	168



# Capítulo 1

## Introducción

Las personas que presentan algún tipo de discapacidad no es algo inusual ni extraño, de hecho una de cada seis personas de la Unión Europea presentan estas características, lo que nos lleva a la cantidad de 80 millones de personas[1]. Ellos, al igual que el resto de personas que residen en Europa tienen una serie de derechos enmarcados dentro de la Carta de los Derechos Fundamentales de la Unión Europea, entre estos derechos se encuentra el de igualdad, el cual representa que las personas con discapacidad tienen que ser autónomos e independientes y vivir su vida con total naturalidad, apartando a un lado las barreras físicas como mentales que se han impuesto durante un período de tiempo extenso tanto en la sociedad como en las personas que confluyen en ella. Una de las partes esenciales de la vida es la actividad laboral, necesaria para poder subsistir e imprescindible en el objetivo fundamental que persiguen estas personas y se ha recalado anteriormente, que es conseguir su independencia.

Aunque las personas con discapacidad tengan derecho a la integración laboral, la realidad es que las personas con discapacidad tienen un índice de pobreza del 70 % superior a la media, una de las causas fundamentales es el hecho de tener un menor acceso al empleo. La discapacidad física juega un papel importante en el infortunio, de tener una mayor complejidad de obtener un trabajo, ya que si los afectados presentan alguna dificultad en las habilidades motrices y de desplazamiento, esta situación confluye, en que si el puesto de trabajo no está adaptado, no se puede optar a la plaza de trabajo. Una de las soluciones viables ante este contratiempo, es la realización de una terapia de rehabilitación[4] para recuperar parte de la movilidad perdida y gozar de una mayor autonomía a nivel motor.

Uno de los elementos clave de toda terapia de rehabilitación, suele ser la ejecución de una serie de movimientos concretos dependiendo del diagnóstico realizado por el especialista, no obstante, existe otra disciplina, el arte[2], que incluye el movimiento como práctica habitual en el desarrollo de sus actividades, ya que para realizar danza, teatro o pintura, es fundamental este hecho. Esta reflexión conduce a la idea de que un complemento idóneo para el sistema de rehabilitación pueda ser realizar unos ejercicios asociados al arte, como por ejemplo el baile o la realización trazos en un lienzo. De esta forma no sólo se promueve el desarrollo a nivel físico, sino que se potencia la creatividad, hace al individuo más perceptivo y va adquiriendo más madurez con esta práctica, lo cual es fundamental en el día a día.

Por último, el elemento idóneo que encaja en este contexto es el dispositivo Kinect, creado por Microsoft y orientado a la consola XBOX 360, como complemento de entrenamiento de este aparato, aunque es posible utilizarlo actualmente en un PC e incluso desarrollar aplicaciones en las cuales este sensor tenga un papel protagonista en la interacción. La novedad y revolución de este dispositivo radica en que es posible la interacción con un software a través del movimiento del cuerpo, sin ningún elemento adicional que pueda entorpecer o hacer más difícil el movimiento, es una experiencia tan natural como bailar, lo que hace pensar que existe una relación directa entre el dispositivo Kinect y las actividades descritas anteriormente.

## 1.1. Análisis del problema

El ámbito de la discapacidad es variado, existen varios tipos de afecciones, un individuo puede poseer discapacidad física, mental, etc. Está claro, que independientemente del tipo de discapacidad que presente, habrá tareas que puedan realizar sin problema y otras que le cueste más, no presenta las mismas necesidades una persona que posea una discapacidad visual, que otra que presente problemas de coordinación, cada uno necesita unas especificaciones diferentes en su entorno para poder realizar las tareas de su día a día como una persona normal. Sin embargo, que una persona posea discapacidad no implica que no pueda realizar una determinada actividad sino que la tendrá que hacer de forma diferente.

En la educación, se utilizan métodos de enseñanza con el alumnado que presenta algún tipo de discapacidad, que implica el uso de instrumentos como libros, pictogramas o incluso balones. Aunque es verdad que también se utilizan las TIC para el procedimiento de aprendizaje del alumno, como puede

## 1.2. OBJETIVOS

---

ser la pizarra digital. La cuestión es que a pesar de utilizar medios digitales y tecnológicos para el desarrollo de habilidades del alumno, éstos necesitan de unos elementos externos, ya sean periféricos como el ratón y el teclado o en el caso de la pizarra digital, es necesaria la utilización de un bolígrafo para su uso. Este planteamiento muestra que por norma general, hay que tener un contacto físico con un dispositivo de entrada para poder hacer uso de esta tecnología. Teniendo en cuenta que según el grado y el tipo de discapacidad física que posea el individuo puede resultar más o menos complicado la realización del ejercicio que en el caso de que pudiera realizar la actividad sin ninguna atadura de esta índole.

Otro aspecto fundamental es que para completar los ejercicios que se proponen, se necesita motivación para no perder atención en ningún momento del transcurso de la actividad, para ello es necesario un entorno atractivo y amigable, donde el individuo se encuentre estimulado ya sea visualmente o por medio de audio, característica que facilitan la tecnología actual.

La cuestión es que cuando se desarrolla algún tipo de tecnología, como puede ser un producto software, no se suele pensar en este tipo de personas, porque representa una minoría de la población y pasa desapercibido. Esta situación conlleva a que estas personas no puedan disfrutar de las ventajas que la tecnología ofrece a todos los usuarios y les pueda ser de utilidad en su vida cotidiana, haciéndoles la vida más cómoda.

## 1.2. Objetivos

Se pretende la creación de un software que mediante el reconocimiento de gestos y el movimiento del cuerpo, las personas con discapacidad, pueda interactuar con la aplicación sin la necesidad de utilizar ningún elemento externo que dificulte de alguna manera la realización de una serie de actividades que puedan ser beneficiosas y necesarias. Entre estas actividades cabe destacar las que están relacionadas con fines artísticos, alentando el desarrollo de habilidades, como la creatividad, la concentración o simplemente el hecho de poder comunicarse, actividades de tipo cognitivo y relacionadas con el aparato locomotor del usuario.

En definitiva esta aplicación tiene la finalidad de desarrollar una serie de habilidades y capacidades que ayude a progresar y mejorar su calidad de vida.

1. Construir un software que promueva la calidad de vida de personas con

discapacidad

2. Realizar ejercicios cuya finalidad es mejorar la coordinación del individuo
3. Definir actividades que ayude en el aprendizaje
4. Estimular al usuario mediante las distintas actividades
5. Realización de actividades artísticas
6. Configurar las actividades para adaptarse a las necesidades del usuario
7. Crear actividades que faciliten la labor docente de los tutores
8. Utilizar feedback visuales y auditivos para estimular y motivar al usuario
9. Ofrecer un modo de interacción con el sistema de manera natural, haciendo uso de dispositivos de entrada lo mínimamente posible

### 1.3. Metodología

Los pasos que se han seguido en la metodología del proyecto han sido los siguientes:

En primer lugar, ha sido necesario un estudio previo de los instrumentos que se iban a utilizar para la realización del proyecto, en este caso el dispositivo Kinect, informándose y probando cómo funcionaba, para posteriormente probar las demos que venían por defecto en el SDK de Microsoft con los alumnos del CEEE Provincial Princesa Sofía. Esta última acción ha tenido la finalidad de comprobar principalmente que las limitaciones en la movilidad de su cuerpo no era un impedimento para utilizar este dispositivo e incluirlo en la educación especial. Una vez probado y superado esta cuestión, había que ver los requisitos necesarios en el sistema sobre la funcionalidad que iba a tener el sistema, qué características del sensor Kinect iban a ser necesarios y sobre qué plataforma se iba a implementar la aplicación. Además, había que definir qué actividades iban a formar el proyecto.

La interfaz de la aplicación era un aspecto muy importante dentro del proyecto, puesto que la aplicación iba a estar controlada con Kinect y era indispensable que se hicieran unos diseños que propiciaran la realización de gestos que no fuera demasiada incómoda.

A continuación se definirían las clases necesarias que compondrían el sistema, para que de esta forma fuera más rápido y eficiente el proceso de implementación. Una vez que estos pasos fueron superados con éxito, se empezó a implementar el modo de selección con Kinect y realizando esta implementación de una manera lógica. Para ello, se desarrolló en una primera instancia las actividades que eran más sencillas, obteniendo de esta forma, experiencia en la programación con Kinect y el framework XNA elegido por su uso en el ámbito de los videojuegos. De este modo la secuenciación de la implementación de los módulos fue de la siguiente manera:

1. **Módulo actividades.** Este módulo se compone principalmente de cuatro ejercicios:

- **Números.** En esta actividad el alumno tendrá que ordenar la secuencia de números, tocando los mismos en la secuencia de correcta. Al principio del ejercicio se le comunicará, cual es la finalidad del ejercicio, además cada vez que toque al número correcto, se escuchará el nombre del número correspondiente.

La actividad tiene tres niveles de dificultad, donde varía la secuencia de números que hay que ordenar.

- **Formas geométricas.** Este ejercicio consta de tres niveles de dificultad, donde varía el número de figuras que aparecen en pantalla, empezando el nivel más fácil con una sola figura. El alumno tiene que tocar la figura geométrica que el tutor ha elegido previamente en la pantalla de configuración. Cuando el niño toque la figura discriminante, se escuchará el nombre de la figura, además de desaparecer las otras figuras y aparecer ésta en el centro de la pantalla de un tamaño mayor. Se han elegido las figuras del mismo color para que el alumno no se distraiga y toque una figura porque el color pueda ser llamativo.

- **Grafomotricidad.** Se distinguen tres tipos de ejercicios.

En el ejercicio de grafomotricidad vertical, el usuario tendrá que tocar el gusano para empezar a dibujar y tendrá que hacer una línea sin salirse del circuito, de arriba hacia abajo hasta llegar a la manzana, una vez conseguido el objetivo, aparecerá una animación en 3D como recompensa.

El ejercicio de grafomotricidad horizontal es igual que el vertical, la única diferencia es, que en vez de dibujar la línea de arriba hacia abajo, tendrá que dibujarla de izquierda a derecha.

El último ejercicio es denominado grafomotricidad libre, esto es así porque el docente tendrá una pantalla de edición donde puede colocar tres figuras que le aparecerán fuera del circuito, dentro del mismo para elevar la dificultad del ejercicio de grafomotricidad horizontal. Se pueden colocar las tres figuras o las que estime oportuna el docente dependiendo del alumno que vaya a realizar la actividad, puede colocar una o dos nada más. La idea de colocar estas figuras es que sean obstáculos para llegar hasta el punto final, es decir, si el usuario tocara en cualquier momento una de las figuras colocadas por el especialista tendría que volver a repetir el ejercicio. Una última consideración a tener en cuenta, es que se puede configurar la realización de la actividad con la mano derecha o izquierda.

- **Causa-efecto.** Esta actividad consiste en aprender el concepto de apagar o encender la luz y que entiendan que sus acciones tiene una respuesta asociada. Para conseguir esta finalidad, se dibuja en la pantalla una bombilla que el usuario tiene que tocar para hacer que la luminosidad de la cámara se oscurezca, si se toca la bombilla cuando está encendida o por el contrario se ilumine la pantalla, si la bombilla se encontraba en estado de apagada.
2. **Módulo coordinación.** Este ejercicio consiste en una serie de círculos, que el experto puede colocar como quiera en la pantalla, según sus necesidades o la de su alumno, siendo el objetivo principal que el alumno vaya uniendo los círculos con líneas. Este ejercicio tiene la peculiaridad de que se puede elegir una serie de joints que no se limita sólo a las manos, como suele ser la costumbre de este tipo de actividades. Un círculo puede tener tres colores según el estado en el que se encuentre, el estado normal, se representa en color naranja, cuando un círculo pasa a ser nodo objetivo, que es simplemente el círculo que el usuario tiene que tocar en ese preciso instante, su color cambia a azul. Por último, cuando el círculo ya ha sido seleccionado, cambia a color verde.
  3. **Módulo arte.** Este módulo se divide en seis ejercicios relacionados con la música, la pintura y la danza:
    - **Música con platillos.** Esta actividad permite mover un avatar en 3D cuando la aplicación reconoce el esqueleto de la persona, imitando los movimientos del usuario. La idea de la aplicación, es que el usuario pueda tocar unos platillos, para conseguir este efecto se ha incluido un sistema de reconocimiento de gestos, el cual permite que suene el sonido de unos platillos cuando el individuo

realiza el gesto de acercar y separar las manos, como si tuviera unos platillos de verdad en sus manos.

- **Música con piano.** Esta actividad, es similar a la anterior, el usuario puede mover un avatar con su cuerpo pero esta vez, cambia el gesto reconocido, al simular que toca un piano, tendrá que levantar y bajar los brazos hacia el piano, haciendo el efecto de que en realidad está tocando uno. En el momento que la aplicación reconozca el gesto sonará el sonido de una tecla de piano, elegida al azar de entre un repertorio, para que no suene siempre el mismo sonido, a diferencia del ejercicio anterior.
- **Música conjunta.** Este ejercicio es una mezcla de los dos anteriores. La diferencia notable en este ejercicio es que se permite interactuar con dos personas al mismo tiempo, uno de ellos podrá tocar los platillos y el otro el piano. La metodología es la misma que los ejercicios anteriores.
- **Pintura.** El ejercicio se ha diseñado para que el usuario pueda pintar lo que desee en una zona de dibujo en blanco. La actividad contiene algunas opciones que el usuario puede cambiar como el color, que puede elegir entre rojo, azul o verde, el grosor del trazo que va desde fino a grueso y por supuesto puede borrar la pantalla para empezar de nuevo a pintar. Otra característica fundamental es que el usuario no tiene problema al ser zurdo o diestro, puesto que, puede elegir previamente la mano que quiere utilizar con la aplicación.
- **Danza interactiva.** Estas dos últimas actividades han sido diseñadas también para que el usuario pueda manejar un avatar en 3D a su voluntad, la diferencia de estas actividades respecto a las anteriores es que, estas van orientadas a la danza mientras que las otras estaban enfocadas al ámbito musical. En esta actividad en concreto, se trata de que el usuario se mueva libremente como si estuviera bailando y mediante una serie de gestos predefinidos en el sistema pueda interactuar con el entorno virtual. El usuario puede levantar los brazos y caerán flores desde el cielo, sin embargo si los deja en cruz, las flores se quedan congeladas en pantalla, mientras que si los baja las flores desaparecen.
- **Danza con objetos.** Esta última actividad permite al usuario poder interactuar con un balón, mediante su avatar. El avatar tendrá a su disposición un balón en sus manos, para que el usua-

rio pueda jugar con él, realizando los movimientos que desee, no obstante, esto no es lo único que puede hacer, ya que la actividad está preparada para que el usuario realice un gesto equivalente a lanzar una pelota y ésta explotará, a la vez que saldrá un sol que ilumine la escena. Para poder reiniciar la actividad y poder volver a interactuar con el balón, el individuo sólo tiene que levantar su brazo izquierdo.

4. **Módulo comunicador.** Se ha implementado el típico comunicador que se utiliza con personas con discapacidad, que no son capaces de comunicarse para que tengan un medio por el cual expresar lo que necesitan pero con gestos. Los comunicadores se basan en una serie de pictogramas que pueden representar a personas, objetos, verbos, siendo normalmente tarjetas hechas de papel o cartón. Con la revolución de los dispositivos móviles, algunos desarrolladores se han atrevido a llevar esta herramienta más allá y han implementado aplicaciones que hacen la misma funcionalidad que las tarjetas pero de forma multimedia. A pesar de todo, en esta aplicación se ha ido un paso más allá y se ha implementado este comunicador con un sistema de reconocimiento de gestos y reconocimiento de voz.

Este comunicador permite la opción al usuario de poder expresar lo que quiere mediante gestos, esto quiere decir, que cada gesto está asociado a un pictograma y el alumno sólo tiene que realizar el gesto relacionado con lo que necesita o quiere expresar. Además cuenta con el reconocimiento de voz como apoyo para el docente, por si éste viera que el alumno quiere expresar una determinada situación y no puede hacerlo mediante el gesto, tiene la posibilidad de decir la palabra asociada al pictograma, haciendo el mismo efecto que si el alumno hubiera hecho el gesto correctamente.

Se dejó el módulo de arte y el comunicador para el final, debido a que el módulo de arte se basaba en el desarrollo 3D y la incorporación de gestos. En el módulo comunicador se trabajó también con gestos, al igual que en la sección de arte pero con gestos mucho más complejos, junto con el reconocimiento de voz.

Durante toda la realización del proyecto se contó con la colaboración del CEEE Provincial Princesa Sofía, lo que permitió que cuando se finalizaba el desarrollo de una actividad se probara con los alumnos, para observar cómo funcionaba ésta y si cumplía con los requisitos establecidos en un principio.

### 1.4. Recursos

Se van a presentar los recursos tanto hardware como software que han sido necesarios para la realización del proyecto.

Herramientas de tipo hardware:

- Dispositivo Kinect XBOX 360

Herramientas de tipo software:

- Visual Studio 2010 Ultimate
- Unity 4.x
- Lenguaje de programación C#
- XNA 4.0 GameStudio
- Software Development Kit (SDK) de Kinect para Windows
- Developer Toolkit de Kinect para Windows
- Speech Software Development Kit 5.1
- Librería Coding4Fun Kinect Toolkit

### 1.5. Planificación

Las tareas que se efectuarán en el transcurso del proyecto son:

1. Análisis e indagación sobre las actividades de aprendizaje y rehabilitación física en alumnos con necesidades educativas especiales.
2. Lectura de documentación sobre cómo programar con Kinect y la tecnología XNA.
3. Especificación de requisitos, diseño e implementación del módulo de actividades.
4. Especificación de requisitos, diseño e implementación del módulo de coordinación.
5. Estudio de la herramienta Unity.
6. Especificación de requisitos, diseño e implementación del módulo de arte.

7. Especificación de requisitos, diseño e implementación del módulo del comunicador.

8. Realización de la memoria.

Tareas	Descripción	Duración
Tarea 1	Análisis e indagación sobre las actividades de aprendizaje y rehabilitación física en alumnos con necesidades educativas especiales	15 días
Tarea 2	Lectura de documentación sobre cómo programar con Kinect y la tecnología XNA	15 días
Tarea 3	Especificación de requisitos, diseño e implementación del módulo de actividades	3 meses
Tarea 4	Especificación de requisitos, diseño e implementación del módulo de coordinación	2 meses
Tarea 5	Estudio de la herramienta Unity	1 mes
Tarea 6	Especificación de requisitos, diseño e implementación del módulo de arte	3 meses
Tarea 7	Especificación de requisitos, diseño e implementación del módulo del comunicador	3 meses
Tarea 8	Realización de la memoria	1 mes y 15 días

Cuadro 1.1: Plan de trabajo

## 1.6. Cronograma

Tarea	Fecha Inicio	Fecha Final
Tarea 1	1/01/2013	15/01/2013
Tarea 2	16/01/2013	30/01/2013
Tarea 3	31/10/2013	30/04/2013
Tarea 4	01/05/2013	30/06/2013
Tarea 5	01/07/2013	31/07/2013
Tarea 6	01/08/2013	31/10/2013
Tarea 7	01/11/2013	31/01/2014
Tarea 8	01/02/2014	17/03/2014

## 1.7. Estructura del documento

La memoria del proyecto fin de carrera se compondrá de los siguientes capítulos:

- **Capítulo 1, Introducción:** En este capítulo se comprenderá mejor el problema a tratar, con un análisis del mismo y la exposición de los objetivos enmarcados en el proyecto. Para cumplir objetivos será necesario haber pensado previamente en otros apartados, como los materiales que se van a utilizar, así como una planificación aproximada del tiempo que se estima tardar en la realización del proyecto.
- **Capítulo 2, Estado del arte:** Se explica brevemente ciertos proyectos que se han realizado con una temática parecida a la de este proyecto o con puntos de interés común. Se finaliza con una descripción de las características, que diferencia a este proyecto de los investigados.
- **Capítulo 3, Tecnología:** Se presenta de una forma más extendida, las características de los medios, hardware y software, que han sido necesarios para la implementación del proyecto.
- **Capítulo 4, Aplicación KiNEEt:** Se expresa cada una de las fases de desarrollo de software (ciclo de vida de una aplicación) que han sido necesarias realizar para obtener el producto deseado. Explicación detallada desde el análisis de requisitos hasta las pruebas realizadas,

adjuntando los diagramas que han sido imprescindibles para completar cada una de las etapas con éxito.

- **Capítulo 5, Resultados:** Se muestra los datos que han sido obtenidos del cuestionario que se les ha presentado a las educadoras del centro, donde se ha estado probando la aplicación para saber su valoración con respecto al software.
- **Capítulo 6, Conclusiones y Trabajos futuros:** En esta sección se desarrolla las conclusiones obtenidas una vez finalizado el proyecto, así como posibles futuras líneas de investigación y características adicionales que se pueden añadir al software diseñado. Por último, se ha agregado los eventos y medios de difusión, donde se ha mostrado la aplicación.
- **Capítulo 7, Bibliografía:** Se exponen unas listas de referencias bibliográficas de revistas y enlaces, de donde se ha obtenido información crucial para la elaboración de la memoria.
- **Anexo A:** Este anexo refleja el manual de usuario para saber detalladamente cómo funciona la aplicación.
- **Anexo B:** En este anexo se exponen los cuestionarios que han rellenado las especialistas del CEEE Provincial Princesa Sofía en la provincia de Almería, aportando su valoración, en relación a diversos aspectos del software.

# Capítulo 2

## Estado del Arte

### 2.1. Introducción

La tecnología Kinect ha revolucionado en cierta medida nuestra mentalidad o simplemente nuestra apreciación de hasta donde se puede llegar con una herramienta si se tiene iniciativa, creatividad y un poco de imaginación. Kinect ha pasado de ser un dispositivo que se creó con fines meramente de entretenimiento para ofrecer una forma totalmente diferente, activa y cómoda, para poder jugar con la consola XBOX 360, a convertirse en un utensilio utilizado en diferentes disciplinas y ámbitos muy dispares, por ello hoy en día, se puede ver este dispositivo en lugares como quirófanos, escuelas, etc. Kinect se está utilizando en áreas relacionadas con la medicina, educación, para la discapacidad, y un sinnúmero de propósitos que se están proponiendo cada día para mejorar la calidad de vida de todas las personas y recalco el término 'todas', y poder hacer uso de esta herramienta más allá de la finalidad con la que en un principio se ideó y desarrolló.

### 2.2. Herramientas

A continuación se van a mostrar una serie de proyectos para corroborar, como asociaciones, centros hospitalarios, universidades se están involucrando para que cada día este dispositivo se vaya insertando en la sociedad, no sólo como un medio de ocio, y poder de esta forma, realizar tareas que antes eran más difíciles o simplemente no eran motivantes para los usuarios.

- **Pictogram room:** Es un conjunto de videojuegos orientados a personas que padecen autismo, en los cuales se trabajan aspectos como el autoreconocimiento, la atención conjunta, la comunicación y las relaciones con el resto de las personas, gracias a estímulos visuales y la música. El desarrollo de estos videojuegos están basados en realidad aumentada y en la interacción natural mediante el dispositivo Kinect, donde el usuario realizará actividades desde identificarse en una fotografía hasta mover la parte del cuerpo que se le está señalando, para así poder ir avanzando en las diferentes categorías que propone la aplicación[5].

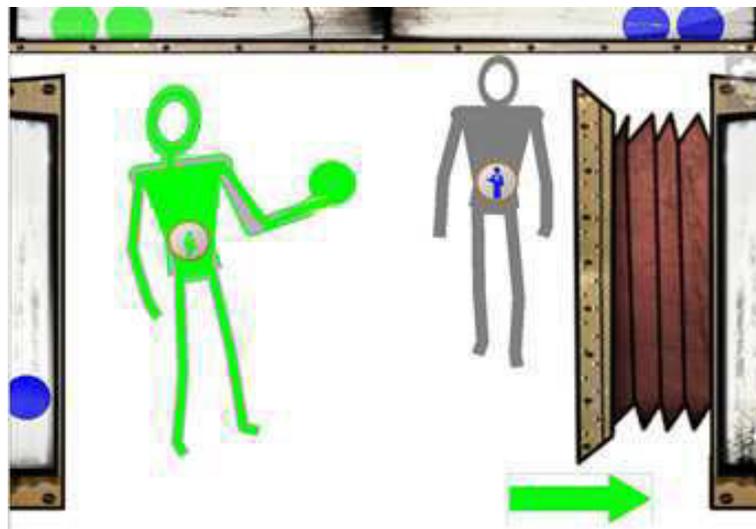


Figura 2.1: Pictogram Room. (<http://fundacionorange.es>)

- **Virtual Rehab:** Este programa está centrado en la telerehabilitación, donde el paciente interactuará con su cuerpo usando Kinect en un entorno de simulación 3D. El paciente podrá realizar los ejercicios de fisioterapia en casa, sin necesidad de desplazarse al centro, debido a que los datos se encuentran en la nube, utilizando la tecnología Microsoft Azure.

Este sistema no sólo pretende ser beneficioso para el paciente sino también para el terapeuta, debido a que el módulo denominado 'Virtual Rehab Manager', le da la opción al especialista de poder planificar, monitorizar y revisar los ejercicios de sus pacientes, así como obtener una evaluación y análisis gráfico de progresos.

Como se describe en su página web[7], esta plataforma cuenta con nueve juegos donde se trabajan diferentes aspectos de la rehabilitación física,

como pueden ser:

- Déficits motores
- Trastornos del movimiento y de la postura
- Problemas de equilibrio y falta de coordinación

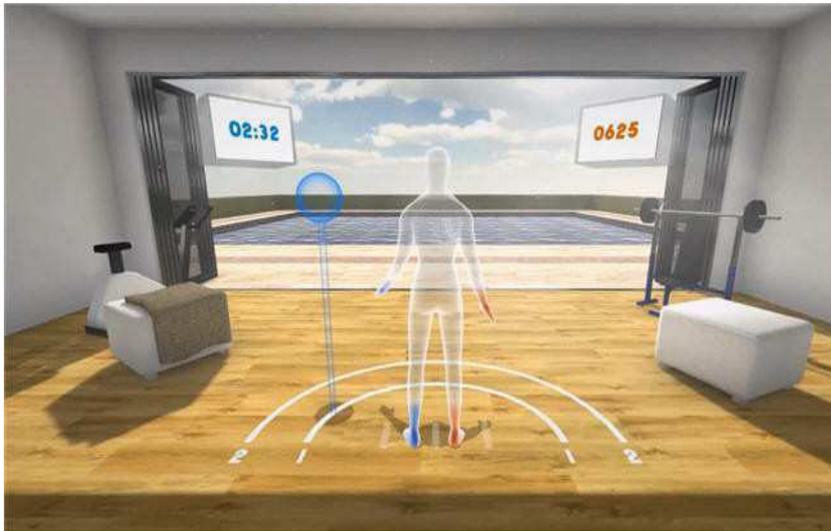


Figura 2.2: Virtual Rehab.

Imagen obtenida de <http://www.xataka.com/otros/virtualrehab-demuestra-las-posibilidades-de-la-rehabilitacion-con-kinect>

- **Proyecto TeKi:** Se trata de un sistema de teleasistencia focalizado sobre todo en los pacientes crónicos, para que puedan poseer una monitorización periódica desde casa[8] y no tener que estar desplazándose continuamente al centro de salud. Sus objetivos son mejorar la calidad de vida del paciente y evitar reingresos hospitalarios.

Para conseguir este fin se hace uso de la tecnología Kinect de Microsoft, puesto que este sistema, además de ser controlado con Kinect, permite que pueda realizar una videoconferencia con el médico, y éste puede proponer si así lo desea una tabla de ejercicios para que el paciente los pueda hacer en casa y con ayuda del dispositivo Kinect que detectará el movimiento del paciente, le informará si está realizando el ejercicio correctamente.



Figura 2.3: Proyecto Teki. ([http://www.ilitia.com/Expertise/Teleasistencia\\_basada\\_en\\_Kinect\\_Teki](http://www.ilitia.com/Expertise/Teleasistencia_basada_en_Kinect_Teki))

- **Neuro at home:** Se trata de la primera plataforma de rehabilitación virtual cuya finalidad consiste en rehabilitar a personas que tengan una lesión cerebral o que padezcan una enfermedad neurodegenerativa como puede ser esclerosis múltiples o la enfermedad de Alzheimer.

Este proyecto[9] tiene dos productos asociados, NeuroAtHome que está indicado para que el paciente puede realizar las actividades en su casa y NeuroAtHome Pro que está orientado para usarlo en los centros de rehabilitación.

Esta aplicación está formada por una serie de tareas orientadas a videojuegos, las cuales según su actividad se encuentran enmarcadas en el área de rehabilitación motora o rehabilitación cognitiva.

En la sección de rehabilitación motora se pueden realizar actividades de equilibrio, velocidad del movimiento o destreza según la necesidad del paciente.

En la sección de rehabilitación cognitiva el paciente puede hacer tareas donde estimulen los procesos cognitivos como pueden ser la memoria, la atención, las funciones ejecutivas y la comprensión.

Todas estas tareas están supervisadas por profesionales sanitarios, que le indicarán las pautas de los ejercicios que tiene que realizar el paciente según su afección y el programa, una vez terminado los ejercicios

enviará la evaluación al equipo clínico, para adaptar sucesivas tareas en relación a la evolución del paciente.

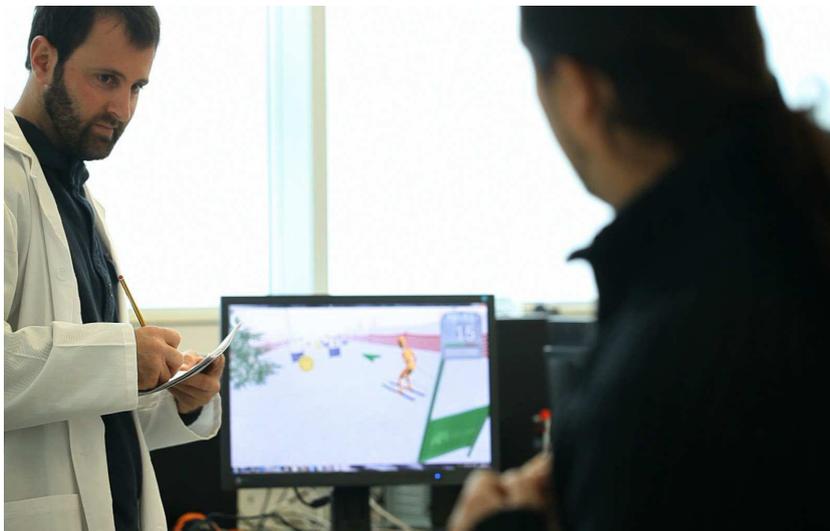


Figura 2.4: Neuro at Home. (<http://www.neuroathome.net/p/tratamientos.html>)

- **Tango:H** Es una plataforma orientada a niños que se encuentran hospitalizados, cuya finalidad consiste en proveer de unos ejercicios sobre rehabilitación física y cognitiva, que cuenta con un editor de ejercicios permitiendo de esta forma que la aplicación sea configurable y personalizable a las necesidades del paciente. Además, al ser pensada principalmente para usuarios de temprana edad, se ha enfocado desde una perspectiva de juego para así tener motivados en todo momento a los niños y puedan seguir el tratamiento de una forma idónea.

La característica principal de este software[10] es que los ejercicios pueden ser creados por los profesionales con ayuda del módulo editor que tiene la plataforma. De este modo los ejercicios son clasificados en tres tipos totalmente diferentes:

- Ejercicios de rehabilitación física donde el paciente con su propio cuerpo tendrá que realizar una serie de movimientos, reconocidos por el sensor Kinect, para alcanzar una serie de objetivos propuestos.
- Ejercicios cognitivos. En este tipo de ejercicios aunque se utilice el movimiento del cuerpo para interactuar con la aplicación, se

proponen objetivos que ayudan a desarrollar habilidades cognitivas, un ejemplo de actividad de esta índole consiste en relacionar la imagen de un animal que aparecerá en pantalla con su sonido correspondiente.

- Ejercicios de libre configuración. Estos ejercicios son los que permiten al especialista poder crear sus propios ejercicios sin estar condicionados por las dos variantes que se han visto anteriormente.



Figura 2.5: Tango:H. (<http://www.iter.es/index.php/mod.pags/mem.detalle/idpag.238/idmenu.1280/chk.9d138a35933b65c676ce37ab8c647da4.html>)

## 2.3. Aplicación KiNEEt

El desarrollo evolutivo se basa en la idea de desarrollar una implementación inicial, exponiéndola a los comentarios del usuario y refinándola a través de las diferentes versiones hasta que se desarrolla un sistema adecuado. Las actividades de especificación, desarrollo y validación se entrelazan en vez de separarse, con una rápida retroalimentación entre éstas. Existen dos tipos de desarrollo evolutivo: Desarrollo exploratorio, donde el objetivo del proceso es trabajar con el cliente para explorar sus requerimientos y entregar un sistema final. El desarrollo empieza con las partes del sistema que se comprenden

## 2.4. COMPARACIÓN ENTRE LAS DIFERENTES HERRAMIENTAS

---

mejor. El sistema evoluciona agregando nuevos atributos propuestos por el cliente.

En esta sección se va a explicar las características fundamentales que tiene la aplicación que se ha desarrollado para este proyecto, denominada KiNEEt. La primera característica que se merece mencionar es que en una misma aplicación se puede encontrar actividades de diferente tipo, mezclando ejercicios que están orientados puramente al campo de la motricidad, como el ejercicio de coordinación, con actividades puramente artísticas, donde se pueden desarrollar habilidades que no se pueden encontrar en actividades motoras o cognitivas como puede ser la creatividad, a la vez que resulta una actividad de ocio y totalmente libre para el usuario, ya que no tiene ningún patrón característico que suponga una atadura. Esta característica se puede apreciar por ejemplo en la actividad de pintar, donde el usuario pinta lo que quiere, según se sienta en ese momento o lo que quiera expresar, lo mismo pasa con la danza, no se tiene que realizar un movimiento concreto para pasar a una siguiente fase o cumplir una serie de objetivos, el individuo hace el movimiento que le apetezca, puesto que el objetivo principal es promover la expresión corporal.

Un aspecto a destacar de esta aplicación es la implementación de un comunicador, que a través del reconocimiento de gestos, el usuario permita comunicar sus necesidades inmediatas. La forma normal de realizar esta tarea es mediante tarjetas que el experto le va enseñando al niño y éste, elige la que considera adecuada según lo que quiere expresar. Con el revolucionario mundo que están abriendo los dispositivos móviles, ahora se pueden ver también comunicadores desarrollados para tablets tanto en Android como en iOS, aun así, para interactuar con estos comunicadores es necesario tocar la pantalla para elegir el pictograma adecuado, incluso arrastrarlo a una determinada zona para completar la acción, con lo cual, poder hacer esto mismo mediante gestos abre nuevas posibilidades para estas personas que quieren comunicarse con los demás pero sus cualidades no se lo permiten.

## 2.4. Comparación entre las diferentes herramientas

	Virtual Rehab	Pictogram Room	Proyecto Teki	Neuro at Home	Tango : H	KiNEEt
Desarrollo de habilidades físicas	SI	NO	SI	SI	SI	SI
Desarrollo de habilidades cognitivas	NO	SI	NO	NO	SI	SI
Desarrollo de habilidades artísticas	NO	NO	NO	NO	NO	SI
Personalización de actividades	NO	NO	SI	SI	SI	
NO						
Uso de avatares 3D	SI	NO	NO	NO	NO	SI
Reconocimiento de gestos	NO	NO	NO	NO	NO	SI
Reconocimiento de voz	NO	NO	NO	NO	NO	SI
2 jugadores	NO	NO	NO	NO	SI	SI
Control interfaz con Kinect	NO	NO	SI	NO	SI	SI

Cuadro 2.1: Tabla comparativa de las diferentes herramientas

**El desarrollo de habilidades físicas**, hace referencia a que la aplicación contiene algún ejercicio que está centrado mejorar la parte física del cuerpo. Todas las actividades menos una, poseen esta cualidad, esto también es debido a que el dispositivo Kinect se basa en el movimiento, con lo cual lo primero que se le puede ocurrir a una persona es utilizarlo con fines rehabilitadores, como sucede con estas aplicaciones. La aplicación KiNEEt, se centra en mejorar las habilidades físicas pero más bien desde un punto de vista de la coordinación, permitiendo poder elegir entre diferentes articulaciones, para que así el movimiento se pueda ajustar mejor a las necesidades del usuario.

**El desarrollo de habilidades cognitivas**, hace referencia a las actividades de la aplicación que permiten realizar procesos cognitivos, como pueden ser la memoria, asociación o causa-efecto, entre otras, es decir, ejercicios que permitan estimular el cerebro. De las actividades analizadas, la mitad se han dedicado a incluir este tipo de actividades, como se ha dicho en el apartado anterior, cuando se usa el dispositivo Kinect, se tiende a pensar más bien en actividades físicas y no tanto en ejercitar la mente.

**El desarrollo de habilidades artísticas**, hace referencia a que la aplicación tiene alguna actividad que fomenta mejora de cualidades de tipo artístico, como es la creatividad, que también son fundamentales para el desarrollo del alumno. Ninguna de las aplicaciones presentadas en este estudio reflejan que tengan ninguna actividad de esta índole, la aplicación KiNEEt, es la única que contiene actividades artísticas como música, danza y pintura, para fomentar este tipo de habilidades.

**Personalización de actividades.** Este apartado se refiere a que alguna de las actividades no sea cerrada, y sea posible modificar alguno de los aspectos de la actividad, creando así una actividad distinta a lo está hecho por defecto. En este proyecto hay unas actividades que permite editar previamente la localización de ciertos nodos e imágenes, lo que provoca que la aplicación se tenga que realizar de una manera distinta al hecho de realizarla con la configuración por defecto, dejando al especialista total libertad para que modifique la actividad a su gusto y necesidades.

**Uso de avatares 3D.** Sólo hay una aplicación más, además de este proyecto que utilice avatares 3D para representar al individuo y que éste disfrute de un entorno virtual donde el avatar es capaz de imitar cada uno de los movimientos que realiza el usuario.

**Reconocimiento de gestos.** Ninguna de las otras aplicaciones ha añadido a sus actividades la posibilidad de reconocer gestos para cumplir objetivos específicos. En KiNEEt se ha hecho que sea posible reconocer determinados

gestos realizados por el usuario, asociando cada gesto a un pictograma completo, con la finalidad de crear un comunicador y permitir a los alumnos que no pueden usar el lenguaje, poder transmitir las peticiones que necesita, de una manera natural.

**Reconocimiento de voz.** Se ha valorado en este apartado que se pueda realizar un reconocimiento de palabras en la aplicación, siendo el resultado que KiNEEt es el único que permite reconocer ciertas palabras, en el módulo de comunicador, con el fin de que el especialista pueda seleccionar el pictograma que desee, controlando de esta forma la actividad, con el reconocimiento de una gramática, asociada a los distintos pictogramas que se han integrado en este ejercicio denominado comunicador.

**2 jugadores.** Este aspecto refleja si existe alguna actividad en el que sea posible interactuar con más de un jugador. El dispositivo Kinect permite sólo realizar el seguimiento de dos esqueletos, lo que permite poder interactuar como máximo con dos jugadores, esto hace que se puedan incluir actividades que sean cooperativas. La aplicación Tango:H permite poder seleccionar entre uno o dos jugadores en sus actividades y aunque KiNEEt, se centra en un entrenamiento más personalizado, permite que alguna actividad se pueda realizar con dos individuos, para así fomentar también las habilidades sociales y no sea tan individualizado.

**Control interfaz con Kinect.** A pesar de usar en todas las actividades con el sensor Kinect, está extendido más el uso de que se pueda manejar la interfaz usando el ratón, relegando este dispositivo, sólo para usarlo en las distintas actividades que se han desarrollado para la aplicación, sin embargo hay actividades como Tango:H o KiNEEt que sí han pensando en sacarle el máximo partido al dispositivo que están usando y poseerle funciones de control de la interfaz de la aplicación, para navegar por ella, mediante el movimiento del usuario.

# Capítulo 3

## Tecnología

Para la realización de este proyecto, al estar orientado a personas con discapacidad se ha utilizado como eje central del mismo el dispositivo de la compañía Microsoft, que ellos mismos han denominado como Kinect. El dispositivo Kinect va a ser el elemento más importante a nivel de hardware para poder cumplir las expectativas de este proyecto, gracias a su infinidad de sensores lo convierten en una herramienta muy útil. Sin embargo la aplicación con la que va a interactuar el usuario es de índole software y al estar enfocado especialmente para niños, se ha intentado que no sea una aplicación seria y aburrida, sino una aplicación amena con la que los niños puedan disfrutar realizando cada una de las actividades que la componen. Por estos motivos se ha elegido XNA 4.0 y Unity 4.x, que están orientadas para el desarrollo de videojuegos, como las opciones idóneas para desarrollar este tipo de aplicación.

### 3.1. Kinect

El sensor Kinect es un dispositivo que fue desarrollado para la consola de la compañía Microsoft, XBOX 360, lógicamente con fines de entretenimiento. Kinect era un dispositivo de entrada como podía ser los propios mandos de esta consola pero que ofrecía una característica que lo diferenciaba del resto: la libertad del movimiento. Mientras que usando el sensor Kinect puedes interactuar con los videojuegos, únicamente con el movimiento y los gestos que puedes realizar con tu propio cuerpo, en otros dispositivos como Wiimote[11], necesitas estar sujetando este mando todo el tiempo, con lo cual, aparte de ser más incómodo, también limita mucho la interacción con el juego. Por

estos motivos Kinect destacó del resto de competidores, pero no sólo en el ámbito de videojuegos, sino que se extendió a otras áreas de conocimiento, algunas de ellas son[12]:

- Sanidad
- Rehabilitación
- Educación
- Militar
- Sistemas de seguridad
- Realidad virtual
- Robótica

En el siguiente apartado se va a explicar cada uno de los componentes que forman este dispositivo y lo hace tan característico.

### 3.1.1. Componentes

**Cámara RGB** Esta cámara (Ver 3.1) es la que nos permite tanto poder grabar vídeo con Kinect, como usar la imagen de vídeo que está reproduciendo este dispositivo en nuestra aplicación. Esta cámara se encarga de crear un stream de vídeo[13], que serán los datos que realmente usemos. Sin embargo, este stream de vídeo no es resuelto con una resolución cualquiera, sino que ya están definidas las resoluciones y los frames, que graba esta cámara. Estas resoluciones son:

- 640x480 a 30 frames por segundo
- 1280x960 a 12 frames por segundo

**Sensor de profundidad** Este sensor está compuesto por un emisor de infrarrojos y un sensor de infrarrojos (Ver 3.1), que funcionan de la siguiente forma:

El emisor de infrarrojos se encarga de proyectar una malla de puntos en su radio de actuación, impregnando a todos los elementos que se encuentran a su alcance con una infinidad de diminutos puntos, que no tenemos la capacidad de ver. Sin embargo, el sensor de infrarrojos sí es capaz de interpretarlos y gracias al reflejo que desprenden estos puntos, este sensor los detecta y transforma en información necesaria para saber a qué distancia se encuentra el objeto punteado del dispositivo Kinect[12].

**Array de micrófonos** Esta serie de micrófonos (Ver 3.1) están repartidos a lo largo del dispositivo con la finalidad de averiguar de donde proviene el sonido. No obstante, esta serie de tres micrófonos, no tienen sólo esa finalidad, sino que si se saben utilizar adecuadamente, permite grabar sonido, al igual que utilizar una de las funcionalidades más atractivas de este sensor, el reconocimiento de voz[3].

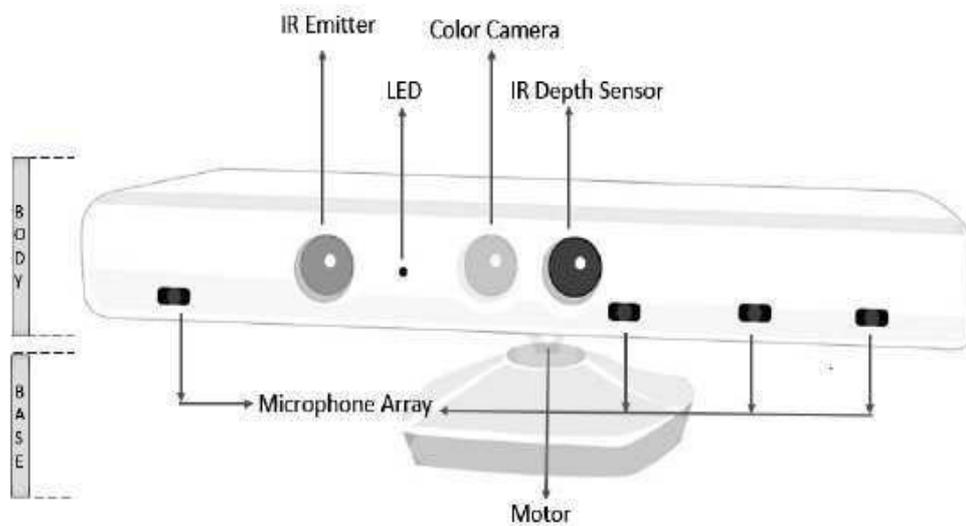


Figura 3.1: Componentes Kinect. (Abhijit, J.2012)

### 3.1.2. Campo de actuación

Un aspecto fundamental que es necesario conocer si se quiere utilizar Kinect, es saber en que área permite reconocer al usuario, puesto que si el individuo se sitúa más cerca o más lejos de las consideraciones recomendadas, el dispositivo no será capaz de reconocer a la persona.

Se recomienda para obtener unos resultados óptimos que el usuario se encuentre entre 1.2 metros y 4 metros[13] de distancia respecto al sensor.

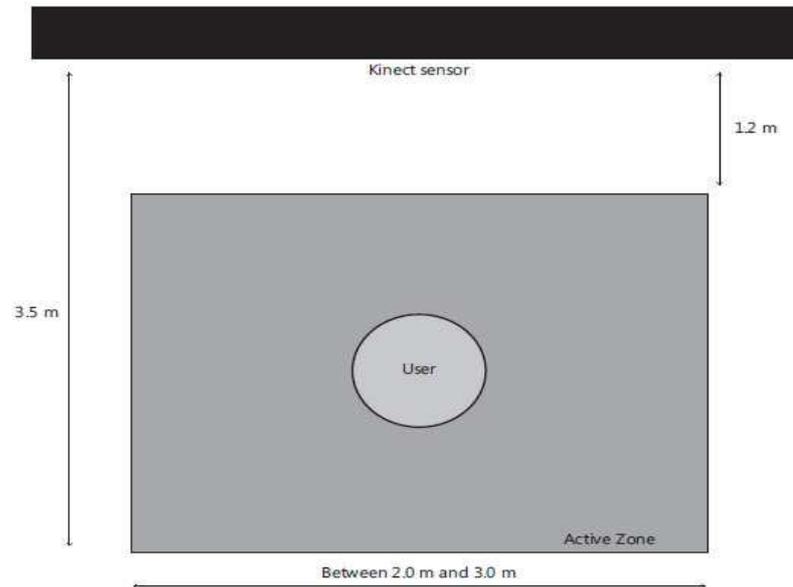


Figura 3.2: Zona de actuación Kinect. (Catuhe, D.2012)

Existe un modo particular llamado 'Near mode', el cual permite que estas distancias sean inferiores, es decir, es posible reconocer a una persona en una distancia inferior a la recomendada. Con este modo activado sería posible reconocer a una persona que se encontrara a una distancia de 40 centímetros hasta 3 metros.

### 3.1.3. Kinect SDK

El SDK desarrollado para Kinect, es el software que nos va a permitir que nuestro ordenador pueda comunicarse con este dispositivo, además de que será indispensable si queremos trabajar con los streams de datos que nos proporciona Kinect, a través de sus variados sensores. Este SDK tiene implantada la opción de utilizar el seguimiento del esqueleto (skeletal tracking)[17], ya que es capaz de identificar hasta 20 'joints' del cuerpo humano. Esta característica va a ser la base de este proyecto, ya que va a facilitar la labor de saber qué gestos está realizando el usuario y así valerse de los movimientos del cuerpo para poder manejar la aplicación.

Existen otras librerías que también son capaces de utilizar la información que proporciona los distintos sensores del sensor, el más conocido es OpenNI, a continuación se muestra una tabla comparativa de ambas librerías [18] [19]:

## KiNEEt: Aplicación para el aprendizaje y rehabilitación en educación especial

Características	Microsoft SDK	OpenNI
Plataformas	Windows	Windows, Linux y Mac OS
Lenguajes	C++ y C#	Python, C, C++, Java y Lisp
Skeletal Tracking	SI	SI
Calibración	NO	SI
Seguimiento predictivo	SI	NO
Seated Mode	SI	NO
Near Mode	SI	NO
Joints	20	20
Audio	SI	NO
Cancelación de ruido	SI	NO
Reconocimiento de voz	SI	NO

Cuadro 3.1: Tabla Comparativa Microsoft SDK y OpenNI

Con las características expuestas en la tabla anterior, se van a proponer las razones de por qué se ha elegido Microsoft SDK como la API predilecta para incorporar al proyecto.

Aunque ambos disponen de la característica 'skeletal tracking', en Microsoft SDK no necesita calibración, esto quiere decir que reconoce el esqueleto humano, con el simple hecho de situarse delante de la cámara, mientras que en OpenNI, es necesario realizar la postura de 'T' para poder reconocer al individuo. Para una persona con discapacidad y acentuando el hecho de que lo normal es, que los usuarios que van a usar la aplicación sean jóvenes, es más cómodo para ellos que no estén sujetos a tener que realizar una postura determinada para poder utilizar el dispositivo, sino que colocándose en una posición normal y natural, el dispositivo pueda reconocer a la persona.

Microsoft SDK también dispone de un modo de tracking denominado Seated mode, este modo permite al sensor poder reconocer a la persona aunque se encuentre sentada, puesto que obvia los joints que se encuentran en la parte inferior del cuerpo y se queda sólo con los que están situados en la parte superior del tronco. Este modo ofrece la ventaja de que si la aplicación la utiliza una persona que se encuentra en sillas de ruedas, gracias a esta característica la aplicación puede funcionar sin ningún tipo de problema, ya que puede seguir reconociendo perfectamente a la persona, a pesar de encontrarse en la posición de sentado.

El modo Near mode, como se ha hecho referencia en el apartado anterior, nos

da la posibilidad de que aunque nos situemos a una distancia del sensor más cercana a la recomendada, nos puede seguir detectando ya que si para el modo estándar, Kinect puede reconocer a una persona que se encuentre mínimo a 80 centímetros de ella, con esta característica es capaz de detectar a usuarios que se encuentren hasta 40 centímetros. Esta cualidad viene muy bien para este tipo de proyecto, porque al trabajar con niños, es difícil en algunas circunstancias que se queden a la distancia que se les ha dicho previamente o simplemente que se les olvide, una vez están interactuando con la aplicación y no respeten la distancia, así que con este modo aunque se acercara no habría ningún problema en ese sentido y la aplicación podría seguir funcionando con normalidad.

Por último, añadir que OpenNI no permite realizar reconocimiento de voz, característica que es imprescindible para este trabajo, debido a que el reconocimiento de voz servirá como medio de apoyo para docentes y especialistas cuando ellos quieran realizar una acción externa en algunas partes de la aplicación. Para ellos, es más cómodo o esencial que puedan dar órdenes como por ejemplo cuando se quiera finalizar una actividad, siendo posible realizar esta acción con una orden hablada, en vez de utilizar gestos.

## 3.2. XNA

Es un framework perteneciente a la plataforma .NET, orientado a la creación de videojuegos para XBOX 360, Windows Phone 7 y Windows. Últimamente ha sido muy utilizado por numerosos desarrolladores para crear videojuegos que incorporen la tecnología Kinect a su modo de juego.



Figura 3.3: Logo XNA

### 3.2.1. Ciclo de vida de un juego

En este apartado se va a exponer la estructura básica del ciclo de vida de un videojuego que ha sido desarrollado con XNA[20].

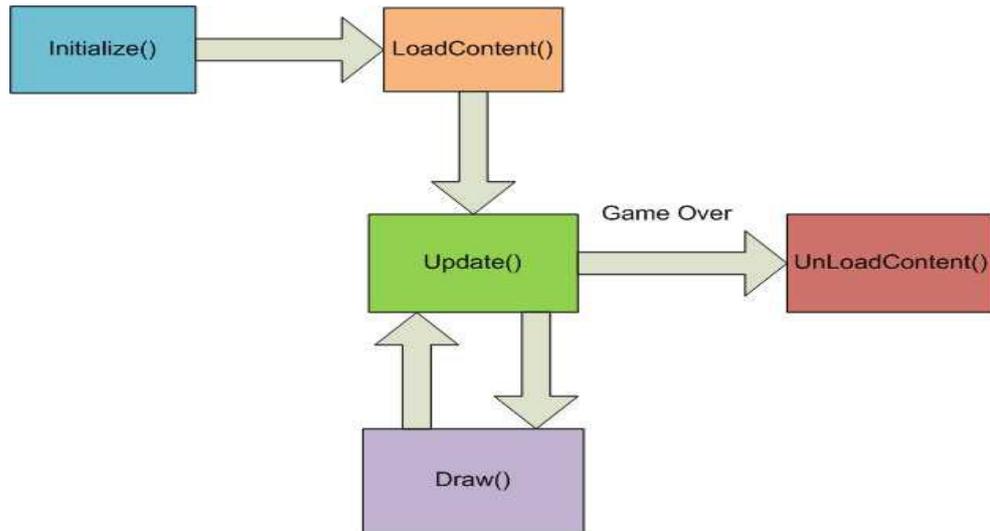


Figura 3.4: Ciclo de vida de un videojuego XNA

#### Initialize()

Este método será el primero al que se llame, en el ciclo de vida del juego y es usado para inicializar tanto las variables como otros componentes del juego.

#### LoadContent()

Este método se llamará justamente después de haber acabado con el método de inicialización y se encargará sobre todo de cargar los componentes gráficos, como pueden ser las texturas, vídeos, modelos 3D, etc. Sin embargo, en este método no sólo se cargarán componentes propios de un videojuego como son componentes gráficos o multimedia, sino que aquí también se cargará cualquier componente externo que necesite el juego, como por ejemplo archivos XML.

#### Game Loop

Cuando se desarrolla un videojuego, esté escrito en XNA o no, el concepto de *game loop* siempre está presente. El *game loop* consiste en la secuencia de llamar al método `update()` y `draw()` todo el tiempo que el programa esté en ejecución. El motivo de seguir este procedimiento es muy sencillo, sólo hay

que pensar que en un videojuego es necesario estar realizando cálculos constantemente (colisiones, cambios de trayectoria, IA,...), los cuales se realizan en el método `update()` y por lógica estos cálculos producirán un cambio en el aspecto visual que de algún modo habrá que reflejar, de este aspecto se encarga el método `draw()`.

#### *Update()*

Este método es el eje central de este esquema, puesto que es el encargado de realizar la lógica del programa. Esta lógica estará integrada por movimiento de personajes, colisiones entre objetos o personajes, cambios en los indicadores de salud o puntuación,..., lo que conlleva a que esté continuamente ejecutándose para poder realizar estos cálculos y comprobar las posibles actualizaciones de los componentes en el transcurso del juego.

#### *Draw()*

Este puede ser el método que genere menos controversia, ya que sin explicación, cualquier persona es capaz de intuir la función del mismo. Este método se va a encargar dentro del ciclo de vida de pintar en pantalla los objetos que sean necesarios que el usuario aprecie visualmente, las imágenes, el color del fondo, los modelos. Será el método que haga posible que los elementos gráficos que hemos cargado en el método `LoadContent()` puedan verse en escena.

#### **UnloadContent()**

Esta es la última llamada que se realiza en el ciclo de vida del juego. Una vez que el game loop ha terminado, situación que sucede cuando la clase `Game` invoca al método `Exit()`, el método `UnloadContent` se pone en funcionamiento para liberar el contenido que ha sido cargado en el método `LoadContent`. Este método brinda la oportunidad, de que si fuera necesario liberar la memoria de un objeto, que necesita de un tratamiento específico para realizar esta acción, se puede ejecutar aquí, sin ningún problema.

### **3.2.2. La clase `DrawableGameComponent`**

Esta clase dota a sus hijos de sus propios métodos `initialize()`, `update()` y `draw()`[21], dichos métodos se ejecutarán cuando se ejecuten los métodos `update()` y `draw()` propios de la clase principal independientemente, realizando así su funcionalidad propia. Además, para añadir el elemento sólo hay que llamar al método `Add()` de la clase `Components`, es decir, con una línea de código se añade el elemento al juego, así de fácil es obtener cuán números de

### 3.3. UNITY

---

elementos sean necesarios para el programa, sin tener que estar escribiendo infinidad de líneas de código para la creación de todos los componentes.

## 3.3. Unity

Unity es lo que se denomina un motor de juego, una herramienta con la que es posible crear un videojuego, pero no te permite crear todos los elementos necesarios para la creación de un videojuego. Un videojuego está compuesto por modelos, sonidos, vídeos, texturas, etc, contenidos que no te permite crear este entorno de trabajo[22] porque no está orientado, ni al ámbito de diseño gráfico, ni de audio, ni ninguno por el estilo. Sin embargo Unity lo que te permite es unir todos estos elementos, que dispongan de un escenario común y que cada uno según un argumento en común, pueda representar su papel en la escena mediante la programación, que sí se puede desarrollar con este entorno.



Figura 3.5: Logo Unity

#### 3.3.1. Assets

Los assets son los recursos que se utilizarán en todos los proyectos de Unity, desde las texturas de imagen, archivos de sonido, modelos 3D, etc. Estos ficheros están contenidos en la carpeta Asset del proyecto y se pueden crear pulsando sobre el menú Assets del entorno de trabajo, mostrando todas las posibilidades que nos ofrece la herramienta.

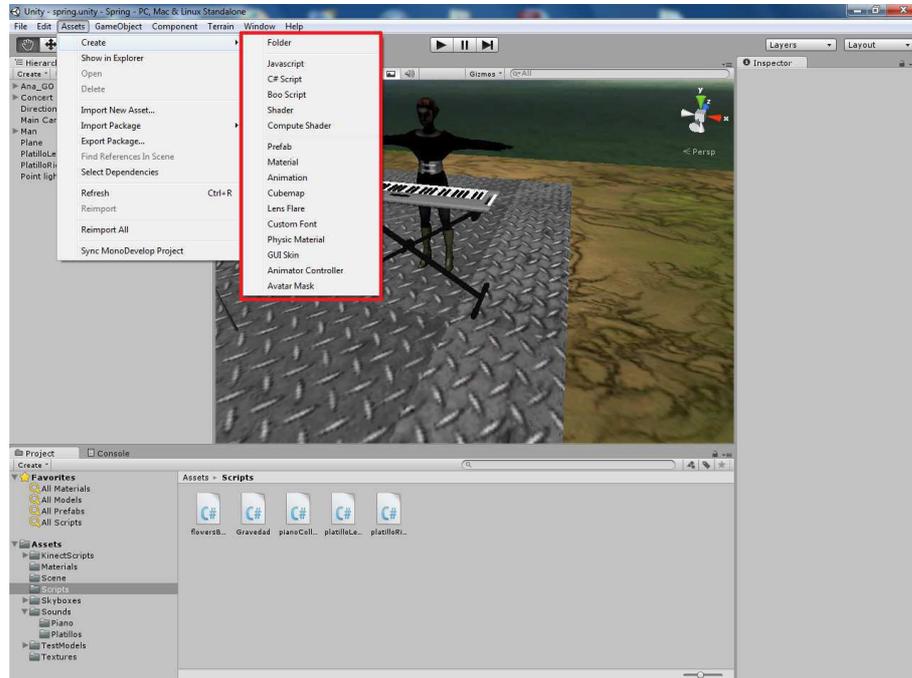


Figura 3.6: Menú Assets

### 3.3.2. Escena

La escena va a ser el componente donde se van a colocar los assets que posee el proyecto, será como el mundo donde se desarrolla toda la acción del juego. Se pueden tener varias escenas en un mismo proyecto y dependiendo de la embergadura del mismo, será lo más conveniente, ya que estas escenas representan a lo que comúnmente se le denomina niveles. Hay que tener en cuenta que en esta herramienta no se pueden abrir simultáneamente dos escenas.

## KiNEEt: Aplicación para el aprendizaje y rehabilitación en educación especial

---

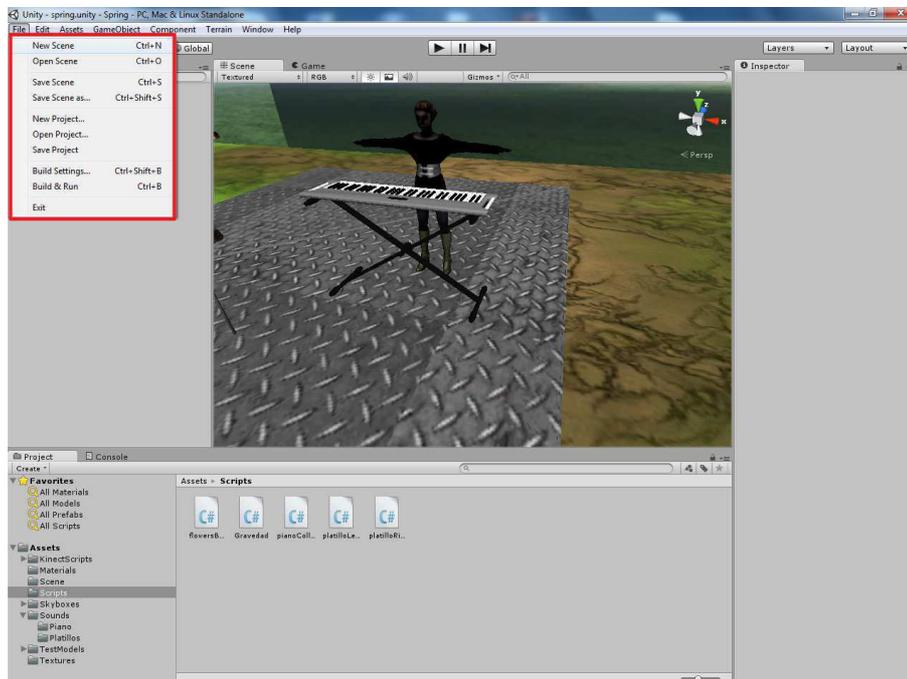


Figura 3.7: Crear nueva escena

### 3.3.3. Game Object

Cualquier objeto activo en la escena actual se denomina *Game Object*, si se sitúa cualquier asset de los mencionados anteriormente en la escena, éste será calificado como Game Object.

Todos los objects tienen el componente transform, que permite cambiar las propiedades de posición, rotación y escala. Para crear un Game Object, se puede pulsar con el botón derecho del ratón en el panel Hierarchy

## KiNEEt: Aplicación para el aprendizaje y rehabilitación en educación especial

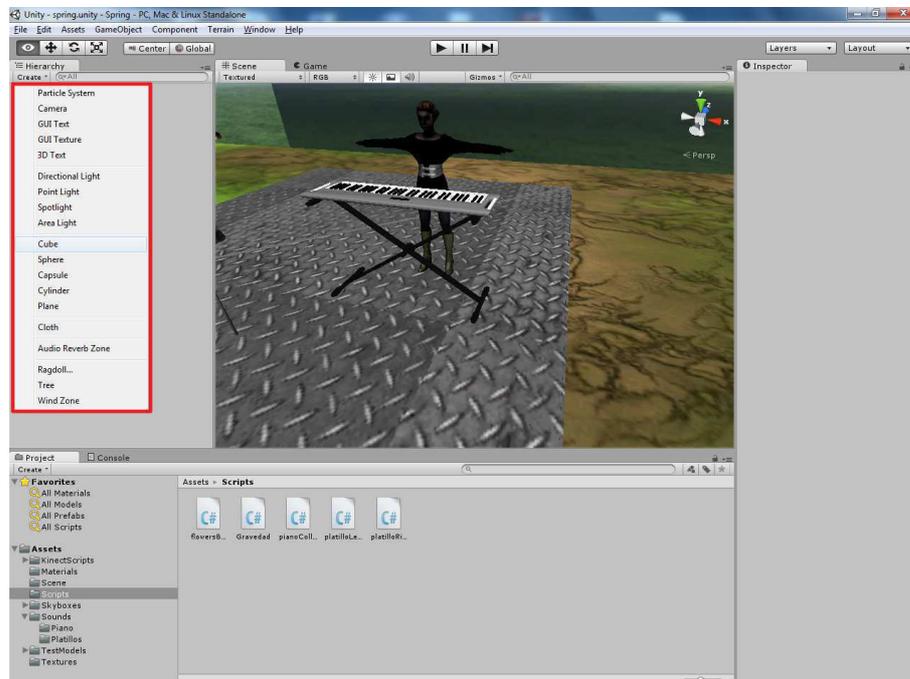


Figura 3.8: Crear Game Object desde el panel Hierarchy

o directamente en el menú GameObject → Create Ohter

## KiNEEt: Aplicación para el aprendizaje y rehabilitación en educación especial

---

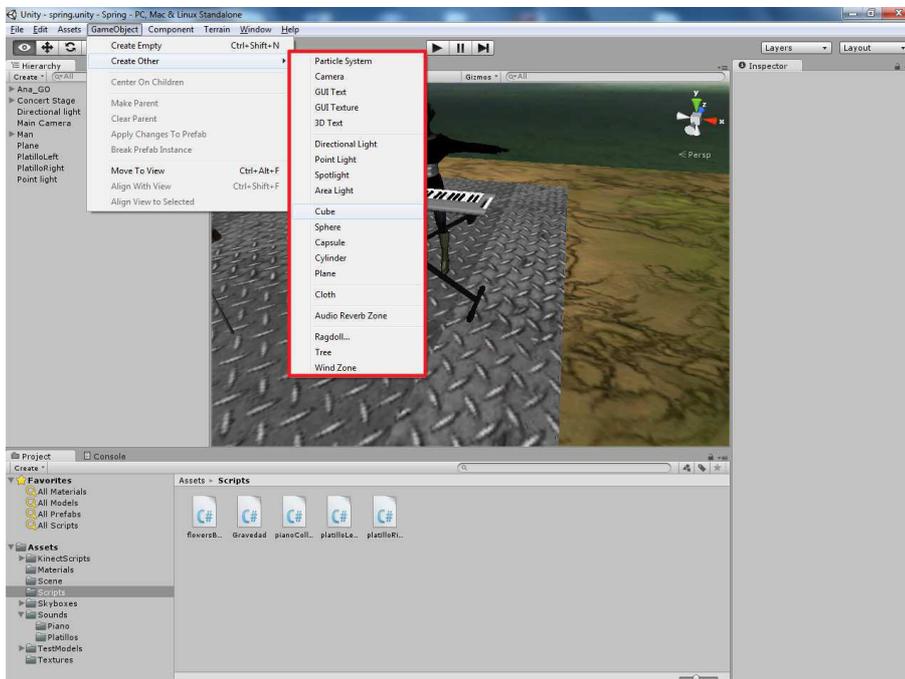


Figura 3.9: Crear Game Object desde el menú GameObject

### 3.3.4. Prefabs

Este elemento se puede considerar como una plantilla, que permite la creación de objetos complejos con múltiples características, que pueden ser reutilizados e instanciados tantas veces como se necesite.

### 3.3.5. Scripts

Uno de los componentes esenciales en Unity son los scripts. Se profundizará más en detalle en la sección orientada a programación en Unity.

### 3.3.6. Descubriendo la interface

La interface de Unity tiene una disposición por defecto de cada uno de sus paneles, si no le resulta cómoda esta distribución, se puede cambiar accediendo a la opción de Layout y bien ya sea eligiendo otra configuración que

viene con el programa por defecto o configurando la suya propia tendrá su interface lista para trabajar.

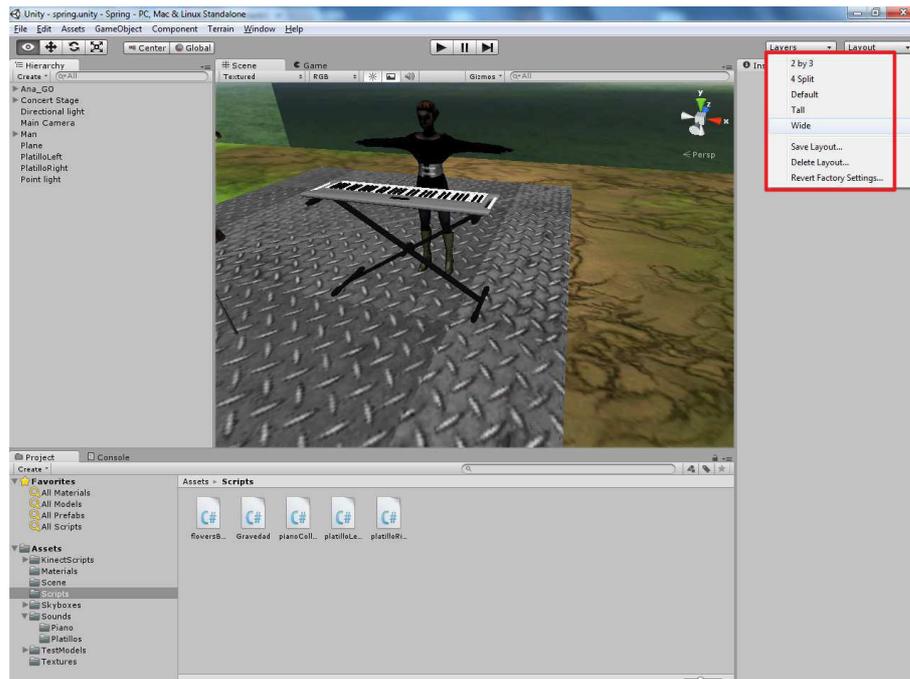


Figura 3.10: Tipos de layout

A continuación se explicará brevemente cada uno de los paneles que conforman la interface de Unity.

### Panel de Jerarquía

En este panel se puede acceder a cada uno de los game objects que se encuentran en la escena, permitiendo un acceso tanto de los elementos principales, como de los subordinados

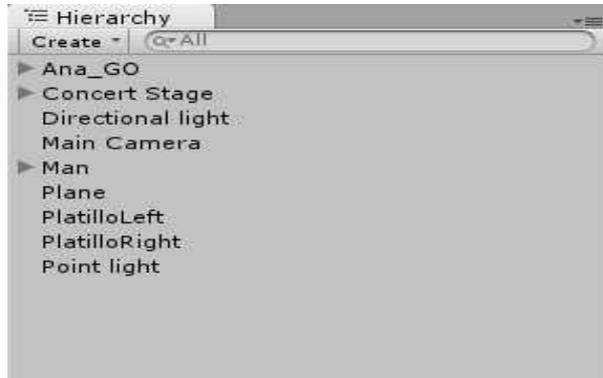


Figura 3.11: Panel de Jerarquía

### Panel de Proyecto

Este panel permite acceder a cada uno de los recursos que se están utilizando en el desarrollo del juego.

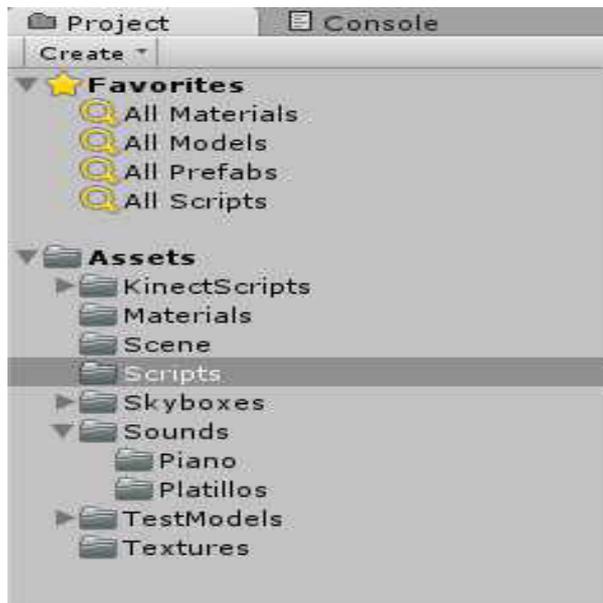


Figura 3.12: Panel de Proyecto

### Panel de Consola

Es donde se va a ir mostrando cada uno de los errores que van apareciendo en la creación del juego. Los mensajes de debug que se utilizarán para depurar la aplicación también aparecerán en este mismo panel.

**Panel Inspector** Este panel muestra los componentes que están asociados a un game object, tales como el transform, los scripts vinculados a éste o incluso los materiales de los que está compuesto.

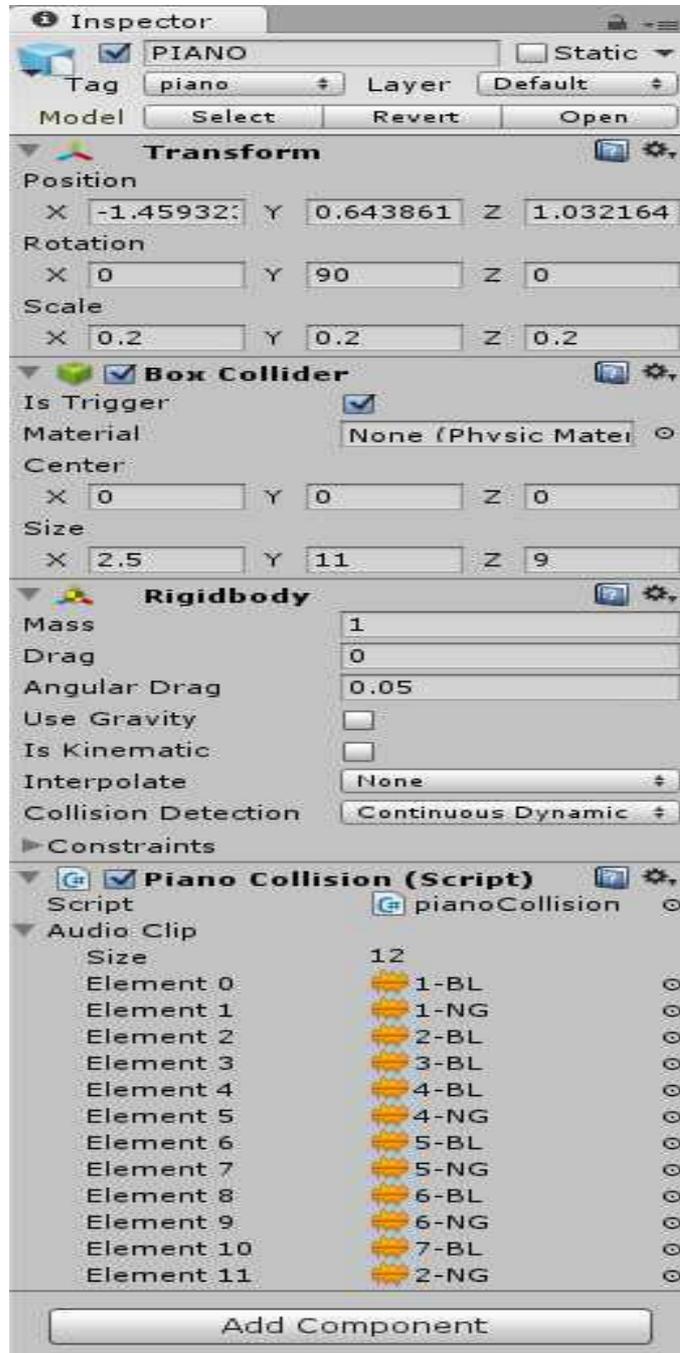


Figura 3.13: Panel Inspector

**Panel de Escena** En este panel se puede visualizar y componer la escena que se tenga seleccionada actualmente.

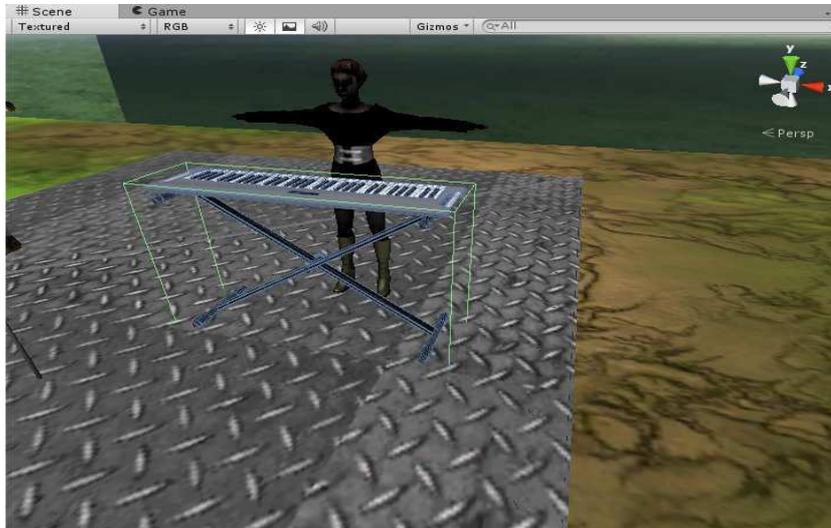


Figura 3.14: Panel de Escena

**Panel de Juego** Este panel ofrecerá previsualizar el juego para ver el resultado final, a través de la vista de la cámara principal.

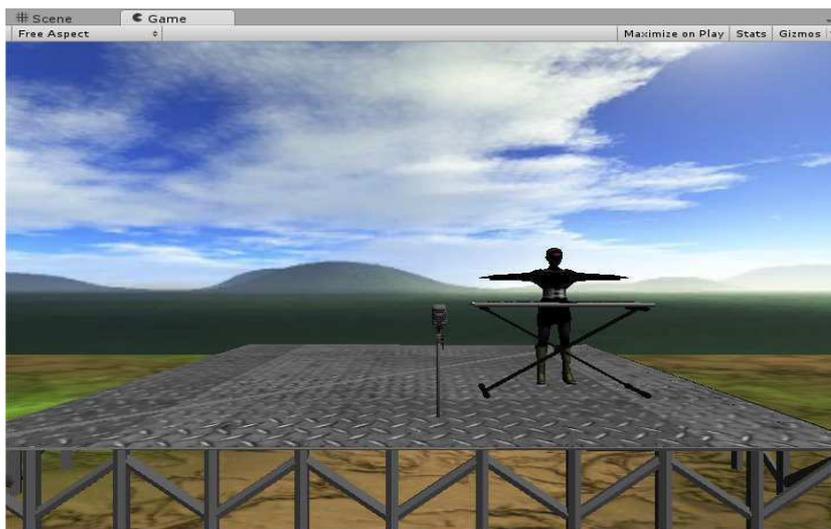


Figura 3.15: Panel de Juego

### 3.3.7. Programación con scripts

En primer lugar, como la programación en Unity se basa en scripts es necesario una breve explicación sobre qué es un script. Un script[24] se puede definir como un conjunto de instrucciones e información, que proporciona cierta funcionalidad a un objeto y permite al motor de juego saber cómo y cuándo necesita hacer uso de esa información. Para crear un script de C#, nos dirigimos al panel de proyectos, Create → C# Script.

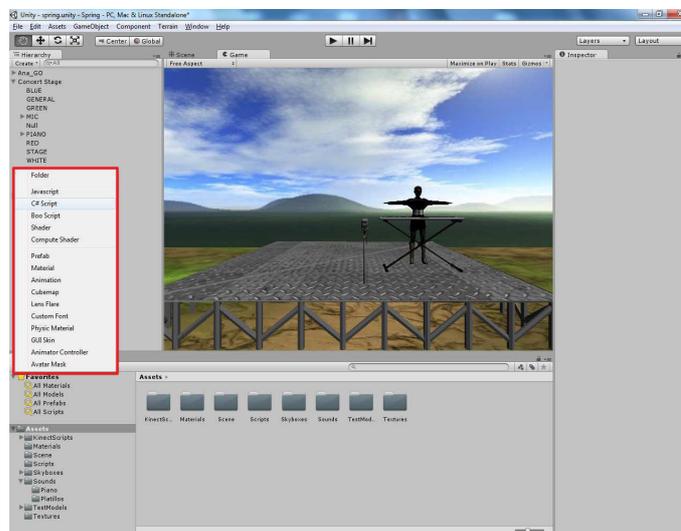


Figura 3.16: Creación de un script en C#

En la siguiente imagen podemos ver el resultado de nuestra acción.

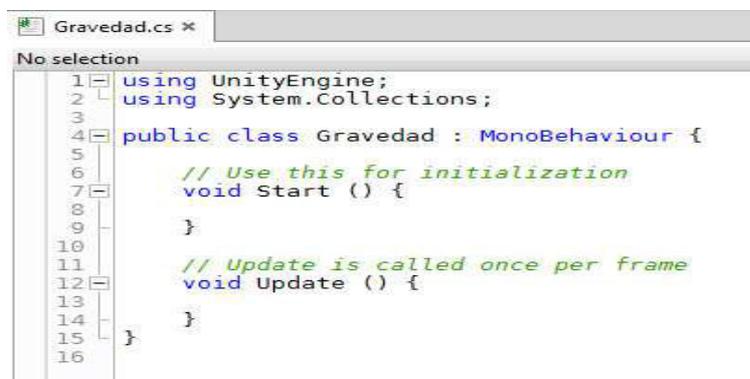


Figura 3.17: Script vacío en C#

Estos scripts están compuestos fundamentalmente por métodos y variables [23].

*Variables de script C#* Las variable, al igual que en cualquier programa escrito con C# se utilizan para almacenar datos. Para declarar estas variables será necesario definir el ámbito, el tipo de dato (C# es fuertemente tipado) y el nombre de la variable. La diferencia fundamental entre crear una variable en Unity y en otro entorno, es que si se crea en un ámbito público con la palabra reservada *public*, podremos acceder a ella desde la interfaz del programa y modificar su valor.

Esta característica facilita mucho la labor del desarrollador cuando se tiene un número considerable de objetos que tienen añadidos el mismo script y se requiere que cada uno tenga un valor distinto de la misma variable, ya que se puede modificar directamente desde la interfaz, sin tener que acceder al script.

*Métodos de script C#* Cuando se crea un script en C# como el de la 3.17, por defecto se crean estos métodos. En el método `Start()`[22] se pueden inicializar las variables que han sido creadas una vez que se haya cargado el juego. El método `Update()` es el método más importante y se encargará de ir comprobando continuamente los cambios que van sucediendo en el objeto, para así proporcionarl la respuesta adecuada a los eventos que van apareciendo en el transcurso del juego. Mientras que el método `Start()` sólo se ejecuta una vez, el método `update` se ejecutará varias veces cada segundo.

Como se ha apreciado, la programación cuando se desarrolla un videojuego es distinta a la de otra aplicación, en este caso, para dotar de funcionalidad a nuestra creación, será necesario crear uno o varios scripts en los game objects que requieran un comportamiento específico o puedan modificar incluso propiedades de otros componentes que se encuentren en escena. En definitiva un script va a a ser el elemento que proporcione el comportamiento a cada uno de los games objects presentes en el juego. Estos scripts aceptan el lenguaje Javascript, C# o Boo.

### **3.4. Principios recomendados entre TIC y discapacidad**

Es una realidad que las TIC han contribuido a mejorar la calidad de vida de las personas y poder realizar tareas que de otra forma serían impensables o se

convertiría en una acción de gran complejidad. Pero también hay que pensar, si este nuevo mundo, no podría ser también un mundo lleno de barreras, para las personas que padecen algún tipo de discapacidad. Esta afirmación es una realidad, porque no se piensa apenas en la accesibilidad y adaptación de las nuevas tecnologías para este colectivo, que puede presentar ciertos problemas al usar las distintas herramientas tecnológicas que existen en el mercado. Esta situación se puede ver en el software, con el uso excesivas de imágenes para transmitir información, en los dispositivos de entrada/salida, cuando se tiene problemas con el ratón o el teclado, además de factores de otra índole como puede ser un coste muy elevado.

En el ámbito educativo, se puede dar que tenga equipos , software con falta de accesibilidad y esto provoque que [30]:

- Marginación
- Falta de autonomía personal

### 3.4.1. Principios básicos

El Diseño Universal pretende paliar los problemas que suponen las TIC, con su filosofía de que puedan ser utilizadas por el mayor número de personas. Los principios del Diseño Universal son[29]:

- Uso equitativo: La aplicación tiene que ser útil para personas que posean diferentes capacidades. KiNEEt ha sido diseñada para que pueda resultar beneficiosa para el mayor número de alumnos con discapacidad, por ese motivo se han incluido actividades para los que tienen discapacidad visual y diferentes niveles según su capacidad intelectual, centrándose también en que el diseño sea agradable para los alumnos.
- Uso flexible: En este sentido la aplicación se tiene que adaptar a las capacidades individuales, por este motivo en este proyecto, todas las actividades tienen una pantalla de configuración, dónde se pueden elegir opciones como si se desea utilizar la mano izquierda o derecha para realizar el ejercicio.
- Uso simple e intuitivo: El uso de la aplicación no tiene suponer ninguna dificultad para los usuarios, haciendo alusión a este fin, se han puesto los botones con una imagen identificativa que expresa claramente la opción que se va a elegir y además cuando se pone el cursor encima de cualquier elemento gráfico de la interfaz aparece un mensaje, que explica la acción que se ejecuta si se interactúa con él. Además, en

determinadas actividades, se explica cómo hacer el ejercicio mediante una voz en off.

- Información perceptible: Esta característica se puede unir con la anterior, puesto que trata de mostrar la información de maneras diferentes, con sonido, imágenes, etc. Esto ha quedado reflejado en el punto anterior.
  
- Tolerancia al error: La aplicación no puede producir ninguna respuesta negativa, aunque el usuario realice alguna acción involuntaria o imprevista. Por este motivo, las pantallas de realización de actividades, que son con las que trabaja el usuario con discapacidad, sólo contiene los elementos que necesita para trabajar, ocultando elementos como botones para salir o volver, con el objetivo de que por culpa de una acción imprevista, se produzca un comportamiento no deseado por parte de la aplicación.
  
- Mínimo esfuerzo físico: La aplicación tiene que suponer una total comodidad para el usuario, lo que implica que éste no tenga que realizar posturas extrañas, que puedan suponer un riesgo para el usuario. Para cumplir esta finalidad, ya que la aplicación se basa en el movimiento del cuerpo, se ha seguido la guía de Microsoft denominada *Human Interface Guidelines* para realizar aplicaciones con Kinect que no resulten perjudiciales para la postura del individuo que usa la aplicación.
  
- Adecuado tamaño de aproximación y uso: En este sentido la aplicación, tiene que funcionar perfectamente y no entrañar ninguna complicación, se encuentre el usuario de pie o sentado. Esta característica ha sido posible completarla con éxito, gracias al *seat mode* que incorpora el SDK Microsoft. El *seat mode* permite reconocer el esqueleto del usuario, independientemente de que éste se encuentre sentado o de pie.

El modelo idóneo para las TIC en educación sería el siguiente:

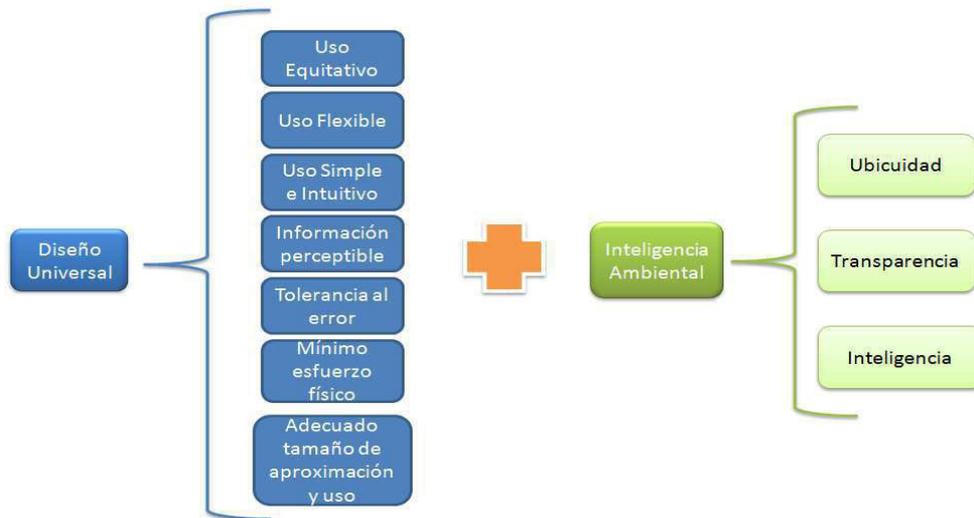


Figura 3.18: Diseño Universal e Inteligencia Ambiental



# Capítulo 4

## Aplicación KiNEEt

### 4.1. Arquitectura

En primer lugar se va a hacer una introducción a la arquitectura del sistema para después explicar qué modelo de proceso del software se ha implantado debido a las características del proyecto.

La arquitectura consta de tres módulos principales que son:

- Módulo de Procesamiento
- Módulo Human-Computer Interaction (HCI)
- Aplicación KiNEEt

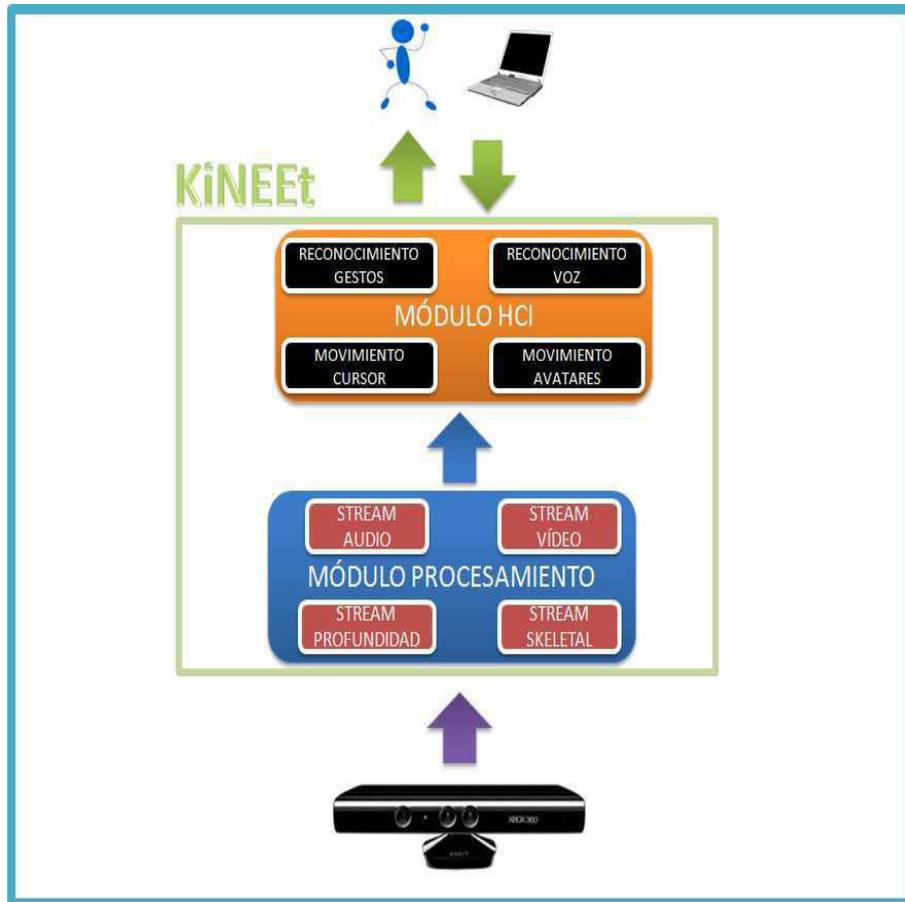


Figura 4.1: Arquitectura del sistema

El módulo de procesamiento contiene todas las operaciones necesarias que se tienen que realizar para poder realizar los datos que ofrece el sensor Kinect tanto de vídeo como de audio, así como el seguimiento del esqueleto para poder tener las posiciones relativas de los distintos joints y que la aplicación pueda hacer uso de esta información, que en definitiva es la base para poder trabajar con Kinect.

Una vez que se tiene a disposición el conjunto de datos necesarios que ofrece el dispositivo Kinect, en el módulo de HCI, se procura darle la máxima utilidad a estos datos. Aquí estaría incluido el reconocimiento de gestos, que utiliza la posición de los joints para poder reconocer con éxito un movimiento definido previamente o el reconocimiento de voz conocido como *speech recognition*, para poder reconocer palabras, incluso una gramática si fuera el caso.

Por último, una vez que el módulo de HCI, confirma que se ha reconocido

un gesto, la fase de aplicación se encarga de realizar la acción necesaria, dependiendo de la información que le haya transmitido el módulo anterior, este módulo ofrecerá una respuesta que se verá visualmente en la aplicación y así obtener el resultado esperado, así como una realimentación adecuada.

## 4.2. Proceso del Software

Como describe perfectamente Ian Somerville en su libro sobre Ingeniería del Software[26], un proceso del software es un conjunto de actividades que conducen a la creación de un producto software. A pesar de que existen diferentes procesos de software, hay una serie de actividades que son comunes a todos ellos:

- Especificación del software. Se define la funcionalidad que va a tener el software.
- Diseño e implementación del software. El software que se vaya a crear debe cumplir con una serie de especificaciones.
- Validación del software. Es necesario comprobar que este software realiza las funciones que ha pedido el cliente.
- Evolución del software. El software no es estático, sino que tiene que cambiar para adaptarse a las necesidades del cliente.

### 4.2.1. Modelo del proceso del software

Los modelos principales que más se utilizan en el proceso del software son:

- El modelo en cascada.
- Desarrollo evolutivo.
- Ingeniería del software basada en componentes.

Como este proyecto se ha basado en un desarrollo evolutivo, se van a detallar sólo los aspectos que tienen relación con este modelo específicamente.

El desarrollo evolutivo consiste en desarrollar una implementación inicial, para poder enseñársela a los clientes y así ir mejorando la aplicación hasta que se desarrolla un software que satisfaga sus especificaciones, como representa el siguiente esquema:

### Imagen de esquema Sommerville pág 63

En la realización del proyecto se ha seguido un desarrollo exploratorio[26], el cual se caracteriza por trabajar con el cliente para obtener sus requerimientos y entregar un sistema final.

Para la realización de sistemas pequeños y medianos es recomendable utilizar un desarrollo evolutivo, mientras que para los sistemas grandes es preferible utilizar un sistema mixto que contenga las mejores características del modelo en cascada y del desarrollo evolutivo.

Se eligió este modelo debido a que este proyecto tenía un cliente que era el CEEE Provincial Princesa Sofía, con la idea de que una serie de docentes propusieran una serie de actividades para desarrollar con Kinect y una vez implementadas éstas, probarlas con los alumnos, con el fin de comprobar que el alumno era capaz de realizarla correctamente, si no se distraía y obtener una realimentación para mejorar las actividades y pudieran ser útiles para los alumnos. Para llevar a cabo este proceso se mantenía una reunión los lunes con las especialistas para ver qué mejoras había que hacer a la aplicación o qué requisitos había que abordar y los viernes se probaba las actividades con los alumnos y así poder ir valorando el trabajo que se iba desarrollando.

#### 4.2.2. Actividades del proceso de software

Los **requisitos** son las actividades que va a realizar el sistema. Existen dos tipos de requisitos: funcionales y no funcionales. La diferencia entre ellos, es que los requisitos funcionales son las acciones que va a tener el sistema, estas acciones definirán el comportamiento del mismo, mientras que los no funcionales serán cualidades que el sistema presenta, como puede ser el rendimiento o tiempo de espera.

Para representar estos requisitos se van a utilizar los casos de uso, este procedimiento de la ingeniería de requisitos, permite ver y definir de una forma más clara estos requisitos. Los elementos principales que forman este modelo y habrá que identificar son [25]:

- Actores: Roles que desempeñan los usuarios que van a utilizar el sistema.
- Casos de Uso: Acciones que los actores pueden desempeñar en la aplicación.
- Relaciones: Relaciones existentes entre los actores y los casos de uso.

El modelo de casos de uso ha sido diseñado según las especificaciones que aconsejan Ila Neustadt y Jim ARlow en su libro sobre UML2[25] y las cito textualmente:

- Mantener los casos de uso breves y sencillos
- Centrarse en el qué, no en el cómo
- Evite descomposición funcional

En cuanto al **diseño**, su objetivo es que la idea que se tenía en fases anteriores y se le estaba dando forma con las especificaciones de los requisitos, se pueda materializar especificando la funcionalidad que tiene que implementar el sistema[25]. Los diagramas de clases va a ser el recurso utilizado para representar las características básicas de esta fase. En esta etapa las clases se diseñan para que pueda ser posible crear el código fuente a partir de ellas, por este motivo tienen que estar bien definidas y completas con sus atributos y operaciones correspondientes, para que la implementación sea una tarea lo más cómoda posible.

La sección de la **implementación** va a expresar los métodos y las nociones más importantes sobre cómo se ha construido este proyecto. La aplicación consta de un conjunto de actividades, la mayoría realizadas con XNA, sólo las que se encuentran en 3D están desarrolladas con Unity. Por este motivo, estas actividades contarán con el modelo que se presento anteriormente, en el apartado sobre XNA y contarán con los métodos Load(), Update() y Draw() principalmente, siendo común para toda ellas.

Otra característica es que la aplicación está dividida en pantallas, es decir, cuando se está en la pantalla de selección de actividades y se selecciona una de ellas, se puede comprobar que cambia la interfaz, apareciendo las actividades que se pueden elegir, en ese momento nos encontramos en una pantalla distinta a la original. Todas estas pantallas tienen en común que necesitan el uso de Kinect, con lo cual se ha diseñado esta clase con el patrón singleton[27], con la finalidad de que la clase que necesite sólo tenga que instanciarla y no tenga que volver a crearla, ya que es común para todas, al tratarse del mismo recurso.

Como se ha comentado es necesario tener la clase Kinect activa en todo momento, pero más indispensable aún es tener la función de skeletal tracking en continuo funcionamiento, debido a que es indispensable para realizar cualquier acción dentro de la aplicación, no sólo para interactuar y completar los ejercicios satisfactoriamente, sino para acciones básicas como seleccionar un ejercicio o salir de la aplicación. Sin embargo, no ocurre con lo mismmo

con el stream de la cámara de color, ya que éste sólo se activará en aquellos casos que sea necesario utilizar la cámara, ya que en todos los ejercicios no se puede activar dicha opción, no se ve necesaria la activación permanente de este flujo de datos.

### 4.2.3. Iteración de procesos

En este sentido se ha seguido un desarrollo en espiral, realizando varios prototipos para finalmente entregar el desarrollo final del mismo, en vez de usar una entrega incremental donde cada una de las actividades del proceso de software se dividen en incrementos que se desarrollan por turnos.

Estos prototipos han ido acorde a los diferentes módulos que componen la aplicación global. Los distintos prototipos se hicieron en este orden:

Los ciclos que se han creado para este tipo de desarrollo han ido acorde a cada uno de los módulos que se componen la aplicación, así que los ciclos del desarrollo en espiral son los siguientes:

1. Ciclo del módulo de actividades.
2. Ciclo del módulo de coordinación.
3. Ciclo del módulo de arte.
4. Ciclo del módulo de comunicador.

### 4.2.4. Ciclos del desarrollo en espiral

En esta sección se describirán las actividades del proceso software de cada uno de los ciclos que componen el proyecto. Estas actividades son los requisitos, diseño, implementación y pruebas, es necesario decir que tanto la implementación como las pruebas se han incluido como un apartado independiente, explicando cómo se ha hecho este proceso de forma genérica para los ciclos.

En la sección de requisitos de cada módulo están involucrados estos cinco actores que son comunes para todos ellos:

- Usuario
- Especialista
- Alumno

- Interfaz XML
- Kinect

#### 4.2.4.1. Ciclo del módulo de actividades

En este módulo, se realizaron los prototipos del módulo correspondiente a las actividades, formada por las actividades de los números, formas geométricas, causa-efecto y grafomotricidad.

Los casos de uso correspondientes a este ciclo se muestran en el Cuadro 4.1. Para ver el diagrama de casos de uso completo mirar Anexo E

Casos de Uso		
Salir aplicación	Ejecutar números	Ejecutar formas geométricas
Ejecutar causa-efecto	Ejecutar grafomotricidad	Trazar línea
Realizar ejercicio	Detectar esqueleto	Realizar feedback
Realizar seguimiento esqueleto	Procesar stream audio	Procesar stream color
Procesar stream profundidad	Leer archivo configuración	Reiniciar ejercicio
Seleccionar actividades	Seleccionar números	Seleccionar formas geométricas
Seleccionar causa-efecto	Seleccionar grafomotricidad	Configurar números
Configurar formas geométricas	Configurar causa-efecto	Configurar grafomotricidad
Crear archivo de configuración	Modificar archivo configuración	

Cuadro 4.1: Tabla Casos de Uso del Módulo Actividades

En cuanto al diseño las clases más relevantes son las que se explican a continuación.

La clase **NumbersScreen**

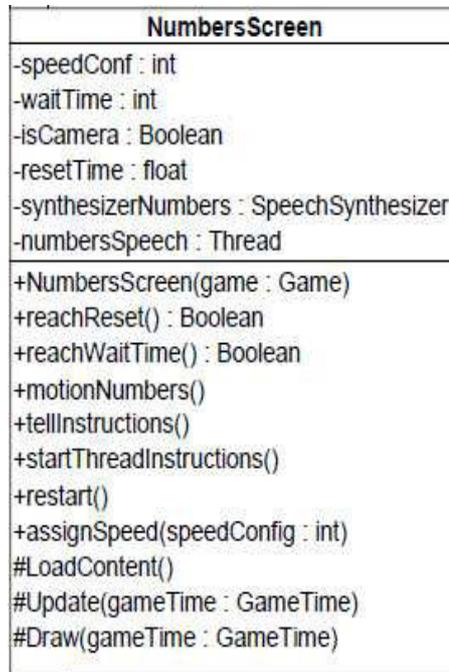


Figura 4.2: Clase de la actividad de Números

Esta clase tiene como atributos principales:

- `speedConf` → Velocidad de caída de los números.
- `waitTime` → Tiempo de espera para pasar a la siguiente secuencia, dependiendo del alumno necesitará más o menos tiempo para prepararse para el siguiente reto de la actividad y asimilar el feedback.
- `resetTime` → Es el tiempo que se necesitará para poder empezar la actividad con otro alumno o preparar al mismo para realizar la actividad otra vez si se considera necesario.
- `isCamera` → Activa la cámara de Kinect para ponerlo como fondo de pantalla, en vez de un fondo abstracto monocolor.

Los métodos más relevantes de esta clase son los siguientes:

En el método `LoadContent()`, cargará todos los elementos que sean necesarios para la visualización correcta del ejercicio, como las imágenes de las manos que hacen de cursor, los propios números o el fondo de pantalla.

En el método `Update(gameTime : gameTime)`, comprobará en cada instante si el alumno ha tocado el número y realizar el feedback correspondiente, la

fase en la que se encuentra de la actividad, si se reinicia la actividad o se da por finalizada.

En el método `Draw(gameTime : gameTime)`, se dibujan los componentes que han sido cargados previamente.

El método `motionNumbers()`, es el que controlará el movimiento de los números, como la velocidad a la que se desplazan o si han llegado al final de la pantalla y tienen que volver a salir por la parte superior.

El método `tellInstructions()`, permitirá mediante síntesis de voz decir el objetivo de la actividad para poner al alumno en situación.

La clase **ShapesScreen**

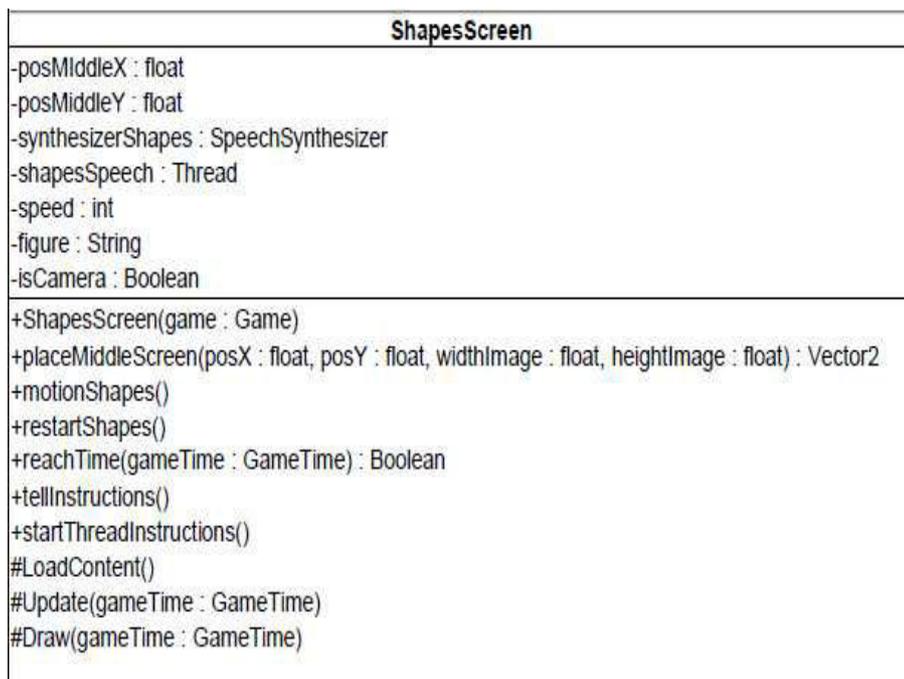


Figura 4.3: Clase de la actividad de Formas Geométricas

Esta clase tiene como atributos principales:

- `speed` → Velocidad de caída de las formas geométricas.
- `figure` → Es la figura que se ha elegido como discriminadora y la única que tendrá efecto cuando haga colisión con cualquiera de las dos manos.

- `isCamera` → Activa la cámara de Kinect para ponerlo como fondo de pantalla, en vez de un fondo abstracto monocolor.

Los métodos más relevantes de esta clase son los siguientes:

En el método `LoadContent()`, cargará todos los elementos que sean necesarios para la visualización correcta del ejercicio, como las imágenes de las manos que hacen de cursor, las distintas figuras que aparecen en escena o el fondo de pantalla.

En el método `Update(gameTime : gameTime)`, comprobará en cada instante si el alumno ha tocado la figura correcta para realizar el feedback correspondiente, si se reinicia la actividad o se da por finalizada.

En el método `Draw(gameTime : gameTime)`, se dibujan los componentes que han sido cargados previamente.

El método `motionShapes()`, controlará ciertos aspectos referentes al movimiento de las figuras, al igual que los números la velocidad de caída o saber si han llegado al final de la pantalla, serán circunstancias a tener en cuenta para la realización de la actividad.

El método `tellInstructions()`, permitirá mediante síntesis de voz decir el objetivo de la actividad para poner al alumno en situación.

El método `placeMiddleScreen(posX : float, posY : float, widthImage : float, heightImage: float)` se llama cuando se tiene que efectuar el feedback, para localizar la figura geométrica discriminante en el centro de la pantalla.

La clase **LightScreen**



Figura 4.4: Clase de la actividad de Causa-efecto

Esta clase tiene como atributos principales:

- level → Indica el nivel que se ha elegido para la realización del ejercicio.
- isOn → Este atributo permite saber si la luz se encuentra en estado encendido o apagado.
- resetTime → Este tiempo sirve para que el alumno se dé cuenta de la reacción causa-efecto que produce apagar y encender la luz, es un tiempo de espera donde el usuario tiene que entender que cuando se hace colisión con la bombilla es posible apagar o encender la luz del entorno.
- isCamera → Activa la cámara de Kinect para ponerlo como fondo de pantalla, en vez de un fondo abstracto monocolor.

Los métodos más relevantes de esta clase son los siguientes:

En el método LoadContent(), cargará todos los elementos que sean necesarios para la visualización correcta del ejercicio, como las imágenes de las manos que hacen de cursor, las distintas figuras que aparecen en escena o el fondo de pantalla.

En el método `Update(gameTime : gameTime)`, comprobará en cada instante si el alumno ha tocado con su mano la bombilla y cambiará el estado de la misma para ofrecer un resultado diferente en la pantalla. Además, según el nivel controlará la posición de la bombilla en la pantalla.

En el método `Draw(gameTime : gameTime)`, se dibujan los componentes que han sido cargados previamente.

El método `switchLight()`, se encargará de alternar la imagen entre bombilla encendida o apagada dependiendo del estado de ésta.

El método `placeLight()`, calculará la nueva posición de la bombilla dependiendo del nivel. Un nivel fácil hará que la bombilla se sitúe cerca del usuario y un nivel más complejo desplazará la bombilla más lejos del alcance del usuario, teniendo éste que esforzarse más y realizar un movimiento más amplio.

La clase **HandwritingScreen**



Figura 4.5: Clase de la actividad de Grafomotricidad

Esta clase tiene como atributos principales:

- `jointHand` → Indica si se va a realizar la actividad con la mano derecha o con la izquierda.

- `isPaint` → Confirma si se ha tocado la figura inicial y permite pintar en la pantalla para poder realizar la actividad.
- `video` → Representa al vídeo que se reproducirá cuando el niño haya completado la actividad con éxito .
- `isCamera` → Activa la cámara de Kinect para ponerlo como fondo de pantalla, en vez de un fondo abstracto monocolor.

Los métodos más relevantes de esta clase son los siguientes:

En el método `LoadContent()`, cargará todos los elementos que sean necesarios para la visualización correcta del ejercicio, como las imagen del lápiz que hace de cursor, la figura inicial y final del ejercicio o el fondo de pantalla.

En el método `Update(gameTime : gameTime)`, comprobará en cada instante si el alumno ha tocado la figura inicial y puede empezar a realizar la actividad o incluso si ha completado con éxito y es posible reproducir el vídeo que representa al feedback para la estimulación del alumno.

En el método `Draw(gameTime : gameTime)`, se dibujan los componentes que han sido cargados previamente.

El método `paintLogic(gameTime : gameTime)`, comprobará cuando se puede pintar en el área destinado para ello.

El método `identifyJoint(jointName : String)`, devuelve el joint que le corresponde a la elección de la mano configurada por el usuario para poder realizar el tracking de ese joint e interactuar de forma correcta en el ejercicio.

El método `loadVideo()`, se encargará de configurar los parámetros del vídeo, para reproducirlo con el formato adecuado.

Para ver el diseño conjunto de la aplicación, es necesario ver el diagrama de clases completo, que se encuentra en el anexo I

#### 4.2.4.2. Ciclo del módulo de coordinación

En este módulo, se realizaron los prototipos del módulo correspondiente a la coordinación, en el cual se encuentra el ejercicio que está orientado a la mejora física del individuo.

Los casos de uso correspondientes a este ciclo se muestran en el Cuadro 4.2. Para ver el diagrama de casos de uso completo mirar Anexo E

Casos de Uso		
Salir aplicación	Ejecutar coordinación	Unir círculos
Realizar ejercicio	Detectar esqueleto	Realizar feedback
Realizar seguimiento esqueleto	Procesar stream audio	Procesar stream color
Procesar stream profundidad	Leer archivo configuración	Reiniciar ejercicio
Seleccionar coordinación	Configurar coordinación	Crear archivo de configuración
Modificar archivo configuración		

Cuadro 4.2: Tabla Casos de Uso del Módulo Coordinación

En cuanto al diseño las clases más relevantes son las que se explican a continuación.

La clase **SkillScreen**

<b>SkillScreen</b>
-positionCircle : Vector2[]
-lines : Vector2[]
-finishExercise : Boolean
-confSkill : ConfSkillXML
-jointName : String
-isCamera : Boolean
-feedback : Thread
-soundSuccess : SoundEffect
-jointType : JointType
+skillScreen(game : Game)
+identifyJoint(jointName : String) : JointType
+feedbackSound()
+playSuccess()
+createPositionCircles()
+addCircles()
+drawLinesConnected()
#LoadContent()
#Update(gameTime : GameTime)
#Draw(gameTime : GameTime)

Figura 4.6: Clase de la actividad de Coordinación

Esta clase tiene como atributos principales:

- `jointType` → Indica el joint con el que se va a ejecutar este ejercicio.
- `positionCircle` → Representa las posiciones que cada uno de los círculos tiene en la pantalla.
- `lines` → Representa la posición de las líneas que se tienen que dibujar en pantalla, debido a la unión de los círculos.
- `isCamera` → Activa la cámara de Kinect para ponerlo como fondo de pantalla, en vez de un fondo abstracto monocolor.

Los métodos más relevantes de esta clase son los siguientes:

En el método `LoadContent()`, cargará todos los elementos que sean necesarios para la visualización correcta del ejercicio, en este caso, los círculos que componen la actividad o el fondo de pantalla.

En el método `Update(gameTime : gameTime)`, comprobará qué estado se encuentra el círculo, si ha sido tocado o es un nodo objetivo o si se ha alcanzado el objetivo del ejercicio.

En el método `Draw(gameTime : gameTime)`, se dibujan los componentes que han sido cargados previamente.

El método `createPositionCircles()`, asigna la posición correspondiente a cada círculo, dependiendo de dónde se haya colocado en la pantalla de edición.

El método `addCircles()`, añade los círculos a la pantalla para poder interactuar con ellos.

El método `drawLinesConnected()`, dibuja cada una de las líneas que están conectadas entre dos círculos.

Para ver el diseño conjunto de la aplicación, es necesario ver el diagrama de clases completo, que se encuentra en el anexo I

#### 4.2.4.3. Ciclo del módulo de arte

En este módulo, se realizaron los prototipos del módulo correspondiente al arte, en el cual el usuario puede realizar las actividades artísticas relacionadas con danza, música y pintura.

Los casos de uso correspondientes a este ciclo se muestran en el Cuadro 4.3. Para ver el diagrama de casos de uso completo mirar Anexo E

Casos de Uso		
Salir aplicación	Ejecutar danza	Ejecutar música
Ejecutar pintura	Seleccionar color rojo	Seleccionar color verde
Seleccionar color azul	Seleccionar trazo fino	Seleccionar trazo medio
Seleccionar trazo grueso	Limpiar área dibujo	Reiniciar ejercicio
Realizar ejercicio	Detectar esqueleto	Realizar feedback
Realizar seguimiento esqueleto	Procesar stream audio	Procesar stream color
Procesar stream profundidad	Leer archivo configuración	Seleccionar arte
Seleccionar danza	Seleccionar música	Seleccionar pintura
Configurar pintura	Crear archivo de configuración	Modificar archivo configuración

Cuadro 4.3: Tabla Casos de Uso del Módulo Arte

La clase **PaintScreen**

<b>PaintScreen</b>
-redButton : Button
-blueButton : Button
-greenButton : Button
-thinButton : Button
-normalButton : Button
-boldButton : Button
-clearButton : Button
-frameRed : Rectangle
-frameBlue : Rectangle
-frameGreen : Rectangle
-frameThin : Rectangle
-frameNormal : Rectangle
-frameBold : Rectangle
-confPaint : ConfPaintXML
-jointHand : String
-isPaint : Boolean
-drawingArea : Rectangle
-currentColor : Color
-currentThick : float
+PaintScreen(game : Game)
+createDrawingArea()
+identifyJoint(jointName : String) : JointType
+redAction()
+greenAction()
+blueAction()
+thinAction()
+normalAction()
+boldAction()
+clearAction()
-collisionRed(gameTime : GameTime)
-collisionGreen(gameTime : GameTime)
-collisionBlue(gameTime : GameTime)
-collisionThin(gameTime : GameTime)
-collisionNormal(gameTime : GameTime)
-collisionBold(gameTime : GameTime)
-collisionClear(gameTime : GameTime)
-paintHandRight(firstSkeleton : Skeleton)
-paintHandLeft(firstSkeleton : Skeleton)
#LoadContent()
#Update(gameTime : GameTime)
#Draw(gameTime : GameTime)

Figura 4.7: Clase de la actividad de Pintar

Esta clase tiene como atributos principales:

- jointType → Indica el joint con el que se va a ejecutar este ejercicio.
- drawingArea → Representa la zona de dibujo donde se va a poder pintar.
- currentColor → Hace referencia al color que está seleccionado y con el que se pintará en el área de dibujo.
- currentThick → Este atributo representa al grosor del trazo en vez de su color.

Los métodos más relevantes de esta clase son los siguientes:

En el método `LoadContent()`, cargará todos los elementos que sean necesarios para la visualización correcta del ejercicio, como las distintas opciones de colores que se puede elegir, los distintos tipos de grosor o botones de inicio y borrado.

En el método `Update(gameTime : gameTime)`, corrobora si se está en modo de selección para poder elegir otro color o grosor o si por el contrario está en modo pintura y puede el usuario comenzar a pintar .

En el método `Draw(gameTime : gameTime)`, se dibujan los componentes que han sido cargados previamente.

El método `paintHandRight(firstSkeleton : Skeleton)`, determina que se pinta con la mano derecha y que es necesario tener el brazo izquierdo apoyado en la cabeza, para poder pintar en la aplicación.

El método `collisionRed(gameTime : gameTime)`, comprueba que se está manteniendo contacto con el botón que representa al color rojo y cuánto tiempo está en contacto el cursor con dicho botón.

El método `collisionGreen(gameTime : gameTime)`, comprueba que se está manteniendo contacto con el botón que representa al color verde y cuánto tiempo está en contacto el cursor con dicho botón.

El método `collisionBlue(gameTime : gameTime)`, comprueba que se está manteniendo contacto con el botón que representa al color azul y cuánto tiempo está en contacto el cursor con dicho botón.

#### **4.2.4.4. Ciclo del módulo de comunicador**

En este módulo, se realizaron los prototipos del módulo correspondiente al comunicador, en el cual se encuentra el comunicador que le permite a los alumnos con dificultades en el lenguaje poder realizar sus peticiones de una manera más cómoda.

Los casos de uso correspondientes a este ciclo se muestran en el Cuadro 4.4. Para ver el diagrama de casos de uso completo mirar Anexo E

Casos de Uso		
Salir aplicación	Ejecutar comunicador	Reconocer gestos
Reconocer voz	Realizar ejercicio	Detectar esqueleto
Realizar feedback	Realizar seguimiento esqueleto	Procesar stream audio
Procesar stream color	Procesar stream profundidad	Leer archivo configuración
Reiniciar ejercicio	Seleccionar comunicador	Configurar comunicador
Crear archivo de configuración	Modificar archivo configuración	

Cuadro 4.4: Tabla Casos de Uso del Módulo Comunicador

### La clase **CommunicatorScreen**

<b>CommunicatorScreen</b>
-kinectRecognizerInfo : RecognizerInfo -recognizerSpeech : SpeechRecognitionEngine -kinectSource : KinectAudioSource -commands : Choices -grammar : Grammar -xml : XMLManager -confCommunicator : ConfCommunicatorXML -jointHand : String -jointType : JointType -isGestureActive : Boolean -hoverPictogram : Pictogram -isHoverPictogram : Boolean -synthesizer : SpeechSynthesizer -threadSpeech : Thread
+CommunicatorScreen(game : Game) +identifyJoint(jointName : String) : JointType +assignNextPhase() : GameState +initializeWordsToSpeech() +findKinectRecognizerInfo() : RecognizerInfo +createChoices() +createGrammar() +setupAudio() +recognizeWords(word : String) +gestureActions(gesture : GestureType) +updateStates(gameTime : GameTime) +loadTextToSpeech() +tellSentence() +startThreadSpeech() #LoadContent() #Update(gameTime : GameTime) #Draw(gameTime : GameTime)

Figura 4.8: Clase de la actividad del Comunicador

Esta clase tiene como atributos principales:

- `kinectSource` → Representa al módulo de audio de Kinect para poder realizar el reconocimiento de voz.
- `kinectRecognizerInfo` → Almacena la información sobre el reconocimiento de voz, como por ejemplo qué idioma se desea reconocer.
- `commands` → Incluye las palabras que van a ser reconocidas mediante la voz.
- `grammar` → Representa la gramática que es capaz de reconocer la aplicación con las palabras que se han elegido.
- `isGestureActive` → Indica si se ha activado al opción de gestos.

Los métodos más relevantes de esta clase son los siguientes:

En el método `LoadContent()`, cargará todos los elementos que sean necesarios para la visualización correcta del ejercicio, como las posiciones donde van los pictogramas principales o el cursor de la mano si no se ha elegido la opción de gestos.

En el método `Update(gameTime : gameTime)`, realiza la acción correspondiente dependiendo del gesto o si ha reconocido alguna palabra de la gramática.

En el método `Draw(gameTime : gameTime)`, se dibujan los componentes que han sido cargados previamente.

El método `createChoices()`, asigna cada una de las palabras que se pretende reconocer.

El método `createGrammar()`, construye la gramática, a partir de las palabras que va a identificar el programa.

El método `gestureActions(gesture : GestureType)`, comprueba qué gesto ha hecho el usuario y si se corresponde con alguno de los implementados.

El método `recognizeWords(word : String)`, comprueba que la palabra que ha pronunciado el usuario se encuentra en la gramática.

### 4.3. Implementación

#### 4.3.1. Movimiento del cursor

Con el fin de mover el cursor, es necesario realizar un seguimiento de uno de los joints que es capaz de identificar el SDK de Kinect, el más común y el que más se utiliza en el proyecto es el designado por la mano derecha, para acceder a un joint es tan fácil como muestra la siguiente línea de código:

```
firstSkeleton.Joints[JointType.HandRight]
```

Los joints tienen una propiedad denominada *Position* que se utilizará para designarla al cursor y así en todo momento poder obtener la posición actual del joint, pero esta posición estará representada en los ejes X, Y, Z, con lo cual no ofrecerá una posición fiable en la pantalla que sólo trabaja con los ejes X, Y. Para una representación fiable de la posición en dos dimensiones, habrá que escalar esta propiedad, en relación a la resolución que se haya escogido de la cámara del dispositivo Kinect.

```
firstSkeleton.Joints[JointType.HandRight].Position
```

#### 4.3.2. El proceso de selección de actividades

La aplicación se puede utilizar en su totalidad con el sensor Kinect, sin uso de teclado o ratón. Esta decisión nos lleva a crear una interfaz donde los menús contienen botones de gran tamaño para así poder utilizar una mano como cursor, que se identificará con la mano derecha del usuario y poder moverla por la interfaz para ir realizando las acciones pertinentes. Este movimiento del cursor se ha explicado en la sección anterior.

Para poder seleccionar una actividad, se mantiene el cursor encima del botón deseado, y una vez detectada la colisión entre el cursor y el botón, se activará una barra en la zona inferior de la pantalla, que irá progresando si se mantiene en contacto el cursor hasta 2 segundos, que será cuando el sistema decidirá que se quiere seleccionar esta actividad y hará la acción que proceda a la actividad correspondiente.

### 4.3.3. Método de colisiones con bounding box

El método de colisiones, que se ha nombrado en la sección anterior para poder seleccionar una acción dentro del programa, se basa en el método denominado comúnmente bounding box. Para implementarlo, es necesaria la creación de un rectángulo tanto en el cursor como en los botones que se componen la interfaz. Estos rectángulos han sido creados con la clase *Rectangle* que nos permite con su método *Intersect*, donde se le pasa por parámetro un argumento de tipo *Rectangle*, saber si ha habido una colisión entre estos dos elementos.

### 4.3.4. Obtener el skeleton más cercano

Una característica que ha sido necesaria implementar, es el de poder reconocer al usuario más cercano al dispositivo Kinect, puesto que al poder reconocer el esqueleto de dos personas al mismo tiempo, éstos pueden interferir en el proceso de selección del sensor Kinect sobre cuál de los dos realiza el seguimiento y por consiguiente toma el control de la aplicación, por ejemplo el cuerpo del especialista puede ser motivo de confusión cuando se quiere reconocer al usuario que está usando la aplicación, haciendo un reconocimiento erróneo del sujeto que verdaderamente está realizando la actividad, por ello estas líneas de código han sido totalmente necesarias para el correcto uso de la aplicación.

En ellas se puede observar como el método *getMinDepth*, selecciona de entre todos los usuarios que ha reconocido el dispositivo Kinect (pueden ser hasta 6), el que menor valor de la componente Z posee en la posición de su esqueleto, para así después elegir en la actividad el esqueleto de la persona que coincide con ese valor obtenido. De esta forma se elegirá siempre al individuo que se encuentre más cerca del sensor Kinect, no habiendo error si el especialista necesita colocarse detrás, para ayudar al alumno con la realización de la actividad, si fuera necesario.

```
public override float getMinDepth(List<Skeleton> skeletons)
{
    float minDepth = 100;
    foreach (Skeleton s in skeletons)
    {
        if (s.Position.Z < minDepth)
            minDepth = s.Position.Z;
    }

    return minDepth;
}
```

Figura 4.9: Método getMinDepth()

```
firstSkeleton = (from trackSkeleton in totalSkeleton
    where trackSkeleton.TrackingState == SkeletonTrackingState.Tracked
    && trackSkeleton.Position.Z == minDepth
    select trackSkeleton).FirstOrDefault();
```

Figura 4.10: Elección del Skeleton más cercano

### 4.3.5. Proceso de reconocimiento de gestos

El proceso de reconocimiento de gestos es un proceso que va a ser fundamental en esta aplicación, para poder realizar determinadas acciones en el programa con un sencillo gesto. El gesto va a ser definido como un movimiento progresivo en un determinado espacio de tiempo que siempre sigue el mismo patrón. Cada gesto tendrá su patrón único para así no poder confundirlo con otros gestos, cuanto mayor sea la diferencia entre ellos, menor será el número de errores, existe una relación inversamente proporcional entre la diferenciación de patrones y los errores que conlleva el reconocimiento de un gesto.

Para el reconocimiento de gestos va a ser fundamental el cálculo de la distancia entre dos joints, ya que se puede saber mediante la propiedad *Position*, como se ha nombrado en secciones anteriores, y controlar en ciertos momentos donde se encuentra el joint de referencia y si está realizando el movimiento adecuado.

Los pasos para reconocer un gesto son básicamente tres:

1. Se reconoce una posición de partida, con el fin de saber cuando empieza el proceso de reconocimiento, de un determinado gesto
2. Es necesario tener un proceso de validación que verifique antes de llegar al estado final que el gesto que se pretende reconocer según la posición de partida, se está realizando con éxito y si esto no fuera así se abortaría el proceso de reconocimiento. Un ejemplo sencillo para entender mejor este procedimiento, podría ser que el objetivo del gesto que se quiere reconocer es tocarse la cabeza. En este punto se verificaría que se está realizando un movimiento hacia arriba del brazo desde la posición inicial, dando como erróneo un movimiento que implicara que el brazo se está desplazando hacia las piernas desde la posición de inicio.
3. El último paso sería reconocer una posición final, dando por finalizado, el proceso de reconocimiento independientemente del resultado final del mismo.

#### 4.3.6. Base de Datos

Los juegos que se desarrollan con XNA, sobre todo para Windows Phone, no suelen utilizar una base de datos como MySQL, Oracle o alguna similar, sino que se utilizan archivos XML. En este proyecto se ha seguido la lógica de utilizar archivos XML como base de datos para el proyecto, por la razón principal de que el sistema no almacena grandes volúmenes de datos y no es necesaria la utilización de una base de datos compleja y de gran tamaño.

Se ha requerido el uso de LINQ to XML para las distintas operaciones que se han realizado con estos archivos, que han sido los procesos de creación, modificación y lectura de los datos en XML. La apuesta por esta interfaz de programación ha sido porque de una manera rápida y sencilla permite escribir consultas en el documento XML, ya que LINQ es el lenguaje que se utiliza en .NET por excelencia para el acceso a base de datos.

Para observar gráficamente a que se está refiriendo, se va a ofrecer un ejemplo:

## 4.4. PRUEBAS

---

```
var configuration = from communicator in document.Descendants("communicator")
                    select new ConfCommunicatorXML
                    {
                        JointKinect = communicator.Element("joint").Value,
                        IsGesture = Convert.ToBoolean(communicator.Element("gestures").Value)
                    };
```

Figura 4.11: Lectura de archivo XML

Como se puede apreciar, en un par de líneas podemos hacer una lectura de los datos con la estructura esencial de una consulta en SQL, pudiendo hacer recuperación de colecciones.

## 4.4. Pruebas

Durante el transcurso del proyecto se ha estado colaborando con CEEE Provincial Princesa Sofía, probando la aplicación con sus estudiantes cuando se tenía un ejercicio terminado para ver la reacción de los alumnos ante la nueva actividad, y al mismo tiempo procurar que se vayan acostumbrando a esta nueva tecnología, puesto que no sólo para ellos, sino para cualquier persona, el cambio puede suponer rechazo y hay que adaptarse a ello.

Además también se ha realizado pruebas con la asociación Asprodalba, debido a que los alumnos del CEEE Provincial Princesa Sofía tienen edades comprendidas entre los 5 y 21 años, mientras que dicha asociación está formada por personas adultas y se pretendía ver la reacción de estas actividades con un público más adulto, con la finalidad de comprobar si las actividades también les podía servir a ellos o por el contrario habría que modificarlas, incluso descartarlas.

Por último, se presentan las fotografías tomadas tanto en el CEEE Provincial Princesa Sofía como en el encuentro Arte + de colombia, donde personas con discapacidad están probando la aplicación. **Colocar imágenes de las pruebas**



# Capítulo 5

## Resultados

### 5.1. Estudio de la aplicación

Desde Enero de 2013 se ha estado colaborando con cuatro tutoras del CEEE Princesa Sofía, para intentar adaptar la aplicación a las necesidades de su alumnado. Además, los alumnos de dichas tutoras han estado probando los diferentes ejercicios que componen la aplicación en el transcurso de su desarrollo, para ver su reacción con este novedoso sistema de ejercicios, del cual el centro no disponía nada parecido en años anteriores, y así poder hacer el proceso de realimentación que fuese oportuno para que tanto los profesionales como los estudiantes puedan obtener una óptima experiencia con la herramienta.

Una vez finalizada la aplicación, coincidiendo con el final del estudio, se ha presentado un cuestionario sobre usabilidad y uso educativo de la aplicación, para que éstas las rellenen de acuerdo a su experiencia en este proyecto. Lamentablemente no ha sido posible obtener la colaboración de las cuatro especialistas, estando una de ellas de baja por maternidad, en la fecha de realización de la misma, así que sólo se dispone la valoración de las restantes tres tutoras.

Sin embargo, se ha obtenido una valoración de un especialista en técnicas educativas y un especialista en informática.

En los anexos, hay un apartado C, en el cual se dispone del formulario entregado para la obtención de resultados del estudio. Posteriormente, se va a proceder a la visualización de los resultados del mismo mediante unas gráficas, que contienen la información obtenida en esta experiencia.

El cuestionario está dividido en tres partes claramente diferenciables:

- Usabilidad
- Uso educativo
- Período de tiempo

En la sección de usabilidad, se ha querido evaluar si la aplicación es fácil de utilizar por parte de los expertos, siendo capaces de acceder a cada una de los ejercicios desarrollados en la aplicación, sin ningún problema de ambigüedad. También era indispensable valorar la característica de usar Kinect en este ámbito, puesto que el sensor Kinect no sólo se ha propuesto como medio para realizar las actividades, sino que se ha utilizado para poder navegar por las diferentes pantallas de la aplicación, así como para seleccionar las opciones de configuración permitidas, en cada una de las actividades.

La sección de uso educativo valoraba si la aplicación resultaba útil en el plano docente. Era primordial en el proyecto que el programa fuera una vía para desarrollar habilidades que se podían estimular con otros procedimientos más tradicionales pero que podían resultar más incómodos para el alumno. Los aspectos fundamentales en esta parte del cuestionario era averiguar si se cumplían con las demandas que necesitaba el alumno para su desarrollo en general, a la vez que para el docente se ha convertido en un medio para conseguir que el alumno se implique más en el proceso de aprendizaje, con la estimulación que puede ofrecer el uso de la tecnología.

La última sección trataba de evaluar si la aplicación iba a servir para un corto período de tiempo o por el contrario este proyecto va a tener un uso continuado en el centro.

Es necesario explicar algunas aclaraciones sobre las gráficas que se van a ilustrar:

- La puntuación que se propuso para evaluar cada una de las sentencias del cuestionario fue de 1 a 5, valorado el 1 como la puntuación más negativa y 5 la más positiva.
- El apartado denominado como 'Período de tiempo' no había que evaluarlo con una puntuación específica, sino que simplemente había que marcar con una X la opción con la que se estuviera conforme.
- Después de cada gráfica se va a exponer la sentencia original del cuestionario para que se tenga una noción más amplia, del apartado que evalúa cada una de las gráficas. Debido a que en la gráfica se ha tenido

que resumir el concepto evaluado por motivos de espacio, se considera oportuno tomar esta medida.

### 5.1.1. Evaluación Usabilidad

En este apartado se van mostrar los resultados obtenidos en el tema de usabilidad en la aplicación.

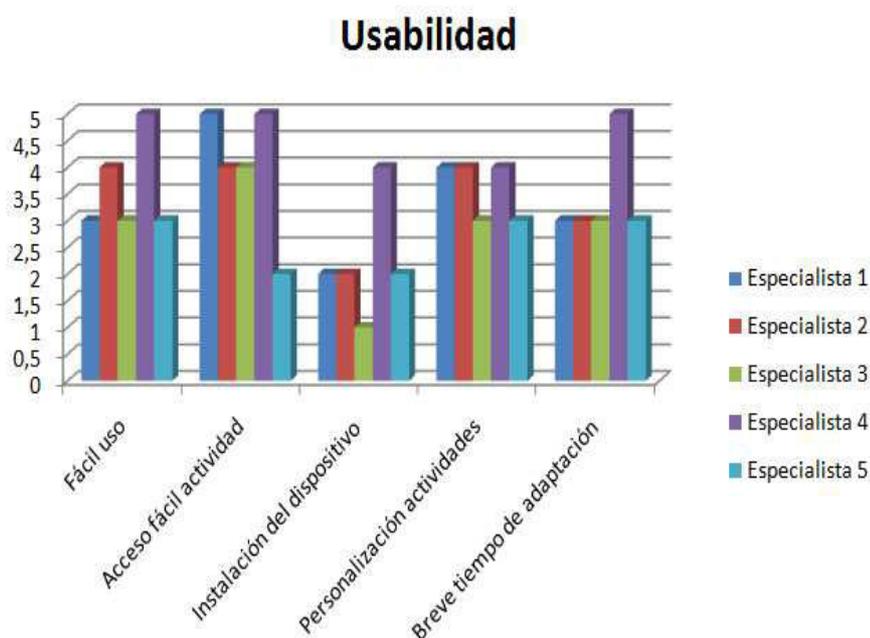


Figura 5.1: Evaluación de Usabilidad

Fácil uso  $\simeq$  La aplicación es fácil de usar

Acceso fácil actividad  $\simeq$  Se accede fácilmente al tipo de actividad que quiero hacer

Instalación del dispositivo  $\simeq$  Sé hacer la instalación del dispositivo sin problemas

Personalización actividades  $\simeq$  La personalización de las actividades, sé hacerlas fácilmente

Breve tiempo de adaptación  $\simeq$  Me he adaptado al uso de la aplicación en un breve período de tiempo

Se ha considerado de vital importancia tener conocimiento acerca de la facilidad o los problemas que se pueden ocasionar al usar esta herramienta creada, esta idea determinó llevar a cabo uno de los métodos usados en usabilidad para evaluar en cierta medida la experiencia de interacción de los usuarios potenciales de la aplicación, se trata de las encuestas. En este caso, estos usuarios mencionados son, las educadoras del Centro Provincial Princesa Sofía, siendo las personas que van a manejar en gran medida la aplicación, proponiendo a los alumnos distintos ejercicios, con los que desarrollar las habilidades que tengan más carencia.

Para la evaluación se han propuesto una serie de características, que debería tener la aplicación, la primera de todas es si la aplicación es fácil de usar, es decir, está claro en qué sección nos encontramos en la aplicación, recordemos que tenemos cuatro módulos (cada uno identificado con un color característico), si se pueden cambiar los parámetros de configuración sin problemas, el modo de edición de las actividades que lo posean es intuitivo y se sabe en todo momento lo que se tiene que hacer en la aplicación, con ayuda de iconos característicos o con títulos que guíen al usuario sobre la acción que se va a proceder a realizar.

En la sentencia se accede fácilmente al tipo de actividad que se quiere hacer, se pretende evaluar si es capaz de identificar el usuario, hacia qué módulo se tiene que dirigir, si éste quiere realizar una actividad artística, usar el comunicador o utilizar una actividad diferente a éstas. La cuestión es si los colores e indicaciones son suficientes para no perderse en la navegabilidad de la aplicación y llegar al objetivo propuesto en un principio, sin mera dificultad para el especialista.

En la instalación del dispositivo, se quiere saber si les resultaría fácil a los expertos instalar esta aplicación, junto a los elementos necesarios para su uso como son los drivers para Kinect y el framework XNA, y si serían capaces por sí mismos, de realizar la instalación sin ningún problema, aunque se cambiara el equipo o cambiaran de centro.

La personalización de actividades hace referencia, a aquellas actividades que se pueden modificar a voluntad para adaptarse a las necesidades tanto del docente como de los alumnos. El objetivo aquí es plantearse, si esa posible modificación que se puede realizar en ciertas actividades es cómoda para este docente o por el contrario prefiere dejar el ejercicio como viene por defecto, antes que personalizarlo a sus preferencias.

La interfaz gráfica, así como la forma de navegación e interacción propician que los especialistas se adapten en un corto período de tiempo a la aplicación

y puedan usarla rápidamente, sabiendo en cada instante lo que tienen que hacer y sobre todo, qué indicaciones les tienen que decir a sus alumnos para que puedan realizar el ejercicio de una manera correcta. Estos son los criterios que se pretenden evaluar en este último apartado de la usabilidad en KiNEEt.

Según la evaluación obtenida de cada uno de los especialistas, con respecto a la usabilidad de la aplicación se pueden obtener las siguientes conclusiones:

La característica que ha salido con peor evaluación, ha sido claramente la de instalación del dispositivo, debido a que puede resultar complicado, quizás para un docente realizar todos los procesos necesarios para poner en funcionamiento de manera autónoma la aplicación en otro equipo. Son demasiados pasos los que hay que seguir entre la instalación de la aplicación y la instalación propia del dispositivo Kinect, junto al framework que también es necesario instalar.

El acceso a las actividades ha sido el aspecto que ha obtenido mayor puntuación, debido a que se ha optado por diferenciar por colores cada uno de los cuatro módulos en los que está dividida la aplicación, estos colores son:

- Naranja → Coordinación
- Azul → Actividades
- Verde → Comunicador
- Violeta → Arte

Cada una de las pantallas relevantes a la actividad como puede ser la pantalla de configuración u opciones, está identificada con el color de fondo, que se asocia a cada actividad. De esta manera tanto el docente como el alumno, sabe perfectamente si se está realizando una actividad de coordinación o arte, por ejemplo.

Además de los colores también se han añadido unos letreros que aparecen cuando se sitúa el cursor en un icono, con el fin de saber, qué actividad se está seleccionando. Este recurso es muy útil en aquellos módulos que contienen diferentes actividades como puede ser el módulo titulado actividades o el de arte.

En lo referente a los diferentes aspectos, que tienen un nota media, se puede argumentar, que la facilidad de uso de la aplicación no ha obtenido mejor nota, en parte por el uso del control de la interfaz únicamente con Kinect, ya que aunque este sistema sea muy novedoso, puede resultar a veces más complejo acceder a ciertas partes de la pantalla, que si se usara un ratón convencional.

En cuanto a la personalización de actividades, parece ser que no ha habido gran dificultad en realizar esta acción, usando el dispositivo Kinect y se entiende cómo hay que realizar esta tarea.

Respecto al tiempo de adaptación, es normal que cuando se utilizan dispositivos que son novedosos y se está haciendo más referencia a los alumnos que a los docentes, se necesite un período de adaptación, ya que no están acostumbrados a utilizar esta clase de herramienta en clase. No es lo mismo, que se haya utilizado Kinect para otras actividades que no sean las planteadas en la aplicación, como empezar desde un principio a enseñarles cómo tienen que usar Kinect y que aprendan a asociar los movimientos que realizan con las acciones que aparecen en pantalla. Es una tarea que requiere tiempo y un uso constante de esta tecnología para poder adaptarse fácil y rápidamente a la herramienta.

### 5.1.2. Evaluación Uso Educativo

En este apartado se van a exponer los resultados obtenidos en el ámbito sobre uso educativo.

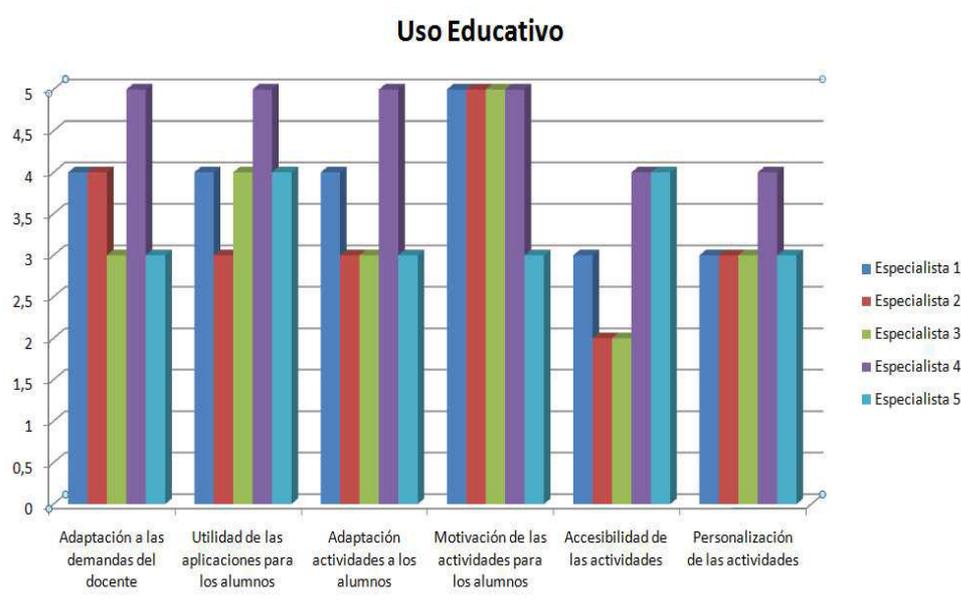


Figura 5.2: Evaluación Uso educativo

Adaptación a las demandas del docente  $\simeq$  Las actividades se adaptan a mis demandas como docente

Utilidad de las aplicaciones para los alumnos  $\simeq$  Las actividades son útiles para mis alumnas

Adaptación actividades a los alumnos  $\simeq$  Las actividades se adaptan a las necesidades de mis alumnas

Motivación de las actividades para los alumnos  $\simeq$  Las actividades son un recurso que utilizo por ser motivante

Accesibilidad de las actividades  $\simeq$  Las actividades son un recurso que utilizo por ser accesible

Personalización de las actividades  $\simeq$  Las actividades son un recurso que utilizo por ser personalizable

Un aspecto fundamental es que la aplicación les sirviera a las educadoras del centro en su labor cotidiana en el centro, para ello se han evaluado diferentes características enfocadas en la parte educativa de la aplicación y las demandas que pueden exigir los profesionales del centro.

Se pretende evaluar si parte de las necesidades que necesita el profesional en su labor docente, se complementa con la aplicación o por el contrario, las actividades que se encuentran en la aplicación no sirvieran en absoluto a las demandas que necesita para la formación de sus alumnos.

Se evalúan también aspectos básicos en cualquier actividad relacionada en el ámbito educativo, como puede ser que los ejercicios de la aplicación se puedan adaptar a los alumnos, dirigida la aplicación a un personal que presenta discapacidad y que pueden ser de diferentes tipos, en una misma clase, se considera imprescindible tener conocimiento, sobre si las actividades pueden ser utilizadas por los diferentes alumnos que son asignados a un docente. Además, se pretende valorar si las actividades resultan de utilidad para mejorar las habilidades tanto físicas como cognitivas.

Por otra parte, se considera necesario valorar, si las actividades del programa, incentivan la motivación del alumno, por la simple razón de que este es un hecho fundamental para que el alumno siga utilizando la aplicación y quiera volver a repetir la actividad por iniciativa propia y no tenga que ser el docente, quien le obligue a tener que realizar el ejercicio por su bienestar.

Según la evaluación obtenida de cada uno de los especialistas, con respecto al uso educativo de la aplicación se pueden obtener las siguientes conclusiones:

Se va a empezar a valorar la característica que ha tenido mayor aceptación entre los especialistas, que ha sido sin lugar a dudas la motivación que provoca estas actividades en los alumnos. No es sorprendente pensar que este

haya sido el factor dominante en el ámbito educativo, ya que al tratar con personas muy jóvenes, la tecnología causa en ellos un gran impacto, la juventud de nuestros días pasan muchas horas delante del ordenador y sino es con el ordenador gastan mucho de su tiempo en dispositivos móviles como los tablets o smartphones que han revolucionado la vida de las personas. Los alumnos les motivan también los retos nuevos y las cosas diferentes para evita la monotonía de usar siempre el mismo procedimiento para aprender los contenidos esenciales. Es por eso que Kinect ha supuesto un cambio no sólo significativo sino, sobre todo, positivo debido a que con el movimiento de su cuerpo pueden interactuar con una aplicación que reacciona a los que ellos hacen, así de una forma visual y auditiva se consigue estimular al alumno para realizar la actividad en forma de juego y que aprendan a la vez que se divierten.

La peor valorada ha sido la que ha hecho referencia a la accesibilidad de las actividades. La causa de esta valoración es debido a que es extremadamente difícil que las actividades puedan resultar útil o fáciles de realizar a todo tipo de afecciones que hay en el mundo de la discapacidad. Como ya se ha explicado, hay diferentes tipos de discapacidad física, mental, sensorial, etc. Adaptar las actividades para que puedan ser utilizadas por un alumno que tiene autismo, parálisis cerebral o discapacidad visual, no es lo mismo y requiere una atención especializada para diseñar el ejercicio para cada uno de ellos. Se pretende con este proyecto abarcar al máximo número de alumnos posibles, teniendo en cuenta que es realmente complejo, que pueda ser utilizado por cada uno de los usuarios del centro.

No ha sido inesperada que se obtenga una valoración bastante aceptable en las características que concierne a las demandas del docente y de los alumnos, puesto que las actividades que se han implementado en esta aplicación han sido elegidas específicamente por los especialistas del centro, en todo el proceso de desarrollo de este programa. Ellos han elegido algunas actividades que les podrían resultar útiles para utilizarlos con sus alumnos, teniendo en cuenta las características que el dispositivo Kinect era capaz de ofrecer. Además de probarse las actividades, con los alumnos del centro una vez que estaban finalizadas o en ocasiones en fase de pruebas, para ir corrigiendo posible defectos y conseguir que se adaptasen de la mejor manera posible a los alumnos de este centro.

### **5.1.3. Evaluación Período de tiempo**

Por último, en este apartado se pretende evaluar la duración que piensan los docentes que le van a dar a la aplicación. Para valorar el tiempo que ellos creen que les va a resultar útil la aplicación a sus respectivos alumnos, se les ha dado las siguientes opciones, para que valoren el espacio temporal que sus alumnos van a utilizarla:

- La aplicación va a ser utilizada durante varios cursos
- La aplicación va a ser utilizada durante un curso
- La aplicación va a ser utilizada más de un mes
- La aplicación va a ser utilizada durante un mes
- La aplicación va a ser utilizada durante una semana

### **5.1.4. Desarrollo y Resultados en el CEEE Princesa Sofía**

Desde el comienzo del proyecto se ha estado colaborando con el CEEE Princesa Sofía, para realizar un conjunto de actividades que les sirviera en su día a día, como docentes y también que les resultara útiles a sus alumnos en su proceso de formación y aprendizaje para convertirse en personas independientes y poder vivir su vida, como cualquier persona.

Los docentes conocían escasamente este dispositivo y no tenían conocimiento de que era posible trabajar con el mismo, en el aula. Una vez que se les enseñó una demostración de cómo funcionaba Kinect y que podía ser utilizado en cualquier ordenador que dispusiera de Windows 7, se tuvo un aspecto diferente de este dispositivo y la posible utilización que le podía ofrecer tanto a ellos como a sus alumnos, en su trabajo diario.

Los especialistas comentaron qué actividades les podía servir a ellos, teniendo en cuenta siempre que el sensor Kinect funciona con movimientos y no es necesaria la utilización de teclado y ratón, es un concepto totalmente diferente de interactuar con la aplicación, pero no sólo de interactuar con la aplicación, sino también de tener en cuenta en el proceso de desarrollo de este software.

La secuenciación de actividades fue como se va a comentar, empezando por actividades más bien de índole cognitivo, como la actividad de causa-efecto o de ordenación de números, pasando por la actividad en el que se trabaja el

aspecto físico, esta actividad es la de coordinación, realizando las actividades referentes al arte para terminar con un comunicador de peticiones.

El sensor Kinect era totalmente nuevo para ellos, con lo cual se requería un proceso de adaptación y aprendizaje propio de los alumnos, que sólo se consiguió haciendo que estos practicaran en un primer momento con las aplicaciones de muestra, que venían por defecto con el SDK de Kinect y después probando las distintas actividades que se iban desarrollando con el fin de que se fueran haciendo al modo de interacción del sensor Kinect, viéndolo como un objeto más del espacio de trabajo.

Aunque poco a poco se fue intentando introducir este dispositivo, había otra tecnología que estaba ya arraigada al centro y en un factor más decisivo para este proyecto, los alumnos. La metodología de uso de una aplicación orientada a pizarra digital es muy diferente al de una aplicación orientada con Kinect, puesto que para usar la pizarra digital es necesario tocar la pantalla con un bolígrafo diseñado especialmente para ella, mientras que con la tecnología Kinect es necesario estar a una distancia mínima para poder reconocer el cuerpo humano y tienes que interactuar con un cursor que aparecerá en tu mano o la parte del cuerpo seleccionada según la actividad.

Esta situación, nos dirige al paradigma de que hay dos diferencias considerables entre el uso de la pizarra digital y Kinect, la distancia y el hecho de que se necesite tener contacto con la pantalla. Como la aplicación orientada a Kinect era proyectada en la pizarra digital, para disponer de una pantalla de más dimensiones que un monitor corriente, y así que a los alumnos les resultara más fácil poder visualizar los elementos que componen la aplicación, surgía una confusión en los alumnos.

Esta confusión era que ellos pensaban que tenían que usar la aplicación como si fuera una aplicación de pizarra, con lo cual esto nos llevaba a la situación de que los alumnos no respetaban la distancia de 1.2m para usar Kinect, debido a que los alumnos iban directamente a la pantalla a tocar el objeto cuando el especialista le daba la instrucción adecuada para realizar correctamente la actividad.

Como solución a este problema, se puso una cinta de un color llamativo en el suelo, indicando al alumno que no podía cruzar esa línea y que durante el tiempo que permaneciera realizando la actividad tenía que estar en esa posición, pero esto no quiere decir que nada más poner la cinta la respuesta fue inmediata, ya que el alumno necesita un tiempo para asociar la cinta con la Kinect y hay niños que se les olvida fácilmente las pautas de la sesión anterior, obligando al docente a repetir las instrucciones de uso hasta que

logra acostumbrarse a esta nueva metodología para ellos.

Otro obstáculo importante que se encontró en este camino, fue el hecho de que el alumno identificase que era él, el que estaba interactuando con la aplicación y más significativo aún que fuera consciente de que con el movimiento de su cuerpo era capaz de realizar acciones que se reflejaban en las actividades que se le proponían para hacer. Realizando distintas pruebas con los alumnos, fue posible averiguar que los alumnos que tenían menos nivel necesitaban verse reflejados en la pantalla para poder completar la actividad y darse cuenta de sus movimientos. Por este motivo, hay aplicaciones del ámbito cognitivo que dan la opción de poder activar la cámara de color que viene integrada con el sensor Kinect para poder utilizarla en estas actividades, y así darle la oportunidad a este grupo de alumnos de poder realizar la actividad como los demás compañeros.

A pesar de todos los esfuerzos, había alumnos que les costaba más adaptarse y para poder realizar los ejercicios necesitaban la ayuda de su tutora para que mediante imitación se fueran acostumbrando a los movimientos que tenían que realizar y al sistema. El procedimiento que seguían los especialistas, era ponerse detrás del alumno y ayudarles a realizar el movimiento, pero esto provocaba que el dispositivo Kinect no tomará la decisión acertada y en vez de coger el cuerpo del alumno, cogía el cuerpo del docente, con lo que la consecuencia directa era elevar el nivel de realización de la actividad y disminuir el grado de fiabilidad de la misma. Con este nuevo inconveniente, se implementó un método que hacía que sólo se detectara el cuerpo que estuviera más cercano al dispositivo Kinect, provocando como resultado, que sólo se detectara el cuerpo del alumno, ignorando completamente al docente.

La adaptación a este nuevo sistema requirió de esfuerzo y tiempo, pero no sin resultados positivos, puesto que los alumnos estaban muy motivados con estas nuevas actividades y la metodología, ya que hacía que con movimientos sencillos obtuvieran una respuesta que ellos valoraban enormemente.

De hecho, había ejercicios para que movieran los brazos y así mejorar la movilidad en estos miembros y no se quedaran más atrofiados todavía, pero no querían hacerlos y suponía una gran dificultad para los especialistas. Sin embargo, con algunos de los ejercicios de KiNEEt consiguieron mover sus extremidades y hacer que se esforzaran para lograr el objetivo propuesto, ya que estaban motivados y los feedback que estaban asociados a cada uno de los ejercicios los alentaba para que siguieran intentándolo, aunque la actividad fuera más difícil y les costara mucho más esfuerzo realizarla. Los alumnos estaban muy motivados por usar el dispositivo Kinect pasar un rato disfrutando de esta tecnología, sin darse cuenta de que estaban desarrollando



Cuadro 5.1: Alumnos del CEEE Princesa Sofía usando KiNEEt

habilidades que antes eran inapreciables no sólo para ellos, sino para el resto de personas que los observábamos con detenimiento.

## 5.2. Estudio de los alumnos

Se ha realizado una valoración de los alumnos con los que se ha estado probando la aplicación KiNEEt para obtener mejoras y posibles cambios a realizar en ésta, comprobando que le pueda ser de utilidad a los alumnos del CEEE Provincial Princesa Sofía. Se ha elegido a conciencia una serie de alumnos, cada uno con unas características distintas, para abordar los diferentes tipos de discapacidad que existen. Éstas son:

- Discapacidad sensorial.
- Discapacidad física.
- Discapacidad intelectual.
- Trastornos de conducta.

Para englobar todas estas categorías, se ha hecho el estudio con una pequeña muestra de alumnos con las siguientes características:

- Alumno con trastorno autista.



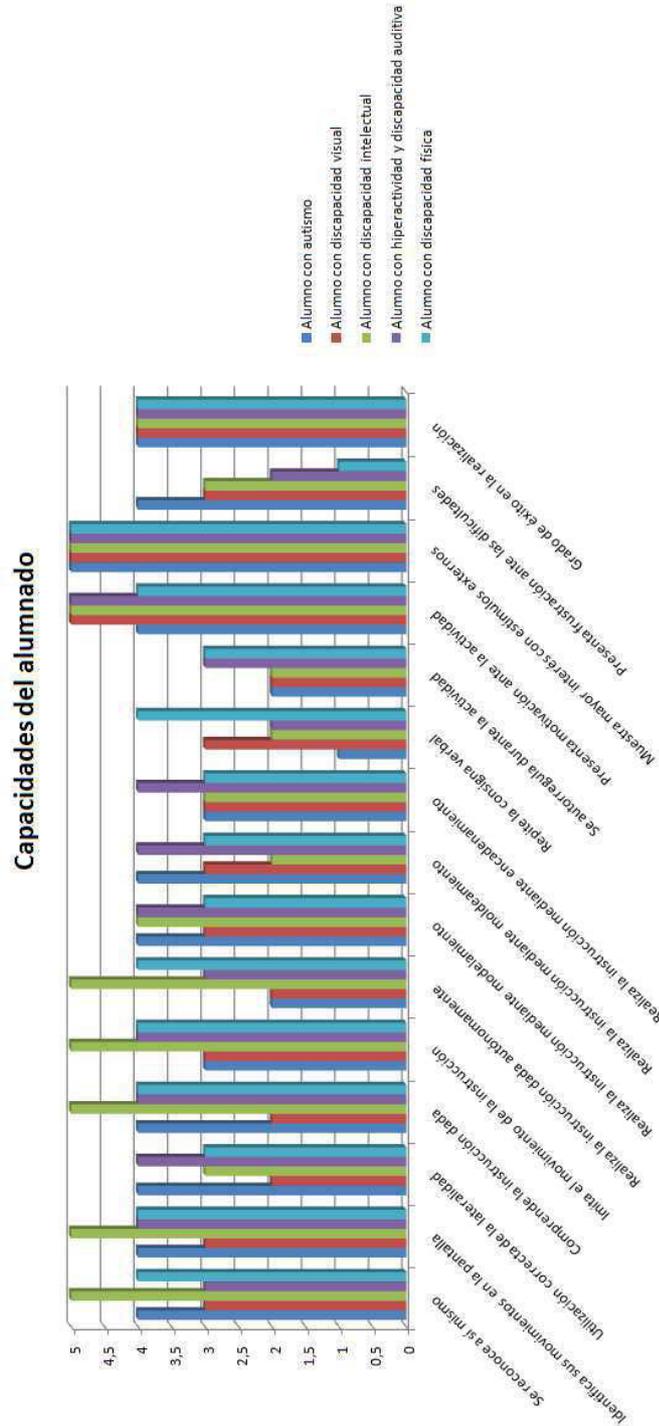


Figura 5.3: Evaluación Capacidades del Alumnado

## 5.2. ESTUDIO DE LOS ALUMNOS

---

La evaluación del alumnado pretende ser una medida que indique cómo están respondiendo los alumnos a la aplicación y al uso del dispositivo Kinect como herramienta complementaria en su formación. Este cuestionario también puede ayudar a averiguar si un alumno es apto para usar la aplicación y el sensor Kinect, puesto que las características de un individuo pueden hacer que este no sea el método más adecuado para su desarrollo.

En este estudio se valoran una serie de características relevantes, como si es capaz de identificarse a sí mismo o los movimientos que realiza, ya que si un alumno no tiene la capacidad de asociar que con el movimiento de su mano puede mover el cursor de la pantalla, se debería reflexionar la causa, ésta puede ser simplemente que el alumno necesite más práctica con la aplicación o que el alumno no sea apto para añadir a su aprendizaje esta herramienta.

Es de vital importancia que el alumno comprenda las instrucciones que se le faciliten al realizar la actividad, puesto que es una metodología totalmente nueva y se necesita que el alumno sea capaz de realizar la instrucción que se le proporciona con el fin de que pueda completar con éxito las actividades.

Otro aspecto a tener en cuenta es saber si son capaces de realizar los ejercicios independientemente o por el contrario se necesita la ayuda de los docentes, como modelamiento, donde el alumno repite el mismo movimiento que realiza el tutor o necesita otro tipo de ayuda, como moldeamiento, que consiste en que el tutor tiene que hacerle el movimiento al alumno, por ejemplo, cogiéndole del brazo para que éste complete la actividad. Esta información puede ser de bastante utilidad para elegir el nivel de dificultad que hay que configurar en el ejercicio y también para saber cuando el alumno puede avanzar al siguiente nivel.

Se ha tratado en el transcurso de este documento, la importancia de la motivación en la realización de las actividades, con lo cual se pretende valorar cómo sería la motivación respecto a un elemento fundamental en este aspecto, como es el feedback. Existe una diferencia notable cuando el alumno recibe un estímulo por su trabajo y cuando no se recompensa adecuadamente.

Las conclusiones que se pueden extraer de esta evaluación son:

Los alumnos presentan mucha motivación por el uso del dispositivo Kinect y de esta aplicación, que a pesar de ser una nueva metodología de trabajo y de interacción con las tecnologías, no ha supuesto un problema para que los niños realicen cada una de las actividades, a la vez que se divierten. Aunque todas las actividades tienen un feedback para alentar y reforzar la actuación del alumno en el ejercicio, es una realidad, que las actividades tienen un tipo diferente de feedback. Algunas actividades sólo poseen un estímulo visual o

un feedback auditivo, mientras que otras actividades poseen ambos modos de feedback, lo que queda claro de la realimentación suministrada por el sistema, es que si posee ambos refuerzos tanto visual como auditivo, el alumno se siente más motivado y estimulado en la realización de las distintas tareas.

Otro aspecto positivo es que los alumnos son capaces de completar con éxito las actividades propuestas sin excesivos problemas, aunque puedan presentar frustración cuando les surge alguna dificultad en la actividad, que les cuesta solventar. El alumno con autismo ha sido el que mayor frustración ha presentado, cuando se ha encontrado con alguna dificultad.

Los alumnos se tienen que adaptar a una práctica que no es habitual en la metodología de enseñanza que se lleva realizando en el centro, lo que supone normal que se les tenga que orientar en la interacción con la aplicación, diciéndole una serie de instrucciones que tienen que realizar o con otro tipo de apoyo, como el modelamiento o moldeamiento comentado anteriormente en esta misma sección.

Estos alumnos en concreto, son capaces de realizar la mayoría de las actividades imitando el movimiento que hace el docente para que comprendan qué acción tienen que realizar en el ejercicio, aunque hay algún alumno como el que posee una discapacidad auditiva e hiperactividad que sí necesita un poco más de ayuda y hay que recurrir en ocasiones al moldeamiento. A pesar de que los alumnos necesiten más o menos ayuda, son capaces de identificarse a sí mismos en la aplicación y sobre todo, identificar que sus movimientos tienen un efecto en la aplicación. Es fundamental que estas condiciones se cumplan a la perfección, porque sino provocaría un grado de dificultad muy elevado, incluso la imposibilidad de completar los ejercicios propuestos por el especialista.

Aunque estos alumnos sean capaces de identificarse a sí mismos, es cierto que cuando se le dice una instrucción, no tienden a repetirla verbalmente, para comprobar que efectivamente han comprendido lo que tienen que hacer. Además, no son capaces en líneas generales de autorregularse en la actividad, es decir, de corregir su comportamiento cuando se aprecia que están realizando el ejercicio de manera incorrecta, lo que provoca que el tutor tenga que repetir las instrucciones varias veces en el transcurso de la actividad.

# Capítulo 6

## Conclusiones y Trabajos Futuros

En primer lugar, se va a enumerar los medios, en los que la aplicación desarrollada en este proyecto, ha sido presentada durante este período de tiempo.

A continuación, se va a exponer las conclusiones que se han obtenido en la realización de este proyecto, comprobando si se han cumplido los objetivos que se habían propuesto en un principio del proyecto.

Por último, una vez hecho el análisis sobre el estado final del proyecto, se proponen una serie de aspectos que se pueden seguir desarrollando en esta aplicación con el objetivo de poder ofrecer una continuidad en esta línea para proseguir con el trabajo ya realizado.

### 6.1. Difusión

En esta sección se quiere plasmar en qué medios se ha presentado la aplicación, durante la realización de este proyecto.



Programa Solidarios, Canal Sur. 14/10/2013



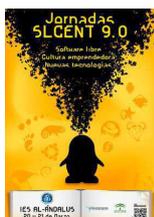
Seminario Internacional de Arte Inclusivo. 27/11 - 29/11/2013



Programa Canal Sur a la Carta. Jornadas de Inclusión Programando Videojuegos. 29/10/2013



Encuentro Arte + Colombia. 26/09/2013



Jornadas SLCENT 21/03/2013



Semana de las Ciencias. Universidad de Almería. 07/11/2013



Jornadas de Informática. Universidad de Almería. 07/03/2014



X CONGRESO INTERNACIONAL Educación Inclusiva: Desafíos y respuestas creativas. Universidad Zaragoza. 19/03/2013



Programme INN-ASNE (Innovative Approaches in Special Needs Education) 03/03 - 16/03/2013.



Asociación Asprodalba



Asociación Asacal



Asociación Capacitarte

## 6.2. CONCLUSIONES

---



JUNTA DE ANDALUCÍA  
CEEE Princesa  
Sofía

Jornadas Aprende con Kinect en Junio 2013



JUNTA DE ANDALUCÍA  
CEEE Princesa  
Sofía

Colaboración en el Grupo de Trabajo del CEP, denominado KINECT. Curso 2012 - 2013



JUNTA DE ANDALUCÍA  
CEEE Princesa  
Sofía

Colaboración en el Grupo de Trabajo del CEP, denominado Uso didáctico de la kinect y otros materiales digitales para el alumnado con discapacidad. Curso 2013 - 2014



JUNTA DE ANDALUCÍA  
CEEE Princesa  
Sofía

Proyecto de Investigación aprobado para el curso académico 2014 - 2015

Cuadro 6.1:

## 6.2. Conclusiones

Esta aplicación cuenta con una serie de ejercicios con finalidades totalmente distintas, divididas fundamentalmente en cuatro módulos:

- **Coordinación:** Enfocada a la mejora física del individuo, con un ejercicio que el docente puede personalizar según estime oportuno, según las necesidades del alumno.
- **Actividades:** Tratan sobre todo el tema cognitivo, aunque algunas también sirven para mejora de la funcionalidad motora del individuo.
- **Arte:** Este módulo contiene actividades sobre música, danza y pintura, las cuales permiten desarrollar habilidades como la creatividad, confianza, comunicación no verbal y colaboración, entre otras. Un aporte fundamental para no centrarse sólo en las cualidades obvias de toda

persona como el nivel cognitivo o físico, que aunque son fundamentales no son los únicos.

- Comunicador: Esta herramienta hace realidad la idea de que las personas que tengan dificultad para comunicarse por diversos motivos, como el autismo o trastornos neurológicos, sean capaces de expresar gráficamente lo que no son capaces de decir con palabras.

Las distintas actividades que abarca el proyecto, son suficientes para que el alumno desarrolle y mejore una serie de cualidades que no sólo le servirán en el ámbito académico, sino más importante todavía, es que le servirá en su día a día, para formarse como persona y que le ayuden en su convivencia con el resto de personas y con la sociedad.

Uno de los aspectos que se ha recalcado en el proyecto ha sido cómo la falta de motivación puede ser un elemento esencial para que la aplicación no tenga el efecto esperado, que no es otro que el alumno se encuentre cómodo en el sistema y realice los ejercicios de una forma divertida, cumpliendo el objetivo que se perseguía con la creación de este proyecto, que era el desarrollo de habilidades tanto cognitivas como físicas. Para conseguir que el individuo no abandone las actividades y se aburra durante la realización de las mismas se ha usado no sólo la tecnología propia para desarrollar un videojuego, sino también la mentalidad. Con esta línea clara desde el principio, se ha contribuido a que mediante el juego el niño aprenda, a la vez que se divierte. Otro elemento desencadenante para este propósito, es la inclusión del término feedback en cada uno de los ejercicios, con el fin de que el alumno no sólo sea capaz de reconocer cuándo ha realizado correctamente el ejercicio, sino que le sirva de estimulante visual y/o auditivo como recompensa del esfuerzo realizado y produzca un aumento de autoestima.

Con el objetivo propuesto de que había que intentar que el número de usuarios con discapacidad fuera lo mayor posible, se ha optado porque los ejercicios posean su sección de configuración. Pretendiendo con la configuración, que las actividades se acomoden a distintos tipos de alumnos dependiendo de la discapacidad presentada, se ha obtenido un medio para que los ejercicios se adapten al alumno y no al contrario, permitiendo incluso el tema de la progresión, puesto que en algunas de las actividades existen niveles de dificultad, así que aparte de la adaptación está incluida también la progresión.

Es agradable observar el hecho de que los niños puedan utilizar la tecnología para sus fines, sin importar para nada su cuerpo, ni limitaciones que éste les pueda proporcionar. Como usando herramientas tan novedosas y versátiles se puede cambiar un concepto tan básico y ancestral como es la educación, por

este motivo partiendo de la premisa de que tanto la educación como el propio bienestar del individuo son indispensables para una vida agraciada. Por ello con ayuda del sensor Kinect se ha conseguido no sólo que cambien su estilo de realizar ciertas tareas o incorporar otras, sino también la mentalidad pero de todos, materializando una idea que puede cambiar la perspectiva de cada uno de nosotros, el hecho de que el movimiento de nuestro cuerpo cogida de la mano de la tecnología pueda hacer aprender, soñar y disfrutar.

## 6.3. Trabajos Futuros

En esta sección se van a proponer propuestas que se pueden seguir desarrollando e investigando para dotar de mayor funcionalidad a esta aplicación.

- El comunicador puede ser una herramienta que se puede explotar mucho más, incluso cabe la posibilidad de pensar en separarlo y hacerlo independiente, como un proyecto para él solo, debido a que puede abarcar otras características que no se han propuesto en este desarrollo. Se podría aumentar el número de gestos reconocidos y en vez de enfocarlo sólo a peticiones, se podrían plasmar otras circunstancias o incluso sentimientos que el alumno desee expresar, abarcando así no sólo el aspecto de los gesto sino la temática del comunicador en sí. Además, se puede proporcionar la posibilidad de que el docente pueda crear sus propios tableros, escogiendo los pictogramas y el tema que más le interese según el tipo de alumno que lo vaya a utilizar.
- En estos días, que los dispositivos móviles están cambiando la forma de comunicarse y se han convertido en una herramienta fundamental en el día a día de las personas. Se podría hacer una aplicación complementaria para dispositivos móviles, la cual permita modificar aspectos de las actividades, desde cualquier parte del mundo, ya que se está por la labro de eliminar las barreras, se puede pretender eliminar la barrera de la distancia, y así poder interactuar con la aplicación desde cualquier lugar y si pensamos un poco más allá desde cualquier parte del mundo, sin restricciones.
- En este proyecto se han abordado un número limitado de actividades, pero que pueden ser ampliadas. Un ejemplo de esto puede ser incorporar una actividad de teatro y así no sólo tener más variedad de ejercicios sino hacer que las actividades sean más colaborativas y puedan participar más individuos en la realización del ejercicio.

- La incorporación del nuevo sensor Kinect desarrollado para XBOX One, y poder realizar actividades que incorporen hasta 6 individuos en el ejercicio, de este modo también se puede trabajar la convivencia y aspectos relacionados con el trabajo en equipo, requisitos fundamentales para mantener buenas relaciones sociales. Además de poder trabajar no sólo con el esqueleto del individuo, sino también con los músculos. lo que permite por ejemplo saber si está haciendo más fuerza con la pierna izquierda que con la derecha al apoyar y corregir malos hábitos posturales.
- En el tema de las TIC y educación, además del diseño universal, otro factor que está en auge es el concepto de inteligencia ambiental, con sus características de ubicuidad, transparencia e inteligencia, promoviendo que los alumnos puedan acceder a la enseñanza desde cualquier lugar, siendo los medios utilizados para este fin imperceptibles para ellos, adaptándose estos recursos a las necesidades de cada uno. La cuestión sería orientar la aplicación para que estuviera involucrada en este modelo de inteligencia ambiental.
- Por último, se podrían incluir perfiles de los alumnos que usan la aplicación, para que sea posible evaluar de manera automática los progresos del usuario y realizar estadísticas con estos datos que faciliten de manera visual la labor de evaluación.

# Bibliografía

- [1] Comunicación de la Comisión al Parlamento Europeo, al Consejo, al Comité Económico y Social Europeo y al Comité de las Regiones - Estrategia Europea sobre Discapacidad 2010-2020: un compromiso renovado para una Europa sin barreras.
- [2] Quintero, A. et al.2012.Experiencias en Artes Interactivas Física- Música – Lingüística – Movimiento corporal. Proceedings of ECIMAG.
- [3] Miles, R.2012. Start Here!: Learn the Kinect API. S4CarlislePublishingServices. Microsoft Press Book.250p.
- [4] Weitzman, M.2005.Terapias de Rehabilitación en Niños con o en riesgo de Parálisis Cerebral. Revista Pediatría Electrónica.2.
- [5] Herrera, G. et al.2012. Pictogram Room: Natural Interaction Technologies to Aid in the Development of Children with Autism. Annuary of Clinical and Health Psychology. 8: 39-44.
- [6] Murillo, A. and Montserrat, M.2012.TÍTULO DEL TFC: Aplicación telemática para uso terapéutico utilizando Kinect para Windows y SDK v. 1.6.Tesis Ingeniería Técnica de Telecomunicación, especialidad en Telemática.Barcelona.España.Universidad Politécnica de Catalunya.59.
- [7] VIRTUALWARE GROUP.2014.Consultado 13 de Febrero de 2014.<http://virtualrehab.info/es/>
- [8] Eusko Jaurlaritz.2011.Consultado 13 de Febrero de 2014.[http://www.osakidetza.euskadi.net/r85-ckcmpn05/es/contenidos/informacion/teki/es\\_teki/teki.html](http://www.osakidetza.euskadi.net/r85-ckcmpn05/es/contenidos/informacion/teki/es_teki/teki.html)
- [9] FIVAN.2012.Consultado 13 de Febrero de 2014.<http://www.neuroathome.org/2013/10/neuroathome-plataforma-rehabilitacion-virtual-kinect.html>

- 
- [10] Grupo de Investigación Interacción, Tecnología y Educación (i-TED).2012.Consultado 13 de Febrero de 2014.<http://tangoh.iter.es/>
- [11] Castellucci, J. and MacKenzie, I.2008. Unigest: Text Entry Using Three Degrees of Motion. CHI '08 Extended Abstracts on Human Factors in Computing Systems. 3549-3554.
- [12] Abhijit, J.2012. Kinect for Windows SDK Programming Guide: Build motion-sensing applications with Microsoft's Kinect for Windows SDK quickly and easily. Packt Publishing Ltd.366p.
- [13] Catuhe, D.2012. Programming with the Kinect for Windows Software Development Kit. Megan Smith-Creed. Microsoft Press Book.207p.
- [14] Webb, J. and Ashley, J.2012. Beginning Kinect Programming with the Microsoft Kinect SDK. Apress Media LLC.306p.
- [15] Yongpradit, P.2012.Fun, Phone, and the Future: Microsoft XNA Game Studio, Windows Phone, and Kinect SDK.Proceedings of the 43rd ACM Technical Symposium on Computer Science Education.321-322.
- [16] Pheatt, C. and McMullen, J.2012.Programming for the Xbox Kinect Sensor: Tutorial Presentation. Journal of Computing Sciences in Colleges.27:140-141.
- [17] Monir, S. et al.2012.Rotation and scale invariant posture recognition using Microsoft Kinect skeletal tracking feature.Intelligent Systems Design and Applications (ISDA),12th International Conference.404-409.
- [18] Hinchman, W.2011.Vectorform Labs.Consultado 04 de Febrero de 2014.<http://labs.vectorform.com/2011/06/windows-kinect-sdk-vs-openni-2/>
- [19] Fernández, E.2012. Control de Software Educativo Mediante Kinect de Microsoft. Tesis Grado Informática. Madrid, España. Universidad Carlos III de Madrid.115p.
- [20] Reed, A.2011. Learning XNA 4.0: Game Development for the PC, XBOX 360 and Windows Phone 7. O'Reilly Media, Inc.515p.
- [21] Miller, T. and Johnson, D.2011.XNA Game Studio 4.0 Programming: Developing for Windows Phone 7 and Xbox 360. Pearson Education, Inc.505p.
- [22] Thorn A.2014. UNITY 4 FUNDAMENTALS. Focal Press.295p.

## BIBLIOGRAFÍA

---

- [23] Norton T.2013.Learning C# by Developing Games with Unity 3D Beginner's Guide.Packt Publishing Ltd.292p.
- [24] Blackman S.2013.Beginning 3D Game Development with Unity 4: All-in-One, Multi-Platform Game Development. Apress Media LLC.778p.
- [25] Arlow, J. and Neustadt, I.2005. UML 2. Anaya Multimedia.608p.
- [26] Sommerville, I.2005.Ingeniería del Software.7 ed. Pearson Addison Wesley.687p.
- [27] Debrauwer, L.2012.Patrones de diseño para C#: Los 23 modelos de diseño: descripción y soluciones ilustradas en UML 2 y C#.Ediciones ENI.359p.
- [28] Marcos, M.2001.HCI (Human computer interaction): concepto y desarrollo.El profesional de la información.10:4-16.
- [29] Fundación Sidar.2007.Consultado 16 de febrero de 2014.<http://www.sidar.org/recur/desdi/usable/dudt.php>
- [30] Consultado 16 de febrero de 2014.<http://ares.cnice.mec.es/informes/17/contenido/4.htm>
- [31] Consultado 16 de febrero de 2014.[http://www.quadernsdigitals.net/datos\\_web/hemeroteca/r\\_1/nr\\_773/a\\_10443/10443.html](http://www.quadernsdigitals.net/datos_web/hemeroteca/r_1/nr_773/a_10443/10443.html)
- [32] Cook, D et al.2009.Ambient intelligence: Technologies, applications, and opportunities.Pervasive and Mobile Computing.5:277-298.
- [33] Ramos, C.2008.Ambient Intelligence—the Next Step for Artificial Intelligence.Intelligent Systems, IEEE.23:15-18.



# Apéndice A

## Manual de instalación

### A.1. Instalar SDK para Kinect

1. Descargar el ejecutable necesario para la instalación:

`http://www.microsoft.com/en-us/kinectforwindowsdev/start.aspx`

2. En la página, pulsar sobre el enlace “Download Latest SDK”

Downloads / Docs All Downloads / Docs

### Software Development Kit (SDK)

The Kinect for Windows SDK enables you to use C++, C#, or Visual Basic to create applications and experiences that support gesture and voice recognition by using the Kinect for Windows sensor and a computer or embedded device. The Developer Toolkit contains additional resources, sample applications with full source code Kinect Studio, and other resources to simplify and speed up application development.

Kinect for Windows SDK	Kinect for Windows Developer Toolkit
The download includes drivers for using the Kinect for Windows sensor on a computer running Windows 8, Windows 7, or Windows Embedded Standard 7. Also included are APIs and device interfaces.	The download includes Human Interface Guidelines, sample applications (with full source code), Kinect Studio, Face Tracking SDK, and other resources.
2013-09-17T00:00:00, 222 MB, English	2013-09-17, 384 MB, English
<a href="#">Download Latest SDK</a>	<a href="#">Download Latest Toolkit</a>

Figura A.1: Descargar SDK Kinect

3. Una vez descargado, hacemos doble click sobre el ejecutable y aparecerá la siguiente pantalla, en la cual debemos hacer dos acciones: a. Aceptar los términos y condiciones de licencia b. Pulsar sobre el icono de instalar

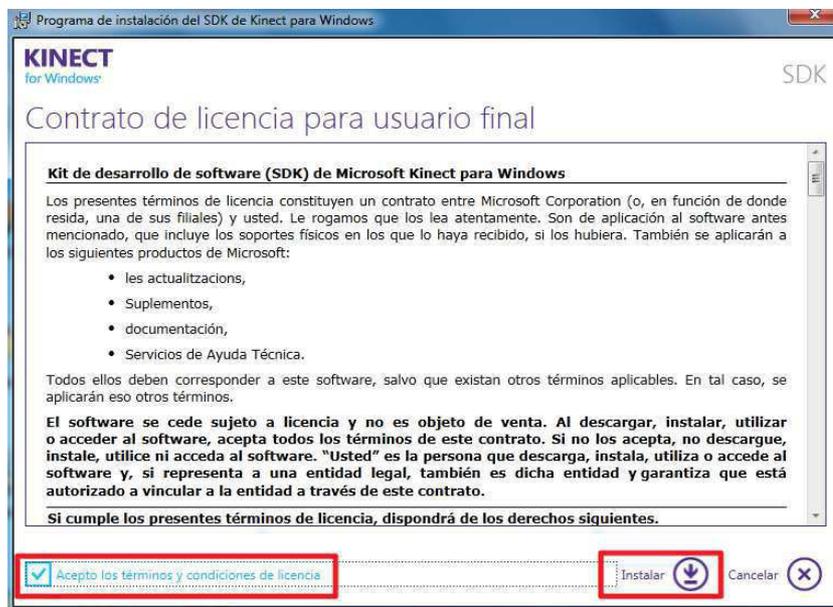


Figura A.2: Instalación SDK

4. A continuación aparecerá el progreso de instalación.

## A.1. INSTALAR SDK PARA KINECT

---

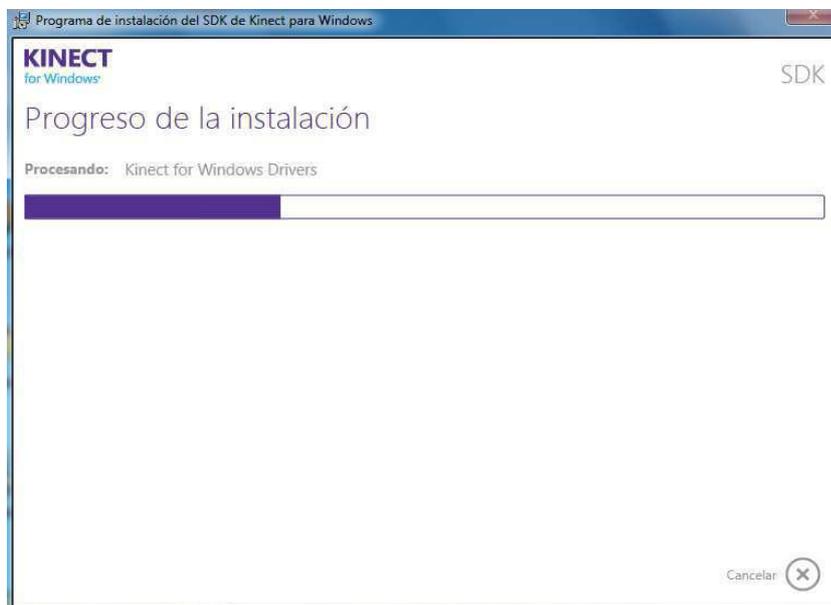


Figura A.3: Instalación SDK

5. Por último, informará que la instalación se ha completado satisfactoriamente.

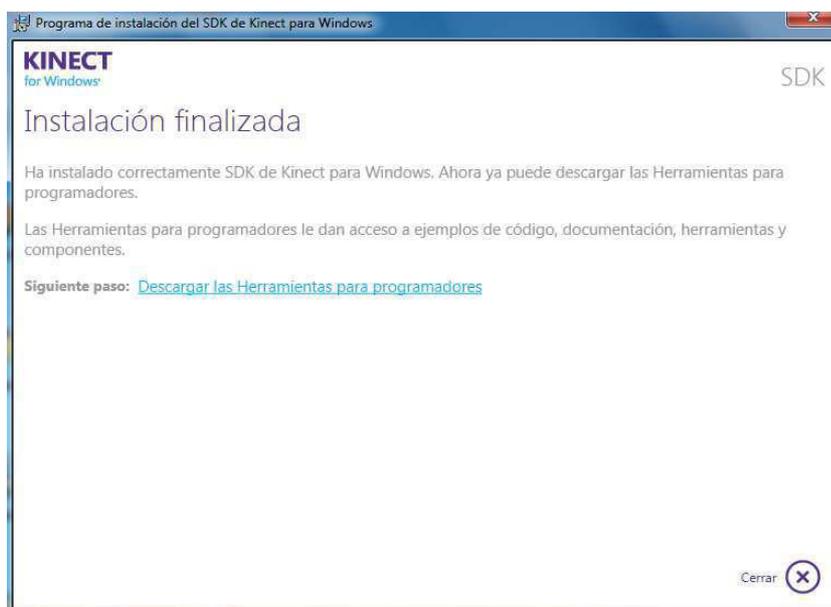


Figura A.4: Instalación SDK

## A.2. Instalar XNA Framework Redistributable 4.0

1. Visitar el siguiente enlace: <http://www.microsoft.com/en-us/download/details.aspx?id=20914>
2. Pulsar sobre el botón “Download”:

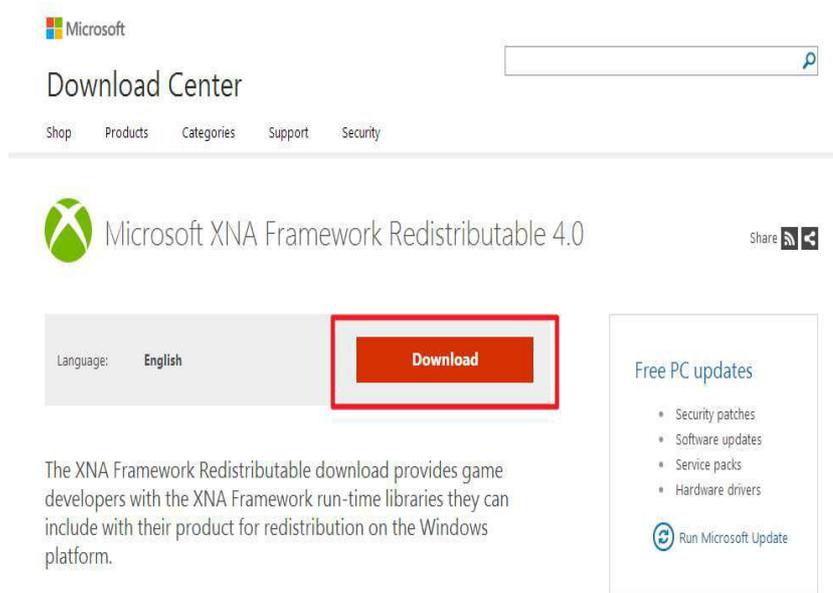


Figura A.5: Descargar XNA

3. Una vez hecho doble click sobre el instalador, se abrirá el asistente de instalación, en el cual se deberá aceptar los términos de licencia y pulsar sobre el botón Next para acceder al siguiente paso.

## A.2. INSTALAR XNA FRAMEWORK REDISTRIBUTABLE 4.0

---

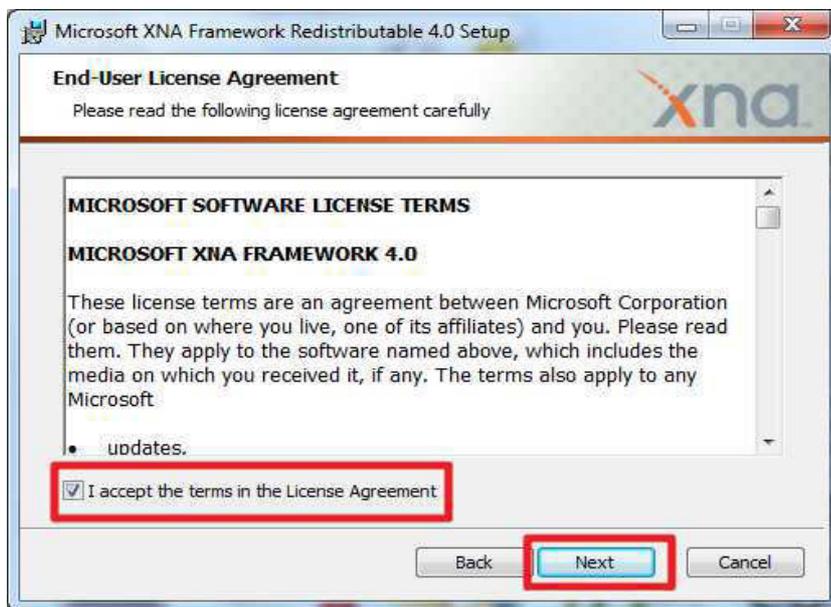


Figura A.6: Instalación XNA

4. A continuación, pulsar sobre el botón Install.

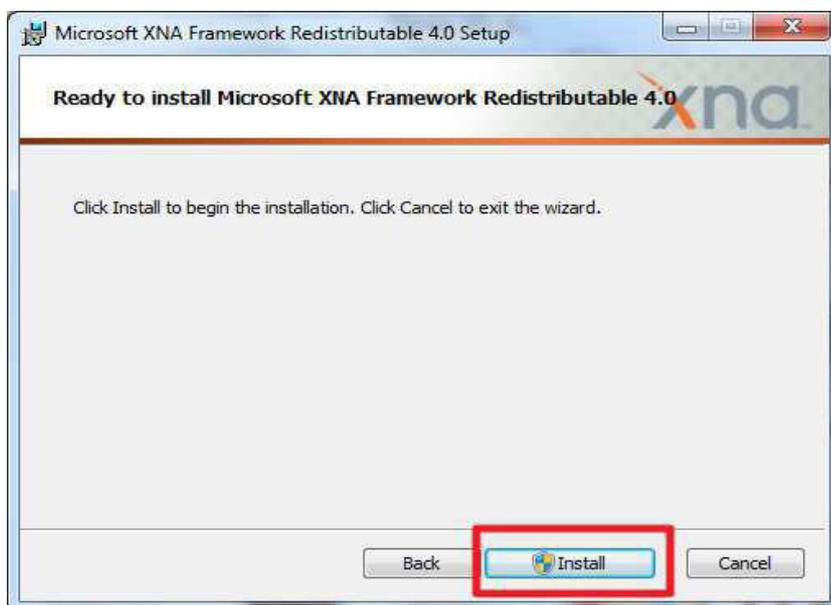


Figura A.7: Instalación XNA

5. Comenzará la instalación del programa.

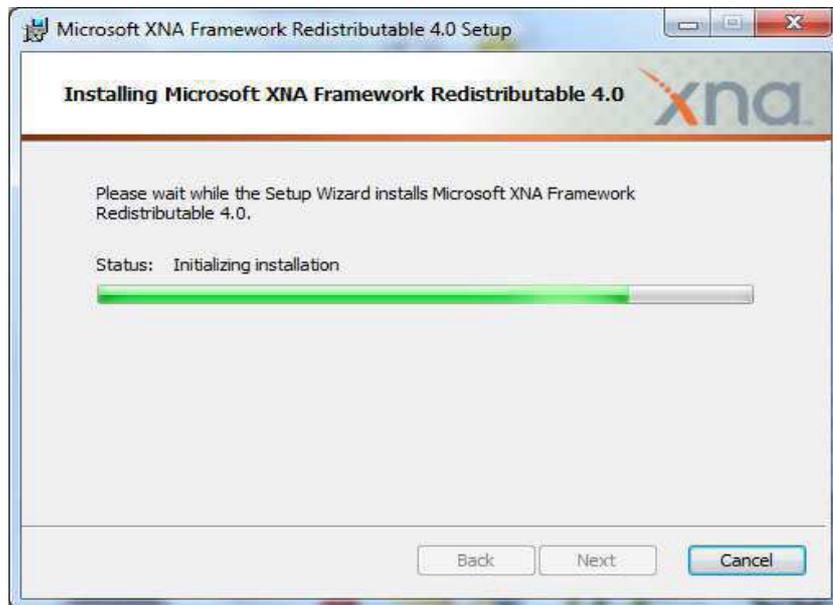


Figura A.8: Instalación XNA

### A.3. Aplicación KiNEEt

1. Copiar archivo Proyecto KiNEEt.rar en la carpeta C:

### A.3. APLICACIÓN KINEET

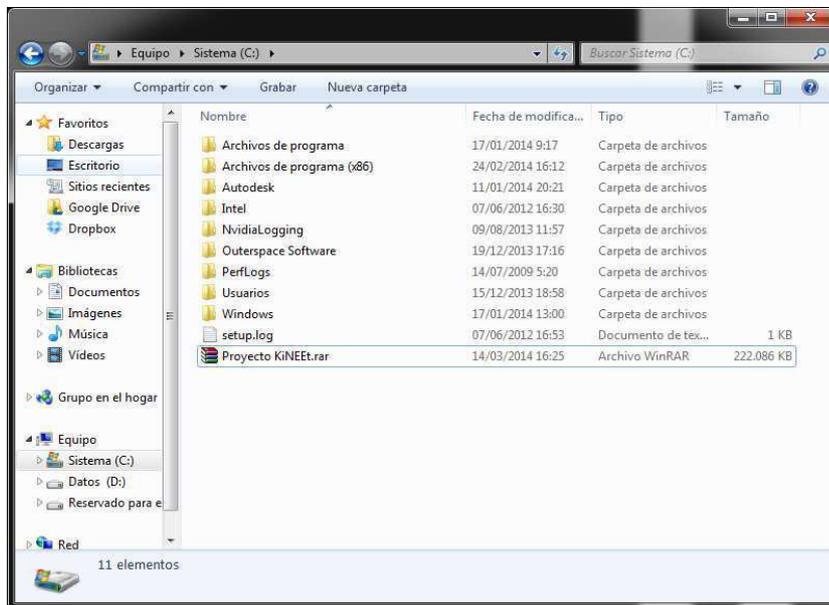


Figura A.9: Copiar archivo

### 2. Descomprimir archivo Proyecto KiNEEt.rar

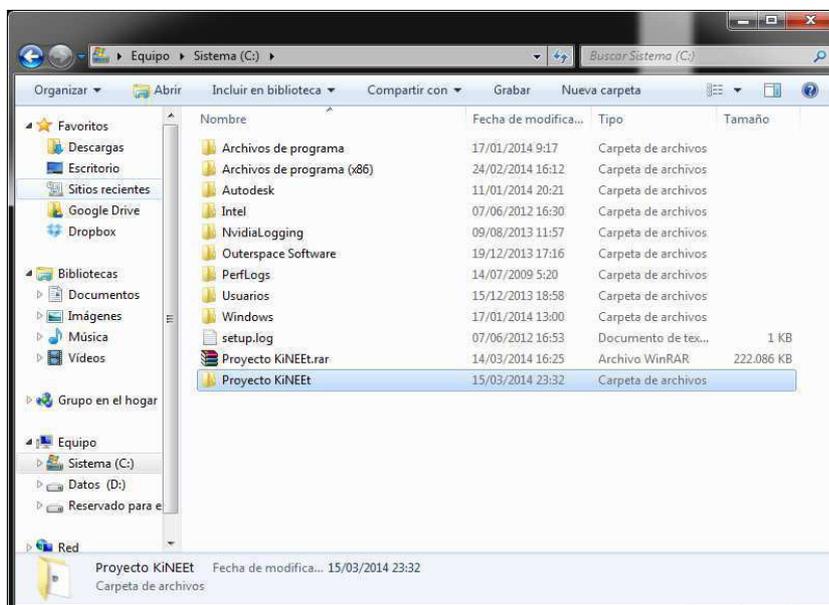


Figura A.10: Descomprimir archivo

### 3. Instalar voz

## APÉNDICE A. MANUAL DE INSTALACIÓN

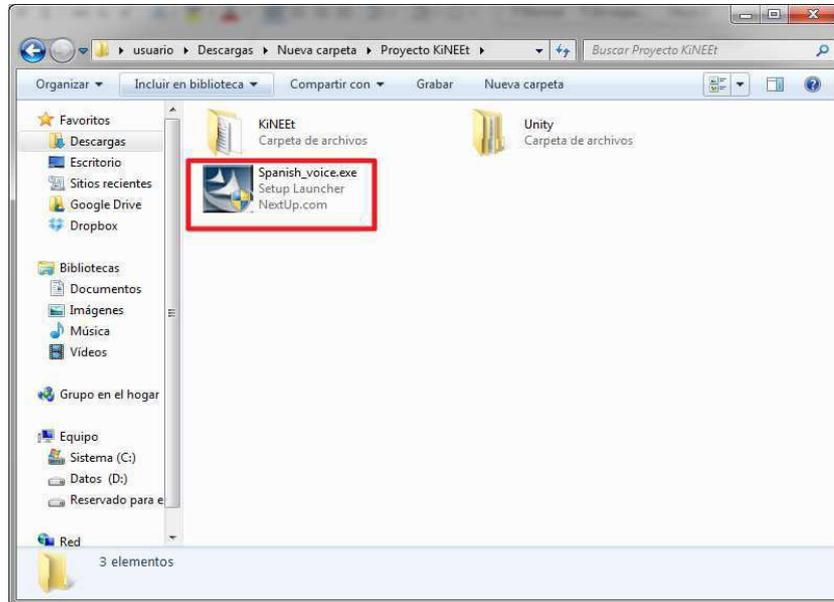


Figura A.11: Instalar voz

### 4. Entrar en la carpeta KiNEEt

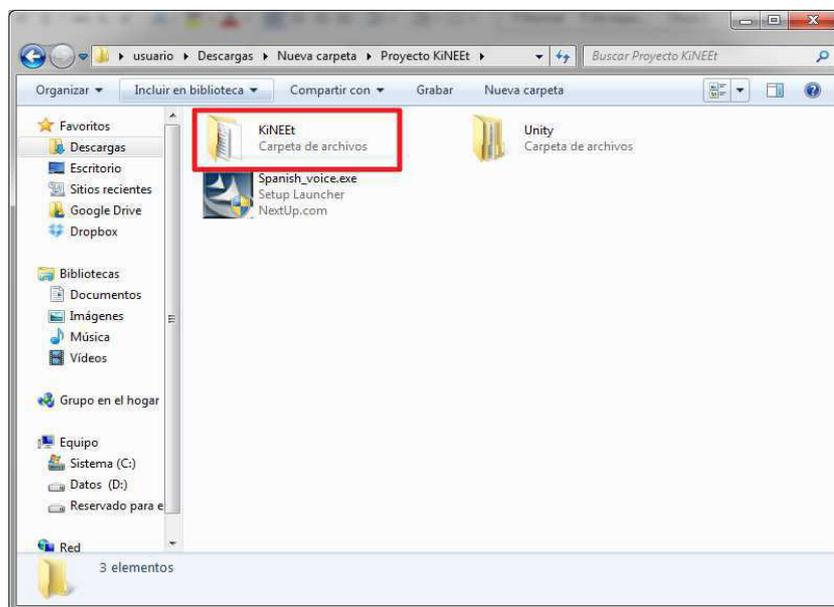


Figura A.12: Carpeta KiNEEt

### 5. Hacer doble click en Kineet.exe

### A.3. APLICACIÓN KINEET

---

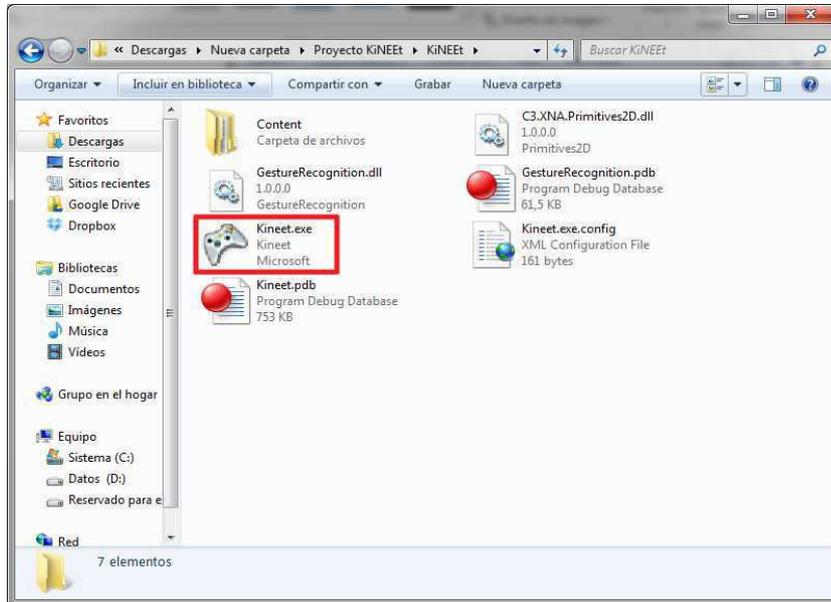


Figura A.13: Doble click en Kineet.exe



# Apéndice B

## Manual de usuario

Al iniciar la aplicación aparecerá la pantalla “splash screen” Figura B.1, que es la pantalla de presentación de este programa. Pulsamos la tecla Enter del teclado para empezar a usar la aplicación.



Figura B.1: Splash Screen

Antes de empezar a usar la aplicación, se van introducir algunas secciones, donde se explican elementos que son comunes en todo el programa, con el fin de ir conociendo un poco más la dinámica de este proyecto y no tener que estar repitiendo vanamente, aspectos innecesarios.

## B.1. Elementos comunes

### B.1.1. Iconos estándar

En este apartado se van a explicar una serie de iconos que tienen la misma función, en cualquier pantalla en la que se encuentre el usuario.

#### Inicio



Figura B.2: Icono inicio

Este icono vuelve a la pantalla de inicio (Ver Figura B.7).

#### Aceptar



Figura B.3: Icono aceptar

Cuando se pulsa este icono, el usuario consiente que está conforme con los cambios efectuados y está listo para continuar con la siguiente pantalla.

#### Reiniciar



Figura B.4: Icono reiniciar

Este icono reinicia el último ejercicio que se haya ejecutado. Aparece sólo en la pantalla de opciones de una actividad determinada.

## Salir



Figura B.5: Icono salir

Este icono permite salir de la aplicación.

### B.1.2. Pantalla de opciones

La pantalla de opciones (Ver Figura B.6 tiene la misma estructura independiente del ejercicio, lo único que cambia es el color de fondo dependiendo de la actividad elegida, para saber más de los colores asociados a cada módulo de la aplicación ver sección B.2



Figura B.6: Pantalla de opciones

## B.2. Pantalla de Inicio

En esta pantalla (Ver Figura B.7) puedes elegir entre uno de los cuatro módulos que está dividida la aplicación, según el tipo de actividad que se quiera realizar. En toda la aplicación se usa Kinect, con lo cual cuando se diga seleccionar en las siguientes líneas de este manual se referirá a que con tu mano derecha, puedes mover la mano que aparece en pantalla y tendrás que situarla encima de los iconos que irán apareciendo en las diferentes pantallas, esperando a que termine la barra de progreso gris, situada en la parte inferior.

Cada módulo tendrá un color asociado para que el usuario sepa en cada instante, en qué módulo se encuentra. La asociación es la siguiente:

- Naranja → Coordinación
- Azul → Actividades
- Verde → Comunicador
- Violeta → Arte

## B.3. MÓDULO COORDINACIÓN

---



Figura B.7: Pantalla de Inicio

- **Naranja** → Seleccionando este icono, se accede a la pantalla de configuración de coordinación.
- **Azul** → Seleccionando este icono, se accede a la pantalla de selección de actividades.
- **Verde** → Seleccionando este icono, se accede a la pantalla de selección del comunicador.
- **Violeta** → Seleccionando este icono, se accede a la pantalla de selección de arte.

## B.3. Módulo Coordinación

### B.3.1. Configuración Coordinación

En esta pantalla (Ver Figura B.8) se pueden configurar diferentes aspectos de la actividad de coordinación, que se explican a continuación:

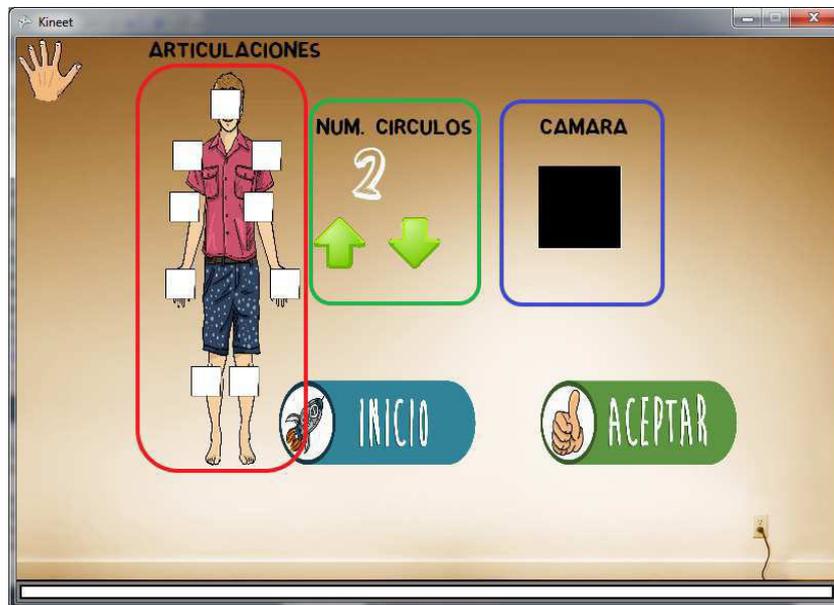


Figura B.8: Pantalla de Configuración de Coordinación

- **Rojo** → En esta imagen del cuerpo humano, se puede activar una casilla, manteniendo el cursor en forma de mano sobre ella, permitiendo elegir una de las articulaciones, con las que se realizará la actividad.
- **Azul** → Seleccionando este icono, se puede activar/desactivar la cámara de Kinect.
- **Verde** → Seleccionando este icono, se puede elegir entre el número de círculos con los que se quiere realizar la actividad de coordinación. El número varía entre 2 y 7.

### B.3.2. Edición Coordinación

Esta pantalla (Ver Figura B.9) permite editar la actividad de coordinación según las necesidades del especialista.

Seleccionando con la mano cualquiera de los círculos, puede arrastrarlo a cualquier parte de la pantalla. Para dejar el círculo en la posición deseada, sólo tendrá que levantar el brazo izquierdo.

El número situado en la parte inferior derecha, le indica el orden en el que se sitúa el círculo, para tenerlo en cuenta cuando el usuario tenga que unir los distintos círculos que le aparezcan en pantalla.

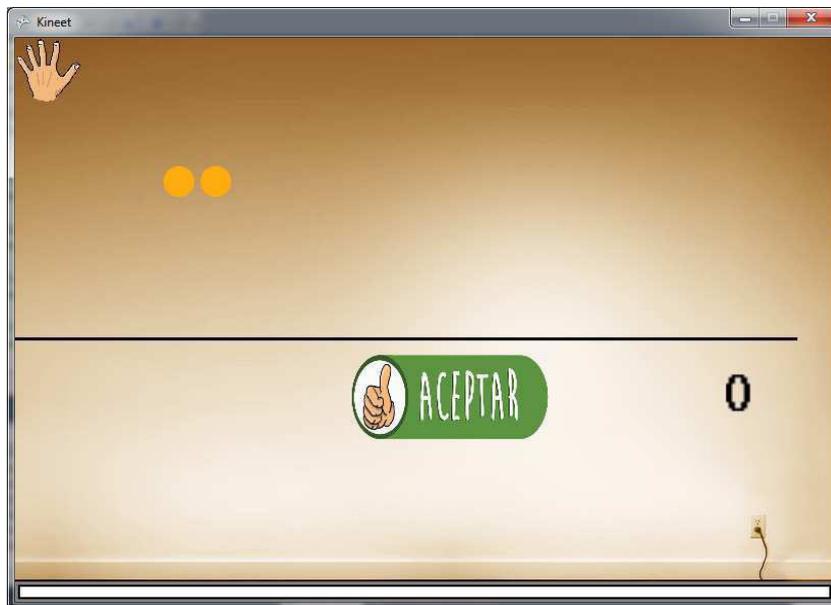


Figura B.9: Pantalla de Edición de Coordinación

### B.3.3. Actividad de Coordinación

En esta pantalla (Ver Figura B.10), el objetivo es unir los distintos círculos con la parte del cuerpo que ha sido seleccionada en la pantalla de configuración.

El círculo objetivo, que es el que tiene que tocar el usuario en este preciso instante, tendrá un color azul. Cuando se haya tocado este círculo, éste cambiará a verde, indicando que se ha efectuado la acción satisfactoriamente. La actividad termina cuando se hayan unido todos los círculos y aparecerá la pantalla de opciones.

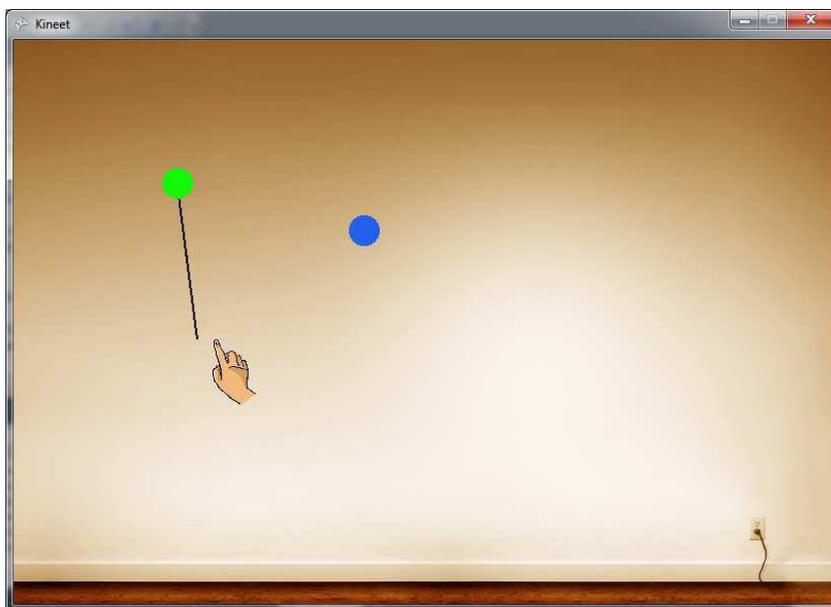


Figura B.10: Pantalla de Coordinación

## B.4. Módulo Actividades

### B.4.1. Selección de actividades

En esta pantalla (Ver Figura B.11) se puede elegir una de las cuatro actividades que cuenta la aplicación:

- Números
- Formas
- Grafomotricidad
- Causa-efecto

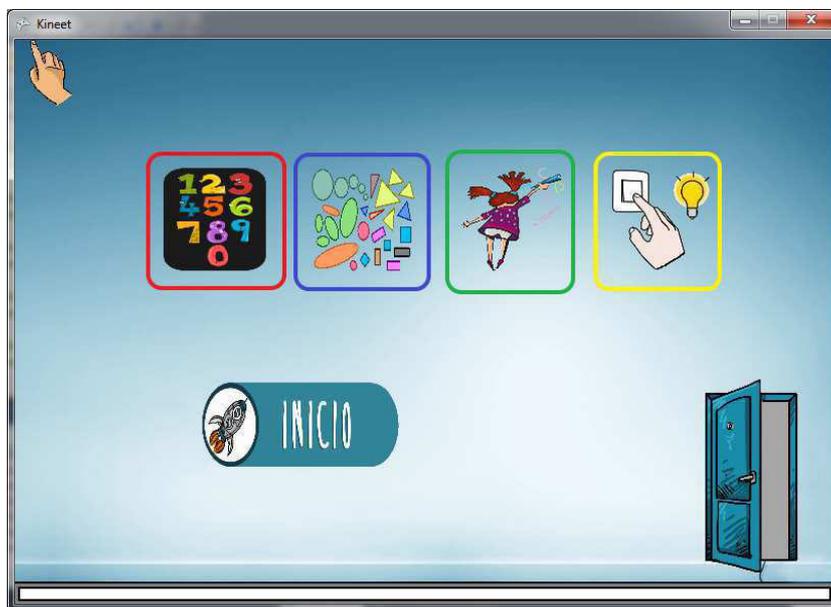


Figura B.11: Pantalla de Selección de Actividades

- **Rojo** → Seleccionando este icono, se accede a la pantalla de configuración de números.
- **Azul** → Seleccionando este icono, se accede a la pantalla de configuración de formas geométricas.
- **Verde** → Seleccionando este icono, se accede a la pantalla de configuración de grafomotricidad.
- **Amarillo** → Seleccionando este icono, se accede a la pantalla de configuración de causa-efecto.

#### B.4.2. Configuración Números

En esta pantalla (Ver B.12) se pueden configurar diferentes aspectos de la actividad de números, como se muestra a continuación:



Figura B.12: Pantalla de Configuración de Números

- **Amarillo** → Esta sección se refiere al nivel de dificultad de la actividad.
  - **Naranja** → Seleccionando esta opción, se elegirá el nivel fácil, donde sólo aparecerá un número cayendo en la pantalla.
  - **Gris** → Seleccionando esta opción, se elegirá el nivel normal, donde tendrá que realizar la ordenación de dos números.
  - **Rosa** → Seleccionando esta opción, se elegirá el nivel difícil, donde tendrá que ordenar tres números.
- **Rojo** → Seleccionando la flecha ascendente subiremos la velocidad con la que se desplazan los números por la pantalla, mientras que si se selecciona la flecha descendente se bajará. El rango varía entre 1 y 10.
- **Azul** → Seleccionando la flecha ascendente subiremos el tiempo que habrá que esperar entre la realización de una ordenación y la siguiente, mientras que si se selecciona la flecha descendente bajará el tiempo. El rango varía entre 1 y 10, medido en segundos.
- **Verde** → Seleccionando este icono, se puede activar/desactivar la cámara de Kineet.

### B.4.3. Actividad Números

La actividad de números, consistirá en tocar los números que aparecerán en pantalla en orden ascendente, con cualquiera de las dos manos. La cantidad de números que aparezcan dependerán del nivel de dificultad elegido previamente.



Figura B.13: Pantalla de la Actividad de Números

### B.4.4. Configuración Formas Geométricas

En esta pantalla (Ver B.14) se pueden configurar diferentes aspectos de la actividad de formas geométricas, como se muestra a continuación:

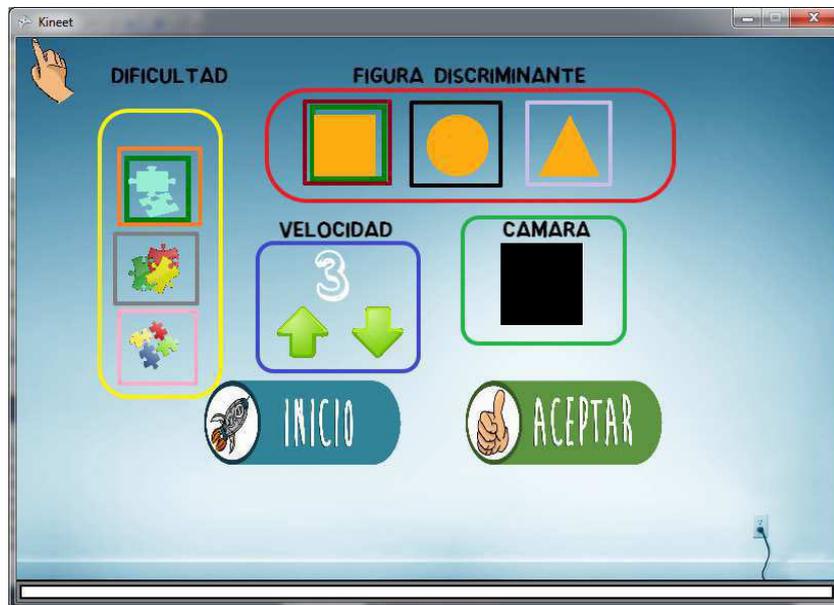


Figura B.14: Pantalla de Configuración de Formas Geométricas

- **Amarillo** → Esta sección se refiere al nivel de dificultad de la actividad.
  - **Naranja** → Seleccionando esta opción, se elegirá el nivel fácil, donde sólo aparecerá una figura cayendo en la pantalla.
  - **Gris** → Seleccionando esta opción, se elegirá el nivel normal, donde tendrá que tocar sólo la figura que se haya elegido como discriminante en esta misma pantalla, de las dos figuras que aparecen en la escena.
  - **Rosa** → Seleccionando esta opción, se elegirá el nivel difícil, donde tendrá que tocar sólo la figura que se haya elegido como discriminante en esta misma pantalla, de las tres figuras que aparecen en la escena.
- **Rojo** → Seleccionando la flecha ascendente subiremos la velocidad con la que se desplazan los números por la pantalla, mientras que si se selecciona la flecha descendente se bajará. El rango varía entre 1 y 10.
- **Azul** → Seleccionando la flecha ascendente subiremos el tiempo que habrá que esperar entre la realización de una ordenación y la siguiente, mientras que si se selecciona la flecha descendente bajará el tiempo. El rango varía entre 1 y 10, medido en segundos.
- **Verde** → Seleccionando este icono, se puede activar/desactivar la cámara.

ra de Kinect.

### B.4.5. Actividad Formas Geométricas

En esta actividad, el objetivo es tocar la figura que el especialista ha elegido como discriminante, de entre un número de figuras que variará, dependiendo del nivel de dificultad elegido.

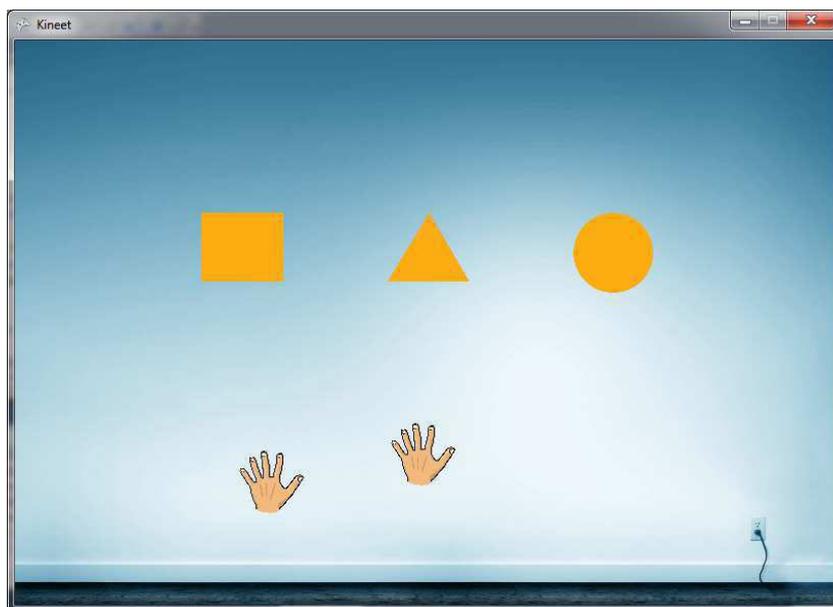


Figura B.15: Pantalla de la Actividad de Formas Geométricas

### B.4.6. Configuración Grafomotricidad

En esta pantalla (Ver B.16) se pueden configurar diferentes aspectos de la actividad de grafomotricidad, como se describe a continuación:



Figura B.16: Pantalla de Configuración de Grafomotricidad

- **Rojo** → Esta sección hace referencia al tipo de actividad de grafomotricidad.
  - **Naranja** → Seleccionando esta opción, se elegirá la actividad de grafomotricidad horizontal.
  - **Amarillo** → Seleccionando esta opción, se elegirá la actividad de grafomotricidad vertical.
  - **Verde** → Seleccionando esta opción, se elegirá la actividad de grafomotricidad libre.
- **Azul** → Seleccionando este icono, se puede activar/desactivar la cámara de Kinect.

#### B.4.7. Actividad Grafomotricidad Horizontal

El objetivo de esta actividad es pintar una línea desde el gusano hasta la manzana describiendo un movimiento horizontal. Para empezar a pintar se tocará la figura inicial (gusano) y se trazará una línea horizontal hasta llegar a la manzana.

Si el lápiz se sale de las líneas horizontales mientras se hace el trazado, tendrá que empezar desde el principio, tocando la figura considerada como

inicial (gusano).

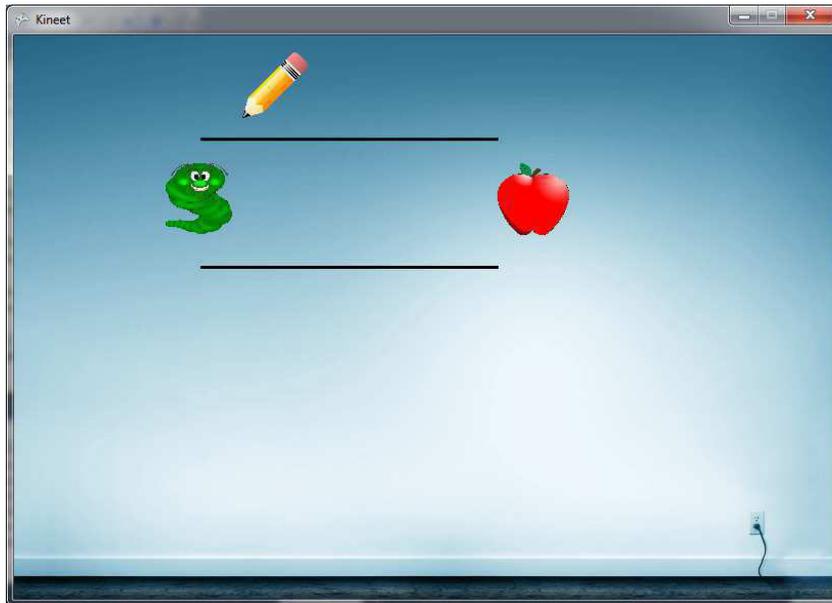


Figura B.17: Pantalla de Grafomotricidad Horizontal

#### B.4.8. Actividad Grafomotricidad Vertical

El objetivo de esta actividad es pintar una línea desde el gusano hasta la manzana pero esta vez describiendo un movimiento vertical. Para empezar a pintar se tocará la figura inicial (gusano) y se trazará una línea vertical hasta llegar a la manzana.

Si el lápiz se sale de las líneas verticales mientras se hace el trazado, tendrá que empezar desde el principio, tocando la figura considerada como inicial (gusano).

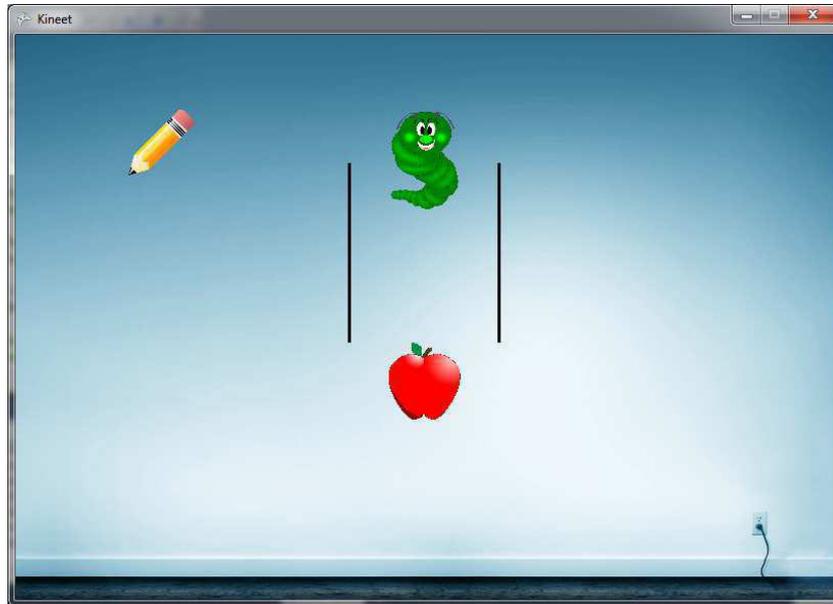


Figura B.18: Pantalla de Grafomotricidad Vertical

### B.4.9. Actividad Grafomotricidad Libre

Esta actividad, es una de las actividades personalizables de la aplicación, por este motivo hay que explicar la parte de edición de la actividad para adaptarlo a las necesidades que tenga tanto el especialista como el alumno, para describir seguidamente la realización de la misma.

#### B.4.9.1. Edición Grafomotricidad Libre

En esta pantalla (Ver Figura B.19), se pueden incluir las tres figuras que aparecen en la parte inferior dentro del circuito, para así aumentar la dificultad del ejercicio, puesto que al realizar la actividad no podrá tocar ninguna de estas figuras añadidas.

Para poder hacer esto, se tendrá que situar la mano encima de cualquiera de estas tres figuras hasta que termine la barra de progresión y se podrá arrastrar por la pantalla. La figura sólo aparecerá en la actividad si se encuentra entre las dos líneas horizontales, obviando el programa cualquier figura que se coloque fuera de éstas. Con lo cual se podrá tener entre 0 y 3 figuras inclusive. Para dejar la figura en la posición que se desee, sólo se tendrá que levantar el brazo izquierdo.



Figura B.19: Pantalla de Edición de Grafomotricidad Libre

#### B.4.9.2. Realización de la Actividad

El objetivo de esta actividad consiste en pintar una línea con el lápiz, desde el coche hasta la casa. El coche será el punto inicial así que no se empezará a pintar hasta que no se haya tocado éste. En este ejercicio particular se tendrán que evitar los obstáculos, así que no va a ser tan fácil como hacer una línea recta, igual que en los anteriores ejercicios de grafomotricidad.

Si se tocan las figuras intermedias entre el coche y la casa o se sale el lápiz de las líneas horizontales, se tendrá que empezar el ejercicio de nuevo, haciendo contacto con la figura inicial (coche).

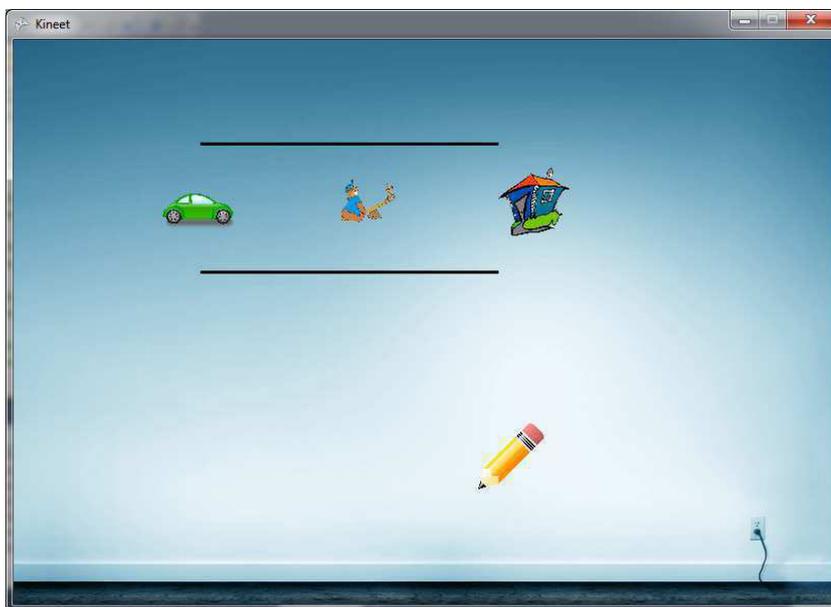


Figura B.20: Pantalla de Grafomotricidad Libre

#### **B.4.10. Configuración Causa – Efecto**

En esta pantalla (Ver B.21) se pueden configurar diferentes variables de la actividad de causa-efecto, como se describe a continuación:



Figura B.21: Pantalla de Configuración de Causa-Efecto

- Rojo → Esta sección hace referencia al tamaño que se mostrará la bombilla en pantalla.
  - Naranja → Seleccionando esta opción, se elegirá el tamaño más pequeño que puede aparecer la bombilla.
  - Verde → Seleccionando esta opción, se elegirá el tamaño estándar.
  - Rosa → Seleccionando esta opción, se elegirá el tamaño más grande.
- Azul → Esta sección se pueden elegir entre dos niveles de dificultad, estos niveles se diferencian en la posición que puede alcanzar la bombilla en la pantalla, en los niveles más avanzados la bombilla aparecerá más alejado del individuo y por consiguiente será más difícil tocarla y en los niveles más fáciles, la bombilla aparecerá cercano al cuerpo del usuario y por lo tanto se verá reducida su dificultad para tocarla.
  - Verde ⇒ Nivel fácil → la bombilla aparecerá en una área próxima al individuo.
  - Amarillo ⇒ Nivel difícil → la posición de la bombilla alternará entre una localización cercana al usuario o más alejada.

### **B.4.11. Actividad Causa – Efecto**

Esta actividad consiste en tocar con una de las dos manos la bombilla que tendrá dos estados:

- **Apagada** → si se presiona en este estado, dará la sensación de que se ha encendido la luz, aumentando la luminosidad de la cámara.
- **Encendida** → si se presiona en este estado, dará la sensación de que se ha apagado la luz, disminuyendo la luminosidad de la cámara.

## **B.5. Módulo Comunicador**

Este módulo tiene implementado un comunicador de peticiones, para que el alumno que tenga dificultades con el lenguaje pueda expresar lo que desea sin necesidad de tener que utilizar ni el lenguaje oral ni el escrito, sólo con la ayuda de pictogramas.

### **B.5.1. Configuración Comunicador**

En esta pantalla (Ver B.22) se pueden configurar diferentes variables del comunicador, como se describe a continuación:



Figura B.22: Pantalla de Configuración de Comunicador

- Rojo → Se puede seleccionar entre la mano izquierda o derecha, para las secciones del comunicador que sea necesario tocar pictogramas.
- Azul → Si se mantiene pulsado, se irá cambiando entre dos estados, indicando si queremos activar la opción de reconocimiento de gestos en el comunicador o se prefiere tocar los pictogramas con alguna de las dos manos.

### B.5.2. Actividad Comunicador

El comunicador tiene una serie de gestos y comandos por voz que es capaz de reconocer (Ver Cuadro B.2), cuando reconoce cualquiera de estos dos modos de expresión el pictograma referenciado pasará a la zona de seleccionado (Ver Figura B.23). Este comunicador consta de cinco pantallas:

- La pantalla para identificarse (Ver Figura B.24), donde se tiene que realizar el gesto asociado a 'yo' o su equivalente en comando de voz.
- La pantalla de petición (Ver Figura B.25), donde se tiene que realizar el gesto asociado a 'quiero' o su equivalente en comando de voz.
- La pantalla de acciones (Ver Figura B.26), donde aparecerán seis verbos que es capaz de reconocer mediante gestos o por voz (Ver Tabla B.2).

- La pantalla de objetos (Ver Figura B.27), ya no permite el reconocimiento de gestos para elegir el objeto deseado tiene que ser tocando con la mano en el pictograma deseado, aunque la función de reconocimiento de voz si está activa. Sin embargo, no todos los verbos tienen asociados unos objetos, sólo los verbos comer (Ver Cuadro B.3), beber (Ver Cuadro B.4) y jugar (Ver Cuadro B.5) permiten acceder a esta pantalla, los restantes verbos pasarán directamente a la pantalla final (Ver Figura B.28).
- La pantalla final (Ver Figura B.28), aparecerá la petición final elegida por el alumno junto con un feedback auditivo, el cual repetirá la frase completa con los pictogramas seleccionados durante la realización de la actividad.

Para poder realizar el siguiente paso tanto en la pantalla de acciones como en la pantalla de objetos, mediante el reconocimiento de voz, habrá que decir la palabra 'listo', una vez que se haya seleccionado el pictograma deseado.

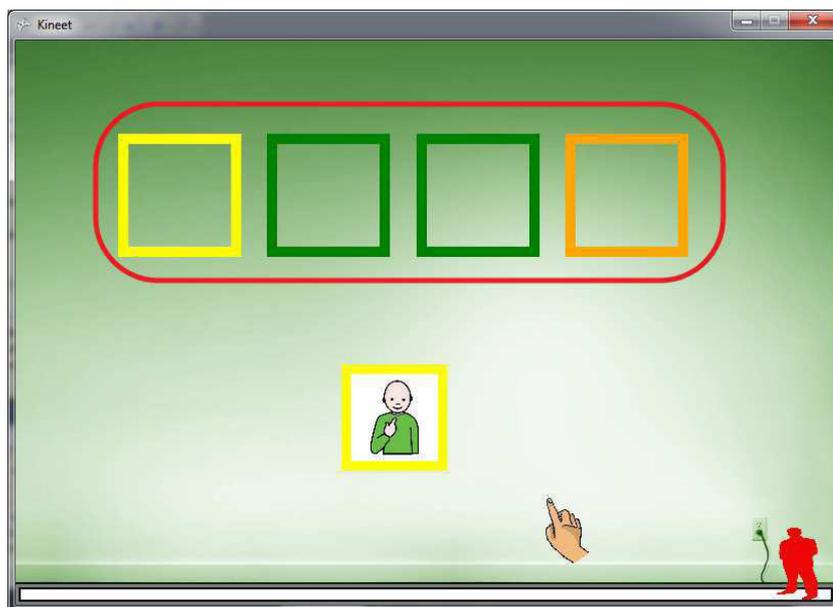


Figura B.23: Zona de Pictogramas Seleccionados

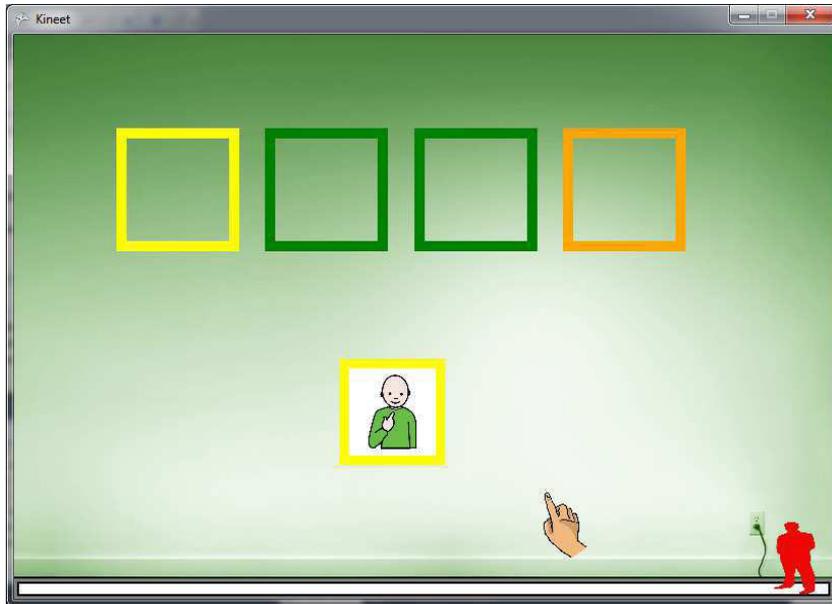


Figura B.24: Pantalla de Identificación

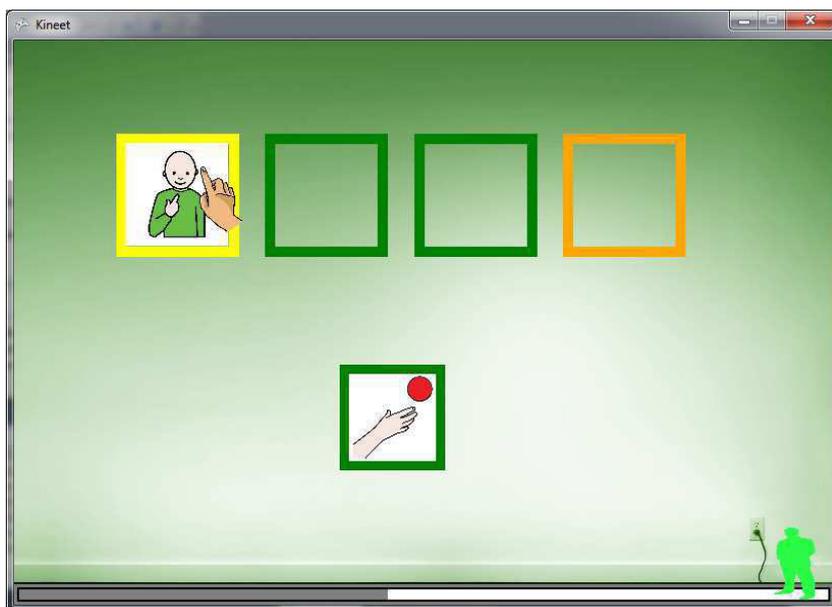


Figura B.25: Pantalla de Peticiones

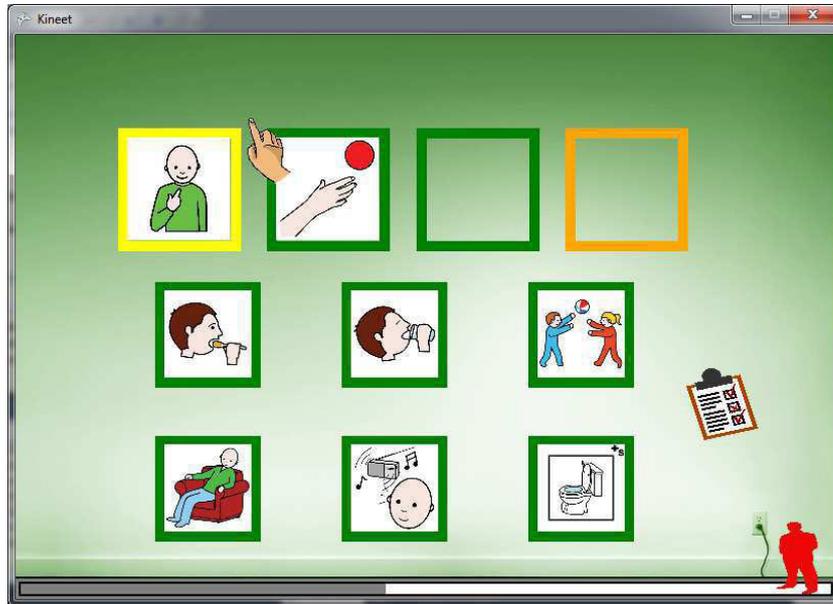


Figura B.26: Pantalla de Acciones



Figura B.27: Pantalla de Objetos

KiNEEt: Aplicación para el aprendizaje y rehabilitación en educación especial

---

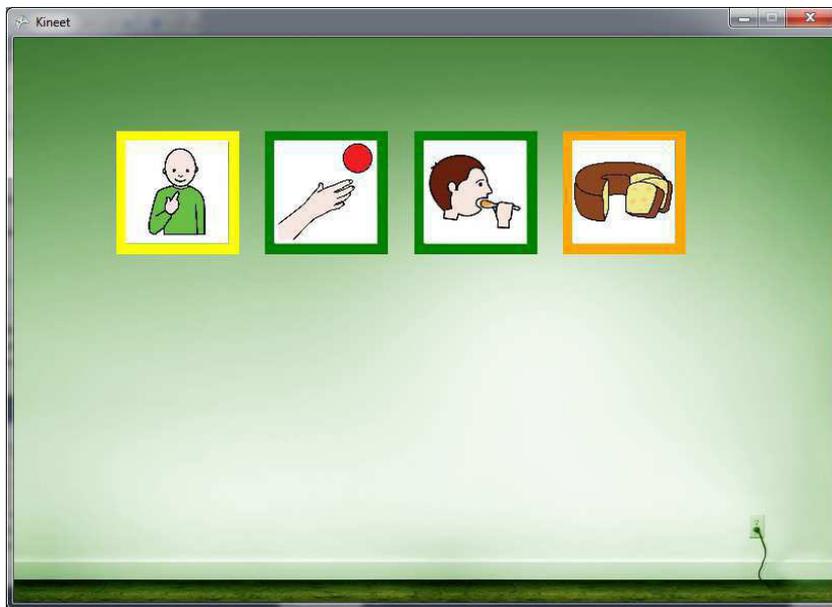
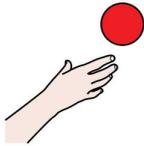


Figura B.28: Pantalla Final

Pictograma	Gesto	Comando por Voz
	La mano derecha en frente del tórax, hay que dirigirla hacia éste	'yo'
	Se desplazará la mano derecha del hombro izquierdo al derecho	'quiero'
	Se desplazará la mano derecha a la boca	'comer'
	Se desplazará ambas manos a la boca	'beber'
	Se desplazarán ambas manos a la oreja derecha, juntando éstas al final	'descansar'
	Se desplazará la mano derecha por encima de la cabeza, apuntando hacia ésta	'ir baño'
	Se desplazará la mano derecha hacia la oreja derecha con la palma extendida	'escuchar música'
	Se cruzarán ambas manos a la altura del tórax, teniendo como punto de unión ambas muñecas	'jugar'

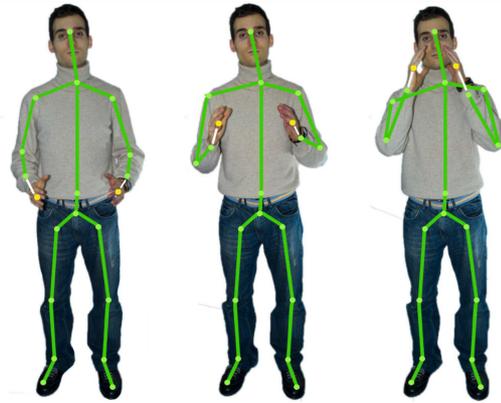
Cuadro B.1: Tabla de Gestos y Comandos por Voz reconocidos por el Comunicador

KiNEEt: Aplicación para el aprendizaje y rehabilitación en educación especial

---

Gestos	Representación gráfica		
'yo'			
'quiero'			
'comer'			

'beber'



'descansar'



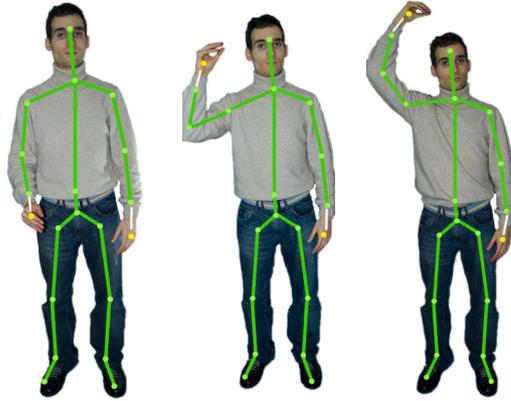
'escuchar  
música'



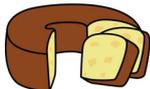
'jugar'



'ir al baño'



Cuadro B.2: Tabla de Gestos

Pictograma	Comando por Voz
	'bocadillo'
	'magdalena'
	'galleta'
	'bizcocho'
	'fruta'
	'yogur'

Cuadro B.3: Tabla de Alimentos

Pictograma	Comando por Voz
	'batido'
	'cola cao'
	'leche'
	'bizcocho'
	'refresco'
	'zumó'

Cuadro B.4: Tabla de Bebidas

Pictograma	Comando por Voz
	'lego'
	'muñeco'
	'galleta'
	'ordenador'
	'pelota'
	'tablet'

Cuadro B.5: Tabla de Juguetes

Los pictogramas utilizados son propiedad de ARASAAC: Sergio Palao Procedencia: ARASAAC (<http://catedu.es/arasaac/>) Licencia: CC (BY-NC-SA)

## B.6. Módulo Arte

### B.6.1. Pantalla de Configuración Genérica para Avatares

Las actividades de arte que contienen avatares, que son las actividades relacionadas con la música y la danza tienen la misma pantalla de configuración (Ver Figura B.29). Esta ventana permite que se configuren dos aspectos en estas actividades, relacionadas con el equipo que se va a utilizar y no tan orientado a las actividades en sí.



Figura B.29: Pantalla de Configuración Genérica

- Rojo → Se puede seleccionar la resolución de pantalla.
- Azul → Se puede seleccionar la calidad gráfica, según el equipo que se tenga.

### B.6.2. Actividad de Música con Platillos

En esta actividad, hay que realizar el gesto de tocar los platillos, es decir, separar y seguidamente juntar las manos, para que suene el sonido de este instrumento. El avatar realiza los gestos que hace el usuario.

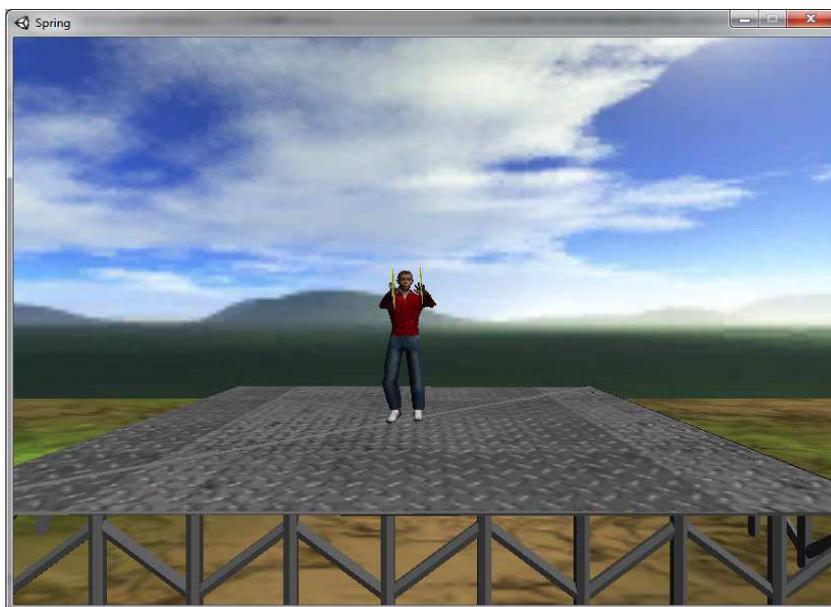


Figura B.30: Pantalla de Actividad de Platinos

### B.6.3. Actividad de Música con Piano

En esta actividad, hay que realizar un gesto como si se tocara el piano, para ello se tiene que separar los brazos del piano y tener una toma de contacto con él. Cuando el gesto sea reconocido sonará una nota al azar de la escala musical de este instrumento. El avatar realiza los gestos que hace el usuario.



Figura B.31: Pantalla de Actividad de Piano

#### B.6.4. Actividad de Música Conjunta

En esta actividad, pueden interactuar dos personas y tocar a la vez los dos instrumentos que se han descrito anteriormente. Los gestos son los mismos que en las dos secciones anteriores, con la particularidad de la participación de dos usuarios al mismo tiempo.



Figura B.32: Pantalla de Actividad de Música Conjunta

### B.6.5. Actividad de Danza Interactiva

En esta actividad, el usuario mediante movimientos libres puede interactuar con el entorno creado. Con este fin, se han reconocido tres posturas:

- Los brazos levantados → hace que caigan flores al escenario.
- Los brazos en cruz → hace que las flores se queden congeladas.
- Los brazos bajados → hace que las flores desaparezcan.



Figura B.33: Pantalla de Actividad de Danza interactiva

### B.6.6. Actividad de Danza con Objetos

En esta actividad, se permite interactuar con una pelota, la cual podrá moverla con su mano derecha, hasta que realice un gesto similar al de lanzar una pelota y ésta ocasione una explosión. Para reiniciar la actividad, hay que levantar el brazo izquierdo.



Figura B.34: Pantalla de Actividad de Danza con objetos

### B.6.7. Configuración Pintura

En esta pantalla (Ver B.35) se pueden configurar diferentes variables de la actividad de pintar, como se describe a continuación:



Figura B.35: Pantalla de Actividad de Configuración de Pintura

### B.6.8. Actividad Pintura

En esta actividad (Ver Figura B.36), se permite pintar con una de las dos manos, según sea el usuario zurdo o diestro, cambiando variables como el color o el tipo de trazo, posteriormente se describen cada una de las opciones configurables del entorno.

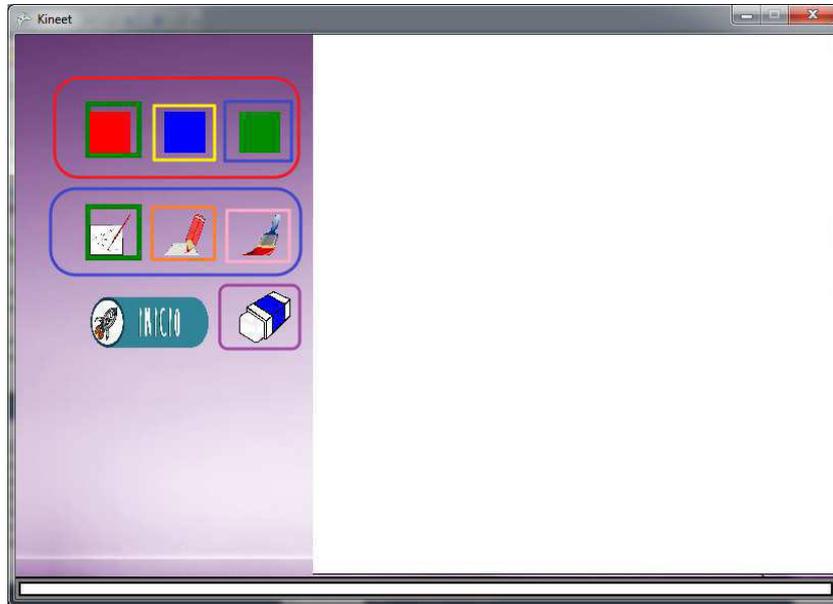


Figura B.36: Pantalla de Actividad de Pintura

- Rojo → En esta sección se encuentran los distintos colores con los que se puede pintar.
  - Verde → Seleccionando esta opción, se puede elegir el color rojo para pintar.
  - Amarillo → Seleccionando esta opción, se puede elegir el color azul para pintar.
  - Azul → Seleccionando esta opción, se puede elegir el color verde para pintar.
- Azul → Esta sección se pueden elegir entre dos niveles de dificultad, estos niveles se diferencian en la posición que puede alcanzar la bombilla en la pantalla, en los niveles más avanzados la bombilla aparecerá más alejado del individuo y por consecuente será más difícil tocarla y en los niveles más fáciles, la bombilla aparecerá cercano al cuerpo del usuario y por lo tanto se verá reducida su dificultad para tocarla.
  - Verde → Seleccionando esta opción, se pintará con un trazo fino.
  - Naranja → Seleccionando esta opción, se pintará con un trazo estándar.
  - Rosa → Seleccionando esta opción, se pintará con un trazo grueso.

- Violeta  $\Rightarrow$  Seleccionando esta opción, se borrará el área de dibujo.



# Apéndice C

## Cuestionarios de la Aplicación

### Instrumento de evaluación didáctica de la aplicación

Valora todas las sentencias de 1 a 5, siendo 1 lo más negativo y 5 lo más positivo.

#### Comprensión y Uso de la aplicación:

<b>Sentencia</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
La aplicación es fácil de usar					
Se accede fácilmente al tipo de actividad que quiero hacer					
La instalación del dispositivo sé hacerla sin problemas					
La personalización de las actividades sé hacerlas fácilmente					
He tardado menos de un par de horas en usar la aplicación					

Cuadro C.1: Usabilidad

#### Uso educativo de la aplicación:

APÉNDICE C. CUESTIONARIOS DE LA APLICACIÓN

---

<b>Sentencia</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
Las actividades se adaptan a mis demandas como docente					
Las actividades son útiles para mis alumnos/as					
Las actividades se adaptan a las necesidades de mis alumnos/as					
Las actividades son un recurso que utilizo por ser motivante					
Las actividades son un recurso que utilizo por ser accesible (características físicas)					
Las actividades son un recurso que utilizo por ser personalizable					

Cuadro C.2: Uso educativo

# Apéndice D

## Cuestionario de las Capacidades del Alumnado

### Evaluación de las capacidades del alumnado

Nombre					
N.E.E.					
Modulo					
Tiempo de realización					
Ejercicio preferido					
	1	2	3	4	5
Se reconoce a sí mismo en la actividad					
Identifica los movimientos de la pantalla con sus propios movimientos					
Utiliza de forma correcta la lateralidad					
Comprende la instrucción dada					
Imita el movimiento de la instrucción					
Realiza la instrucción dada de forma autónoma					

APÉNDICE D. CUESTIONARIO DE LAS CAPACIDADES DEL  
ALUMNADO

---

Realiza la instrucción con algún tipo de ayuda:					
Modelamiento					
Moldeamiento					
Encadenamiento					
Repite la consigna verbal					
Se autorregula durante la actividad					

Presenta motivación ante la actividad					
Muestra mayor interés ante las actividades con sonidos, estímulos visuales, feedback,...					
Presenta frustración ante dificultades en el uso de la aplicación					
Grado de éxito en el realización de la tarea					
Tiempo de latencia de respuesta					

- 1 – Nunca
- 2- Poco
- 3- a veces
- 4- bastante
- 5- siempre

## Apéndice E

### Diagrama de Casos de Uso

## APÉNDICE E. DIAGRAMA DE CASOS DE USO

---

## Apéndice F

### Diagrama de Flujo de Coordinación y Comunicador

APÉNDICE F. DIAGRAMA DE FLUJO DE COORDINACIÓN Y  
COMUNICADOR

---

## Apéndice G

### Diagrama de Flujo de Actividades

APÉNDICE G. DIAGRAMA DE FLUJO DE ACTIVIDADES

---

## Apéndice H

### Diagrama de Flujo de Arte

APÉNDICE H. DIAGRAMA DE FLUJO DE ARTE

---

# Apéndice I

## Diagrama de Clases

