



UNIVERSIDAD TÉCNICA DEL NORTE



INSTITUTO DE POSTGRADO

MAESTRÍA EN INGENIERÍA DE SOFTWARE

**“INTERFACES NATURALES DE USUARIO PARA LA ENSEÑANZA DE
UBICACIÓN ESPACIAL A NIÑOS DE EDUCACIÓN PREESCOLAR:
DETERMINACIÓN DE DIRETRICES Y DISEÑO DE APLICACIÓN”**

**Trabajo de Investigación previo a la obtención del Título de Magíster en
Ingeniería del Software**

DIRECTOR:

Carlos Xavier Rosero Chandi

AUTORES:

Mario Roberto Merlo Rosas

Fernando Daniel Recalde Salazar

IBARRA - ECUADOR

2019

UNIVERSIDAD TÉCNICA DEL NORTE
MAESTRÍA EN INGENIERÍA DE SOFTWARE

CERTIFICADO

MSC. CARLOS XAVIER ROSE ROSE CHANDI

CERTIFICA

En calidad de director del trabajo de grado titulado: “**INTERFACES NATURALES DE USUARIO PARA LA ENSEÑANZA DE UBICACIÓN ESPACIAL A NIÑOS DE EDUCACIÓN PREESCOLAR: DETERMINACIÓN DE DIRECTRICES Y DISEÑO DE APLICACIÓN**”, presentado por los Ingenieros; Mario Roberto Merlo Rosas y Fernando Daniel Recalde Salazar, como requisito previo para la obtención del título de MAGISTER en Ingeniería de Software, ha sido guiado y revisado periódicamente y cumple normas establecidas en el Reglamento de Estudiantes de la Universidad Técnica del Norte, por lo que doy fe que dicho trabajo reúne los requisitos suficientes para ser sometido a presentación pública y evaluación por parte del jurado examinador que se designe.

Ibarra, 19 de febrero del 2019.



MSc. **Carlos Xavier Rosero Chandi**
DIRECTOR DEL PROYECTO



UNIVERSIDAD TÉCNICA DEL NORTE
Resolución N° 001-073 CEAACES -2013-13
INSTITUTO DE POSTGRADO



CARTA DE ACEPTACIÓN ASESOR

Ibarra, 05 de abril del 2019

Dra. Lucía Yépez V MSc.

Directora

Instituto de Postgrado

ASUNTO: Conformidad con el documento final

Señora Directora:

Nos permitimos informar a usted que hemos revisado el Trabajo de Grado “INTERFACES NATURALES DE USUARIO PARA LA ENSEÑANZA DE UBICACIÓN ESPACIAL A NIÑOS DE EDUCACIÓN PREESCOLAR: DETERMINACIÓN DE DIRECTRICES Y DISEÑO DE APLICACIÓN” de los maestranes Fernando Daniel Recalde Salazar y Mario Roberto Merlo Rosas, certificamos que han sido acogidas y satisfechas todas las observaciones realizadas.

Atentamente,

	Apellidos y Nombres	Firma
Tutor/a	MSc. Carlos Xavier Rosero Chandi	
Asesor/a	MSc. Vicente Alexander Guevara Vega	

UNIVERSIDAD TÉCNICA DEL NORTE
MAESTRÍA EN INGENIERÍA DE SOFTWARE
DECLARACIÓN DE RESPONSABILIDAD

Ing. Mario Roberto Merlo Rosas

Ing. Fernando Daniel Recalde Salazar

DECLARAMOS QUE,

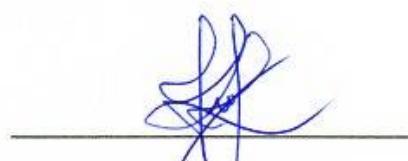
El trabajo de grado denominado: “**INTERFACES NATURALES DE USUARIO PARA LA ENSEÑANZA DE UBICACIÓN ESPACIAL A NIÑOS DE EDUCACIÓN PREESCOLAR: DETERMINACIÓN DE DIRECTRICES Y DISEÑO DE APLICACIÓN**”, ha sido desarrollado en base a una investigación exhaustiva, respetando derechos intelectuales de terceros, cuyas fuentes se incorporan en la bibliografía.

En virtud de esta declaración, nos responsabilizamos del contenido, veracidad y alcance científico del proyecto de grado en mención.

Ibarra, 19 de febrero 2019



Ing. Fernando Daniel Recalde Salazar
1715651632



Ing. Mario Roberto Merlo Rosas
100223664-2



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO		
CÉDULA DE IDENTIDAD:	1002236642	
APELLIDOS Y NOMBRES:	MERLO ROSAS MARIO ROBERTO	
DIRECCIÓN:	BARTOLOME GARCÍA 1-68	
EMAIL:	mariomr289@gmail.com	
TELÉFONO FIJO:	062954696	TELÉFONO MÓVIL: 0991866904

DATOS DE CONTACTO		
CÉDULA DE IDENTIDAD:	1715651632	
APELLIDOS Y NOMBRES:	RECALDE SALAZAR FERNANDO DANIEL	
DIRECCIÓN:	HUERTOS FAMILIARES – CALLE GUAYAS Y 13 DE ABRIL 3-47	
EMAIL:	fdrs_2127@hotmail.com	
TELÉFONO FIJO:	062545337	TELÉFONO MÓVIL: 0989283683

DATOS DE LA OBRA	
TÍTULO:	“INTERFACES NATURALES DE USUARIO PARA LA ENSEÑANZA DE UBICACIÓN ESPACIAL A NIÑOS DE EDUCACIÓN PREESCOLAR: DETERMINACIÓN DE DIRECTRICES Y DISEÑO DE APLICACIÓN”
AUTOR (ES):	MERLO ROSAS MARIO ROBERTO RECALDE SALAZAR FERNANDO DANIEL
FECHA: DD/MM/AAAA	23/04/2019
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	<input type="checkbox"/> PREGRADO <input checked="" type="checkbox"/> POSGRADO
TITULO POR EL QUE OPTA:	MAGISTER EN INGENIERIA DE SOFTWARE
ASESOR /DIRECTOR:	MSC. CARLOS XAVIER ROSERO CHANDI

2. CONSTANCIAS

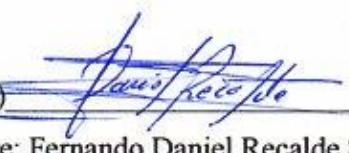
El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 10 días del mes de mayo de 2019

EL AUTOR:

(Firma) 
Nombre: Mario Roberto Merlo Rosas

EL AUTOR:

(Firma) 
Nombre: Fernando Daniel Recalde Salazar



UNIVERSIDAD TÉCNICA DEL NORTE

MAESTRÍA EN INGENIERÍA DE SOFTWARE

CESIÓN DE DERECHOS DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

Fernando Daniel Recalde Salazar, con número de cédula de identidad Nro. 171565163-2 y Mario Roberto Merlo Rosas, con número de cédula de identidad Nro. 100223664-2, manifestamos nuestra voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la Ley de Propiedad Intelectual del Ecuador, artículos 4, 5 y 6, en calidad de autor de la obra o trabajo de grado denominado: **“INTERFACES NATURALES DE USUARIO PARA LA ENSEÑANZA DE UBICACIÓN ESPACIAL A NIÑOS DE EDUCACIÓN PREESCOLAR: DETERMINACIÓN DE DIRECTRICES Y DISEÑO DE APLICACIÓN”**, que ha sido desarrollado para optar por el título de Magíster en Ingeniería de Software, en la Universidad Técnica del Norte facultada para ejercer plenamente los derechos cedidos anteriormente. En nuestra condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia suscribimos este documento en el momento de realizar la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Técnica del Norte.

Ing. Fernando Daniel Recalde Salazar
171565163-2

Ing. Mario Roberto Merlo Rosas
100223664-2

Ibarra, 10 días del mes de mayo de 2019.

DEDICATORIA

Dedico este trabajo a Dios, por ser el guía de mi trayecto profesional para cumplir mis metas planteadas.

A mis hijos Jhoel y Evans, quienes son el mejor regalo de Dios.

A mis Abuelitos quienes son una inspiración y un camino de testimonio de vida a seguir.

Fernando

El presente trabajo se lo dedico en primer lugar a mi Padre Rigoberto y a mi Madre Isabel que han sabido inculcarme sus valores y sus enseñanzas para convertirme en un hombre de bien.

Además, a mi hermano Milton, mi hermana Doris, mi cuñada Sandra que me han apoyado y me han aconsejado durante toda mi vida y mi carrera universitaria.

Por último, a mis sobrinos Melany, Dylan, Jhosephine, Evangeline, Angelique y Meredith que han sabido alegrarme en los momentos más difíciles, esperando seguir siendo una inspiración positiva en sus vidas.

Mario

AGRADECIMIENTO

En primer lugar, Agradezco a Dios por darme la oportunidad de vivir.

A la Unidad Educativa Fiscomisional La Inmaculada Concepción, por abrir las puertas y dar apertura en la realización de este proyecto.

Al Asesor de tesis MSc. Carlos Xavier Rosero Chandi por su dedicación y apoyo incondicional en el desarrollo del presente proyecto.

Fernando

Quiero agradecer siempre en primer lugar a Dios por sobre todas las cosas por haberme dado la vida y concederme todas las oportunidades de ser mejor en la vida.

Agradezco a la Universidad Técnica del Norte por otorgarme la oportunidad de estudiar y alcanzar todas mis metas profesionales en esta Institución tan prestigiosa del país. Al Instituto de Posgrado por acogerme en su programa de Maestría.

Doy gracias a los Docentes de la Maestría en Ingeniería de Software por los conocimientos impartidos. A la Unidad Educativa Fiscomisional La Inmaculada Concepción por concedernos la apertura en la realización de este proyecto.

Agradezco a mi Tutor de tesis MSc. Carlos Xavier Rosero Chandi por sus conocimientos, dedicación y apoyo incondicional en desarrollo del presente proyecto.

A mi hermano Milton por ayudarme en el diseño de algunos de los personajes usados en los videojuegos de ubicación espacial.

Mario

ÍNDICE DE CONTENIDOS

CERTIFICADO	II
CARTA DE ACEPTACIÓN ASESOR	III
DECLARACIÓN DE RESPONSABILIDAD	IV
AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	V
CESIÓN DE DERECHOS DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	VII
DEDICATORIA	VIII
AGRADECIMIENTO	IX
ÍNDICE DE CONTENIDOS	X
ÍNDICE DE TABLAS	XVI
ÍNDICE DE FIGURAS	XIX
RESUMEN	XXII
ABSTRACT	XXIII
CAPÍTULO I	1
EL PROBLEMA	1
1.1. Problema de Investigación	1
1.1.1. Contextualización del Problema	1
1.1.2. Planteamiento del Problema	3
1.1.3. Proposición	3
1.1.4. Variables	3
1.2. Objetivos de la Investigación	4
1.2.1. Objetivo General	4
1.2.2. Objetivos Específicos	4
1.3. Justificación	4
CAPÍTULO II	5

MARCO REFERENCIAL	5
2.1. Antecedentes.....	5
2.2. Fundamentación Tecnológica.....	6
2.3. Fundamentación Legal	7
2.4. Referentes Teóricos	9
2.4.1. Noción Espacial	9
2.4.2. Conceptos Espaciales.....	9
2.4.3. Pensamiento Infantil	11
2.4.4. Videojuegos en la Educación.....	11
2.4.5. Arquitectura y Diseño de un Videojuego.....	13
2.4.6. Interfaz Natural de Usuario (NUI)	15
2.4.7. Ventajas de las Interfaces Naturales de Usuario (NUI)	16
2.4.8. Usabilidad	17
2.4.9. Calidad Interactiva y Experiencia de Usuario	19
2.4.10. Calidad en Uso	19
2.4.11. Jugabilidad en Videojuegos	20
CAPÍTULO III	21
MARCO METODOLÓGICO	21
1.1. Descripción del Área de Estudio	21
1.1.1. Misión	21
1.1.2. Visión.....	21
1.1.3. Ubicación	21
1.1.4. Beneficiarios	22
1.2. Diseño y Tipo de Investigación	22
1.2.1. Tipo de Investigación.....	22
1.2.2. Diseño de la Investigación	22
1.2.2.1. Modalidad de la Investigación	22

1.2.2.2.	Tipos o Niveles de Investigación	22
1.3.	Procedimiento de Investigación.....	23
1.3.1.	Población y Muestra	23
1.3.1.1.	Población.....	23
1.3.1.2.	Muestra.....	23
1.3.2.	Operacionalización de Variables	23
1.3.3.	Métodos.....	25
1.3.4.	Metodología de Desarrollo de Software	26
1.3.5.	Estrategias Técnicas.....	27
1.3.6.	Instrumentos de Investigación	28
1.4.	Análisis e Interpretación de Resultados.....	28
1.4.1.	Análisis de la Observación.....	28
1.4.2.	Interpretación de la Observación	31
1.4.3.	Análisis de la Entrevista.....	31
1.4.4.	Interpretación de la Entrevista	33
CAPÍTULO IV		34
PROPUESTA		34
4.1.	Algoritmo Propuesto.....	34
4.1.1.	Pasos del Algoritmo Propuesto	36
4.2.	Fase de Planificación del Proyecto	38
4.2.1.	Historias de Usuario.....	38
4.2.2.	Plan de Versiones.....	46
4.2.3.	Plan de Iteraciones	47
4.2.4.	Propósito	49
4.2.5.	Alcance	50
4.2.6.	Definiciones, Siglas y Abreviaciones	50
4.2.7.	Perspectiva del Producto.....	50

4.2.8.	Interfaces con el Usuario	50
4.2.9.	Interfaces con el Hardware	51
4.2.10.	Interfaces con el Software	52
4.2.10.1.	Sistema Operativo Ubuntu	52
4.2.10.2.	Librería Libfreenect.....	53
4.2.11.	Restricciones de Memoria.....	53
4.2.12.	Modos de Operación de Usuario.....	53
4.2.13.	Funciones del Producto	54
4.2.14.	Características del Usuario.....	55
4.2.15.	Restricciones Generales	56
4.2.15.1.	Restricciones de Hardware.....	56
4.2.15.2.	Restricciones del Kinect.....	56
4.2.15.3.	Restricciones de Software	57
4.2.16.	Suposiciones.....	57
4.2.17.	Dependencias	57
4.2.18.	Requisitos Específicos.....	58
4.2.18.1.	Requerimientos Funcionales	58
4.2.18.2.	Requisitos No Funcionales.....	69
4.3.	Fase de Diseño.....	69
4.3.1.	Vista Lógica	70
4.3.2.	Vista de Implementación	72
4.3.3.	Vista de Procesos	72
4.3.4.	Vista Física.....	77
4.3.5.	Vista de escenarios.....	78
4.3.6.	Prototipos Conceptuales de la Interfaz de Usuario (IU)	78
4.4.	Fase de Codificación	86
4.4.1.	Estructura General del Videojuego UBIC	86

4.5.	Fase de Pruebas	90
4.5.1.	Pruebas Físicas.....	91
4.5.2.	Pruebas Alpha	91
4.5.2.1.	Diseño de los Casos de Prueba.....	91
4.5.2.2.	Ejecución de las Pruebas	102
4.5.3.	Pruebas Beta.....	107
	CAPÍTULO V.....	112
	RESULTADOS	112
5.1.	Selección de la Muestra	112
5.2.	Evaluación de la Usabilidad	113
5.2.1.	Metodología propuesta para evaluar la Usabilidad.....	113
5.2.2.	Tareas Propuestas.....	114
5.2.3.	Consideraciones Generales para la Evaluación	118
5.2.4.	Prueba de Efectividad	118
5.2.5.	Prueba de Eficiencia	119
5.2.6.	Prueba de Satisfacción	119
5.3.	Pruebas y Resultados	121
5.3.1.	Configuración de la Simulación.....	121
5.3.2.	Prueba de Efectividad	122
5.3.3.	Prueba de Eficiencia	123
5.3.4.	Prueba de Satisfacción	125
5.3.5.	Discusión Final Sobre la Evaluación del Videojuego de Ubicación Espacial	127
	CAPÍTULO VI	129
	CONCLUSIONES Y RECOMENDACIONES	129
6.1.	Conclusiones.....	129
6.2.	Recomendaciones	129
6.3.	Trabajo Futuro	130

REFERENCIAS	131
ANEXOS	139
Anexo A: Cuestionario de Preguntas de la Entrevista.....	139
ANEXO B: Manual de Instalación de los Prerrequisitos para la Implementación del Videojuego “UBIC”	141
Anexo C: Cuestionario de Escala de Usabilidad del Sistema	159
Anexo D: Cuestionario de Comodidad / Facilidad de Uso	160
Anexo E: Test de Evaluación de Ubicación Espacial.....	161

ÍNDICE DE TABLAS

Tabla 1. Población	23
Tabla 2. Operacionalización Variable Independiente	24
Tabla 3. Operacionalización Variable Dependiente	24
Tabla 4. Métodos Generales	26
Tabla 5. Ficha de Bajada de Observación Primero de Básica paralelo "A"	29
Tabla 6. Ficha de Bajada de Observación Primero de Básica paralelo "B"	30
Tabla 7. Ficha de Bajada de Observación Primero de Básica paralelo "C"	31
Tabla 8. Ficha de Bajada de Entrevistas	32
Tabla 9. Equivalencias de Cartas de la Baraja Inglesa	38
Tabla 10. Historia de Usuario Número1. Creación del Sistema de Seguimiento de la Mano	39
Tabla 11. Historia de Usuario Número 2. Ingreso a la Pantalla Principal de la Aplicación.....	39
Tabla 12. Historia de Usuario Número 3. Ingreso al Menú de Videojuegos de la Aplicación.....	39
Tabla 13. Historia de Usuario Número 4. Ingreso al Videojuego de Entrenamiento	40
Tabla 14. Historia de Usuario Número 5. Actividad de Entrenamiento Izquierda.....	40
Tabla 15. Historia de Usuario Número 6. Actividad de Entrenamiento Derecha	41
Tabla 16. Historia de Usuario Número 7. Actividad de Entrenamiento Arriba	41
Tabla 17. Historia de Usuario Número 8. Actividad de Entrenamiento Abajo.....	42
Tabla 18. Historia de Usuario Número 9. Pantalla de Fin de Juego.....	42
Tabla 19. Historia de Usuario Número 10. Ingreso al Videojuego de Izquierda - Derecha.....	43
Tabla 20. Historia de Usuario Número 11. Ingreso al Videojuego de Arriba - Abajo..	43
Tabla 21. Historia de Usuario Número 12. Ingreso al Videojuego Adentro - Afuera..	44
Tabla 22. Historia de Usuario Número 13. Ingreso al Videojuego de Laberinto	45
Tabla 23. Historia de Usuario Número 14. Pantalla de Instrucciones del Videojuego .	45
Tabla 24. Cronograma del Plan de Versiones	46
Tabla 25. Planificación de la Iteración 1	47
Tabla 26. Planificación de la Iteración 2	47
Tabla 27. Planificación de la Iteración 3	48
Tabla 28. Planificación de la Iteración 4	48

Tabla 29. Planificación de la Iteración 5	48
Tabla 30. Planificación de la Iteración 6	48
Tabla 31. Planificación de la Iteración 7	49
Tabla 32. Siglas y Abreviaciones	50
Tabla 33. Características del Usuario	55
Tabla 34. Requerimiento Funcional 1. Seguimiento de la Mano	59
Tabla 35. Requerimiento Funcional 2. Creación de la Pantalla Principal de la Aplicación.....	59
Tabla 36. Requerimiento Funcional 3. Creación del Menú de Videojuegos de la Aplicación.....	60
Tabla 37. Requerimiento Funcional 4. Creación de la Pantalla de Instrucciones del Videojuego.....	60
Tabla 38. Requerimiento Funcional 5. Desarrollo del Videojuego Izquierda - Derecha	61
Tabla 39. Requerimiento Funcional 6. Desarrollo de la Pantalla de Fin de Juego.....	62
Tabla 40. Requerimiento Funcional 7. Desarrollo del Videojuego Arriba - Abajo	62
Tabla 41. Requerimiento Funcional 8. Desarrollo del Videojuego Laberinto	63
Tabla 42. Requerimiento Funcional 9. Desarrollo del Videojuego de Entrenamiento Izquierda	65
Tabla 43. Requerimiento Funcional 10. Desarrollo del Videojuego de Entrenamiento Derecha.....	66
Tabla 44. Requerimiento Funcional 11. Desarrollo del Videojuego de Entrenamiento Arriba.....	66
Tabla 45. Requerimiento Funcional 12. Desarrollo del Videojuego de Entrenamiento Abajo	67
Tabla 46. Requerimiento Funcional 13. Desarrollo del Videojuego Adentro - Afuera	68
Tabla 47. Ítems de Prueba	91
Tabla 48. Caso de Prueba RF-01	91
Tabla 49. Caso de Prueba RF-02	92
Tabla 50. Caso de Prueba RF-03	92
Tabla 51. Caso de Prueba RF-04	92
Tabla 52. Caso de Prueba RF-05	93
Tabla 53. Caso de Prueba RF-06	94
Tabla 54. Caso de Prueba RF-07	94

Tabla 55. Caso de Prueba RF-08	95
Tabla 56. Caso de Prueba RF-09	97
Tabla 57. Caso de Prueba RF-10	97
Tabla 58. Caso de Prueba RF-11	98
Tabla 59. Caso de Prueba RF-12	99
Tabla 60. Caso de Prueba RF-13.1	99
Tabla 61. Caso de Prueba RF-13.2	100
Tabla 62. Caso de Prueba RF-13.3	101
Tabla 63. Caso de Prueba RF-13.4	101
Tabla 64. Ejecución de Pruebas. Primera Vuelta	102
Tabla 65. Ejecución de Pruebas. Segunda Vuelta.	103
Tabla 66. Ejecución de Pruebas. Tercera Vuelta.	104
Tabla 67. Ejecución de Pruebas. Cuarta Vuelta.	105
Tabla 68. Índice de Severidad Total	106
Tabla 69. Población de la Investigación	112
Tabla 70. Cuestionario de Escala de Usabilidad del Sistema	120
Tabla 71. Cuestionario de Comodidad / Facilidad de Uso	121
Tabla 72. Efectividad de las Niñas al Realizar las Tareas	122
Tabla 73. Radio de la Comparación de los Promedios de las Metodologías Analizadas	123
Tabla 74. Resultados de la Encuesta de Escala de Usabilidad	125
Tabla 75. Resultados de los Porcentajes de Satisfacción de Usuario	125
Tabla 76. Comodidad/Facilidad de Uso	126

ÍNDICE DE FIGURAS

Figura 1. Arquitectura Clásica de un Videojuego	14
Figura 2. Dimensiones de la Usabilidad. Estándar ISO 9241-11	18
Figura 3. Jugabilidad como Medida de la Calidad de la Experiencia del Usuario.....	20
Figura 4. Fases de la Metodología ágil de desarrollo de software XP	27
Figura 5. Observación Primero de Básica paralelo "A"	28
Figura 6. Observación Primero de Básica paralelo "B"	29
Figura 7. Observación Primero de Básica paralelo "C"	30
Figura 8. Metodología Lean Startup.....	34
Figura 9. Algoritmo Propuesto	36
Figura 10. Interfaces de Usuario.....	51
Figura 11. Dispositivo Kinect y sus Componentes.....	51
Figura 12. Adaptador USB y Conector de Electricidad	52
Figura 13. Diagrama de Comportamiento del Dispositivo Kinect	55
Figura 14. Distancia del Usuario con Respecto al Kinect	56
Figura 15. Kinect Estado Conectado.	56
Figura 16. Kinect Detectando al Objeto	57
Figura 17. Visión Arquitectónica de la Aplicación	70
Figura 18. Diagrama de Clases.....	71
Figura 19. Diagrama de Paquetes	72
Figura 20. Diagrama de Secuencia del Videojuego de Entrenamiento	73
Figura 21. Diagrama de Secuencia del Videojuego Izquierda - Derecha.....	74
Figura 22. Diagrama de Secuencia del Videojuego Arriba - Abajo	75
Figura 23. Diagrama de Secuencia del Videojuego Adentro - Afuera.....	76
Figura 24. Diagrama de Secuencia del Videojuego Laberinto	77
Figura 25. Diagrama de Despliegue	78
Figura 26. Diagrama de Casos de Uso	78
Figura 27. Prototipo de la Pantalla Inicial	79
Figura 28. Prototipo de la Pantalla del Menú de los Videojuegos	79
Figura 29. Prototipo de la Pantalla de Instrucciones del Videojuego.....	80
Figura 30. Prototipo del Videojuego de Entrenamiento. Actividad Izquierda	80
Figura 31. Prototipo del Videojuego de Entrenamiento. Actividad Derecha	81
Figura 32. Prototipo del Videojuego de Entrenamiento. Actividad Arriba.....	81

Figura 33. Prototipo del Videojuego de Entrenamiento. Actividad Abajo	82
Figura 34. Prototipo de Pantalla de Fin de Juego	82
Figura 35. Prototipo del Videojuego Izquierda – Derecha.....	83
Figura 36. Prototipo del Videojuego Arriba – Abajo	83
Figura 37. Prototipo del Videojuego Adentro - Afuera. Actividad Adentro.....	84
Figura 38. Prototipo del Videojuego Adentro - Afuera. Actividad Afuera.....	84
Figura 39. Prototipo del Videojuego Adentro - Afuera. Actividad Encima.....	85
Figura 40. Prototipo del Videojuego Adentro - Afuera. Actividad Debajo	85
Figura 41. Prototipo del Videojuego Laberinto.....	86
Figura 42. Código del Archivo HandTracking.py	87
Figura 43. Propósito del Programa de HandTracking	87
Figura 44. Mano Cerrada - Evento de Hacer Click	88
Figura 45. Código del Archivo KinectMenu.py	89
Figura 46. Código del Archivo main.py	90
Figura 47. Gráfica de Ejecución de las Pruebas. Primera Vuelta.....	103
Figura 48. Gráfico de Ejecución de las Pruebas. Segunda Vuelta	104
Figura 49. Gráfico de Ejecución de las Pruebas. Tercera Vuelta	105
Figura 50. Gráfico de Ejecución de las Pruebas. Cuarta Vuelta.....	106
Figura 51. Gráfico del Índice de Severidad Total	106
Figura 52. Pantalla Inicial del Videojuego UBIC.....	107
Figura 53. Pantalla del Menú de los Videojuegos	108
Figura 54. Pantalla de Instrucciones del Videojuego Entrenamiento.....	108
Figura 55. Pantalla del Videojuego de Entrenamiento	109
Figura 56. Pantalla del Videojuego Izquierda Derecha	109
Figura 57. Pantalla del Videojuego Arriba Abajo	110
Figura 58. Pantalla del Videojuego Adentro Afuera	110
Figura 59. Pantalla del Videojuego Laberinto.....	111
Figura 60. Pantalla de Fin de Juego.....	111
Figura 61. Tarea 1 (t1). Videojuego Izquierda - Derecha	114
Figura 62. Tarea 2 (t2). Videojuego Arriba - Abajo.....	115
Figura 63. Tarea 3 (t3). Videojuego Adentro - Afuera.....	116
Figura 64. Tarea 4 (t4). Videojuego Laberinto.....	117
Figura 65. Efectividad para Realizar las Tareas t ₁ a t ₄	123
Figura 66. Tiempo Promedio de Desarrollo de las Tareas t ₁ a t ₄	124

Figura 67. Radio de los Tiempos de Desarrollo para las Tareas t_1 a t_4	124
Figura 68. Satisfacción de Usuario a Partir de la Encuesta de la Escala de Usabilidad del Sistema.....	126

UNIVERSIDAD TÉCNICA DEL NORTE
INSTITUTO DE POSGRADO
MAESTRÍA EN INGENIERÍA DE SOFTWARE

**“INTERFACES NATURALES DE USUARIO PARA LA ENSEÑANZA DE
UBICACIÓN ESPACIAL A NIÑOS DE EDUCACIÓN PREESCOLAR:
DETERMINACIÓN DE DIRECTRICES Y DISEÑO DE APLICACIÓN”**

Autores: Mario Roberto Merlo Rosas y Fernando Daniel Recalde Salazar

Tutor: Carlos Xavier Rosero C.

Año: 2019

RESUMEN

La presente investigación se centra en el estudio de como las Interfaces Naturales de Usuario ayudan en la enseñanza de la ubicación espacial a niños de preescolar. Tomando como caso de estudio el desarrollo de varios videojuegos en el lenguaje de programación Python y pygame bajo la plataforma Linux utilizando el dispositivo Kinect como herramienta de accesibilidad a las Interfaces Naturales de Usuario, estos son evaluados con las niñas de los primeros años de educación básica de la Unidad Educativa Fiscomisional Inmaculada Concepción. Se toma como punto de partida la recolección de información referente al tema, por medio de la investigación documental de libros, revistas y artículos científicos, se realiza entrevistas a expertos del tema, a las docentes de la unidad educativa y se hace la observación del método de enseñanza de la ubicación espacial impartido por docentes a las niñas de preescolar. Estos datos recolectados ayudan con la propuesta que tiene XP Programming como metodología ágil para el desarrollo de software. Se inició con la creación de las historias de usuario que permitieron establecer los requerimientos funcionales de los videojuegos que determinaron las directrices de la aplicación, se crea prototipos conceptuales (storyboard) de los videojuegos que son testeados por el personal docente y las niñas de preescolar para poder conocer la aceptación y el uso de dichos prototipos de software, por parte de ellas. Con los resultados se procede al desarrollo de la arquitectura de software usando el modelo de cuatro vistas más uno, para así dar paso a la programación de cuatro videojuegos más uno de práctica. Finalmente, con el prototipo de la aplicación de software desarrollado se realiza la evaluación de la eficiencia, efectividad y satisfacción de usuario con las niñas de preescolar para determinar el grado de usabilidad que tiene la propuesta.

Palabras clave: Interfaces Naturales de Usuario, Ubicación Espacial, Kinect, Metodología XP, Usabilidad, Eficiencia, Efectividad, Satisfacción de Usuario.

UNIVERSIDAD TÉCNICA DEL NORTE
INSTITUTO DE POSGRADO
MAESTRÍA EN INGENIERÍA DE SOFTWARE

**“INTERFACES NATURALES DE USUARIO PARA LA ENSEÑANZA DE
UBICACIÓN ESPACIAL A NIÑOS DE EDUCACIÓN PREESCOLAR:
DETERMINACIÓN DE DIRECTRICES Y DISEÑO DE APLICACIÓN”**

Autores: Mario Roberto Merlo Rosas y Fernando Daniel Recalde Salazar

Tutor: Carlos Xavier Rosero C.

Año: 2019

ABSTRACT

The present investigation focuses on the study of how Natural User Interfaces help in the teaching of spatial location to preschool children. Taking as a case study the development of several videogames in the programming language Python and pygame under the Linux platform using the Kinect device as an accessibility tool to the Natural User Interfaces, these are evaluated with girls from the first years of basic education of the Immaculate Concepción Fiscomisional Educational Unit. The starting point is the collection of information on the subject, through documentary research of books, journals and scientific articles, interviews are conducted with experts on the subject, teachers of the educational unit and observation of the method is made of teaching the spatial location taught by teachers to preschool girls. These collected data help with the proposal that XP Programming has as an agile methodology for software development. It began with the creation of the user stories that allowed to establish the functional requirements of the videogames that determined the guidelines of the application, conceptual prototypes (storyboard) of the videogames that are tested by the teaching staff and the preschool girls were created. be able to know the acceptance and use of such software prototypes, by them. With the results we proceed to the development of the software architecture using the model of four views plus one, in order to give way to the programming of four video games plus one practice. Finally, with the prototype of the software application developed, the evaluation of efficiency, effectiveness and user satisfaction with the preschool girls is done to determine the degree of usability that the proposal has.

Keywords: Natural User Interfaces, Spatial Location, Kinect, XP Methodology, Usability, Efficiency, Effectiveness, User Satisfaction.

CAPÍTULO I

EL PROBLEMA

1.1. Problema de Investigación

1.1.1. Contextualización del Problema

Desde el nacimiento, todas las acciones humanas tienen una realización espacial, para los adultos muchos factores espaciales son irrefutables debido a que ellos han comprendido los elementos del medio en base a la experiencia. Sin embargo, para los niños de educación infantil existen diferentes procesos y factores que intervienen en el desarrollo del conocimiento espacial (Piaget e Inhelder, 2007). Es necesario considerar estos parámetros al trabajar con niños para tener claras las expectativas y exigencias en su formación.

La capacidad espacial de los niños crece desde el conocimiento del espacio perceptual estático e inmediato hasta el conocimiento conceptual verdadero del espacio (Romero, 1977). El primero considera las percepciones de lo que los niños pueden ver o captar a través de sus sentidos, mientras que el segundo comprende la internalización de un sistema de operaciones mentales reversibles que abandona el punto egocéntrico del espacio.

La orientación espacial es una de las habilidades básicas más relevantes en relación al desarrollo del aprendizaje de los niños. Además, es la más compleja en su tratamiento debido a que no es única, sino que comprende un conjunto de capacidades, dentro de éstas se encuentran la memoria visual, la coordinación oculo-manual, la comparación de forma, tamaño y distancia, el razonamiento serial, el giro mental de figuras, la noción de dirección, de posicionamiento y la estructuración del esquema corporal; y depende también, en gran medida, del proceso de lateralización y del desarrollo psicomotor (Jiménez, 2009).

Por otro lado, la educación inicial se ha constituido de mayor importancia en el desarrollo integral del niño (Gutiérrez, 2010), en las aulas se trabaja constantemente en la creación de materiales novedosos para que el niño se interese por cada una de las actividades,

construyendo su propio conocimiento a través del uso de tecnología. Es necesario considerar que siempre se deben respetar los tres ejes de la educación inicial: juego, arte y afectividad (Cabezas, 2014).

El material educativo se convierte en una herramienta valiosa de motivación y apoyo, porque promueve el aprendizaje autónomo al aproximar el material de estudio al alumno, a través de diversos recursos didácticos (Aguilar, 2011). El material para los niños debe de ser novedoso y creativo, cada material o juguete le ayuda en su expresión creadora, estimulando sus capacidades y potencialidades (Parreño, 2016).

La necesidad incide en la motivación y cuanto más motivado esté un niño más aprovechará los recursos de que dispone para aprender, más estrategias utilizará para conseguir su objetivo y cuantas más utilice, más rápido y mejor aprenderá (Sánchez, 2008). Por lo tanto, la actividad lúdica es atractiva y motivadora, capta la atención de los niños hacia la materia (Andreu y García, 2000).

Cuando se utilizan interfaces naturales de usuario (NUI), el niño puede utilizar las competencias adquiridas para comunicarse con otros objetos y personas de su entorno, y así interactuar de mejor manera con el computador, para hacer sentir a la computación mucho más intuitiva y expandir las formas de interactuar y experimentar la tecnología (Duque y Vásquez, 2015).

Kinect permite interactuar a los usuarios con el computador sin la necesidad de tener un contacto físico mediante una interfaz natural de usuario que reconoce gestos, comandos de voz, y objetos e imágenes, puede propiciar prácticas pedagógicas que generen la adquisición de una fuerte inteligencia cinética-corporal por parte del estudiantado. Kinect se presenta como una herramienta con gran potencial para mejorar la interacción en el aula, de manera que el docente pueda manipular sus materiales didácticos digitales mediante la utilización de nuevas e interesantes formas de interactuar con el computador, este proceso motiva a los estudiantes y captura su atención (Lozada, Molina y Guffante, 2015).

El desarrollo de estas interfaces en los últimos años ha tomado fuerza en la educación, por lo tanto, se sigue explorando los diversos campos de aplicación. En diversas áreas se

han logrado grandes avances como lo son los robots inteligentes los cuales pueden interactuar con los seres humanos respondiendo a sus comandos y gestos corporales. En la educación ha ayudado al desarrollo integral psicomotriz y la lingüística. Las NUI han ayudado en el desarrollo de niños y jóvenes autistas a desarrollar sus habilidades sociales y movimientos corporales (Ibarra, González, Pulido, Rodríguez y Sánchez, 2017).

1.1.2. Planteamiento del Problema

Los niños que cursan los primeros años de educación básica no han desarrollado completamente su ubicación espacial. Además, las herramientas utilizadas por los profesores para este propósito, no despiertan el interés de los niños.

Así, es necesario desarrollar este conjunto de habilidades por medio de la enseñanza lúdica.

Actualmente, científicos crean Interfaces Naturales de Usuario (NUI, Natural User Interfaces) orientadas a la educación básica de niños entre 4 y 6 años de edad, sin embargo, no están claras las directrices que se puedan seguir en el momento de desarrollarlas. Además, no existe un estudio formal de usabilidad (considerando que el uso que da el usuario es diferente al uso para el cual la interfaz fue diseñada).

1.1.3. Proposición

El uso de interfaces naturales de usuario mejorará el desarrollo de la ubicación espacial de niños en preescolar.

1.1.4. Variables

Variable Independiente: El uso de Interfaces Naturales de Usuario.

Variable Dependiente: Desarrollo de la ubicación espacial de niños en preescolar.

1.2. Objetivos de la Investigación

1.2.1. Objetivo General

Determinar un conjunto de directrices a seguir en el diseño y creación de interfaces naturales de usuario para el desarrollo de la ubicación espacial en niños de educación preescolar.

1.2.2. Objetivos Específicos

- Fundamentar la investigación utilizando información bibliográfica.
- Crear un conjunto de directrices a seguir en el diseño y la realización de bosquejos para las interfaces gráficas.
- Diseñar e implementar algoritmos para mejorar el desarrollo de la ubicación espacial en niños de preescolar en base a cámaras de profundidad.
- Evaluar la usabilidad de las interfaces desarrolladas considerando eficacia, eficiencia y satisfacción de los niños.

1.3. Justificación

La orientación espacial es una de las funciones básicas fundamentales para el desarrollo holístico de los niños, puesto que un buen fortalecimiento de esta neurofunción permite una base sólida sobre la que se incrementará los demás conocimientos a adquirir durante su vida cotidiana y académica, además un proceso consolidado de esta función evita muchos problemas en el desempeño de las funciones diarias y en el proceso de enseñanza aprendizaje.

Esta investigación se fundamenta en el estudio de la usabilidad de interfaces basadas en visión artificial, puesto que las interfaces tradicionales tienen un grado de dificultad y necesitan un aprendizaje previo para su utilización.

Se abre una brecha del cambio en la educación tradicional conjuntamente con la tecnología, por tanto, se pretende estimular al niño de forma divertida con el fin de mejorar su aprendizaje en la orientación espacial.

CAPÍTULO II

MARCO REFERENCIAL

2.1. Antecedentes

“La relación hombre-máquina es un tema evaluado desde el inicio del desarrollo de la tecnología, motivo de estudio que pretende facilitar el uso de los dispositivos y mejorar la comunicación que se da con ellos. El desarrollo y la integración de diferentes tecnologías han permitido que se genere un cambio en la interacción con estos dispositivos creando una interacción diferente. Aunque queda en manos del Ingeniero de Software la responsabilidad de determinar la distribución en pantalla de los diferentes elementos es evidente el desconocimiento en lo que se refiere a las normas y estudios que facilitan la creación de interfaces eficientes y ergonómicas.” (Galeano, 2014).

Al revisar la bibliografía sobre la temática propuesta, fue investigada por diferentes universidades, en la Escuela Politécnica Nacional del Ecuador, Facultad de Ingeniería Eléctrica y Electrónica en el año 2014 los Ingenieros María Teresa Calderón y Diego Javier Maldonado con el tema: “Control por visión de un cuadrúpedo utilizando ROS” empleando OpenCV concluyeron; que mediante el uso de recursos y herramienta contenidos en el framework ROS, es posible desarrollar diferentes aplicaciones para el seguimiento de objetos , ya sea por su color o forma básica, pues el framework ROS se caracteriza por apoyar a la reutilización de código para la investigación y desarrollo de la robótica; también recomienda implementar un algoritmo de procesado muy robusto y sensible que permita el desarrollo de aplicaciones en entornos poco controlados. (Calderón y Maldonado, 2014).

En la Universidad Carlos III de Madrid, Facultad de Ingeniería en Sistemas y Automática en el año 2013 el Ingeniero Álvaro Loras con el tema: “Percepción de la actividad de un conductor mediante cámaras” empleando Lenguaje C++ concluye que para diferenciar entre un brazo y otro se segmentó sus centroides por profundidad y filtrado, que el proyecto funciona mejor en ambientes de interior que de exterior ya que esto compromete el cálculo computacional para que el resto de la aplicación funcione correctamente en tiempo real. (Loras, 2013)

En el trabajo de Maestría elaborado por Erap Samet (2012) denominado “Gesture Based PC Interface with Kinect Sensor” concluye que la aplicación desarrollada ofrece la capacidad de controlar la computadora con un clic y sin contacto de manera que puede encontrar aplicaciones en muchas áreas de la industria del entretenimiento o en un campo tan delicado como la medicina. Los resultados logrados han demostrado que la ideología propuesta no solo garantiza el mismo nivel de interacción que el teclado y el mouse, sino que también puede ajustarse para las necesidades específicas de diferentes áreas de aplicación. Que el sensor Kinect proporciona una valiosa alternativa a los sistemas de captura de movimiento más costosos. (Erap, 2012)

En el estudio realizado por (González, Padilla, Gutiérrez y Cabrera, 2008) titulado “De la Usabilidad a la Jugabilidad: Diseño de Videojuegos Centrado en el Jugador” concluyen que: La usabilidad no es suficiente para determinar la experiencia del jugador. Por ello se presenta el concepto de jugabilidad como extensión de la usabilidad para videojuegos.

De todo lo expuesto el enfoque de la presente investigación pertenece al área de la interacción humano-máquina. Su objetivo principal consiste en el desarrollo e implementación de una interfaz basada en gestos que permite al usuario interactuar con una computadora equipada con sensor Kinect, evitando el uso del teclado, mouse o cualquier otro dispositivo que requiera contacto físico, para así crear una herramienta que sirva de refuerzo en la enseñanza del aprendizaje de la ubicación espacial en los niños de preescolar.

2.2. Fundamentación Tecnológica

Es considerablemente reconocido que hoy en día las computadoras juegan un rol especial en el proceso del aprendizaje. El desarrollo de la tecnología de información y comunicación (TIC), unido a los esfuerzos contemporáneos en el desarrollo de nuevos sistemas de enseñanza-aprendizaje, ha influido en la adopción de tecnologías para el desarrollo de aplicaciones educativas. Esto es debido entre otros factores a la sofisticación incremental de las tecnologías multimedia e interactivas, así como la necesidad de aumentar la flexibilidad en el acceso a la educación. (Alva, 2005)

Varios estudios proponen el uso de aplicaciones de software dentro del proceso de enseñanza-aprendizaje, de diversas formas, como el uso de los videojuegos aplicados a la educación básica como herramienta didáctica para el desarrollo de habilidades cognitivas (Contreras, 2014), algunos videojuegos son escenarios adecuados cuando se asocian a actividades que dirigen la atención de los niños hacia el aprendizaje de propiedades y relaciones espaciales. (Macías y Quintero, 2011)

O el diseño e implementación de videojuegos como herramienta de aprendizaje, proponiendo el diseño de juegos adaptados para personas con dificultades en la lectura, problemas de autismo o de parálisis cerebral aplicando las nuevas tecnologías del juego como mecanismos de interacción para superar las barreras de la discapacidad. (González, Cabrera y Gutiérrez, 2014)

2.3. Fundamentación Legal

El uso de las Tecnologías de la información en la Educación, dentro de la constitución de la República del Ecuador (2008) se determinó que los siguientes artículos son de interés:

Art. 16.- Todas las personas, en forma individual o colectiva, tienen derecho a:

2. El acceso universal a las tecnologías de información y comunicación. (Const., 2008)

Art. 347.- Será responsabilidad del Estado:

8. Incorporar las tecnologías de la información y comunicación en el proceso educativo y propiciar el enlace de la enseñanza con las actividades productivas o sociales. (Const., 2008)

De acuerdo a lo que la Constitución de la República del Ecuador (2008), establece tiene una relación con el Código de la Niñez y la Adolescencia en su artículo 37, en cuanto al derecho que tienen todos los niños a una educación de calidad en la cual se deberá utilizar herramientas y Metodologías adecuadas para que cumplan con cabalidad los objetivos de los programas de educación planteados en el artículo 38 del presente código. (Código de la Niñez y Adolescencia, 2003)

Es de suma importancia definir que la investigación se realizó en la etapa de educación primaria en la cual los estudiantes desarrollan sus destrezas. Con lo cual es pertinente complementar lo expuesto en la Constitución con lo que establece la Ley Orgánica de Educación Intercultural LOEI en su artículo 2, que habla de los principios generales de la actividad educativa, en su literal h donde contempla el interaprendizaje y multiaprendizaje que considera a los instrumentos tecnológicos como una ayuda en el refuerzo de la educación logrando alcanzar en los niños un desarrollo personal y colectivo. (LOEI, 2015)

En el Art. 6 de la LOEI, se establece las obligaciones que tiene el estado en garantizar la educación la calidad, en el literal j compete el de garantizar la alfabetización digital y el uso de las tecnologías de la información en el proceso educativo. (LOEI, 2015)

En referencia a las destrezas que las niñas de primeros años de educación básica de entre 4 a 6 años de edad deben desarrollar el Currículo de EGB y BGU establece que “las tecnologías de la información y de la comunicación formarán parte del uso habitual como instrumento facilitador para el desarrollo del currículo.” (Ministerio de Educación, 2016)

En cuanto a los ámbitos de desarrollo y aprendizaje que se deberían tomar en cuenta para la presente investigación el Currículo de EGB y BGU establece:

Relaciones lógico matemática:

M.1.4.2. Reconocer la posición de objetos del entorno: derecha, izquierda.

M.1.4.3. Reconocer la derecha e izquierda de los demás.

M.1.4.4. Distinguir la ubicación de objetos del entorno según las nociones arriba/abajo, delante/atrás y encima/debajo.

Comprensión y expresión oral y escrita:

LL.15.15. Utiliza recursos digitales para satisfacer sus necesidades de ocio y aprendizaje. (Ministerio de Educación, 2016)

2.4. Referentes Teóricos

2.4.1. Noción Espacial

Es de vital importancia que las personas desde temprana edad deban desarrollar los conceptos espaciales para ubicar de una mejor manera en el espacio que nos rodea, debido a que esto nos ayuda en variadas actividades profesionales, como, por ejemplo: un piloto de avión, topógrafo, ingeniero, arquitecto, etc., o en otras profesiones en las cuales la relación no es tan evidente como los cirujanos que requieren orientarse dentro del cuerpo del paciente cuando son intervenidos, para localizar el órgano o la parte enferma a ser tratada. (Macías y Quintero, 2011).

La orientación espacial es importante en la construcción del espacio por los niños, tanto para su vida en sociedad como para sus futuros aprendizajes. (Saiz, 1998). Sin embargo, en el actual currículo de Educación General Básica solo se encuentra presente en el Nivel Inicial o Primer Grado, en cuanto a lo que tiene que ver con las relaciones lógico matemáticas. En las cuales se incentiva al niño a que desarrolle las destrezas con criterios de desempeño, verificando su realización con indicadores de logro. (Ministerio de Educación, 2016)

La orientación espacial en los niños es una habilidad básica que tiene que ver con la capacidad de aprendizaje, lo cual es parte del desarrollo psicomotor y se manifiesta en la escritura, lectura y memorización.

Detrás de, al lado de, arriba, abajo, en el frente, son algunas expresiones necesarias y que utilizamos en nuestro día a día. Por esta razón, resulta indispensable que los niños diferencien cada expresión y logren ubicarse sin dificultad.

2.4.2. Conceptos Espaciales

Se podría definir a la orientación espacial como la capacidad que posee el individuo de adaptarse al espacio que lo rodea a partir de la organización de su propio cuerpo, como respuesta a su percepción visual, auditiva y táctil-kinestésica, para comprender y llegar a dominar el concepto de orientación espacial, significa que ya hemos adquirido una serie

de nociones espaciales que expresan las diferentes posiciones y orientaciones que puede tener un objeto o un cuerpo.

Para Piaget, “la adquisición de la noción espacio está ligada intrínsecamente a la adquisición del conocimiento de los objetos, por lo que un niño de meses ya empieza a desarrollarlo a través del desplazamiento de objetos” (Holloway, 1986), Piaget desde la concepción del ser humano ha clasificado la noción espacial en tres etapas:

Espacio Topológico: Que inicia desde el nacimiento hasta los tres años de vida, comprende en la limitación del campo visual y motricidad en sus inicios. Como convivencia, diferenciación de objetos, orden de objetos, secuencia lógica de ubicación.

Esquema Euclíadiano: Comprende entre los tres y siete años de edad, en esta etapa ya va consolidando las nociones y esquemas corporales. Como Tamaño (Pequeño, Grande), dirección (Delante, Atrás), orientación y lateralidad (Izquierda y derecha).

Espacio Proyectivo y Racional: A partir de los siete años de edad el niño desarrolla el pensamiento y espacio y representa mentalmente su lateralidad izquierda y derecha. (Piaget, 1948)

Para poder ayudar al niño a que desarrolle el concepto de ubicación espacial, en la etapa de educación inicial se le debe favorecer con diferentes actividades lúdicas que fomente una buena base de conocimientos sobre esta temática.

En la teoría de desarrollo de los esquemas evolutivos de Piaget advierte que la noción espacial, se adquiere a través de la propia experiencia con el medio y del aprendizaje lingüístico. En la etapa de educación inicial, se deberá iniciar al niño en las siguientes nociones espaciales:

La orientación de un objeto o del propio cuerpo, ya que todo objeto se orienta a través de 3 dimensiones: La lateralidad (izquierda o derecha), la profundidad (delante o detrás) y la anterioridad (antes y después).

La posición en el espacio en relación a la interioridad, estar dentro de un espacio o la exterioridad, estar fuera de un espacio. (Piaget, 1948)

2.4.3. Pensamiento Infantil

El niño de entre los 0 - 6 años de edad, presentan varias limitaciones que los maestros deben conocer y tener presente al momento de su educación, como el egocentrismo que tiene el niño, que hace que sólo piense en él e ignore otros puntos de vista (Flavell, 1982), además desconocen el razonamiento deductivo. Todo esto conlleva a que existan diferencias entre el pensamiento de un adulto y de un niño.

Piaget propone una serie de etapas de desarrollo en los seres humanos donde en cada periodo se caracteriza por la presencia de ciertos procesos y estructuras mentales. Estas etapas empiezan desde el nacimiento, durante los 2 primeros años, el niño aún no tiene nociones de ubicación, desde los 2 hasta los 4 años, adquiere ciertas nociones como diferenciación de distancias formas.

Tomando en cuenta lo anterior podemos concluir una serie de pautas o metodologías que un maestro de Primer Año de Educación Inicial debe respetar para que el desarrollo del niño sea el correcto, y así favorezca su desarrollo global.

1. Hay que tener en cuenta las diferentes etapas evolutivas por la que pasan los niños y respetar su ritmo madurativo. Es decir, hay que adaptar las actividades a su nivel cognitivo, según vayan desapareciendo sus limitaciones, y no exigir un nivel superior.
2. En el Primer Año de Educación Inicial, el niño se encuentra en la Etapa Preoperacional de Piaget, por lo que debemos dar importancia a las siguientes características: centración, conservación y egocentrismo. Ya que influyen directamente en el desarrollo espacial.
3. Hay que trabajar el espacio complementando dos planos: uno sensorio-motriz (a través de juegos relacionados con el esquema corporal) y otro intelectual (verbalizando las acciones corporales y dibujos que representen dichas acciones). (Fernández y Ramiro, 2015)

2.4.4. Videojuegos en la Educación

Las personas a lo largo de los años han podido administrar su ocio y hacer de él un factor de desarrollo cultural, ya que está demostrado que los juegos pueden ayudar en el

desarrollo de cualidades o habilidades psicológicas, evidenciando así la incorporación de estos en los salones de clase como una herramienta de reforzamiento en el proceso de aprendizaje. (Valiño, 2002)

Los nuevos enfoques pedagógicos se ven apoyados en las tecnologías y en los videojuegos como herramientas que favorecen los nuevos modelos de enseñanza-aprendizaje. Lo que nos lleva a empezar a definir una nueva disciplina denominada *Game Based Learning*(GBL). (Prensky, 2001) La misma que destaca tres aspectos interesantes en el uso de videojuegos dentro del modelo de aprendizaje:

1. En GBL convergen las necesidades y el estilo de los estudiantes actuales.
2. GBL es motivador y divertido.
3. GBL es versátil, debido a que se adapta a cualquier materia, información o habilidad.

Existen amplias posibilidades de enseñanza a partir del uso de videojuegos, también hay que considerar que los docentes van a encontrar una serie de limitaciones al momento de incorporarlos en el salón de clase como puede ser:

- Una necesidad de poseer una infraestructura específica.
- El problema de desarrollar metodologías específicas para poder realizar una implementación eficaz.
- Deberán poseer una cierta formación y experiencia para poder utilizarlos efectivamente.
- Se presenta una dificultad al momento de ajustar los horarios de clase con el tiempo que se va a dedicar al juego.
- Los contenidos que proponen los videojuegos comerciales no están adaptados al currículo escolar.

Lo que ha conllevado a la necesidad de tener nuevos materiales didácticos que apoyen en el proceso de una enseñanza tradicional, es de aquí donde aparecen los llamados Serious Game que son diseñados específicamente para la enseñanza-aprendizaje en los centros educativos. (Castro y López, 2013)

“Los Serious Game son juegos con una intención educativa, que se encuentran diseñados y concebidos para que produzcan el mismo impacto que los videojuegos de entretenimiento de mayor éxito. Estos valoran en su diseño la capacidad de divertir y entretener como base del aprendizaje, utilizan los mismos criterios en cuanto al diseño

narrativo y la jugabilidad que los videojuegos comerciales y también son similares en cuanto al formato o los lenguajes” (Gil y Vida, 2007)

Los videojuegos se han convertido en mecanismos de mucha influencia a nivel cultural y social tanto en niños, adolescentes y adultos. Siendo así una gran ayuda en varios aspectos de la vida social. Los Serious Game se los puede clasificar como:

- **Los Edutainment.** – Son videojuegos que su objetivo principal es enseñar a las personas.
- **Training Game.** - Ayudan a entrenar al jugador en una serie de hábitos que sean prácticos.
- **News Game.** - Son medios de comunicación que ayudan a difundir algún mensaje informativo a través de un juego.
- **Advergaming.** – Son videojuegos que sirven como ayuda en la emisión de campañas publicitarias.

El uso de los videojuegos en el proceso de enseñanza puede aportar muchas ventajas:

- Éxito escolar
- Habilidades cognitivas
- Motivación
- Atención y Concentración

Los videojuegos educativos apoyan de una manera sustancial a la tarea de enseñanza-aprendizaje, debido a que estos son desarrollados como juegos multimedia en los cuales su objetivo es enseñar de forma lúdica. (Padilla, González, Gutiérrez, Cabrera y Paderewski, 2009)

Si se potencia varios factores como la motivación, atención, concentración y emoción del jugador pueden convertir al videojuego en una gran herramienta de reforzamiento pedagógico en el proceso de enseñanza-aprendizaje de algo tan importante como es la ubicación espacial.

2.4.5. Arquitectura y Diseño de un Videojuego

Un videojuego está construido, básicamente en una arquitectura en capas constituidas de la siguiente forma como se muestra en la Figura 1:

- **La Mecánica del Juego.** - Aquí se encuentran los elementos más importantes de un videojuego. Se concreta el género al cual pertenece como *arcade*, simulación, plataformas, rol, etc.; se definen las reglas, los objetivos y la forma en la cual se logrará conseguirlos. Debe contener los elementos de la historia que se quiere contar o como se la va a contar.
- **El Motor del Juego.** - Esta tiene referencia a una serie de rutinas que permitirán la ejecución de todos los elementos que componen el videojuego. Aquí se controlará como se presenta cada elemento del juego y como se interactúan con ellos.
- **La Interfaz del Juego.** - Esta capa se encarga de la interacción directa del jugador con el videojuego para mantener un diálogo fluido entre los dos, presentando todos los contenidos, opciones, escenas del mundo virtual, y sus controles necesarios para permitir la interacción. Es un elemento muy importante, y deberá de ser lo más sencillo y familiar a la naturaleza del videojuego, debido a que influye en una gran medida la jugabilidad final del mismo. (González, Padilla, Gutiérrez y Cabrera, 2008)



Figura 1. Arquitectura Clásica de un Videojuego

Nota: Adaptado de González, Padilla, Gutiérrez y Cabrera (2008).

2.4.6. Interfaz Natural de Usuario (NUI)

Las Interfaces Naturales de Usuario (NUI) se constituyen como nuevos métodos para la Interacción Humano Computador (HCI) y el diseño de aplicaciones informáticas basadas en interfaces, con las cuales las interacciones se realizan a partir de las acciones naturales de los seres humanos, tal y como éstos realizan sus actividades en el mundo físico, sin la necesidad de periféricos para ingresar los datos, aprovechando de esta forma los conocimientos que sobre este entorno tienen los seres humanos de manera innata.

Para interactuar con sistemas basados en NUI se han venido utilizando diversas modalidades de entrada, tales como el tacto, reconocimiento de gestos, seguimiento de movimientos, comandos de voz, entre otros. (Lozada, Rivera y Molina, 2014)

Todos al momento de usar una máquina, un dispositivo o PC están obligados a usar una interfaz externa para interactuar con ellos, esta interfaz puede ser de fácil uso, pero la forma de comunicación entre la interfaz y el ser humano no es innata esto quiere decir que las personas tienen que aprender la manera de usar la interfaz.

Pero la NUI es controlada por la interacción natural humana, por los movimientos naturales de las personas, por los gestos que realizan y por la voz.

Desde la década de 1970 se desarrolla una serie de estrategias de interfaz para el usuario que utilizan la interacción natural con el mundo real, como una alternativa de la interfaz de línea de comandos (CLI) o de la interfaz gráfica de usuario (GUI). En la CLI, los usuarios tenían que aprender un medio artificial de entrada, el teclado, y una serie de insumos codificados, que tenían un rango limitado de respuestas, donde la sintaxis de los comandos era estricta. Luego, el ratón activó la interfaz gráfica de usuario, los usuarios pueden aprender fácilmente los movimientos y las acciones del ratón. La GUI se basó en metáforas para interactuar con el contenido o los objetos en pantalla. La Interfaz Natural del Usuario (NUI) se basa en redes neuronales artificiales de algoritmos complejos que describen de forma acertada y muy exacta las medidas del cuerpo, color de la piel, etc., estas características permiten que a través de una cámara se pueda obtener información de las imágenes tomadas, luego se procede a un entrenamiento neuronal de la red que puede llegar a identificar a los seres humanos, sus movimientos y sus gestos. Esta NUI es

usada e investigada por desarrolladores y diseñadores de software para mejorar la experiencia al usuario (Microsoft, 2017). Entre los equipos que permiten realizar están:

El sensor Kinect fue desarrollado por Microsoft, nació como un nuevo controlador para la videoconsola Xbox 360, permite manejar la video-consola sin necesidad de un controlador tradicional o mando, puede controlarla solamente con movimientos de su cuerpo y órdenes de voz (Microsoft, 2017).

La próxima generación de interacción hombre-máquina, NUI permite a las personas interactuar con cualquier dispositivo, en cualquier lugar, utilizando los movimientos y el lenguaje que utilizan todos los días en sus vidas. Las aplicaciones de Microsoft Kinect para Windows abren una amplia gama de nuevas posibilidades para que las personas interactúen con las computadoras de una manera que se sienta natural. De los negocios a las artes, de la educación al juego, y más allá, NUI amplía los horizontes de desarrollo de aplicaciones.

2.4.7. Ventajas de las Interfaces Naturales de Usuario (NUI)

Los dispositivos de entrada modernos hacen más flexibles y capaces a las NUI y al enfocarse en comportamientos naturales hacen más fáciles a estas de entenderse y aprenderse, pero en base a estos argumentos se debe estar consciente de que no siempre implementar una interfaz de este tipo es lo adecuado y nacen otras interrogantes y lineamientos a considerar al desarrollar una interfaz de este tipo (Ibarra, González, Pulido, Rodríguez y Sánchez, 2017). Algunas ventajas de utilizar NUI:

- Es una interfaz diseñada para utilizar habilidades aprendidas anteriormente para interactuar con contenido.
- Está determinada en base a su estilo de interacción y no se limita o define por algún dispositivo de entrada en particular.
- Pueden incorporar elementos como voz y movimientos corporales para controlar aplicaciones.
- La gran exposición a estas tecnologías en la juventud los hace casi natos en el entendimiento y aprendizaje de estas interfaces, así como crea un vínculo a temprana edad.

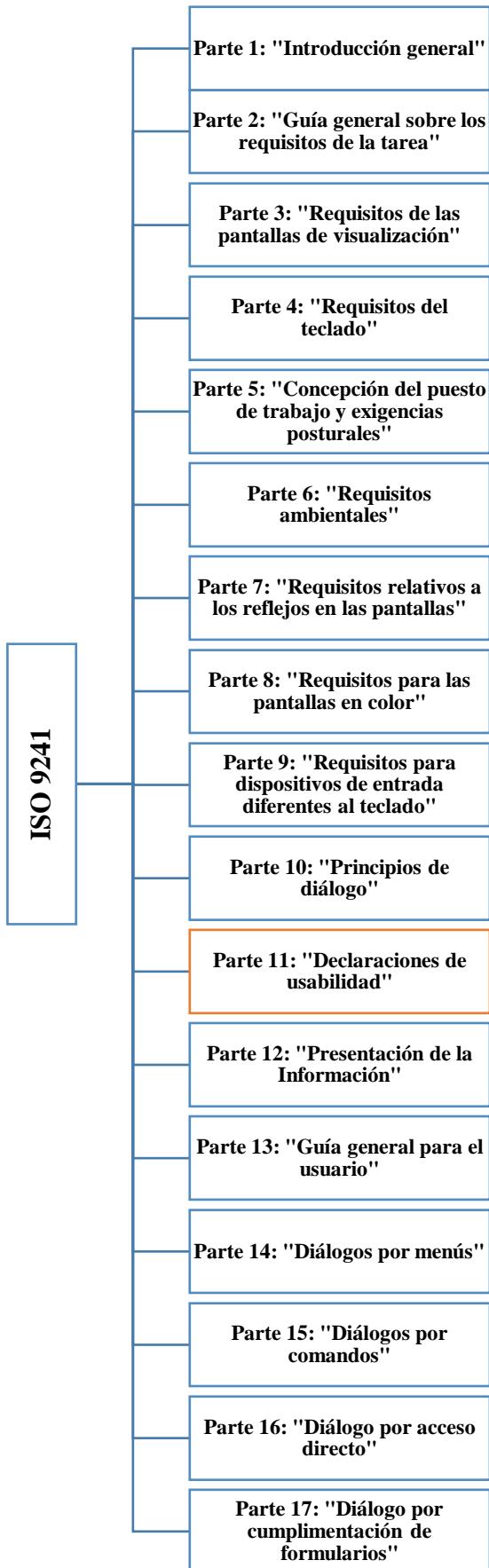
2.4.8. Usabilidad

La usabilidad nos describe como los usuarios emplean un software en particular. El estándar ISO 9241-11:2018 que forma parte de la serie ISO 9241 que está orientado a la calidad en usabilidad y ergonomía en HW y SW, define la usabilidad como "la medida en la que un producto se puede usar por determinados usuarios para conseguir objetivos específicos con efectividad, eficiencia y satisfacción en un contexto de uso especificado".

La usabilidad es cada vez un atributo más importante del software, se puede decir que un sistema es usable cuando es fácil de aprender y fácil de utilizar, en resumen, se diría que la usabilidad se define como el usuario emplea el software. La Organización Internacional de Normalización (ISO) establece que la usabilidad de un producto es la extensión a la cual el usuario puede emplearlo para lograr sus objetivos con eficacia, eficiencia y satisfacción. (Brooke, 1996)

Algunos autores, como Beltré Ferreras, en sus trabajos resaltan que la definición del estándar ISO 9241-11 (Figura 2) contiene en su norma una visión sobre la aceptabilidad de un producto, la cual se fundamenta en que para evaluar la usabilidad de un software se debe garantizar de que el software sea interactivo, fácil de aprender, efectivo de usar y agradable desde la perspectiva del usuario. Por lo tanto, los atributos de usabilidad a evaluar serán:

- **Eficacia:** Representa la exactitud con la cual los usuarios alcanzan sus metas especificadas, esto se refiere a cuan preciso puedan los usuarios realizar sus tareas logrando completar sus objetivos.
- **Eficiencia:** Los recursos gastados con relación a la certeza con la cual los usuarios logran las metas. Se considera la cantidad de esfuerzo requerido para alcanzar el nivel de efectividad cuando el usuario logra los objetivos. Es la relación entre el nivel de efectividad y el consumo de recursos.
- **Satisfacción:** la comodidad y la aceptabilidad del uso. Se establece qué tan cómodo se siente el usuario al usar la interfaz de software.



Dimensiones de Usabilidad

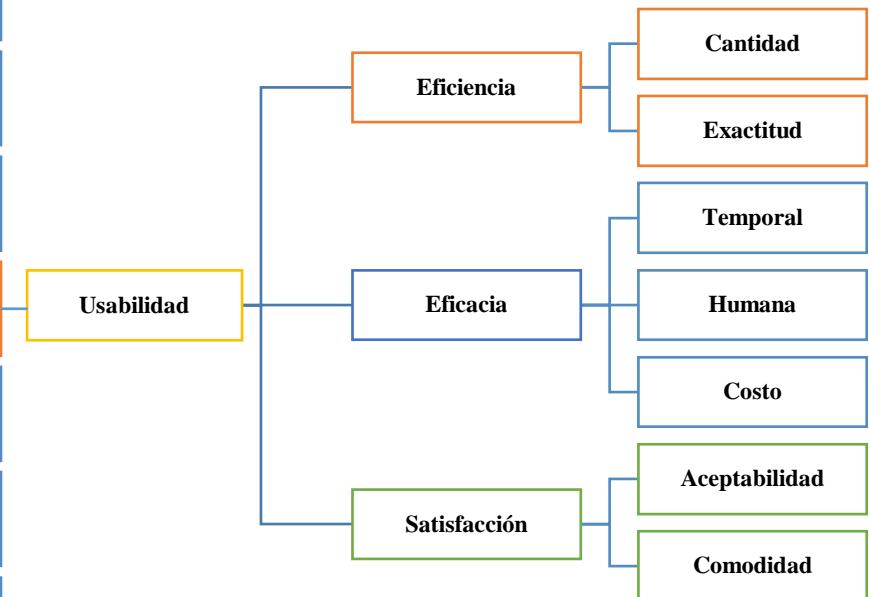


Figura 2. Dimensiones de la Usabilidad. Estándar ISO 9241-11

Nota: Adaptado de Beltré (2008).

De manera general, según los estudios realizados por Beltré Ferreras la ISO 9241-11 separa la usabilidad de la calidad del trabajo; se centra en el producto y no en el usuario.

Nielsen define la usabilidad como un atributo de calidad que mide cuán fáciles de usar son las interfaces de usuario, además de la utilidad y capacidad de un sistema para satisfacer las necesidades del usuario. (Nielsen, 1994)

La satisfacción del usuario y el correcto y eficiente desempeño de su trabajo es lo que determina el grado de aceptación de un producto y por tanto su usabilidad.

2.4.9. Calidad Interactiva y Experiencia de Usuario

La norma ISO 8402 define a la calidad como “la aptitud de un producto o servicio para satisfacer las necesidades del usuario” (ISO-8402-92, 1992).

La calidad de un videojuego viene dada por dos factores: el primero cubre la utilidad funcional y el otro se preocupa de como los usuarios hacen uso de dicha funcionalidad. Esta medida es mucho más cercana a la calidad propia y algunos estándares la denominan Calidad en Uso. (ISO/IEC 25010, 2011)

2.4.10. Calidad en Uso

La Calidad en Uso surge de la evolución de las propiedades de la usabilidad dentro de los diversos estándares de calidad: ISO/IEC 9241-11: 2018 y la ISO/IEC 9126-1: 2001. Tomando en cuenta diversos atributos como la eficiencia, efectividad y satisfacción de usuario en la experiencia interactiva, con el objetivo de proporcionar una cantidad de atributos medibles y una serie de métricas que ayuden a caracterizar la experiencia subjetiva interactiva de una manera tangible y objetiva; y la parte dinámica de la UX dependiente del contexto del usuario. (González y Gutiérrez, 2014)

En los videojuegos la base del éxito se da en asegurar un correcto grado de calidad de la experiencia interactiva. Una de las medidas usadas es analizar el grado de jugabilidad como se muestra en la Figura 3.

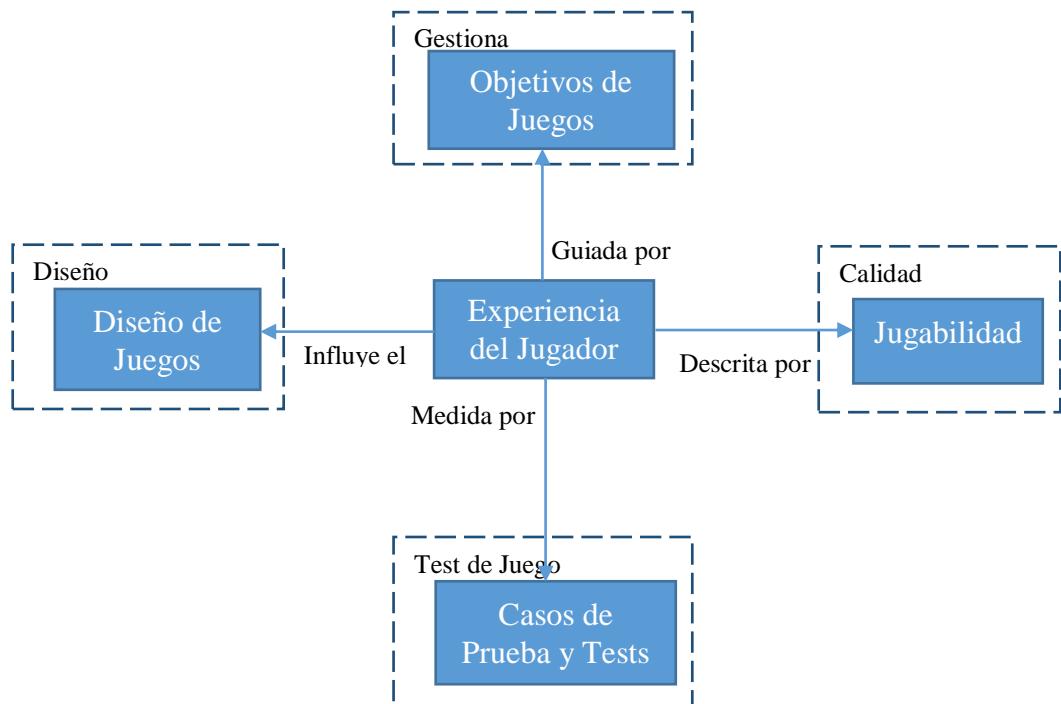


Figura 3. Jugabilidad como Medida de la Calidad de la Experiencia del Usuario.

Nota: Tomado de González y Gutiérrez (2014).

2.4.11. Jugabilidad en Videojuegos

La jugabilidad es un término empleado en el diseño y análisis de videojuegos, el cual es empleado para medir la calidad de la experiencia del juego, en los términos de las mecánicas de juego, los elementos de software implementados y del proceso entre el jugador y el videojuego.

A la jugabilidad se la define como “el conjunto de propiedades que describen la experiencia del jugador ante un sistema de juego determinado, cuyo principal objetivo es divertir y entretenir de forma satisfactoria y creíble ya sea solo o en compañía”. (González y Gutiérrez, 2014)

CAPÍTULO III

MARCO METODOLÓGICO

1.1. Descripción del Área de Estudio

La investigación se realizó en la “UNIDAD EDUCATIVA FISCOMISIONAL LA INMACULADA CONCEPCIÓN” (UEFLIC).

1.1.1. Misión

La Unidad Educativa "La Inmaculada Concepción" es una institución educativa particular católica. Bajo la protección de María Inmaculada y el Carisma de San Vicente de Paúl y Santa Luisa de Marillac ofrece una educación integral a la niñez y juventud. Forma mujeres cristianas, críticas, emprendedoras y solidarias con amor afectivo (calidez) y efectivo (calidad), promoviendo desde la doctrina Cristo - céntrica, la libertad de pensamiento, la dignidad del ser humano, pilares fundamentales sobre los cuales se sustenta el buen vivir evangélico.

1.1.2. Visión

La Unidad Educativa "La Inmaculada Concepción" en el 2018 será un referente de excelencia académica, cumpliendo con los estándares de calidad establecidos por el Ministerio de Educación, coherente con el Carisma Vicenciano, con una Comunidad Educativa dotada de pensamiento fluido, creativo, innovador, con propuestas visionarias y audaces, capaces de resolver sus problemas y del entorno.

1.1.3. Ubicación

La unidad Educativa se encuentra ubicada en el país Ecuador, provincia de Imbabura, ciudad de Ibarra, parroquia San Francisco, calles: Simón Bolívar y Pedro Moncayo 5-45.

1.1.4. Beneficiarios

Los principales beneficiarios son las niñas de primeros de básica entre 4 a 6 años de edad quienes no tienen nociones de ubicación espacial y docentes parvularios las cuales se les brindara una herramienta de software para reforzar la enseñanza.

1.2. Diseño y Tipo de Investigación

1.2.1. Tipo de Investigación

El enfoque de la tesis es cuantitativa porque se recogen y analizan datos sobre variables discretas. Y cualitativa porque se hacen registros narrativos de los fenómenos que son estudiados mediante técnicas como la observación participante obteniendo variables nominales.

1.2.2. Diseño de la Investigación

1.2.2.1. Modalidad de la Investigación

La investigación es bibliográfica porque se utilizan fuentes como: libros, documentos, artículos, revistas, etc. Para la construcción del marco teórico tanto de la **variable independiente:** El uso de Interfaces Naturales de Usuario. En la **variable dependiente:** Desarrollo de la ubicación espacial de niños en preescolar.

Se aplica la modalidad de campo porque se busca obtener la información del desempeño en la utilización de las Interfaces Naturales de Usuario por parte de las niñas de la Unidad Educativa Fiscomisional la Inmaculada Concepción.

La investigación es experimental porque se manipula la **Variable Independiente:** El uso de Interfaces Naturales de Usuario para observar los efectos que ocurren con la **Variable Dependiente:** Desarrollo de la ubicación espacial de niños en preescolar.

1.2.2.2. Tipos o Niveles de Investigación

Investigación Exploratoria. - Recae en esta categoría al no haberse realizado investigaciones acerca de esta temática en la ciudad de Ibarra.

Investigación Descriptiva. - Por medio de la recolección, análisis y conclusiones se llega a identificar la relación entre la variable independiente y la variable dependiente.

Investigación Correlacional. - Por medio de esta se mide la relación entre la variable independiente y la variable dependiente.

1.3. Procedimiento de Investigación

1.3.1. Población y Muestra

La investigación se realiza con todas las niñas de 5 y 6 años de edad de los primeros años de educación básica conjuntamente con las docentes de los respectivos grados. Debido a que en esta etapa de la vida según Piaget las niñas empiezan a desarrollar los conceptos de lateralidad.

1.3.1.1. Población

Tabla 1. Población

Grados	Total de niñas	Docente
Primero de básica “A”	23	1
Primero de básica “B”	23	1
Primero de básica “C”	24	1
Total	70	3

Nota: Tomado del Sistema de Matriculación de la UEFLIC (2018)

1.3.1.2. Muestra

Al ser una investigación cualitativa el diseño de la misma evoluciona a lo largo del proyecto. Por lo tanto, en el muestreo la decisión sobre el mejor modo de obtener los datos se toma en el campo permitiendo reflejar la realidad de los diversos puntos de vista de los participantes. Las muestras son pequeñas no aleatorias. (Martín y Salamanca, 2007)

1.3.2. Operacionalización de Variables

Variable Independiente: El uso de Interfaces Naturales de Usuario.

Tabla 2. Operacionalización Variable Independiente

Conceptualización	Dimensiones	Indicadores	Ítems básicos	Técnicas e instrumentos	Fuente de información
Las Interfaces Naturales de Usuario (NUI) se constituyen como nuevos métodos para la Interacción Humano Computador (HCI) y el diseño de aplicaciones informáticas basadas en interfaces con las cuales las interacciones se realizan a partir de las acciones naturales de los seres humanos, tal y como éstos realizan sus actividades en el mundo físico de todos los días, sin la necesidad de periféricos para ingresar los datos, aprovechando de esta forma los conocimientos que sobre este entorno tenemos los seres humanos de manera innata. (Lozada, Rivera y Molina, 2014)	Interfaz Natural de Usuario (NUI).	Permite a los usuarios interactuar con el hardware y software realizando una semejanza al mundo real.	¿Cómo se diseñan las interfaces naturales de usuario?	Revisión Bibliográfica.	Libros y Artículos.
	Videojuego de Ubicación Espacial (UBIC).	Métricas de Usabilidad: Eficacia Eficiencia Satisfacción del Usuario	¿Qué características de Usabilidad tiene el Videojuego de Ubicación Espacial (UBIC)?	Revisión Bibliográfica. Escalas para medir Actitudes.	Libros y Artículos. Dirigido a las niñas de 5 y 6 años de edad de los primeros años de educación básica.

Variable Dependiente: Desarrollo de la ubicación espacial de niños en preescolar.

Tabla 3. Operacionalización Variable Dependiente

Conceptualización	Dimensiones	Indicadores	Ítems básicos	Técnicas e instrumentos	Fuente de información
Es la capacidad que los niños utilizan para relacionar objetos, tomando como referencia los mismos objetos	Ubicación Espacial.	Definiciones de ubicación espacial.	¿Qué entendemos por	Revisión Bibliográfica. Entrevista.	Libros y Artículos. Docentes parvularios.

<p>y compararlos con otros dando lugar a comparaciones de los atributos que posee como: formas, medida, distancias, reconocimiento de objetos y nombres del mundo que lo rodea. Permitiendo su ubicación y espacio.</p> <p>(Piaget, 1948)</p>			ubicación espacial?		
	Métodos lúdicos de enseñanza .	Juegos. Canciones.	¿Cómo es el método de enseñanza en la ubicación espacial?	Revisión Bibliográfica. Observación. Entrevista.	Libros y Artículos. Docentes parvularios. Niñas de 5 y 6 años de edad de los primeros años de educación básica.
	Herramientas didácticas de aprendizaje.	Libro del Gobierno. Hojas de Trabajo. Objetos que permitan la fácil identificación de la ubicación. Dibujos representativos como: Animales, personas y formas	¿Qué tipos de instrumentos facilitan el aprendizaje de la ubicación espacial?	Revisión Bibliográfica. Observación. Entrevista.	Libros y Artículos. Docentes parvularios. Niñas de 5 y 6 años de edad de los primeros años de educación básica.

1.3.3. Métodos

Deductivo: “La deducción es un proceso que parte de un principio general ya conocido para inferir de él, consecuencias particulares”. (Gutiérrez, 2006).

Este método permite partir de modelos generales para el diseño de las estrategias y recursos que se implementan en la construcción del conjunto de directrices de Interfaces Naturales de Usuario.

Inductivo: “Este método utiliza el razonamiento para obtener conclusiones que parten de hechos particulares aceptados como válidos, para llegar a conclusiones cuya aplicación sea de carácter general”. (Bernal, 2010).

Permite analizar los datos obtenidos en el diagnóstico para llegar a determinar las estrategias, recursos, materiales y medios que intervienen en el proceso de desarrollo de la ubicación espacial de niños en preescolar.

Analítico – Sintético: “El análisis consiste en descomponer en partes algo complejo, en desintegrar un hecho o una idea en sus partes, para mostrarlas, describirlas, numerarlas y para explicar las causas de los hechos o fenómenos que constituyen el todo”. (Leiva, 2007).

Este método permite realizar un análisis de las diferentes técnicas para crear un conjunto de directrices a seguir en el diseño y la realización de bosquejos para las interfaces gráficas.

Tabla 4. Métodos Generales

MÉTODOS GENERALES	Deductivo
	Inductivo
	Analítico
	Sintético

1.3.4. Metodología de Desarrollo de Software

El proceso de desarrollo del videojuego de ubicación espacial denominado “UBIC”, por ser un nombre fácil de recordar y está compuesto por las cuatro letras iniciales de la palabra ubicación, seguirá las fases de la metodología ágil de desarrollo de software XP, (Fowler y Beck, 2000) como se muestra en la figura 4.

I Fase Planificación del Proyecto

II Fase de Diseño

III Fase Codificación

IV Fase Pruebas

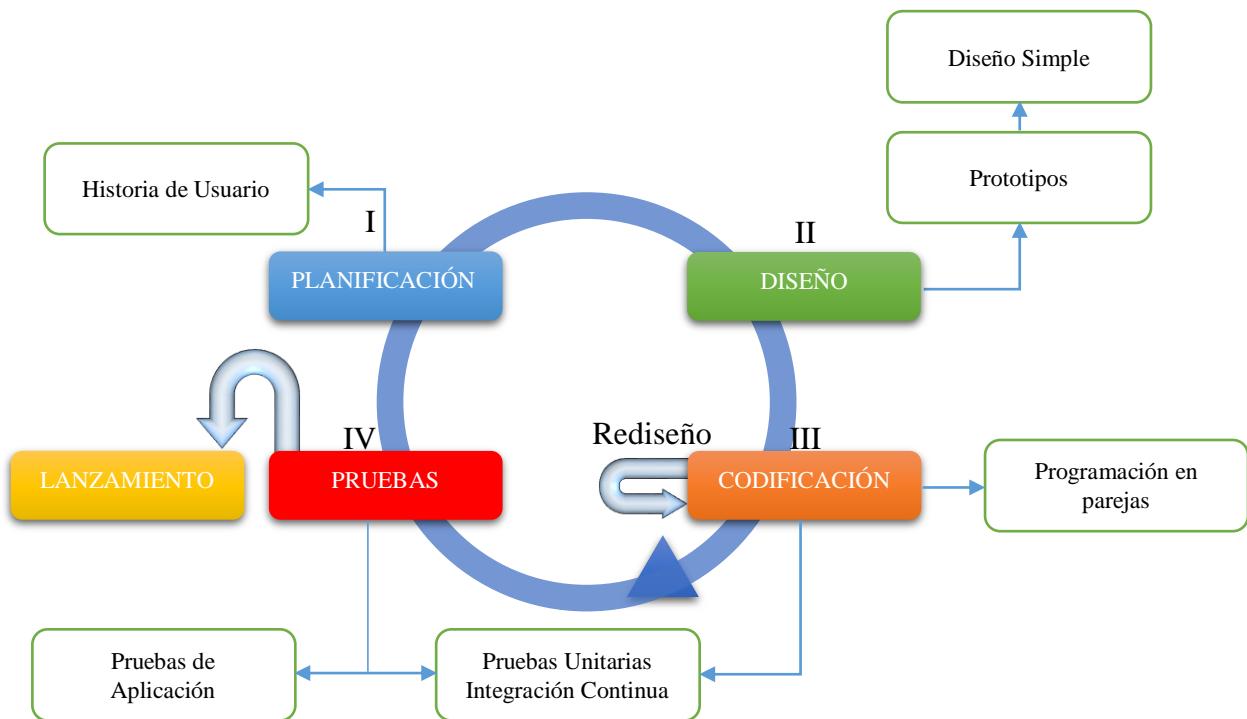


Figura 4. Fases de la Metodología ágil de desarrollo de software XP

1.3.5. Estrategias Técnicas

Para el desarrollo se utilizó las siguientes técnicas:

Entrevista: La cual se aplica a las docentes parvularios de los primeros años de educación básica de la UEFLIC y a expertos en la materia.

Observación Directa: Se la realiza mediante visitas a la UEFLIC para observar la manera en la cual se enseña las nociones de ubicación espacial.

Escalas para medir Actitudes: Se mide el grado de eficacia, eficiencia y satisfacción del uso del videojuego UBIC por parte de las niñas de los primeros años de educación básica.

Datos Secundarios: Se revisa investigaciones realizadas acerca de utilización de interfaces naturales de usuario con Kinect.

1.3.6. Instrumentos de Investigación

Para recolectar la información de la muestra seleccionada y poder resolver el problema de la investigación se seleccionó los siguientes instrumentos:

- Para el caso de la entrevista las preguntas del cuestionario.
- Para la observación se utilizó como instrumento la ficha de observación.
- Celular, como equipo de comunicación.
- Cámara fotográfica, que facilita recabar evidencia de la investigación.
- Kinect, que permite realizar pruebas de interfaces gráficas.

1.4. Análisis e Interpretación de Resultados

1.4.1. Análisis de la Observación

En la observación realizada a las niñas de 5 y 6 años de edad de los primeros años de educación básica conjuntamente con las docentes de los respectivos grados de la UEFLIC, se evidencia la metodología tradicional de enseñanza-aprendizaje de ubicación espacial.

Observación Primero de Básica paralelo “A”



Figura 5. Observación Primero de Básica paralelo "A"

Tabla 5. Ficha de Bajada de Observación Primero de Básica paralelo "A"

Ficha de Bajada de Observación	
Observadores	Fernando Recalde Mario Merlo
Actividad Observada:	Enseñanza de Lateralidad Izquierda y Derecha
Notas de Campo ¿Qué me llamó la atención?	<p>En la enseñanza aprendizaje de ubicación espacial la docente la realiza por medio de una actividad lúdica.</p> <p>En la enseñanza de lateralidad izquierda y derecha la docente usa partes del cuerpo para enseñar.</p> <p>La docente asocia la actividad de escritura para enseñanza de izquierda y derecha.</p> <p>La docente realiza preguntas y respuestas con respecto a la actividad realizada para observar si esta cimentado los conocimientos</p> <p>Las niñas pueden realizar la actividad sentadas o levantadas de su pupitre.</p> <p>Las niñas asocian sus partes del cuerpo como: manos, ojos, cabeza, pies, para aprender su lateralidad.</p> <p>Las niñas cuando la docente dice mano izquierda levantan la mano izquierda. Cuando la docente dice mano derecha levantan la mano derecha.</p>

Observación Primero de Básica paralelo "B"



Figura 6. Observación Primero de Básica paralelo "B"

Tabla 6. Ficha de Bajada de Observación Primero de Básica paralelo "B"

Ficha de Bajada de Observación	
Observadores	Fernando Recalde Mario Merlo
Actividad Observada:	Enseñanza de Arriba y Abajo
Notas de Campo ¿Qué me llamó la atención?	<p>En la enseñanza aprendizaje de ubicación espacial la docente la realiza por medio de una actividad lúdica.</p> <p>En la enseñanza de arriba y abajo la docente usa partes del cuerpo para enseñar.</p> <p>La docente asocia la actividad de canto “arriba abajo uno, dos y tres” para enseñanza de arriba y abajo.</p> <p>La docente no participa en la actividad para observar si esta cimentado los conocimientos</p> <p>Las niñas deben realizar la actividad levantadas de su pupitre.</p> <p>Las niñas asocian sus partes del cuerpo como: manos, ojos, cabeza, pies, para aprender su ubicación.</p> <p>Las niñas cuando la docente dice arriba levantan los brazos. Cuando la docente dice abajo, bajan los brazos</p>

Observación Primero de Básica paralelo “C”



Figura 7. Observación Primero de Básica paralelo "C"

Tabla 7. Ficha de Bajada de Observación Primero de Básica paralelo "C"

Ficha de Bajada de Observación	
Observadores	Fernando Recalde Mario Merlo
Actividad Observada:	Enseñanza de Arriba y Abajo
Notas de Campo ¿Qué me llamó la atención?	<p>En la enseñanza aprendizaje de ubicación espacial la docente la realiza por medio de una actividad lúdica.</p> <p>En la enseñanza de adentro y afuera la docente relaciona objetos como pelotas de diferentes formas y colores para enseñar.</p> <p>La docente asocia la actividad por medio de indicaciones “El lugar donde se encuentran ubicadas o guardadas las pelotas” para la enseñanza de la ubicación.</p> <p>La docente asocia la actividad por medio de indicaciones “Las pelotas son de diferentes colores, tamaños y formas” para enseñanza de ubicación.</p> <p>La docente da una instrucción a una niña para observar si esta cimentado los conocimientos.</p> <p>Las niñas deben realizar la actividad levantadas de su pupitre.</p> <p>Las niñas diferencian colores, tamaños y formas, para aprender su ubicación.</p> <p>La niña cuando la docente dice adentro la niña selecciona una pelota y ubica adentro del cesto de pelotas, cuando la docente dice afuera la niña selecciona una pelota y ubica afuera del cesto de pelotas.</p>

1.4.2. Interpretación de la Observación

Para la enseñanza de la ubicación espacial las docentes realizan actividades lúdicas permitiendo la fácil explicación de los diferentes temas como: lateralidad izquierda y derecha, arriba y abajo, adentro y afuera. Las docentes relacionan partes del cuerpo y asocian actividades de canto conjuntamente con objetos.

1.4.3. Análisis de la Entrevista

La entrevista (ver anexo A) se aplicó a las docentes parvularios de los primeros años de educación básica.

Tabla 8. Ficha de Bajada de Entrevistas

Nombre del entrevistador		Ing. Fernando Recalde Ing. Mario Merlo			
Duración de la entrevista		15 minutos			
Docente Parvulario 1		Docente Parvulario 2		Docente Parvulario 3	
Lic. Cristina Palomeque Fecha de entrevista: 15-01-2018 Perfil de Usuario: Mujer de 42 años Estado civil casada Con hijos Docente parvulario subnivel 1 Docente Fiscal		Lic. Betty Reascos Fecha de entrevista: 16-01-2018 Perfil de Usuario: Mujer de 27 años Estado civil casada Docente parvulario subnivel 1 Con hijos Docente Fiscal		Lic. Ana Tarambis Fecha de entrevista: 17-01-2018 Perfil de Usuario: Mujer de 38 años Estado civil divorciada Docente parvulario subnivel 1 Con hijos Docente Particular	
Hechos	Aspectos Negativos	Aspectos Positivos	Atajos/ Soluciones Informales a problemas Actuales	Ideas	Casos
Docente parvulario de grado. Las nociones básicas de ubicación espacial deben estar desarrolladas en el primer quimestre del año escolar.	No utiliza medios digitales para la enseñanza aprendizaje. No existe una innovación tecnológica. Las aulas son de 23 y 24 niñas. No todas las niñas tienen cursado el inicial I y II. Los libros de preescolar del gobierno garantizan el aprendizaje en cuanto a la ubicación espacial.	Utilizan dinámicas para enseñanza de la ubicación espacial. Las niñas que han cursado el Inicial I y II ya tienen nociones básicas para su ubicación espacial.	Utilizan Pictogramas. Utilizan grabadora. Utilizan una manilla de colores según el día para diferenciar su ubicación espacial. Las actividades a realizar en clase ya están establecidas en el libro del gobierno para preescolar.	Utilizar pupitres unipersonales para el aprendizaje adecuado de las niñas. Las aulas deben tener un límite de 15 niñas. Utilizar medios tecnológicos para mejorar el aprendizaje sean: fáciles de utilizar, intuitivos y que llamen la atención de las niñas. Se sugiere el color rosado para el diseño de un entorno gráfico tecnológico.	Toda Actividad siempre tiene que ir relacionado con la ubicación espacial. Como, por ejemplo: Pinte de color amarillo que se encuentra al lado derecho del lago. Para trabajar las actividades en la ubicación espacial deben ser al inicio de la jornada diaria escolar.

				<p>Se sugiere utilizar de forma prudente el color negro para el diseño de un entorno grafico tecnológico.</p>	
				<p>Se sugiere utilizar como mascota al perro en el diseño de un entorno grafico tecnológico.</p>	

1.4.4. Interpretación de la Entrevista

Las docentes parvularios de grado manifiestan que las nociones básicas de ubicación espacial deberían estar conceptualizadas en los prerrequisitos del primer grado que son inicial I e inicial II, por motivo que las nociones de ubicación espacial ya deben estar desarrolladas en el primer quimestre del año escolar.

Por tal razón las docentes para la enseñanza de la ubicación espacial utilizan dinámicas lúdicas de acuerdo al tema.

Como refuerzo utilizan el libro del gobierno para preescolar que contiene actividades sobre ubicación espacial que pueden ser aplicadas en la hora de clase.

También se debe considerar un factor importante que en las primeras horas de la mañana es donde las niñas tienen un aprendizaje más significativo que en el tránscurso de las horas que se termina la mañana.

La importancia de tecnología ayudaría como reforzamiento de enseñanza aprendizaje ya que las niñas aprenden de forma rápida e intuitiva cuando observan colores, objetos y sonidos.

CAPÍTULO IV

PROPUESTA

4.1. Algoritmo Propuesto

Para realizar el presente proyecto se propone un algoritmo que establece las directrices a seguir en el desarrollo del videojuego de aprendizaje de ubicación espacial “UBIC”, aplicado en la enseñanza-aprendizaje de niñas de preescolar de la UEFLIC.

Este se basa en las metodologías: Design Thinking, Lean UX, Agile y Growth Hacking que en conjunto forman la Metodología Lean Startup, que es usada en el proceso de emprender un nuevo negocio o Startup. La metodología Lean Startup propuesta por la empresa Nordstrom Innovation Lab en la conferencia International Design Management Research DMI Conference del 2012, fue desarrollada por Stephane Debucque, en la que propone un enfoque integrador de las metodologías antes mencionadas para desarrollar la capacidad de innovación de las empresas como muestra la figura 8. (Emprende a Conciencia, 2019)

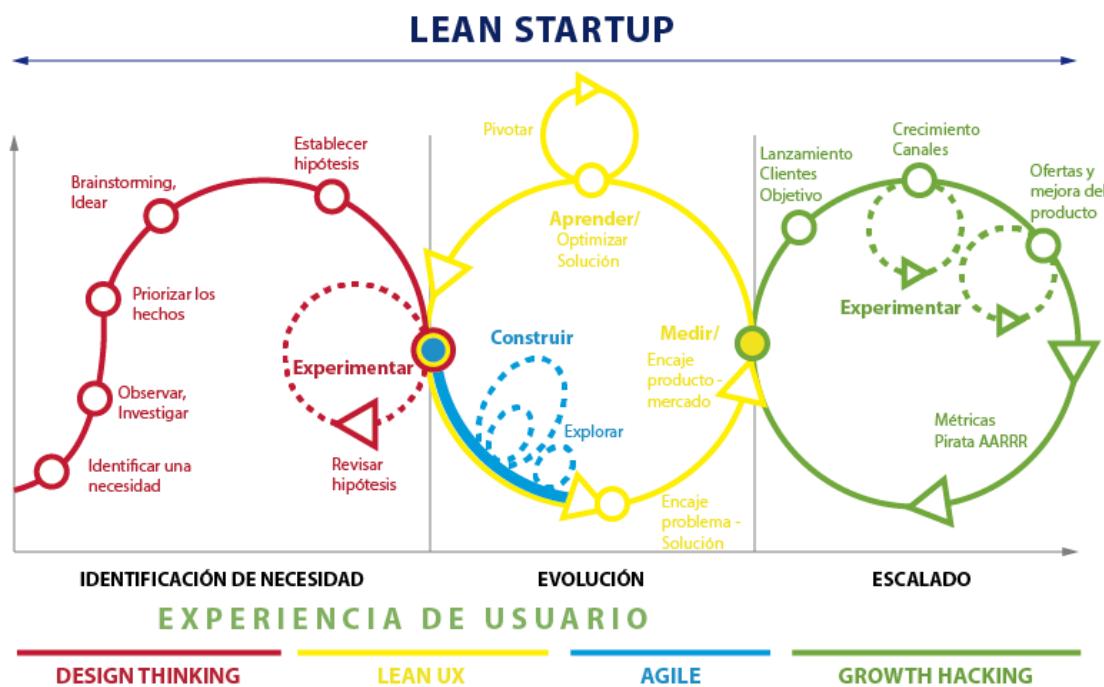


Figura 8. Metodología Lean Startup

Nota: Tomado de Emprende a Conciencia (2019).

Cada una de estas metodologías ayudan en los pasos necesarios para minimizar los riesgos y avanzar en el proceso de maduración de una startup lo más rápido posible.

Design Thinking: Explora el problema. Se usa la empatía, ideación, prototipado y experimentación para resolver problemas reales.

Lean UX: Construye, mide y valida que lo que se está construyendo sea lo que el cliente realmente necesita y va a usar. Prototipado rápido, lanzar un Producto Mínimo Viable (MVP) que resuelva las necesidades reales del cliente, prototipos ágiles sobre wireframes pesados y hojas de especificaciones.

Agile: Es una mentalidad para organizarse, utiliza la colaboración y varias entregas en tiempos cortos que podemos testear con clientes reales.

Growth Hacking: Una vez alcanzado el Producto Mínimo Final (PMF), se empieza a optimizar el embudo de marketing, encontrando a los canales objetivo y la propuesta de valor correcta. Esto nos da resultados que se pueden medir y volver a mejorar con ello.

Lean Startup: Busca una validación temprana de una MVP para optimizar tiempos y esfuerzos en productos / servicios en los que realmente los clientes estén interesados. Es tener en la mente: “Falla temprano, rápido y frecuentemente”. (Gothelf, 2013)

Para crear un algoritmo que satisfaga los objetivos propuestos del presente proyecto se toma en consideración la técnica de Design Thinking en el ciclo de investigación debido a que esta técnica nos ayuda en la forma de resolver problemas en el proceso de innovación. Se aplica Lean UX y Agile en el ciclo de desarrollo del software debido a que utilizamos XP como metodología ágil de desarrollo de software. No se toma en cuenta la parte Growth Hacking debido a que no es un producto que requiere una comercialización.

Tomando en consideración los beneficios de las metodologías antes mencionadas se propone un algoritmo para el desarrollo de la aplicación de software que está ilustrado en la figura 9.



Figura 9. Algoritmo Propuesto

4.1.1. Pasos del Algoritmo Propuesto

El algoritmo propuesto se basa en dos partes que nos ayudan a determinar de una mejor manera la experiencia del usuario:

- **Parte1: Identificación de necesidad.** – Está relacionado al ciclo de investigación del proyecto.
- **Parte2: Evolución.** – Se encuentra en concordancia con el ciclo de desarrollo del software y la metodología ágil XP.

Cada parte está compuesta de varios pasos que se enumeran a continuación:

Parte1: Identificación de necesidad

1. **Identificar una necesidad.** – En este paso se establece el problema que se quiere resolver, los objetivos generales y específicos.

2. Observar, investigar. – Con el problema bien definido, se realiza el proceso de investigar por medio de las técnicas de recolección de información de fuentes bibliográficas, entrevistas y la observación a los participantes del proyecto.

3. Brainstorming, idear. - Cuando ya se tienen los datos más relevantes del problema analizados, se puede empezar a realizar una lluvia de ideas de las diferentes propuestas para solucionar el problema planteado.

4. Validar. – Aquí se realiza pruebas en papel con prototipos rápidos (mockups) que se evalúan con pruebas de usabilidad o pruebas de concepto en papel.

5. Visualizar. – Se crea un prototipo más funcional utilizando herramientas de diseño de interfaces, en las cuales se puede especificar colores, formas, iconografía, tipos de fuentes, personajes; para simular de una mejor forma el producto final.

6. Pruebas. – Se realizan más pruebas de usabilidad con las personas involucradas en el proyecto para determinar si la solución es la más factible.

De esta forma se daría por terminado el primer ciclo de investigación del algoritmo. Para entrar en el segundo ciclo de desarrollo del software se debe especificar que el videojuego se lo realizó empleando la metodología ágil de desarrollo de software XP permitiendo potenciar las relaciones interpersonales entre el cliente y los desarrolladores promoviendo trabajo en equipo, propiciando un buen ambiente de trabajo, basada en una retroalimentación continua entre los involucrados por la simplicidad de las soluciones implementadas. (Orjuela y Rojas, 2008)

Parte2: Evolución

1. Requerimientos. – El prototipo que se desarrolló se convierte en los requerimientos de la aplicación.

2. Codificación / Desarrollo. - Se escribe el código fuente de la aplicación en un lenguaje entendible para el computador.

3. Pruebas. – Se ejecutan las pruebas alpha y beta para corregir cualquier error o defectos encontrados en el proceso de desarrollo de la aplicación.

4. Despliegue. - Se realiza la implementación de la aplicación.

5. Medición. – Se aplica escala de medición de la usabilidad del software, determinando la eficiencia, eficacia y satisfacción del usuario.

Como es un ciclo iterativo se puede regresar al ciclo de investigación para obtener nuevamente información relevante para el desarrollo de la aplicación, tomando en cuenta que cada uno de los sprint puede ser un prototipo de entrada en el siguiente sprint como requerimientos nuevos, desarrollo, pruebas, despliegue y medición; generando un ciclo iterativo dentro del algoritmo propuesto.

4.2. Fase de Planificación del Proyecto

4.2.1. Historias de Usuario

Para la estimación de las historias de usuario se usa la técnica de planning póker, porque permite realizar un consenso con todas las personas involucradas en el proyecto.

En la aplicación de la técnica de planning póker, se usó las cartas de la baraja inglesa, especificadas en la tabla 9.

Tabla 9. Equivalencias de Cartas de la Baraja Inglesa

VALOR ASIGNADO	CARTA	VALOR ASIGNADO	CARTA
1	A	16	J
2	2	32	Q
4	4	64	K
8	8		

Nota: Tomado de Kniberg (2007)

A continuación, se describen las historias de usuario (HU) que contendrá el videojuego de ubicación espacial (UBIC).

Tabla 10. Historia de Usuario Número1. Creación del Sistema de Seguimiento de la Mano

Historia de Usuario			
Número:	1	Nombre:	Creación del Sistema de Seguimiento de la Mano
Usuario	Jugador		
Modificación de Historia Número	1	Interacción Asignada:	1
Prioridad en Negocio: (Baja / Media / Alta)	Alta	Puntos Estimados:	16
Riesgo del Desarrollo: (Baja / Media / Alta)	Alta	Puntos Reales:	
Descripción:	Se crea la estructura de la Interfaces Naturales de Usuario para la detección de la mano que actúa como dispositivo externo similar al mouse.		
Observaciones:	Mano Abierta: Simula el movimiento del puntero del mouse. Mano Cerrada: Simula el evento de clic del mouse.		

Nota: Adaptado de Rivadeneira, Vilanova Miranda y Cruz. (2013)

Tabla 11. Historia de Usuario Número 2. Ingreso a la Pantalla Principal de la Aplicación

Historia de Usuario			
Número:	2	Nombre:	Ingreso a la Pantalla Principal de la Aplicación
Usuario	Jugador		
Modificación de Historia Número	1	Interacción Asignada:	2
Prioridad en Negocio: (Baja / Media / Alta)	Baja	Puntos Estimados:	2
Riesgo del Desarrollo: (Baja / Media / Alta)	Baja	Puntos Reales:	
Descripción:	Al ingresar a la pantalla principal se muestra un menú con dos opciones: 1. Entrar: Al hacer clic se despliega la pantalla del menú de los videojuegos. 2. Salir: Al hacer clic sale de la aplicación.		
Observaciones:	Si el jugador presiona la tecla ESC, sale de la aplicación.		

Tabla 12. Historia de Usuario Número 3. Ingreso al Menú de Videojuegos de la Aplicación

Historia de Usuario			
Número:	3	Nombre:	Ingreso al Menú de Videojuegos de la Aplicación
Usuario	Jugador		
Modificación de Historia Número	1	Interacción Asignada:	3
Prioridad en Negocio: (Baja / Media / Alta)	Baja	Puntos Estimados:	4

Riesgo del Desarrollo: (Baja / Media / Alta)	Baja	Puntos Reales:	
Descripción:			
Al ingresar a la pantalla del menú de juegos se muestra un menú con cinco opciones:			
<ol style="list-style-type: none"> 1. Entrenamiento: Al hacer clic ingresa al videojuego de entrenamiento. 2. Izquierda Derecha: Al hacer clic ingresa al videojuego de Izquierda y Derecha. 3. Arriba Abajo: Al hacer clic ingresa al videojuego de Arriba y Abajo. 4. Adentro Afuera: Al hacer clic ingresa al videojuego de Adentro y Afuera, Encima y Debajo. 5. Laberinto: Al hacer clic ingresa al videojuego de Laberinto. 			
Observaciones:			
Si el jugador presiona la tecla ESC, sale de la aplicación.			

Tabla 13. Historia de Usuario Número 4. Ingreso al Videojuego de Entrenamiento

Historia de Usuario			
Número:	4	Nombre:	Ingreso al Videojuego de Entrenamiento
Usuario	Jugador		
Modificación de Historia Número	1	Interacción Asignada:	5
Prioridad en Negocio: (Baja / Media / Alta)	Media	Puntos Estimados:	8
Riesgo del Desarrollo: (Baja / Media / Alta)	Media	Puntos Reales:	
Descripción:			
Al ingresar al videojuego de entrenamiento tiene 4 actividades a completar:			
<ol style="list-style-type: none"> 1. La actividad no necesariamente puede ser terminada porque tiene un tiempo estimado de 15 segundos para pasar a la siguiente actividad. 2. Cada actividad completada correctamente tiene un puntaje de las veces que haya hecho clic correctamente. 3. Al hacer clic en cualquiera de los dos círculos aparece otra pantalla con otra actividad. 			
Observaciones:			
Cada actividad es descrita en otra historia de usuario			

Tabla 14. Historia de Usuario Número 5. Actividad de Entrenamiento Izquierda

Historia de Usuario			
Número:	5	Nombre:	Actividad de Entrenamiento Izquierda
Usuario	Jugador		
Modificación de Historia Número	2	Interacción Asignada:	5
Prioridad en Negocio: (Baja / Media / Alta)	Media	Puntos Estimados:	4
Riesgo del Desarrollo: (Baja / Media / Alta)	Media	Puntos Reales:	
Descripción:			

Aparece en la pantalla dos círculos: Círculo con relleno: Se encuentra en el lado izquierdo de la pantalla tomando como punto de referencia el centro de la pantalla, permitiendo asignar puntaje como respuesta correcta. Círculo sin relleno: Se encuentra en el lado derecho de la pantalla tomando como punto de referencia el centro de la pantalla, el cual no asigna ningún puntaje.
Observaciones:
Si el jugador presiona la tecla ESC, sale de la aplicación. Si el jugador presiona la tecla P, pausa el videojuego durante 5 segundos. Si el jugador presiona la tecla Back Space, regresa al menú de juegos.

Tabla 15. Historia de Usuario Número 6. Actividad de Entrenamiento Derecha

Historia de Usuario			
Número:	6	Nombre:	Actividad de Entrenamiento Derecha
Usuario	Jugador		
Modificación de Historia Número	2	Interacción Asignada:	5
Prioridad en Negocio: (Baja / Media / Alta)	Media	Puntos Estimados:	4
Riesgo del Desarrollo: (Baja / Media / Alta)	Media	Puntos Reales:	
Descripción:			
Aparece en la pantalla dos círculos: Círculo sin relleno: Se encuentra en el lado izquierdo de la pantalla tomando como punto de referencia el centro de la pantalla, el cual no asigna ningún puntaje. Círculo con relleno: Se encuentra en el lado derecho de la pantalla tomando como punto de referencia el centro de la pantalla, permitiendo asignar puntaje como respuesta correcta.			
Observaciones:			
Si el jugador presiona la tecla ESC, sale de la aplicación. Si el jugador presiona la tecla P, pausa el videojuego durante 5 segundos. Si el jugador presiona la tecla Back Space, regresa al menú de juegos.			

Tabla 16. Historia de Usuario Número 7. Actividad de Entrenamiento Arriba

Historia de Usuario			
Número:	7	Nombre:	Actividad de Entrenamiento Arriba
Usuario	Jugador		
Modificación de Historia Número	2	Interacción Asignada:	5
Prioridad en Negocio: (Baja / Media / Alta)	Media	Puntos Estimados:	4
Riesgo del Desarrollo: (Baja / Media / Alta)	Media	Puntos Reales:	
Descripción:			

Aparece en la pantalla dos círculos: Círculo con relleno: Se encuentra en la parte superior de la pantalla tomando como punto de referencia el centro de la pantalla, permitiendo asignar puntaje como respuesta correcta. Círculo sin relleno: Se encuentra en la parte inferior de la pantalla tomando como punto de referencia el centro de la pantalla, el cual no asigna ningún puntaje.
Observaciones:
Si el jugador presiona la tecla ESC, sale de la aplicación. Si el jugador presiona la tecla P, pausa el videojuego durante 5 segundos. Si el jugador presiona la tecla Back Space, regresa al menú de juegos.

Tabla 17. Historia de Usuario Número 8. Actividad de Entrenamiento Abajo

Historia de Usuario			
Número:	8	Nombre:	Actividad de Entrenamiento Abajo
Usuario	Jugador		
Modificación de Historia Número	2	Interacción Asignada:	5
Prioridad en Negocio: (Baja / Media / Alta)	Media	Puntos Estimados:	4
Riesgo del Desarrollo: (Baja / Media / Alta)	Media	Puntos Reales:	
Descripción:			
Aparece en la pantalla dos círculos: Círculo sin relleno: Se encuentra en la parte superior de la pantalla tomando como punto de referencia el centro de la pantalla, el cual no asigna ningún puntaje. Círculo con relleno: Se encuentra en la parte inferior de la pantalla tomando como punto de referencia el centro de la pantalla, permitiendo asignar puntaje como respuesta correcta.			
Observaciones:			
Si el jugador presiona la tecla ESC, sale de la aplicación. Si el jugador presiona la tecla P, pausa el videojuego durante 5 segundos. Si el jugador presiona la tecla Back Space, regresa al menú de juegos.			

Tabla 18. Historia de Usuario Número 9. Pantalla de Fin de Juego

Historia de Usuario			
Número:	9	Nombre:	Pantalla de Fin de Juego
Usuario	Jugador		
Modificación de Historia Número	2	Interacción Asignada:	3
Prioridad en Negocio: (Baja / Media / Alta)	Baja	Puntos Estimados:	8
Riesgo del Desarrollo: (Baja / Media / Alta)	Media	Puntos Reales:	
Descripción:			

<p>La pantalla aparece cuando el juego finalice mostrando la siguiente información: La frase fin de juego con el nombre del videojuego, el puntaje obtenido, el tiempo de realización de la actividad y el menú de opciones:</p> <ol style="list-style-type: none"> 1. Reiniciar Juego: Al hacer clic se reinicia el videojuego. 2. Menú de Juegos: Al hacer clic se despliega la pantalla del menú de los videojuegos. 3. Salir: Al hacer clic sale de la aplicación
Observaciones:
Si el jugador presiona la tecla ESC, sale de la aplicación.

Tabla 19. Historia de Usuario Número 10. Ingreso al Videojuego de Izquierda - Derecha

Historia de Usuario			
Número:	10	Nombre:	Ingreso al Videojuego de Izquierda - Derecha
Usuario	Jugador		
Modificación de Historia Número	3	Interacción Asignada:	2
Prioridad en Negocio: (Baja / Media / Alta)	Alta	Puntos Estimados:	16
Riesgo del Desarrollo: (Baja / Media / Alta)	Alta	Puntos Reales:	
Descripción:			
Al ingresar al videojuego de Izquierda - Derecha debe completar la siguiente actividad:			
<ol style="list-style-type: none"> 1. Se muestra un personaje enemigo llamada “Gallina”, que se mueve de izquierda a derecha en la parte superior de la pantalla, cada cierto tiempo arroja huevos de forma vertical, los cuales el jugador tiene que recoger en una cesta, aumentando el puntaje por cada huevo obtenido. 2. El jugador se mueve de izquierda a derecha haciendo clic con el mouse o con NUI conjuntamente con Kinect. 3. Cada huevo recogido tiene un valor de 1 punto. 4. El juego finaliza cuando un huevo pase el límite de la altura de la pantalla o el jugador haya recogido un total de diez huevos. 5. Cuando finaliza el videojuego se muestra la pantalla de fin de juego. 			
Observaciones:			
Si el jugador presiona la tecla ESC, sale de la aplicación. Si el jugador presiona la tecla P, pausa el videojuego durante 5 segundos. Si el jugador presiona la tecla Back Space, regresa al menú de juegos. Si el jugador presiona la Barra Espaciadora, desactiva el sonido del videojuego. Si el jugador presiona la tecla v, activa el sonido del videojuego.			

Tabla 20. Historia de Usuario Número 11. Ingreso al Videojuego de Arriba - Abajo

Historia de Usuario			
Número:	11	Nombre:	Ingreso al Videojuego de Arriba - Abajo
Usuario	Jugador		
Modificación de Historia Número	2	Interacción Asignada:	3

Prioridad en Negocio: (Baja / Media / Alta)	Alta	Puntos Estimados:	16
Riesgo del Desarrollo: (Baja / Media / Alta)	Alta	Puntos Reales:	
Descripción:			
<p>Al ingresar al videojuego de Arriba – Abajo debe completar la siguiente actividad:</p> <ol style="list-style-type: none"> 1. Se muestra una lista de obstáculos, que se mueven de derecha a izquierda de la pantalla, cada cierto tiempo aparecen obstáculos en la parte superior e inferior, los cuales el jugador tiene que esquivar y recoger las estrellas para aumentar el puntaje. 2. El jugador se mueve de arriba hacia abajo haciendo clic con el mouse o con NUI conjuntamente con Kinect. 3. Cada estrella recogida tiene un valor de 1 punto. 4. El juego finaliza cuando el jugador choque con un obstáculo o el jugador haya recogido un total de veinte estrellas. 5. Cuando finaliza el videojuego se muestra la pantalla de fin de juego. 			
Observaciones:			
<p>Si el jugador presiona la tecla ESC, sale de la aplicación.</p> <p>Si el jugador presiona la tecla P, pausa el videojuego durante 5 segundos.</p> <p>Si el jugador presiona la tecla Back Space, regresa al menú de juegos.</p> <p>Si el jugador presiona la Barra Espaciadora, desactiva el sonido del videojuego.</p> <p>Si el jugador presiona la tecla v, activa el sonido del videojuego.</p>			

Tabla 21. Historia de Usuario Número 12. Ingreso al Videojuego Adentro - Afuera

Historia de Usuario			
Número:	12	Nombre:	Ingreso al Videojuego de Adentro - Afuera
Usuario	Jugador		
Modificación de Historia Número	3	Interacción Asignada:	6
Prioridad en Negocio: (Baja / Media / Alta)	Alta	Puntos Estimados:	32
Riesgo del Desarrollo: (Baja / Media / Alta)	Alta	Puntos Reales:	
Descripción:			
<p>Al ingresar al videojuego de Adentro - Afuera:</p> <ol style="list-style-type: none"> 1. Aparece una imagen en la parte inferior de la pantalla que le indica al jugador la actividad que debe realizar en la parte superior. 2. El jugador en la parte superior de la pantalla debe realizar la acción que le indica la figura de la parte inferior. 3. La figura aparece por un tiempo de 30 segundos. 4. El puntaje es asignado si completa la actividad 5. Cada actividad completada tiene un valor de 1 punto. 6. El juego continúa cuando el jugador no complete una actividad porque se terminó el tiempo. 7. El juego finaliza cuando el jugador complete las cuatro actividades. 8. Cuando finaliza el videojuego se muestra la pantalla de fin de juego. 			
Observaciones:			

Si el jugador presiona la tecla ESC, sale de la aplicación.
 Si el jugador presiona la tecla P, pausa el videojuego durante 5 segundos.
 Si el jugador presiona la tecla Back Space, regresa al menú de juegos.
 Si el jugador presiona la Barra Espaciadora, desactiva el sonido del videojuego.
 Si el jugador presiona la tecla v, activa el sonido del videojuego.

Tabla 22. Historia de Usuario Número 13. Ingreso al Videojuego de Laberinto

Historia de Usuario						
Número:	13	Nombre:	Ingreso al Videojuego de Laberinto			
Usuario	Jugador					
Modificación de Historia Número	2	Interacción Asignada:	4			
Prioridad en Negocio: (Baja / Media / Alta)	Alta	Puntos Estimados:	32			
Riesgo del Desarrollo: (Baja / Media / Alta)	Alta	Puntos Reales:				
Descripción:						
Al ingresar al videojuego de Laberinto debe completar la siguiente actividad: <ol style="list-style-type: none"> El jugador llamado “Ardilla” tiene como objetivo recoger las nueces que se encuentran dentro del laberinto, aumentando el puntaje por cada nuez recogida. El jugador se mueve de izquierda a derecha, arriba y abajo haciendo clic con el mouse o con NUI conjuntamente con Kinect. Cada nuez recogida tiene un valor de 1 punto. El juego finaliza cuando la ardilla llegue a su objetivo final que es un árbol que se encuentra al otro lado del laberinto. Cuando finaliza el videojuego se muestra la pantalla de fin de juego. 						
Observaciones:						
Si el jugador presiona la tecla ESC, sale de la aplicación. Si el jugador presiona la tecla P, pausa el videojuego durante 5 segundos. Si el jugador presiona la tecla Back Space, regresa al menú de juegos. Si el jugador presiona la Barra Espaciadora, desactiva el sonido del videojuego. Si el jugador presiona la tecla v, activa el sonido del videojuego.						

Tabla 23. Historia de Usuario Número 14. Pantalla de Instrucciones del Videojuego

Historia de Usuario			
Número:	14	Nombre:	Pantalla de Instrucciones del Videojuego
Usuario	Jugador		
Modificación de Historia Número	1	Interacción Asignada:	5
Prioridad en Negocio: (Baja / Media / Alta)	Media	Puntos Estimados:	2
Riesgo del Desarrollo: (Baja / Media / Alta)	Baja	Puntos Reales:	
Descripción:			

La pantalla aparece al inicio de cada videojuego mostrando las instrucciones del juego.
 Contiene las reglas del videojuego.
 Tiene un menú con dos opciones:
Anterior: Para regresar a la pantalla del menú de los videojuegos.
Siguiente: Para iniciar el videojuego.

Observaciones:

Si el jugador presiona la tecla ESC, sale de la aplicación.
 Si el jugador presiona la tecla Back Space, regresa al menú de juegos.

4.2.2. Plan de Versiones

Una vez determinadas las historias de usuario se realiza un cronograma del plan de versiones el cual contempla las historias de usuario en un orden de las iteraciones establecidas. En un total de 7 iteraciones que son desarrolladas en 105 días laborables (1 semana = 5 días de 8 horas diarias dando un total de 40 horas semana). (Cohn, 2005)

Tabla 24. Cronograma del Plan de Versiones

Orden	Número de Historia	Nombre De La Historia	Estimación	Iteración	Tiempo/Semanas	Tiempo/Días
1	1	Creación del Sistema de Seguimiento de la Mano	16	1	3	15
2	2	Ingreso a la Pantalla Principal de la Aplicación	2	2	1	5
3	3	Ingreso al Menú de Videojuegos de la Aplicación	4	2	2	5
4	14	Pantalla de Instrucciones del Videojuego	2	2	3	5
5	10	Ingreso al Videojuego de Izquierda - Derecha	16	3	2	10
6	9	Pantalla de Fin de Juego	8	3	3	5
7	11	Ingreso al Videojuego de Arriba - Abajo	16	4	3	15
8	13	Ingreso al Videojuego de Laberinto	32	5	3	15
9	4	Ingreso al Videojuego de Entrenamiento	8	6	1	4
10	5	Actividad de Entrenamiento Izquierda	4	6	2	3
11	6	Actividad de Entrenamiento Derecha	4	6	2	3
12	7	Actividad de Entrenamiento Arriba	4	6	3	3
13	8	Actividad de Entrenamiento Abajo	4	6	3	2
14	12	Ingreso al Videojuego de Adentro - Afuera	32	7	3	15
				TOTAL	21	105

4.2.3. Plan de Iteraciones

Después de clasificar y ordenar las historias de usuario se determinan 7 iteraciones a ser desarrolladas y probadas en cada ciclo de acuerdo al orden establecido.

Tabla 25. Planificación de la Iteración 1

Iteración	1
-----------	---

Orden	Número de Historia	Tarea	Nombre de la historia	Asignado	Tiempo/Semanas	Tiempo/Días
1	1		Creación del Sistema de Seguimiento de la Mano		3	15
		1	Análisis	Analista		1
		2	Diseño	Arquitecto		2
		3	Implementación	Programador		10
		4	Pruebas	Tester		2
					Total	15

Tabla 26. Planificación de la Iteración 2

Iteración	2
-----------	---

Orden	Número de Historia	Tarea	Nombre de la historia	Asignado	Tiempo/Semanas	Tiempo/Días
2	2		Ingreso a la Pantalla Principal de la Aplicación		1	5
		1	Análisis	Analista		1
		2	Diseño	Arquitecto		1
		3	Implementación	Programador		2
		4	Pruebas	Tester		1
3	3		Ingreso al Menú de Videojuegos de la Aplicación		2	5
		1	Análisis	Analista		1
		2	Diseño	Arquitecto		1
		3	Implementación	Programador		2
		4	Pruebas	Tester		1
4	14		Pantalla de Instrucciones del Videojuego		3	5
		1	Análisis	Analista		1
		2	Diseño	Arquitecto		1
		3	Implementación	Programador		2
		4	Pruebas	Tester		1
					Total	15

Tabla 27. Planificación de la Iteración 3

Iteración	3
-----------	---

Orden	Número de Historia	Tarea	Nombre de la historia	Asignado	Tiempo/Semanas	Tiempo/Días
5	10		Ingreso al Videojuego de Izquierda - Derecha		2	10
		1	Análisis	Analista		1
		2	Diseño	Arquitecto		2
		3	Implementación	Programador		5
		4	Pruebas	Tester		2
6	9		Pantalla de Fin de Juego		3	5
		1	Análisis	Analista		1
		2	Diseño	Arquitecto		1
		3	Implementación	Programador		2
		4	Pruebas	Tester		1
					Total	15

Tabla 28. Planificación de la Iteración 4

Iteración	4
-----------	---

Orden	Número de Historia	Tarea	Nombre de la historia	Asignado	Tiempo/Semanas	Tiempo/Días
7	11		Ingreso al Videojuego de Arriba - Abajo		3	15
		1	Análisis	Analista		2
		2	Diseño	Arquitecto		2
		3	Implementación	Programador		9
		4	Pruebas	Tester		2
					Total	15

Tabla 29. Planificación de la Iteración 5

Iteración	5
-----------	---

Orden	Número de Historia	Tarea	Nombre de la historia	Asignado	Tiempo/Semanas	Tiempo/Días
8	13		Ingreso al Videojuego de Laberinto		3	15
		1	Análisis	Analista		2
		2	Diseño	Arquitecto		2
		3	Implementación	Programador		9
		4	Pruebas	Tester		2
					Total	15

Tabla 30. Planificación de la Iteración 6

Iteración	6
-----------	---

Orden	Número de Historia	Tarea	Nombre de la historia	Asignado	Tiempo/Semanas	Tiempo/Días
9	4		Ingreso al Videojuego de Entrenamiento		1	4
		1	Análisis	Analista		1

		2	Diseño	Arquitecto		1
		3	Implementación	Programador		1
		4	Pruebas	Tester		1
10	5		Actividad de Entrenamiento Izquierda		2	3
		1	Análisis	Analista		0.5
		2	Diseño	Arquitecto		0.5
		3	Implementación	Programador		1.5
		4	Pruebas	Tester		0.5
11	6		Actividad de Entrenamiento Derecha		2	3
		1	Análisis	Analista		0.5
		2	Diseño	Arquitecto		0.5
		3	Implementación	Programador		1.5
		4	Pruebas	Tester		0.5
12	7		Actividad de Entrenamiento Arriba		3	3
		1	Análisis	Analista		0.5
		2	Diseño	Arquitecto		0.5
		3	Implementación	Programador		1.5
		4	Pruebas	Tester		0.5
13	8		Actividad de Entrenamiento Abajo		3	2
		1	Análisis	Analista		0.5
		2	Diseño	Arquitecto		0.5
		3	Implementación	Programador		0.5
		4	Pruebas	Tester		0.5
					Total	15

Tabla 31. Planificación de la Iteración 7

Iteración	7
-----------	---

Orden	Número de Historia	Tarea	Nombre de la historia	Asignado	Tiempo/Semanas	Tiempo/Días
14	12		Ingreso al Videojuego de Adentro - Afuera		3	15
		1	Análisis	Analista		2
		2	Diseño	Arquitecto		2
		3	Implementación	Programador		9
		4	Pruebas	Tester		2
					Total	15

4.2.4. Propósito

Permite especificar los requerimientos funcionales y no funcionales para el desarrollo del software. Dando una visión clara de los componentes y funcionalidades del videojuego, especificando detalles, características y restricciones.

Sirve como guía para todas las personas involucradas en el desarrollo del videojuego.

4.2.5. Alcance

UBIC es un videojuego especialmente diseñado para la interacción del usuario en su ubicación espacial de izquierda a derecha, arriba y abajo, adentro y afuera, encima y debajo.

Permite reforzar la enseñanza-aprendizaje de la ubicación espacial de las niñas de preescolar.

4.2.6. Definiciones, Siglas y Abreviaciones

Tabla 32. Siglas y Abreviaciones

Termino	Descripción
GUI	Interfaces Graficas de Usuario
NUI	Interfaces Naturales de Usuario

4.2.7. Perspectiva del Producto

Es un videojuego creado para fortalecer el proceso de enseñanza-aprendizaje de las niñas de 5 y 6 años, en la temática de ubicación espacial. El software está diseñado para ser didáctico y que pueda ser utilizado en la clase de una manera fácil.

4.2.8. Interfaces con el Usuario

UBIC cuenta con interfaces que permitan interactuar con el usuario mejorando la experiencia cuando utilice el software.

La Figura 10 presenta las interfaces usadas en el desarrollo del videojuego de ubicación espacial.

Teclado	<ul style="list-style-type: none"> • Interfaz que permite pausar el juego con la tecla P, Barra Espaciadora que permite desactivar el sonido de fondo, tecla v para activar el sonido de fondo, tecla de Back Space que permite regresar de una pantalla a otra, tecla Esc para salir del juego.
Mouse	<ul style="list-style-type: none"> • Interfaz que permite seleccionar las diferentes opciones del menú y navegar dentro del juego haciendo un clic.
Pantalla	<ul style="list-style-type: none"> • Interfaz usada para mostrar las diferentes interfaces graficas del juego, el tamaño o resolución de la pantalla es 1024 x 768 pixeles.
GUI	<ul style="list-style-type: none"> • Interfaz gráfica que permitirá interactuar de forma fácil con el usuario.
Kinect	<ul style="list-style-type: none"> • Interfaz que permite seleccionar los diferentes opciones del menú y navegar dentro del juego haciendo un clic.

Figura 10. Interfaces de Usuario

4.2.9. Interfaces con el Hardware

El videojuego interactúa con el hardware externo Kinect. Este es un dispositivo electrónico de juego diseñado para consolas de Xbox 360, el cual permite reconocer gestos, voz e imágenes.

La Figura 11 presenta los componentes del dispositivo Kinect.

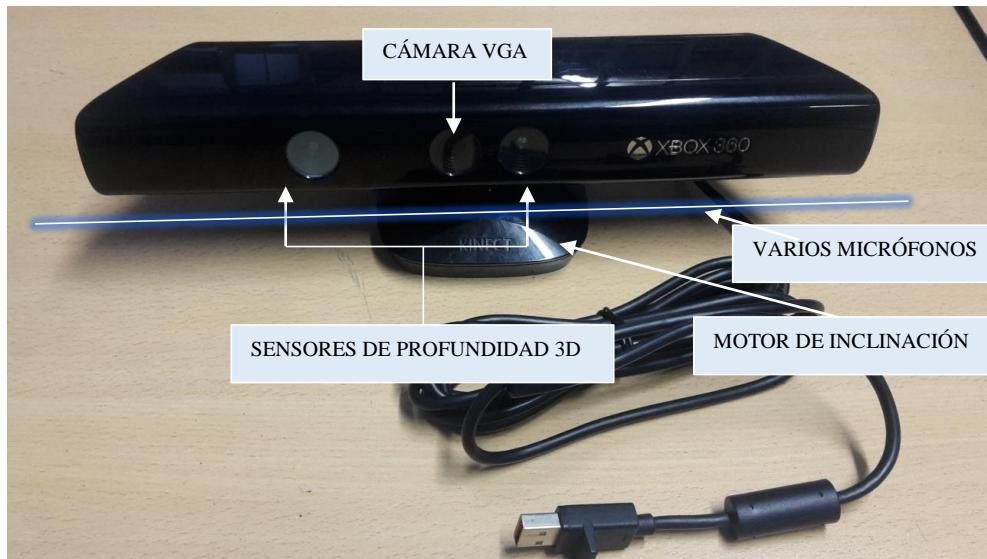


Figura 11. Dispositivo Kinect y sus Componentes

1. Sensores de profundidad 3-D: Estos sensores tridimensionales permiten el seguimiento del cuerpo dentro del área del juego.

2. Una Cámara RGB (Rojo, Verde, Azul): Ayuda a identificarlo y captura imágenes y videos.
3. Varios Micrófonos: Tiene un conjunto de micrófonos en el borde frontal inferior del aparato que permiten el reconocimiento de voz.
4. Inclinación motorizada: Un impulso mecánico en la base del sensor Kinect inclina de manera automática hacia arriba o abajo según sea necesario.
(Microsoft XBOX, 2018).

Conexión:

El sensor Kinect modelo 1473, necesita un adaptador para conectar el dispositivo a un puerto USB y directamente a la energía mostrado en la Figura 12.



Figura 12. Adaptador USB y Conector de Electricidad

4.2.10. Interfaces con el Software

4.2.10.1. Sistema Operativo Ubuntu

Es un sistema operativo de código abierto para computadores de escritorio y portátiles, es una distribución de Linux basada en la arquitectura de Debian. Contiene herramientas de escritorio y desarrollo preestablecidos que están distribuidos bajo una licencia libre o de código abierto, también permite descargar aplicaciones y actualizaciones desde su centro de software de Ubuntu. (Ubuntu, 2018)

La versión utilizada del sistema operativo Ubuntu es 16.04 LTS.

4.2.10.2. Librería Libfreenect

Es un controlador de espacio de usuario para Microsoft Kinect. Se ejecuta en Linux, OSX y Windows y admite imágenes RGB (Red, Green, Blue) y profundidad, motores, acelerómetro, LED (luz) y Audio. (OpenKinect, 2018)

Nota: Si tiene el Kinect Versión 2 (Xbox One), use Open Kinect/Libfreenect 2.

4.2.11. Restricciones de Memoria

El computador necesita para el funcionamiento adecuado 12 Gigabytes de Memoria RAM.

4.2.12. Modos de Operación de Usuario

1. Pantalla inicial: Al iniciar el videojuego debe seleccionar una de las dos opciones: entrar a los videojuegos y salir.
2. Entrenamiento: El usuario tiene una interfaz de práctica que le permite conocer la interacción que se muestra en los videojuegos con actividades de entrenamiento: Izquierda, Derecha, Arriba y Abajo.
3. Salir del videojuego: Permite al usuario abandonar el videojuego, de igual manera presionando una sola vez la tecla ESC.
4. Pausar el videojuego: Permite al usuario pausar el videojuego durante 5 segundos presionando una sola vez la tecla p.
5. Regresar a la pantalla anterior: Permite al usuario regresar a una pantalla anterior con la tecla Back Space. Si se encuentra en un videojuego iniciado automáticamente cerrara el videojuego en el que se encuentra.
6. Desactivar el sonido del videojuego: Permite al usuario desactivar el sonido de fondo del videojuego presionando una sola vez la Barra Espaciadora.
7. Activar el sonido del videojuego: Permite al usuario activar el sonido de fondo del videojuego presionando una sola vez la tecla v.
8. Menú de videojuegos: Contiene un listado de cuatro videojuegos y el entrenamiento, permitiendo seleccionar uno a la vez.

9. Mover el personaje del videojuego Izquierda - Derecha: Durante la partida del videojuego el usuario se desplaza de izquierda a derecha por medio de NUI representada por la detección de la mano cerrada que tiene como semejanza al clic del mouse.
10. Mover el personaje del videojuego Arriba - Abajo: Durante la partida del videojuego el usuario se desplaza de arriba hacia abajo por medio de NUI representada por la detección de la mano cerrada que tiene como semejanza al clic del mouse.
11. Mover el personaje del videojuego Adentro - Afuera: Durante la partida del videojuego el usuario arrastra los objetos de adentro hacia afuera y viceversa, aplicando los conocimientos adquiridos de los videojuegos anteriores Izquierda - Derecha y Arriba - Abajo por medio de NUI representada por la detección de la mano cerrada que tiene como semejanza al clic del mouse.
12. Mover el personaje del videojuego Laberinto (arriba y abajo, derecha e izquierda): Durante la partida del videojuego el usuario se desplazará de arriba hacia abajo, derecha e izquierda por medio de NUI representada por la detección de la mano cerrada que tiene como semejanza al clic del mouse.
13. Instrucciones: Al iniciar cada videojuego se muestra las instrucciones del mismo.
14. Pantalla de Fin de Juego: Contiene el puntaje obtenido por el jugador, el tiempo de realización de la actividad del videojuego y tres opciones: Reiniciar videojuego, Regresar al Menú de Videojuegos y Salir.

4.2.13. Funciones del Producto

El videojuego es educativo y didáctico permitiendo reforzar las nociones básicas con respecto a la ubicación espacial.

En la Figura 13 se presenta el diagrama de flujo del comportamiento y la lógica del reconocimiento de gestos por medio del dispositivo Kinect.

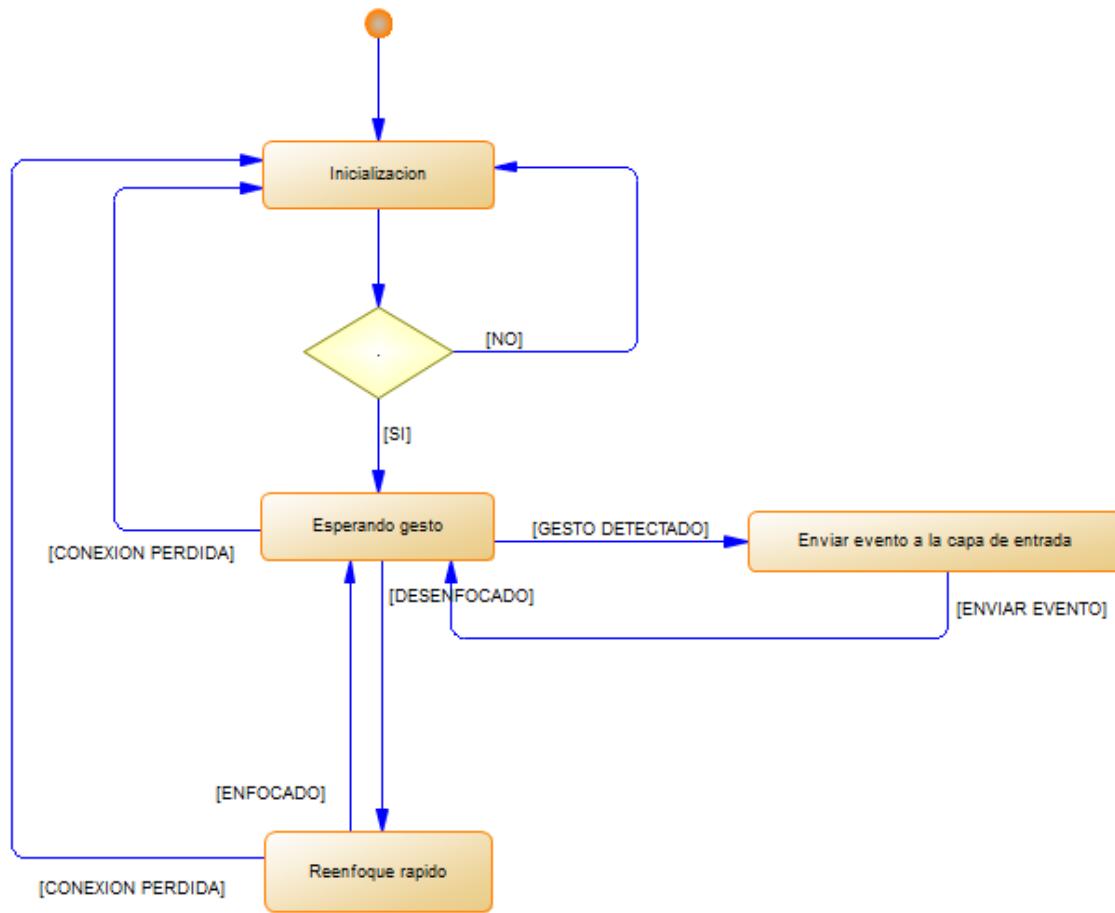


Figura 13. Diagrama de Comportamiento del Dispositivo Kinect

4.2.14. Características del Usuario

En la tabla 33 se describe las características principales de los usuarios.

Tabla 33. Características del Usuario

Rol	Descripción	Estudios o Experiencia Técnica
Niñas entre 5 y 6 años de edad.	Utiliza el videojuego como medio de aprendizaje.	Reconocer objetos colores y nociones básicas de ubicación espacial.
Docente Parvulario.	Utiliza el videojuego como herramienta de fortalecimiento de la enseñanza de ubicación espacial.	Tener conocimientos acerca del manejo del computador.

4.2.15. Restricciones Generales

- El usuario debe leer todas las instrucciones para entender la dinámica del videojuego.
- El videojuego tiene un solo idioma que es el español.
- Cada videojuego termina cuando el personaje no complete la actividad.
- Si en el videojuego el personaje no completa la actividad, debe iniciar nuevamente para volver a jugar.
- En cada videojuego se debe completar un puntaje límite para terminar la actividad.

4.2.15.1. Restricciones de Hardware

- Mouse.
- Teclado.
- Kinect Xbox360 versión 1.
- Adaptador USB y conector de electricidad.

4.2.15.2. Restricciones del Kinect

- La Figura 14 indica la distancia del usuario con respecto al Dispositivo Kinect que debe ser de 3 metros.

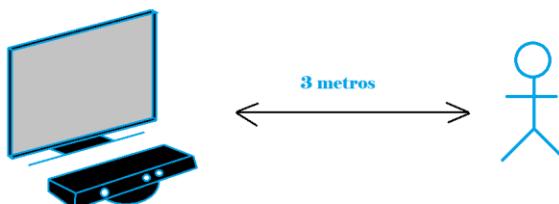


Figura 14. Distancia del Usuario con Respecto al Kinect

- La Figura 15 muestra el estado de conexión del dispositivo Kinect cuando la luz verde está encendida indica que el Kinect está conectado.

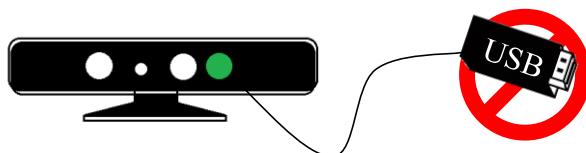


Figura 15. Kinect Estado Conectado.

- La Figura 16 muestra cuento el dispositivo Kinect detecta un objeto, la luz roja se enciende indicando que el Kinect detecta tu cuerpo.

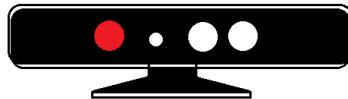


Figura 16. Kinect Detectando al Objeto

4.2.15.3. Restricciones de Software

- El videojuego está desarrollado en lenguaje Python versión 2.7.12.
- El videojuego funciona bajo el Sistema Operativo Ubuntu 16.04 LTS.
- El videojuego necesita instalar las siguientes librerías:
 - Librería pygame.
 - Librería Freenect.
 - Librería Numpy.
 - Librería OpenCV (cv y cv2).
 - Librería Sys.
 - Librería Time.
 - Librería Random.
 - Librería OS.
 - Librería Xlib.
- El control de versiones se realiza a través de Git y el repositorio está almacenado en GitHub.

4.2.16. Suposiciones

- El usuario debe tener un conocimiento en el manejo del computador.
- El desarrollador debe tener conocimiento respecto a lógica de programación.
- Se debe seguir los requerimientos y no cambiarlos en el desarrollo del videojuego.

4.2.17. Dependencias

- El equipo de desarrollo debe disponer de tiempo en el trámite del proyecto.
- La disponibilidad de las niñas y docentes que permitan aportar información en la investigación del ámbito educativo con respecto a la ubicación espacial.
- Funcionamiento correcto del equipo de desarrollo del videojuego.

- Disponibilidad de las niñas que realizan las pruebas.

4.2.18. Requisitos Específicos

En esta sección se explica los requerimientos de la aplicación de software dividiendo en dos grupos como:

- 1. Requerimientos funcionales:** Son enunciados acerca de servicios que el sistema debe proveer, de cómo debe reaccionar el sistema a entradas particulares y de cómo debe comportarse el sistema en situaciones específicas. En algunos casos, los requerimientos funcionales también explican lo que no debe hacer el sistema.
- 2. Requerimientos no funcionales:** Son limitaciones sobre servicios o funciones que ofrece el sistema. Incluyen restricciones tanto de temporización y del proceso de desarrollo, como impuestas por los estándares. Los requerimientos no funcionales se suelen aplicar al sistema como un todo, más que a características o a servicios individuales del sistema. (Sommerville, 2011)

4.2.18.1. Requerimientos Funcionales

Para realizar la especificación de requerimientos del videojuego se usa una tabla con los siguientes campos descritos a continuación:

- Código del Requerimiento: Requerimientos Funcionales RF-01 y Requerimientos no Funcionales RNF-01.
- Historia de Usuario: Historia de Usuario HU-01.
- Nombre del Requerimiento: Nombre que se asigna al requerimiento.
- Descripción: Breve resumen del requisito.
- Precondición: Requisito para que aplicar la actividad actual.
- Secuencia: Pasos a seguir.
- Post Condición: Es el resultado de muestra la actividad.
- Excepciones: Eventos que pueden darse en la aplicación si no cumple la condición.
- Prioridad: Alta, Media, Baja.
- Tipo: Requerimiento Funcional RF o Requerimiento no Funcional RNF.

Tabla 34. Requerimiento Funcional 1. Seguimiento de la Mano

Código	RF-01	
Historia de Usuario	HU-01	
Nombre del Requerimiento	Seguimiento de la Mano.	
Descripción	La detección de la mano lo realiza como dispositivo externo similar al mouse, simulando el movimiento del puntero del mouse con la mano abierta, también simulando el evento de clic del mouse con la mano cerrada.	
Precondición	Los drivers del Kinect deben estar instalados correctamente.	
Secuencia	Paso	Acción
	1	Inicialización del Kinect.
	2	Esperando el gesto “Mano abierta o mano cerrada”.
	3	Gesto detectado “Mano abierta = Movimiento del mouse”, Mano cerrada = clic el mouse”.
	4	Enviando gesto a la capa de entrada “Simulación del mouse”.
	5	Evento enviado .
Post-Condición	Dibuja la silueta de la mano en la pantalla, con el gesto reconocido.	
Excepciones	Paso	Acción
	1	Se puede perder el enfoque de la mano y no hay un reconocimiento valido.
	2	Perdida de la conexión cuando se inicializa.
Prioridad	Alta.	

Tabla 35. Requerimiento Funcional 2. Creación de la Pantalla Principal de la Aplicación

Código	RF-02	
Historia de Usuario	HU-02	
Nombre del Requerimiento	Creación de la Pantalla Principal de la Aplicación.	
Descripción	Es la primera pantalla que aparece al ejecutar la aplicación, la cual contiene el nombre de la aplicación y un menú con dos opciones: Entrar: Al hacer clic se despliega la pantalla del menú de los videojuegos. Salir: Al hacer clic sale de la aplicación.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando e implementado.	
Secuencia	Paso	Acción
	1	Creación del tamaño de la pantalla 1024 x 768 px en full screen (Pantalla completa).
	2	Creación del fondo de la pantalla 1024 x 768 px.
	3	Creación del menú con las dos opciones: entrar y salir.
	4	Crear las acciones del menú: Entrar: Al hacer clic se despliega la pantalla del menú de los videojuegos. Salir: Al hacer clic sale de la aplicación.
	5	Creación del ícono que interactúa en el menú de opciones.
Post-Condición	Ingreso al menú de videojuegos de la aplicación.	
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
Prioridad	Baja.	

Tabla 36. Requerimiento Funcional 3. Creación del Menú de Videojuegos de la Aplicación

Código	RF-03	
Historia de Usuario	HU-03	
Nombre del Requerimiento	Creación del Menú de Videojuegos de la Aplicación.	
Descripción	Es la pantalla del menú donde se encuentran los videojuegos con cinco opciones: Entrenamiento, Izquierda - Derecha, Arriba - Abajo, Adentro – Afuera y Laberinto.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando e implementado. Pantalla principal de la aplicación.	
Secuencia	Paso	Acción
	1	Creación del fondo de la pantalla 1024 x 768 px.
	2	Creación del menú con cinco opciones: Entrenamiento, Izquierda - Derecha, Arriba - Abajo, Adentro - Afuera y Laberinto.
	3	Crear las acciones del menú: Las opciones de los videojuegos: Entrenamiento, Izquierda - Derecha, Arriba - Abajo, Adentro - Afuera, Laberinto, acceden a la pantalla de instrucciones de su respectivo videojuego.
Post-Condición	Se abre la pantalla de instrucciones.	
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
Prioridad	Baja.	

Tabla 37. Requerimiento Funcional 4. Creación de la Pantalla de Instrucciones del Videojuego

Código	RF-04	
Historia de Usuario	HU-14	
Nombre del Requerimiento	Creación de la Pantalla de Instrucciones del Videojuego.	
Descripción	La pantalla aparecerá al inicio de cada videojuego mostrando las instrucciones de las reglas del videojuego.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando e implementado. Menú de videojuegos de la aplicación.	
Secuencia	Paso	Acción
	1	Creación del fondo de la pantalla 1024 x 768 px.
	2	Creación de menú con dos opciones: Anterior y Siguiente.
	3	Crear las acciones del menú: Anterior: Para regresar a la pantalla del menú de los videojuegos. Siguiente: Para iniciar el videojuego.
	4	Creación del título instrucciones con el nombre del videojuego.
	5	Creación de las instrucciones de cada juego.
	6	Mostrar las instrucciones en la pantalla.
Post-Condición	Con el botón siguiente se abre cada uno de los videojuegos.	
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
	2	Si el jugador presiona la tecla Back Space, regresa al menú de los videojuegos.
Prioridad	Media.	

Tabla 38. Requerimiento Funcional 5. Desarrollo del Videojuego Izquierda - Derecha

Código	RF-05	
Historia de Usuario	HU-10	
Nombre del Requerimiento	Desarrollo del Videojuego Izquierda – Derecha.	
Descripción	El videojuego Izquierda - Derecha permite interactuar NUI conjuntamente con Kinect.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando. Pantalla de instrucciones implementada.	
Secuencia	Paso	Acción
	1	Creación del fondo de la pantalla 1024 x 768 px.
	2	Creación de los iconos izquierda y derecha.
	4	Creación del menú con las dos opciones: izquierda y derecha.
	5	Crear un personaje del jugador que es una niña con una canasta.
	6	Ubicar el personaje del jugador en la parte central inferior de la pantalla.
	7	Crear las acciones del menú: Izquierda: Mueve el personaje del jugador hacia el lado izquierdo de la pantalla. Derecha: Mueve el personaje del jugador hacia el lado Derecho de la pantalla.
	8	Crear e implementar el sonido de fondo del videojuego.
	9	Crear un personaje enemigo que es una “Gallina”.
	10	Crear el objeto “huevos” a recoger por el personaje del jugador.
	11	Ubicar al personaje enemigo en la parte superior derecha de la pantalla.
	12	Crear la acción de la gallina arrojando huevos de forma vertical, con un tiempo de 2 segundos.
	13	Crear el movimiento del personaje enemigo de izquierda a derecha.
	14	Crear un marcador que contará los huevos que recoge con una valoración de un punto.
	15	Crear un temporizador inicializado en cero.
	16	Crear la acción de finalizar el videojuego, cuando un huevo pase el límite de la altura de la pantalla, o cuando el puntaje sea igual a 10.
	17	Llamar a pantalla de fin de juego enviando el puntaje obtenido, el tiempo de realización de la actividad y un identificador del videojuego.
Post-Condición	Pantalla de fin de juego	
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
	2	Si el jugador presiona la tecla P, pausa el videojuego durante cinco segundos.
	3	Si el jugador presiona la tecla Back Space, regresa al menú de los videojuegos.
	4	Si el jugador presiona la Barra Espaciadora, desactiva el sonido del videojuego.
	5	Si el jugador presiona la tecla v, activa el sonido del videojuego.
Prioridad	Alta.	

Tabla 39. Requerimiento Funcional 6. Desarrollo de la Pantalla de Fin de Juego

Código	RF-06	
Historia de Usuario	HU-09	
Nombre del Requerimiento	Desarrollo de Pantalla de Fin de Juego.	
Descripción	Muestra una pantalla con el puntaje obtenido, el tiempo de realización de la actividad y un menú de opciones: Reiniciar Videojuego, Regresar al Menú de los Videojuegos y Salir.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando. Pantalla de los videojuegos.	
Secuencia	Paso	Acción
	1	Creación del fondo de la pantalla 1024 x 768 px.
	2	Creación del título fin de juego con el nombre del videojuego finalizado.
	3	Creación del subtítulo puntaje con el marcador obtenido por el jugador.
	4	Creación del subtítulo tiempo con el tiempo realizado de la actividad.
	5	Creación del menú con las tres opciones: Reiniciar Videojuego, Regresar al Menú de los Videojuegos y Salir.
	6	Crear las acciones del menú: Reiniciar Juego: Al hacer clic se reinicia el videojuego. Menú de Juegos: Al hacer clic se despliega la pantalla del menú de los videojuegos. Salir: Al hacer clic sale de la aplicación.
Post-Condición	Reiniciar Juego: Al hacer clic se reinicia el videojuego. Menú de Juegos: Al hacer clic se despliega la pantalla del menú de los videojuegos. Salir: Al hacer clic sale de la aplicación	
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
Prioridad	Baja.	

Tabla 40. Requerimiento Funcional 7. Desarrollo del Videojuego Arriba - Abajo

Código	RF-07	
Historia de Usuario	HU-11	
Nombre del Requerimiento	Desarrollo del Videojuego Arriba – Abajo.	
Descripción	El videojuego permite interactuar NUI conjuntamente con Kinect.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando. Pantalla de instrucciones implementada.	
Secuencia	Paso	Acción
	1	Creación del fondo de la pantalla 1024 x 768 px.
	2	Creación de los iconos Arriba y Abajo.
	4	Creación del menú con las dos opciones: Arriba y Abajo.
	5	Crear un personaje del jugador que es una pez.
	6	Ubicar el personaje del jugador en la parte central izquierda de la pantalla.
	7	Crear las acciones del menú:

		Arriba: Mueve el personaje del jugador hacia la parte superior de la pantalla. Abajo: Mueve el personaje del jugador hacia la parte inferior de la pantalla.
8		Crear e implementar el sonido de fondo del videojuego.
9		Crear una lista de obstáculos que son los enemigos.
10		Crear el objeto “estrella” a recoger por el personaje del jugador.
11		Ubicar la lista de obstáculos en la parte derecha de la pantalla.
12		Crear la acción de los obstáculos para que se muevan de derecha a izquierda.
13		Ubicar las estrellas entre los obstáculos.
14		Crear en marcador que cuenta el número de estrellas recogidas por el personaje del jugador con una valoración de un punto.
15		Crear un temporizador inicializado en cero.
16		Crear la acción de finalizar el videojuego, cuando el personaje del jugador choque con un obstáculo o cuando el puntaje sea igual a 20.
17		Llamar a pantalla de fin de juego enviando el puntaje obtenido, el tiempo de realización de la actividad y un identificador del videojuego.
Post-Condición		Pantalla de fin de juego
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
	2	Si el jugador presiona la tecla P, pausa el videojuego durante cinco segundos.
	3	Si el jugador presiona la tecla Back Space, regresa al menú de los videojuegos.
	4	Si el jugador presiona la Barra Espaciadora, desactiva el sonido del videojuego.
	5	Si el jugador presiona la tecla v, activa el sonido del videojuego.
Prioridad		Alta.

Tabla 41. Requerimiento Funcional 8. Desarrollo del Videojuego Laberinto

Código	RF-08	
Historia de Usuario	HU-13	
Nombre del Requerimiento	Desarrollo del Videojuego de Laberinto.	
Descripción	El videojuego de laberinto permite interactuar NUI conjuntamente con Kinect.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando. Pantalla de instrucciones implementada.	
Secuencia	Paso	Acción
	1	Creación del fondo de la pantalla 1024 x 768 px.
	2	Creación de los iconos Arriba y Abajo, Izquierda y Derecha.
	3	Creación del menú con las cuatro opciones: Arriba, Abajo, Izquierda y Derecha.

	4	<p>Crear un archivo plano de mapa.txt que contendrá el mapa con los siguientes caracteres:</p> <table border="1"> <thead> <tr> <th>Carácter</th><th>Equivalencia</th><th>Imagen</th></tr> </thead> <tbody> <tr> <td>K</td><td>Nuez</td><td>q.png</td></tr> <tr> <td>-</td><td>Pared horizontal</td><td>h.png</td></tr> <tr> <td> </td><td>Pared vertical</td><td>v.png</td></tr> <tr> <td>T</td><td>Esquina superior izquierda</td><td>si.png</td></tr> <tr> <td>L</td><td>Esquina inferior izquierda</td><td>ii.png</td></tr> <tr> <td>7</td><td>Esquina superior derecha</td><td>sd.png</td></tr> <tr> <td>J</td><td>Esquina inferior derecha</td><td>id.png</td></tr> <tr> <td>A</td><td>Unión en T inferior</td><td>ta.png</td></tr> <tr> <td>S</td><td>Unión en T superior</td><td>ts.png</td></tr> <tr> <td>D</td><td>Unión en T derecha</td><td>td.png</td></tr> <tr> <td>Z</td><td>Unión en T izquierda</td><td>ti.png</td></tr> </tbody> </table>	Carácter	Equivalencia	Imagen	K	Nuez	q.png	-	Pared horizontal	h.png		Pared vertical	v.png	T	Esquina superior izquierda	si.png	L	Esquina inferior izquierda	ii.png	7	Esquina superior derecha	sd.png	J	Esquina inferior derecha	id.png	A	Unión en T inferior	ta.png	S	Unión en T superior	ts.png	D	Unión en T derecha	td.png	Z	Unión en T izquierda	ti.png
Carácter	Equivalencia	Imagen																																				
K	Nuez	q.png																																				
-	Pared horizontal	h.png																																				
	Pared vertical	v.png																																				
T	Esquina superior izquierda	si.png																																				
L	Esquina inferior izquierda	ii.png																																				
7	Esquina superior derecha	sd.png																																				
J	Esquina inferior derecha	id.png																																				
A	Unión en T inferior	ta.png																																				
S	Unión en T superior	ts.png																																				
D	Unión en T derecha	td.png																																				
Z	Unión en T izquierda	ti.png																																				
	5	Crear el objeto pared que sirve para la creación del mapa																																				
	6	Crear el objeto “nuez” a recoger por el personaje del jugador																																				
	7	Crear el objeto “mapa” que se utiliza para crear el laberinto usando el archivo mapa.txt y las imágenes de las paredes y de la nuez.																																				
	8	Crear un personaje del jugador que es una ardilla.																																				
	9	Ubicar el personaje del jugador en la parte superior derecha de la pantalla.																																				
	10	<p>Crear las acciones del menú:</p> <p>Arriba: Mueve el personaje del jugador hacia arriba dentro del laberinto</p> <p>Abajo: Mueve el personaje del jugador hacia abajo dentro del laberinto</p> <p>Izquierda: Mueve el personaje del jugador hacia el lado izquierdo dentro del laberinto</p> <p>Derecha: Mueve el personaje del jugador hacia el lado derecho dentro del laberinto</p>																																				
	11	Crear e implementar el sonido de fondo del videojuego.																																				
	12	Crear un personaje enemigo que es una “Árbol”.																																				
	13	Ubicar el personaje enemigo en la parte inferior izquierda.																																				
	14	Crear la acción del movimiento del personaje del jugador por medio de las NUI.																																				
	15	Crear en marcador que cuenta el número de nueces recogidas por el personaje del jugador con una valoración de un punto.																																				
	16	Crear un temporizador inicializado en cero.																																				
	17	Crear la acción de finalizar el videojuego, cuando el personaje del jugador choque con el personaje enemigo, que indica que ha completado el laberinto.																																				
	18	Llamar a pantalla de fin de juego enviando el puntaje obtenido, el tiempo de realización de la actividad y un identificador del videojuego.																																				
Post-Condición		Pantalla de fin de juego																																				
Excepciones	Paso	Acción																																				
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.																																				
	2	Si el jugador presiona la tecla P, pausa el videojuego durante cinco segundos.																																				

	3	Si el jugador presiona la tecla Back Space, regresa al menú de los videojuegos.
	4	Si el jugador presiona la Barra Espaciadora, desactiva el sonido del videojuego.
	5	Si el jugador presiona la tecla v, activa el sonido del videojuego.
Prioridad	Alta.	

Tabla 42. Requerimiento Funcional 9. Desarrollo del Videojuego de Entrenamiento Izquierda

Código	RF-09	
Historia de Usuario	HU-04, HU-05	
Nombre del Requerimiento	Desarrollo del Videojuego de Entrenamiento Izquierda.	
Descripción	El videojuego de entrenamiento izquierda permite interactuar NUI conjuntamente con Kinect.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando. Pantalla de instrucciones implementada.	
Secuencia	Paso	Acción
	1	Creación del fondo de la pantalla 1024 x 768 px, en la cual se divide la pantalla con una línea vertical en el centro de la pantalla, en el lado izquierdo de la pantalla se dibuja un círculo con relleno y en el lado derecho un círculo sin relleno.
	2	Creación de los iconos izquierda y derecha.
	3	Creación del menú con las dos opciones: izquierda y derecha.
	4	Creación de un temporizador de 15 segundos por actividad.
	5	Creación del puntaje: Muestra el puntaje de la actividad.
	6	Crear Carita: Muestra una imagen de carita feliz o triste en pantalla.
	7	Crear las acciones del menú: Izquierda: Si selecciona el círculo con relleno aparece una imagen de una carita feliz verificando que es “Correcto”. Derecha: Si selecciona el círculo sin relleno aparece una imagen de una carita triste verificando que es “Incorrecto”. Nota: No aparecen las letras de derecha a izquierda solo aparecen los íconos.
	8	Para la evaluación de la actividad correcta se suma el número de veces que hizo clic en el círculo con relleno.
	9	Muestra el puntaje en el lado superior izquierdo de la pantalla.
Post-Condición	Pantalla de videojuego de entrenamiento derecha.	
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
	2	Si el jugador presiona la tecla P, pausa el videojuego durante cinco segundos.
	3	Si el jugador presiona la tecla Back Space, regresa al menú de los videojuegos.
Prioridad	Media.	

Tabla 43. Requerimiento Funcional 10. Desarrollo del Videojuego de Entrenamiento Derecha

Código	RF-10	
Historia de Usuario	HU-04, HU-06	
Nombre del Requerimiento	Desarrollo del Videojuego de Entrenamiento Derecha.	
Descripción	El videojuego de entrenamiento derecha permite interactuar NUI conjuntamente con Kinect.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando. Pantalla de instrucciones implementada.	
Secuencia	Paso	Acción
	1	Creación del fondo de la pantalla 1024 x 768 px, en la cual se divide la pantalla con una línea vertical en el centro de la pantalla, en el lado izquierdo de la pantalla se dibuja un círculo sin relleno y en el lado derecho un círculo con relleno.
	2	Creación de los iconos izquierda y derecha.
	3	Creación del menú con las dos opciones: izquierda y derecha.
	4	Creación de un temporizador de 15 segundos por actividad.
	5	Creación del puntaje: Muestra el puntaje de la actividad.
	6	Crear Carita: Muestra una imagen de carita feliz o triste en pantalla.
	7	Crear las acciones del menú: Izquierda: Si selecciona el círculo sin relleno aparece una imagen de una carita triste verificando que es “Incorrecto”. Derecha: Si selecciona el círculo con relleno aparece una imagen de una carita feliz verificando que es “Correcto”.
	8	Para la evaluación de la actividad correcta se suma el número de veces que hizo clic en el círculo con relleno.
	9	Muestra el puntaje en el lado superior izquierdo de la pantalla.
Post-Condición	Pantalla de videojuego de entrenamiento arriba.	
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
	2	Si el jugador presiona la tecla P, pausa el videojuego durante cinco segundos.
	3	Si el jugador presiona la tecla Back Space, regresa al menú de los videojuegos.
Prioridad	Media.	

Tabla 44. Requerimiento Funcional 11. Desarrollo del Videojuego de Entrenamiento Arriba

Código	RF-11	
Historia de Usuario	HU-04, HU-07	
Nombre del Requerimiento	Desarrollo del Videojuego de Entrenamiento Arriba.	
Descripción	El videojuego de entrenamiento arriba permite interactuar NUI conjuntamente con Kinect.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando. Pantalla de instrucciones implementada.	
Secuencia	Paso	Acción

	1	Creación del fondo de la pantalla 1024 x 768 px, en la cual se divide la pantalla con una línea Horizontal en el centro de la pantalla, en la parte superior de la pantalla se dibuja un círculo con relleno y en la parte inferior de la pantalla un círculo sin relleno.
	2	Creación de los iconos arriba y abajo.
	3	Creación del menú con las dos opciones: arriba y abajo.
	4	Creación de un temporizador de 15 segundos por actividad.
	5	Creación del puntaje: Muestra el puntaje de la actividad.
	6	Crear Carita: Muestra una imagen de carita feliz o triste en pantalla.
	7	Arriba: Si selecciona el círculo con relleno aparece una imagen de una carita feliz verificando que es “Correcto”. Abajo: Si selecciona el círculo sin relleno aparece una imagen de una carita triste verificando que es “Incorrecto”.
	8	Para la evaluación de la actividad correcta se suma el número de veces que hizo clic en el círculo con relleno.
	9	Muestra el puntaje en el lado superior izquierdo de la pantalla.
Post-Condición		Pantalla de videojuego de entrenamiento abajo
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
	2	Si el jugador presiona la tecla P, pausa el videojuego durante cinco segundos.
	3	Si el jugador presiona la tecla Back Space, regresa al menú de los videojuegos.
Prioridad		Media.

Tabla 45. Requerimiento Funcional 12. Desarrollo del Videojuego de Entrenamiento Abajo

Código	RF-12	
Historia de Usuario	HU-04, HU-08	
Nombre del Requerimiento	Desarrollo del Videojuego de Entrenamiento Abajo	
Descripción	El videojuego de entrenamiento abajo permite interactuar NUI conjuntamente con Kinect.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando. Pantalla de instrucciones implementada.	
Secuencia	Paso	Acción
	1	Creación del fondo de la pantalla 1024 x 768 px, en la cual se divide la pantalla con una línea Horizontal en el centro de la pantalla, en la parte superior de la pantalla se dibuja un círculo sin relleno y en la parte inferior de la pantalla un círculo con relleno.
	2	Creación de los iconos arriba y abajo.
	3	Creación del menú con las dos opciones: arriba y abajo.
	4	Creación de un temporizador de 15 segundos por actividad.
	5	Creación del puntaje: Muestra el puntaje de la actividad.
	6	Crear Carita: Muestra una imagen de carita feliz o triste en pantalla.
	7	Arriba: Si selecciona el círculo sin relleno aparece una imagen de una carita triste verificando que es “Incorrecto”. Abajo: Si selecciona el círculo con relleno aparece una imagen de una carita feliz verificando que es “Correcto”.

	8	Para la evaluación de la actividad correcta se suma el número de veces que hizo clic en el círculo con relleno.
	9	Muestra el puntaje en el lado superior izquierdo de la pantalla.
Post-Condición	Pantalla de fin de juego con los resultados de las actividades.	
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
	2	Si el jugador presiona la tecla P, pausa el videojuego durante cinco segundos.
	3	Si el jugador presiona la tecla Back Space, regresa al menú de los videojuegos.
Prioridad	Media.	

Tabla 46. Requerimiento Funcional 13. Desarrollo del Videojuego Adentro - Afuera

Código	RF-13	
Historia de Usuario	HU-12	
Nombre del Requerimiento	Desarrollo del Videojuego Adentro - Afuera	
Descripción	El videojuego adentro - afuera permite interactuar NUI conjuntamente con Kinect.	
Precondición	El sistema de seguimiento de la mano debe estar funcionando. Pantalla de instrucciones implementada.	
Secuencia	Paso	Acción
	1	Creación del fondo de la pantalla 1024 x 768 px, en la cual se divide con una línea Horizontal el centro de la pantalla, en la parte inferior de la pantalla tiene una imagen de la primera actividad que debe realizar: Una caja que contendrá un objeto llamado “oso”. En la parte superior derecha de la pantalla se encuentra la imagen de una “caja”, donde debe completar la primera actividad que consiste en arrastrar el objeto oso adentro de la caja.
	2	Creación del fondo de la pantalla 1024 x 768 px, en la cual se divide con una línea Horizontal el centro de la pantalla, en la parte inferior de la pantalla tiene una imagen de la segunda actividad que debe realizar: Una caja a un lado del objeto llamado “oso”. En la parte superior derecha de la pantalla se encuentra la imagen de una “caja”, donde debe completar la segunda actividad que consiste en arrastrar el objeto oso afuera de la caja.
	3	Creación del fondo de la pantalla 1024 x 768 px, en la cual se divide con una línea Horizontal el centro de la pantalla, en la parte inferior de la pantalla tienen una imagen de la tercera actividad que debe realizar: Un objeto llamado “oso” que se encuentra encima de una mesa. En la parte superior de la pantalla se encuentra la imagen de una “mesa”, donde debe completar la tercera actividad que consiste en arrastrar el objeto oso encima de la mesa.
	4	Creación del fondo de la pantalla 1024 x 768 px, en la cual se divide con una línea Horizontal el centro de la pantalla, en la parte inferior de la pantalla tiene una imagen de la cuarta actividad que debe realizar: Un objeto llamado “oso” que se encuentra debajo de una mesa. En la parte superior de la pantalla se encuentra la imagen de una “mesa”, donde debe completar la cuarta actividad que consiste en arrastrar el objeto oso debajo de la mesa.

	5	Crear un objeto que se mueve por la pantalla “Oso”.
	6	Crear un objeto objetivo que sirve para la consecución de las tareas.
	7	Ubicar el objeto llamado “oso”, en la parte superior izquierda de la pantalla.
	8	Crear e implementar el sonido de fondo del videojuego.
	9	Crear el movimiento del objeto “Oso” a ser arrastrado.
	10	Creación de un temporizador de 30 segundos por actividad.
	11	Crear en marcador que cuenta la actividad correcta con una valoración de un punto.
	12	Crear la acción de finalizar el videojuego, cuando la actividad no se ha completado.
	13	Llamar a pantalla de fin de juego enviando el puntaje obtenido por el jugador y un identificador del videojuego.
Post-Condición	Pantalla de fin de juego.	
Excepciones	Paso	Acción
	1	Si el jugador presiona la tecla ESC, sale de la aplicación.
	2	Si el jugador presiona la tecla P, pausa el videojuego durante cinco segundos.
	3	Si el jugador presiona la tecla Back Space, regresa al menú de los videojuegos.
	4	Si el jugador presiona la Barra Espaciadora, desactiva el sonido del videojuego.
	5	Si el jugador presiona la tecla v, activa el sonido del videojuego.
Prioridad	Alta.	

4.2.18.2. Requisitos No Funcionales

El presente proyecto tiene como objetivo cumplir con las métricas de usabilidad especificadas en el estándar ISO/IEC 9241-11, considerando:

- Eficacia.
- Eficiencia.
- Satisfacción del usuario.

4.3. Fase de Diseño

La aplicación está fundamentada en el modelo de vista arquitectura 4+1 (del estándar IEEE 1471-2000), en el cual plantea las siguientes vistas.

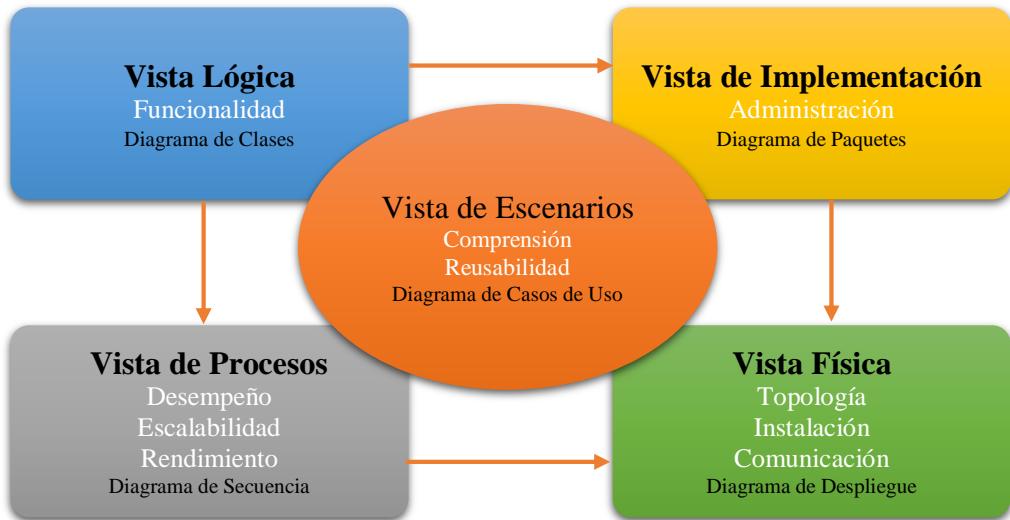


Figura 17. Visión Arquitectónica de la Aplicación

Nota: Adaptado de Romero - Moreno (2004).

4.3.1. Vista Lógica

Describe la funcionalidad que el sistema ofrece a los usuarios finales. Es decir, representa lo que el videojuego debe hacer enfocándose en los requerimientos funcionales resumidos en el diagrama de clases. (Kruchten, 1995)

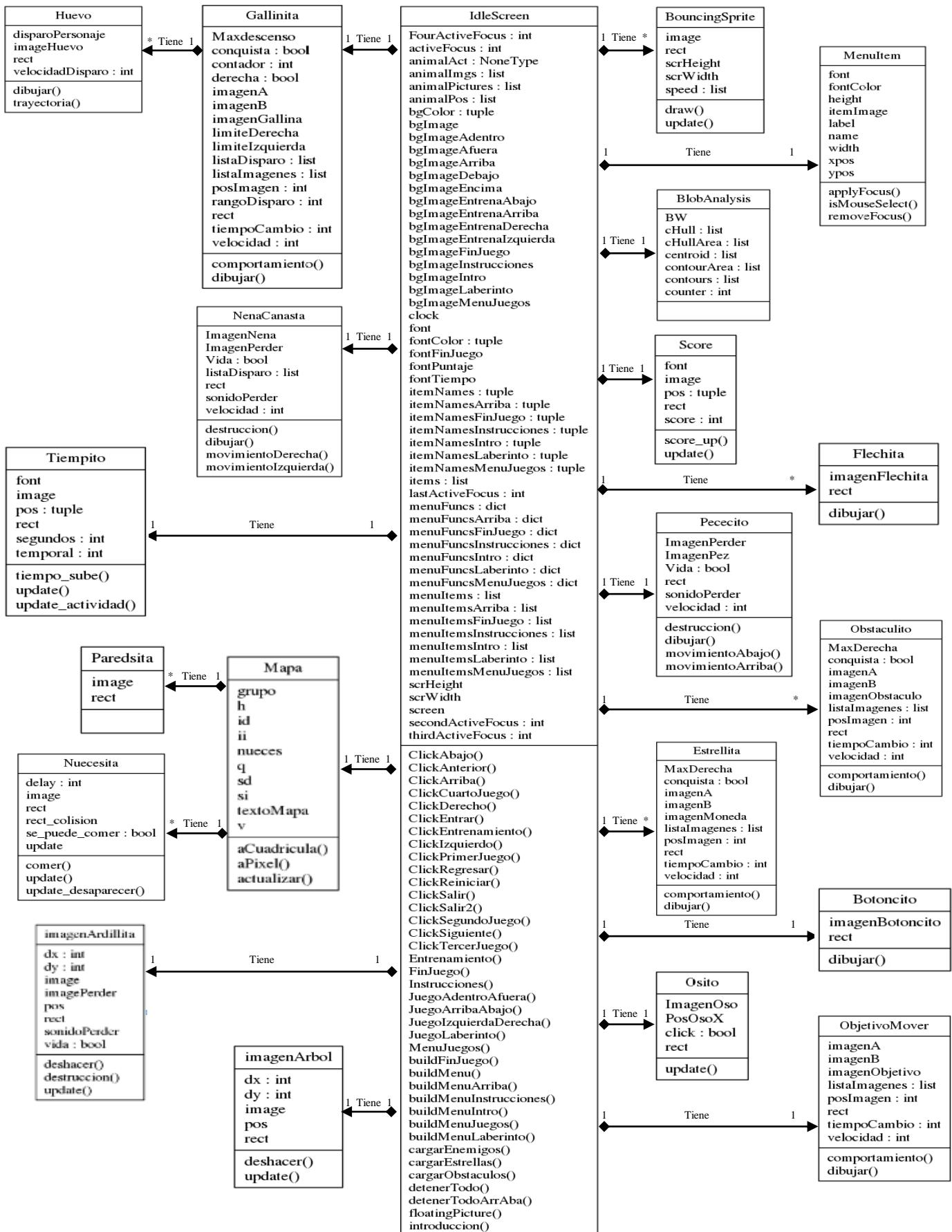


Figura 18. Diagrama de Clases

4.3.2. Vista de Implementación

Se muestra el videojuego desde la perspectiva del programador. Se centra en la organización de las clases del software en el ambiente de desarrollo.

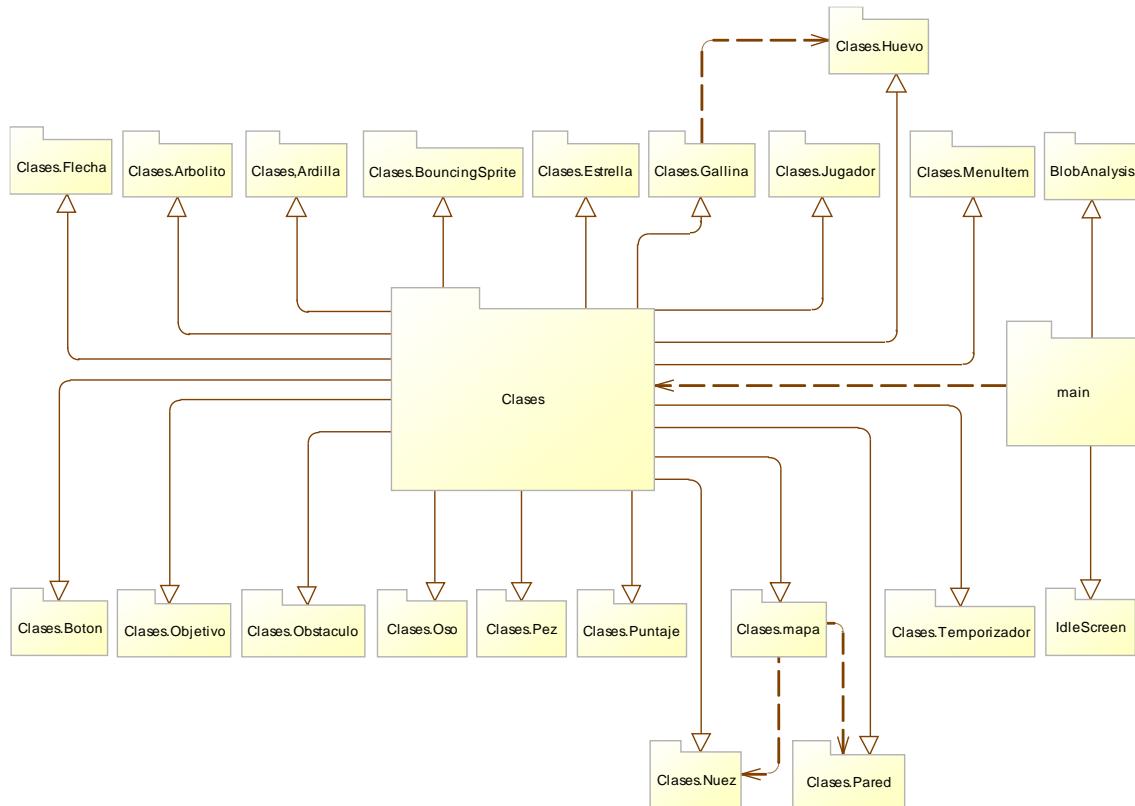


Figura 19. Diagrama de Paquetes

4.3.3. Vista de Procesos

Se muestra los procesos y la forma de cómo se comunican entre ellos. Se representa desde la perspectiva de un integrador de sistemas.

Aquí se muestran los diagramas de secuencia de todos los videojuegos desarrollados en la aplicación.

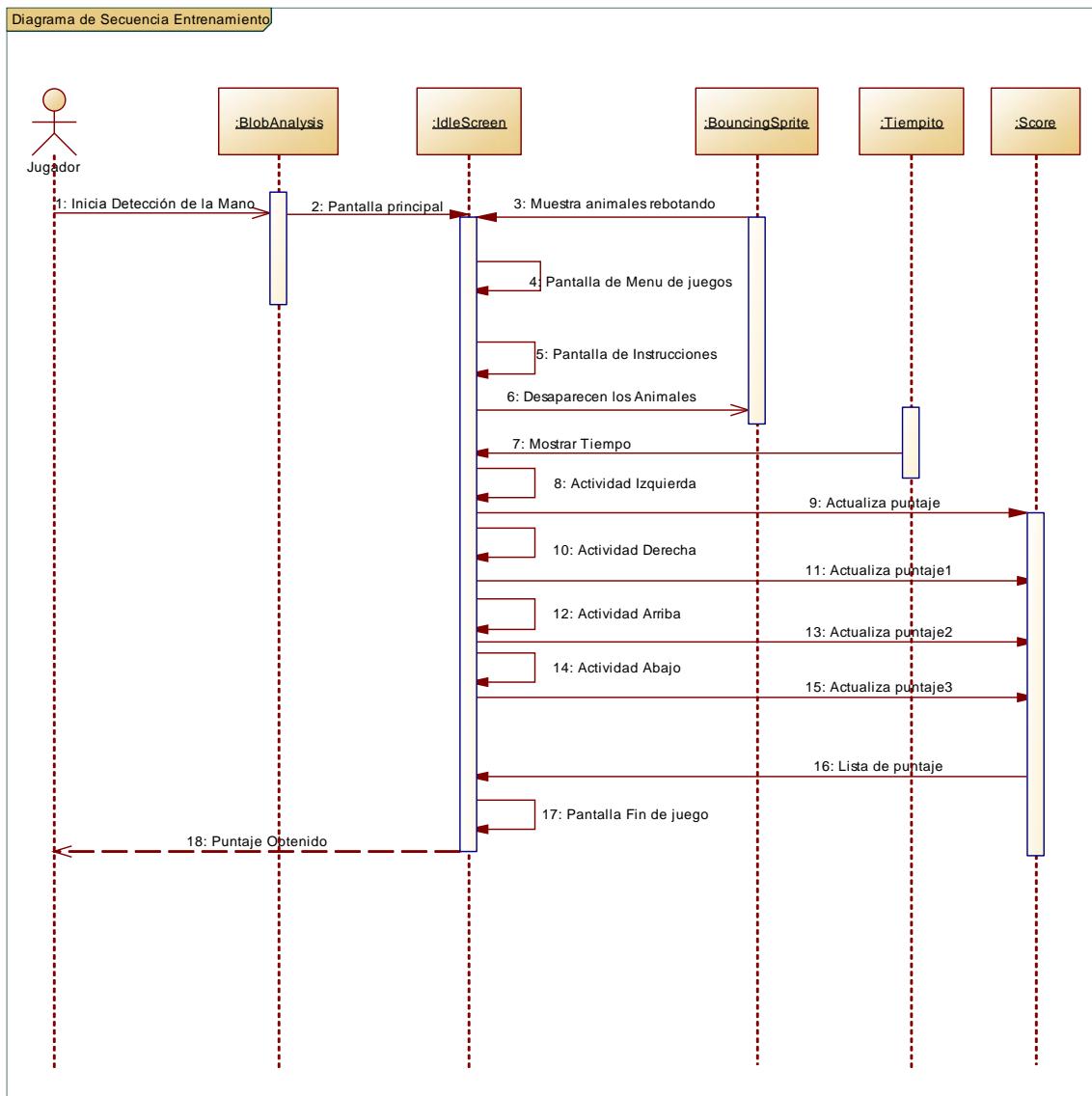


Figura 20. Diagrama de Secuencia del Videojuego de Entrenamiento

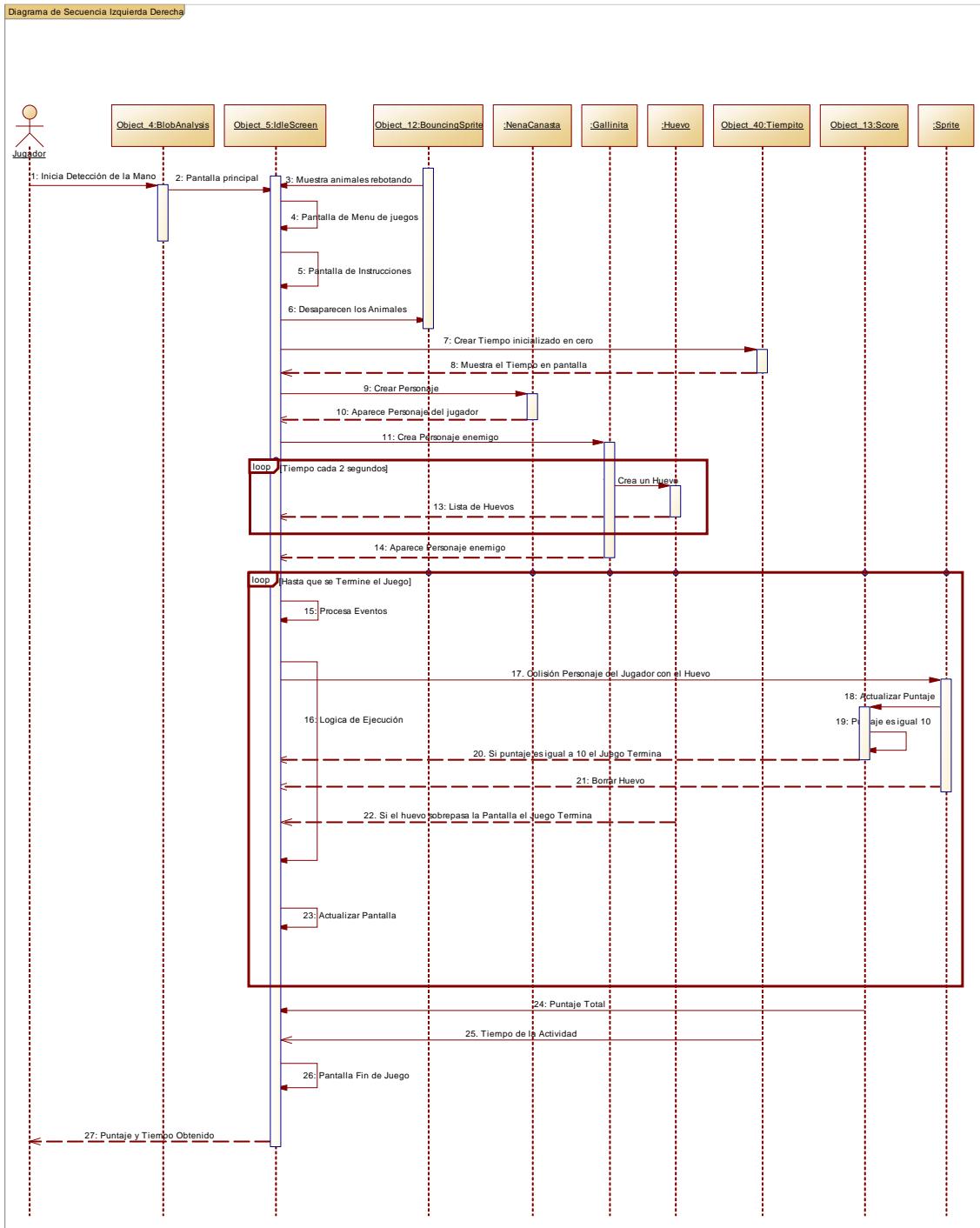


Figura 21. Diagrama de Secuencia del Videojuego Izquierda - Derecha

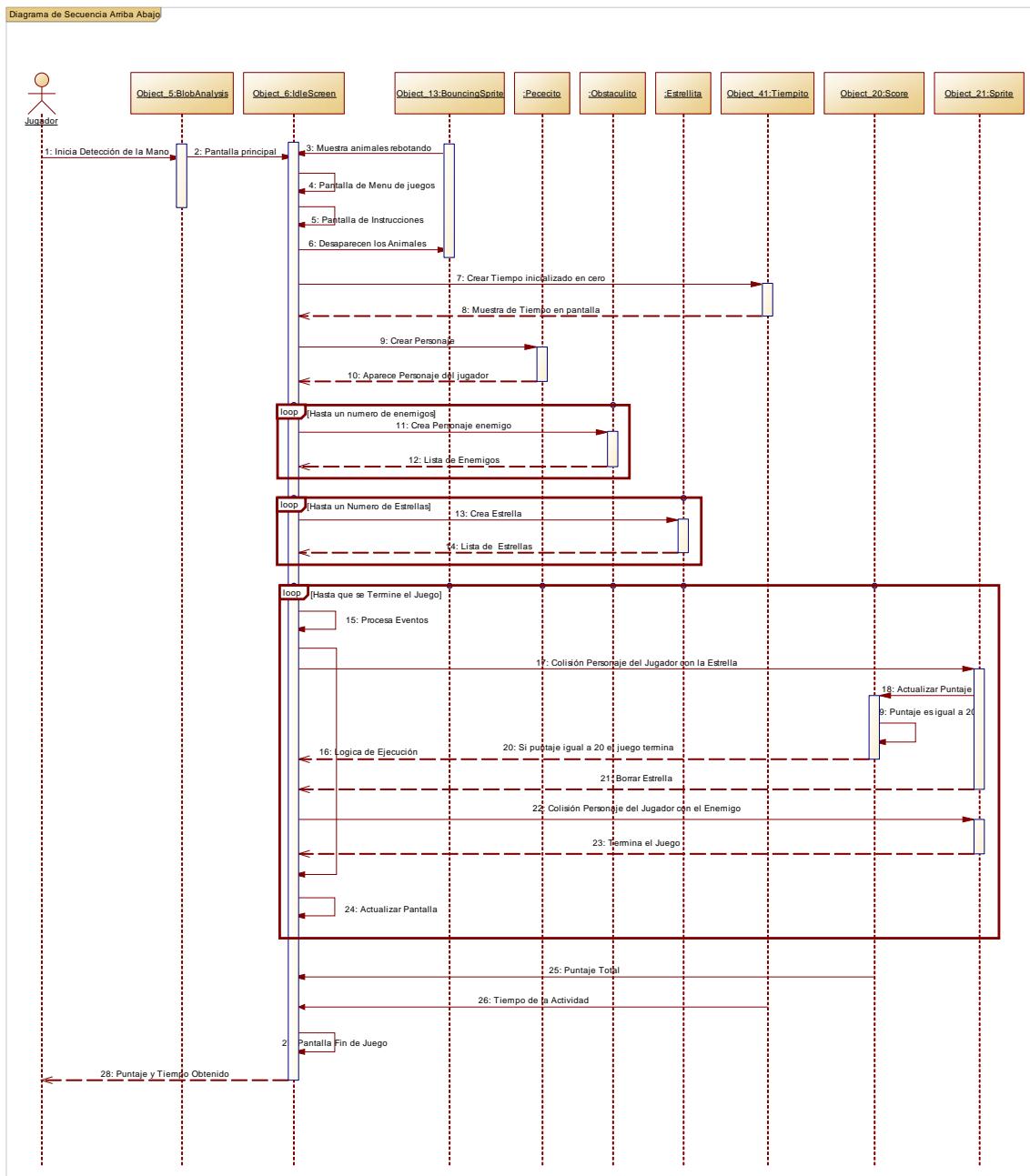


Figura 22. Diagrama de Secuencia del Videojuego Arriba - Abajo

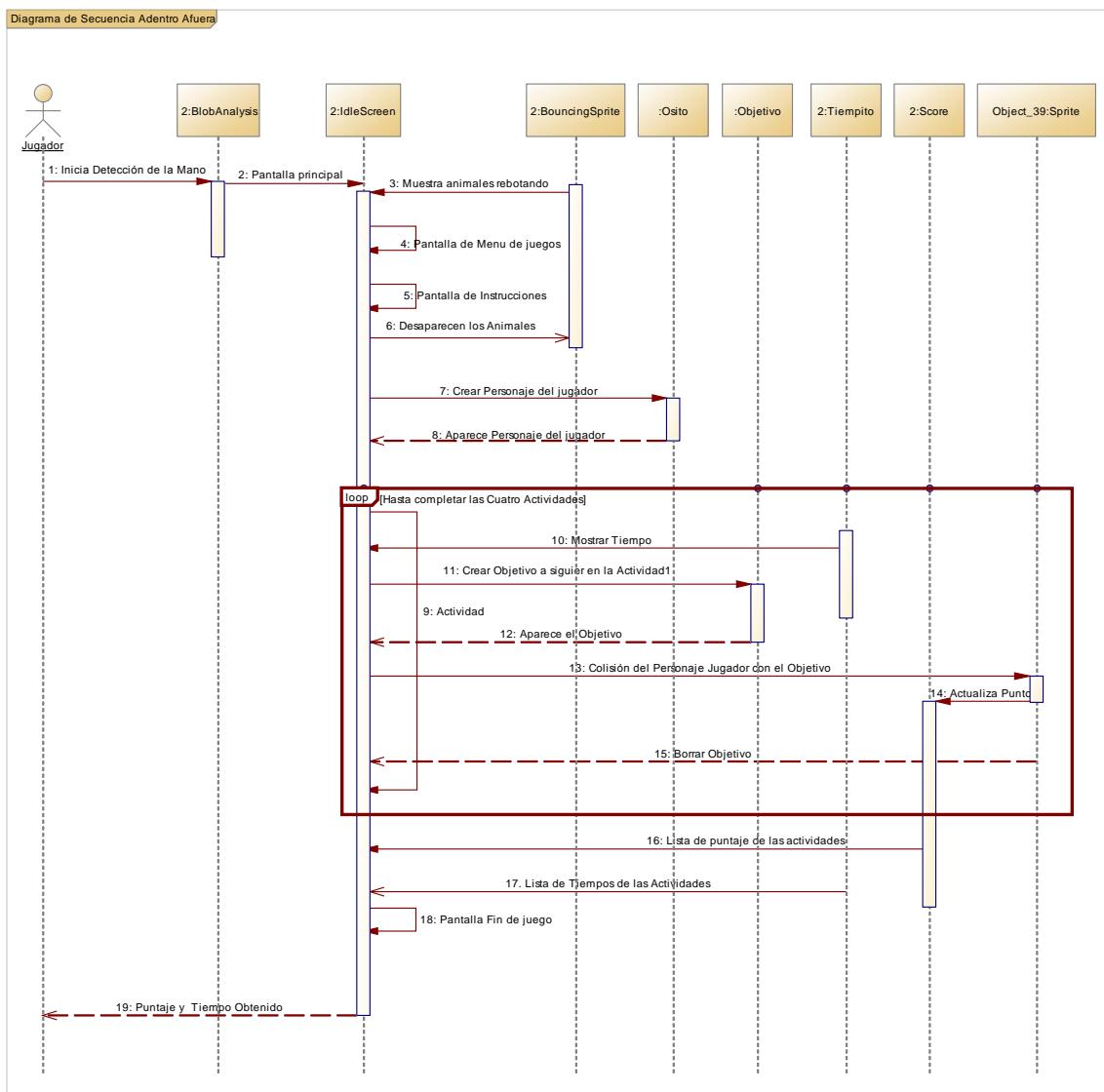


Figura 23. Diagrama de Secuencia del Videojuego Adentro - Afuera

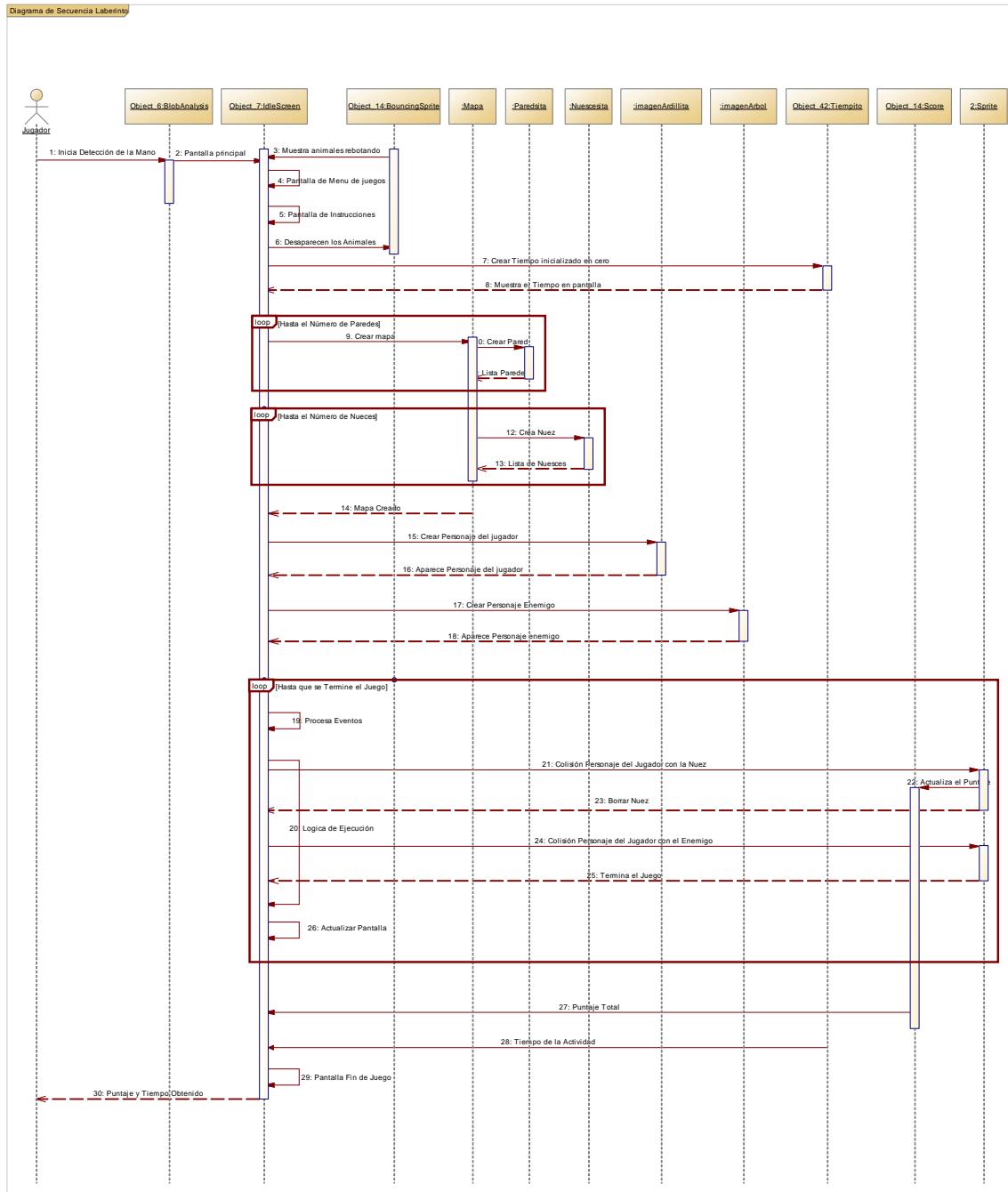


Figura 24. Diagrama de Secuencia del Videojuego Laberinto

4.3.4. Vista Física

Se muestran todos los componentes físicos del sistema y sus conexiones. Se representa desde la perspectiva de un Ingeniero en Sistemas.

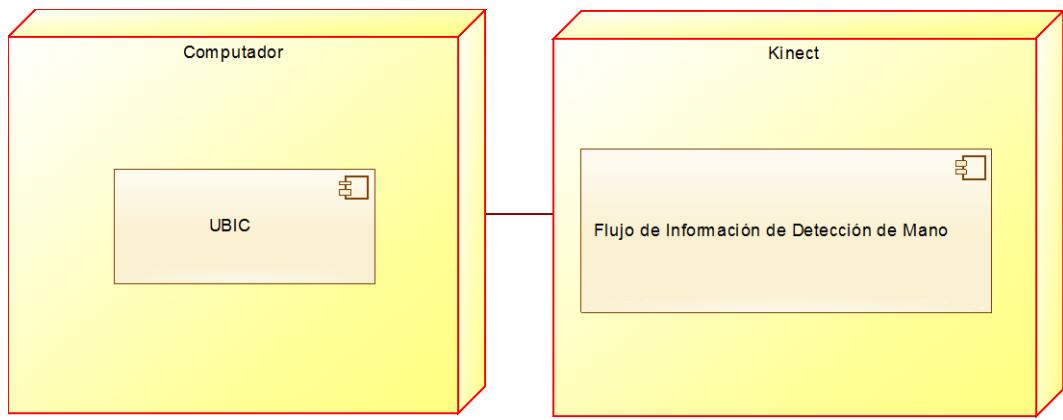


Figura 25. Diagrama de Despliegue

4.3.5. Vista de escenarios

El videojuego desarrollado tiene las siguientes funcionalidades, las cuales son descritas en el diagrama de casos de uso.

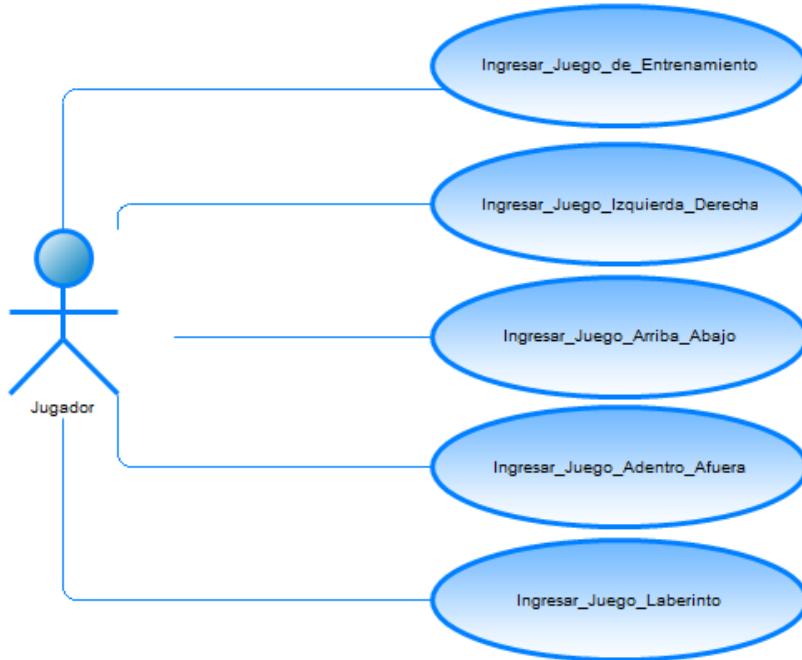


Figura 26. Diagrama de Casos de Uso

4.3.6. Prototipos Conceptuales de la Interfaz de Usuario (IU)

A continuación, se presentan los prototipos conceptuales (mockups) de las interfaces de usuario con la finalidad de validar los requerimientos funcionales de cada uno de los videojuegos por medio de las storyboards.

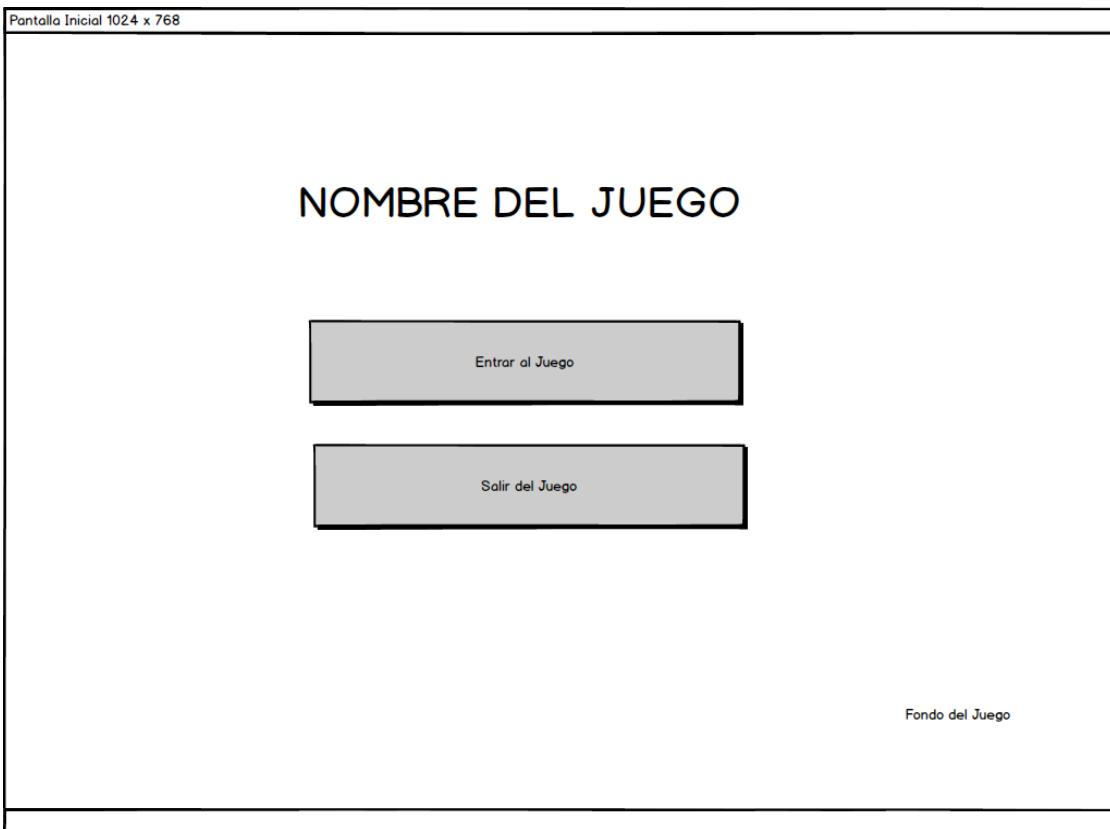


Figura 27. Prototipo de la Pantalla Inicial

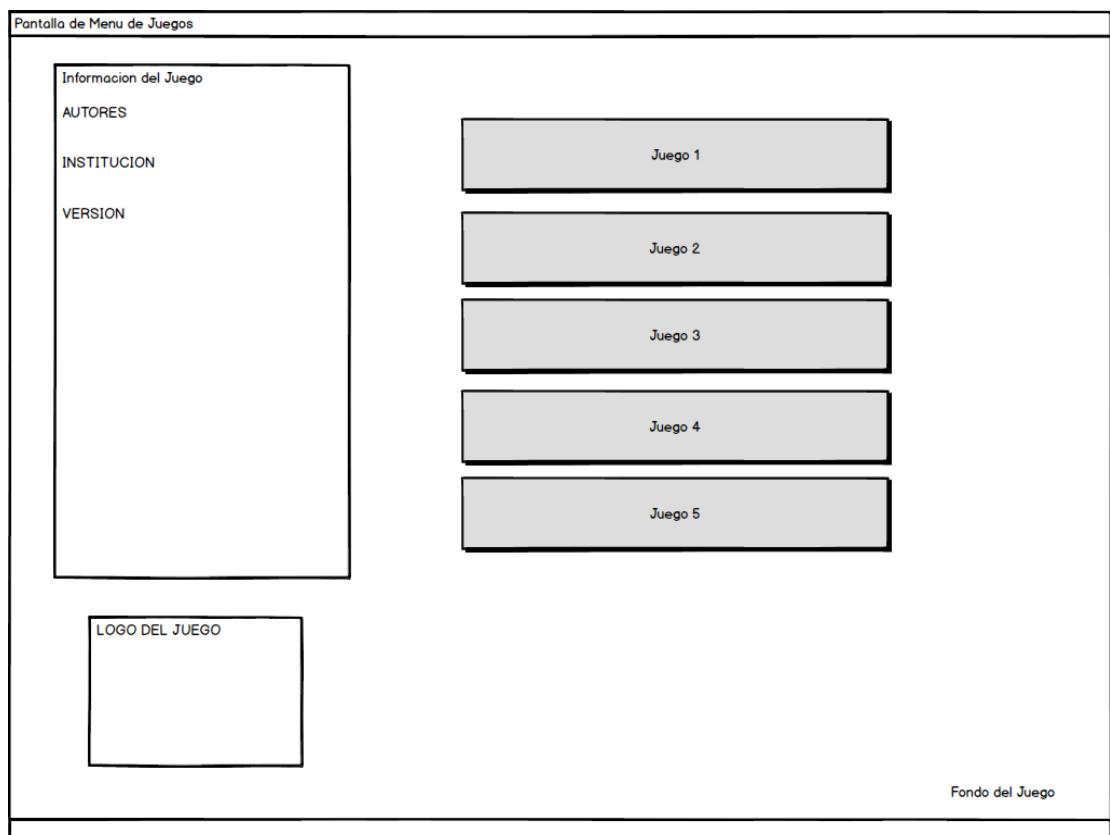


Figura 28. Prototipo de la Pantalla del Menú de los Videojuegos

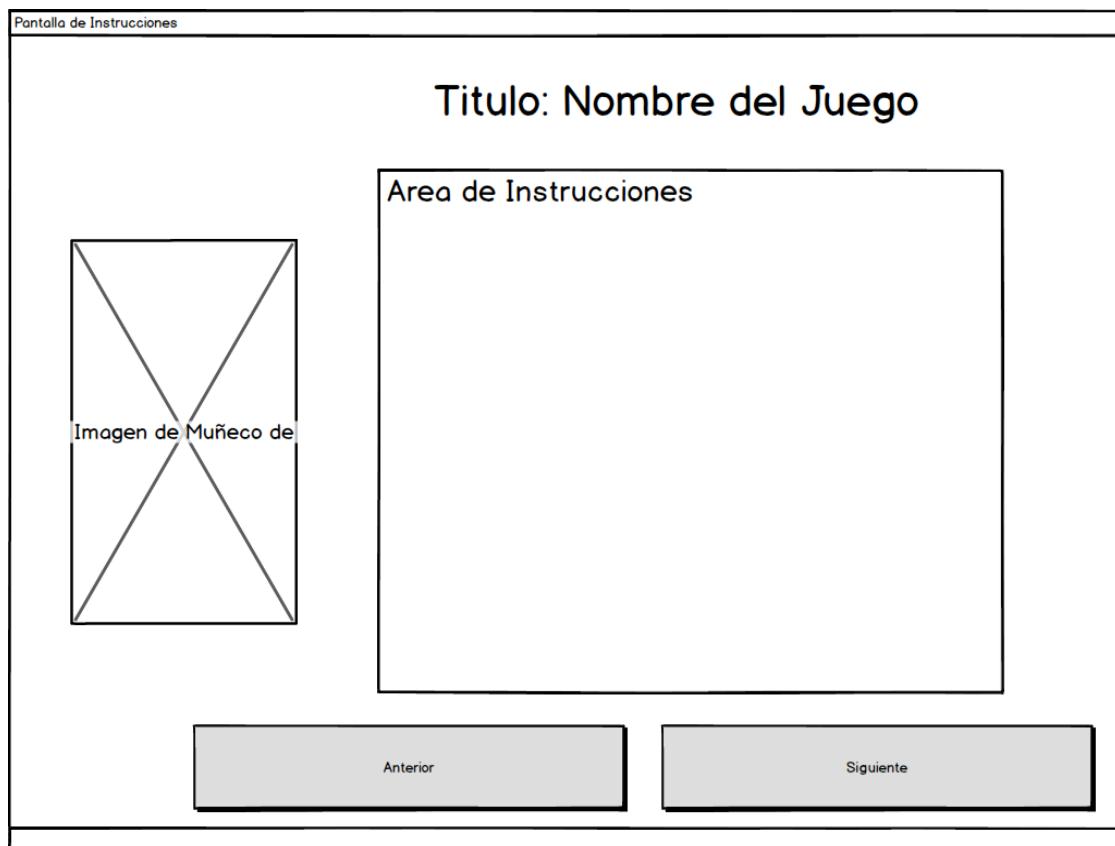


Figura 29. Prototipo de la Pantalla de Instrucciones del Videojuego

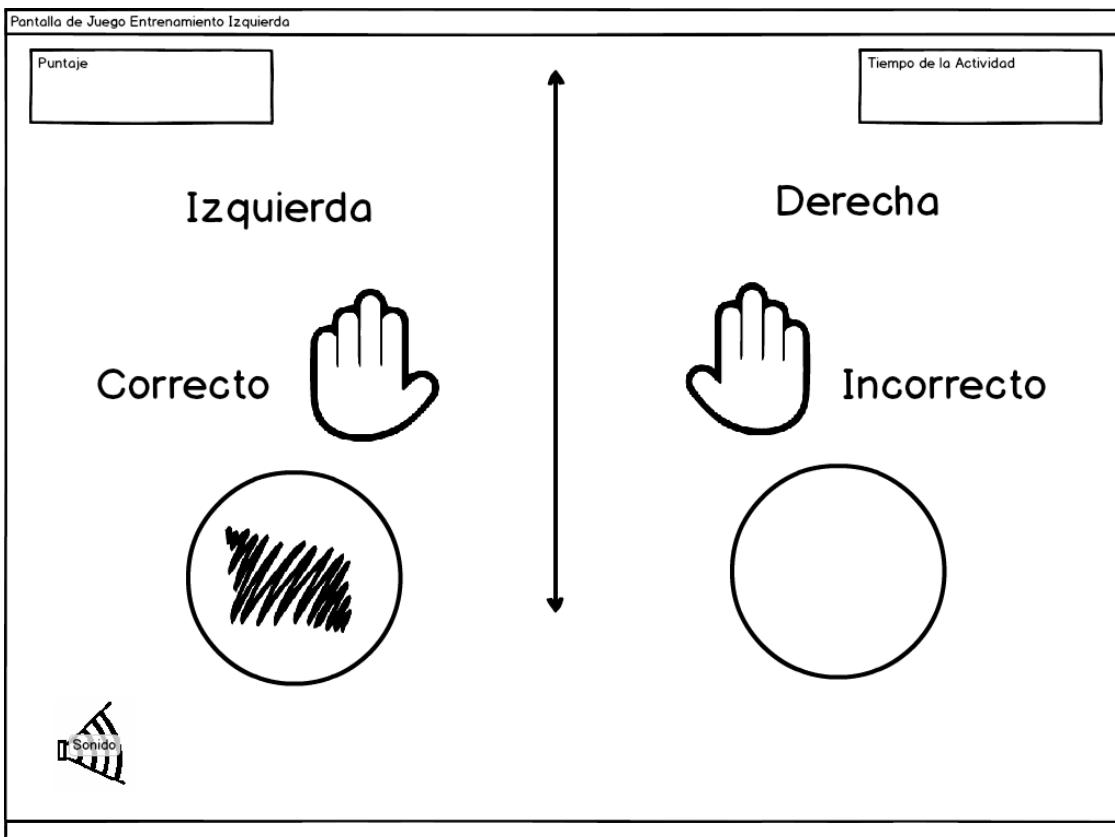


Figura 30. Prototipo del Videojuego de Entrenamiento. Actividad Izquierda

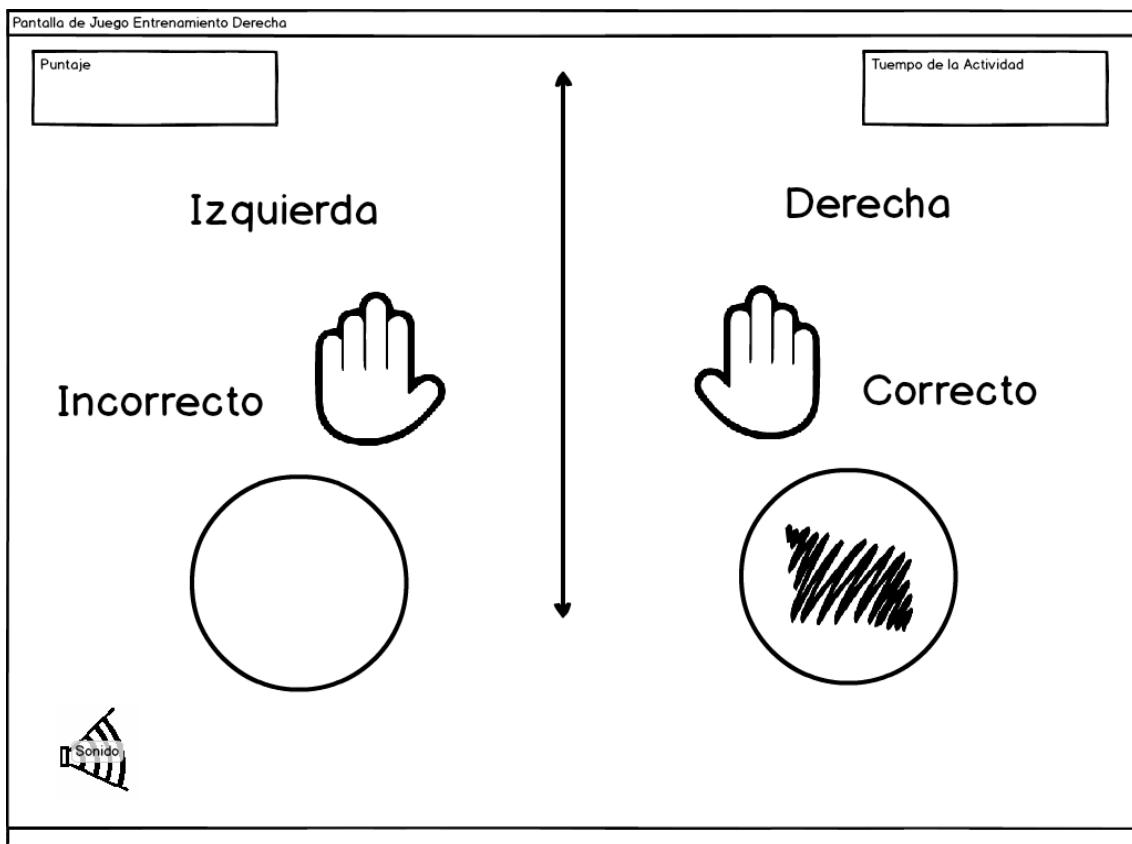


Figura 31. Prototipo del Videojuego de Entrenamiento. Actividad Derecha

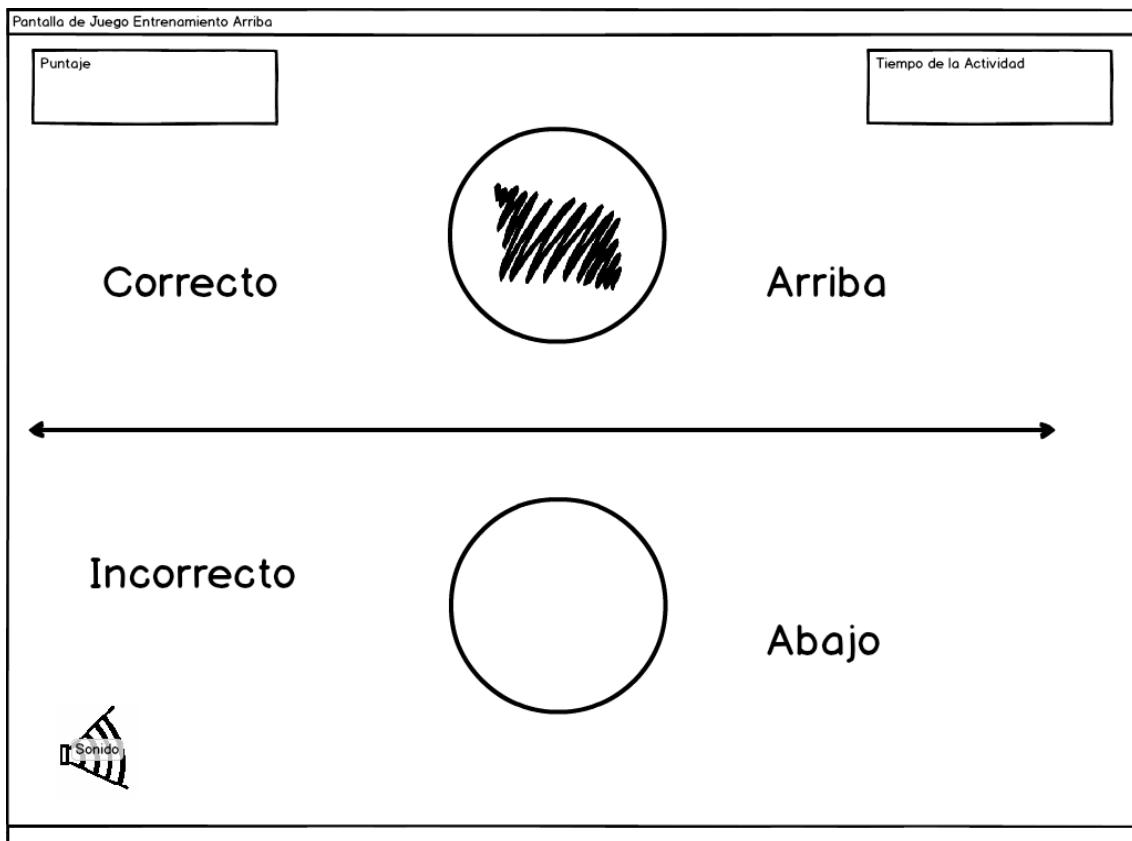


Figura 32. Prototipo del Videojuego de Entrenamiento. Actividad Arriba

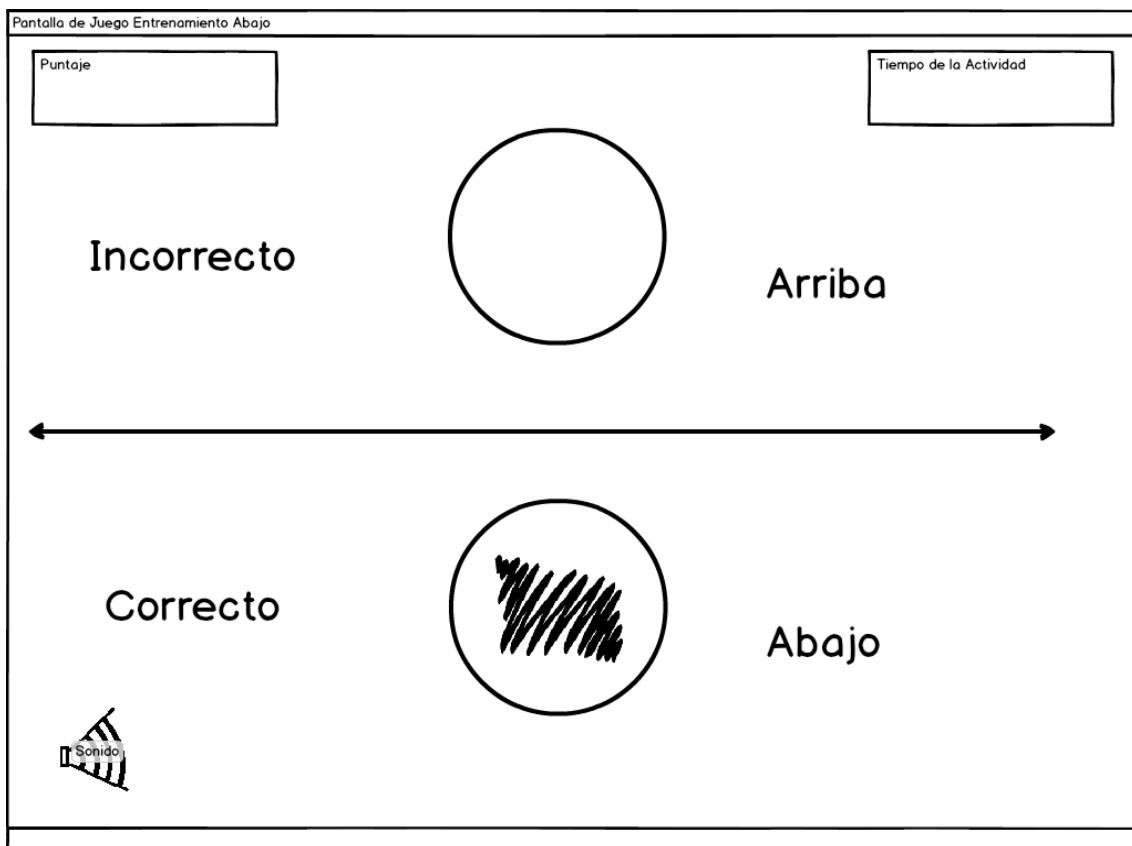


Figura 33. Prototipo del Videojuego de Entrenamiento. Actividad Abajo

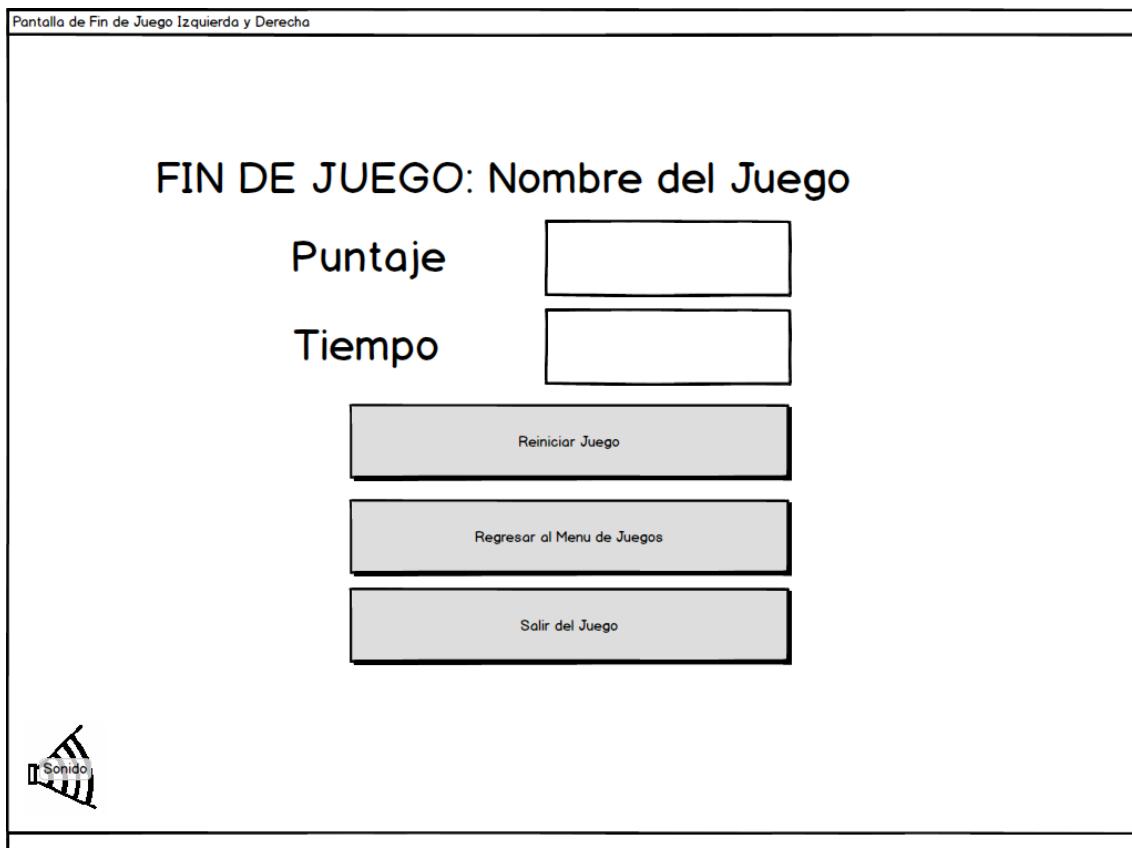


Figura 34. Prototipo de Pantalla de Fin de Juego

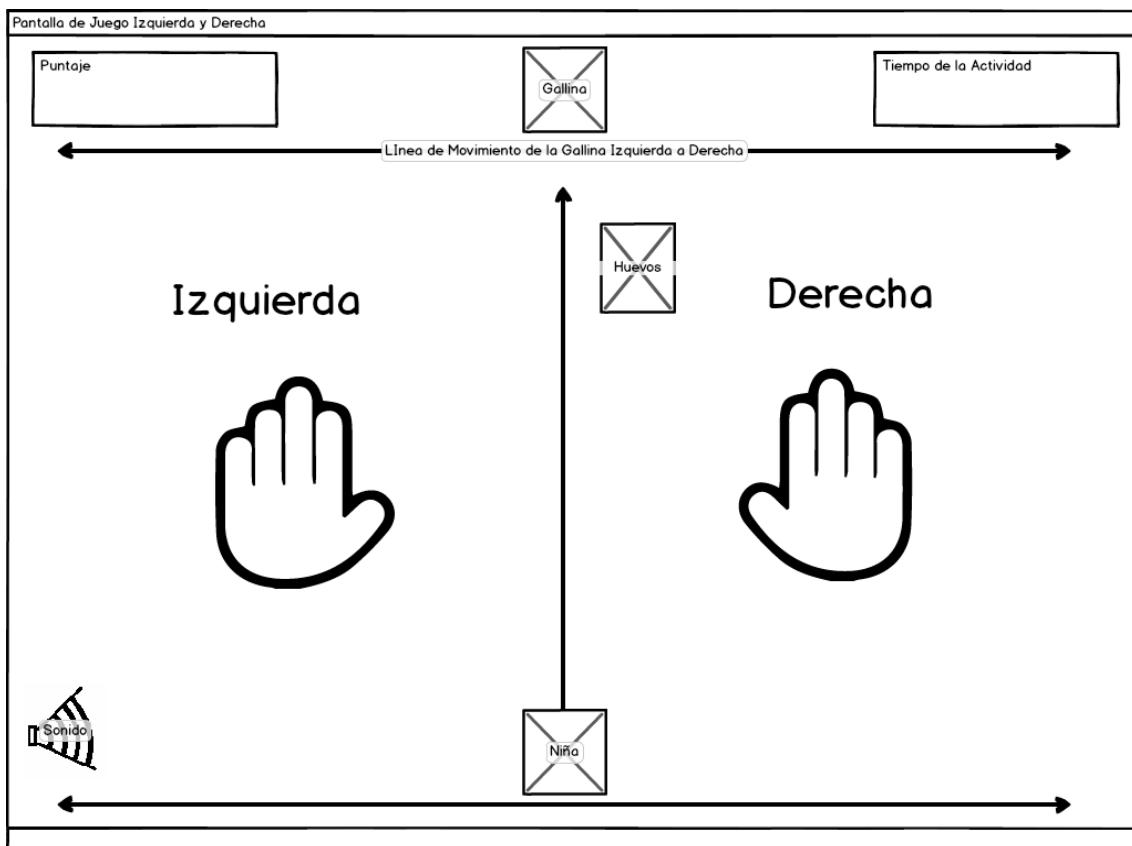


Figura 35. Prototipo del Videojuego Izquierda – Derecha

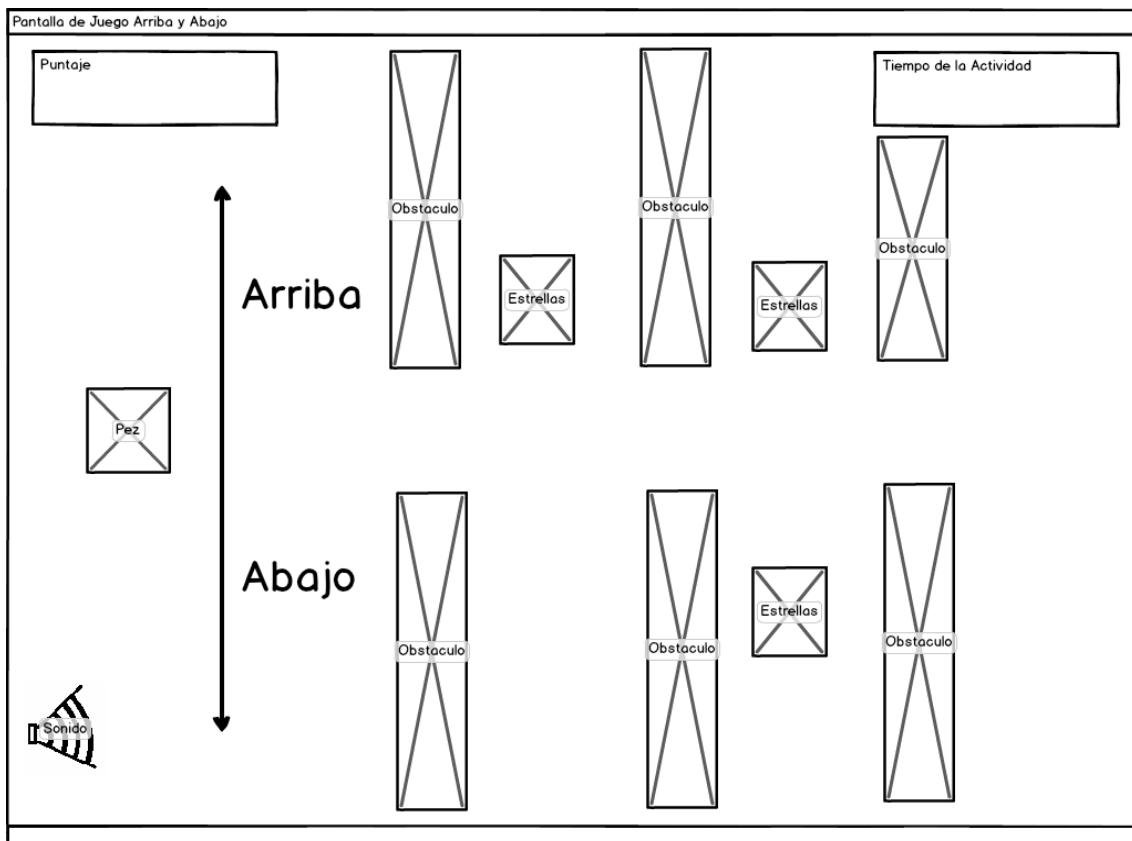


Figura 36. Prototipo del Videojuego Arriba – Abajo

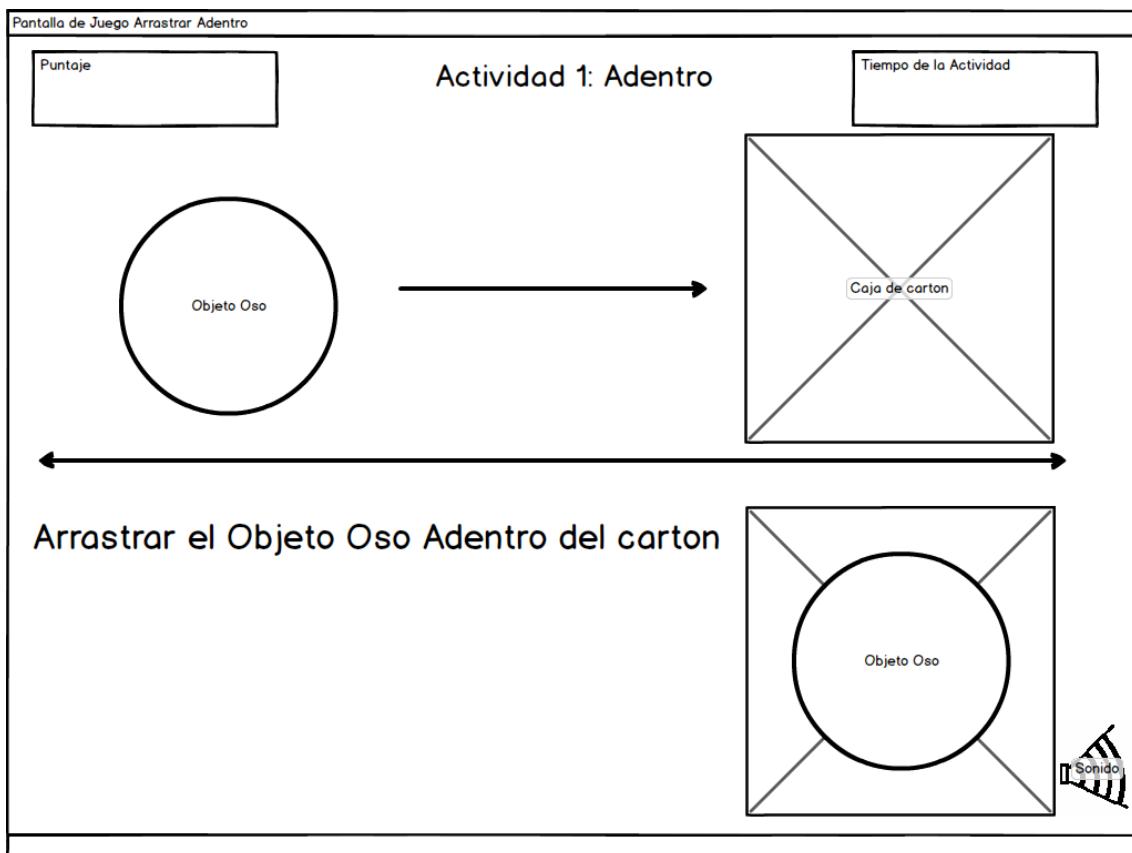


Figura 37. Prototipo del Videojuego Adentro - Afuera. Actividad Adentro

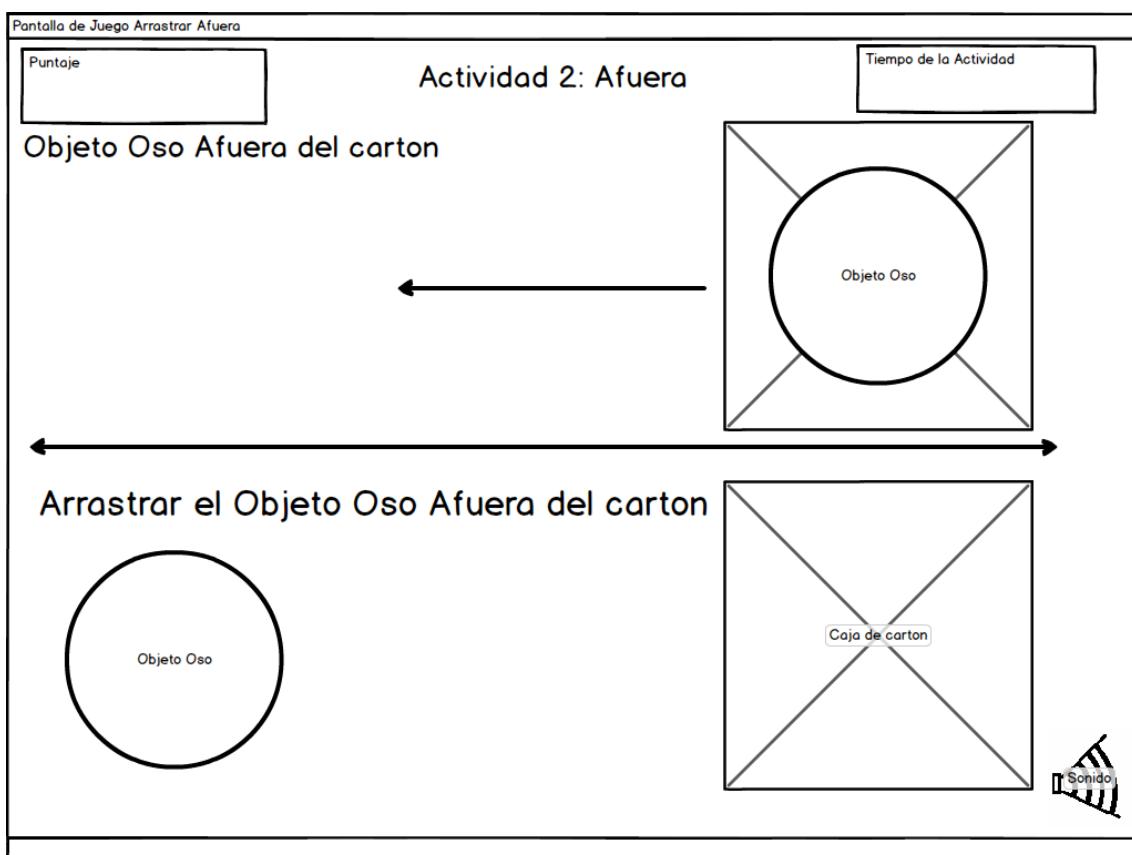


Figura 38. Prototipo del Videojuego Adentro - Afuera. Actividad Afuera

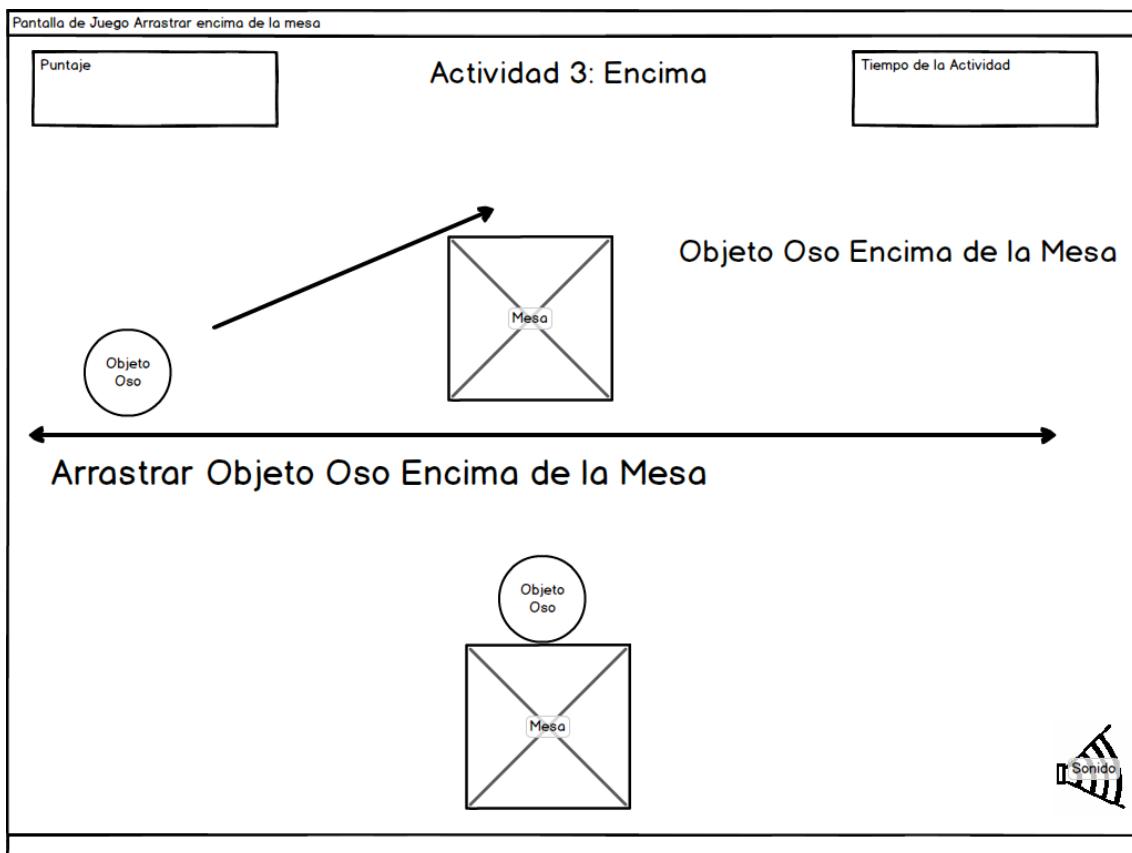


Figura 39. Prototipo del Videojuego Adentro - Afuera. Actividad Encima

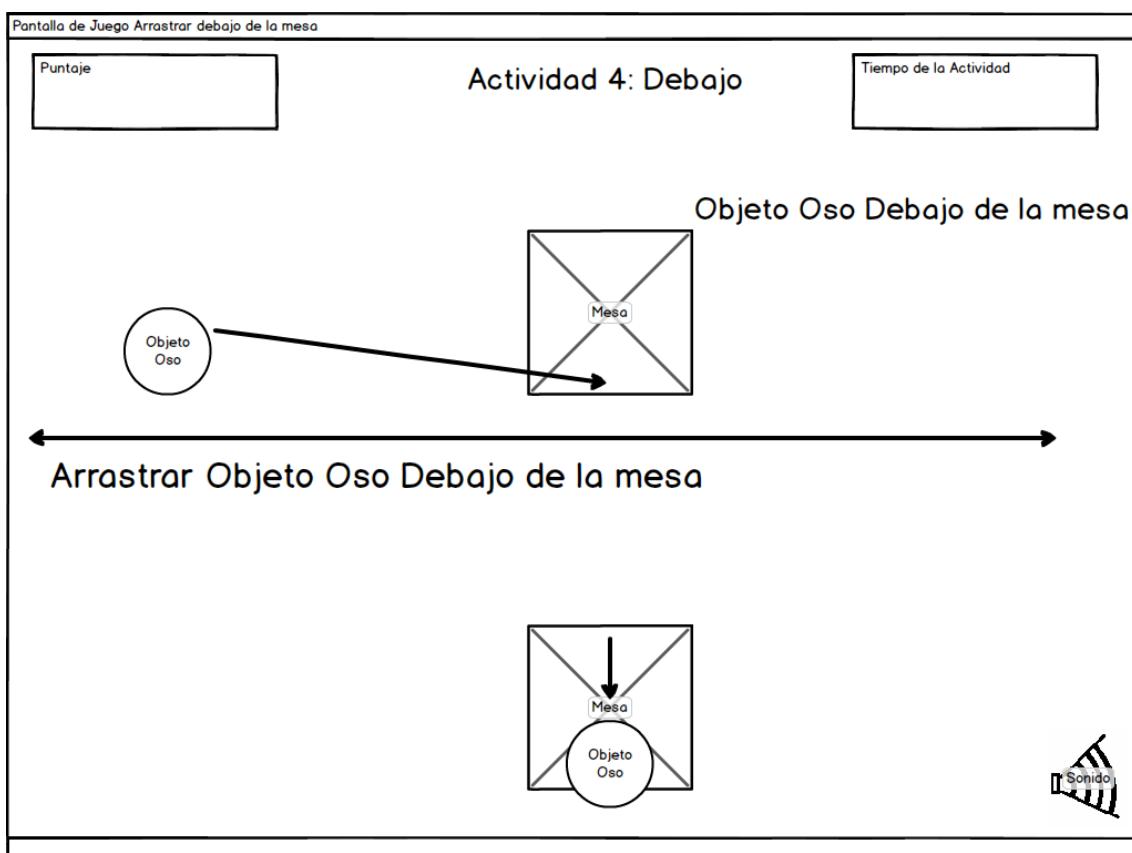


Figura 40. Prototipo del Videojuego Adentro - Afuera. Actividad Debajo

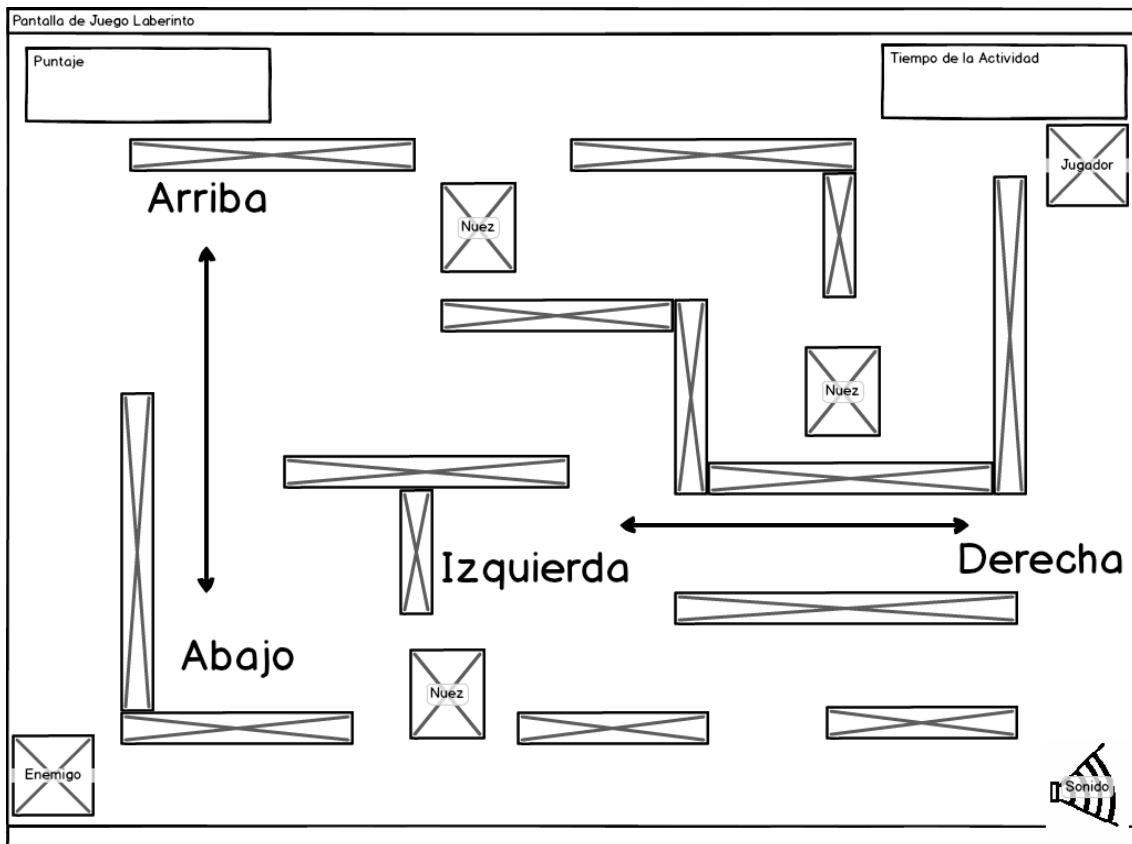


Figura 41. Prototipo del Videojuego Laberinto

4.4. Fase de Codificación

El videojuego de enseñanza de ubicación espacial “UBIC” se lo codifica utilizando las herramientas de software libre descritas en el apartado de restricciones de Software.

4.4.1. Estructura General del Videojuego UBIC

Para la implementación del videojuego “UBIC”, que usa el Microsoft Kinect, Python y OpenKinect es necesario instalar los prerequisitos descritos en el anexo B.

Para la simulación del movimiento del mouse y el evento del clic izquierdo por medio del OpenKinect, se usa el código en Python hecho por Alexander James Wallar, que realiza el reconocimiento de la mano (Hand Traking.py) este programa se ejecuta en GNU/Linux. (ActiveState Software Inc., 2018)

```

HandTracking.py
1  #-*- coding: utf-8 -*-
2  from freenect import sync_get_depth as get_depth #Utiliza freenect para obtener informacion detallada del Kinect
3  import numpy as np #Importar NumPy
4  import cv, cv2 #Utiliza tanto cv como cv2
5  import pygame #Utiliza pygome
6
7  #Las siguientes bibliotecas se usan para la manipulacion del mouse
8  from Xlib import X, display
9  import Xlib.XK
10 import Xlib.error
11 import Xlib.ext.xtest
12
13 constList = lambda length, val: [val for _ in range(length)] #Da una lista de longitud del tamaño lleno con la variable val. La longitud es una lista y val es dinamica
14
15 """
16 Esta clase es una forma menos extensa de regionprops () desarrollada por MATLAB. Encuentra propiedades de contornos y los establece en campos
17 """
18 class BlobAnalysis:
19     def __init__(self,BW): #Constructor. BW es una imagen binaria en forma de una matriz numpy
20         self.BW = BW
21         cs = cv.FindContours(cv.fromarray(self.BW.astype(np.uint8)),cv.CreateMemStorage(),mode = cv.CV_RETR_EXTERNAL) #Encuentra los contornos
22         counter = 0
23
24     Estas son listas dinamicas usadas para almacenar variables
25
26     centroid = list()
27     cHull = list()
28     contours = list()
29     cHullArea = list()
30     contourArea = list()
31     while cs: #Iterar a traves de CuSeg, cs.
32         if abs(cv.ContourArea(cs)) > 2000: #Filtra contornos de menos de 2000 pixeles en el area
33             contourArea.append(cv.ContourArea(cs)) #Se agrega contourArea con el area de contorno mas reciente

```

Figura 42. Código del Archivo HandTracking.py

La Figura 43 muestra al usuario que trabaja con el programa de handTracking.py. La figura amarilla es el cuerpo del usuario. Cuando la mano del usuario se acerca al Kinect más allá de cierto umbral, el programa detecta la mano automáticamente mostrada por la figura verde.

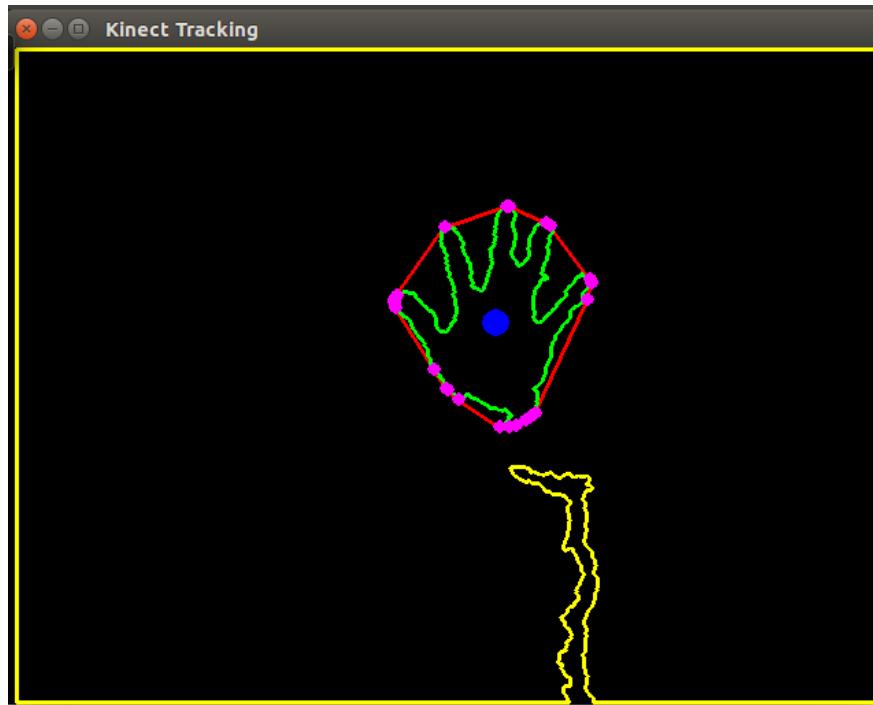


Figura 43. Propósito del Programa de HandTracking

El objeto 'Mano' está compuesto por los siguientes elementos:

- **Perímetro de la mano:** El perfil de la mano, corresponde a la figura verde de la imagen.
- **Casco convexo de la mano:** Representado con líneas rojas. Controla el clic movimiento.
- **Vértices casco convexo:** Como puntos rosados en la imagen.
- **Centroide del casco convexo:** Representado como un punto azul. Esto es lo que realmente controla el ratón.

La Figura 44 muestra el evento de hacer un clic, en el cual el usuario debe cerrar su mano.

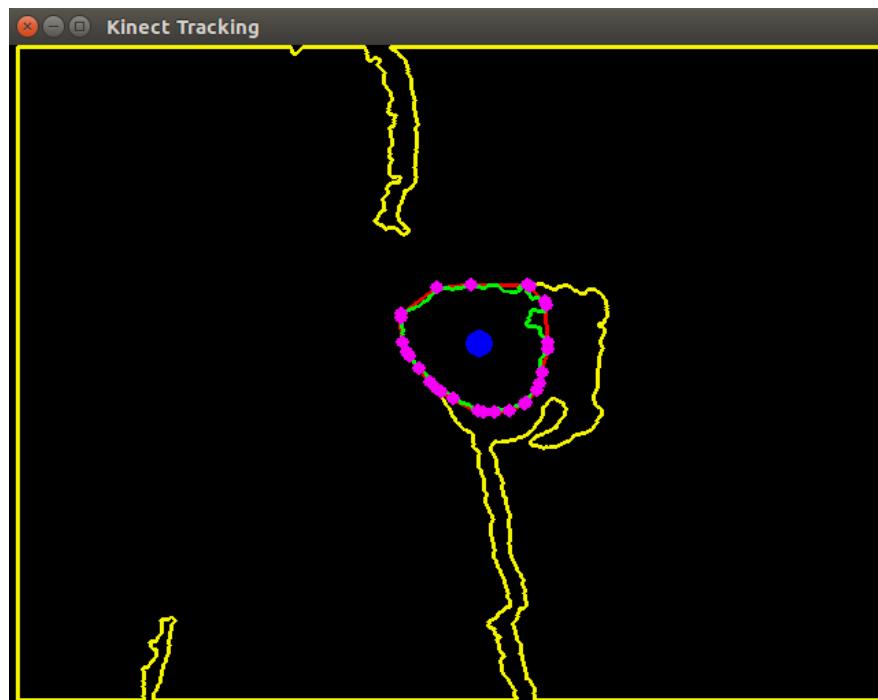


Figura 44. Mano Cerrada - Evento de Hacer Click

El programa detecta que la mano está cerrada por dos razones:

1. El área convexa del casco se vuelve más pequeña.
2. La proporción de espacio desperdiciado aumenta hasta casi al 100%.

Se llama "Relación de espacio desperdiciado " a la diferencia de la Área convexa del casco y Área de la mano.

$$\text{Relación de espacio desperdiciado} = \frac{\text{Área de la mano}}{\text{Área convexa del casco}}$$

If Relación de espacio desperdiciado ~ 60%

ManoAbierta = Verdadero

Else Relación de espacio desperdiciado ~ 90%

ManoCerrada = Verdadero

(Larriba, Martí y Riba, 2013)

Para la generación del Menú de opciones que aparecen en cada una de las pantallas del videojuego se utiliza el siguiente código de un proyecto llamado VPRZ Interactive display for kids on ZOO desarrollado por Lukas Svaty. (VRPZ, 2015)

```

kinectmenu.py

class IdleScreen():
    def __init__(self, screen):
        pygame.init()
        self.screen = screen
        self.scrWidth = self.screen.get_rect().width
        self.scrHeight = self.screen.get_rect().height
        self.bgColor = (0, 0, 0)
        self.bgImage = pygame.transform.flip(pygame.image.load("../graphics/mainbg.jpg").convert(), 1, 0)
        self.clock = pygame.time.Clock()
        self.font = pygame.font.SysFont("LDFComicSans", 40)
        self.fontColor = (255, 255, 255)
        self.menuItems = []
        self.itemNames = ("New game", "Quit")
        self.menuFuncs = { "New game" : self.startNewGame,
                           "Quit" : sys.exit}
        self.animalImgs = []
        self.animalPictures = ["bison.png", "elephant.png", "giraffe.png", "goat.png", "lion.png",
                               "monkey.png", "sheep.png"]
        self.activeFocus = 0
        self.lastActiveFocus = 1

    # Crea una matriz con etiquetas de menu listas para dibujar
    def buildMenu(self):
        self.items = []

        for index, item in enumerate(self.itemNames):
            label = pygame.transform.flip(self.font.render(item, 1, self.fontColor), 1, 0)
            width = label.get_rect().width
            height = label.get_rect().height + 30
            posx = (self.scrWidth / 2) - (width / 2)
            totalHeight = len(self.itemNames) * height
            posy = (self.scrHeight / 2) - (totalHeight / 2) + (index * height)

```

Figura 45. Código del Archivo KinectMenu.py

Para la realización de los videojuegos se utilizaron diversos manuales de pygame como por ejemplo el manual de desarrollo de videojuegos con Pygame de códigofacilito, desarrollo de videojuego de Laberinto de Tom y Jerry, etc.

El código se lo elabora con una clase principal llamada IdleScreen que es la que invoca a las otras clases involucradas en cada uno de los videojuegos por medio de las funciones

internas de esta clase. Para la creación de cada uno de los menús de las interfaces se las realiza con la ayuda de la clase MenuItem.

La clase BlobAnalysis es en la cual se encuentra integrado todo el código concerniente al reconocimiento de la mano, es la que interactúa directamente con el Dispositivo Kinect.

```
# Juego de La Galina y los Huevos (Derecha e Izquierda)
def JuegoDerIzq(self):
    global done
    global identidad
    screenloop = True
    (depth,_) = get_depth()
    # Lista de cache en blanco para el area convexa del casco
    HullAreaCache = constList(5,12000)
    # Lista de cache en blanco para la relacion de area del area de contorno al area de casco convexo
    areaRatioCache = constList(5,1)
    # Iniciar lista de centroides
    centroidList = list()
    screenFlipped = pygame.display.set_mode((self.scrWidth, self.scrHeight), pygame.FULLSCREEN)
    # Iterator boolean -> Indica a programa cuando finalizar
    # Muy importante bool para la manipulacion del raton
    dummy = False
    # Cargar sonido principal
    pygame.mixer.music.load('Sonidos/DonkeyKongCountry3-JangleBells.mp3')
    pygame.mixer.music.play(3)
    # Cargar La Palabra de Fin del Juego
    Texto = pygame.transform.flip(self.font.render("Fin del Juego", 1, (255, 14, 0)), 1, 0)
    # Instancia del Objeto Nave Espacial
    jugador = Jugador.NenaCanasta(self.scrWidth,self.scrHeight)
    # Instancia del objeto Invasor
    #enemigo = Invasor(100,100)
    self.cargarEnemigos()
    # Instancia del Objeto Puntaje
    puntos = Puntaje.Score(self.fontPuntaje, (900, 50))
    # Instancia del Objeto Proyectil para el Jugador
    # DemoProyectil = Proyectil(self.scrWidth/2,self.scrHeight-80,"../Imagenes/bala.png", True)
    # Instancia del Objeto Proyectil para el enemigo
    # ProyectilInvasor = Proyectil(self.scrWidth/4,self.scrHeight-700,"../Imagenes/disparob.jpg", False)
    # Verificar si un jugador gano o perdio
```

Figura 46. Código del Archivo main.py

4.5. Fase de Pruebas

El videojuego al igual que otros proyectos de software pasa por una etapa donde se corrigen los errores encontrados durante el proceso de programación asegurando así su funcionalidad. La diferencia radica entre otros tipos de software y los videojuegos en que estos requieren que se refine una característica fundamental que es la de producir diversión de manera efectiva (Jugabilidad).

Para llevar a cabo este propósito se divide en tres fases: pruebas físicas, pruebas alpha y pruebas beta.

4.5.1. Pruebas Físicas

Para el desarrollo de estas pruebas utilizamos los prototipos, simulando los eventos que suceden en el videojuego, dando como resultado un modelo que ayude a prevenir problemas de programación, así como de las características de diseño.

4.5.2. Pruebas Alpha

Estas pruebas se realizan cuando ya contamos con un producto terminado, su propósito es corregir defectos y mejorar la jugabilidad. En la tabla 47 se detallan los ítems de prueba.

Tabla 47. Ítems de Prueba

Código de Requerimiento	Nombre	Tipo
RF-01	Creación del Sistema de Seguimiento de la Mano.	Funcional.
RF-02	Ingreso a la Pantalla Principal de la Aplicación.	Funcional
RF-03	Ingreso al Menú de Videojuegos de la Aplicación.	Funcional
RF-04	Pantalla de Instrucciones del Videojuego.	Funcional
RF-05	Ingreso al Videojuego Izquierda – Derecha.	Funcional
RF-06	Pantalla de Fin de Juego.	Funcional
RF-07	Ingreso al Videojuego Arriba – Abajo.	Funcional
RF-08	Ingreso al Videojuego Laberinto.	Funcional
RF-09	Actividad del Videojuego de Entrenamiento Izquierda.	Funcional
RF-10	Actividad del Videojuego de Entrenamiento Derecha.	Funcional
RF-11	Actividad del Videojuego de Entrenamiento Arriba.	Funcional
RF-12	Actividad del Videojuego de Entrenamiento Abajo.	Funcional
RF-13	Ingreso al Videojuego Adentro – Afuera.	Funcional
RNF-01	Cumplir con las Métricas de Usabilidad.	No Funcional

4.5.2.1. Diseño de los Casos de Prueba

Tabla 48. Caso de Prueba RF-01

RF-01	Creación del Sistema de Seguimiento de la Mano.			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Kinect conectado.	Ingresar al software.	Ingresar al software.	Satisfactoria
2	Kinect no conectado.	Mensaje de error "No conectado".	Mensaje de error "No conectado".	Satisfactoria
3	Cámara reconoce la mano.	Reconoce la mano.	Reconoce la mano.	Satisfactoria

4	Mano abierta.	Mueve el cursor del mouse.	Mueve el cursor del mouse.	Satisfactoria
5	Mano cerrada.	Ejecuta clic del mouse.	Ejecuta clic del mouse.	Satisfactoria

Tabla 49. Caso de Prueba RF-02

Ingreso a la Pantalla Principal de la Aplicación.				
RF-02	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Clic en entrar.	Ingresar al menú de videojuegos.	Ingresar al menú de videojuegos.	Satisfactoria
3	Clic en salir.	Salir del videojuego.	Salir del videojuego.	Satisfactoria
4	Tecla ESC.	Salir del videojuego.	Salir del videojuego.	Satisfactoria

Tabla 50. Caso de Prueba RF-03

Ingreso al Menú de Videojuegos de la Aplicación.				
RF-03	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Clic en Entrenamiento.	Pantalla de instrucciones del videojuego de entrenamiento.	Pantalla de instrucciones del videojuego de entrenamiento.	Satisfactoria
3	Clic en Izquierda – Derecha.	Pantalla de instrucciones del videojuego Izquierda – Derecha.	Pantalla de instrucciones del videojuego Izquierda – Derecha.	Satisfactoria
4	Clic en Arriba – Abajo.	Pantalla de instrucciones del videojuego Arriba – Abajo.	Pantalla de instrucciones del videojuego Arriba – Abajo.	Satisfactoria
5	Clic en Adentro – Afuera.	Pantalla de instrucciones del videojuego Adentro – Afuera.	Pantalla de instrucciones del videojuego Adentro – Afuera.	Satisfactoria
6	Clic en Laberinto.	Pantalla de instrucciones del videojuego Laberinto.	Pantalla de instrucciones del videojuego Laberinto.	Satisfactoria
7	Tecla ESC.	Salir del videojuego.	Salir del videojuego.	Satisfactoria

Tabla 51. Caso de Prueba RF-04

Pantalla de Instrucciones del Videojuego				
RF-04	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Número de videojuego (1= Entrenamiento, 2= Izquierda-Derecha, 3=Arriba-Abajo,	Instrucción de cada videojuego.	Instrucción de cada videojuego.	Satisfactoria

	4=Adentro-Afuera, 5=Laberinto).			
3	Clic en Anterior.	Pantalla del menú de los videojuegos.	Pantalla del menú de los videojuegos.	Satisfactoria
4	Clic en Siguiente, número videojuego = 1	Pantalla videojuego Entrenamiento.	Pantalla videojuego Entrenamiento.	Satisfactoria
5	Clic en Siguiente, número videojuego = 2	Pantalla videojuego Izquierda - Derecha	Pantalla videojuego Izquierda – Derecha.	Satisfactoria
6	Clic en Siguiente, número videojuego = 3	Pantalla videojuego Arriba – Abajo.	Pantalla videojuego Arriba – Abajo.	Satisfactoria
7	Clic en Siguiente, número videojuego = 4	Pantalla videojuego Adentro – Afuera.	Pantalla videojuego Adentro – Afuera.	Satisfactoria
8	Clic en Siguiente, número videojuego = 5	Pantalla videojuego Laberinto.	Pantalla videojuego Laberinto.	Satisfactoria
9	Tecla ESC.	Salir del videojuego.	Salir del videojuego.	Satisfactoria

Tabla 52. Caso de Prueba RF-05

RF-05	Ingreso al Videojuego Izquierda – Derecha.			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Personaje del jugador.	Se muestra en pantalla.	Se muestra en pantalla.	Satisfactoria
3	Personaje del enemigo.	Se muestra en pantalla.	Se muestra en pantalla.	Satisfactoria
4	Clic en la flecha con dirección a la izquierda.	Se mueve el personaje del jugador a la izquierda.	Se mueve el personaje del jugador a la izquierda.	Satisfactoria
5	Clic en la flecha con dirección a la derecha.	Se mueve el personaje del jugador a la derecha.	Se mueve el personaje del jugador a la derecha.	Satisfactoria.
6	Personaje del jugador.	Se mantenga en los límites horizontales de la pantalla.	Se mantenga en los límites horizontales de la pantalla.	Satisfactoria
7	Personaje del enemigo.	Se mantenga en los límites horizontales de la pantalla.	Se mantenga en los límites horizontales de la pantalla.	Satisfactoria
8	Personaje del enemigo.	Se mueva de izquierda a derecha automáticamente.	Se mueva de izquierda a derecha automáticamente.	Satisfactoria
9	Huevos de la gallina.	Arroje los huevos cada 2 segundos.	Arroje los huevos cada 2 segundos.	Satisfactoria
10	Colisión personaje jugador con huevos.	Recoja los huevos.	Recoja los huevos.	Satisfactoria
11	Marcador.	Muestra el marcador.	Muestra el marcador.	Satisfactoria
12	Un punto por huevo.	Actualiza el marcador.	Actualiza el marcador.	Satisfactoria
13	Temporizador.	Muestra el tiempo.	Muestra el tiempo.	Satisfactoria
14	Huevo pasa el límite vertical de la pantalla.	Fin del videojuego.	Fin del videojuego.	Satisfactoria
15	Puntaje total, tiempo y número del videojuego.	Se envía a la pantalla de fin de juego.	Se envía a la pantalla de fin de juego.	Satisfactoria
16	Tecla ESC.	Salir del videojuego.	Salir del videojuego.	Satisfactoria
17	Tecla P.	Pausa el videojuego durante cinco segundos.	Pausa el videojuego durante cinco segundos.	Satisfactoria

18	Tecla Back Space.	Regresa al menú de los videojuegos.	Regresa al menú de los videojuegos.	Satisfactoria
19	Barra Espaciadora.	Desactiva el sonido del videojuego.	Desactiva el sonido del videojuego.	Satisfactoria
20	Tecla V.	Activa el sonido del videojuego.	Activa el sonido del videojuego.	Satisfactoria

Tabla 53. Caso de Prueba RF-06

RF-06	Pantalla de Fin de juego			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Número del videojuego.	Título y nombre del videojuego.	Título y nombre del videojuego.	Satisfactoria
3	Puntaje total.	Muestra el puntaje.	Muestra el puntaje.	Satisfactoria
4	Tiempo total.	Muestra el tiempo.	Muestra el tiempo.	Satisfactoria
5	Si número de videojuego = 1 que equivale al videojuego Entrenamiento.	Se muestra el puntaje y el tiempo de cada actividad del videojuego de entrenamiento.	Se muestra el puntaje y el tiempo de cada actividad del videojuego de entrenamiento.	Satisfactoria
6	Si número de videojuego = 4 que equivale al videojuego Adentro - Afuera.	Se muestra el puntaje y el tiempo de cada actividad del videojuego Adentro - Afuera.	Se muestra el puntaje y el tiempo de cada actividad del videojuego Adentro - Afuera.	Satisfactoria
7	Si número de videojuego = 2,3 o 5 que equivalen a los videojuegos de Izquierda – Derecha, Arriba – Abajo y Laberinto respectivamente.	Se muestra el puntaje y el tiempo de cada uno de los videojuegos dependiendo el número.	Se muestra el puntaje y el tiempo de cada uno de los videojuegos dependiendo el número.	Satisfactoria
8	Clic en Reiniciar Juego.	Reinicia el videojuego.	Reinicia el videojuego.	Satisfactoria
9	Clic en Menú Juegos.	Despliega la pantalla del menú de los videojuegos.	Despliega la pantalla del menú de los videojuegos.	Satisfactoria
10	Clic en Salir	Salir del videojuego.	Salir del videojuego.	Satisfactoria
11	Tecla ESC	Salir del videojuego.	Salir del videojuego.	Satisfactoria

Tabla 54. Caso de Prueba RF-07

RF-07	Ingreso al Videojuego Arriba - Abajo			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Personaje del jugador.	Se muestre en pantalla.	Se muestre en pantalla.	Satisfactoria

3	Personaje del enemigo.	Se muestre de forma alternada en la parte inferior y superior de la pantalla.	Se muestre de forma alternada en la parte inferior y superior de la pantalla.	Satisfactoria
4	Clic en la flecha con dirección para arriba.	Se mueve el personaje del jugador para arriba.	Se mueve el personaje del jugador para arriba.	Satisfactoria
5	Clic en la flecha con dirección para abajo.	Se mueve el personaje del jugador para abajo.	Se mueve el personaje del jugador para abajo.	Satisfactoria
6	Personaje del jugador.	Se mantenga en los límites verticales de la pantalla.	Se mantenga en los límites verticales de la pantalla.	Satisfactoria
7	Personaje del enemigo	Sobrepasa los límites horizontales de la pantalla.	Sobrepasa los límites horizontales de la pantalla.	Satisfactoria
8	Personaje del enemigo.	Se mueva de derecha a izquierda automáticamente.	Se mueva de derecha a izquierda automáticamente.	Satisfactoria
9	Estrellas de mar.	Aparecen en columnas de 5 estrellas en medio del personaje enemigo.	Aparecen en columnas de 5 estrellas en medio del personaje enemigo.	Satisfactoria
10	Colisión personaje jugador con cada estrella.	Recoja las estrellas.	Recoja las estrellas.	Satisfactoria
11	Marcador.	Muestre el marcador.	Muestre el marcador.	Satisfactoria
12	Un punto por estrella.	Actualice el marcador.	Actualice el marcador.	Satisfactoria
13	Tiempo.	Muestre el tiempo.	Muestre el tiempo.	Satisfactoria
14	Colisión del personaje jugador con el enemigo.	Fin del videojuego.	Fin del videojuego.	Satisfactoria
15	Puntaje total, tiempo y número del videojuego.	Se envía a la pantalla de fin de juego.	Se envía a la pantalla de fin de juego.	Satisfactoria
16	Tecla ESC.	Salir del Videojuego	Salir del Videojuego	Satisfactoria
17	Tecla P.	Pausa el videojuego durante cinco segundos.	Pausa el videojuego durante cinco segundos.	Satisfactoria
18	Tecla Back Space.	Regresa al menú de los videojuegos.	Regresa al menú de los videojuegos.	Satisfactoria
19	Barra Espaciadora	Desactiva el sonido del videojuego.	Desactiva el sonido del videojuego.	Satisfactoria
20	Tecla V	Activa el sonido del videojuego.	Activa el sonido del videojuego.	Satisfactoria

Tabla 55. Caso de Prueba RF-08

RF-08	Ingreso al Videojuego Laberinto			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Escribir los caracteres del mapa.	Revisar en mapa.txt que se cree el laberinto y las nueces.	Revisar en mapa.txt que se cree el laberinto y las nueces.	Satisfactoria
3	Objeto pared.	Clase creada.	Clase creada.	Satisfactoria

4	Objeto Nuez	Clase creada.	Clase creada.	Satisfactoria
5	Objeto Mapa	Clase creada.	Clase creada.	Satisfactoria
6	mapa.txt (Objetos: Pared, Nuez, Mapa)	Creación del mapa y nueces en la pantalla.	Creación del mapa y nueces en la pantalla.	Satisfactoria
7	Personaje del jugador.	Se muestre en la parte superior derecha de la pantalla.	Se muestre en la parte superior derecha de la pantalla.	Satisfactoria
8	Personaje del enemigo.	Se muestre en la parte inferior izquierda de la pantalla.	Se muestre en la parte inferior izquierda de la pantalla.	Satisfactoria
9	Clic en la flecha con dirección para arriba.	Se mueve el personaje del jugador para arriba dentro del laberinto.	Se mueve el personaje del jugador para arriba dentro del laberinto.	Satisfactoria
10	Clic en la flecha con dirección para abajo.	Se mueve el personaje del jugador para abajo dentro del laberinto.	Se mueve el personaje del jugador para abajo dentro del laberinto.	Satisfactoria
11	Clic en la flecha con dirección a la izquierda.	Se mueve el personaje del jugador a la izquierda dentro del laberinto.	Se mueve el personaje del jugador a la izquierda dentro del laberinto.	Satisfactoria
12	Clic en la flecha con dirección a la derecha.	Se mueve el personaje del jugador a la derecha dentro del laberinto.	Se mueve el personaje del jugador a la derecha dentro del laberinto.	Satisfactoria
13	Personaje del jugador.	Se mantenga en los límites verticales y horizontales de la pantalla.	Se mantenga en los límites verticales y horizontales de la pantalla.	Satisfactoria
14	Colisión personaje jugador con cada nuez.	Recoja las nueces.	Recoja las nueces.	Satisfactoria
15	Marcador.	Muestre el marcador.	Muestre el marcador.	Satisfactoria
16	Un punto por nuez.	Actualice el marcador.	Actualice el marcador.	Satisfactoria
17	Tiempo.	Muestre el tiempo.	Muestre el tiempo.	Satisfactoria
18	Colisión del personaje jugador con el enemigo.	Fin del videojuego.	Fin del videojuego.	Satisfactoria
19	Puntaje total, tiempo y número del videojuego.	Se envía a la pantalla de fin de juego.	Se envía a la pantalla de fin de juego.	Satisfactoria
20	Tecla ESC.	Salir del Videojuego	Salir del Videojuego	Satisfactoria
21	Tecla P.	Pausa el videojuego durante cinco segundos.	Pausa el videojuego durante cinco segundos.	Satisfactoria
22	Tecla Back Space.	Regresa al menú de los videojuegos.	Regresa al menú de los videojuegos.	Satisfactoria
23	Barra Espaciadora	Desactiva el sonido del videojuego.	Desactiva el sonido del videojuego.	Satisfactoria
24	Tecla V	Activa el sonido del videojuego.	Activa el sonido del videojuego.	Satisfactoria

Tabla 56. Caso de Prueba RF-09

RF-09	Actividad del Videojuego de Entrenamiento Izquierda			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Fondo de la pantalla.	En el lado izquierdo de la pantalla se dibuja un círculo con relleno y en el lado derecho un círculo sin con relleno.	En el lado izquierdo de la pantalla se dibuja un círculo con relleno y en el lado derecho un círculo sin con relleno.	Satisfactoria
3	Menú izquierda y derecha.	Muestra en pantalla el menú izquierda y derecha.	Muestra en pantalla el menú izquierda y derecha.	Satisfactoria
4	Temporizador.	Muestra el temporizador en el lado superior derecho de la pantalla.	Muestra el temporizador en el lado superior izquierdo de la pantalla.	Satisfactoria
5	Puntaje.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Satisfactoria
6	Clic en el lado izquierdo de la pantalla.	Muestra una imagen de carita feliz en la pantalla.	Muestra una imagen de carita feliz en la pantalla.	Satisfactoria
7	Clic en el lado derecho de la pantalla.	Muestra una imagen de carita triste en la pantalla.	Muestra una imagen de carita triste en la pantalla.	Satisfactoria
8	Cada vez que se da clic en el lado izquierdo de la pantalla.	Actualiza el marcador sumando un punto por cada clic.	Actualiza el marcador sumando un punto por cada clic.	Satisfactoria
9	Temporizador = 15 segundos	Pasa a la actividad derecha.	Pasa a la actividad derecha.	Satisfactoria
10	Puntaje total de la actividad.	Se almacena en el vector listapuntos en la posición 0.	Se almacena en el vector listapuntos en la posición 0.	Satisfactoria
11	Tecla ESC	Salir del videojuego.	Salir del videojuego.	Satisfactoria
12	Tecla Back Space.	Regresa al menú de los videojuegos.	Regresa al menú de los videojuegos.	Satisfactoria

Tabla 57. Caso de Prueba RF-10

RF-10	Actividad del Videojuego de Entrenamiento Derecha			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Fondo de la pantalla	En el lado derecho de la pantalla se dibuja un círculo con relleno y en el lado izquierdo un círculo sin relleno.	En el lado derecho de la pantalla se dibuja un círculo con relleno y en el lado izquierdo un círculo sin relleno.	Satisfactoria
3	Menú izquierda y derecha.	Muestra en pantalla el menú izquierda y derecha.	Muestra en pantalla el menú izquierda y derecha.	Satisfactoria
4	Temporizador.	Muestra el temporizador en el lado superior derecho de la pantalla.	Muestra el temporizador en el lado superior izquierdo de la pantalla.	Satisfactoria
5	Puntaje.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Satisfactoria

6	Clic en el lado izquierdo de la pantalla.	Muestra una imagen de carita triste en la pantalla.	Muestra una imagen de carita triste en la pantalla.	Satisfactoria
7	Clic en el lado derecho de la pantalla.	Muestra una imagen de carita feliz en la pantalla.	Muestra una imagen de carita feliz en la pantalla.	Satisfactoria
8	Cada vez que se da clic en el lado derecho de la pantalla.	Actualiza el marcador sumando un punto por cada clic.	Actualiza el marcador sumando un punto por cada clic.	Satisfactoria
9	Temporizador = 15 segundos.	Pasa a la actividad arriba.	Pasa a la actividad arriba.	Satisfactoria
10	Puntaje total de la actividad.	Se almacena en el vector listapuntos en la posición 1.	Se almacena en el vector listapuntos en la posición 1.	Satisfactoria
11	Tecla ESC.	Salir del Juego.	Salir del Juego.	Satisfactoria
12	Tecla Back Space.	Regresa al menú de los videojuegos.	Regresa al menú de los videojuegos.	Satisfactoria

Tabla 58. Caso de Prueba RF-11

Actividad del Videojuego de Entrenamiento Arriba					
RF-11	Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Fondo de la pantalla.	En el lado de arriba de la pantalla se dibuja un círculo con relleno y en el lado de abajo un círculo sin relleno.	En el lado de arriba de la pantalla se dibuja un círculo con relleno y en el lado de abajo un círculo sin relleno.	En el lado de arriba de la pantalla se dibuja un círculo con relleno y en el lado de abajo un círculo sin relleno.	Satisfactoria
3	Menú arriba y abajo.	Muestra en pantalla el menú arriba y abajo.	Muestra en pantalla el menú arriba y abajo.	Muestra en pantalla el menú arriba y abajo.	Satisfactoria
4	Temporizador.	Muestra el temporizador en el lado superior derecho de la pantalla.	Muestra el temporizador en el lado superior derecho de la pantalla.	Muestra el temporizador en el lado superior derecho de la pantalla.	Satisfactoria
5	Puntaje.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Satisfactoria
6	Clic en la parte de arriba de la pantalla.	Muestra una imagen de carita feliz en la pantalla.	Muestra una imagen de carita feliz en la pantalla.	Muestra una imagen de carita feliz en la pantalla.	Satisfactoria
7	Clic en la parte de abajo de la pantalla.	Muestra una imagen de carita triste en la pantalla.	Muestra una imagen de carita triste en la pantalla.	Muestra una imagen de carita triste en la pantalla.	Satisfactoria
8	Cada vez que se da clic en la parte de arriba de la pantalla.	Actualiza el marcador sumando un punto por cada clic.	Actualiza el marcador sumando un punto por cada clic.	Actualiza el marcador sumando un punto por cada clic.	Satisfactoria
9	Temporizador = 15 segundos.	Pasa a la actividad abajo.	Pasa a la actividad abajo.	Pasa a la actividad abajo.	Satisfactoria
10	Puntaje total de la actividad.	Se almacena en el vector listapuntos en la posición 2.	Se almacena en el vector listapuntos en la posición 2.	Se almacena en el vector listapuntos en la posición 2.	Satisfactoria
11	Tecla ESC.	Salir del Juego.	Salir del Juego.	Salir del Juego.	Satisfactoria
12	Tecla Back Space.	Regresa al menú de los videojuegos.	Regresa al menú de los videojuegos.	Regresa al menú de los videojuegos.	Satisfactoria

Tabla 59. Caso de Prueba RF-12

RF-12	Actividad del Videojuego de Entrenamiento Abajo			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Fondo de la pantalla.	En el lado de arriba de la pantalla se dibuja un círculo sin relleno y en el lado de abajo un círculo con relleno.	En el lado de arriba de la pantalla se dibuja un círculo sin relleno y en el lado de abajo un círculo con relleno.	Satisfactoria
3	Menú arriba y abajo.	Muestra en pantalla el menú arriba y abajo.	Muestra en pantalla el menú arriba y abajo.	Satisfactoria
4	Temporizador.	Muestra el temporizador en el lado superior derecho de la pantalla.	Muestra el temporizador en el lado superior derecho de la pantalla.	Satisfactoria
5	Puntaje.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Satisfactoria
6	Clic en la parte de arriba de la pantalla.	Muestra una imagen de carita triste en la pantalla.	Muestra una imagen de carita triste en la pantalla.	Satisfactoria
7	Clic en la parte de abajo de la pantalla.	Muestra una imagen de carita feliz en la pantalla.	Muestra una imagen de carita feliz en la pantalla.	Satisfactoria
8	Cada vez que se da clic en la parte de abajo de la pantalla.	Actualiza el marcador sumando un punto por cada clic.	Actualiza el marcador sumando un punto por cada clic.	Satisfactoria
9	Temporizador = 15 segundos.	Pasa a la pantalla de fin de juego.	Pasa a la pantalla de fin de juego.	Satisfactoria
10	Puntaje total de la actividad.	Se almacena en el vector listapuntos en la posición 3.	Se almacena en el vector listapuntos en la posición 3.	Satisfactoria
11	Tecla ESC.	Salir del Juego.	Salir del Juego.	Satisfactoria
12	Tecla Back Space.	Regresa al menú de los videojuegos.	Regresa al menú de los videojuegos.	Satisfactoria

Tabla 60. Caso de Prueba RF-13.1

RF-13.1	Ingreso al Videojuego de Adentro-Afuera, Actividad Adentro			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Fondo de la pantalla.	En la parte inferior de la pantalla se muestra la actividad 1 (Adentro): Una caja que contiene un objeto llamado “oso”.	En la parte inferior de la pantalla se muestra la actividad 1 (Adentro): Una caja que contiene un objeto llamado “oso”.	Satisfactoria
3	Fondo de la pantalla.	La caja se encuentra en la parte superior derecha de la pantalla.	La caja se encuentra en la parte superior derecha de la pantalla.	Satisfactoria
4	Personaje del jugador.	Se muestra en pantalla en el lado superior izquierdo.	Se muestra en pantalla en el lado superior izquierdo.	Satisfactoria
5	Personaje objetivo.	Se muestra en pantalla en el lado superior derecho sobre la caja dibujada.	Se muestra en pantalla en el lado superior derecho sobre la caja dibujada.	Satisfactoria

6	Temporizador.	Muestra el tiempo de la actividad y el tiempo total en el lado superior derecho de la pantalla.	Muestra el tiempo de la actividad y el tiempo total en el lado superior derecho de la pantalla.	Satisfactoria
7	Puntaje.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Satisfactoria
8	Colisión personaje jugador con el objetivo.	Arrastre el objeto "oso" al objetivo.	Arrastre el objeto "oso" al objetivo.	Satisfactoria
9	Cumple la actividad.	Actualiza el marcador y muestra mensaje "Lo lograste".	Actualiza el marcador y muestra mensaje "Lo lograste".	Satisfactoria
10	Mensaje "Lo lograste".	Detener tiempo de la actividad.	Detener tiempo de la actividad.	Satisfactoria
11	Temporizador = 30 segundos.	Pasa a la actividad afuera.	Pasa a la actividad afuera.	Satisfactoria
12	Puntaje total de la actividad.	Se almacena en el vector listapuntos en la posición 0.	Se almacena en el vector listapuntos en la posición 0.	Satisfactoria
13	Tiempo utilizado para la actividad.	Se almacena en el vector listatiempo en la posición 0.	Se almacena en el vector listatiempo en la posición 0.	Satisfactoria

Tabla 61. Caso de Prueba RF-13.2

RF-13.2 Ingreso al Videojuego de Adentro-Afuera, Actividad Afuera				
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Fondo de la pantalla.	En la parte inferior de la pantalla se muestra la actividad 2 (Afuera): Una caja que tiene un objeto llamado "oso" afuera de la caja.	En la parte inferior de la pantalla se muestra la actividad 2 (Afuera): Una caja que tiene un objeto llamado "oso" afuera de la caja.	Satisfactoria
3	Fondo de la pantalla.	La caja se encuentra en la parte superior derecha de la pantalla.	La caja se encuentra en la parte superior derecha de la pantalla.	Satisfactoria
4	Personaje del jugador.	Se muestra en pantalla en el lado superior derecho.	Se muestra en pantalla en el lado superior derecho	Satisfactoria
5	Personaje del objetivo	Se muestra en pantalla en el lado superior izquierdo.	Se muestra en pantalla en el lado superior izquierdo.	Satisfactoria
6	Temporizador.	Muestra el tiempo de la actividad y el tiempo total en el lado superior derecho de la pantalla.	Muestra el tiempo de la actividad y el tiempo total en el lado superior derecho de la pantalla.	Satisfactoria
7	Puntaje.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Satisfactoria
8	Colisión personaje jugador con el objetivo.	Arrastre el objeto "oso" al objetivo.	Arrastre el objeto "oso" al objetivo.	Satisfactoria
9	Cumple la actividad.	Actualiza el marcador y muestra mensaje "Lo lograste".	Actualiza el marcador y muestra mensaje "Lo lograste".	Satisfactoria
10	Mensaje "Lo lograste".	Detener tiempo de la actividad.	Detener tiempo de la actividad.	Satisfactoria
11	Temporizador = 30 segundos.	Pasa a la actividad encima.	Pasa a la actividad encima.	Satisfactoria

12	Puntaje total de la actividad.	Se almacena en el vector listapuntos en la posición 1.	Se almacena en el vector listapuntos en la posición 1.	Satisfactoria
13	Tiempo utilizado para la actividad.	Se almacena en el vector listatiempo en la posición 1.	Se almacena en el vector listatiempo en la posición 1.	Satisfactoria

Tabla 62. Caso de Prueba RF-13.3

RF-13.3	Ingreso al Videojuego de Adentro-Afuera, Actividad Encima			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria
2	Fondo de la pantalla.	En la parte inferior de la pantalla se muestra la actividad 3 (Encima): Un objeto llamado "oso" que se encuentra encima de una mesa.	En la parte inferior de la pantalla se muestra la actividad 3 (Encima): Un objeto llamado "oso" que se encuentra encima de una mesa.	Satisfactoria
3	Fondo de la pantalla.	La mesa se encuentra en la parte superior de la pantalla.	La mesa se encuentra en la parte superior de la pantalla.	Satisfactoria
4	Personaje del jugador.	Se muestra en pantalla en el lado superior izquierdo.	Se muestra en pantalla en el lado superior izquierdo.	Satisfactoria
5	Personaje objetivo.	Se muestre en pantalla en el lado superior encima de la mesa dibujada.	Se muestre en pantalla en el lado superior encima de la mesa dibujada.	Satisfactoria
6	Temporizador.	Muestra el tiempo de la actividad y el tiempo total en el lado superior derecho de la pantalla.	Muestra el tiempo de la actividad y el tiempo total en el lado superior derecho de la pantalla.	Satisfactoria
7	Puntaje.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Satisfactoria
8	Colisión personaje jugador con el objetivo.	Arrastre el objeto "oso" al objetivo.	Arrastre el objeto "oso" al objetivo.	Satisfactoria
9	Cumple la Actividad.	Actualiza el marcador y muestra mensaje "Lo lograste".	Actualiza el marcador y muestra mensaje "Lo lograste".	Satisfactoria
10	Mensaje "Lo lograste".	Detener tiempo de la actividad.	Detener tiempo de la actividad.	Satisfactoria
11	Temporizador = 30 segundos.	Pasa a la actividad debajo.	Pasa a la actividad debajo.	Satisfactoria
12	Puntaje total de la actividad.	Se almacena en el vector listapuntos en la posición 2.	Se almacena en el vector listapuntos en la posición 2.	Satisfactoria
13	Tiempo utilizado para la actividad.	Se almacena en el vector listatiempo en la posición 2.	Se almacena en el vector listatiempo en la posición 2.	Satisfactoria

Tabla 63. Caso de Prueba RF-13.4

RF-13.4	Ingreso al Videojuego de Adentro-Afuera, Actividad Debajo			
Acción	Entrada	Valor Esperado	Valor Obtenido	Resultado Obtenido
1	Resolución de la pantalla.	Full screen 1024 px X 768px.	Full screen 1024 px X 768px.	Satisfactoria

2	Fondo de la pantalla.	En la parte inferior de la pantalla se muestra la actividad 4 (Deabajo): Un objeto llamado "oso" que se encuentra debajo de una mesa.	En la parte inferior de la pantalla se muestra la actividad 4 (Deabajo): Un objeto llamado "oso" que se encuentra debajo de una mesa.	Satisfactoria
3	Fondo de la pantalla.	La mesa se encuentra en la parte superior de la pantalla.	La mesa se encuentra en la parte superior de la pantalla.	Satisfactoria
4	Personaje del jugador.	Se muestra en pantalla en el lado superior izquierdo.	Se muestra en pantalla en el lado superior izquierdo.	Satisfactoria
5	Personaje objetivo.	Se muestra en pantalla en el lado superior debajo de la mesa dibujada.	Se muestra en pantalla en el lado superior debajo de la mesa dibujada.	Satisfactoria
6	Temporizador.	Muestra el tiempo de la actividad y el tiempo total en el lado superior derecho de la pantalla.	Muestra el tiempo de la actividad y el tiempo total en el lado superior derecho de la pantalla.	Satisfactoria
7	Puntaje.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Muestra el puntaje en el lado superior izquierdo de la pantalla.	Satisfactoria
8	Colisión personaje jugador con el objetivo.	Arrastre el objeto "oso" al objetivo.	Arrastre el objeto "oso" al objetivo.	Satisfactoria
9	Cumple la Actividad.	Actualiza el marcador y muestra mensaje "Lo lograste".	Actualiza el marcador y muestra mensaje "Lo lograste".	Satisfactoria
10	Mensaje "Lo lograste".	Detener tiempo de la actividad.	Detener tiempo de la actividad.	Satisfactoria
11	Temporizador = 30 segundos.	Pasa a la pantalla de fin de juego.	Pasa a la pantalla de fin de juego.	Satisfactoria
12	Puntaje total de la actividad.	Se almacena en el vector listapuntos en la posición 3.	Se almacena en el vector listapuntos en la posición 3.	Satisfactoria
13	Tiempo utilizado para la actividad.	Se almacena en el vector listatiempo en la posición 3.	Se almacena en el vector listatiempo en la posición 3.	Satisfactoria
14	Tecla ESC.	Salir del Videojuego.	Salir del Videojuego.	Satisfactoria
15	Tecla P.	Pausa el videojuego durante cinco segundos.	Pausa el videojuego durante cinco segundos.	Satisfactoria
16	Tecla Back Space.	Regresa al menú de los videojuegos.	Regresa al menú de los videojuegos.	Satisfactoria
17	Barra Espaciadora	Desactiva el sonido del videojuego.	Desactiva el sonido del videojuego.	Satisfactoria
18	Tecla V	Activa el sonido del videojuego.	Activa el sonido del videojuego.	Satisfactoria

4.5.2.2. Ejecución de las Pruebas

Tabla 64. Ejecución de Pruebas. Primera Vuelta

Ejecución Primera Vuelta				
Requerimiento	Error	Satisfactorio	Total Acciones	Índice de Severidad
RF-01	3	2	5	60%
RF-02	3	1	4	75%
RF-03	5	2	7	71%
RF-04	7	2	9	78%
RF-05	18	2	20	90%

RF-06	8	3	11	73%
RF-07	14	6	20	70%
RF-08	18	6	24	75%
RF-09	8	4	12	67%
RF-10	8	4	12	67%
RF-11	9	3	12	75%
RF-12	8	4	12	67%
RF-13.1	10	3	13	77%
RF-13.2	9	4	13	69%
RF-13.3	8	5	13	62%
RF-13.4	10	8	18	56%
TOTALES	146	59	205	71%

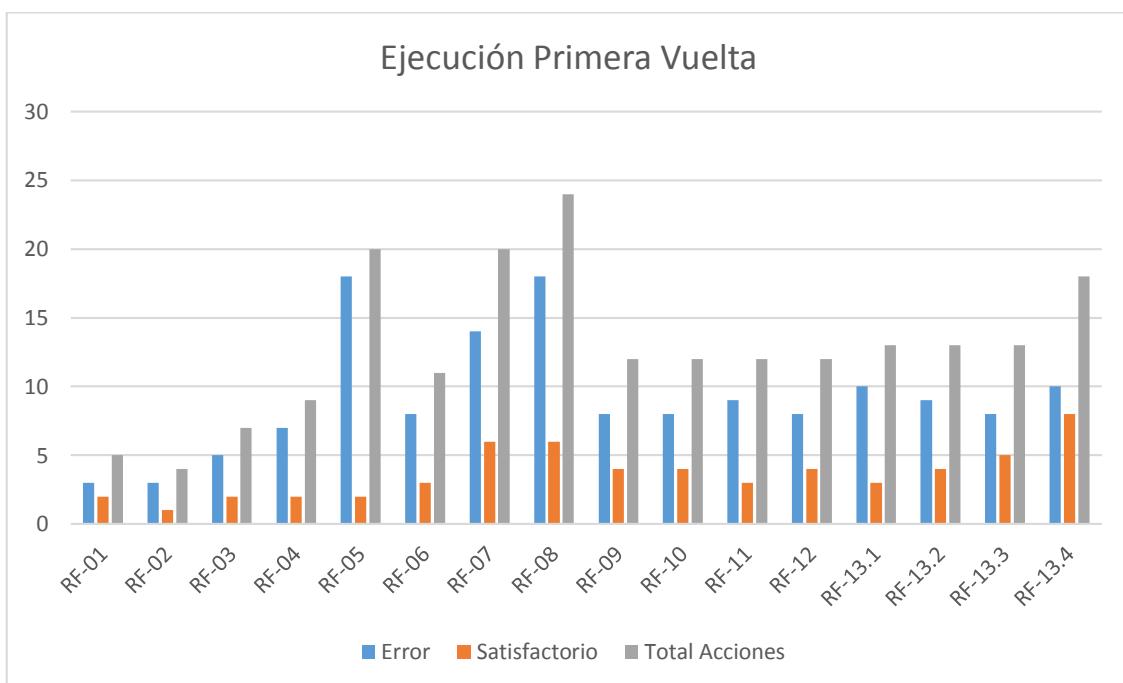


Figura 47. Gráfica de Ejecución de las Pruebas. Primera Vuelta

Tabla 65. Ejecución de Pruebas. Segunda Vuelta.

Ejecución Segunda Vuelta				
Requerimiento	Error	Satisfactorio	Total Acciones	Índice de Severidad
RF-01	1	4	5	20%
RF-02	1	3	4	25%
RF-03	2	5	7	29%
RF-04	3	6	9	33%
RF-05	10	10	20	50%
RF-06	6	5	11	55%
RF-07	9	11	20	45%
RF-08	12	12	24	50%
RF-09	5	7	12	42%
RF-10	4	8	12	33%

RF-11	5	7	12	42%
RF-12	3	9	12	25%
RF-13.1	6	7	13	46%
RF-13.2	5	8	13	38%
RF-13.3	4	9	13	31%
RF-13.4	6	12	18	33%
TOTALES	82	123	205	40%

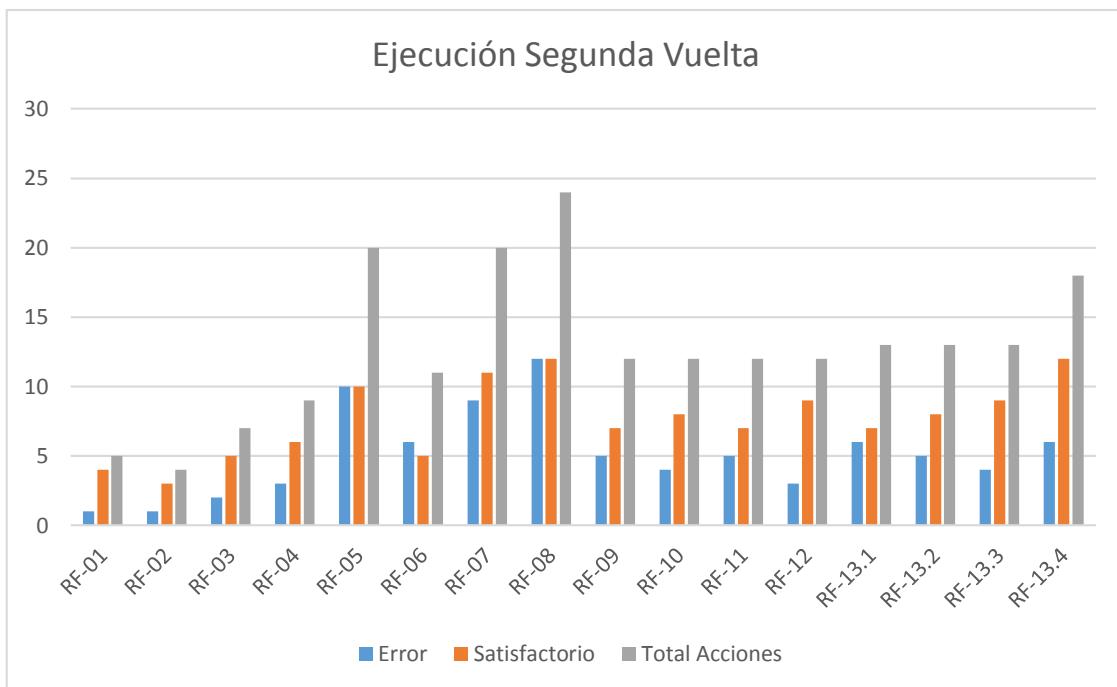


Figura 48. Gráfico de Ejecución de las Pruebas. Segunda Vuelta

Tabla 66. Ejecución de Pruebas. Tercera Vuelta.

Ejecución Tercera Vuelta				
Requerimiento	Error	Satisfactorio	Total Acciones	Índice de Severidad
RF-01	0	5	5	0%
RF-02	0	4	4	0%
RF-03	1	6	7	14%
RF-04	3	6	9	33%
RF-05	7	13	20	35%
RF-06	4	7	11	36%
RF-07	10	10	20	50%
RF-08	12	12	24	50%
RF-09	8	4	12	67%
RF-10	6	6	12	50%
RF-11	4	8	12	33%
RF-12	2	10	12	17%
RF-13.1	2	11	13	15%
RF-13.2	3	10	13	23%
RF-13.3	1	12	13	8%

RF-13.4	3	15	18	17%
TOTALES	66	139	205	32%

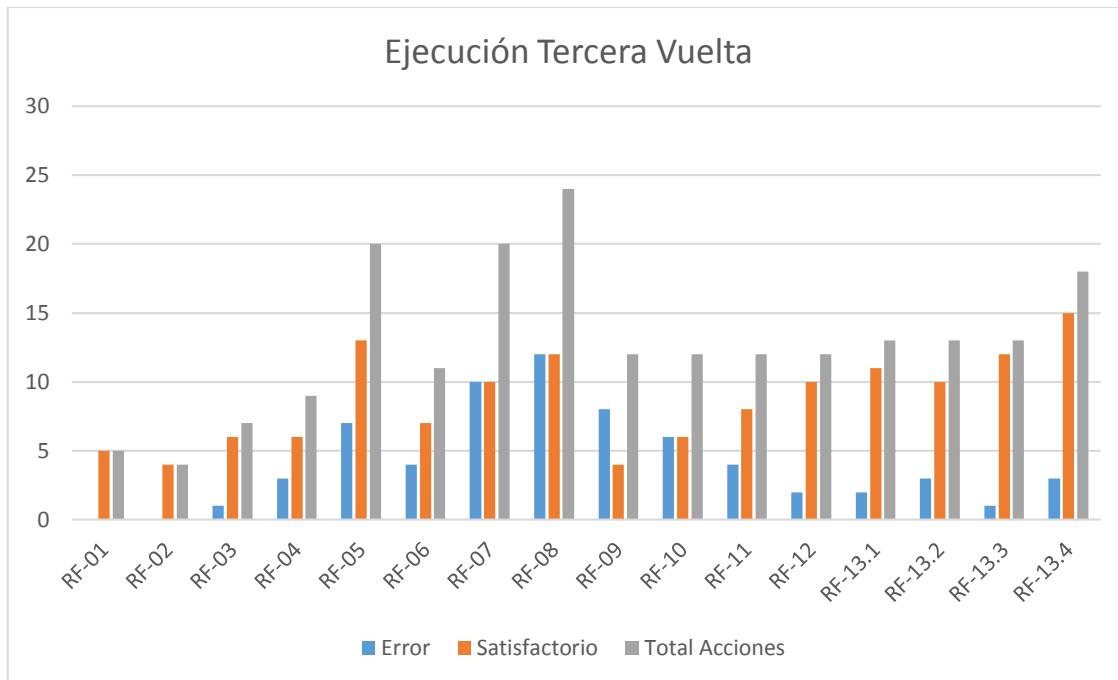


Figura 49. Gráfico de Ejecución de las Pruebas. Tercera Vuelta

Tabla 67. Ejecución de Pruebas. Cuarta Vuelta.

Ejecución Cuarta Vuelta				
Requerimiento	Error	Satisfactorio	Total Acciones	Índice de Severidad
RF-01	0	5	5	0%
RF-02	0	4	4	0%
RF-03	0	7	7	0%
RF-04	0	9	9	0%
RF-05	0	20	20	0%
RF-06	0	11	11	0%
RF-07	0	20	20	0%
RF-08	0	24	24	0%
RF-09	0	12	12	0%
RF-10	0	12	12	0%
RF-11	0	12	12	0%
RF-12	0	12	12	0%
RF-13.1	0	13	13	0%
RF-13.2	0	13	13	0%
RF-13.3	0	13	13	0%
RF-13.4	0	18	18	0%
TOTALES	0	205	205	0%

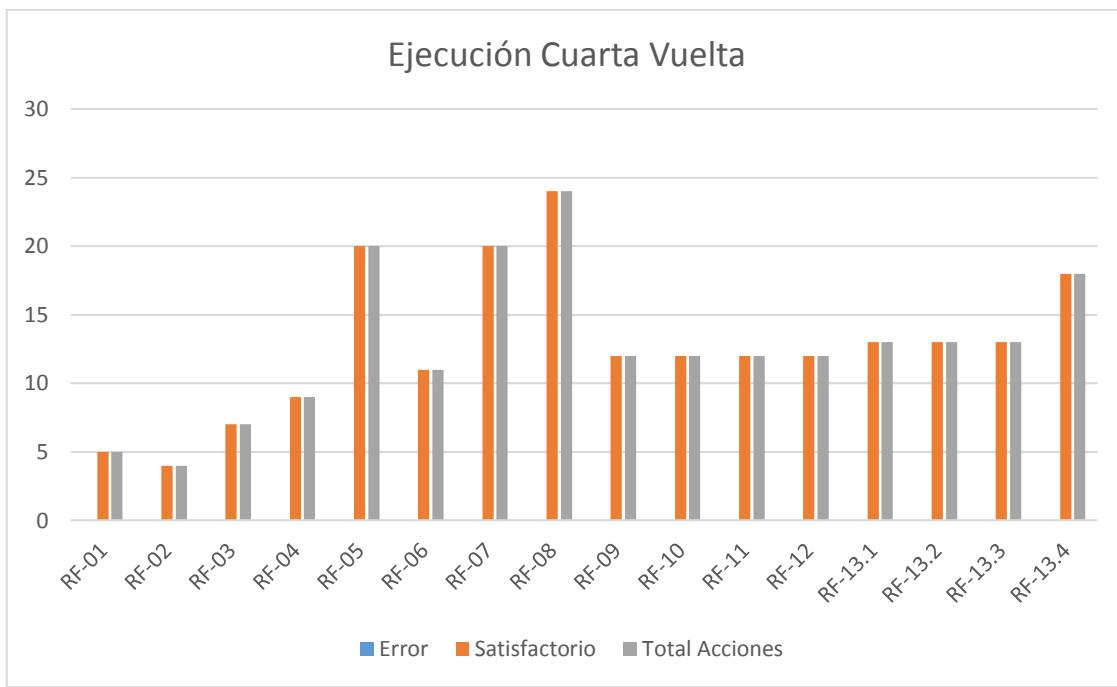


Figura 50. Gráfico de Ejecución de las Pruebas. Cuarta Vuelta

Tabla 68. Índice de Severidad Total

Vueltas	Índice de Severidad
1	71%
2	40%
3	32%
4	0%

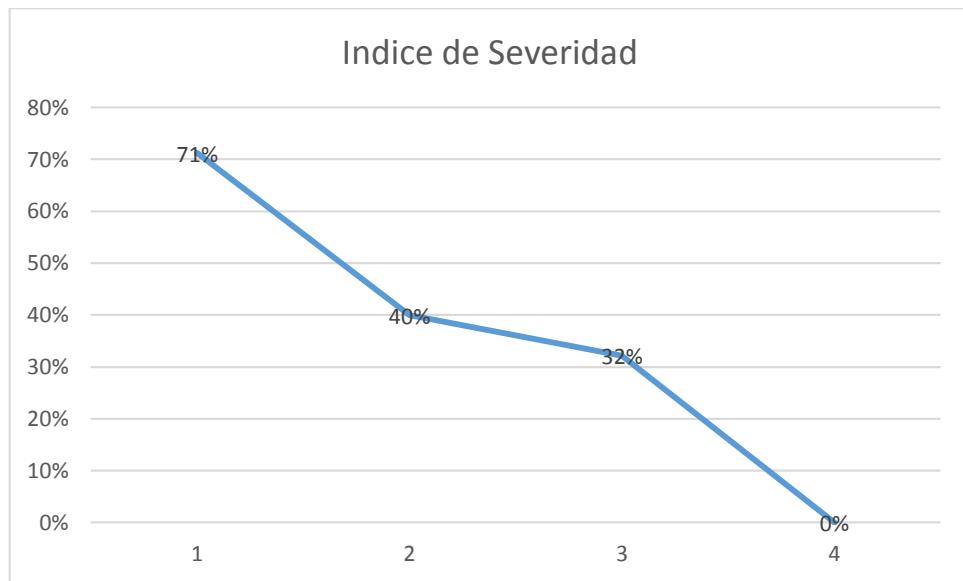


Figura 51. Gráfico del Índice de Severidad Total

4.5.3. Pruebas Beta

Estas pruebas se las lleva a cabo con las niñas de primero de básica y las docentes de la UEFLIC. Aquí se evalúa lo concerniente a los requisitos no funcionales de las métricas de usabilidad (Eficiencia, Eficacia y Satisfacción de Usuario).

En estas pruebas el videojuego debe de tener la menor cantidad posible de defectos menores y ningún defecto medio o crítico.

El desarrollo de estas pruebas se presenta en el siguiente capítulo de los resultados del proyecto.

A continuación, se presentan las pantallas finales del Videojuego UBIC:



Figura 52. Pantalla Inicial del Videojuego UBIC



JUEGO de UBICACIÓN ESPACIAL

Autores:

Fernando Recalde
Mario Merlo

VERSIÓN 1.0



Menú PRINCIPAL

Entrenamiento
Izquierda Derecha

Arriba Abajo
Adentro Afuera
Laberinto



Instrucciones del Juego: Entrenamiento

Tarea del jugador:

- El jugador deberá realizar 4 actividades.
- El jugador puede mover de izquierda a derecha, arriba y abajo.
- La actividad no necesariamente puede ser terminada porque tiene un tiempo estimado de 15 segundos para pasar a la siguiente actividad.

Juego:

El juego termina en un tiempo de 60 Segundos.

Si el jugador presiona la tecla ESC, saldrá de la aplicación.
Si el jugador presiona la tecla Back Space, regresara al menú de juegos.

Anterior ➤ **Siguiente**

Figura 53. Pantalla del Menú de los Videojuegos

Puntos: 6

Tiempo: 5

Izquierda

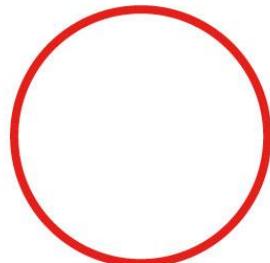


Figura 55. Pantalla del Videojuego de Entrenamiento

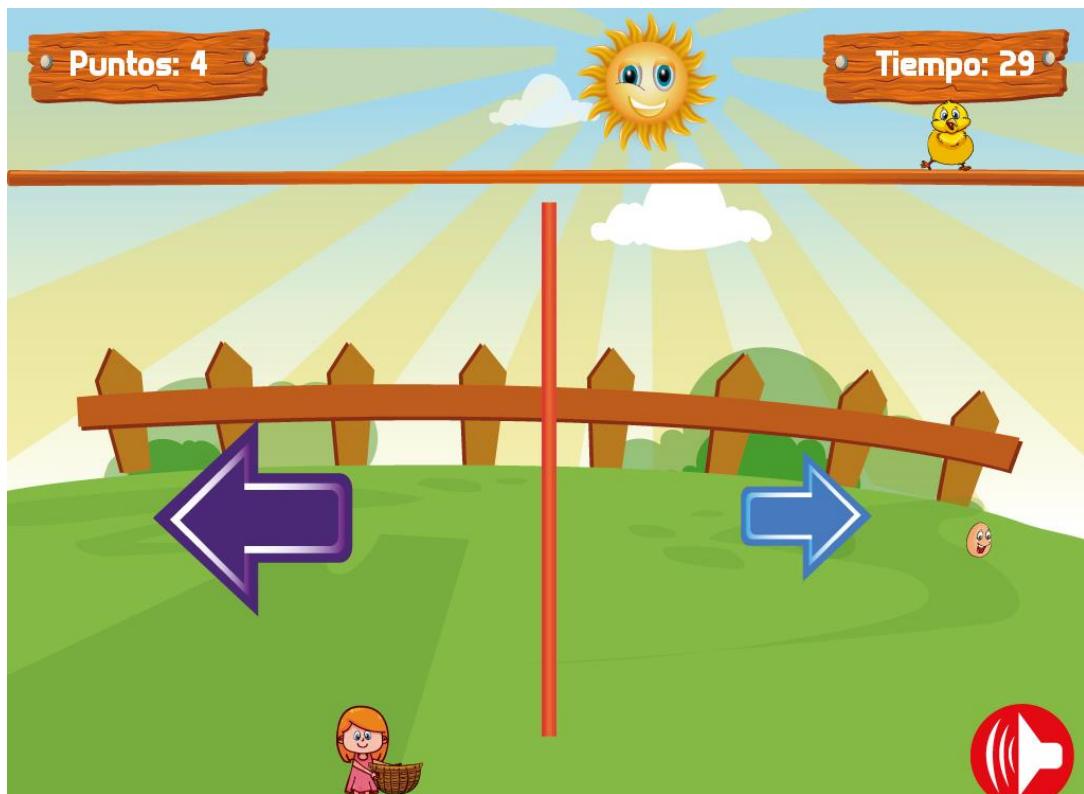


Figura 56. Pantalla del Videojuego Izquierda Derecha

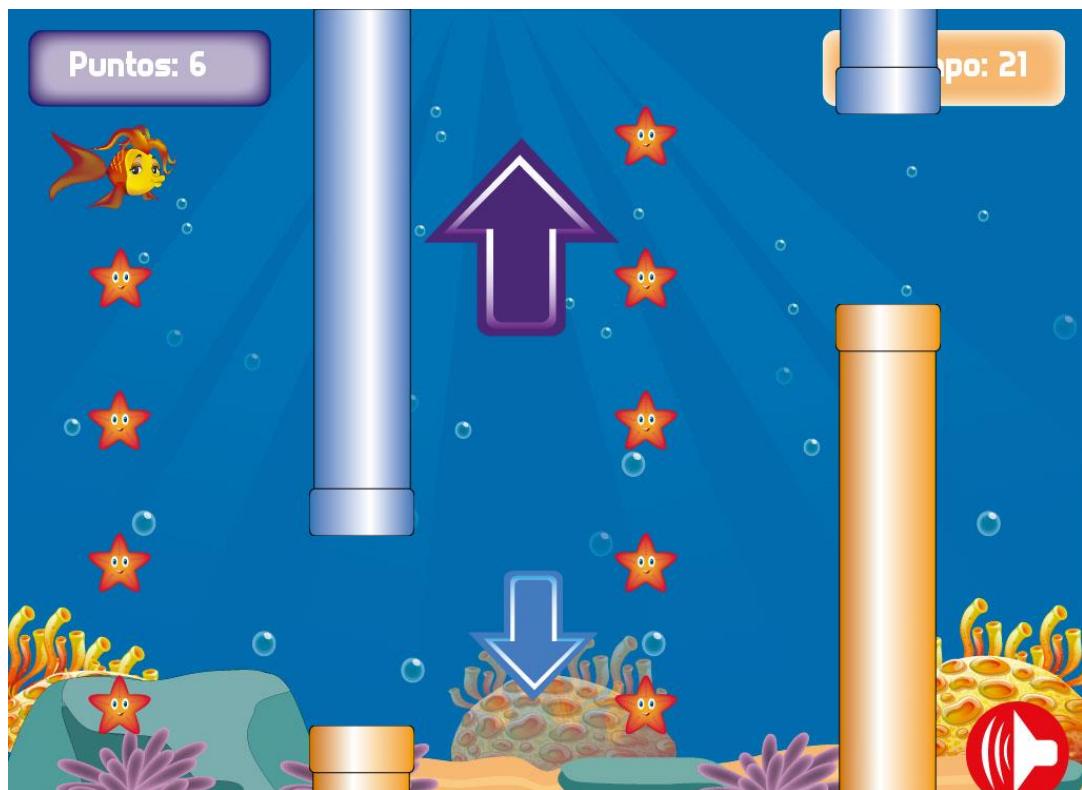


Figura 57. Pantalla del Videojuego Arriba Abajo



Figura 58. Pantalla del Videojuego Adentro Afuera

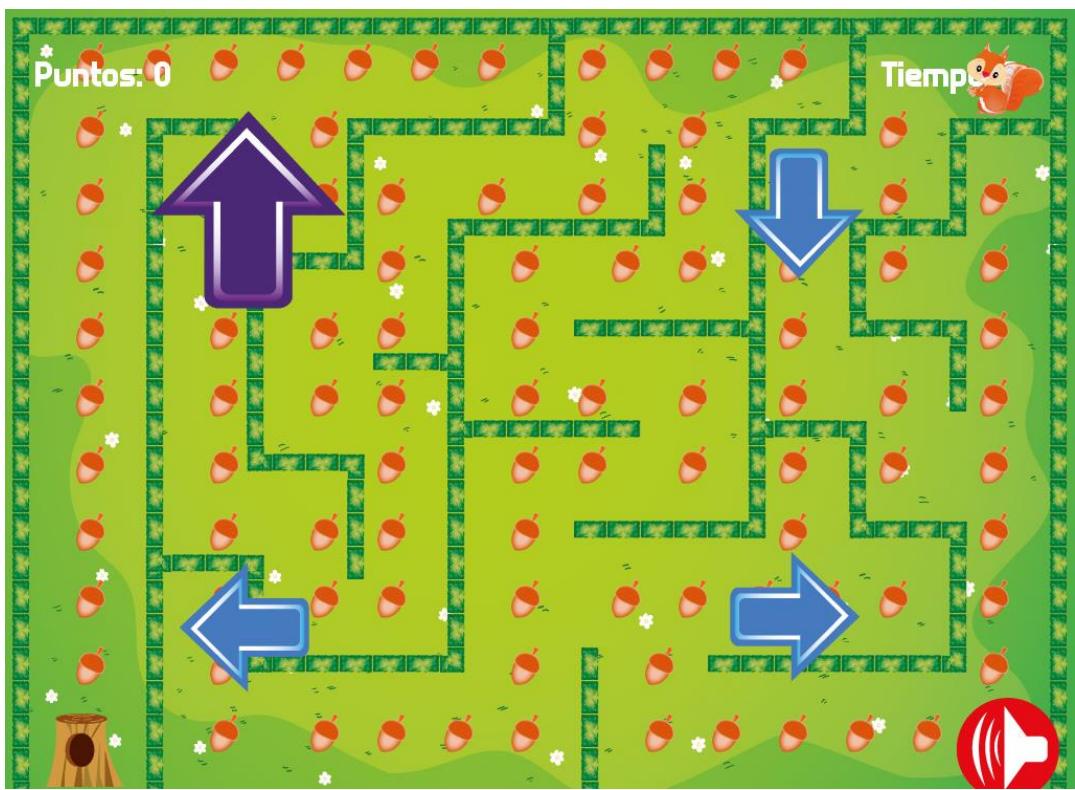


Figura 59. Pantalla del Videojuego Laberinto



JUEGO DE UBICACIÓN ESPACIAL

Figura 60. Pantalla de Fin de Juego

CAPÍTULO V

RESULTADOS

5.1. Selección de la Muestra

Para el desarrollo de la presente investigación se identifica una población descrita en la tabla 69 que está integrada por las niñas de preescolar de los primeros años de Educación Básica (periodo académico de septiembre del 2017 a Julio del 2018) de la UEFLIC de la ciudad de Ibarra.

Tabla 69. *Población de la Investigación*

Grados	Total de niñas
Primero de básica “A”	23
Primero de básica “B”	23
Primero de básica “C”	24
Total	70

El grupo objetivo de la presente investigación son las niñas de preescolar de los primeros años de básica de entre 5 a 6 años de edad quienes no tienen nociones de ubicación espacial. Del total de niñas identificadas en la población se selecciona una muestra probabilística de 10 niñas seleccionadas de manera aleatoria para garantizar su representatividad. (Lagares y Puerto, 2001)

Lo que nos representa un factor de muestreo del 14% que es el porcentaje de la población que representa la muestra.

$$\frac{n}{N} * 100\% = \frac{10}{70} * 100\% = 14\%$$

Y un factor de elevación de 7 que representa el número de elementos que hay en la población por cada elemento de la muestra.

$$\frac{N}{n} = \frac{70}{10} = 7$$

5.2. Evaluación de la Usabilidad

Los atributos de usabilidad a ser evaluados son eficacia, eficiencia y la satisfacción de usuario. Se propone un método de investigación cuantitativo-experimental para la recopilación de información sobre los tipos de evaluaciones que corresponden a cada uno de estos atributos. La eficacia y eficiencia se evalúan a través de observación directa; y para determinar la satisfacción del usuario se utiliza como instrumento la encuesta.

5.2.1. Metodología propuesta para evaluar la Usabilidad

Para el desarrollo del experimento se establecen dos grupos cada uno conformado por 10 niñas seleccionadas aleatoriamente del total de la población.

El primer grupo de niñas aprenden las nociones de ubicación espacial por medio del uso de los videojuegos de la presente propuesta, lo que nos lleva a la realización de la parte experimental del método que consiste en varias tareas con un grado de dificultad ascendente que son realizadas por las niñas en la aplicación de software en un periodo de tiempo limitado. Posteriormente, el evaluador registra, analiza e interpreta estos datos considerando los atributos de usabilidad.

El segundo grupo de niñas aprenden las nociones de ubicación espacial utilizando el método tradicional de enseñanza que conlleva el uso de canciones respecto al tema de ubicación espacial, y la ayuda de las partes del cuerpo como referencia. Este método está explicado con mayor detalle en el ítem 3.4 del capítulo 3, acerca de la observación realizada en el aula de clase.

Después se realiza una comparación entre los dos métodos. El primer método propuesto de enseñanza de ubicación espacial utilizando una herramienta tecnológica como es el uso de los videojuegos, en contraposición con el segundo método tradicional de enseñanza de ubicación espacial. Esta comparación se la realiza por medio de un test de tareas que realizan los dos grupos de niñas en un tiempo determinado.

5.2.2. Tareas Propuestas

Las tareas a ser realizadas en el experimento están descritas en función de cada uno de los cuatro videojuegos propuestos en la aplicación teniendo un nivel de dificultad ascendente. También se propone un videojuego adicional de entrenamiento que es ejecutado antes de los otros videojuegos sirviendo como ayuda en el uso de la interfaz natural de usuario NUI, el mismo que no será evaluado.

Tarea 1 (t1), Videojuego Izquierda – Derecha. El jugador debe mover el personaje de la niña con una canasta (personaje jugador) de izquierda a derecha recolectando los huevos que son arrojados por una gallina (personaje enemigo) cada 2 segundos, la cual se mueve de izquierda a derecha en la parte superior de la pantalla, si los huevos no son recogidos y sobrepasan el alto de la pantalla el videojuego termina. Para poder completar la tarea debe recoger un total de diez huevos en un tiempo menor a 150 segundos. Ver Figura 61 para una mejor comprensión.

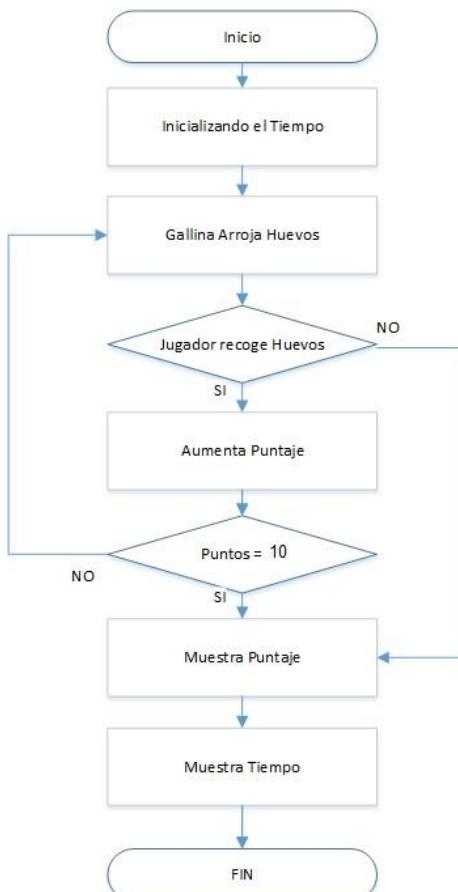


Figura 61. Tarea 1 (t1). Videojuego Izquierda - Derecha

Tarea 2 (t2), Videojuego Arriba – Abajo. El jugador debe mover el personaje del pez (personaje jugador) de arriba abajo recolectando las estrellas que están ubicadas en medio de los obstáculos (personaje enemigo) que se mueven de izquierda a derecha los cuales el jugador tiene que esquivar para que el videojuego no termine. Para completar la tarea debe recoger un total de veinte estrellas en un tiempo menor a 160 segundos. Ver Figura 62 para una mejor comprensión.

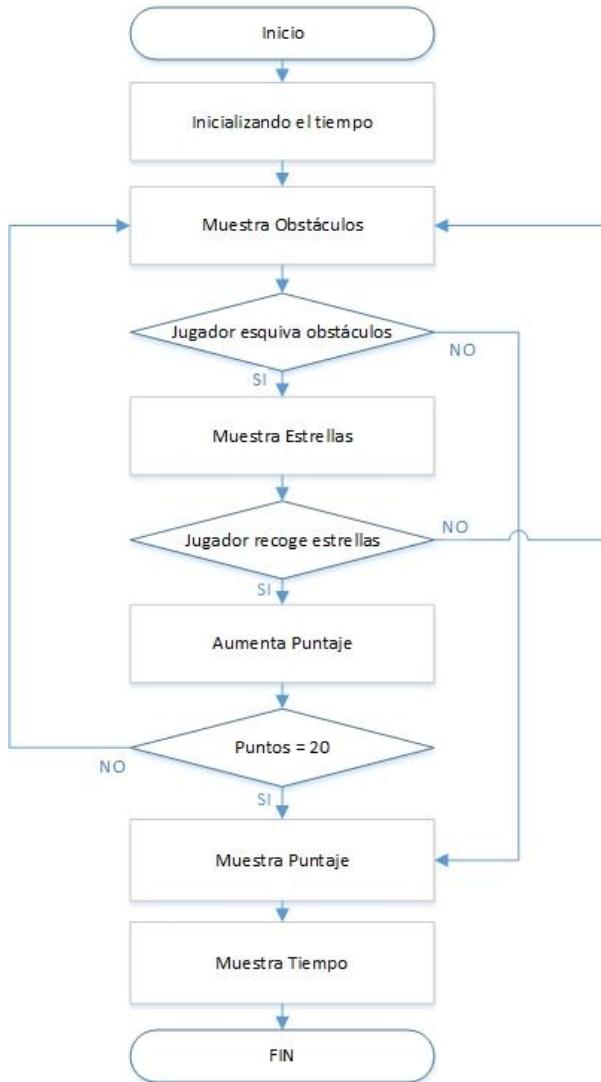


Figura 62. Tarea 2 (t2). Videojuego Arriba - Abajo

Tarea 3 (t3), Videojuego Adentro – Afuera. El jugador debe realizar las cuatro actividades propuestas, las mismas que son descritas en la parte inferior de la pantalla. En cada una de estas actividades el jugador debe mover el personaje del oso (personaje jugador) hacia el objetivo propuesto (personaje enemigo), cada actividad debe ser

realizada en un tiempo de 30 segundos. Si el jugador no completa una de las actividades el videojuego continúa con la siguiente actividad. Para completar la tarea el jugador debe culminar las cuatro actividades en un tiempo menor de 30 segundos por cada actividad dando un total de 120 segundos. Ver Figura 63 para una mejor comprensión.

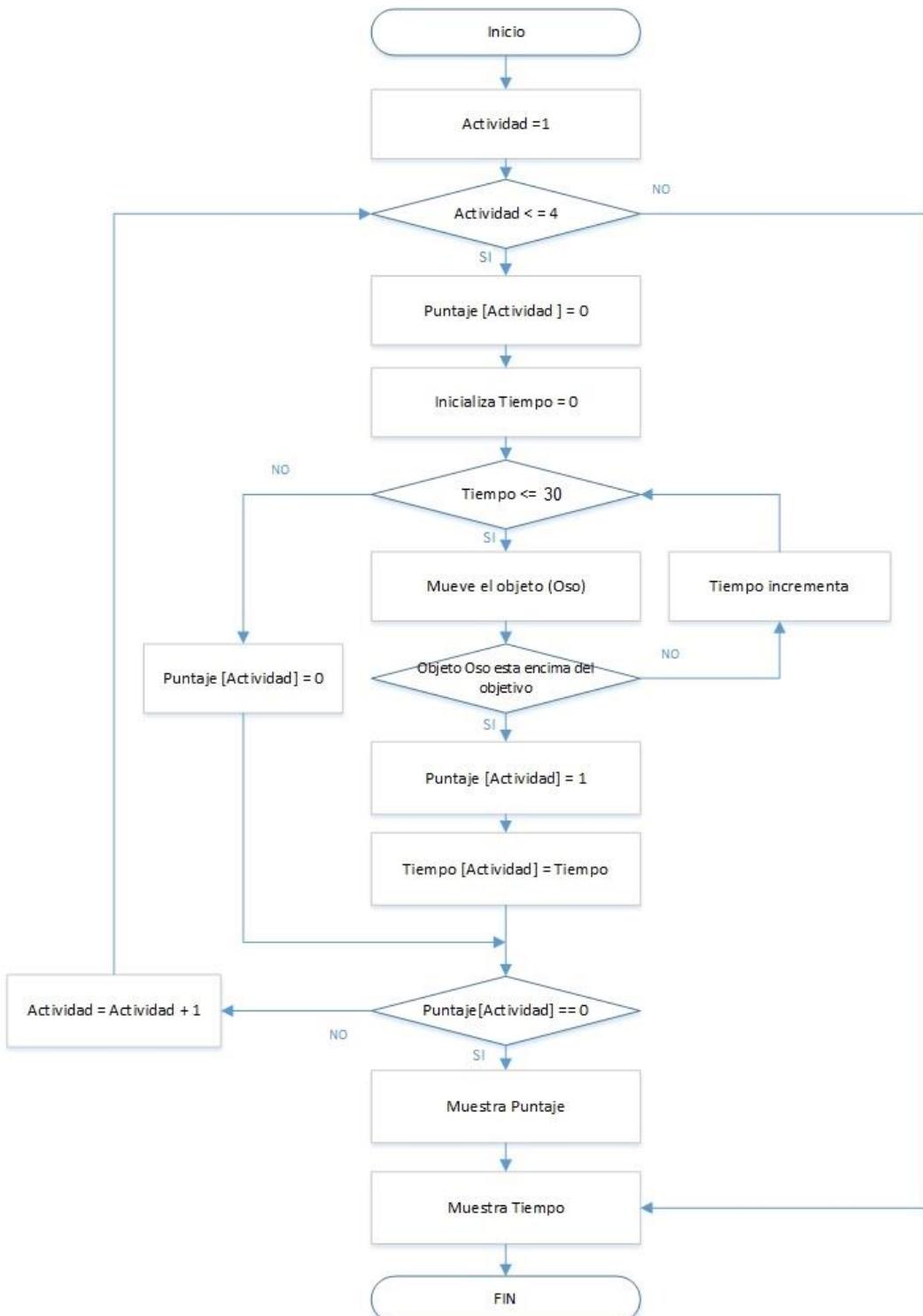


Figura 63. Tarea 3 (t3). Videojuego Adentro - Afuera

Tarea 4 (t4), Videojuego Laberinto. El jugador debe mover el personaje de la ardilla (personaje jugador) de izquierda a derecha de arriba a abajo, dentro del laberinto recolectando las nueces hasta llegar a chocar con el árbol (personaje enemigo) que representa el final del laberinto. Para completar la tarea el jugador debe determinar el camino más corto hacia el árbol del laberinto en un tiempo menor a 300 segundos. Ver Figura 64 para una mejor comprensión.

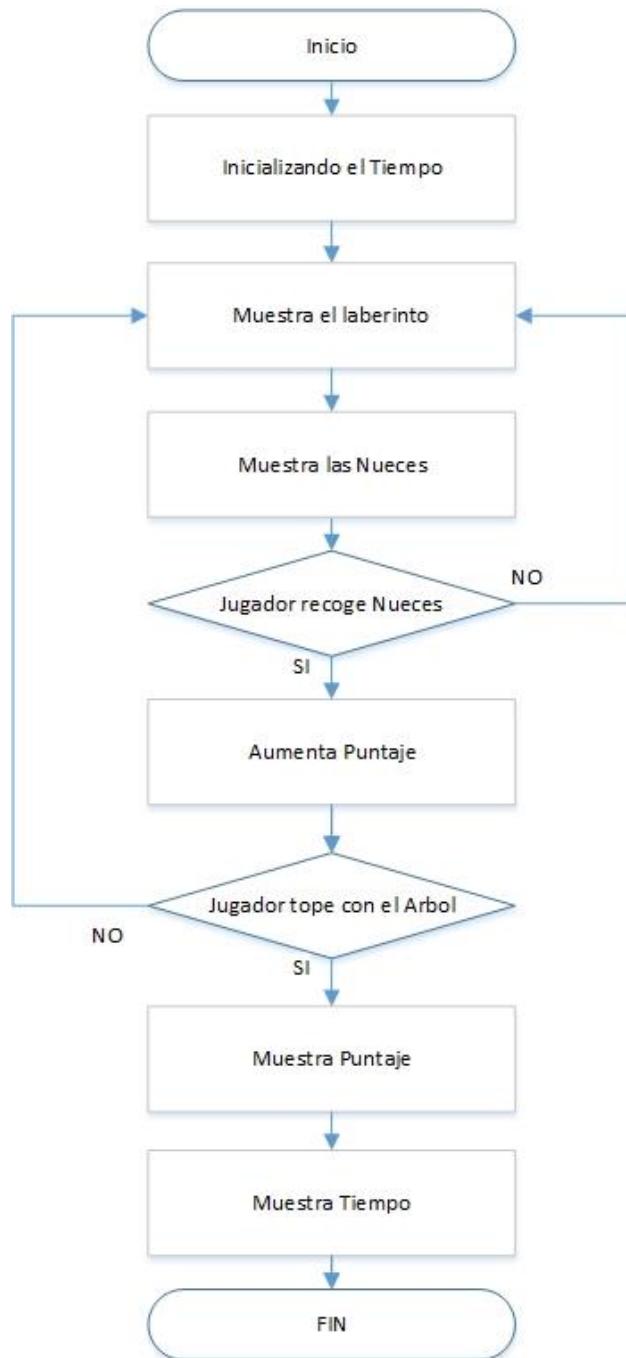


Figura 64. Tarea 4 (t4). Videojuego Laberinto

Para realizar la comparación entre la metodología propuesta y la metodología tradicional de enseñanza-aprendizaje de la ubicación espacial se realizó conjuntamente con las docentes parvularios de la unidad educativa un test de evaluación de cada aspecto propuesto de las nociones de ubicación espacial (izquierda, derecha, arriba, abajo, adentro, afuera, encima, debajo). El mismo que se puede observar en el Anexo C.

5.2.3. Consideraciones Generales para la Evaluación

Primero se realizó una pequeña capacitación a cada una de las Docentes de los Primeros Años de Educación Básica de la UEFLIC, sobre el uso de la aplicación de software propuesta. Demostrando su forma de uso y la manera en la que interactúa con el usuario. Luego se entregó a las Docentes las tareas con sus respectivos diagramas de flujo que explican de forma gráfica el desarrollo de las mismas, para que puedan de esta manera dar indicaciones a las niñas que participaron de la simulación.

Para que las niñas se familiaricen con el uso de la aplicación de software se les hizo utilizar el videojuego de entrenamiento, el cual no va a ser evaluado. Solo se lo usa como una ayuda en el proceso de enseñanza-aprendizaje.

Hay que considerar que para que una niña haya completado satisfactoriamente la tarea, debe cumplir rigurosamente con el objetivo planteado en cada una, antes o en un tiempo igual al tiempo límite establecido. Si la tarea no cumple con los objetivos propuestos, la tarea se declara insatisfactoria y se le asigna un tiempo de desarrollo igual al tiempo límite.

5.2.4. Prueba de Efectividad

Considera el cumplimiento (si o no) de las m tareas t_x donde $x \in \{1, 2, 3, \dots, m\}$. La efectividad e_x en cada tarea es acotado dentro del rango $[0, 1]$ de la siguiente manera: baja, $0 \leq e_x \leq 0.33$, media, $0.33 < e_x \leq 0.67$, y alta, $0.67 < e_x \leq 1$. Se calcula utilizando la relación.

$$e_x = \frac{v_x}{n} \quad (1)$$

Donde v_x es la cantidad de niñas que completaron la tarea x , satisfactoriamente, y n es el total de la muestra evaluada.

5.2.5. Prueba de Eficiencia

Esta prueba consiste en comparar los tiempos de desarrollo s_x de las m tareas t_x obtenidas tanto con la metodología propuesta de enseñanza de ubicación espacial como con la metodología tradicional. La media del tiempo para cada tarea \bar{s}_x se calcula usando la ecuación.

$$\bar{s}_x = \frac{1}{n} \sum_{i=1}^n s_{x_i} \quad (2)$$

Donde $s_{x_i} \in \mathbb{R}$ es el tiempo que la niña uso para completar la tarea x ; n es el total de la muestra. El radio r_{s_x} es calculado como la relación entre \bar{s}_x de las dos metodologías comparadas.

5.2.6. Prueba de Satisfacción

Esta prueba se basa en dos evaluaciones: satisfacción y comodidad / facilidad de uso. Ambas tienen una escala de calificación de cinco discretos niveles dentro del rango [1,5]. En ambas, se considera para la estadística el valor medio y la desviación estándar. Para calcular el valor medio \bar{y}_q se usa la siguiente ecuación.

$$\bar{y}_q = \frac{1}{n} \sum_{i=1}^n y_{q_i} \quad (3)$$

Donde q representa cada pregunta definida como $\{ q \in \mathbb{N} | 1 \leq q \leq 10 \}$ para la evaluación de satisfacción, y $\{ q \in \mathbb{N} | 1 \leq q \leq 4 \}$ para la evaluación comodidad / facilidad de uso; y_{q_i} es la calificación que el usuario coloca en cada pregunta q , n es el tamaño de la muestra.

La desviación estándar $\overline{S_q}$ se calcula con la ecuación.

$$\overline{S_q} = \sqrt{\frac{\sum_{i=1}^n (\overline{y_q} - y_{q_i})^2}{n}} \quad (4)$$

Donde todos los parámetros ya fueron definidos previamente.

Para evaluar la satisfacción de las niñas cuando usan la aplicación de software propuesta se empleó el cuestionario de usabilidad del sistema SUS (Brooke, 1996). La Escala de usabilidad del sistema (SUS) proporcionó una herramienta confiable y rápida para medir la usabilidad. Consiste en un cuestionario de 10 ítems con cinco opciones de respuesta que pueden seleccionar los encuestados el cual esta descrito en el Anexo C. El número 1 indica total desacuerdo y 5 total acuerdo con las preguntas contestadas, como se muestra en la tabla 70. Para esto, el procedimiento indicado por Brooke es usado para obtener una medida de satisfacción dentro del rango [0, 100%]. Le permite evaluar una amplia variedad de productos y servicios, incluidos hardware, software, dispositivos móviles, sitios web y aplicaciones. El algoritmo descrito por Brooke es el siguiente:

1. Para los elementos impares (preguntas 1, 3, 5, 7 y 9): Reste 1 de las respuestas obtenidas por los usuarios.
2. Para los ítems pares (preguntas 2, 4, 6, 8, 10): Reste 5 menos las respuestas de los usuarios.
3. Esto escala todos los valores de 0 a 4.
4. Sume las respuestas convertidas de cada usuario y multiplique ese total por 2.5. Esto convierte el rango de valores posibles de 0 a 100 en lugar de 0 a 40. La medida de satisfacción es obtenida en un máximo del 100%. (Rosero y Vaca, 2018)

Tabla 70. *Cuestionario de Escala de Usabilidad del Sistema*

Pregunta (q)	Estado [Valorado entre 1 y 5]
1	Me gustaría usar esta Aplicación de Software con frecuencia.
2	Encuentro esta Aplicación de Software innecesariamente compleja.
3	Creo que esta Aplicación de Software es fácil de usar.
4	Necesitaría ayuda para usar esta Aplicación de Software.

5	Las diversas funciones están bien integradas (constituyen un todo).
6	Hay demasiada incoherencia en esta Aplicación de Software.
7	La mayoría de las personas aprendería a usar esta Aplicación de Software muy rápidamente.
8	Me parece muy engorrosa y difícil de usar.
9	Tengo mucha confianza usándolo.
10	Necesito aprender muchas cosas antes de poder comenzar a utilizar esta Aplicación de Software.

Nota: Adaptado de Brooke (1996)

Para medir la comodidad y facilidad de uso de la interacción entre el usuario y la Aplicación de Software, la encuesta propuesta por Manresa, Ponsa, Varona y Perales (Manresa, Ponsa, Varona y Perales, 2010) es adaptada en la Tabla 71; en cada aspecto evaluado, el valor de 5 indica comodidad y /o facilidad de uso. El cuestionario se describe en mayor detalle en el Anexo D.

Tabla 71. Cuestionario de Comodidad / Facilidad de Uso

Pregunta (q)	Estado [Valorado entre 1 y 5]
1	El esfuerzo mental requerido para el desarrollo de las tareas ha sido [1 mucho esfuerzo, 5 poco esfuerzo].
2	La velocidad de funcionamiento es [1 muy lenta, 5 muy rápida].
3	La comodidad es [1 muy incómodo, 5 muy cómodo].
4	En general, el manejo de la Aplicación de Software es [1 muy difícil, 5 muy fácil].

Nota: Adaptado de Manresa, Ponsa, Varona y Perales (2010)

5.3. Pruebas y Resultados

A continuación, la metodología de evaluación propuesta se empleó en una aplicación de software que contiene varios videojuegos que son usados para la enseñanza-aprendizaje de la ubicación espacial. Además, se realiza el correspondiente análisis de resultados.

5.3.1. Configuración de la Simulación

La aplicación a evaluar en la presente simulación es un software que contiene un total de cuatro videojuegos y uno adicional de entrenamiento, diseñados para que sirvan de ayuda en el reforzamiento de la enseñanza-aprendizaje de la ubicación espacial en las niñas de preescolar de los primeros años de educación básica.

La aplicación de software fue desarrollada en Python 2.7.12, usando la librería especializada en el desarrollo de videojuegos pygame, esta interactúa con el uso de las Interfaces Naturales de Usuario NUI, precisamente con el movimiento, y el gesto de cerrar la mano del jugador, por medio del dispositivo o Sensor Kinect de Microsoft en su versión 1, modelo 1473.

Dicha aplicación es instalada en un computador con Sistema Operativo Linux Ubuntu 16.04 LTS, con un procesador Intel Core i5 a 2.50GHz y 12 Gb de RAM.

La muestra seleccionada para la evaluación está compuesta de niñas de entre cinco y seis años de edad, sin conocimientos o nociones previas de ubicación espacial de los primeros años de educación básica de la UEFLIC.

Los resultados de efectividad, eficiencia y satisfacción se muestran en pruebas que consideran las tareas estipuladas en la sección anterior. Los tiempos máximos de ejecución después de las cuales estas tareas se consideran fracasados son {150, 160, 120, 300} segundos, respectivamente.

5.3.2. Prueba de Efectividad

La tabla 72 muestra los resultados de la evaluación de efectividad donde se detalla el cumplimiento o no de las tareas realizadas por parte de las niñas de la UEFLIC.

Tabla 72. Efectividad de las Niñas al Realizar las Tareas

Usuario	Tarea 1 (t ₁)	Tarea 2 (t ₂)	Tarea 3 (t ₃)	Tarea 4 (t ₄)
1	Si	Si	No	Si
2	No	Si	Si	No
3	Si	No	Si	Si
4	Si	Si	No	Si
5	No	Si	Si	Si
6	Si	Si	No	Si
7	Si	Si	Si	No
8	No	Si	Si	Si
9	Si	Si	No	Si
10	Si	No	Si	Si

La Figura 65 muestra la efectividad existente e_x correspondientes a cada tarea calculadas a través de la ecuación (1).

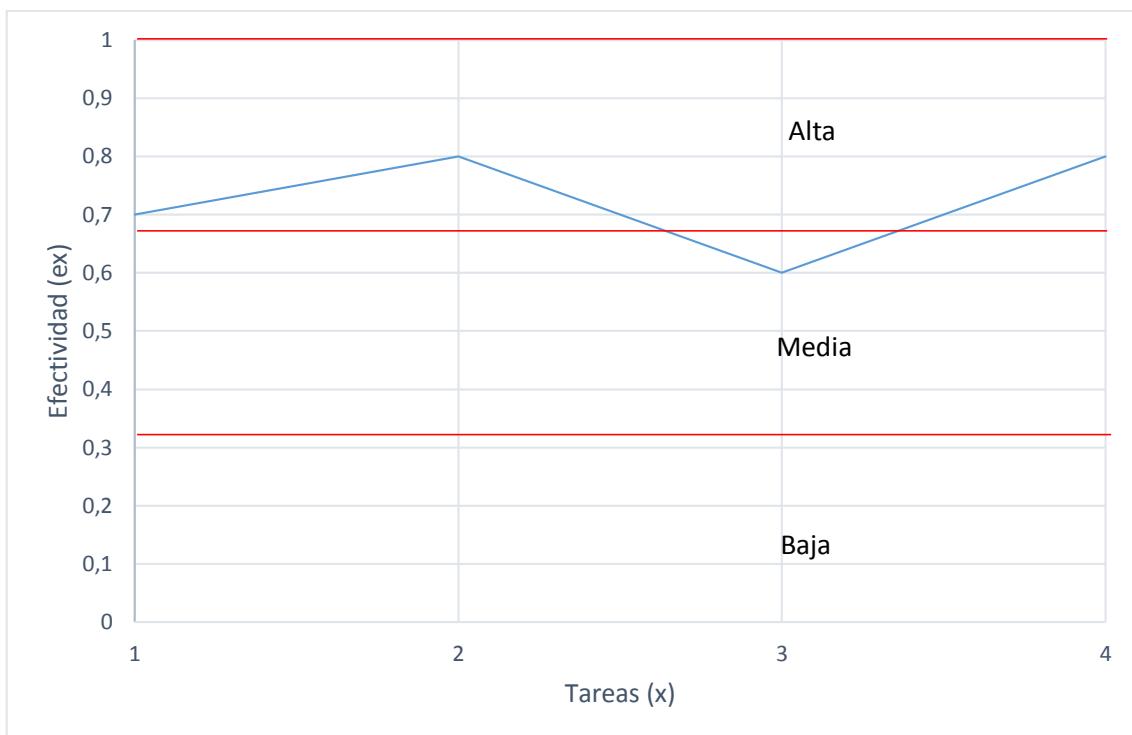


Figura 65. Efectividad para Realizar las Tareas t_1 a t_4

5.3.3. Prueba de Eficiencia

Consiste en comparar los tiempos de desarrollo para las tareas t_1 a t_4 obtenidas tanto con la metodología propuesta de enseñanza de ubicación espacial como con la metodología tradicional por medio del test de evaluación descrito en el Anexo E. En la Figura 66 se muestra el tiempo promedio \bar{s}_x de las tareas x que es calculado para cada metodología usando la ecuación (2). En la tabla 73 se muestran los promedios de cada una de las metodologías y el radio entre los tiempos de desarrollo r_{s_x} que es visualizado de una mejor manera en la Figura 67.

Tabla 73. Radio de la Comparación de los Promedios de las Metodologías Analizadas

Tareas	Promedio Metodología Tradicional	Promedio Metodología Propuesta	Radio
t_1	130,8	109,3	0,84
t_2	125,6	91,2	0,73
t_3	90,6	62,9	0,69
t_4	223,8	177,3	0,79

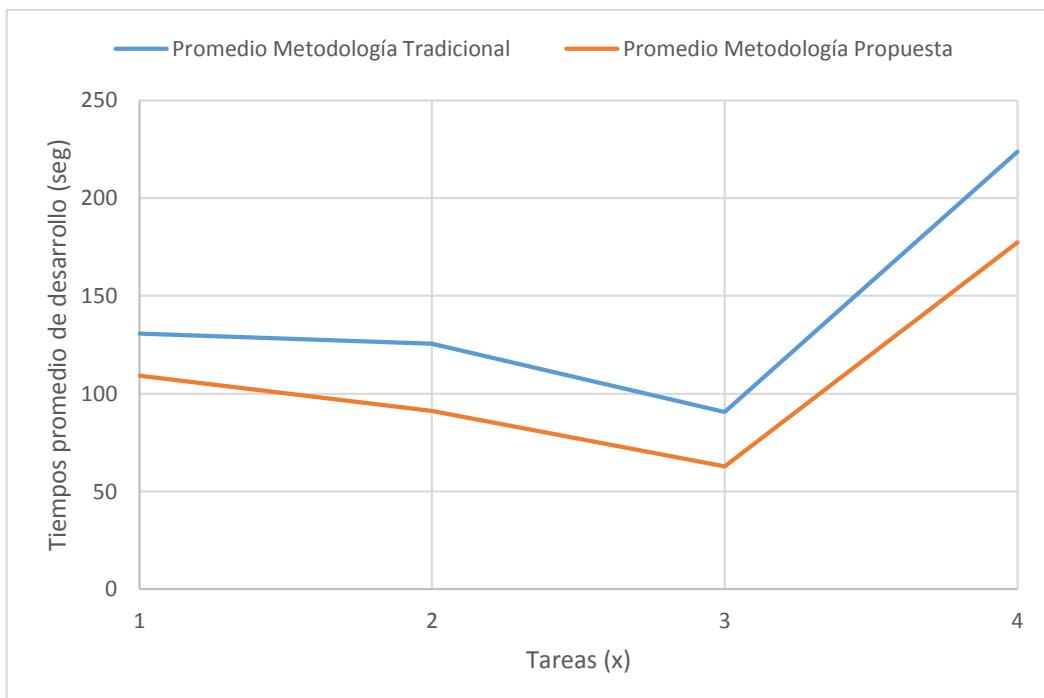


Figura 66. Tiempo Promedio de Desarrollo de las Tareas t₁ a t₄

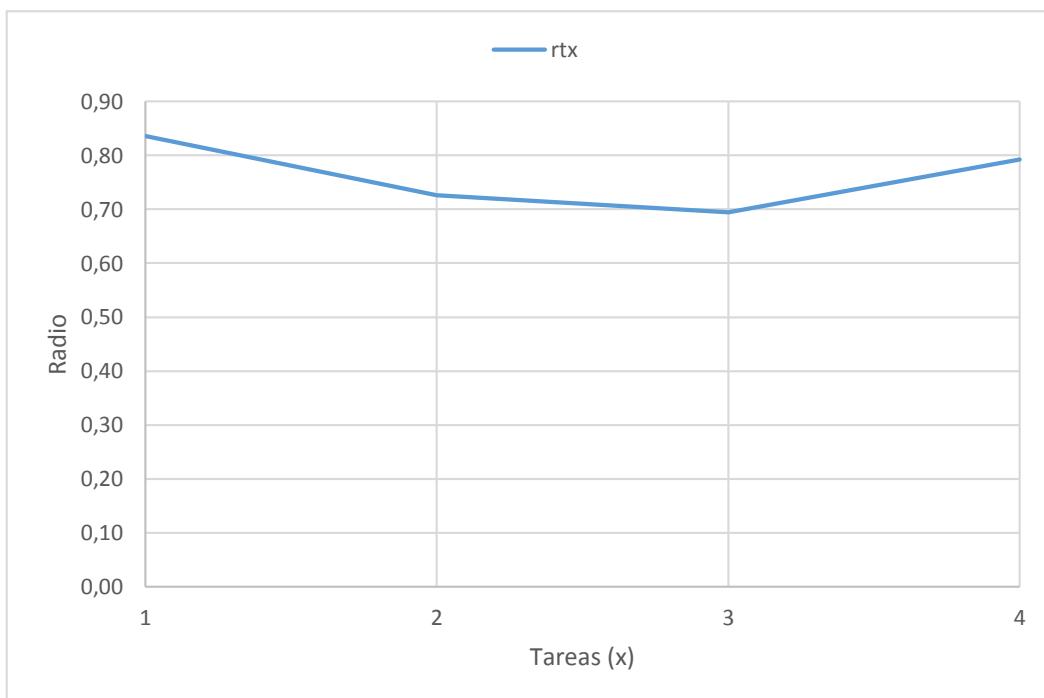


Figura 67. Radio de los Tiempos de Desarrollo para las Tareas t₁ a t₄

5.3.4. Prueba de Satisfacción

Los resultados de la encuesta de escala de usabilidad se pueden observar en la Tabla 74; se han calculado el valor medio \bar{y}_q y la desviación estándar \bar{S}_q a través de las ecuaciones (3) y (4) respectivamente.

Tabla 74. Resultados de la Encuesta de Escala de Usabilidad

Pregunta (q)	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	Media (\bar{y}_q)	Desviación Estándar (\bar{S}_q)
1	4	5	3	4	5	4	2	5	4	5	$\bar{y}_1 = 4,1$	$\bar{S}_1 = 0,99$
2	1	2	3	1	2	1	2	2	1	1	$\bar{y}_2 = 1,6$	$\bar{S}_2 = 0,70$
3	5	5	4	5	4	5	3	4	5	5	$\bar{y}_3 = 4,5$	$\bar{S}_3 = 0,71$
4	2	1	2	1	3	1	1	2	1	2	$\bar{y}_4 = 1,6$	$\bar{S}_4 = 0,70$
5	5	4	5	5	4	5	4	5	4	5	$\bar{y}_5 = 4,6$	$\bar{S}_5 = 0,52$
6	1	1	2	1	2	1	2	1	1	2	$\bar{y}_6 = 1,4$	$\bar{S}_6 = 0,52$
7	4	5	4	5	5	4	5	4	5	3	$\bar{y}_7 = 4,4$	$\bar{S}_7 = 0,70$
8	2	1	1	3	1	1	2	1	3	1	$\bar{y}_8 = 1,6$	$\bar{S}_8 = 0,84$
9	5	5	4	5	4	5	5	4	5	3	$\bar{y}_9 = 4,5$	$\bar{S}_9 = 0,71$
10	1	2	1	2	1	1	2	1	1	2	$\bar{y}_{10} = 1,4$	$\bar{S}_{10} = 0,52$

Utilizando el algoritmo de Broke se obtuvo los siguientes valores mostrados en la tabla 75; en la que se calcula la satisfacción de Usuario en un porcentaje de 0 a 100 % y la media de Satisfacción de Usuario.

Tabla 75. Resultados de los Porcentajes de Satisfacción de Usuario

Pregunta (q)	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10
1	3	4	2	3	4	3	1	4	3	4
2	4	3	2	4	3	4	3	3	4	4
3	4	4	3	4	3	4	2	3	4	4
4	3	4	3	4	2	4	4	3	4	3
5	4	3	4	4	3	4	3	4	3	4
6	4	4	3	4	3	4	3	4	4	3
7	3	4	3	4	4	3	4	3	4	2
8	3	4	4	2	4	4	3	4	2	4
9	4	4	3	4	3	4	4	3	4	2
10	4	3	4	3	4	4	3	4	4	3
Puntaje Total	36	37	31	36	33	38	30	35	36	33
Valor de Conversión del Rango de (0 a 100)	2,5	2,5	2,5	2,5	2,5	2,5	2,5	2,5	2,5	2,5
Satisfacción de Usuario (%)	90	92,5	78	90	82,5	95	75	87,5	90	82,5

Media de Satisfacción de Usuario	86,3	86,3	86,3	86,3	86,3	86,3	86,3	86,3	86,3	86,3
----------------------------------	------	------	------	------	------	------	------	------	------	------

La Figura 68 muestra los Resultados de la Satisfacción de Usuario en porcentajes de la encuesta de la escala de Usabilidad del sistema.

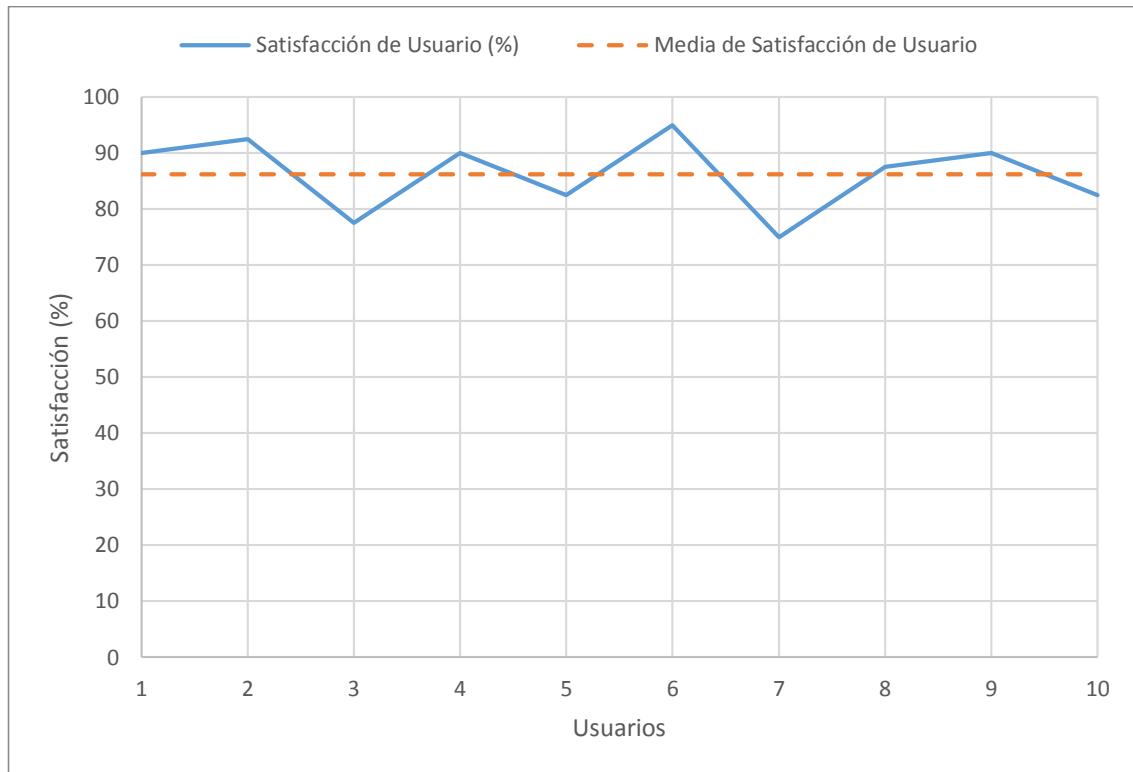


Figura 68. Satisfacción de Usuario a Partir de la Encuesta de la Escala de Usabilidad del Sistema.

Por otro lado, los datos obtenidos de la encuesta de comodidad/facilidad de uso se encuentran en la Tabla 76; también se han calculado el valor medio \bar{y}_q y la desviación estándar \bar{S}_q a través de las ecuaciones (3) y (4) respectivamente y se realiza una apreciación final para cada pregunta.

Tabla 76. Comodidad/Facilidad de Uso

Pregunta (q)	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	Media (\bar{y}_q)	Desviación Estándar (\bar{S}_q)	Apreciación Final
1	5	4	5	4	5	3	5	5	4	5	$\bar{y}_1 = 4,5$	$\bar{S}_1 = 0,71$	Poco esfuerzo
2	4	4	5	5	5	2	4	4	5	4	$\bar{y}_2 = 4,2$	$\bar{S}_2 = 0,92$	Rápido
3	5	5	3	5	4	5	5	4	5	2	$\bar{y}_3 = 4,3$	$\bar{S}_3 = 1,06$	Cómodo
4	4	5	2	4	5	5	4	5	4	5	$\bar{y}_4 = 4,3$	$\bar{S}_4 = 0,95$	Fácil de Usar

5.3.5. Discusión Final Sobre la Evaluación del Videojuego de Ubicación Espacial

La Figura 65 muestra que la efectividad de los usuarios al utilizar el videojuego de enseñanza de ubicación espacial se mantiene estable entre los valores de 0,6 a 0,8 mostrando una alta efectividad en su uso. Teniendo en cuenta que las niñas con las que se realiza la evaluación son inexpertas en el uso de los Videojuegos propuestos para la enseñanza de la ubicación espacial, se puede prever que existan fallos y no se alcance una efectividad de 1 incluso en la tarea que presenta menor dificultad.

En la Figura 67 se muestran los resultados de la evaluación de eficiencia. La curva de radio de tiempo de desarrollo permanece siempre menor que 1, lo que nos indica que los tiempos con la Metodología Propuesta son menores que con la Metodología Tradicional. Se presenta un descenso 0,84 a 0,69 hasta la tercera tarea evidenciando un incremento en la eficiencia conforme la dificultad de las tareas aumenta, aunque al final en la cuarta tarea presentó un aumento de 0,69 a 0,79 se podría decir que la eficiencia de la metodología propuesta no se altera en mayor medida.

Si bien el atributo de satisfacción permite identificar percepciones y sentimientos subjetivos de los usuarios en relación con el uso del videojuego de enseñanza de ubicación espacial, los valores medios y las desviaciones estándar muestran una apreciación cuantitativa de ellos. En las tablas 74 y 76 se observan directamente las calificaciones de cada pregunta a través de los valores medios \bar{y}_q , mientras que las dispersiones en las respuestas se muestran mediante las desviaciones estándar \bar{S}_q .

La tabla 74 evidencia una valoración global de satisfacción positiva del usuario al interactuar con el videojuego de enseñanza de ubicación espacial debido a los favorables valores medios encontrados. Las desviaciones estándar muestran valores relativamente bajos, lo que evidencia una concordancia en las apreciaciones de todos los usuarios evaluados. Además, en la tabla 75 se muestran los valores de satisfacción de cada uno de los usuarios dentro de una escala de 0 a 100% con una media del 86,3% que corrobora una aceptación positiva en el uso del videojuego de enseñanza de ubicación espacial, mismos datos que son representados gráficamente en la Figura 68.

La tabla 76 muestra que los usuarios se sienten cómodos al utilizar el Videojuego de enseñanza de ubicación espacial y que además lo consideran fácil de utilizar. Sin embargo, las desviaciones estándar son relativamente grandes lo que evidencia que los usuarios tuvieron respuestas dispersas dentro de la escala de calificación.

A pesar de la existencia de resultados favorables en términos de usabilidad y comodidad / facilidad de uso, las desviaciones estándar corroboran el desacuerdo de los usuarios que no cumplieron con las actividades satisfactoriamente. La evaluación de satisfacción traduce preguntas cualitativas en respuestas cuantitativas que permiten medir cuan fácil y cómodo, el videojuego es manipulado por el usuario.

CAPÍTULO VI

CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones

- La investigación del tema en ubicación espacial, se fundamentó usando estudios de autores reconocidos como Piaget aportando una visión favorable en la temática del proceso de aprendizaje de los niños.
- Existen algunos tipos de videojuegos como de entretenimiento y los llamados Serious Game (Videojuegos Serios), que al ser usados en la educación aportan ventajas en el desarrollo de las habilidades de los niños.
- Se propone un algoritmo siguiendo directrices en el diseño y bosquejo de las interfaces gráficas, combinando las técnicas de investigación científica con la metodología de desarrollo ágil XP.
- Los prototipos del videojuego (izquierda-derecha, arriba-abajo, adentro-afuera y laberinto) fueron evaluados con las niñas de prescolar, permitiendo una evolución de un producto mínimo viable a un producto final que, por medio de una cámara de profundidad como el Kinect, fue utilizada como herramienta de reforzamiento del aprendizaje de la ubicación espacial.
- En general, las medidas subjetivas de las pruebas de satisfacción apoyan las pruebas de eficiencia y efectividad del videojuego evaluado; esto nos proporciona medidas aceptables de que la metodología de enseñanza de ubicación espacial propuesta tiene un rendimiento aceptable.

6.2. Recomendaciones

- Se sugiere el uso de tecnologías de la información y comunicación para la enseñanza-aprendizaje en el reforzamiento de ubicación espacial.
- Se recomienda realizar una investigación más profunda en las áreas de Inteligencia Artificial, Visión Artificial y el uso de Cámaras de profundidad en la solución de diversos problemas de la vida diaria como puede ser identificación de objetos, colores, imágenes, reconocimiento a largas distancias.
- En la realización de proyectos de software cortos es recomendable el uso de metodología de desarrollo ágil como XP, porque abarcan las diferentes fases de

desarrollo del software de una manera práctica permitiendo una programación con menores errores y organizada, adaptándose rápidamente al medio en el que utiliza, generando nuevas versiones para ser utilizadas.

- Cuando se desarrolla software es conveniente involucrar al usuario durante todas las etapas del desarrollo del proyecto, porque al momento del inicio del proyecto el usuario tiene una visión incompleta de lo que necesita, el ingeniero de software establece requerimientos que permiten tener una visión clara al usuario para que el producto final sea lo más aceptable posible.

6.3. Trabajo Futuro

- Hacer un estudio más profundo sobre las directrices propuestas.
- Evaluar las directrices del algoritmo propuesto, comparándolas con otras metodologías que se encuentren en el estado del arte.
- Realizar un estudio de usabilidad sobre las directrices de tipos color, formas, iconografía y fuentes de letras más recomendadas en el desarrollo de prototipos de interfaces naturales de usuario.

REFERENCIAS

ActiveState Software Inc. (2018). ActiveState Code >> Recipes. Recuperado el 21 de febrero de 2018 de <http://code.activestate.com/recipes/578104-openkinect-mouse-control-using-python/>

Aguilar, R. (2011). *La Educación a Distancia: Fundamentos, Teorías y Contribuciones*. Loja, Ecuador: Editorial UTPL.

Alva – Obeso, M.E. (2005). *Metodología de Medición y Evaluación de la Usabilidad en Sitios Web Educativos*. (Tesis Doctoral). Universidad de Oviedo, Oviedo, España.

Andreu – Andrés, M. A. y García – Casas, M. (2000). Actividades lúdicas en la enseñanza de LFE: el juego didáctico, *I Congreso Internacional de español para Fines Específicos, At Amsterdam, Holanda*, Ministerio de Educación Cultura y Deporte, 1, 121 – 125.

Bernal, C. A. (2010). *Metodología de la Investigación*, 3º Edición, Bogotá D.C., Colombia: Pearson Educación.

Beltré – Ferreras, H. J. (2008). *Aplicación de la usabilidad al proceso de desarrollo de páginas web*. (Tesis de Maestría). Universidad Politécnica de Madrid, Madrid, España.

Brooke, J. (1996). SUS-A quick and dirty usability scale, *Usability evaluation in industry*, 189(194), 4 – 7.

Cabezas – Portilla, E. P. (2014). *Orientación Espacial en la Pre-Escritura de niños de primero de educación básica de la Unidad Educativa “La Salle”, Quito, Período Lectivo 2011-2012*, (Trabajo de grado). Universidad Central del Ecuador, Quito, Ecuador

Calderón – Jácome, M. T. y Maldonado – Andrade, D. J. (2014). *Control por visión de un cuadricóptero utilizando ROS*. (Trabajo de grado). Escuela Politécnica Nacional, Quito, Ecuador.

Castro – Pena, M. L. y López – Gómez, S. (2013). Climántica, The Game: Crea y gestiona territorios sostenibles. Proceso de creación y evaluación de la usabilidad de un Serious Game. *Challenges 2013: Aprender a cualquier hora e em qualquer lugar, learning anytime anywhere*, 1(1), 45 - 59

Código de la Niñez y Adolescencia. *Ley N° 2002-100*, Quito, Ecuador, 3 de Julio de 2003.

Cohn, M. (2005). *Agile Estimating and Planning*, Primera Edición, Estados Unidos de Norteamérica: Pearson Education.

Constitución de la República del Ecuador (2008). *Registro Oficial, N° 449*, Quito, Ecuador, 20 de octubre 2008.

Contreras – Delgado, E.C. y Contreras – González, I.I, (2014). Desarrollo de habilidades cognitivas mediante videojuegos en niños de educación básica, *Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, 1(12), 1 -19.

Duque, E. y Vásquez, A. (2015). *NUI para la educación. Eliminando la discriminación tecnológica en la búsqueda de la Inclusión digital*. Centro de Investigaciones, Corporación Universitaria Americana, Componente Apropiación Social de TIC, UPB – Antioquia Digital, Colombia.

Emprende a Conciencia (2019). Cómo se aplican las metodologías Design Thinking, Lean, Agile y Growth Hacking de manera eficiente. Recuperado el 11 de febrero de 2019 de <https://emprendeaconciencia.com/blog/como-se-aplican-las-metodologias-design-thinking-lean-agile-y-growth-hacking>.

Erap, S. (2012). *Gesture Based PC Interface with Kinect Sensor*, (Tesis de Maestría). University of Technology Tallinna Technikaülikool, Tallinn, Estonia

Flavell, J. (1982). *La Psicología evolutiva de Jean Piaget*. Barcelona: Ediciones Paidós.

Fernández – Domínguez, J.M. y Ramiro – Roca, E.P. (2015). *El Concepto Espacio en Educación Infantil*, (Tesis de Maestría). Universidad Jaume I, Castelló de la Plana, España

Fowler, M. y Beck, K. (2000). *Planning Extreme Programming*, Primera Edición, octubre 12, Addison Wesley.

Galeano – Echeverri, O.J. (2014). Consideraciones en el Desarrollo de Interfaces Naturales Gestuales, *Revista CINTEX*, 19(1), 183 – 193.

Gil, A. y Vida, T. (2007). *Los videojuegos*, Barcelona: UOC.

González – Sánchez, J.L. y Gutiérrez – Vela, F.L. (2014). Jugabilidad como medida de calidad en el desarrollo de videojuegos, *GEDES – Universidad de Granada, CoSECivi*.

González – Sánchez, J.L., Padilla – Zea, N., Gutiérrez, F.L. y Cabrera, M.J. (2008). De la Usabilidad a la Jugabilidad: Diseño de Videojuegos Centrado en el Jugador, *IX Congreso Internacional Interacción*, 1(1), 100 – 108.

González, J.L., Cabrera, M.J. y Gutiérrez, F.L., (2014). Diseño de videojuegos aplicados a la Educación Especial, *Universidad de Granada*

Gothelf, J. (2013). *Lean UX*, Primera Edición, marzo 2013, California, Estados Unidos: O'Reilly Media, Inc.

Gutiérrez, A. (2006). *Curso de Métodos de Investigación*, 2º Edición, Quito, Ecuador: Casa de la Cultura Ecuatoriana.

Gutiérrez - Ruiz, S. (2010). *Práctica Educativa y Creatividad en Educación Infantil*, (Tesis Doctoral). Universidad de Málaga, Málaga, España.

Holloway, G.E.T. (1982). *Concepción del espacio en el niño según Piaget*, Barcelona: Ediciones Paidós.

Ibarra – Esquer, J. E., González – Hernández, I. A., Pulido – Sandoval, N. G., Rodríguez – Alejo, E. A. y Sánchez – Vásquez, S. (2017). Desarrollo de interfaces naturales para aplicaciones educativas, *XIV Concurso de Creatividad Científica y Tecnología Universidad Autónoma de Baja California*, Baja California, México.

IEEE 1471-2000. (2000). IEEE Recommended Practice for Architecture Description of Software-Intensive Systems. ISO Press.

ISO 9000. (2005). Quality management systems – Fundamentals and vocabulary. ISO Press.

ISO 9241-110. (2018). Ergonomic of human-system interaction – Part 110: Dialogue principles. ISO / TC159/SC4, Edición 1. (04 – 2018)

ISO/IEC-25010. (2011). ISO/IEC 25010-3: Systems and software engineering: Software product quality and system quality in use models. ISO Press.

Jiménez, I. (2009). *Conocimiento del profesor para la enseñanza de las Matemáticas. Contribución de la metodología de resolución de problemas y las TIC al desarrollo de las competencias básicas*. Consejería de Educación de la Junta de Andalucía, 2008-09, PIV-036/08.

Kniberg, H. (2007). *Scrum y XP desde las trincheras. Cómo hacemos SCRUM*. Primera Edición, Estados Unidos de Norteamérica: C4Media Inc.

Kruchten, P. (1995). Planos Arquitectónicos: El Modelo de “4+1” Vistas de la Arquitectura del Software, *IEEE Software*, 12(6), 1 – 16.

Lagares – Barreiro, P. y Puerto – Albandoz, J. (2001). Población y muestra. Técnicas de muestreos. *MaMaEuSch Management Mathematics for European Schools*, Universidad de Sevilla.

Larriba – Gil, F., Martí – Carrillo, F. y Riba – Pi, E. (2013). *Human Interaction and Teleoperation, PRACTICUM: Kinect*, (Tesis de Maestría). UPC.

Leiva, F. (2007). *Nociones de Metodología de la Investigación Científica*, 4º Edición, Quito, Ecuador: Editorial Cámara Ecuatoriana del Libro – Núcleo de Pichincha.

Ley Orgánica Reformatoria a la Ley Orgánica de Educación Intercultural, (LOEI), *Registro Oficial N° 572*, Quito, Ecuador, 25 de agosto de 2015.

Loras – Navas, A. (2013). *Percepción de la actividad de un conductor mediante cámaras 3D*, (Tesis de Maestría). Universidad Carlos III, Madrid, España.

Lozada – Yanez, R. M., Rivera - Escriba, L. A. y Molina, F. T. (2014). Interfaces de Usuario Natural, *V Congreso Peruano de Investigación de Operaciones y de Sistemas - La Investigación de Operaciones y las TIC para una Diversidad Productiva Competitiva, At Lima - Perú*, 5, DOI: 10.13140/RG.2.1.5092.2324

Lozada, R., Molina, F. y Guffante, T. (2015). *Potencialidades de Kinect para la Educación*, Conferencia: I Congreso Internacional: Educación Contemporánea, Calidad Educativa y Buen Vivir, Riobamba, Ecuador, Volumen: 1, DOI: 10.13140/RG.2.1.1618.8247.

Macías, G., Quintero, R. (2011). Los videojuegos como una alternativa para el estudio y desarrollo de la orientación espacial, *Investigación en Educación Matemática XV. Ciudad Real: Sociedad Española de Investigación en Educación Matemática, SEIEM*, 405-416

Manresa – Yee, C., Ponsa, P., Varona, J. y Perales, F. (2010). User Experience to Improve the Usability of a Vision-based Interface, *Interacting with Computers, Special Issue on Inclusion and Interaction: Designing Interaction for Inclusive Populations*, 22(6), 594 – 605.

Martín – Crespo, C. y Salamanca – Castro, A.B. (2007). El muestreo en la Investigación cualitativa, *NURE Investigación*, 27(1), marzo-abril.

Microsoft (2017). Kinect for Windows Human Interface Guidelines v1.8.0. Recuperado el 4 de octubre de 2017 de <https://msdn.microsoft.com/en-us/library/jj663791.aspx>

Microsoft XBOX (2018). Componentes del sensor Kinect. Recuperado el 10 de julio de 2018 de <https://support.xbox.com/es-EC/xbox-360/kinect/kinect-sensor-components>

Ministerio de Educación (2016). *Educación General Básica PREPARATORIA*, 1º Edición. Quito, Ecuador: Santillana S.A.

Nielsen, J. (1994). Guerrilla HCI: Using Discount Usability Engineering to Penetrate the Intimidation Barrier. Recuperado el 13 de septiembre de 2018 de <http://www.nngroup.com/articles/guerrilla-hci/>

OpenKinect (2018). Bienvenido al Proyecto OpenKinect. Recuperado el 13 de julio de 2018 de https://openkinect.org/wiki/Main_Page/es

Orjuela – Duarte, A. y Rojas, M. (2008). Las Metodologías de Desarrollo Ágil como una Oportunidad para la Ingeniería del Software Educativo, *Revista Avances en Sistemas e Informática*, 5(2), 159 – 171.

Padilla – Zea, N., González – Sánchez, L., Gutiérrez, F. L., Cabrera, M. J. y Paderewski, P. (2009). Diseño de videojuegos colaborativos y educativos centrado en la Jugabilidad, *Revista Iberoamericana de Tecnologías de Aprendizaje IEEE-RITA*, 4(3), 191 – 198.

Parreño – Ruiz, D. F. (2016). *Elaboración y Aplicación de una guía didáctica con estrategias metodológicas activas "aprendo a ubicarme en el espacio", para desarrollar la inteligencia espacial en los niños de 3 a 4 años de la unidad educativa "simón rodríguez" parroquia Lican, provincia de Chimborazo período 2014- 2015*, (Tesis de Maestría). Universidad Nacional de Chimborazo, Riobamba, Ecuador.

Piaget, J. (1948). *La Representation de l'espacez dans l'enfant*. Preser Universitaire, Paris.

Piaget, J. e Inhelder, B. (2007). *Psicología del niño*, 17º Edición, Madrid, España: Ediciones Morata.

Prensky, M. (2001) *Digital game-based learning*. New York: McGraw-Hill.

Rivadeneira, S., Vilanova, G., Miranda, M. y Cruz, D. (2013). El modelado de requerimientos en las metodologías ágiles, *XV Workshop de Investigadores en Ciencias de la Computación*, Universidad Nacional de la Patagonia Austral, 1(1), 383 – 387.

Romero – Brest, G. (1977). Educación psicomotriz y retraso mental. *Psicomotricidad educación y movimiento*, España (1977), 7-22.

Romero – Moreno, G. (2004). *UML con Rational Rose*, Primera Edición, Lima Perú: Grupo Editorial Megabyte S.A.C.

Rosero, X. y Vaca, C. (2018). Metodología para evaluación de usabilidad del entorno de desarrollo integrado de arduino, *SATHIRI*, 13(1), 214 – 226.

Saiz, I. E. (1998). *La ubicación espacial en los primeros años de escolaridad*. Educación Matemática, 10(02), 71-87.

Sánchez – Benítez, G. (2008). *Las Estrategias de Aprendizaje a través del Componente Lúdico*, (Tesis de Maestría). Universidad de Alcalá, Alcalá de Henares, España.

Sommerville, I. (2011), *Ingeniería de Software*, 9º Edición, México D.F.: Pearson Educación.

Ubuntu (2018). Ubuntu for desktops. Recuperado el 12 de julio de 2018 de <https://www.ubuntu.com/desktop>

Valiño, G. (2002). La relación Juego y Escuela: aportes teóricos para su comprensión y promoción. *Revista Conceptos*. Boletín de la Universidad del Museo Argentino. 77(2), 15 – 25.

VPRZ. (2015). VPRZ Interactive display for kids on ZOO. Recuperado el 27 de marzo de 2018 de <https://github.com/StLuke/VRPZ>

ANEXOS

Anexo A: Cuestionario de Preguntas de la Entrevista

1. ¿Cuál es la estrategia didáctica que realiza para que los estudiantes aprendan?

2. ¿Cómo atrae la atención de sus estudiantes?

3. ¿Cuál es la importancia de que las niñas cuenten con un aprendizaje previo?

4. ¿Cuál es la actividad que utiliza para que participen los estudiantes en clase?

5. ¿Cuál es el tiempo que debería tomarse para que las destrezas de ubicación espacial sean desarrolladas?

6. ¿Qué tipo de materiales usa en sus clases?

7. La edad de las niñas es importante para el ingreso al primer año básica.

8. Tiene alguna dificultad al momento de enseñar la ubicación espacial.

9. Existe algún requisito para el ingreso del primer año.

10. Utilizan alguna herramienta para poder reforzar el aprendizaje de ubicación espacial.

11. Utilizan algún instintivo para diferenciar la lateralidad.

12. Hay algún inconveniente cuando una niña es zurda.

13. Utilizan medios tecnológicos para enseñar en el aula de clase.

14. ¿Cuál es el medio tecnológico que utilizan para el reforzamiento de enseñanza de ubicación espacial?

15. Si no utilizan medios tecnológicos hay alguna herramienta que usen en la enseñanza de ubicación espacial.

16. En el aula que trabajan es óptimo el espacio para trabajar con las niñas.

17. Si tiene espacio suficiente para trabajar y realizar actividades en el aula.

18. El número de estudiantes para los primeros de básica para mejorar el aprendizaje debe ser:

19. Les gustaría utilizar medios electrónicos para reforzar su enseñanza a los niños.

20. Creen que los colores influyen en el estado de animo de las estudiantes.

21. Con su experiencia como docente que colores cree usted que son los más adecuados para usar en una interfaz gráfica.

22. ¿Qué colores cree usted que no se debería considerar en un entorno gráfico?

23. La altura de las estudiantes promedio es.

24. ¿Cuántas niñas de otra etnia tiene en su aula de clases?

25. Creen que la educación debe ser en grupo o unipersonal.

ANEXO B: Manual de Instalación de los Prerrequisitos para la Implementación del Videojuego “UBIC”

1. Abrir una terminal y ejecutar los siguientes comandos

Ejecutar el siguiente comando para abrir el administrador:

sudo su

contraseña 1234admin

sudo apt-get update

sudo apt-get upgrade

2. Instalar las dependencias necesarias

```
sudo apt-get install git-core cmake freeglut3-dev pkg-config build-essential libxmu-dev  
libxi-dev libusb-1.0-0-dev
```

3. Clonamos la librería del repositorio

```
git clone git://github.com/OpenKinect/libfreenect.git
```

4. Instalar el libfreenect

cd libfreenect

mkdir build

cd build

cmake -L ..

make

sudo make install

sudo ldconfig /usr/local/lib64/

5. Para usar Kinect como usuario no root haga lo siguiente

sudo adduser \$USER video

sudo adduser \$USER plugdev

6. También crea un archivo con reglas para el administrador de dispositivos Linux – nano es editor de texto que utiliza Ubuntu

```
sudo nano /etc/udev/rules.d/51-kinect.rules
```

A continuación, copie y pegue lo siguiente y guárdelo

```
# ATTR{product}=="Xbox NUI Motor"
SUBSYSTEM== "usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02b0",
MODE="0666"

# ATTR{product}=="Xbox NUI Audio"
SUBSYSTEM== "usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02ad",
MODE="0666"

# ATTR{product}=="Xbox NUI Camera"
SUBSYSTEM== "usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02ae",
MODE="0666"

# ATTR{product}=="Xbox NUI Motor"
SUBSYSTEM== "usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02c2",
MODE="0666"

# ATTR{product}=="Xbox NUI Motor"
SUBSYSTEM== "usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02be",
MODE="0666"

# ATTR{product}=="Xbox NUI Motor"
SUBSYSTEM== "usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02bf",
MODE="0666"
```

7. Desconéctese y vuelva a iniciarse. Ejecute el siguiente comando en un terminal para probar si libfreenect está instalado correctamente.

freenect-glview

Esto debe hacer que aparezca una ventana que muestra la profundidad y las imágenes RGB. Al presionar 'w' en el teclado, el kinect se inclina hacia arriba y al presionar 'x' el kinect se inclina hacia abajo. Hay varias otras opciones de control que se enumeran en la terminal cuando se ejecuta "freenect-glview"

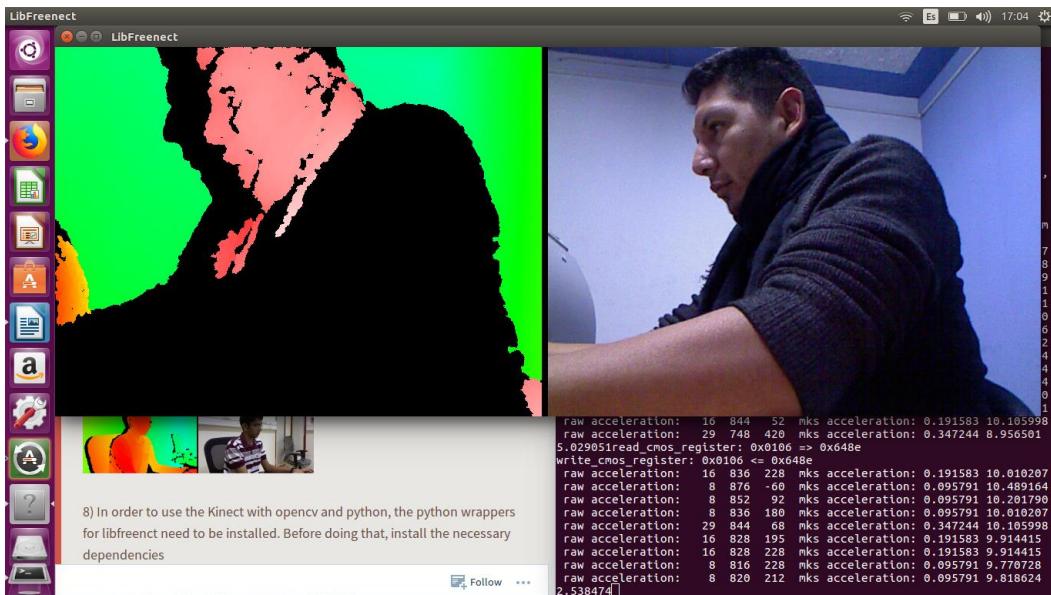
Ejecución del comando freenect-glview

```

root@frportatil-Satellite-L55-A:~$ sudo su
[sudo] password for frportatil:
freenect: no se encontró la orden
Kinect camera test
Number of devices found: 1
Found sibling device [same parent]
GL thread
[Stream 70] Negotiated packet size 1920
write_register: 0x0105 <= 0x00
write_register: 0x0006 <= 0x00
write_register: 0x0012 <= 0x03
write_register: 0x0013 <= 0x01
write_register: 0x0014 <= 0x1e
write_register: 0x0006 <= 0x02
write_register: 0x0017 <= 0x00
[Stream 80] Negotiated packet size 1920
write_register: 0x000c <= 0x00
write_register: 0x000d <= 0x01
write_register: 0x000c <= 0x0e
write_register: 0x0005 <= 0x01
write_register: 0x0047 <= 0x00
'c' - tilt up, 'x' - level, 'y' - tilt down, '0'-'6' - select LED mode,
'f' - change video format, 'm' - mirror video, 'o' - rotate video with
accelerometer
'e' - auto exposure, 'b' - white balance, 'r' - raw color, 'n' - near mode (K4W only)
raw acceleration: -33 780 -190 mks acceleration: -0.395140 9.339667
raw acceleration: 16 844 8 mks acceleration: 0.191583 10.105998
raw acceleration: -31 844 84 mks acceleration: -0.371192 10.10599
raw acceleration: 32 860 -12 mks acceleration: 0.383166 10.297581
raw acceleration: 29 860 -12 mks acceleration: 0.347244 10.297581
raw acceleration: 25 850 -20 mks acceleration: 0.383166 10.120946
raw acceleration: 32 845 -12 mks acceleration: 0.383166 10.117972
raw acceleration: 32 848 -12 mks acceleration: 0.383166 10.153894
raw acceleration: 29 848 -12 mks acceleration: 0.347244 10.153894
raw acceleration: 16 848 -28 mks acceleration: 0.191583 10.153894
raw acceleration: 4 852 -12 mks acceleration: 0.047896 10.201790

```

Resultado de la ejecución del comando freenect-glview



Instalación de las dependencias

```

sudo apt install zlib1g-dev mesa-common-dev libgl1-mesa-dev libglu1-mesa-dev
sudo apt install git-core cmake freeglut3-dev pkg-config build-essential libxmu-dev \
libxi-dev libusb-1.0-0-dev doxygen graphviz mono-complete

```

```

Terminal
install and setup kinect xbox360 1473 [ubuntu 16.04.64bit / debian 64bit / windows]
ff install and setup kinect

http://www.30paperclips.net/programming/kinect-on-ubuntu-with-opencv/
+ zlbig-dev ya está en su versión más reciente (1:1.2.8.dfsg-2ubuntu4.1)
+ Fijado zlbig-dev como instalado manualmente.
Se actualizarán los siguientes paquetes:
  libgl1-mesa-dev libgl1-mesa-glx libglapi-mesa mesa-common-dev
  4 actualizados, 0 nuevos se instalarán, 0 para eliminar y 162 no actualizados.
Se necesita descargar 682 kB de archivos.
Se utilizarán 26,6 kB de espacio de disco adicional después de esta operación.
Descargar http://cc.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libgl1-mesa-dev amd64 17.2.8-0ubuntu0-16.04.1 [4,456 kB]
Descargar http://cc.archive.ubuntu.com/ubuntu xenial-updates/main amd64 mesa-common-dev amd64 17.2.8-0ubuntu0-16.04.1 [525 kB]
Descargar http://cc.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libgl1-mesa-glx amd64 17.2.8-0ubuntu0-16.04.1 [130 kB]
Descargar http://cc.archive.ubuntu.com/ubuntu xenial-updates/main amd64 libglapi-mesa amd64 17.2.8-0ubuntu0-16.04.1 [22,8 kB]
Descargados 682 kB en 2s (228 kB/s)
(Leyendo la base de datos ... 224175 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../libgl1-mesa-dev_17.2.8-0ubuntu0-16.04.1_and64.deb ...
Desempaquetando libgl1-mesa-dev:amd64 (17.2.8-0ubuntu0-16.04.1) sobre (17.2.4-0ubuntu1-16.04.4) ...
Preparando para desempaquetar .../libgl1-mesa-glx_17.2.8-0ubuntu0-16.04.1_and64.deb ...
Desempaquetando libgl1-mesa-glx:amd64 (17.2.8-0ubuntu0-16.04.1) sobre (17.2.4-0ubuntu1-16.04.4) ...
Preparando para desempaquetar .../libglapi-mesa_17.2.8-0ubuntu0-16.04.1_and64.deb ...
Desempaquetando libglapi-mesa:amd64 (17.2.8-0ubuntu0-16.04.1) sobre (17.2.4-0ubuntu1-16.04.4) ...
Procesando disparadores para llbc-bin (2.23-0ubuntu10) ...
Configurando mesa-common-dev:amd64 (17.2.8-0ubuntu0-16.04.1) ...
Configurando libglapi-mesa:amd64 (17.2.8-0ubuntu0-16.04.1) ...
[########################################.....]

```

Instalación del freenect driver

Para la instalación del driver freenect, escribimos en la consola de Ubuntu el siguiente comando:

```
sudo apt install freenect python-freenect
```

Comando para probar el freenect

freenect-glview o freenect-cppview

Comando para descargar opencv

```
sudo apt install python-opencv
```

Aparece la pregunta s/n y escribimos s

Comando wget : Descarga en la línea de comandos

```
wget
```

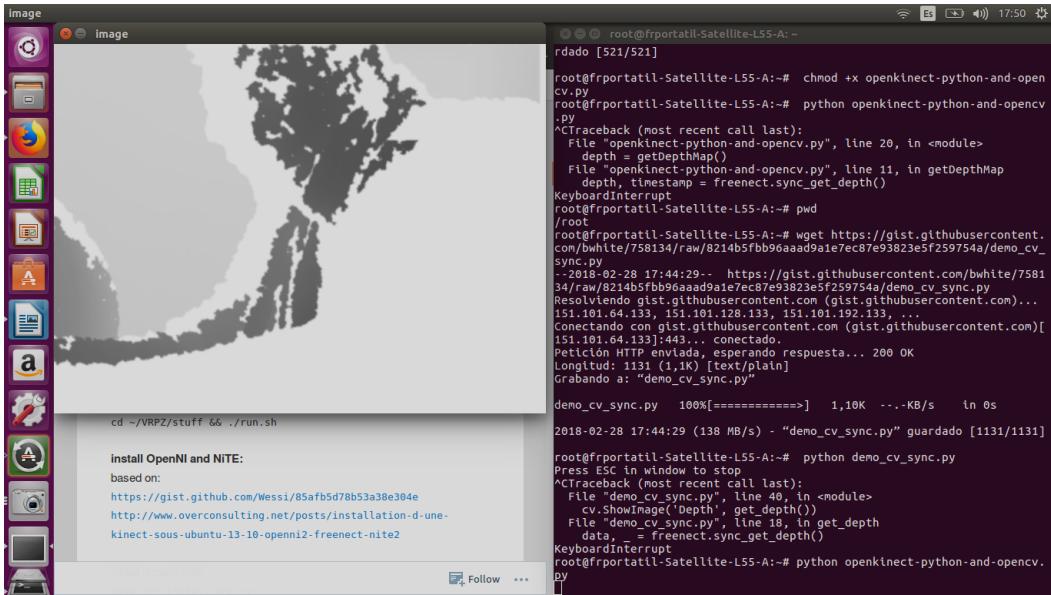
```
https://gist.githubusercontent.com/anonymous/126c50cf9233ade3c79b35e4e22f7629/raw/62bea505b6d9d24079903bf210c873f7d84e4cd6/openkinect-python-and-opencv.py
```

Comando chmod: asigna permisos de descarga

```
chmod +x openkinect-python-and-opencv.py
```

Comando para ejecuta la aplicación que funciona con opencv

```
python openkinect-python-and-opencv.py
```



Instalación de pygame

Para la instalación de la librería pygame escribimos en la consola de Ubuntu el siguiente comando:

```
sudo apt install python-pygame
```

En el trascurso de la instalación debe aparecer un mensaje de s/n, donde escribimos “s” para continuar con la instalación.

Instalamos xlib

Para la instalación de la librería Xlib escribimos en la consola de Ubuntu el siguiente comando:

```
sudo apt-get install python-xlib
```

Ejecutar el código de reconocimiento de la mano

Abrimos en el navegador de internet el código para el reconocimiento del mouse con la mano.

<http://code.activestate.com/recipes/578104-openkinect-mouse-control-using-python/>

Abrimos una nueva terminal en ingresamos a cada una de las carpetas con cd libfreenect, cd wrappers, cd Python.

Se crea el archivo:

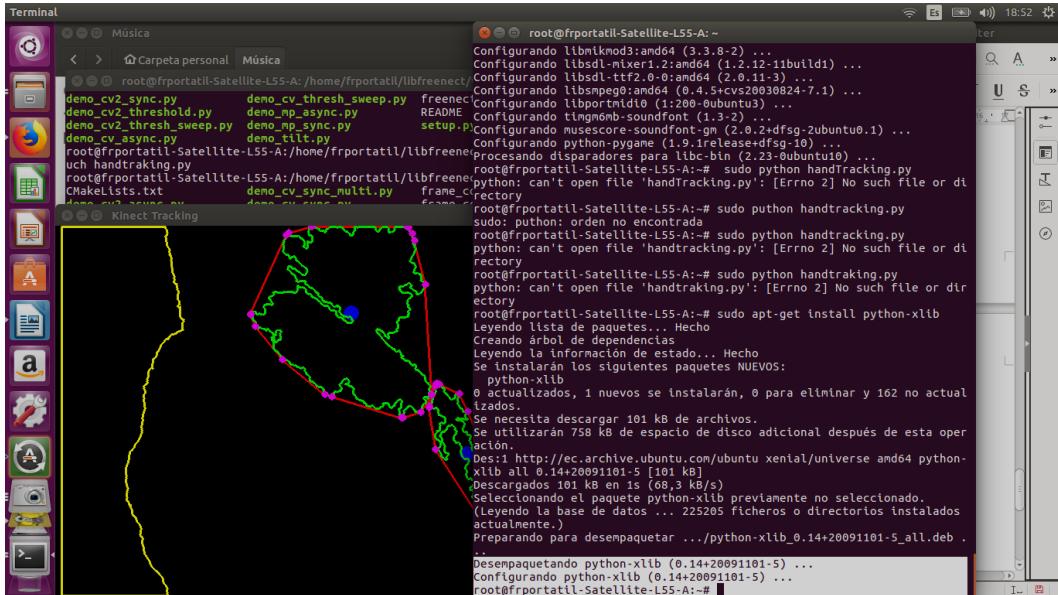
```
libfreenect/wrappers/python/ touch handtracking.py
```

Editamos, copiamos el codigo del sitio y pegamos, salimos con ctrl + x.

```
libfreenect/wrappers/python/ nano handtracking.py
```

ejecutamos con el siguiente comando

```
sudo python handtracking.py
```



8. Para utilizar Kinect con opencv y python, es necesario instalar los wrappers de Python para libfreenect.

Antes de hacerlo, instale las dependencias necesarias

```
sudo apt-get install cython
```

```
sudo apt-get install python-dev
```

```
sudo apt-get install python-numpy
```

aparece un mensaje s/n escribir s

9. Ir al directorio/libfreenect/wrappers/python y correr el siguiente comando:

Escribir lo siguiente:

```
cd ..
```

```
cd home
```

```
cd frportatil este es el nombre del usuario
```

```
cd libfreenect/wrappers/python
```

```
sudo python setup.py install
```

10. crear un nuevo archivo

crear el archivo

```
libfreenect/wrappers/python/ touch kinect_test.py
```

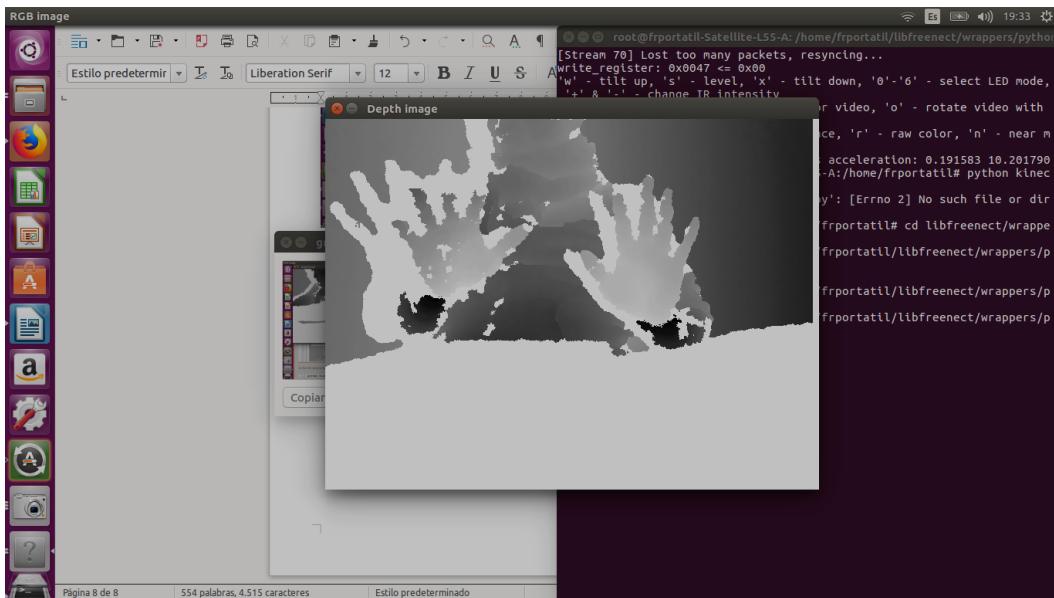
Editamos, copiamos el código del paso 10 del sitio y pegamos, salimos con ctrl + x.

```
libfreenect/wrappers/python/ nano kinect_test.py
```

ejecutamos con el siguiente comando

```
sudo python kinect_test.py
```

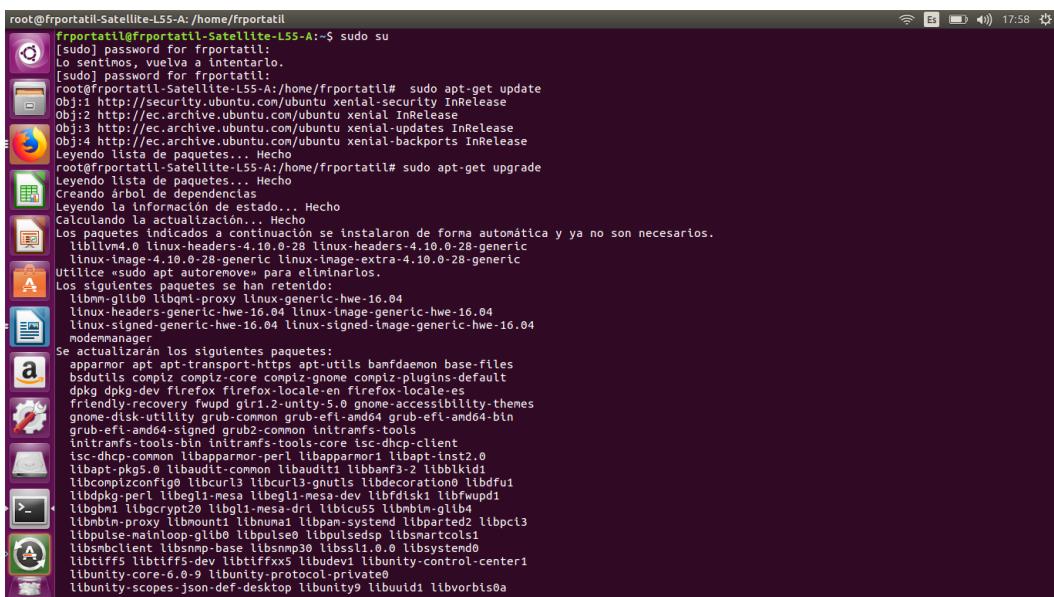
debe aparecer la siguiente ventana



Open CV

Paso 1. Instalar Open CV dependencias en Ubuntu 16.04

```
sudo apt-get update
```



```
sudo apt-get upgrade
```

aparece un mensaje s /n, se debe presionar el s

La instalación tiene un tiempo de 10 a 15 minutos

A continuación, instalaremos algunas herramientas de desarrollador:

```
sudo apt-get install build-essential cmake pkg-config
```

El paquete pkg-config ya está instalado en su sistema, pero asegúrese de incluirlo en el comando apt-get anterior, por si acaso. El programa cmake se usa para configurar automáticamente nuestra compilación OpenCV.

OpenCV es una biblioteca de procesamiento de imágenes y visión por computadora. Por lo tanto, OpenCV necesita poder cargar varios formatos de archivos de imagen desde el disco como JPEG, PNG, TIFF, etc. Para cargar estas imágenes desde el disco, OpenCV realmente llama a otras bibliotecas de E / S de imagen que realmente facilitan la carga y decodificación proceso. Instalamos los necesarios a continuación:

```
sudo apt-get install libjpeg8-dev libtiff5-dev libjasper-dev libpng12-dev
```

Bueno, ahora tenemos bibliotecas para cargar imágenes desde el disco, pero ¿y el video? Use los siguientes comandos para instalar los paquetes utilizados para procesar las transmisiones de video y acceder a las imágenes de las cámaras:

```
sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev
```

```
sudo apt-get install libxvidcore-dev libx264-dev
```

OpenCV viene con un conjunto muy limitado de herramientas GUI. Estas herramientas GUI nos permiten mostrar una imagen en nuestra pantalla (cv2.imshow), esperar / registrar pulsaciones de teclas (cv2.waitKey), rastrear eventos del mouse (cv2.setMouseCallback) y crear elementos simples de la GUI, como controles deslizantes y barras de seguimiento. De nuevo, no debe esperar crear aplicaciones de GUI completas con OpenCV; estas son solo herramientas simples que le permiten depurar su código y crear aplicaciones muy simples.

Internamente, el nombre del módulo que maneja las operaciones de la GUI de OpenCV es highgui. El módulo highgui se basa en la biblioteca GTK, que debe instalar utilizando el siguiente comando:

```
sudo apt-get install libgtk-3-dev
```

A continuación, instalamos bibliotecas que se utilizan para optimizar varias funcionalidades dentro de OpenCV, como operaciones de matriz:

```
sudo apt-get install libatlas-base-dev gfortran
```

Concluiremos el Paso # 1 instalando los encabezados de desarrollo y las bibliotecas de Python para Python 2.7 y Python 3.5 (de esa manera usted tiene ambos):

```
sudo apt-get install python2.7-dev python3.5-dev
```

Nota: Si no instala los encabezados de desarrollo Python y la biblioteca estática, se encontrará con problemas durante el Paso 4, donde ejecutaremos cmake para configurar nuestra compilación. Si estos encabezados no están instalados, entonces el comando cmake no podrá determinar automáticamente los valores correctos del intérprete de Python y las bibliotecas de Python. En resumen, el resultado de esta sección se verá "vacío" y no podrá construir los enlaces de Python. Cuando llegue al Paso 4, tómese el tiempo para comparar su salida del comando con el mío.

Paso 2: Download the OpenCV source

```
cd ~
```

```
wget -O opencv.zip https://github.com/Itseez/opencv/archive/3.1.0.zip  
unzip opencv.zip
```

La descarga se demora de 5 a 10 minutos

The screenshot shows a terminal window with the following command history:

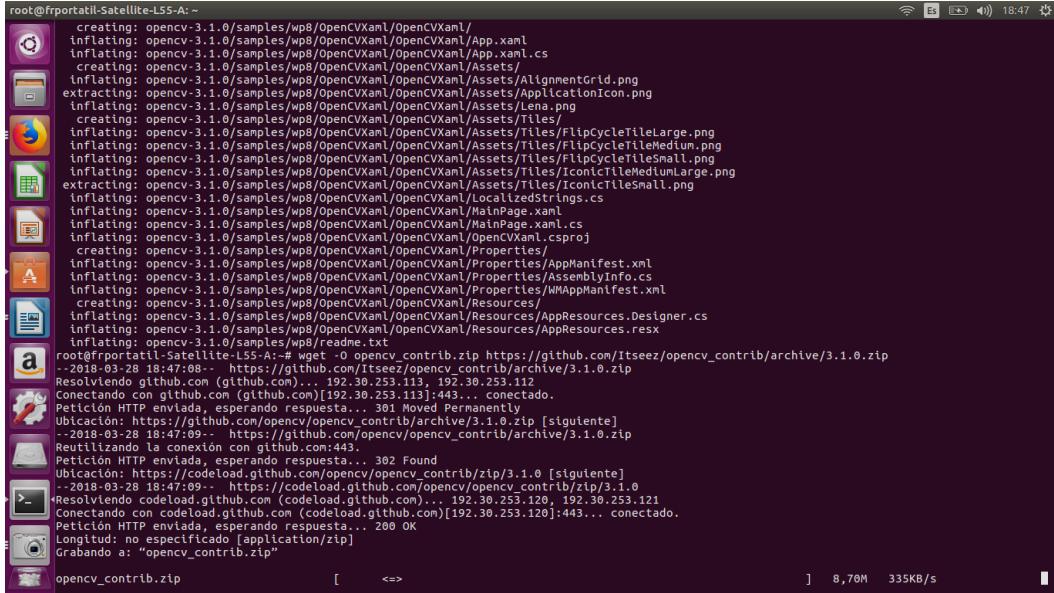
```
root@frportatil-Satellite-L55-A:~  
Utilice «sudo apt autoremove» para eliminarlos.  
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 8 no actualizados.  
root@frportatil-Satellite-L55-A:/home/frportatil# sudo apt-get install libatlas-base-dev gfortran  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
gfortran ya está en su versión más reciente (4:5.3.1-1ubuntu1).  
libatlas-base-dev ya está en su versión más reciente (3.10.2-9).  
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.  
libblkdev4.0 linux-headers-4.10.0-28 linux-image-4.10.0-28-generic  
linux-image-4.10.0-28-generic linux-image-extra-4.10.0-28-generic  
Utilice «sudo apt autoremove» para eliminarlos.  
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 8 no actualizados.  
root@frportatil-Satellite-L55-A:/home/frportatil# sudo apt-get install python2.7-dev python3.5-dev  
Creando lista de paquetes... Hecho  
Leyendo la información de estado... Hecho  
python2.7-dev ya está en su versión más reciente (2.7.12-1ubuntu0-16.04.3).  
python3.5-dev ya está en su versión más reciente (3.5.2-2ubuntu0-16.04.4).  
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.  
libblkdev4.0 linux-headers-4.10.0-28 linux-headers-4.10.0-28-generic  
linux-image-4.10.0-28-generic linux-image-extra-4.10.0-28-generic  
Utilice «sudo apt autoremove» para eliminarlos.  
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 8 no actualizados.  
root@frportatil-Satellite-L55-A:/home/frportatil# cd ~  
root@frportatil-Satellite-L55-A:~# wget -O opencv.zip https://github.com/Itseez/opencv/archive/3.1.0.zip  
2018-03-28 18:38:03 -> https://github.com/Itseez/opencv/archive/3.1.0.zip  
Resolviendo github.com (github.com)[192.30.255.11]:443... Conectado.  
Petición HTTP enviada, esperando respuesta... 301 Moved Permanently  
Ubicación: https://github.com/opencv/opencv/archive/3.1.0.zip [siguiente]  
--2018-03-28 18:38:03 -> https://github.com/opencv/opencv/archive/3.1.0.zip  
Reutilizando la conexión con github.com:443.  
Petición HTTP enviada, esperando respuesta... 302 Found  
Ubicación: https://codeload.github.com/opencv/opencv/zip/3.1.0 [siguiente]  
--2018-03-28 18:38:04 -> https://codeload.github.com/opencv/opencv/zip/3.1.0  
Resolviendo codeload.github.com (codeload.github.com)[192.30.253.121]:443... Conectado.  
Petición HTTP enviada, esperando respuesta... 200 OK  
Longitud: no especificado [application/zip]  
Grabando a: "opencv.zip"  
[  ] 6,86M 494KB/s
```

Cuando se lanzan nuevas versiones de OpenCV, puede consultar el OpenCV GitHub oficial y descargar la última versión cambiando el número de versión de .zip.

Sin embargo, aún no hemos terminado de descargar el código fuente; también necesitamos el repositorio opencv_contrib:

```
wget -O opencv_contrib.zip https://github.com/Itseez/opencv_contrib/archive/3.1.0.zip
```

se demora en descargar 5 min



```
root@frportatil-Satellite-L55-A: ~
  creating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/app.xaml
  creating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/app.xaml.cs
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/app.xaml.cs
  extracting: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/ApplicationIcon.png
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Lena.png
  creating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/FlipCycleTileLarge.png
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/FlipCycleTileMedium.png
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/FlipCycleTileSmall.png
  extracting: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/IconicTileMediumLarge.png
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/IconicTileSmall.png
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/IconicTitleMediumLarge.png
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/IconicTitleSmall.png
  extracting: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/IconicTitleMediumLarge.png
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/IconicTitleSmall.png
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/IconicTitleMediumLarge.png
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Assets/Tiles/IconicTitleSmall.png
  creating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Properties/
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Properties/AppManifest.xml
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Properties/AssemblyInfo.cs
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Properties/WAppManifest.xml
  creating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Resources/
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Resources/AppResources.Designer.cs
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Resources/AppResources.resx
  inflating: opencv-3.1.0/samples/wp8/OpenCVXaml/OpenCVXaml/Resources/readme.txt
root@frportatil-Satellite-L55-A: # wget -O opencv_contrib.zip https://github.com/Itseez/opencv_contrib/archive/3.1.0.zip
--2018-03-28 18:47:08-- https://github.com/Itseez/opencv_contrib/archive/3.1.0.zip
Resolving github.com (github.com)... 192.30.253.113, 192.30.253.112
Conectando con github.com (github.com)[192.30.253.113]:443... conectado.
Petición HTTP enviada, esperando respuesta... 301 Moved Permanently
Ubicación: https://codeload.github.com/opencv/opencv_contrib/archive/3.1.0.zip [siguiente]
--2018-03-28 18:47:09-- https://codeload.github.com/opencv/opencv_contrib/archive/3.1.0.zip
Reutilizando la conexión con github.com:443...
Petición HTTP enviada, esperando respuesta... 302 Found
Ubicación: https://codeload.github.com/opencv/opencv_contrib/zip/3.1.0 [siguiente]
--2018-03-28 18:47:09-- https://codeload.github.com/opencv/opencv_contrib/zip/3.1.0
Resolviendo codeload.github.com (codeload.github.com)... 192.30.253.120, 192.30.253.121
Conectando con codeload.github.com (codeload.github.com)[192.30.253.120]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: no especificado [application/zip]
Grabando a: "opencv_contrib.zip"

[          =>          ] 8,70M 335KB/s
```

```
unzip opencv_contrib.zip
```

¿Por qué nos molestamos en descargar también el repositorio contrib?

Bueno, queremos que la instalación completa de OpenCV 3 tenga acceso a las características como SIFT y SURF. En OpenCV 2.4, SIFT y SURF se incluyeron en la instalación predeterminada de OpenCV. Sin embargo, con el lanzamiento de OpenCV 3+, estos paquetes se han movido a contrib, que alberga ya sea (1) módulos que están actualmente en desarrollo o (2) módulos que están marcados como "no libres" (es decir, patentados).

Nota: Es posible que necesite expandir los comandos anteriores utilizando el botón "<=>" durante su copia y pegado. El .zip en 3.1.0.zip puede cortarse en ventanas de navegador más pequeñas. Para su comodidad, he incluido la URL completa tanto del archivo opencv como del archivo opencv_contrib a continuación:

<https://github.com/Itseez/opencv/archive/3.1.0.zip>

https://github.com/Itseez/opencv_contrib/archive/3.1.0.zip

Cabe mencionar que tanto las versiones de opencv como de opencv_contrib deben ser las mismas (en este caso, 3.1.0). Si los números de las versiones no coinciden, fácilmente

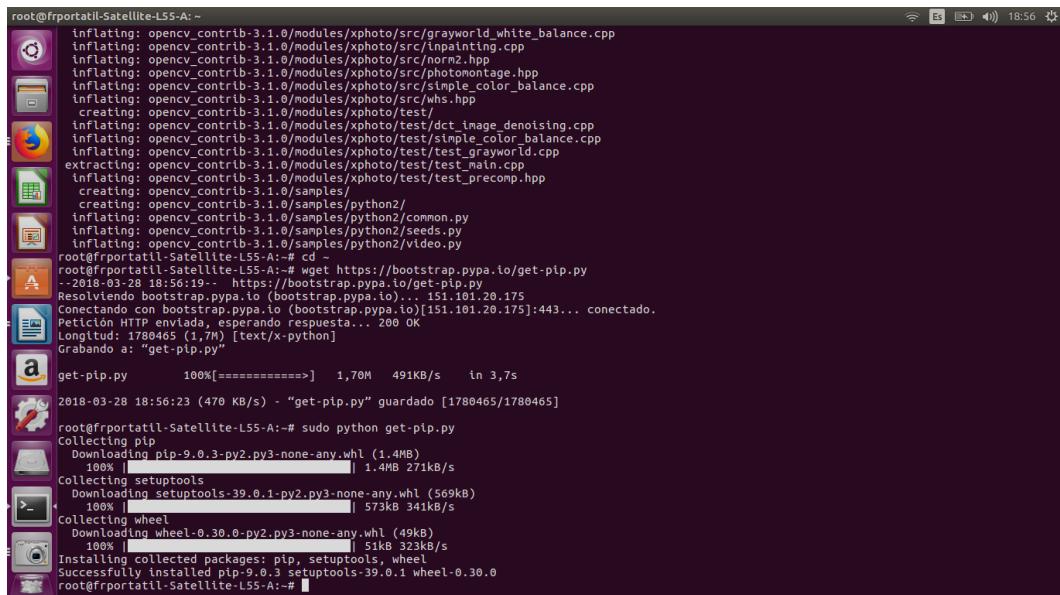
podría encontrarse con errores de tiempo de compilación (o peor, errores de tiempo de ejecución que son casi imposibles de depurar).

Paso # 3: Configura tu entorno de Python - Python 2.7 o Python 3

Instalar Python 2.7

Ahora estamos listos para comenzar a configurar nuestro entorno de desarrollo Python para la compilación. El primer paso es instalar pip, un administrador de paquetes de Python:

```
cd ~  
wget https://bootstrap.pypa.io/get-pip.py  
sudo python get-pip.py
```



```
root@frportatil-Satellite-L55-A: ~  
Inflating: opencv_contrib-3.1.0/modules/xphoto/src/grayworld_white_balance.cpp  
Inflating: opencv_contrib-3.1.0/modules/xphoto/src/inpainting.cpp  
Inflating: opencv_contrib-3.1.0/modules/xphoto/src/norm2.hpp  
Inflating: opencv_contrib-3.1.0/modules/xphoto/src/photomontage.hpp  
Inflating: opencv_contrib-3.1.0/modules/xphoto/src/simple_color_balance.cpp  
Inflating: opencv_contrib-3.1.0/modules/xphoto/src/whs.hpp  
creating opencv_contrib-3.1.0/modules/xphoto/test/  
Inflating: opencv_contrib-3.1.0/modules/xphoto/test/dct_image_denoising.cpp  
Inflating: opencv_contrib-3.1.0/modules/xphoto/test/simple_color_balance.cpp  
Inflating: opencv_contrib-3.1.0/modules/xphoto/test/test_grayworld.hpp  
extracting: opencv_contrib-3.1.0/modules/xphoto/test/test_main.cpp  
Inflating: opencv_contrib-3.1.0/modules/xphoto/test/test_precomp.hpp  
creating: opencv_contrib-3.1.0/samples/  
creating: opencv_contrib-3.1.0/samples/python2/  
Inflating: opencv_contrib-3.1.0/samples/python2/common.py  
Inflating: opencv_contrib-3.1.0/samples/python2/seeds.py  
Inflating: opencv_contrib-3.1.0/samples/python2/video.py  
root@frportatil-Satellite-L55-A: # cd ~  
root@frportatil-Satellite-L55-A: # wget https://bootstrap.pypa.io/get-pip.py  
--2018-03-28 18:56:19-- https://bootstrap.pypa.io/get-pip.py  
Resolviendo bootstrap.pypa.io (bootstrap.pypa.io)... 151.101.20.175  
Conectando con bootstrap.pypa.io (bootstrap.pypa.io)[151.101.20.175]:443... conectado.  
Petición HTTP enviada, esperando respuesta... 200 OK  
Longitud: 1780465 (1,7M) [text/x-python]  
Grabando a: "get-pip.py"  
get-pip.py      100%[=====] 1,70M  491KB/s   in 3,7s  
2018-03-28 18:56:23 (470 KB/s) - "get-pip.py" guardado [1780465/1780465]  
root@frportatil-Satellite-L55-A: # sudo python get-pip.py  
Collecting pip  
  Downloading pip-9.0.3-py2.py3-none-any.whl (1.4MB)  
    100% |████████████████████████████████| 1.4MB 271kB/s  
Collecting setuptools  
  Downloading setuptools-39.0.1-py2.py3-none-any.whl (569kB)  
    100% |████████████████████████████████| 573kB 341kB/s  
Collecting wheel  
  Downloading wheel-0.30.0-py2.py3-none-any.whl (49kB)  
    100% |████████████████████████████████| 51kB 323kB/s  
Installing collected packages: pip, setuptools, wheel  
Successfully installed pip-9.0.3 setuptools-39.0.1 wheel-0.30.0  
root@frportatil-Satellite-L55-A: #
```

Los paquetes virtualenv y virtualenvwrapper de Python le permiten crear entornos de Python independientes e independientes para cada proyecto en el que se esté trabajando.

El uso de estos paquetes le permite resolver el "Proyecto X depende de la versión 1.x, pero el Proyecto Y necesita un dilema 4.x.

Si bien es cierto que puede instalar OpenCV con enlaces de Python sin entornos virtuales de Python, es recomendable utilizarlos.

```
sudo pip install virtualenv virtualenvwrapper  
sudo rm -rf ~/get-pip.py ~/.cache/pip
```

```
echo -e "\n# virtualenv and virtualenvwrapper" >> ~/.bashrc
echo "export WORKON_HOME=$HOME/.virtualenvs" >> ~/.bashrc
echo "source /usr/local/bin/virtualenvwrapper.sh" >> ~/.bashrc
```

Después de editar nuestro archivo ~ / .bashrc, tenemos que volver a cargar los cambios:

```
source ~/.bashrc
```

Nota: Llamar a la fuente en .bashrc solo tiene que hacerse una vez para nuestra sesión de shell actual. Cada vez que abrimos una nueva terminal, los contenidos de .bashrc se ejecutarán automáticamente (incluidas nuestras actualizaciones).

Ahora que hemos instalado virtualenv y virtualenvwrapper, el siguiente paso es crear realmente el entorno virtual de Python; lo hacemos utilizando el comando mkvirtualenv.

Pero antes de ejecutar este comando, debe hacer una elección: ¿Desea usar Python 2.7 o Python 3?

El resultado de su elección determinará qué comando ejecutará en la siguiente sección.
Creando su entorno virtual de Python

Si decide usar Python 2.7, use el siguiente comando para crear un entorno virtual de Python 2.7:

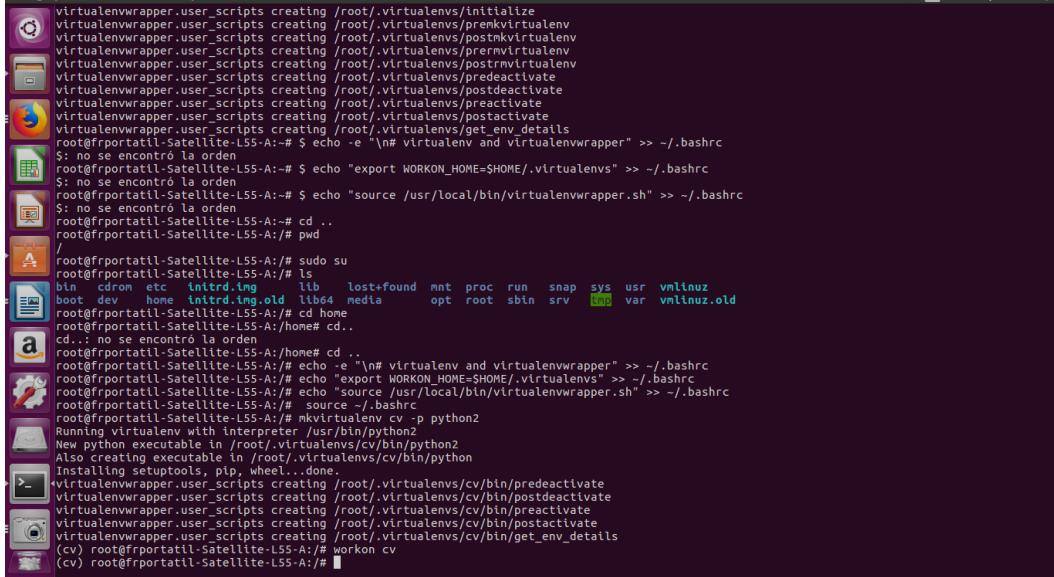
```
mkvirtualenv cv -p python2
```

Verificando que estás en el entorno virtual "cv"

Si alguna vez reinicias tu sistema Ubuntu; cerrar sesión y volver a iniciar sesión; o abra una nueva terminal, necesitará usar el comando workon para volver a acceder a su entorno virtual cv. Un ejemplo del comando workon sigue:

```
workon cv
```

Para validar que se encuentra en el entorno virtual de cv, simplemente examine su línea de comando: si ve el texto (cv) precediendo su solicitud, entonces se encuentra en el entorno virtual de cv:
aparece la siguiente ventana en el cual inicia con (cv)



```

root@frportatil-Satellite-L55-A:/
virtualenvwrapper.user_scripts creating /root/.virtualenvs/initialize
virtualenvwrapper.user_scripts creating /root/.virtualenvs/premkvirtualenv
virtualenvwrapper.user_scripts creating /root/.virtualenvs/postmkvirtualenv
virtualenvwrapper.user_scripts creating /root/.virtualenvs/prermvirtualenv
virtualenvwrapper.user_scripts creating /root/.virtualenvs/postrmvirtualenv
virtualenvwrapper.user_scripts creating /root/.virtualenvs/predeactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/postdeactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/preactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/postactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/get_env_details
root@frportatil-Satellite-L55-A:# $ echo -e "\n# virtualenv and virtualenvwrapper" >> ~/.bashrc
$: no se encontró la orden
root@frportatil-Satellite-L55-A:# $ echo "export WORKON_HOME=$HOME/.virtualenvs" >> ~/.bashrc
$: no se encontró la orden
root@frportatil-Satellite-L55-A:# $ echo "source /usr/local/bin/virtualenvwrapper.sh" >> ~/.bashrc
$: no se encontró la orden
root@frportatil-Satellite-L55-A:# cd ..
root@frportatil-Satellite-L55-A:# pwd
root@frportatil-Satellite-L55-A:/#
root@frportatil-Satellite-L55-A:# sudo su
root@frportatil-Satellite-L55-A:# ls
bin  cdrom  etc  intrd.img  lib  lost+found  mnt  proc  run  snap  sys  usr  vmlinuz
boot  dev  home  intrd.img.old  lib64  media  opt  root  sbin  srv  tmp  var  vmlinuz.old
root@frportatil-Satellite-L55-A:# cd home
root@frportatil-Satellite-L55-A:/home# cd ..
cd..: no se encontró la orden
root@frportatil-Satellite-L55-A:/home# cd ..
root@frportatil-Satellite-L55-A:/home# cd ..
root@frportatil-Satellite-L55-A:# echo -e "\n# virtualenv and virtualenvwrapper" >> ~/.bashrc
root@frportatil-Satellite-L55-A:# echo "export WORKON_HOME=$HOME/.virtualenvs" >> ~/.bashrc
root@frportatil-Satellite-L55-A:# echo "source /usr/local/bin/virtualenvwrapper.sh" >> ~/.bashrc
root@frportatil-Satellite-L55-A:# mkvirtualenv cv -p python2
Running virtualenv with interpreter /usr/bin/python2
New python executable in /root/.virtualenvs/cv/bin/python2
Also creating executable in /root/.virtualenvs/cv/bin/python
Installing setuptools, pip, wheel...done.
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/predeactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/postdeactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/preactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/postactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/get_env_details
(cv) root@frportatil-Satellite-L55-A:/# workon cv
(cv) root@frportatil-Satellite-L55-A:/# pip install numpy
Collecting numpy
  Downloading numpy-1.14.2-cp27-cp27mu-manylinux1_x86_64.whl (12.1MB)
    14% |███████████| 1.8MB 523kB/s eta 0:00:20

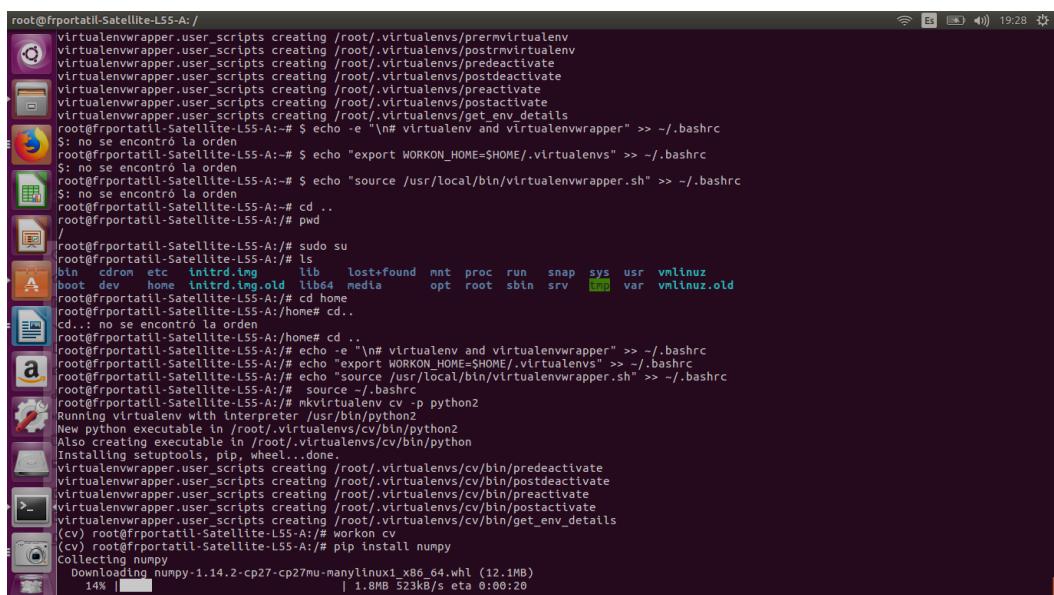
```

Nota: De lo contrario, si no ve el texto cv, entonces no se encuentra en el entorno virtual cv:

Instala NumPy en tu entorno virtual de Python

El último paso antes de compilar OpenCV es instalar NumPy, un paquete de Python utilizado para el procesamiento numérico. Para instalar NumPy, asegúrese de estar en el entorno virtual cv (de lo contrario, NumPy se instalará en la versión del sistema de Python en lugar del entorno de cv). Desde allí ejecuta el siguiente comando:

pip install numpy



```

root@frportatil-Satellite-L55-A:/
virtualenvwrapper.user_scripts creating /root/.virtualenvs/premkvirtualenv
virtualenvwrapper.user_scripts creating /root/.virtualenvs/postmkvirtualenv
virtualenvwrapper.user_scripts creating /root/.virtualenvs/prermvirtualenv
virtualenvwrapper.user_scripts creating /root/.virtualenvs/postrmvirtualenv
virtualenvwrapper.user_scripts creating /root/.virtualenvs/predeactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/postdeactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/preactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/postactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/get_env_details
root@frportatil-Satellite-L55-A:# $ echo -e "\n# virtualenv and virtualenvwrapper" >> ~/.bashrc
$: no se encontró la orden
root@frportatil-Satellite-L55-A:# $ echo "export WORKON_HOME=$HOME/.virtualenvs" >> ~/.bashrc
$: no se encontró la orden
root@frportatil-Satellite-L55-A:# $ echo "source /usr/local/bin/virtualenvwrapper.sh" >> ~/.bashrc
$: no se encontró la orden
root@frportatil-Satellite-L55-A:# cd ..
root@frportatil-Satellite-L55-A:# pwd
root@frportatil-Satellite-L55-A:/#
root@frportatil-Satellite-L55-A:# sudo su
root@frportatil-Satellite-L55-A:# ls
bin  cdrom  etc  intrd.img  lib  lost+found  mnt  proc  run  snap  sys  usr  vmlinuz
boot  dev  home  intrd.img.old  lib64  media  opt  root  sbin  srv  tmp  var  vmlinuz.old
root@frportatil-Satellite-L55-A:# cd home
root@frportatil-Satellite-L55-A:/home# cd ..
cd..: no se encontró la orden
root@frportatil-Satellite-L55-A:/home# cd ..
root@frportatil-Satellite-L55-A:/home# cd ..
root@frportatil-Satellite-L55-A:# echo -e "\n# virtualenv and virtualenvwrapper" >> ~/.bashrc
root@frportatil-Satellite-L55-A:# echo "export WORKON_HOME=$HOME/.virtualenvs" >> ~/.bashrc
root@frportatil-Satellite-L55-A:# echo "source /usr/local/bin/virtualenvwrapper.sh" >> ~/.bashrc
root@frportatil-Satellite-L55-A:# mkvirtualenv cv -p python2
Running virtualenv with interpreter /usr/bin/python2
New python executable in /root/.virtualenvs/cv/bin/python2
Also creating executable in /root/.virtualenvs/cv/bin/python
Installing setuptools, pip, wheel...done.
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/predeactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/postdeactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/preactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/postactivate
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/get_env_details
(cv) root@frportatil-Satellite-L55-A:/# workon cv
(cv) root@frportatil-Satellite-L55-A:/# pip install numpy
Collecting numpy
  Downloading numpy-1.14.2-cp27-cp27mu-manylinux1_x86_64.whl (12.1MB)
    14% |███████████| 1.8MB 523kB/s eta 0:00:20

```

Paso # 4: Configurando y compilando OpenCV en Ubuntu 16.04

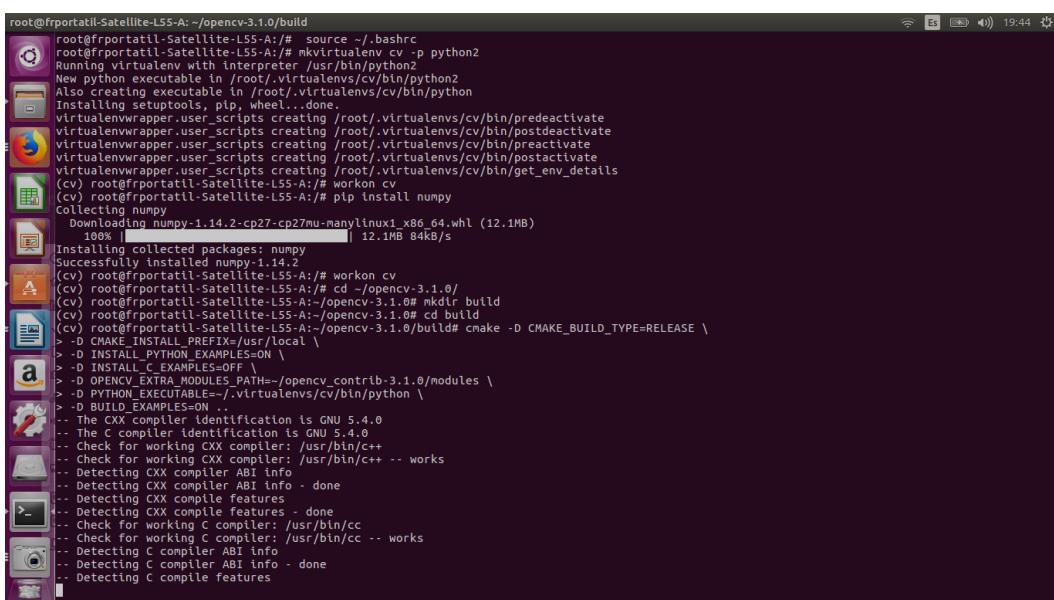
En este punto, todos nuestros prerequisitos necesarios han sido instalados, ¡ahora estamos listos para compilar y OpenCV!

Pero antes de hacerlo, verifique nuevamente que se encuentra en el entorno virtual cv examinando su mensaje (debe ver el texto (cv) que lo precede), y si no, use el comando workon:

```
workon cv
```

debe ubicarse (cv) en la línea de código

```
cd ~/opencv-3.1.0/  
mkdir build  
cd build  
cmake -D CMAKE_BUILD_TYPE=RELEASE \  
-D CMAKE_INSTALL_PREFIX=/usr/local \  
-D INSTALL_PYTHON_EXAMPLES=ON \  
-D INSTALL_C_EXAMPLES=OFF \  
-D OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib-3.1.0/modules \  
-D PYTHON_EXECUTABLE=~/virtualenvs/cv/bin/python \  
-D BUILD_EXAMPLES=ON ..
```



```
root@frportatil-Satellite-L55-A: ~/opencv-3.1.0/build  
root@frportatil-Satellite-L55-A:/# source ~/.bashrc  
root@frportatil-Satellite-L55-A:/# mkvirtualenv cv -p python2  
Running virtualenv with interpreter /usr/bin/python2  
New python executable in /root/.virtualenvs/cv/bin/python2  
Also creating executable in /root/.virtualenvs/cv/bin/python  
Installing setuptools, pip, wheel...done.  
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/predeactivate  
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/postdeactivate  
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/preactivate  
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/postactivate  
virtualenvwrapper.user_scripts creating /root/.virtualenvs/cv/bin/get_env_details  
(cv) root@frportatil-Satellite-L55-A:/# workon cv  
(cv) root@frportatil-Satellite-L55-A:/# pip install numpy  
Collecting numpy  
  Downloading numpy-1.14.2-cp27-cp27mu-manylinux1_x86_64.whl (12.1MB)  
    100% |██████████| 12.1MB 84kB/s  
Installing collected packages: numpy  
Successfully installed numpy-1.14.2  
(cv) root@frportatil-Satellite-L55-A:/# workon cv  
(cv) root@frportatil-Satellite-L55-A:/~opencv-3.1.0/  
(cv) root@frportatil-Satellite-L55-A:/~opencv-3.1.0# cd build  
(cv) root@frportatil-Satellite-L55-A:/~opencv-3.1.0/build# cmake -D CMAKE_BUILD_TYPE=RELEASE \  
> -D CMAKE_INSTALL_PREFIX=/usr/local \  
> -D INSTALL_PYTHON_EXAMPLES=ON \  
> -D INSTALL_C_EXAMPLES=OFF \  
> -D OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib-3.1.0/modules \  
> -D PYTHON_EXECUTABLE=~/virtualenvs/cv/bin/python \  
> -D BUILD_EXAMPLES=ON ..  
-- The CXX compiler identification is GNU 5.4.0  
-- The C compiler identification is GNU 5.4.0  
-- Check for working CXX compiler: /usr/bin/c++  
-- Check for working C compiler: /usr/bin/c++ -- works  
-- Detecting CXX compiler ABI info  
-- Detecting CXX compiler ABI info - done  
-- Detecting CXX compile features  
-- Detecting CXX compile features - done  
-- Check for working C compiler: /usr/bin/cc  
-- Check for working C compiler: /usr/bin/cc -- works  
-- Detecting C compiler ABI info  
-- Detecting C compiler ABI info - done  
-- Detecting C compile features
```

Los comandos anteriores cambian el directorio a ~ / opencv-3.1.0, que si ha seguido este tutorial es donde descargó y desarchivó los archivos .zip.

Nota: Si está obteniendo un error relacionado con stdlib.h: No existe ningún archivo o directorio durante la fase cmake o make de este tutorial, también deberá incluir la siguiente opción en CMake: -D ENABLE_PRECOMPILED_HEADERS = OFF. En este caso, le sugiero que elimine su directorio de compilación, lo vuelva a crear y vuelva a ejecutar CMake con la opción anterior incluida. Esto resolverá el error stdlib.h.

Dentro de este directorio creamos un subdirectorio llamado build y lo cambiamos. El directorio de compilación es donde se realizará la compilación real.

Finalmente, ejecutamos cmake para configurar nuestra compilación.

Antes de pasar a la compilación real de OpenCV, ¡asegúrese de examinar la salida de CMake!

Si está compilando OpenCV en Ubuntu 16.04 con compatibilidad con Python 2.7, asegúrese de que la sección Python 2 incluya rutas válidas para la ruta de Intérprete, Bibliotecas, numpy y paquetes. Tu salida debería ser similar a la mía a continuación:

```
root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build
-- OpenCL:
  Version:           dynamic
  Include path:      /root/opencv-3.1.0/3rdparty/include/opencl/1.2
  Use AMDFFT:        NO
  Use AMDBLAS:       NO

-- Python 2:
  Interpreter:      /root/.virtualenvs/cv/bin/python (ver 2.7.12)
  Libraries:         /usr/lib/x86_64-linux-gnu/libpython2.7.so (ver 2.7.12)
  numpy:             /root/.virtualenvs/cv/local/lib/python2.7/site-packages/numpy/core/include (ver 1.14.2)
  packages path:    lib/python2.7/site-packages

-- Python 3:
  Interpreter:      /usr/bin/python3 (ver 3.5.2)
  Python (for build): /root/.virtualenvs/cv/bin/python

-- Java:
  ant:               NO
  JNI:               NO
  Java wrappers:    NO
  Java tests:       NO

-- Matlab:
  Matlab:           Matlab not found or implicitly disabled

-- Documentation:
  Doxygen:          /usr/bin/doxygen (ver 1.8.11)
  PlantUML:         NO

-- Tests and samples:
  Tests:             YES
  Performance tests: YES
  C/C++ Examples:   YES

-- Install path:     /usr/local
-- cvconfig.h is in: /root/opencv-3.1.0/build

-- Configuring done
-- Generating done
-- Build files have been written to: /root/opencv-3.1.0/build
(cv) root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build#
```

Nuevamente, observe cómo mi ruta de Intérprete, Bibliotecas, numpy y paquetes se han establecido correctamente.

Si no ve los entornos virtuales de cv en estas rutas de acceso variables, es casi seguro que no se encuentra en el entorno virtual de cv antes de ejecutar CMake.

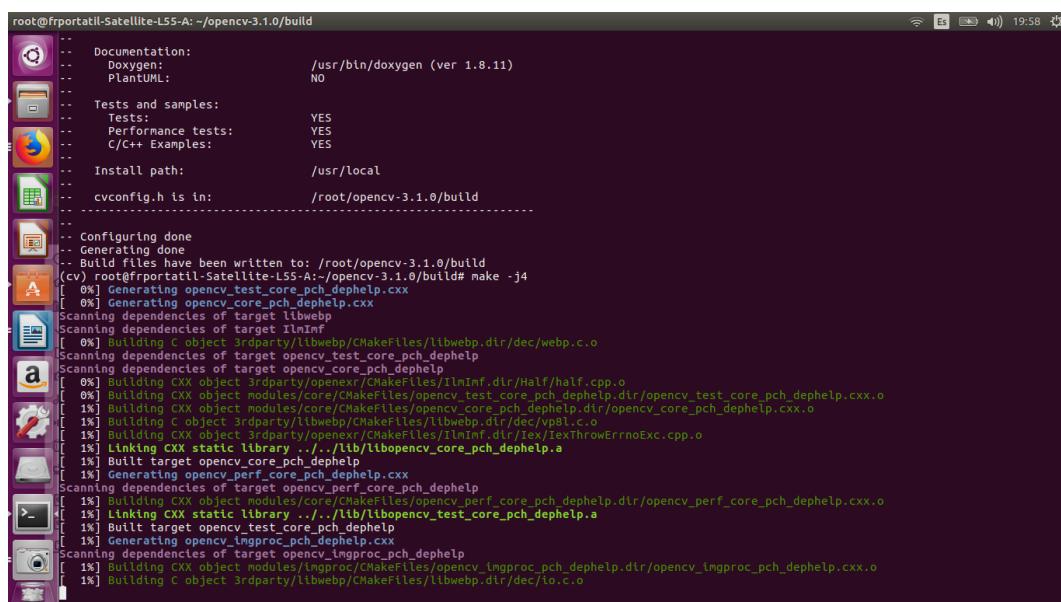
Si ese es realmente el caso, simplemente acceda al entorno virtual cv llamando a workon cv y vuelva a ejecutar el comando CMake mencionado anteriormente.

Suponiendo que su comando CMake salió sin ningún error, ahora puede compilar OpenCV:

```
make -j4
```

El modificador -j controla el número de procesos que se utilizarán al compilar OpenCV; querrá establecer este valor en la cantidad de procesadores / núcleos en su máquina. En mi caso, tengo un procesador de cuatro núcleos, así que configuro -j4.

El uso de procesos múltiples permite que OpenCV compile más rápido; sin embargo, hay momentos en los que se tocan las condiciones de carrera y las bombas de compilación salen. Si bien no se puede decir realmente si este es el caso sin una gran cantidad de experiencia previa compilando OpenCV, si se encuentra con un error, mi primera sugerencia sería ejecutar make clean para vaciar la compilación, seguido de la compilación usando solo un único núcleo:



```
root@frportatil-Satellite-L55-A: ~/opencv-3.1.0/build
-- Configuration
  Documentation:
    Doxygen:           /usr/bin/doxygen (ver 1.8.11)
    PlantUML:          NO
  Tests and samples:
    Tests:             YES
    Performance tests: YES
    C/C++ Examples:   YES
  Install path:        /usr/local
  cvconfig.h is in:   /root/opencv-3.1.0/build
-- Configuring done
-- Generating done
-- Build files have been written to: /root/opencv-3.1.0/build
[cv] root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build# make -j4
[ 0%] Generating opencv_test_core_pch_dephelp.cxx
[ 0%] Generating opencv_core_pch_dephelp.cxx
Scanning dependencies of target libwebp
Scanning dependencies of target libwebp
[ 0%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/webp.c.o
Scanning dependencies of target opencv_test_core_pch_dephelp
Scanning dependencies of target opencv_core_pch_dephelp
[ 0%] Building CXX object 3rdparty/opencv3/CMakeFiles/IlmImf.dir/Half/half.cpp.o
[ 0%] Building CXX object modules/core/CMakeFiles/opencv_test_core_pch_dephelp.dir/opencv_test_core_pch_dephelp.cxx.o
[ 1%] Building CXX object modules/core/CMakeFiles/opencv_core_pch_dephelp.dir/opencv_core_pch_dephelp.cxx.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/vp8l.c.o
[ 1%] Building CXX object 3rdparty/opencv3/CMakeFiles/IlmImf.dir/tx/txThrowErrnoExc.cpp.o
[ 1%] Linking CXX static library ../../lib/libopencv_core_pch_dephelp.a
[ 1%] Built target opencv_core_pch_dephelp
[ 1%] Generating opencv_perf_core_pch_dephelp.cxx
Scanning dependencies of target opencv_perf_core_pch_dephelp
[ 1%] Building CXX object modules/core/CMakeFiles/opencv_perf_core_pch_dephelp.dir/opencv_perf_core_pch_dephelp.cxx.o
[ 1%] Linking CXX static library ../../lib/libopencv_test_core_pch_dephelp.a
[ 1%] Built target opencv_test_core_pch_dephelp
[ 1%] Generating opencv_imgproc_pch_dephelp.cxx
Scanning dependencies of target opencv_imgproc_pch_dephelp
[ 1%] Building CXX object modules/imgproc/CMakeFiles/opencv_imgproc_pch_dephelp.dir/opencv_imgproc_pch_dephelp.cxx.o
[ 1%] Building C object 3rdparty/libwebp/CMakeFiles/libwebp.dir/dec/io.c.o
```

Nota: Se tarda de 15 a 20 minutos

```
make clean
```

```
make
```

Aparece la siguiente pantalla de instalación que tiene un tiempo de 30 a 45 minutos.

```

root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build
[100%] Linking CXX executable ../../bin/cpp-example-kmeans
[100%] Built target example_kmeans
Scanning dependencies of target example_fitellipse
[100%] Built target example_kmeans
Scanning dependencies of target tutorial_bg_sub
[100%] Linking CXX executable ../../bin/cpp-example-shape_example
[100%] Building CXX object samples/cpp/CMakeFiles/example_fitellipse.dir/fitellipse.cpp.o
[100%] Building CXX object samples/cpp/CMakeFiles/tutorial_bg_sub.dir/tutorial_code/video/bg_sub.cpp.o
[100%] Built target example_shape_example
Scanning dependencies of target example_stitching
[100%] Building CXX object samples/cpp/CMakeFiles/example_stitching.dir/stitching.cpp.o
[100%] Linking CXX executable ../../bin/cpp-example-facetedetect
[100%] Built target example_facetedetect
Scanning dependencies of target example_camsiftdemo
[100%] Building CXX object samples/cpp/CMakeFiles/example_camsiftdemo.dir/camsiftdemo.cpp.o
[100%] Linking CXX executable ../../bin/cpp-example-fitellipse
[100%] Built target example_fitellipse
Scanning dependencies of target example_ffildemo
[100%] Building CXX object samples/cpp/CMakeFiles/example_ffildemo.dir/ffildemo.cpp.o
[100%] Linking CXX executable ../../bin/cpp-tutorial-bg_sub
[100%] Built target example_ffildemo
[100%] Linking CXX executable ../../bin/cpp-example-camsiftdemo
[100%] Built target example_camsiftdemo
[100%] Linking CXX executable ../../bin/cpp-example-stitching
[100%] Linking CXX executable ../../bin/cpp-example-ffildemo
[100%] Built target example_stitching
[100%] Built target example_ffildemo
[cv] root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build# make clean
[cv] root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build# make
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dec/webrtc.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dec/vp8.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dec/lo.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dec/idec.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dec/frame.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dec/layer.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dec/tree.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dec/quint.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dec/decifne.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dec/alpha.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dsp/vp8.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dsp/dec_neon.c.o
[ 0%] Building C object 3rdparty/libwebrtc/CMakeFiles/libwebrtc.dir/dsp/dec.c.o

```

El último paso es instalar OpenCV 3 en Ubuntu 16.04:

`sudo make install`

`sudo ldconfig`

Paso #5: Finalmente tu OpenCV instalado.

Después de ejecutar `sudo make install`, tus enlaces de Python 2.7 para OpenCV 3 ahora deberían estar ubicados en `/usr/local/lib/python2.7/site-packages/`. Puede verificar esto usando el comando `ls`:

`ls -l /usr/local/lib/python2.7/site-packages/`

Debe aparecer la siguiente pantalla: total 1972

`-rw-r--r-- 1 root staff 2016608 Sep 15 09:11 cv2.so`

```

root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build
-- Installing: /usr/local/share/OpenCV/samples/python/lk_homography.py
-- Installing: /usr/local/share/OpenCV/samples/python/morphology.py
-- Installing: /usr/local/share/OpenCV/samples/python/objdetect/lk_homography.py
-- Installing: /usr/local/share/OpenCV/samples/python/lk_track.py
-- Installing: /usr/local/share/OpenCV/samples/python/facedetect.py
-- Installing: /usr/local/share/OpenCV/samples/python/hooughlines.py
-- Installing: /usr/local/share/OpenCV/samples/python/video.py
-- Installing: /usr/local/share/OpenCV/samples/python/gabor_threads.py
-- Installing: /usr/local/share/OpenCV/samples/python/asift.py
-- Installing: /usr/local/share/OpenCV/samples/python/camshift.py
-- Installing: /usr/local/share/OpenCV/samples/python/ftline.py
-- Installing: /usr/local/share/OpenCV/samples/python/watershed.py
-- Installing: /usr/local/share/OpenCV/samples/python/digits_video.py
-- Installing: /usr/local/share/OpenCV/samples/python/digits.py
-- Installing: /usr/local/share/OpenCV/samples/python/opencv_version.py
-- Installing: /usr/local/share/OpenCV/samples/python/calibrate.py
-- Installing: /usr/local/share/OpenCV/samples/python/turucontours.py
-- Installing: /usr/local/share/OpenCV/samples/python/demo.py
-- Installing: /usr/local/share/OpenCV/samples/python/find_obj.py
-- Installing: /usr/local/share/OpenCV/samples/python/opt_flow.py
-- Installing: /usr/local/share/OpenCV/samples/python/common.py
-- Installing: /usr/local/share/OpenCV/samples/python/grabcut.py
-- Installing: /usr/local/share/OpenCV/samples/python/video_threaded.py
-- Installing: /usr/local/share/OpenCV/samples/python/plane_ar.py
-- Installing: /usr/local/share/OpenCV/samples/python/dft.py
-- Installing: /usr/local/share/OpenCV/samples/python/color_histogram.py
-- Installing: /usr/local/share/OpenCV/samples/python/texture_flow.py
-- Installing: /usr/local/share/OpenCV/samples/python/edge.py
-- Installing: /usr/local/share/OpenCV/samples/python/deconvolution.py
[cv] root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build# sudo ldconfig
[cv] root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build# ls -l /usr/local/lib/python2.7/site-packages/
total 1972
-rw-r--r-- 1 root staff 2016608 mar 28 20:47 cv2.so
[cv] root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build# total 1972
El programa «total» no está instalado. Puede instalarlo escribiendo:
apt install radiance
[cv] root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build# ls -l /usr/local/lib/python2.7/site-packages/
total 1972
-rw-r--r-- 1 root staff 2016608 mar 28 20:47 cv2.so
[cv] root@frportatil-Satellite-L55-A:~/opencv-3.1.0/build# 
```

Nota: en algunos casos, puede encontrar que OpenCV se instaló en /usr/local/lib/python2.7/dist-packages en lugar de /usr/local/lib/python2.7/site-packages (note dist-packages versus site-packages). Si sus enlaces cv2.so no están en el directorio site-packages, asegúrese de marcar dist-pakages.

El último paso es sincronizar nuestras vinculaciones OpenCV cv2.so en nuestro entorno virtual cv para Python 2.7:

```
cd ~/.virtualenvs/cv/lib/python2.7/site-packages/
```

```
ln -s /usr/local/lib/python2.7/site-packages/cv2.so cv2.so
```

Paso #6: Probando tu OpenCV instalado.

Para verificar que su instalación funciona:

Abre una nueva terminal y ejecute el comando workon para acceder al entorno virtual cv Python.

Intenta importar las vinculaciones de Python + OpenCV.

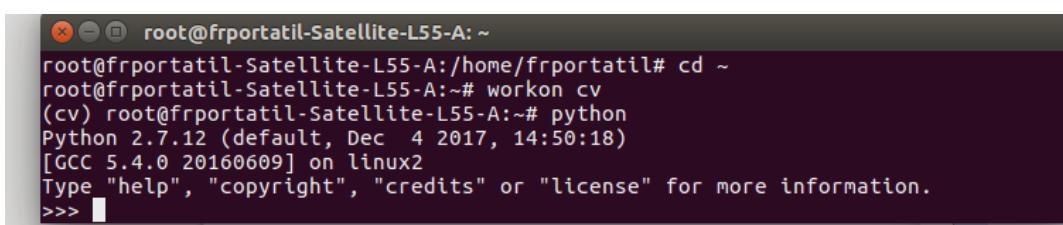
He demostrado cómo realizar estos pasos a continuación:

```
cd ~
```

```
workon cv
```

```
python
```

Debe salir la siguiente pantalla



```
root@frportatil-Satellite-L55-A: ~
root@frportatil-Satellite-L55-A:/home/frportatil# cd ~
root@frportatil-Satellite-L55-A:~/# workon cv
(cv) root@frportatil-Satellite-L55-A:~/# python
Python 2.7.12 (default, Dec  4 2017, 14:50:18)
[GCC 5.4.0 20160609]
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

Anexo C: Cuestionario de Escala de Usabilidad del Sistema

1. Me gustaría usar esta Aplicación de Software con frecuencia.

total desacuerdo								total acuerdo	
1		2		3		4		5	

2. Encuentro esta Aplicación de Software innecesariamente compleja.

total desacuerdo								total acuerdo	
1		2		3		4		5	

3. Creo que esta Aplicación de Software es fácil de usar.

total desacuerdo								total acuerdo	
1		2		3		4		5	

4. Necesitaría ayuda para usar esta Aplicación de Software.

total desacuerdo								total acuerdo	
1		2		3		4		5	

5. Las diversas funciones están bien integradas (constituyen un todo).

total desacuerdo								total acuerdo	
1		2		3		4		5	

6. Hay demasiada incoherencia en esta Aplicación de Software.

total desacuerdo								total acuerdo	
1		2		3		4		5	

7. La mayoría de las personas aprendería a usar esta Aplicación de Software muy rápidamente.

total desacuerdo								total acuerdo	
1		2		3		4		5	

8. Me parece muy engorrosa y difícil de usar.

total desacuerdo								total acuerdo	
1		2		3		4		5	

9. Tengo mucha confianza usándolo.

total desacuerdo								total acuerdo	
1		2		3		4		5	

10. Necesito aprender muchas cosas antes de poder comenzar a utilizar esta Aplicación de Software.

total desacuerdo								total acuerdo	
1		2		3		4		5	

Anexo D: Cuestionario de Comodidad / Facilidad de Uso

1. El esfuerzo mental requerido para el desarrollo de las tareas ha sido:

Mucho esfuerzo								Poco esfuerzo
1		2		3		4		5

2. La velocidad de funcionamiento es:

Muy lenta								Muy rápida
1		2		3		4		5

3. La comodidad es:

Muy incómodo								Muy cómodo
1		2		3		4		5

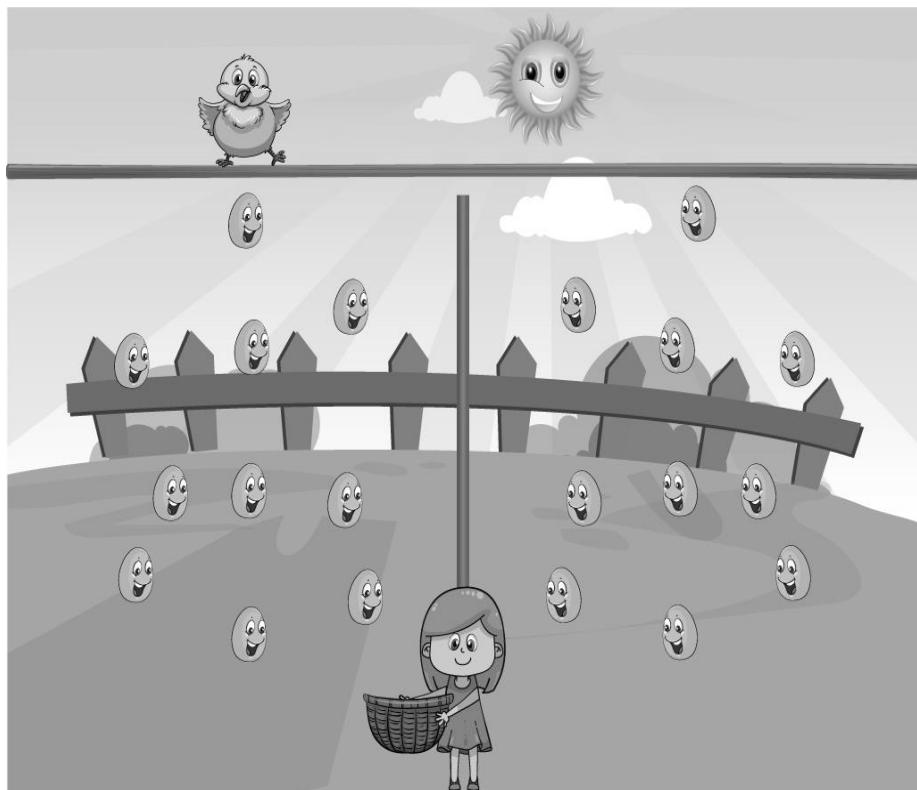
4. En general, el manejo de la Aplicación de Software es:

Muy difícil								Muy fácil
1		2		3		4		5

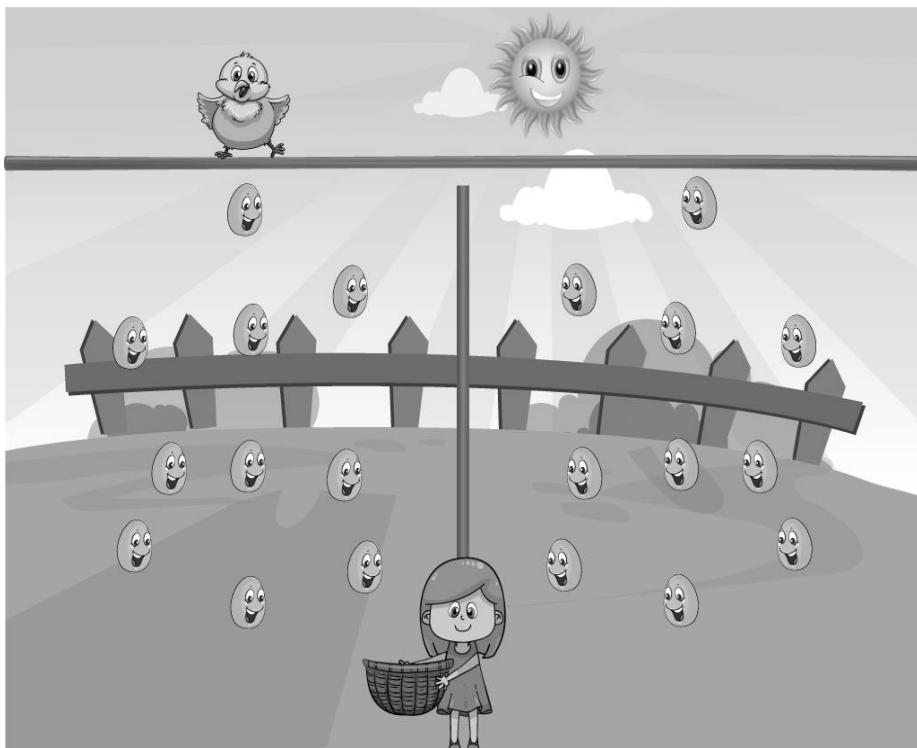
Anexo E: Test de Evaluación de Ubicación Espacial

Actividad Izquierda y Derecha

Encierre en un círculo los huevos que se encuentran en el lado izquierdo de la niña.

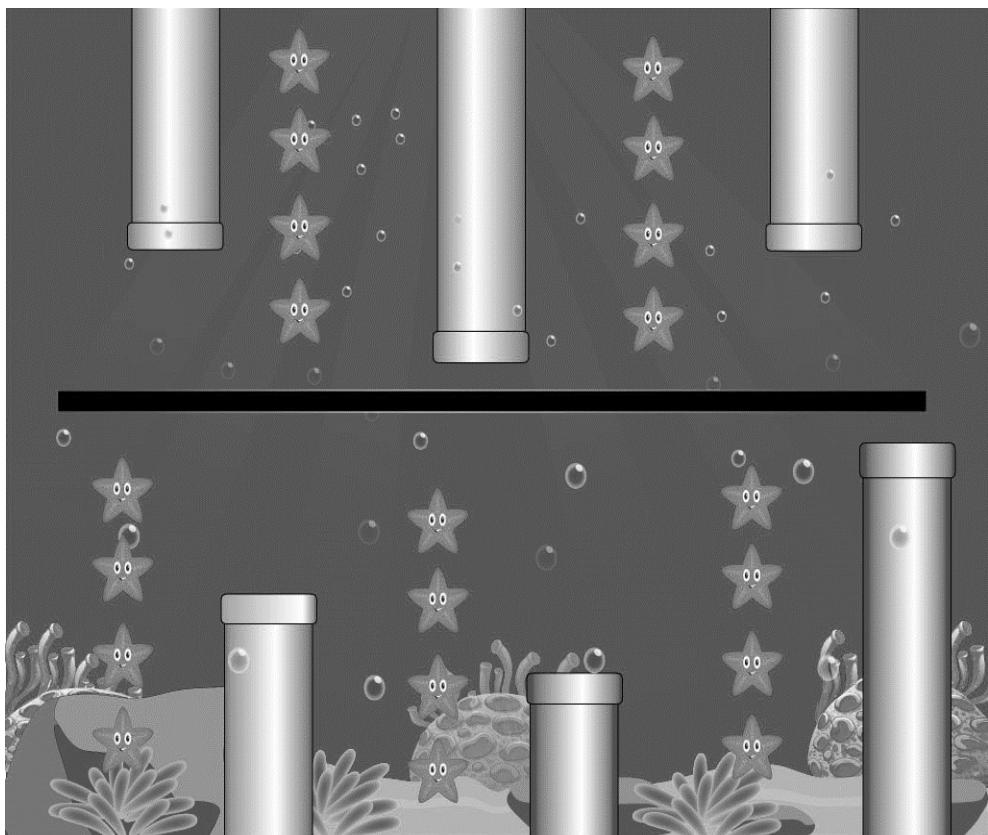


Encierre en un círculo los huevos que se encuentran en el lado derecho de la niña.

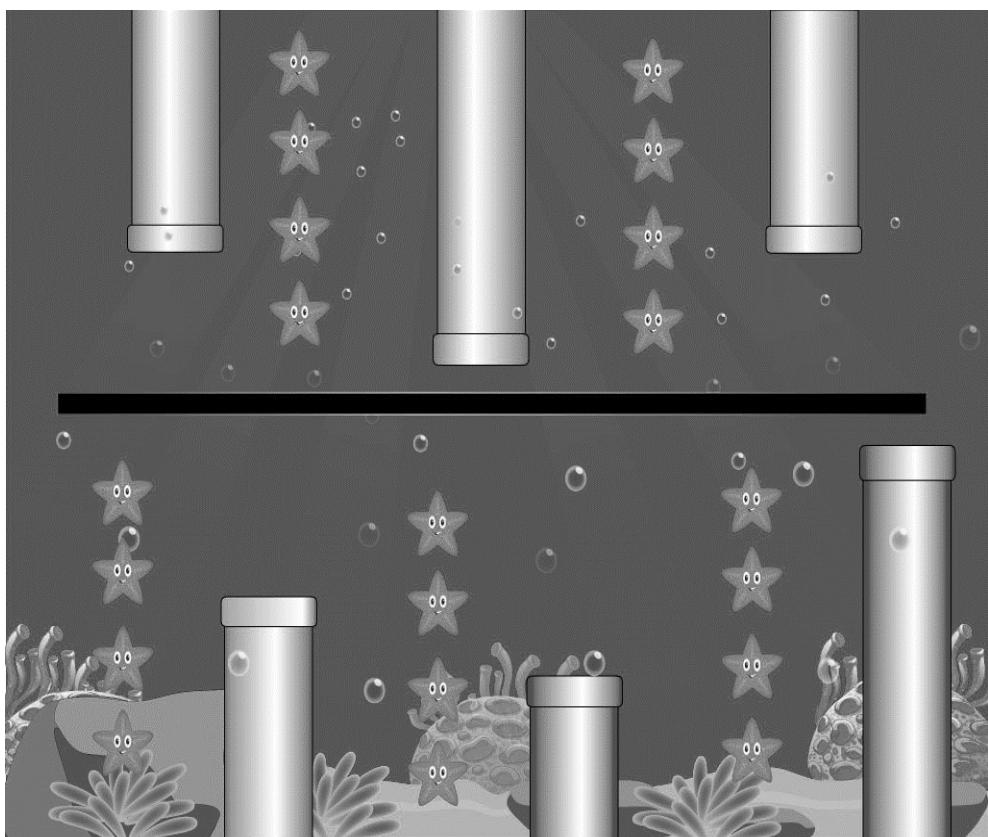


Actividad Arriba y Abajo

Encierre en un círculo las estrellas que se encuentran arriba de la línea negra



Encierre en un círculo las estrellas que se encuentran abajo de la línea negra.



Encierre en un círculo la pelota que está arriba del niño

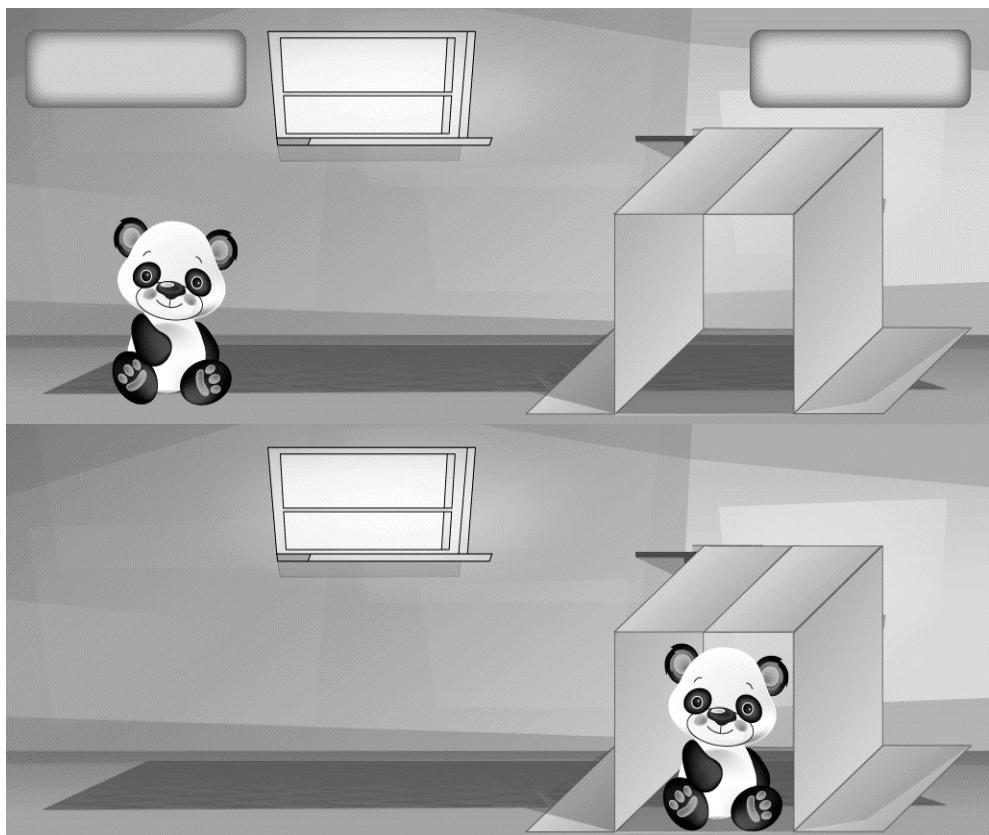


Encierre en un círculo la pelota que esta abajo del niño

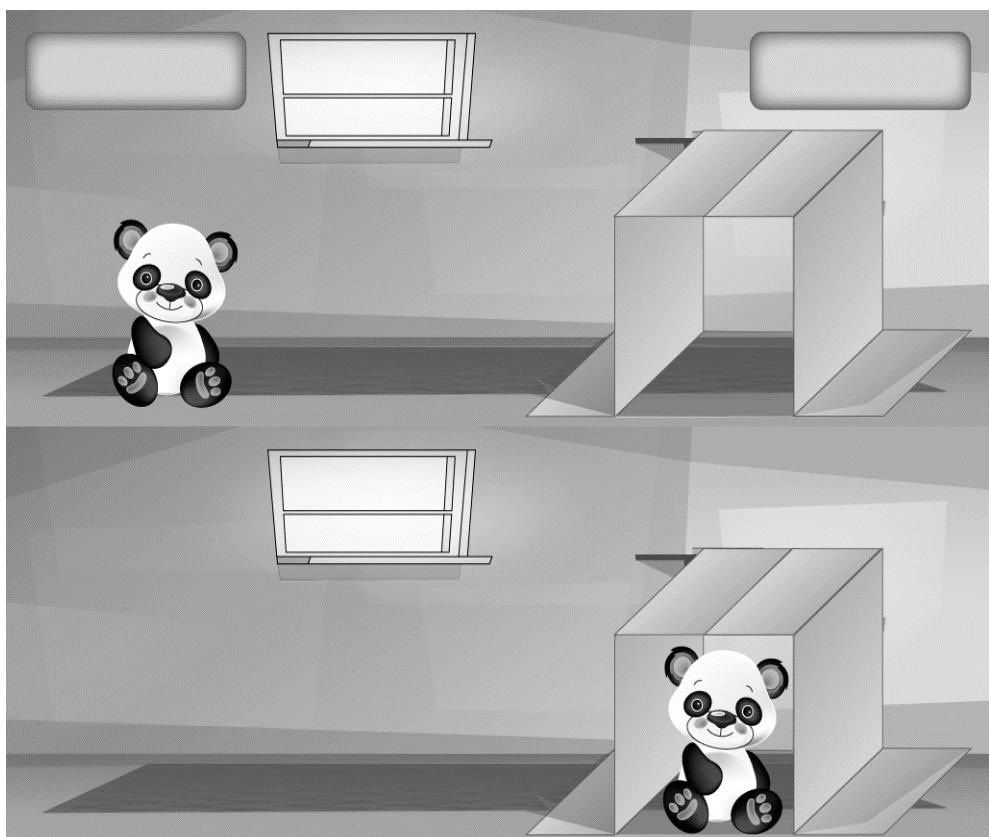


Actividad Adentro y Afuera

Encierre en un círculo el oso que se encuentran adentro de la caja

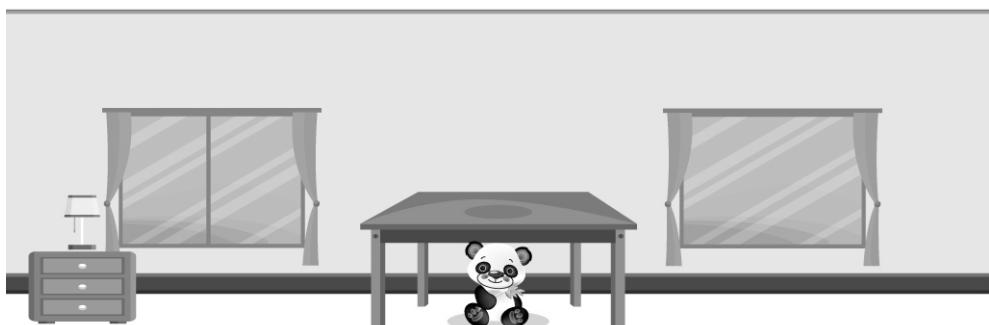
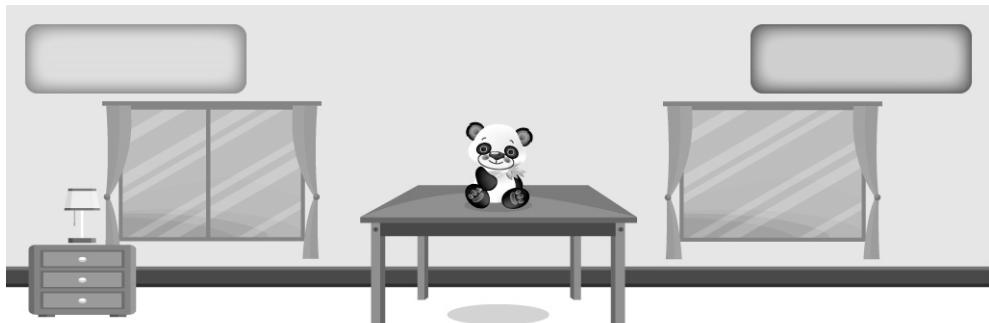


Encierre en un círculo el oso que se encuentran afuera de la caja

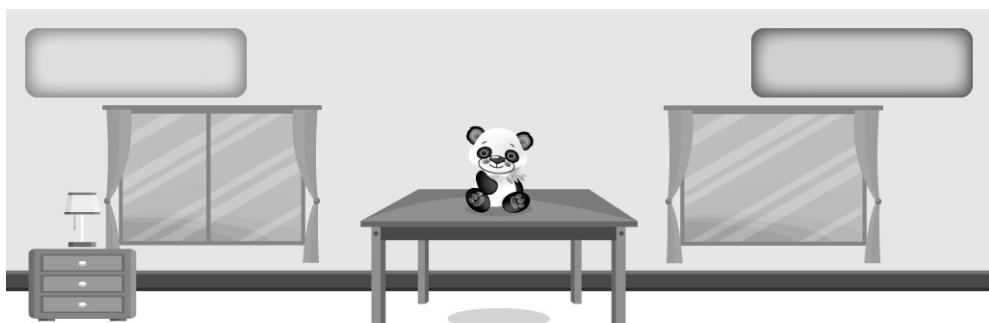


Actividad Encima y Debajo

Encierre en un círculo el oso que se encuentra encima de la mesa



Encierre en un círculo el oso que se encuentra debajo de la mesa



Actividad Laberinto

Ayuda a la ardilla a llegar al árbol.

