

Pontificia Universidad Católica de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Informática

**APRENDIZAJE DE PIANO UTILIZANDO
TECNOLOGÍA KINECT BASADO EN EL PARADIGMA
DE JUEGOS SERIOS**

ISMAEL IGNACIO FUENZALIDA LEGÜE

LUIS EDUARDO RIQUELME GÓMEZ

INFORME FINAL DEL PROYECTO
PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO DE EJECUCIÓN EN INFORMÁTICA

Enero de 2014

Pontificia Universidad Católica de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Informática

**APRENDIZAJE DE PIANO UTILIZANDO
TECNOLOGÍA KINECT BASADO EN EL PARADIGMA
DE JUEGOS SERIOS**

ISMAEL IGNACIO FUENZALIDA LEGÜE

LUIS EDUARDO RIQUELME GÓMEZ

Profesor Guía: **José Miguel Rubio León**

Profesor Co-Referente: **Rodolfo Villarroel Acevedo**

Carrera: **Ingeniería de Ejecución en Informática**

Enero de 2014

Dedicatoria:

A nuestros familiares, amigos y grandes personas que con entereza brindaron su apoyo incondicional a lo largo de nuestra carrera universitaria.

Resumen

El presente proyecto consiste en construir un producto *software* para el aprendizaje de piano utilizando la tecnología Kinect basado en el paradigma de juegos serios. En consecuencia, se debió implementar la detección de dedos o *FingerTracking* en inglés, lo cual es una innovación en el uso de la tecnología Kinect. Junto con lo anterior, se estudió la problemática; para generar el análisis, diseño e implementación de la solución propuesta. Para realizar lo anterior, se procedió a generar diagramas UML que permitieran realizar una buena fase de análisis del sistema propuesto ya que, una buena fase de análisis permitirá un buen desarrollo del sistema. Además, se estudió una metodología de aprendizaje de piano con el fin de que las lecciones propuestas en el *software* sean lo más cercanas a la realidad. Para finalmente, construir el producto *software* que cumpla con los requerimientos expresados a lo largo del documento.

Palabras Clave: *Kinect, Juegos serios, FingerTracking, UML.*

Abstract

This project is to build a software product for learning piano using Kinect technology based the paradigm of serious games. Accordingly, it should implement FingerTracking, which is an innovation in the use of the Kinect technology. Along with this, the problem was studied to generate the analysis, design and implementation of the proposed solution. To perform this, we proceeded to generate UML diagrams that allow to make a good analysis phase of the proposed system as a good analysis phase allows a good development system. Moreover, a learning methodology studied piano so that the lesson software proposals are as close to reality. To eventually build the software product that meets the requirements expressed throughout the document.

Keywords: *Kinect, Serious Games, FingerTracking, UML.*

Índice general

Resumen	i
Índice general.....	ii
Índice de figuras	iv
Índice de tablas	v
1 Introducción	1
1.1 Alcances y propósito del documento.....	1
2 Problema y solución	2
2.1 Descripción del problema.....	2
2.2 Objetivos generales.....	4
2.3 Objetivos específicos.....	4
2.4 Descripción de la solución.....	4
2.5 Planificación	5
2.5.1 Plan de trabajo.....	5
2.5.2 Carta Gantt	5
3 Marco Teórico	8
3.1 Definiciones.....	8
3.2 Serious Games	9
3.3 Realidad Aumentada.....	9
3.4 Kinect.....	10
3.5 Interior de la Kinect.....	11
3.6 Interfaz Natural de Usuario	12
3.7 Metodología aprendizaje de Piano.....	14
3.7.1 Metodología aprendizaje.....	14
3.8.1 Reunión inicial con especialistas en Piano	15
3.8.2 Reunión final con especialistas en Piano	16
4 Especificación de requerimientos	17
4.1 Funcionales	17
4.2 No funcionales.....	17
4.3 Diagrama Casos de Uso.....	18

4.3.1	Diagrama General del Sistema.....	18
4.3.2	Caso de uso Utilizar Piano Libre	19
4.3.3	Caso de uso Realizar Curso	19
4.4	Análisis de alternativas para implementación.....	21
5	Solución propuesta	22
5.1	Arquitectura	22
5.2	Metodología.....	23
5.3	Plataforma de Desarrollo	25
5.4	Diseño de la Solución	26
5.4.1	Diagrama de Actividad Realizar Curso	26
5.4.2	Diagrama de Actividad Piano Libre	27
5.4.3	Diagrama de Relacional	28
6	Pruebas de Software	29
6.1	Planificación de pruebas	29
6.2	Especificación de Casos de Prueba.....	31
6.2.1	Caso de prueba Creación cuenta de usuario.....	31
6.2.2	Caso de prueba Interactuar con Piano Libre	33
6.2.3	Caso de prueba Interacción con lecciones del curso.....	34
7	Conclusiones y trabajo futuro.....	36
8	Referencias.....	38
1	Anexos	39
A:	Detalle de Actividades.....	40
B:	Estudio de Factibilidad	45
B:1	Factibilidad Económica	45
B:2	Factibilidad Técnica.....	46
B:3	Factibilidad Operativa.....	47
B:4	Factibilidad Legal	48
B:5	Conclusión de Factibilidad.....	48
C:	Diseño de Interfaces	49

Índice de figuras

Figura 2.1 Esquema gráfico de la solución.	4
Figura 3.1 Realidad Aumentada: salida, entrada y registro.	10
Figura 3.2 Funcionamiento de la Kinect.....	11
Figura 3.3 Vista Frontal Kinect.....	11
Figura 3.4 Vista Frontal Interna Kinect	12
Figura 3.5 Interfaz Natural de Usuario sobre pantalla táctil	12
Figura 3.6 Interfaz Natural de Usuario mediante Kinect	13
Figura 4.1 Diagrama general de casos de uso.	18
Figura 4.2 Caso de uso Utilizar Piano Libre.....	19
Figura 4.3 Caso de uso Realizar Curso.	19
Figura 6.1 Arquitectura Lógica.	22
Figura 6.2 Etapas Metodología de Prototipado.....	24
Figura 6.3 Diagrama de actividad Realizar Curso	26
Figura 6.4 Diagrama de actividad Piano Libre	27
Figura 6.5 Diagrama Relacional	28
Figura B.1 Interfaz Menú Principal.	49
Figura B.2 Interfaz Usuarios.	49
Figura B.3 Interfaz Agregar Usuario.	50
Figura B.4 Interfaz Selección de Usuario.	51
Figura B.5 Interfaz Eliminar Usuario.....	51
Figura B.6 Interfaz Piano Libre.	52
Figura B.7 Interfaz Curso Piano.	52
Figura B.8 Interfaz Curso Piano – Lección 1.....	53
Figura B.9 Interfaz Curso Piano – Lección 2 Paso 1.....	53
Figura B.10 Interfaz Curso Piano – Lección 2 Paso 2.....	54
Figura B.11 Interfaz Curso Piano – Lección 3 Paso 1.....	54
Figura B.12 Interfaz Curso Piano – Lección 3 Paso 2.....	55
Figura B.13 Interfaz Curso Piano – Lección 4 Paso 1.....	55
Figura B.14 Interfaz Curso Piano – Rehacer Lección.....	56
Figura B.15 Interfaz Menú Principal – Ayuda.....	56
Figura B.16 Interfaz Piano Libre – Ayuda.....	57
Figura B.17 Interfaz Curso Piano – Ayuda.....	57

Índice de tablas

Tabla 2.1 Fase recolección y refinamiento de requisitos.....	5
Tabla 2.2 Fase diseño rápido.....	5
Tabla 2.3 Fase prototipado, parte 1.....	5
Tabla 2.4 Fase prototipado, parte 2.....	6
Tabla 2.5 Fase Evaluación del Prototipo.....	6
Tabla 2.6 Fase Refinamiento del Prototipo.....	6
Tabla 2.7 Fase Producto de Ingeniería.....	7
Tabla 3.1 Metodología de aprendizaje de Piano - Lecciones.....	14
Tabla 4.1 Caso de uso específico: Elegir número lección.	20
Tabla 5.1 Plan de Prueba Creación de cuenta de usuario.	29
Tabla 5.2 Plan de Prueba Interactuar Piano Libre.....	30
Tabla 5.3 Plan de Prueba Interacción con lecciones del curso.	31
Tabla 5.4 Caso de Prueba Creación cuenta de usuario.	32
Tabla 5.5 Caso de prueba Interactuar con Piano Libre.	33
Tabla 5.6 Caso de prueba Interacción con lecciones del curso.....	34
Tabla B.1 Factibilidad económica.....	45

1 Introducción

1.1 Alcances y propósito del documento

La sociedad, en la actualidad, está avanzando en diversas materias de la vida cotidiana; en este sentido, se desarrollan investigaciones y estudios para lograr niveles de calidad de vida cada vez más superiores; un factor catalizador de este proceso ha sido la vertiginosa evolución de la informática y sus campos involucrados. De hecho, es evidente que las personas requieren mayores prestaciones de los artefactos tecnológicos que se comercializan en el mercado, lo que genera la necesidad de desarrollar nuevas aplicaciones por medio de la creación de software. Uno de los recursos tecnológicos que se han desarrollado para innovar y ampliar el campo cotidiano de la informática, es la tecnología de realidad aumentada en conjunto con dispositivos que permiten una interacción natural entre las máquinas y el cuerpo humano, como lo es la tecnología del sensor Kinect.

El presente informe trata sobre el proceso de análisis, diseño y desarrollo de una aplicación para una computadora con dispositivo Kinect, con las características descritas en el párrafo anterior y además integrando un paradigma llamado juegos serios, que es aplicado en el área musical, específicamente con el instrumento de piano y un curso del mismo. Para cumplir este propósito, el equipo de trabajo se propone diversas metas a cumplir; cada una de ellas explicitadas en diversos puntos del documento.

Este informe presenta, en primer lugar, la descripción de lo que el equipo ha definido como misión y alcances de la aplicación a desarrollar. Luego, se diseña la estructura del sistema a desarrollar; especificando su arquitectura, requerimientos y medios por los cuales el futuro usuario establecerá contacto con el sistema. Además, se desarrolla cada concepto básico involucrado en el proyecto, como lo es juegos serios, realidad aumentada, dispositivo Kinect, etc.; en conjunto a la planificación del proyecto mismo. Siguiendo a esto, se destacan diversos puntos en relación al diseño de la solución, describiendo diagramas de actividad y uno relacional. Todo lo anterior con el fin de obtener parámetros precisos acerca de lo que el sistema ofrecerá cuando éste es implementado. Finalmente, se muestran imágenes del sistema final ya desarrollado y las conclusiones del mismo.

2 Problema y solución

2.1 Descripción del problema

Desde la antigüedad el uso de instrumentos musicales ha sido común entre toda la comunidad, éstos amenizan ambientes y permiten a los oyentes disfrutar de variadas melodías que pueden llegar a provocar sensaciones inigualables y lograr emanar recuerdos, momentos y vivencias sólo al escuchar una pieza musical. Es por esta misma razón que muchas personas, que gustan de la música, sienten de ella un atractivo con el cual pueden pasar horas experimentando; en específico las que deciden dedicarse a la creación e interpretación de piezas musicales con un instrumento en particular, que podría ser tanto una guitarra como un piano u otro. Como se puede inferir, para lograr esto se necesita tener idea de música y del instrumento en sí, por lo que requieren una vasta experiencia y aprendizaje del instrumento; lo que conlleva tiempo, costos y dedicación.

En la actualidad la forma de aprender un instrumento musical adecuadamente, en específico el piano, sigue siendo de la manera tradicional; requiriendo cursos con profesores que enseñen este instrumento junto con una disponibilidad de horario para éstas; además de la necesidad de tener un piano o -en su defecto- un teclado para poder llevar a cabo el aprendizaje. Lo anterior supone un costo que muchas veces puede ser elevado para una persona que no tiene los medios para pagar un curso; esto sin considerar el costo que puede existir en transporte tanto de la persona como del piano mismo en casos de lejanía con el lugar del curso.

Por otra parte, existe una monotonía en estas enseñanzas que pueden provocar aburrimiento de parte del alumno, y con esto, causar un mal aprendizaje; pues -aunque haya mucho interés- siempre puede existir esa necesidad de curiosear maneras más entretenidas o diferentes para experimentar. Una de estas maneras son los llamados *serious games*, rama que trata de juegos creados para un propósito principal -como educación-, más que sólo para diversión. Con este concepto se hace un contraste a la manera tradicional con la cual se podría enseñar un instrumento musical, pues además, un juego para el usuario podría abaratar costos como los mencionados anteriormente. Sin embargo, a pesar de esto, aprender un instrumento musical como lo es el piano en un ambiente informático puede llegar a ser complicado si no existe una interfaz natural de usuario que permita la interacción del usuario con el sistema sin manipular un dispositivo de mando/entrada en sus manos; algo que la tecnología del sensor Kinect sí permite.

En otro aspecto importante, cabe mencionar la metodología que se sigue en un curso de piano tradicional; por fases o lecciones, desde aspectos básicos de las notas musicales sobre las teclas, hasta complejos acordes y composiciones con ambas manos. La forma clásica de seguir curso de piano se podría resumir en; primero, conocer lo que es un piano, tanto conceptualmente como interactuando con él, con cada tecla, desde la parte más grave hasta lo más agudo -o escalas musicales-, algo así como fusionarse en cuerpo y alma con el piano, consiguiendo agilizar los dedos, utilizando una sola mano y luego ambas; como recomendación -según la tendencia normal en cursos de piano- la mano derecha es la que se debe practicar primero. Segundo; conocer algo de teoría básica sobre las notas en pentagrama y teclas, escalas musicales y posición correcta de los dedos sobre el teclado, además de potenciar el sentido del ritmo de la mano izquierda en conjunto a una melodía simple de la mano derecha; mejorando lentamente la coordinación entre ambas. Tercero; empezar a manejar acordes, arpeggios y conseguir generar melodía agradable, cambiando ritmo, velocidades y los acordes mismos; también con ambas manos. Cuarto; aprender distintas técnicas que ayudan a un mejor manejo y generación de melodía; técnicas como barrido con una mano, desplazamiento dedil, etc. [1]

Como se puede apreciar, una lección de piano conlleva una serie de etapas necesarias para manejar un piano o teclado de forma “experta”. Esto requiere tiempo y meses -y hasta años- de ensayo y práctica, siguiendo paso a paso cada etapa, comprendiendo y discerniendo todo lo que se hace. Claramente, la manera tradicional es la más adecuada para aprender, pues lo primordial es interactuar directamente con un piano y con un profesor guía; sin embargo, la posibilidad de ver esto en un ambiente informático utilizando realidad aumentada puede resultar tan novedoso que las ansias por ejercitar en él pueden estimular de sobremanera el aprendizaje.

2.2 Objetivos generales

El objetivo principal que se busca cumplir es desarrollar una herramienta de apoyo al aprendizaje del piano, utilizando la tecnología Kinect y el paradigma de juegos serios.

2.3 Objetivos específicos

Para conseguir el objetivo general, antes descrito, será necesario cumplir con los siguientes objetivos específicos:

- Reconocer y explicar la tecnología Kinect y sus herramientas de desarrollo.
- Reconocer y explicar los conceptos de juegos serios e interfaz natural de usuario.
- Analizar y aplicar las metodologías de aprendizaje de piano y lecturas de música.
- Desarrollar una aplicación para el aprendizaje de piano mediante Kinect.

2.4 Descripción de la solución

El sistema que se busca implementar para cumplir con los objetivos definidos con anterioridad, posee una plataforma desarrollada con la tecnología Kinect permitiendo interactuar de forma directa con los usuarios. Esta plataforma consiste una aplicación de interacción natural donde el usuario mediante la utilización de sus manos podrá acceder a los contenidos de la aplicación.



Figura 2.1 Esquema gráfico de la solución.

Tabla 2.4 Fase prototipado, parte 2.

Id.	Nombre de tarea	Comienzo	Fin	Duración	may 2013											jun 2013																					
					19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
18	PROTOTIPADO	24/04/2013	25/06/2013	45d																																	
26	Implementar Lógica de Negocio	20/05/2013	07-06-2013	15d																																	
27	Testeo y Corrección de errores	10/06/2013	14-06-2013	5d																																	
28	Correcciones de Requerimientos	14/06/2013	14-06-2013	1d																																	
29	Entrega de prototipo	21/06/2013	21/06/2013	0d																																	
30	Entrega de Informe final	21/06/2013	21/06/2013	0d																																	

Tabla 2.5 Fase Evaluación del Prototipo.

Id.	Nombre de tarea	Comienzo	Fin	Duración	sep 2013											oct 2013														
					9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	1	2	3	4
1	EVALUACIÓN DEL PROTOTIPO	09-09-2013	09-10-2013	23d																										
2	Reunión con profesor guía	23-09-2013	23-09-2013	0d																										
3	Documentación de Abstrac y Resumen	26-09-2013	26-09-2013	1d																										
4	Realización de plan de prueba	26-09-2013	27-09-2013	2d																										
5	Estudio de nuevas librerías	30-09-2013	09-10-2013	8d																										
6	Reunión Jefa Departamento de Piano (IMUS)	09-10-2013	09-10-2013	0d																										

Tabla 2.6 Fase Refinamiento del Prototipo.

Id.	Nombre de tarea	Comienzo	Fin	Duración	oct 2013											nov 2013											dic 2013																			
					10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	1	2	3	4	5	6	7	8
7	REFINAMIENTO DEL PROTOTIPO	10-10-2013	13-12-2013	47d																																										
8	Pruebas de Librerías con dos Kinect	10-10-2013	22-10-2013	9d																																										
9	Estudios de la metodología de aprendizaje de piano	14-10-2013	16-10-2013	3d																																										
10	Reunión Profesor Guía	16-10-2013	16-10-2013	0d																																										
11	Diseño e implementación de interfaces de «Realizar Curso»	17-10-2013	25-10-2013	7d																																										
12	Entrega de informe de avance	25-10-2013	25-10-2013	0d																																										
13	Implementación de Lecciones	11-11-2013	11-12-2013	23d																																										
14	Pruebas Modulares	11-12-2013	12-12-2013	2d																																										
15	Pruebas de Integración	13-12-2013	13-12-2013	1d																																										

Tabla 2.7 Fase Producto de Ingeniería.

Id.	Nombre de tarea	Comienzo	Fin	Duración	dic 2013													
					12	13	14	15	16	17	18	19	20	21	22	23	24	25
16	PRODUCTO DE INGENIERÍA	13-12-2013	27-12-2013	11d														
17	Corrección de errores	13-12-2013	17-12-2013	3d														
18	Entrega de Software final	13-12-2013	13-12-2013	0d														
19	Revisión de Software	17-12-2013	17-12-2013	0d														
20	Corrección de errores de Software	17-12-2013	27-12-2013	9d														
21	Entrega de informe Final	27-12-2013	27-12-2013	0d														

Para ver el detalle de actividades Véase el anexo A.

3 Marco Teórico

En el presente punto se describirán las tecnologías y conceptos necesarios para comprender de manera adecuada toda la temática detrás del proyecto.

3.1 Definiciones

Serious Games: *Serious Games* o juegos serios -en español-, es un paradigma que consiste en juegos enfocados no sólo con el fin de entretenimiento, sino que primordialmente especializados en la enseñanza en distintos ámbitos de la educación, así como también para entrenar e informar.

Realidad Aumentada: Ésta se puede entender como la creación de una realidad mixta en tiempo real, la que es posible combinando la realidad misma con elementos virtuales. Esta es posible gracias a un conjunto de dispositivos que insertan información virtual a la información física ya existente, es decir, elementos virtuales insertados en la realidad, diferenciándose completamente de la realidad virtual, ya que ésta no reemplaza a la realidad física, sino que sobreimprime los datos virtuales al mundo real.

NUI: Interfaz Natural de Usuario o NUI (*Natural User Interfaces* en inglés), es la capacidad de interactuar con una máquina usando nada más que el cuerpo humano, esto quiere decir que no se necesita ningún dispositivo de mando o de entrada como podría ser un ratón, teclado alfanumérico, *joystick*, etc., pasando a ser el cuerpo humano el mando de control, utilizando tanto movimientos gestuales con las manos o con cualquier otra zona del cuerpo. En dispositivos con pantallas capacitivas multitáctiles, que también se consideran de interfaz natural de usuario, las yemas de los dedos son el control. Entre las interfaces naturales se considera el reconocimiento de voz, reconocimiento de escritura, *touch*, reconocimiento de movimiento y visual.

Kinect: El sensor Kinect fue presentado por Microsoft como un “controlador de juego libre y entretenimiento” desarrollado originalmente para la videoconsola Xbox 360 en el año 2010 y a partir del año 2011 para PC con Windows 7 y posteriormente Windows 8. El objetivo principal de ésta es permitir a los usuarios controlar e interactuar con la consola sin necesidad de un controlador de videojuegos convencional o de cualquier contacto físico con el dispositivo. Esto se consigue a través de una interfaz natural de usuario que es capaz de reconocer una gran cantidad de gestos, comandos de voz, objetos e imágenes.

3.2 Serious Games

El área de desarrollo y creación de videojuegos siguiendo el paradigma de juegos serios ha surgido como una manera inteligente de combinar los beneficios de los videojuegos, su poder de penetración en la población y las necesidades de educación y formación efectiva tanto a nivel político-institucional como empresarial y comercial. Entre las características distintivas de este tipo de videojuegos con relación a los videojuegos comerciales se encuentran:

- Están destinados para la educación, el entrenamiento en habilidades determinadas, la comprensión de procesos complejos, sean sociales, políticos, económicos o religiosos; también para publicitar productos y servicios.
- Están vinculados en forma evidente con algún aspecto de la realidad. Esto favorece la identificación del jugador con el área de la realidad que se está representando en el ambiente virtual, por ejemplo si se asume el rol en el juego de un dirigente político que debe tomar decisiones difíciles en las que se pone en peligro la vida de algunas personas.
- Constituyen un ambiente tridimensional virtual en el que se le permite una práctica "segura" a los aprendices en algunas áreas. En los casos de entrenamiento, por ejemplo, en el campo militar, se entrena a los soldados a manipular las armas.
- Hay intereses manifiestos en sus contenidos (políticos, económicos, psicológicos, religiosos, etc.).

Por otra parte, en cuanto a la entretención que podría brindar este tipo de videojuegos, depende -en muchos casos- de quién juega y para qué juega. Por ejemplo, en un videojuego para aprender a pilotar un avión, para el aspirante a piloto podría llegar a ser estresante no lograr superar los niveles del juego, sin embargo, para un joven cualquiera que practica con el videojuego sólo por placer o pasatiempo podría resultar muy divertido.

3.3 Realidad Aumentada

Como se mencionó en su definición, la realidad aumentada mantiene el mundo real del usuario enriqueciéndolo con la presencia de elementos virtuales. Esto se consigue siguiendo ciertos aspectos básicos y fundamentales para su correcto funcionamiento:

- Seguimiento muy preciso de la posición y movimiento del usuario.
- Precisar una adecuada fusión entre imágenes reales y objetos virtuales.



Figura 3.1 Realidad Aumentada: salida, entrada y registro.

Como es de suponer, para trabajar con esta tecnología se necesitan tres componentes esenciales: Visualización (salida), ubicación de objetos virtuales en el mundo real (registro), y métodos de interacción (entrada). Estos en conjunto funcionan de la siguiente forma: Una cámara captura la información del mundo real (entrada), el sistema de posicionamiento determina la posición y orientación del usuario en cada momento, con esta información se genera el escenario virtual que se va a mezclar con la señal de video capturada por la cámara para generar la escena aumentada (registro). Esta escena compuesta por la información real y la virtual se presenta al usuario a través del dispositivo de visualización (salida).

La realidad aumentada tiene aplicaciones en distintas áreas; entre ellas el área militar, aquí podemos encontrar distintos usos como BWC (*body wearable computer*), con este dispositivo es posible controlar en todo momento la posición de los compañeros, consultar mapas del terreno, calcular y adjudicarse objetivos, etcétera. Una de las grandes ventajas es que no ocupa un espacio adicional, pues se integra a la ropa o el cuerpo, y así como es utilizada en este campo, es posible aplicar en áreas que ayudan directamente a las personas, como es en el ámbito de la medicina, específicamente en las cirugías, pues es posible superponer en tiempo real la reconstrucción 3D de las estructuras internas del paciente sobre la imagen de video del mismo, reduciendo costos tanto económicos como tiempo durante las intervenciones. Lo mismo se puede realizar en otros campos mezclando la realidad y la virtualidad en áreas lúdicas, arqueológicas, pedagógicas, urbanísticas, etc.

3.4 Kinect

El sensor Kinect de Microsoft permite una manera diferente de interactuar con una computadora, utilizando 3 principales componentes -o llamados sensores- para su funcionamiento: Sensor de detección de video mediante una cámara web de alta resolución; detección de sonido utilizando un arreglo de micrófonos; y por último, un sensor que permite la detección de profundidad usando una cámara especial de profundidad. Este último sensor ha hecho que la Kinect sea lo que hoy en día es: un dispositivo innovador y especial; pues la tecnología del sensor detector de profundidad -desarrollado por PrimeSense- lo hace diferente, proyectando una luz infrarroja invisible para el ojo humano, en conjunto con otra cámara que es capaz de detectar esta luz, generando una especie de

mapa donde estamos ubicados y así poder medir la profundidad o distancias de los objetos en el ambiente.

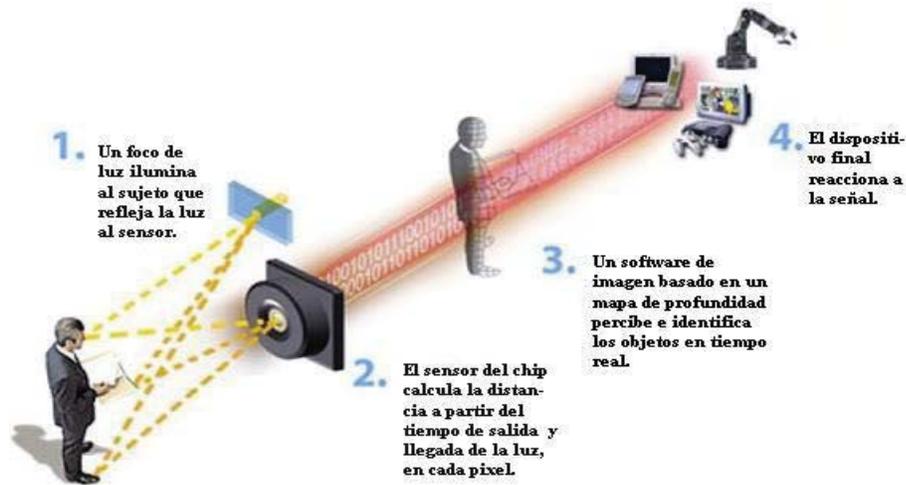


Figura 3.2 Funcionamiento de la Kinect

3.5 Interior de la Kinect

Dentro del sensor Kinect se puede apreciar una gran cantidad de *hardware* muy complejo; entre una colección de sensores y una seria cantidad de potencia de procesamiento, lo que además hace que exista un ventilador interno para mantener la electrónica fría de tanto procesamiento.



Figura 3.3 Vista Frontal Kinect

Como se aprecia en la Figura 9, el sensor Kinect posee una gran complejidad interna que es capaz de generar grandes funcionalidades y permite a los desarrolladores realizar aplicaciones mucho más futuristas, pudiendo cada vez más ir innovando en éstas. En el proyecto descrito en este informe se hará uso de esta tecnología, sin usar los sensores de audio o micrófonos, sino que se utilizará tanto el sensor de video como el de profundidad.

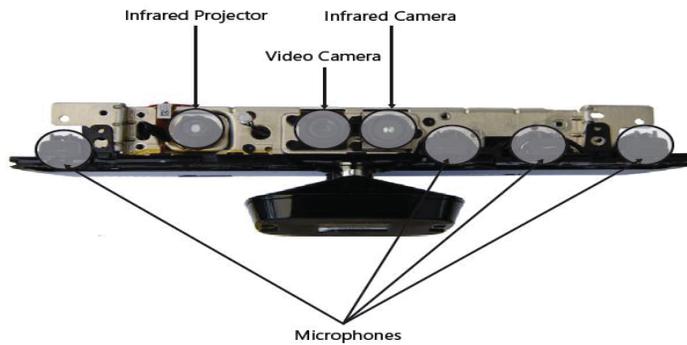


Figura 3.4 Vista Frontal Interna Kinect

3.6 Interfaz Natural de Usuario

La Interfaz Natural de Usuario o NUI, como se mencionó en la definición, es la capacidad de interactuar con una máquina utilizando únicamente partes del cuerpo humano como dispositivo de mando; esto quiere decir que se evitan controles tipo *joystick*, ratón, teclado, etcétera; pasando a ser sólo zonas del cuerpo humano el “dispositivo”. Algunos ejemplos que se pueden encontrar en muchos sectores, son los actuales dispositivos móviles con la tecnología *touch*; éstos poseen una interfaz natural de usuario tomando como dispositivo los propios dedos de una persona, pues éstos manejan naturalmente las aplicaciones y el sistema de los dispositivos en sí. Además, siguiendo con el mismo tipo de NUI, se puede encontrar toda la serie de dispositivos *touch* que existen en el mercado: *tablets*, televisores con capacidad *touch* y en definitiva, toda la gama de pantallas táctiles.



Figura 3.5 Interfaz Natural de Usuario sobre pantalla táctil

Continuando con lo anterior, además se puede encontrar interacciones NUI usando el dispositivo y sensor Kinect -el cual será utilizado en este proyecto-. Esta interacción mediante este dispositivo precisa en el manejo con todo el cuerpo humano o partes de éste, como lo son las manos; todo depende del tipo de aplicación el cual se maneje, pero en general, la utilización del dispositivo Kinect permite una interacción a cuerpo completo con la máquina o lo que se esté llevando a cabo en el momento; interactuando con las manos, con los brazos, piernas, incluso con la cabeza.

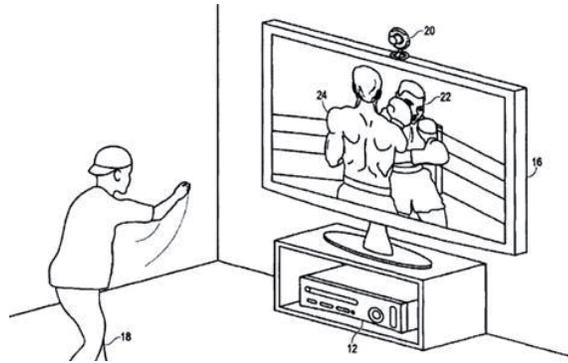


Figura 3.6 Interfaz Natural de Usuario mediante Kinect

3.7 Metodología aprendizaje de Piano

3.7.1 Metodología aprendizaje

Como se mencionó anteriormente, la metodología erradica desde el libro “Piano básico de Bastien, nivel 0 elemental”, en conjunto a las recomendaciones de los especialistas. Para el proyecto actual, se implementarán 4 lecciones en total, integradas en un “Curso”.

Tabla 3.1 Metodología de aprendizaje de Piano - Lecciones.

N°	Título Lección	Descripción
1	Numeración de dedos y posición de manos.	En esta lección el usuario podrá conocer la posición correcta de las manos frente al teclado, esto indicando el número de cada dedo y posicionándolo correctamente sobre las teclas que corresponden. Cada dedo de una mano lleva un número; el pulgar es el número 1, índice el 2, medio el 3, anular el 4 y finalmente el meñique el 5. Misma numeración en cada mano. [9]
2	Digitación. Aprendiendo agudas y graves.	El usuario comenzará a digitar sobre el piano, conociendo el sonido de las teclas agudas y graves, con un ejercicio de ritmo para lograr identificar estos sonidos. [9]
3	Notas musicales en el piano.	El usuario podrá aprender las notas musicales básicas sobre el piano. Éstas son el Do, Re, Mi, Fa, Sol, La y Si. Para esto, primeramente se indicarán las notas sobre el piano para así reconocerlas, tanto desde el “Do central” hacia la izquierda como hacia la derecha. Luego se ejercerá una breve canción de ejemplo con el uso de éstas. Aquí el usuario podrá elegir con qué mano trabajar la canción. [9]
4	Revisión de los conocimientos.	En esta lección, el usuario será puesto a prueba con algunas preguntas acerca de lo aprendido con anterioridad. Esto con el fin de reforzar los conocimientos adquiridos en lecciones anteriores. [9]

3.8.1 Reunión inicial con especialistas en Piano

En la fase de refinamiento de prototipo se llevó a cabo una reunión con profesores especialistas en la enseñanza de piano con el fin de conocer en profundidad lo que es la metodología y el aprendizaje básico de este instrumento.

A continuación se da una breve información acerca de los docentes:

- **Manuel Montero.** Intérprete Musical en Piano, Universidad de Chile y Pontificia Universidad Católica de Valparaíso. Postgrado en Hochschule für Musik “Mozarteum”, Salzburgo, Austria. A la fecha, docente de piano en el Instituto de Música de la Pontificia Universidad Católica de Valparaíso. [10]
- **María Angélica Beláustegui.** Intérprete superior en piano, Universidad de Chile. A la fecha, docente de piano en el Instituto de Música de la Pontificia Universidad Católica de Valparaíso. [10]
- **Patricia Escobar.** Intérprete musical mención Piano. Universidad de Chile. Profesora especializada en piano básico. A la fecha, jefa departamento de piano y docente del Instituto de Música de la Pontificia Universidad Católica de Valparaíso. [10]

En esta reunión los especialistas mostraron un gran interés y curiosidad por el proyecto, puesto que era algo novedoso en lo que es tecnología. Se dio una breve charla acerca del proyecto y objetivos esperados, mencionando el aspecto práctico del aprendizaje de piano. Luego, éstos dieron críticas constructivas y recomendaciones generales en cuanto al piano virtual, entre ellos:

- Crear un piano virtual lo más parecido a uno real.
- Como recomendación usar mínimo 2 octavas de teclas (conjunto de 8 teclas).
- Usar como referencia la tecla “Do central” para la mitad del piano.
- Mostrar -en alguna lección- la numeración de dedos de acuerdo a la metodología de aprendizaje.
- Mostrar -en alguna lección- las notas musicales en las teclas.

Por último, se recomendó seguir la línea metodológica del libro “Piano básico de Bastien, nivel 0 elemental”, para realizar las lecciones, comprendidas en un máximo de 3 para un aprendizaje básico e inicial del instrumento. Finalizando con una cuarta lección de revisión y diagnóstico de los conocimientos.

3.8.2 Reunión final con especialistas en Piano

Ya en la fase final del proyecto, se coordinó una reunión con los mismos especialistas, con el fin de dar a conocer el *software* ya terminado y para evaluar, desde su perspectiva, lo que son las lecciones que se aplicaron a este. Además, para poder conocer las críticas, problemas y/o trabajos a futuro o recomendaciones que pudiesen existir. En esta reunión el equipo de trabajo volvió a presentar la aplicación, recordando la finalidad y objetivos. Sin embargo, se hizo el enfoque principal en el curso de piano, mostrando cada lección una por una y explicando el motivo de cada propuesta aplicada. Los especialistas presenciaron cada lección, y fueron dando sus opiniones mientras se mostraba la aplicación. Cabe destacar que en esta oportunidad no pudo asistir el docente Manuel Montero.

Al finalizar la presentación del *software* frente a los especialistas, uno de ellos interactuó directamente con el piano virtual en la funcionalidad “Piano Libre”, pues encontraban algo raro. Al probar nuevamente esta funcionalidad, se pudo notar que había un desfase de sonido en las teclas negras. Se empezó a analizar tecla por tecla; blancas y negras, para ir haciendo un análisis del sonido de cada una. Se pudo concluir que las blancas estaban bien, sin embargo las negras estaban desfasadas hacia tonos más agudos. Un error que había pasado de ser percibido por el equipo de trabajo y que posterior al encuentro fue solucionado.

Además del problema mencionado anteriormente, se dieron un par de recomendaciones para un trabajo futuro y/o mejora del curso. Estas fueron:

- La posición de manos, en la lección 1, adaptarla a las siguientes lecciones; esto quiere decir, que no sea al centro, sino posicionando las manos para tocar la canción “Los Globos”, presente en la lección 3. Sino, como alternativa, dejar la posición de la mano izquierda a la altura del Sol y la derecha al Do Central (como estaba), e indicar que se posicione el dedo pulgar, no la mano completa, pues puede causar confusión, ya que al pedir que se posicione la mano completa, ésta abarca demasiado, sin saber qué parte de la mano posicionar exactamente.
- Al momento de indicar que se toque alguna canción con la mano izquierda, cambiar la posición de las “plicas” (que son las figuras musicales, como la negra o la blanca). Esto para un mejor aprendizaje, pues se suele asociar las plicas en posición invertida para la mano izquierda, y como se conoce normalmente, para la mano derecha. Mencionando esta diferencia en algún apartado.

Concluyendo, se puede decir que esta reunión fue de suma importancia, puesto que se encontró un problema que podría considerarse grave para un aprendizaje y se dieron recomendaciones para un futuro. El equipo de trabajo salió muy contento por la buena aceptación que tuvo el *software*, pues a pesar del problema encontrado, se distinguió la novedosa y llamativa forma de enseñar el instrumento de piano.

4 Especificación de requerimientos

A continuación se hace un breve listado de los requisitos para la realización de este proyecto, los cuales están divididos en funcionales, no funcionales.

4.1 Funcionales

- Detectar movimiento de dedos con el dispositivo Kinect.
- Aplicar acciones a cada movimiento captado, específicamente a la presión de los dedos.
- Crear un piano virtual el cual genere sonidos en conjunto a la presión de los dedos.
- Brindar la posibilidad de interactuar con el piano virtual libremente.
- Dar la opción de seguir un curso básico con el piano, mediante diferentes lecciones.
- Sistema que permita al usuario almacenar su puntaje y progreso de acuerdo a la lección que está realizando.
- Dar la posibilidad de elegir qué mano se quiere perfeccionar durante la lección, si la lección lo permite, o utilizar ambas manos de forma predeterminada.

4.2 No funcionales

- Seguridad en control de acceso de usuarios no autorizados por el sistema.
- Tiempo de respuesta aceptable de la aplicación con el usuario; no más de 2 segundos de retraso.
- Usabilidad (Interfaz Natural de Usuario). Definiendo el tiempo de entrenamiento en 4 horas.
- Dispositivo Kinect funcionando correctamente en computadora del usuario.
- Posición correcta de la Kinect, considerando distancia e iluminación adecuada.
- Poseer conocimientos mínimos de música; esto involucra notas musicales básicas.

4.3 Diagrama Casos de Uso

Un caso de uso es una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso. A continuación se muestra las funcionalidades que deberá cumplir el sistema.

4.3.1 Diagrama General del Sistema

La aplicación que se va a desarrollar con la ayuda de la tecnología Kinect cuenta con las siguientes funcionalidades, las cuales se detallarán en el modelo de caso de uso que se mostrará a continuación.

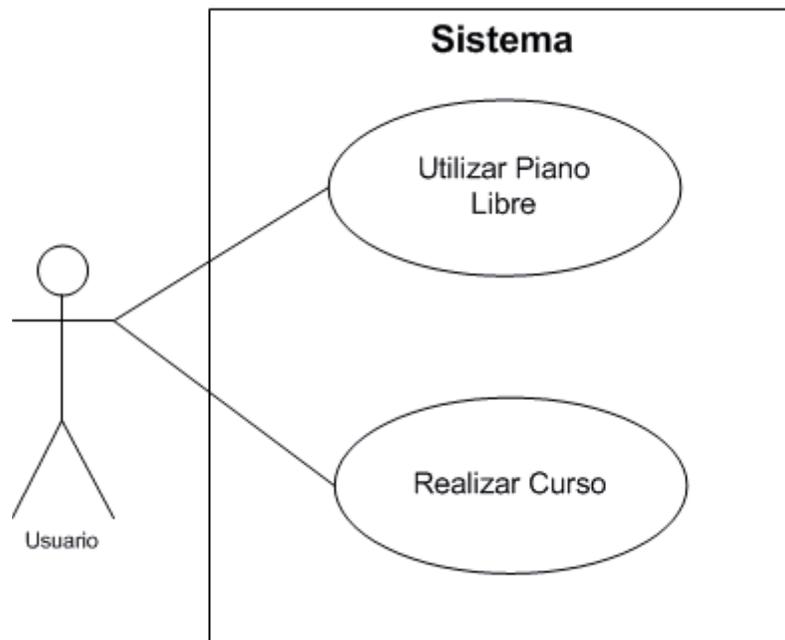


Figura 4.1 Diagrama general de casos de uso.

4.3.2 Caso de uso Utilizar Piano Libre

El usuario mediante la utilización de la Kinect y la detección de sus movimientos podrá acceder a la funcionalidad Utilizar Piano Libre.

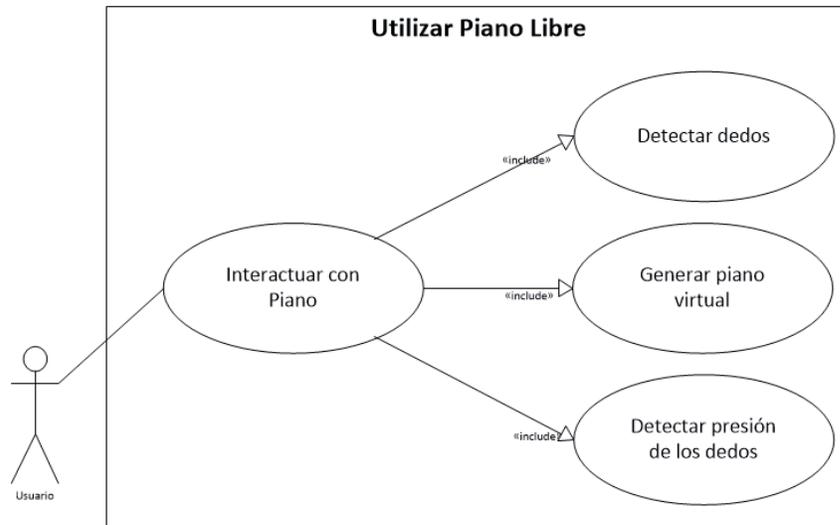


Figura 4.2 Caso de uso Utilizar Piano Libre

4.3.3 Caso de uso Realizar Curso

El usuario mediante la utilización de la Kinect y la detección de sus movimientos podrá acceder a la funcionalidad Realizar Curso.

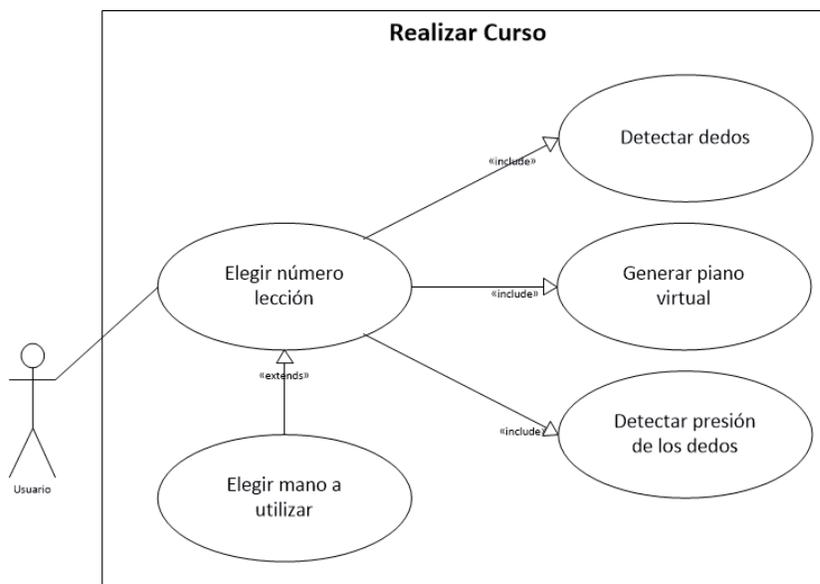


Figura 4.3 Caso de uso Realizar Curso.

Tabla 4.1 Caso de uso específico: Elegir número lección.

Caso de uso	2.1 Elegir número lección
Actor Principal	Usuario
Participantes e intereses	El usuario mediante la realización de este caso de uso, busca aprender una lección de piano
Precondiciones	Usuario seleccionado en el sistema.
Postcondiciones	Se guarda la información relevante de la lección. Puntaje.
Escenario Principal	<ol style="list-style-type: none"> 1.- El usuario selecciona una lección a realizar. 2.- El sistema analiza, encuentra y reconoce los dedos del usuario. 3.- El sistema genera un piano virtual. 4.- El sistema asigna una nota musical a cada tecla del piano. 5.- Se cargan los datos de acuerdo a la lección seleccionada. 6.- El usuario comienza a practicar una lección. 7.- El usuario realiza la lección. 8.- El sistema almacena la información de la lección realizada.
Extensiones	<ol style="list-style-type: none"> 2.- El sistema no reconoce los dedos del usuario. 2.- El sistema vuelve a detectar los dedos del usuario hasta reconocerlos. Al reconocerlos continúa. 5.- El sistema no puede cargar información de la lección. El sistema manda un mensaje de error informando que no se puede obtener la información. 6.- El sistema detecta que la lección tiene la posibilidad de practicar manos independientes. El sistema ofrece al usuario la posibilidad de elegir mano a practicar. 6.- El usuario selecciona mano a practicar. 6.- El sistema carga lección de acuerdo a las características seleccionadas.
Requisitos Especiales	
Frecuencia de ocurrencia	Alta.

Para llevar a cabo los requerimientos anteriores, el equipo de trabajo tuvo que realizar un análisis de factibilidad, para proyectar cuán factible es realizar un sistema de estas características. Para ver esta factibilidad vaya al anexo B.

4.4 Análisis de alternativas para implementación

Dentro de las actividades que corresponden al presente proyecto, el cual culminará con el diseño y posterior construcción del sistema se posee un estudio que define la manera en que distintas opciones se adaptan a lo que requiere el equipo de trabajo. En otras palabras, se establecen alternativas con diferentes entornos de trabajo para desarrollar la aplicación.

- **Alternativa de Software Libre**

Se define que esta opción no representará gastos en herramientas de desarrollo para la realización del sistema. Se plantea la utilización del IDE (Entorno de Desarrollo Integrado) Eclipse, utilizando el SDK desarrollado para Java llamado JNECT. De esta manera, lograr la construcción de la arquitectura de software especificada en capítulos posteriores. Además, como motor de base de datos se define el uso de PostgreSQL, alternativa de libre acceso que entrega grandes prestaciones. Se usará sistema operativo Linux para esta opción.

- **Alternativa de Software Pagado**

En esta alternativa se plantea el uso de herramientas que significarán una mayor inversión para que el equipo de trabajo pueda desarrollar su labor. En primer lugar, se determina que el motor que gestionará el sistema de bases de datos será Microsoft SQL Server Compact Edition que, puede ser conseguido de forma gratuita mediante la página oficial de Microsoft. En segundo lugar, se determina el uso de un sistema Operativo Windows para establecer sobre éste el entorno de trabajo de esta alternativa. En último lugar, se define el uso del IDE Microsoft Visual Studio que, aunque en un principio es pagado para su uso, utilizando el medio de DreamSpark.com, sitio de Microsoft el cual facilita a estudiantes diversos *software* -como el anterior-, se obtiene sin costo alguno [4].

Para ver los resultados del presente análisis de alternativas para implementación consulte el Anexo B.

5 Solución propuesta

5.1 Arquitectura

Evaluando las características que la aplicación debe satisfacer, especificadas mediante requerimientos capturados en el proceso de análisis; se determina que debe contar con una arquitectura lógica de tres capas o layers.

A continuación se presentarán las tres capas con las cuales contará el sistema:

- **Capa presentación:** Este nivel es el que interactúa de forma directa con los usuarios. Es la lógica que presenta el sistema y sus funcionalidades. Indica qué información pedir y cómo almacenarla. Esta capa también es conocida como interfaz gráfica, que debe ser amigable, fácil de entender y utilizar para el usuario. Además, se relaciona directamente con la capa de Procesamiento.
- **Capa procesamiento:** Es en donde residen los programas que se ejecutarán, donde se reciben las peticiones del usuario y se envían las respuestas después de realizarse el proceso. Además, en esta capa se presentan todas las reglas que deben cumplirse a lo largo de la ejecución del sistema. Esta capa se comunica directamente con la responsable de la interfaz con el usuario, para recibir peticiones y otorgar respuestas a éstas. Asimismo, interactúa con la capa de almacenamiento de datos para guardar o recuperar los datos en la base de datos.
- **Capa acceso a datos:** En esta capa es donde residen los datos. Está compuesta por uno o más gestores de bases de datos que realizan actividades tales como recibir las solicitudes de almacenamiento de los datos, o la recuperación de éstos desde la capa de procesamiento.

Mientras que de manera física, la aplicación constará con una Kinect para la detección de los movimientos del usuario y un PC para realizar las tareas de procesamiento de información y almacenamiento de ésta.

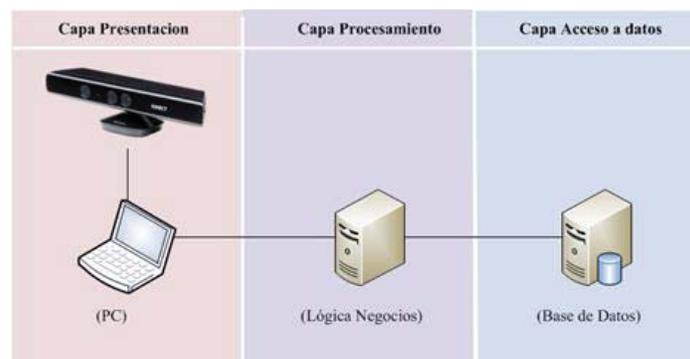


Figura 5.1 Arquitectura Lógica.

5.2 Metodología

La metodología que se decidió utilizar para este proyecto es la realización de prototipado. Esta metodología fue elegida porque se ajusta de mejor manera a las necesidades del proyecto, donde se está trabajando con tecnologías innovadoras, centradas en las necesidades del usuario. Para esto se realiza una captura de requerimientos generales o globales para la construcción de un prototipo inicial y de la manera más rápida posible. Además, la realización de este prototipo ayuda a crear una retroalimentación entre el usuario y el desarrollador [8]. Cabe destacar que la realización de este prototipo debe ser con la misma tecnología que se va a utilizar para la versión final pero, utilizando la menor cantidad de recursos posibles.

Este marco teórico es evolutivo, cada vez que un prototipo se ajusta a las necesidades del usuario se realiza una iteración a la versión final. La metodología ayuda a que el desarrollador comprenda de mejor manera lo que se busca desarrollar y que el usuario vea resultados a corto plazo.

Dentro de las ventajas de la utilización de este marco teórico, podemos encontrar que:

- Este modelo es útil cuando el cliente conoce los objetivos generales para el software, pero no identifica los requisitos detallados de entrada, procesamiento o salida. [8]
- También ofrece un mejor enfoque cuando el responsable del desarrollo del software está inseguro de la eficacia de un algoritmo, de la adaptabilidad de un sistema operativo o de la forma que debería tomar la interacción humano-máquina.
- Disminuye el riesgo de construir un software que no se ajuste a las necesidades del usuario.
- Exige al desarrollador contar con las herramientas necesarias para la creación de prototipos.
- Respeta el ciclo de vida del software.
- Ayuda a que el usuario sienta mayor grado de aceptación con el software desarrollado.
- Presenta una reducción de costos y de tiempo.
- Además ayuda a mejorar la comunicación entre el usuario final y el desarrollador.

Dentro de esta metodología podemos encontrar dos tipos de prototipos.

- Desechables: De vida útil corta. Son utilizados para realizar ajustes a los requerimientos globales. Además ayuda a construir de mejor manera las interfaces gráficas, realizando interfaces más completas y amigables para los usuarios. [8]
- Evolutivos: Prototipo que es capaz de transformarse en un software final. Sin embargo, este software no cuenta con una buena documentación y calidad. [8]

Dado las características del proyecto que se va a realizar, los prototipos que se van a utilizar serán forma principal evolutivos. Es por eso que a lo largo de la construcción del proyecto en cuestión, se realizarán tres prototipos de forma aproximada. Lo anterior, se explica por la cantidad de entregables que hay que tener durante el año. Además, de esta manera se busca realizar un software que pueda ajustarse de la manera más cercana posible a los requerimientos del proyecto, otorgando así un buen producto de software.

A continuación se presentará de manera gráfica las fases a seguir para realizar la construcción de este tipo de software.

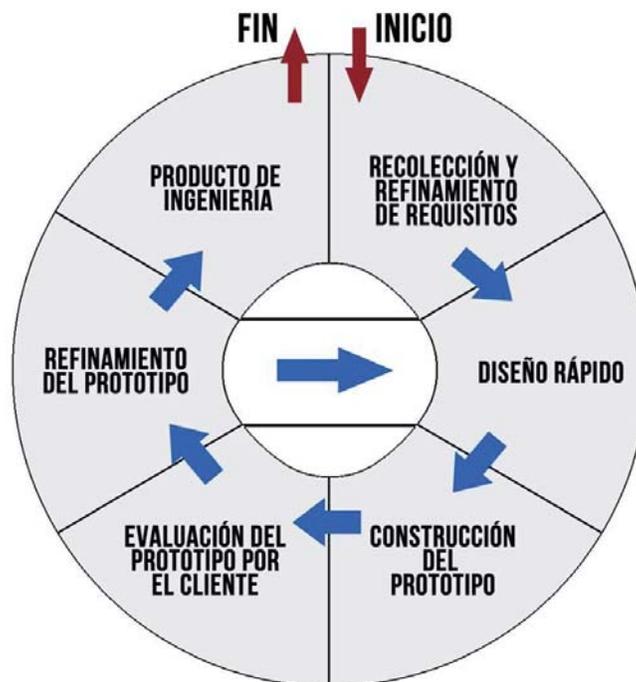


Figura 5.2 Etapas Metodología de Prototipado

5.3 Plataforma de Desarrollo

Para lograr desarrollar la aplicación, se debe contar con una serie de lenguajes o metalenguajes ya sea, para la realización de la capa de presentación, la capa lógica y la capa de datos. En el caso específico de esta aplicación la capa de presentación será desarrollada en XAML.

Lo anterior se debe a que, la tecnología Kinect presenta dicho lenguaje para la realización de interfaces de usuario de manera más intuitiva. De esta manera, cumplir con los estándares de calidad aceptables para la realización de una aplicación con la tecnología antes mencionada. También, este lenguaje permite al desarrollador, buscar que los movimientos con los que usuario interactúa con la aplicación sean más naturales, y no obligar al usuario a realizar movimientos que pudiesen ser incómodos para lograr la interacción.

La capa lógica será desarrollada en C# debido a que este lenguaje de programación ofrece todas la ventajas de la programación orientada a objetos. Por lo que, el resultado de la construcción de la aplicación será un software más robusto y acorde al realidad que se plantea modelar. Este lenguaje ayuda a la utilización de librerías que ayudan a la detección de dedos, y de esta manera lograr que el código sea más homogéneo y escalable en el tiempo.

La capa de datos será desarrollada en Microsoft SQL Server Compact Edition. Pese a que el equipo de trabajo no cuenta con experiencia en este motor de base de datos, es el que se ajusta de mejor manera a las características del proyecto. Dado que, permite la implementación de una base de datos local sin la conexión a ésta mediante la dirección IP.

5.4 Diseño de la Solución

5.4.1 Diagrama de Actividad Realizar Curso

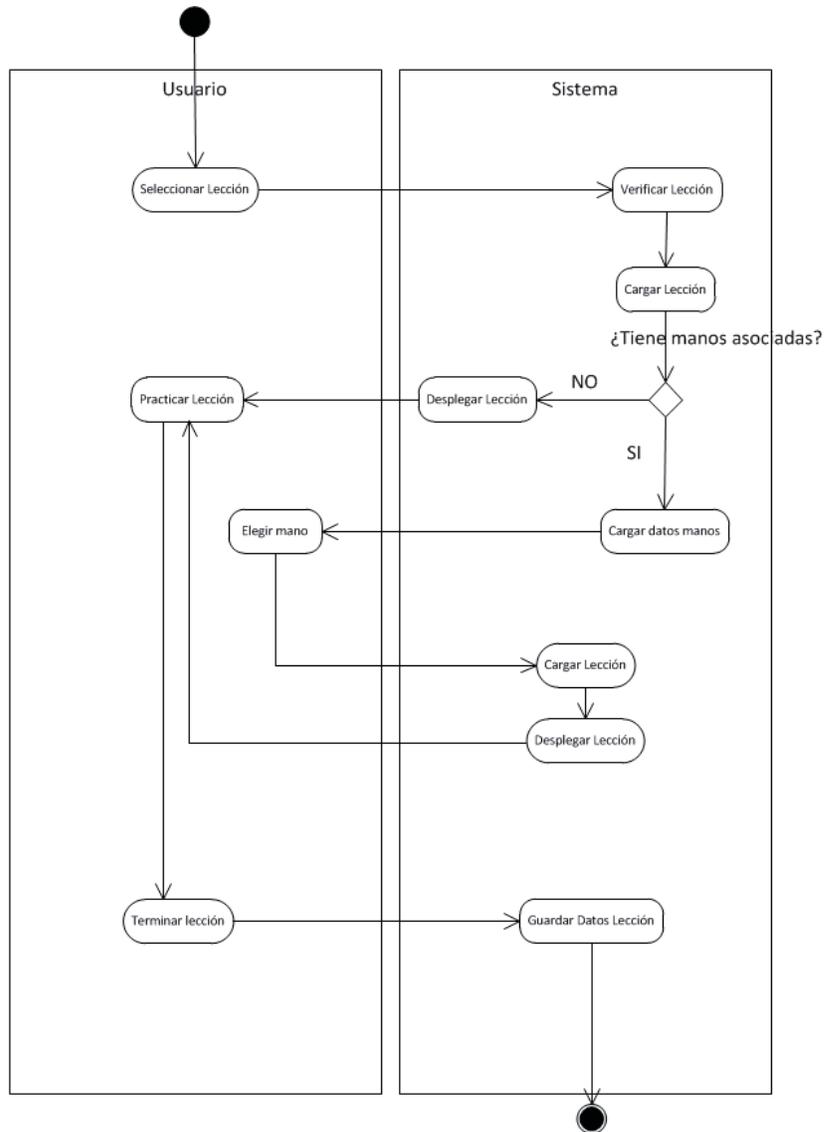


Figura 5.3 Diagrama de actividad Realizar Curso

La utilidad de usar este diagrama de actividad, es mostrar la interacción que debe realizar el usuario al momento de querer realizar una lección. Muestra que al momento de elegir una lección, ésta podría contar con la opción de elegir una mano a practicar, por lo que se muestra las actividades que el sistema debe realizar para cumplir con dicha tarea. Este diagrama permite tener una visión un poco más específica de cómo debe funcionar el sistema, para cumplir con las características de las lecciones implementadas.

5.4.2 Diagrama de Actividad Piano Libre

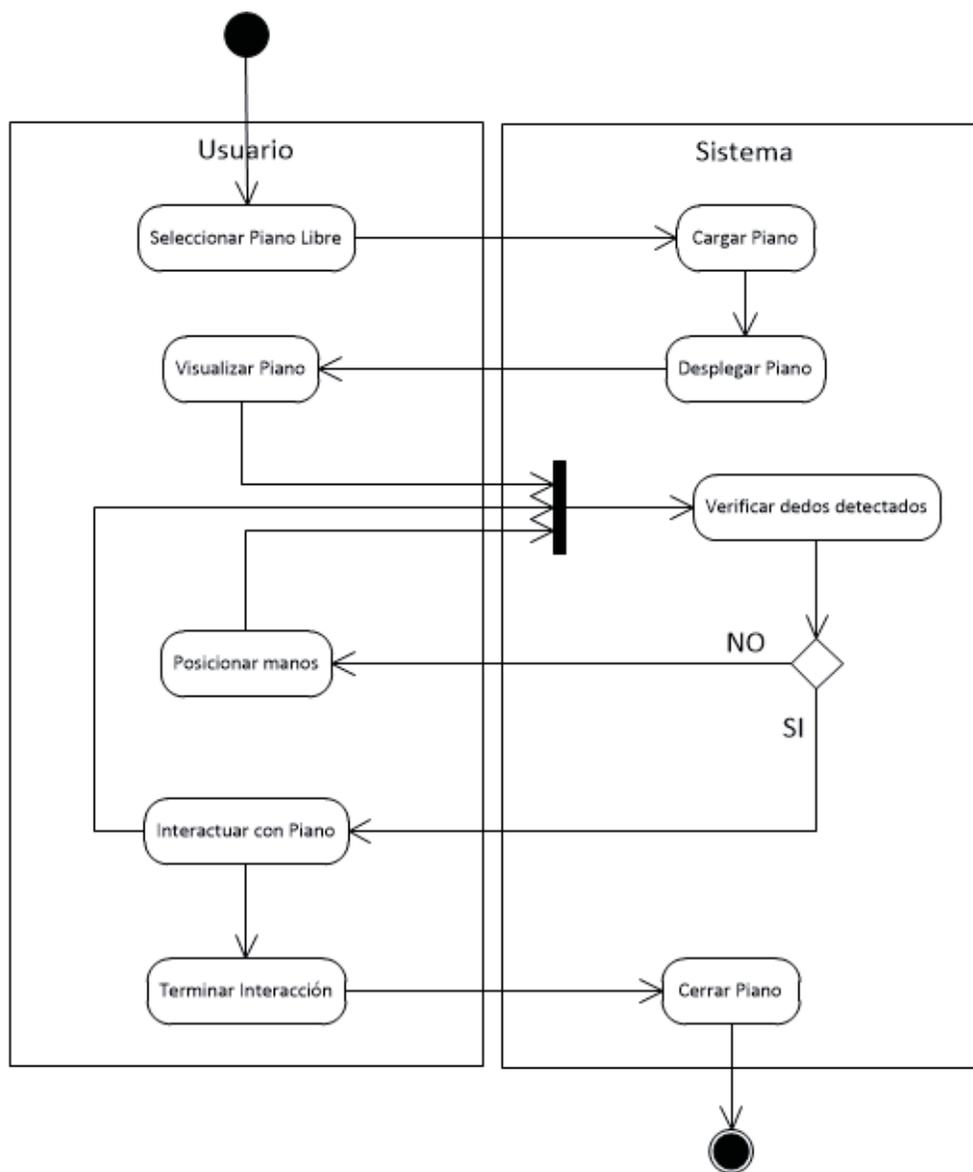


Figura 5.4 Diagrama de actividad Piano Libre

La utilidad de usar este diagrama de actividad es, mostrar en mayor detalle la funcionalidad de Piano Libre. En esta funcionalidad, lo primero que se debe realizar es cargar un piano en la pantalla para que el usuario pueda interactuar con él. Luego, se debe verificar que las manos, y los dedos, sean detectados por el sistema, de esta manera se podrá utilizar el piano sin problemas. Sin embargo, en caso de que las manos no sean detectadas, será necesario posicionar las manos nuevamente y el sistema contorneará éstas como aviso que están siendo detectadas.

5.4.3 Diagrama de Relacional

A continuación se presenta el esquema de tablas y relaciones entre éstas, las cuales serán el punto de partida para la posterior implementación de la base de datos que dará soporte al proyecto, desarrollado por el equipo de trabajo. En el siguiente diagrama se pueden identificar las cuatro entidades que se encuentran en el anterior modelo de datos. También se denotan los campos que forman parte de cada tabla, estableciendo de forma clara cuál será el identificador (o clave primaria) de cada una y los que provienen de otras tablas dentro de la base (claves foráneas).

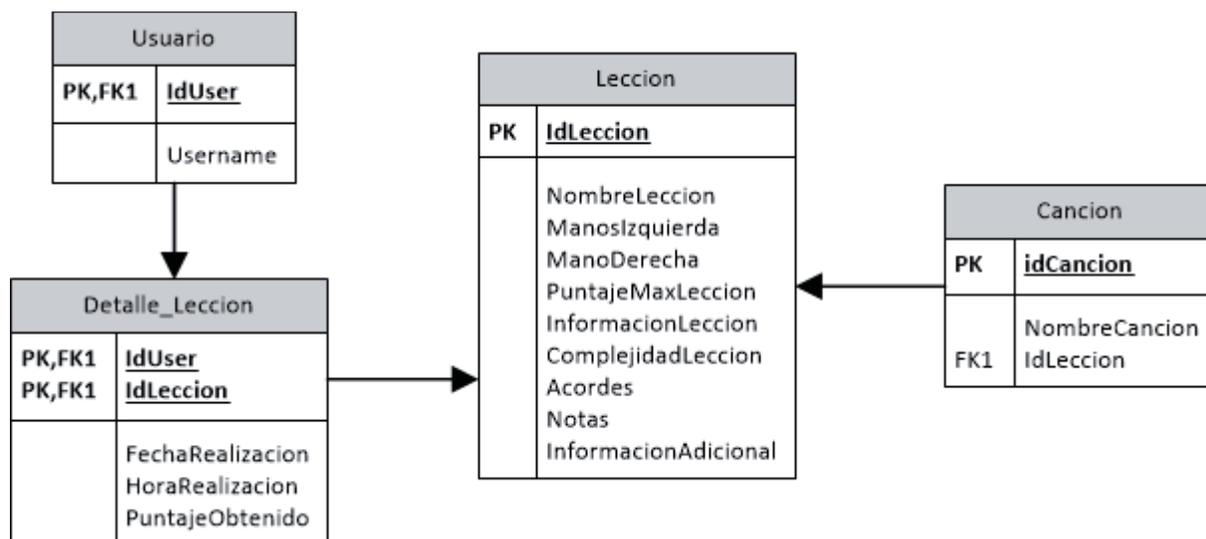


Figura 5.5 Diagrama Relacional

Para ver el diseño de interfaces Véase el Anexo C

6 Pruebas de Software

La creación de un plan y diseño de pruebas para la aplicación, permiten comprobar el correcto funcionamiento de este sistema. Para ello, se revisarán las interfaces y la codificación del sistema, buscando errores y corrigiéndolos. Lo anterior, se realizará utilizando, de manera principal, cuatro tipos de prueba: validación de datos, revisión de usabilidad de las interfaces, pruebas de caja negra y de caja blanca.

La idea fundamental de usar estos cuatro tipos de pruebas es: dado que, el que usará el sistema será de forma principal el usuario, por lo que las interfaces deben ser amigables y fáciles de utilizar, para que éste pueda usar el sistema de la manera más productiva posible, permitiendo de esta manera que el usuario logre el objetivo principal de la aplicación. Además, se usarán las pruebas de caja negra para confirmar que los datos de salida sean los esperados, utilizando datos de entrada correctos. En caso de que al ingresar datos de entrada correctos y la salida no sea la esperada se utilizará prueba de caja blanca, donde este tipo prueba busca encontrar el lugar específico donde los datos son alterados.

En primer lugar se planifican las pruebas antes mencionadas, luego se especifican las funcionalidades del sistema que serán puestas a prueba. Luego, se establecen los datos de entrada que serán utilizados en para obtener la respuesta esperada. Además de lo anterior, se deben establecer una serie de metodologías de pruebas, que permitan comprobar que las funcionalidades probadas sean realizadas de forma correcta por el sistema.

6.1 Planificación de pruebas

Para poder obtener la planificación, se debe tener conocimiento de las distintas formas en las que las funcionalidades serán probadas. Luego de conocer e identificar los procedimientos a realizar, se evalúan los datos que servirán para probar las funcionalidades del sistema. Las funcionalidades que se probarán serán:

Tabla 6.1 Plan de Prueba Creación de cuenta de usuario.

Plan de Prueba Creación de cuenta de usuario	
Identificador del plan	PP-CreaCuentaUsuario
Alcance	Se probará la correcta creación de cuenta de usuario.
Items a probar	La cuenta de usuario debe haber sido creada y almacenada en el sistema.
Estrategia	Se realizarán pruebas de validación y verificación de los datos ingresados para la creación de la cuenta.
Tangibles	Especificaciones de casos de pruebas.

Procedimientos especiales	<ul style="list-style-type: none"> Se debe haber iniciado el programa con la Kinect conectada y funcionando.
Recursos	<ul style="list-style-type: none"> Sistema operativo con Windows 7 u 8, de 64 bits. Kinect en perfecto estado. En buen lugar con iluminación adecuada.
Calendario	<ul style="list-style-type: none"> Comienzo: 10 de Noviembre del 2013. Duración: 2 días. Hito: Revisión total de la creación de cuentas.
Manejo de riesgos	No existen riesgos en cuanto a la manipulación de datos, ya que no existe la condición necesaria para que -con los datos ingresados- se generen modificaciones drásticas en la aplicación, más allá de un usuario nuevo.
Responsables	Luis Riquelme

Tabla 6.2 Plan de Prueba Interactuar Piano Libre.

Plan de Prueba Interactuar con Piano Libre	
Identificador del plan	PP-PianoLibre
Alcance	Se inspeccionará el funcionamiento correcto al interactuar con Piano Libre.
Items a probar	El piano virtual debe generar sonidos ad hoc a los movimientos de los dedos con el sensor Kinect.
Estrategia	Se realizarán pruebas de ensayo y error frente a diversas situaciones que se pudiesen presentar al momento de interactuar con el piano virtual.
Tangibles	Especificaciones de casos de pruebas.
Procedimientos especiales	<ul style="list-style-type: none"> Se debe haber iniciado el programa con la Kinect conectada y funcionando.
Recursos	<ul style="list-style-type: none"> Sistema operativo con Windows 7 u 8, de 64 bits. Kinect en perfecto estado. En buen lugar con iluminación adecuada.
Calendario	<ul style="list-style-type: none"> Comienzo: 28 de Noviembre del 2013. Duración: 2 días. Hito: Revisión total de la funcionalidad Piano Libre
Manejo de riesgos	No existen riesgos en cuanto a la manipulación de datos, ya que no se hacen modificaciones a éstos.
Responsables	Luis Riquelme

Tabla 6.3 Plan de Prueba Interacción con lecciones del curso.

Plan de Prueba Interacción con lecciones del curso	
Identificador del plan	PP-InteraccionLecciones
Alcance	Se probará la interacción con cada lección del Curso.
Items a probar	Las lecciones del curso deben ejecutarse adecuadamente al interactuar con el piano virtual.
Estrategia	Se realizarán pruebas a cada lección del curso, con el fin de corroborar que la acción de cada lección corresponda y sea ejecutada correctamente frente a la interacción con el usuario mediante el dispositivo Kinect.
Tangibles	Especificaciones de casos de pruebas.
Procedimientos especiales	<ul style="list-style-type: none"> • Se debe haber iniciado el programa con la Kinect conectada y funcionando.
Recursos	<ul style="list-style-type: none"> • Sistema operativo con Windows 7 u 8, de 64 bits. • Kinect en perfecto estado. En buen lugar con iluminación adecuada.
Calendario	<ul style="list-style-type: none"> • Comienzo: 29 de Junio del 2013. • Duración: 2 días. Hito: Revisión total de las lecciones del curso.
Manejo de riesgos	La aplicación correcta de la lección generará modificaciones en el puntaje del usuario, por lo que hay manipulación de datos.
Responsables	Luis Riquelme

6.2 Especificación de Casos de Prueba

En este apartado se muestran los casos de prueba correspondientes a los planes mencionados anteriormente. La finalidad es tener un conjunto de condiciones o variables con las que se pueda determinar si un requisito de la aplicación es parcial o completamente satisfactorio. Para esto, se explica cómo realizar los diferentes escenarios identificados en el plan de pruebas correspondiente.

6.2.1 Caso de prueba Creación cuenta de usuario

En este apartado se muestran los casos de prueba correspondientes a los planes mencionados anteriormente. La finalidad es tener un conjunto de condiciones o variables con las que se pueda determinar si un requisito de la aplicación es parcial o completamente satisfactorio. Para esto, se explica cómo realizar los diferentes escenarios identificados en el plan de pruebas correspondiente.

Tabla 6.4 Caso de Prueba Creación cuenta de usuario.

Visión general	
Identificador	P1.0
Creador de caso de prueba	Luis Riquelme
Versión	1.0
Nombre	Caso de prueba Creación cuenta de usuario
Propósito	Verificar que los usuarios se crean correctamente en el sistema. Además validar el ingreso de datos inválidos.
Dependencias	El teclado del computador debe estar funcionando.
Actividades	
Ambiente de prueba/ configuración	Ambiente de desarrollo de Visual Studio.
Inicialización	<ul style="list-style-type: none"> • Ingresar a la aplicación. • Ir al menú usuarios y presionar sobre el botón agregar.
Finalización	<ul style="list-style-type: none"> • Si no se visualiza el usuario agregado, se debe verificar su existencia en la base de datos. • Si nombre de usuario se almacena con datos inválidos, se debe verificar en la base de datos para descartar incongruencia de datos ante la visualización de éste.
Acciones	<ul style="list-style-type: none"> • Escribir un nombre de usuario mediante el teclado.
Datos de entrada	<ul style="list-style-type: none"> • Nombre de usuario con un mínimo de 4 caracteres y máximo de 14. Sólo letras, sin signos ni caracteres especiales.
Resultados	
Salida esperada	Notificación de usuario creado satisfactoriamente.
Salida obtenida	Notificación de usuario creado satisfactoriamente.
Resultado	Correcto.
Severidad	Mayor.
Seguimiento	No se encontraron defectos en este caso de prueba.
Estado	Terminado.

6.2.2 Caso de prueba Interactuar con Piano Libre

Tabla 6.5 Caso de prueba Interactuar con Piano Libre.

Visión general	
Identificador	P2.0
Creador de caso de prueba	Luis Riquelme
Versión	1.0
Nombre	Caso de prueba Interactuar con Piano Libre
Propósito	Verificar que el piano virtual concuerde con la interacción del usuario frente a la Kinect. Tanto en los sonidos producidos como en el funcionamiento de la detección de dedos en sí.
Dependencias	La Kinect deberá estar iniciada y funcionando correctamente en el sistema.
Actividades	
Ambiente de prueba/ configuración	Ambiente de desarrollo de Visual Studio.
Inicialización	<ul style="list-style-type: none"> • Ingresar a la aplicación. • Ir a la función de Piano Libre.
Finalización	<ul style="list-style-type: none"> • Si al posicionar la mano frente a la Kinect no se muestra un identificador sobre cada dedo, se debe comprobar que el uso de la librería está siendo bien utilizada, además de verificar que la Kinect está recibiendo información. • Si al posicionar un dedo ya detectado sobre una tecla y ésta no emite un sonido, se debe verificar que la programación de la interacción de los dedos con las teclas esté realizada de manera idónea. Para esto se debe considerar el buen uso de la librería y la gestión correcta de las teclas frente a la pantalla.
Acciones	<ul style="list-style-type: none"> • Poner mano frente a dispositivo Kinect para que éste la detecte. • Una vez detectada la mano e identificados los dedos, ponerlas frente al piano virtual y acercar un dedo a ella.
Datos de entrada	En esta funcionalidad no existen datos de entrada, más que la información que proporciona la Kinect al sistema.
Resultados	
Salida esperada	Sonido acorde a las teclas cuando los dedos se posan sobre ellas.

Salida obtenida	Sonido acorde a las teclas cuando los dedos se posan sobre ellas.
Resultado	Correcto.
Severidad	Grave.
Seguimiento	No se encontraron defectos en este caso de prueba.
Estado	Terminado.

6.2.3 Caso de prueba Interacción con lecciones del curso

Tabla 6.6 Caso de prueba Interacción con lecciones del curso.

Visión general	
Identificador	P3.0
Creador de caso de prueba	Luis Riquelme
Versión	1.0
Nombre	Caso de prueba Interacción con lecciones del curso
Propósito	Verificar el correcto funcionamiento de las lecciones. Sin tomar en cuenta la detección de dedos ni el funcionamiento del piano virtual. (Mismo módulo que piano libre)
Dependencias	La Kinect deberá estar iniciada y funcionando correctamente en el sistema.
Actividades	
Ambiente de prueba/ configuración	Ambiente de desarrollo de Visual Studio.
Inicialización	<ul style="list-style-type: none"> • Ingresar a la aplicación. • Ir a la función de Curso Piano. • Elegir una lección.
Finalización	<ul style="list-style-type: none"> • Si en alguna lección el puntaje no aumenta luego de ser completada, se debe verificar la base de datos y comprobar que no hubo aumento. Si lo hubo, se necesitará revisar el código del mismo. • Si en alguna lección, las notas musicales no se representan correctamente en el piano virtual, se deberá analizar el módulo del piano. • Si no existió interacción de las manos con el piano virtual en conjunto a una lección, se debe revisar el uso y la integración de las lecciones al módulo del piano.
Acciones	<ul style="list-style-type: none"> • Poner mano frente a dispositivo Kinect para que éste la detecte.

	<ul style="list-style-type: none"> • Una vez detectada la mano e identificados los dedos, seguir las indicaciones de la lección.
Datos de entrada	En esta funcionalidad no existen datos de entrada, más que la información que proporciona la Kinect al sistema.
Resultados	
Salida esperada	<ul style="list-style-type: none"> - Sonido acorde a la presión de los dedos sobre las teclas. - Puntaje obtenido e información acorde a la lección realizada.
Salida obtenida	<ul style="list-style-type: none"> - En cada lección no se presentó problemas con el piano virtual (La lección2, 3 y 4 requerían interacción con piano virtual). - Puntaje obtenido e información acorde a cada lección realizada.
Resultado	Correcto.
Severidad	Grave.
Seguimiento	No se encontraron defectos en este caso de prueba.
Estado	Terminado.

7 Conclusiones y trabajo futuro

El proceso de análisis y diseño de un sistema es fundamental si de manera posterior se desea obtener éxito en el proceso de construcción del mismo. Es importante cambiar el paradigma, el cual se refiere a que es imprescindible la destinación de mayores recursos, ya sea capital o tiempo, a esta última actividad en desmedro de las dos primeras. Se hace imperioso, entonces, una modificación a lo que se consideraba como legítimo y añadir lo siguiente: un mal proceso de análisis y diseño llevará de forma inevitable al fracaso del desarrollo del sistema.

La realización del presente documento tiene como finalidad, mostrar todas las etapas desde el inicio hasta el final del desarrollo del sistema. Para realizar estas etapas se trabajó con una metodología como la de Prototipado, que permite al equipo de trabajo realizar un análisis rápido de los requerimientos que se buscan satisfacer, los cuales se detallaron en los diagramas UML expuestos a lo largo del documento. Además, gracias a la utilización de esta metodología se pudo realizar una división de actividades, donde el trabajo en paralelo contribuyó de gran manera a cumplir con los plazos establecidos, mejorando la eficacia con la que las actividades fueron desarrolladas.

Todas estas actividades se desarrollaron con la idea de generar un producto final útil y real, que pueda -de alguna forma- ser usado fácilmente por una persona con ganas de aprender piano o simplemente, por la curiosidad de interactuar con una aplicación novedosa mediante un dispositivo Kinect. Las actividades planificadas fueron realizadas, por lo que se generó un sistema que cumple con las expectativas propuestas, sin embargo se dejará como trabajo futuro el desarrollo de canciones más complejas para reforzar el aprendizaje del usuario. Lo descrito de forma anterior, se establece por las limitaciones de hardware que existen hoy en día. Dado que la detección de dedos y la interacción no es lo suficiente fluida para garantizar que el usuario no se frustre practicando lecciones de gran complejidad como lo son las canciones. En cuanto a las limitaciones de hardware, es importante recordar que el dispositivo actual, en teoría, no es lo suficientemente potente para realizar las tareas que en este proyecto de software se necesitan.

A modo de cierre, del presente documento, el equipo de trabajo declara haber realizado un software de calidad para los usuarios. Lo anterior, de forma principal por el cumplimiento de los requerimientos que fueron planteados desde comienzo del año académico, los cuales fueron determinados en una etapa temprana donde los integrantes del equipo no tenían conocimiento de las herramientas que eran necesarias para cumplir con los objetivos del proyecto. Es por lo anterior, que el equipo de trabajo pasó por dificultades a la hora de implementar las ideas en las que se basa el proyecto, como lo es la detección de dedos. Sin embargo, la perseverancia y el trabajo en equipo permitió que el sistema se ajuste a los estándares que se le exige a los egresados de la PUCV.

Junto con lo anterior, el equipo de trabajo declara haber obtenido un aprendizaje significativo en el manejo de nuevas tecnologías como la Kinect y las herramientas necesarias para desarrollar software para dichas tecnologías; ya que estos conocimientos eran parte de la motivación inicial del proyecto, junto con la innovación en el ámbito del aprendizaje de música como se describió a lo largo del documento. Si bien es cierto, que una parte del equipo tenía conocimientos de piano, la otra parte no contaba con ningún conocimiento en el área, por lo que el esfuerzo en algunas partes del desarrollo fue extenuante, y conllevó a dicha parte a adquirir conocimientos a través de la metodología que se utilizó.

Además de los conocimientos descritos en el párrafo anterior, el equipo tuvo que asistir a reuniones con especialistas del área, para asegurarse que el sistema cumpla con las características necesarias para generar conocimiento básico en los usuarios, y apegado a las metodologías que utilizan los profesores de música para enseñar un instrumento musical como el piano. También, en dichas reuniones; sobre todo en la inicial, el equipo de trabajo obtuvo una gran cantidad de mejoras que tenían que ser implementadas, como por ejemplo: que el piano virtual se ajuste a un piano real y lo importante que es el “do central” en el aprendizaje de este instrumento. Además de obtener una metodología de enseñanza que fuera didáctica y de fácil comprensión; es por eso que se utilizó el libro: “Piano Básico de Bastien Nivel Elemental” ya que dicho libro, es una metodología para el aprendizaje de niños, cumpliendo lo descrito anteriormente.

Finalmente, el equipo de trabajo concluye que; pese a tener una gran dificultad en la detección de dedos, por lo ya mencionado a lo largo del documento, fue factible crear un sistema con las características necesarias para generar aprendizaje en los usuarios. Además, el aprendizaje obtenido por equipo en los lenguajes de C# y XAML permitirá que los integrantes de este proyecto cuenten con otras herramientas para competir en el mundo laboral. Finalmente, la experiencia de trabajar con un dispositivo reciente como la Kinect permitió que el equipo de trabajo desarrollara habilidades blandas como la tolerancia al fracaso al momento de implementar posibles soluciones, la perseverancia cuando las ideas a desarrollar no funcionaban como lo era esperado, además de lo importante que es el trabajo en equipo, dado que el trabajo en equipo permitió que cuando faltaban energías para implementar partes del sistema, el equipo fortalecía las esperanzas de entregar un software de calidad en los plazos establecidos y cumpliendo con las exigencias de la universidad.

8 Referencias

- [1] Alan Belkin, compositor, *General Principles of Piano Technique*, <https://www.webdepot.umontreal.ca/Usagers/belkina/MonDepotPublic/Piano/PianoTechnique.html>.
- [2] Jarrett Webb and James Ashley, *Beginning Kinect Programming with the Microsoft Kinect SDK*, Apress, pp. 1-25. 2012.
- [3] Marcos Zuñiga Barraza, *Documentación Kinect “Primeros Pasos”*, Universidad Técnica Federico Santa María, pp. 1-25.
- [4] Sitio Web Microsoft, *DreamSpark*, <http://www.dreamspark.com>. Revisado el 10 de Abril, 2013.
- [5] Sitio Web PcFactory, *precios referenciales*, <https://www.pcfactory.cl/producto/7560-Windows.7.Starter.OEM.32.bit>. Revisado el 10 de Abril, 2013
- [6] Roger S. Pressman, *Ingeniería del Software. Un enfoque práctico.*, Mc Graw Hill. Quinta Edición, Madrid – 2002.
- [7] Rob Miles, *Start Here - Learn the Kinect API*, MS Press, pp. 1-11. 2012.
- [8] Roger Pressman - Ingeniería de Software – Sexta Edición, pp. 55-57.
- [9] James Bastien – Piano Básico de Bastien Nivel Elemental, pp 4-9.
- [10] Sitio Web ConservatorioPucv, <http://conservatoriopucv.cl/docentes>. Revisado el 26 de Diciembre, 2013.

1 Anexos

A: Detalle de Actividades

En la Fase de Recolección y Refinamiento de Requisitos, se procede a recolectar la información necesaria para comenzar a dar solución a la problemática antes descrita.

- Reunión con profesor guía (HITO): Se define la estructura del informe de avance que debe presentar el equipo de trabajo.
- Estudio de tecnología Kinect: Se procede a instalar programas básicos para hacer funcionar el dispositivo. Además se ejecutan algunos ejemplos. Duración 1 día.
- Definición del problema: Se define el problema y la solución que será implementada en el sistema. Duración: 1 día.
- Planificación: Se define la planificación del equipo de trabajo, es decir, se realiza la carta Gantt y se detallan las actividades que ésta posee. Duración: 3 días.
- Definición de objetivos: Se define el alcance del proyecto y los objetivos que éste busca cumplir. Duración: 1 día.
- Definición de Metodología: Se fija de forma definitiva la metodología con la que se ha de trabajar, basándose en lo ya se ha hecho hasta el momento. Duración: 3 días.
- Reunión con Profesor guía (HITO): Se realiza reunión con profesor guía para revisar el estado de avance del proyecto. Además, de realizar consultas y resolución de dudas. Duración: 1 día.
- Corrección de objetivos: Los integrantes del equipo de trabajo corrigen los errores en los objetivos encontrados por el profesor.
- Análisis de Requerimientos: Se analizan los requisitos funcionales que debe cumplir el sistema, además se definen los requisitos no funcionales, que permitirán establecer los aspectos que no dependen de la implementación del sistema. Duración: 1 día.

En la fase de Diseño Rápido se procede a realizar los primeros diagramas para satisfacer las necesidades del sistema. Se tratan de hacer lo más cercano a la realidad para construir un prototipo lo más funcional posible.

- Construcción de casos de uso: Se elaboran los casos de uso, definiendo así los actores y las funcionalidades del sistema desarrollado por el equipo de trabajo. Duración: 1 día.
- Definición de arquitectura: Se define la arquitectura lógica que va a utilizar el sistema durante su implementación. Duración: 3 días.
- Realización de Marco Teórico: Se comienzan a explicar las tecnologías con las que se va a implementar el proyecto. Además de explicar conceptos importantes para la implementación. Duración: 10 días.
- Reunión con profesor guía (HITO): Se muestran los avances hasta el momento. Duración 1 día.
- Entrega de informe de avance (HITO): Se entrega de informe de avance el proyecto, el cual contiene: Problema solución, Casos de uso, arquitectura lógica, especificación de requerimientos, marco teórico. Duración: 1 día.

En la fase de Prototipado se procede a construir el primer prototipo de la aplicación. Así el usuario final podrá validar los requerimientos del sistema en construcción. También se podrá realizar nuevos diagramas que ayudarán a comprender de mejor manera la construcción del sistema.

- Modelo de datos: Se confecciona el modelo de entidad-relación (MER), vista fundamental del sistema para la implementación de los datos con los que trabajará la solución. Duración: 2 días.
- Análisis de Riesgos: Se realiza una evaluación de riesgos que podría afectar la realización del proyecto. Duración 4 días.
- Diseño de interfaces: Se diseñan las interfaces del *software*, con el objetivo de orientar al programador con respecto al servicio que se le entregará al usuario. Duración: 2 días.
- Cruce de interfaces con diagramas: Se realiza una comparación entre las interfaces y los diagramas realizados hasta el momento. De esta manera se garantiza que se cumplan los requerimientos solicitados. Duración 1 día.
- Reunión con el Profesor guía (HITO): Se realiza reunión con el profesor guía para revisar lo realizado hasta el momento. Duración 1 día.

- Implementar base de datos: Se integran en el sistema los aspectos relacionados con la capa de acceso a los datos correspondientes a ambas plataformas que conforman la solución desarrollada. Duración: 1 día.
- Implementar capa de presentación: Se integra al sistema las interfaces entre usuario y sistema, las cuales permitirán cumplir con las funcionalidades capturadas durante el proceso de análisis y diseño de la solución; es decir, por medio de los modelos de casos de uso y el diseño de interfaces, se llevan a la práctica esta capa de la arquitectura. Duración: 8 días.
- Implementar lógica de negocio: Se integran al sistema los métodos necesarios para concretar el real funcionamiento de la solución; es decir, se establece la conexión entre lo que el usuario requiere y la información almacenada en la capa de datos. Duración: 15 días.
- Correcciones de requerimientos: Al ser un prototipo se muestra al usuario final, de esta manera se busca que el usuario apruebe los requerimientos realizados hasta el momento ó se agregan correcciones al prototipo. La idea es acercarse de mejor manera a las necesidades del usuario. Duración 1 día.
- Entrega de informe final (HITO): Se entrega el informe final de la asignatura, el cual contiene todas las actividades realizadas hasta el momento. Duración 1 día.
- Entrega de prototipo (HITO): Se entrega un prototipo funcional al usuario. La idea es al menos, entregar un prototipo con la detección de una mano. Duración 1 día.

En la Fase de Evaluación del prototipo, se procede a evaluar las características del prototipo presentado. Verificando que los requerimientos se vean plasmados en el *software*.

- Reunión con profesor guía (HITO): Se define los elementos que deben ser agregados al informe. Se discute la estructura de trabajo que debe seguirse en Proyecto 2.
- Documentación de *Abstract* y Resumen: Se realiza el resumen del proyecto. Realizando lo anterior en español e inglés. Duración 1 día.
- Realización de plan de prueba: Se establecen los casos que serán probados en el sistema. Lo anterior, siguiendo el uso de “caja negra” y de “caja blanca” Duración: 2 días.
- Estudio de nuevas librerías: Con el fin de establecer una mejor interacción con el usuario, se realizan estudios a otras librerías que utilizan dos Kinect para la detección de dedos. Duración: 8 días.

- Reunión con Jefa de Departamento de Piano del Instituto de Música (HITO): Se establece una reunión con un especialista en la enseñanza de Piano. Lo anterior, con el fin de conseguir una metodología a aplicar en el software.

En la Fase de Refinamiento del prototipo, se procede a elaborar mejoras al prototipo realizado. Lo anterior, con el fin de mejorar y ajustar dicho prototipo a la versión final y llegar a plasmar los requerimientos de mejor manera.

- Pruebas de Librerías con dos *Kinect*: Se realizan pruebas con las librerías que se estudiaron anteriormente. Duración: 9 días.
- Estudios de metodología de aprendizaje de piano: Este estudio se establece para definir la dificultad a que se enfrentará el equipo de trabajo. De esta manera, poder establecer la cantidad de lecciones que podrían ser implementadas. Duración: 3 días.
- Reunión con Profesor guía (HITO): Se realiza reunión con profesor guía para revisar el estado de avance del proyecto. Además, de realizar consultas y resolución de dudas.
- Diseño e implementación de interfaces de “Realizar Curso”: Se realiza diseño de interfaces utilizando la tecnología *Kinect*. Además, se busca que las interfaces cumplan de buena manera los requerimientos del sistema. Duración 7 días.
- Entrega de informe de avance (HITO): Se entrega la versión de avance del informe de proyecto. Además de un CD con la versión actualizada del software.
- Implementación de Lecciones: Luego de las reuniones con los profesores del IMUS (Instituto de Música) y luego de realizar los estudios de la metodología se decide cuantas lecciones serán implementadas y luego se realiza la implementación. Duración: 23 días.
- Pruebas modulares: Se realizan pruebas de los módulos del sistema, con el fin de corregir los problemas de dichos módulos. Lo anterior, con el fin de disminuir los fallos con del sistema. Duración 2 días
- Pruebas de Integración: Con las pruebas modulares ya realizadas, se procede a integrar los módulos probados. Lo anterior, con el fin de entregar el sistema completo en su contexto. Duración 1 día.

En la Fase de Producto de Ingeniería, se procede a realizar las últimas correcciones de errores que pueda tener el *software*. Gracias a dichas correcciones el sistema es validado

como un producto que cumple con los requerimientos y plazos establecidos al comienzo de este.

- Corrección de errores: Esta corrección de errores, se realiza en caso de que las pruebas de integración o modulares resulten exitosas, es decir, que al realizar la prueba se encuentren errores. Duración 3 días.
- Entrega de *Software* final (HITO): Se procede a la entrega final del software. En esta entrega se busca que el anterior, no cuente con ningún error, es decir, un *software* cien por ciento funcional y operacional.
- Revisión de *Software* (HITO): Se realiza una última revisión del sistema. En este caso, se realiza con el profesor guía y el correferente. En esta revisión se busca encontrar posibles fallos que el equipo de trabajo no haya detectado.
- Corrección de errores de *software*: Esta tarea se realiza siempre y cuando en la revisión con el profesor guía y el correferente se encuentren fallos y deban ser solucionados. Duración 10 días.
- Entrega de informe final (HITO): Se entrega el informe final de la asignatura, el cual contiene todas las actividades realizadas hasta ese momento.

B: Estudio de Factibilidad

B:1 Factibilidad Económica

La factibilidad económica corresponde a la comparación entre las alternativas descritas con anterioridad, para determinar su viabilidad en cuanto al uso de recursos financieros. La siguiente tabla resume la información con respecto a los gastos en inversión inicial que comprende cada opción.

Tabla B.1 Factibilidad económica.

Herramientas	Detalle	Alternativa Software Libre	Detalle	Alternativa Software Pagado
Motor Base Datos	PostgreSQL	0	Microsoft SQL Server Compact Edition	0
Herramienta Desarrollo	Eclipse	0	Microsoft Visual Studio	0 [4]
Sistema Operativo	Linux	0	Microsoft Windows	34.990 [5]
Totales		0		34.990

Los valores correspondientes a *software* están con su correspondiente referencia al final del documento.

Como se puede apreciar en la Tabla 6, la opción de software libre no prevé de ningún costo para el equipo de trabajo, no así la alternativa de software pagado el cual remite un costo en adquisición del sistema operativo, con un costo de \$34.990. Se estableció, mediante conversaciones y suposiciones, una Tasa de Retorno Mínima Aceptable (TRMA) del 15% y un promedio de ingresos por aplicación de \$18.000 e ingresos anuales de \$270.000, con estos datos se analizará el Valor Actualizado Neto (VAN) asociado a las alternativas, para poder observar cuál es el Período de Recuperación del Capital (PRK) de cada una.

- **Alternativa de Software Libre**

$$0 = -I_0 + \text{Ingresos Anuales} * (P/A, 15\%, T)$$

$$0 = -0 + 270000 * (P/A, 15\%, T)$$

$$0/270000 = (P/A, 15\%, T)$$

$$0 = (P/A, 15\%, T)$$

$$PRK = 0$$

Se podrá recuperar la inversión inicial en 0 años

- **Alternativa de Software Pagado**

$$0 = -I_0 + \text{Ingresos Anuales} * (P/A, 15\%, T)$$

$$0 = -34990 + 270000 * (P/A, 15\%, T)$$

$$34990/270000 = (P/A, 15\%, T)$$

$$0.12959 = (P/A, 15\%, T)$$

$$PRK = 0.149$$

Se podrá recuperar la inversión inicial en 0.149 años.

B:2 Factibilidad Técnica

Para evaluar este aspecto se realizará un análisis a cada una de las alternativas para describir la factibilidad técnica de su implementación; de este modo se podrá ver de manera general la disponibilidad con la que el equipo de trabajo cuenta para la realización del proyecto, y así poder estar en conocimiento de los riesgos que cada una de las alternativas puede traer.

- **Alternativa de Software Libre**

Requerimientos de *Software*

- Sistema Operativo: Linux, cualquier distribución de éste. Por ejemplo, Fedora, Ubuntu, Mandriva, etc.
- IDE: Eclipse, con la incorporación de JNECT. SDK de Kinect realizado en Java.
- SGBD: PostgreSQL, debido a que el equipo de trabajo cuenta con experiencia en dicho gestor.

Requerimientos de *Hardware*

- Procesador: 64-bit (x64) Dual-core GHZ o procesador superior.
- USB 2.0 bus dedicado.
- 2 GB de Ram.
- Microsoft Kinect.

- **Alternativa de Software Pagado**

Requerimientos de *Software*

- Sistema Operativo: Microsoft Windows 7, Started Edition.
- IDE: Microsoft Visual Studio.
- SGBD: Microsoft SQL Server Compact Edition, pese a que el equipo de desarrollo no cuenta con experiencia en su utilización.

Requerimientos de *Hardware*

Al igual que en la alternativa de software libre se necesitan los mismos requerimientos de *hardware* para llevar a cabo la realización del proyecto.

B:3 Factibilidad Operativa

En este apartado se considera la relación de las alternativas con el recurso humano, es decir, se evalúa si el equipo de trabajo está capacitado para operar con cada opción planteada.

- **Alternativa de Software Libre**

El equipo no presenta inconvenientes con utilizar Eclipse como IDE para desarrollar la aplicación, pues poseen experiencias anteriores con éste. Para utilizar el motor de base de datos PostgreSQL, se prevé una situación similar, aunque con algunas diferencias en cuanto a la experiencia con procedimientos almacenados. La utilización de Linux como Sistema Operativo no presenta mayores problemas; no obstante, se requeriría de la adquisición de un curso para profundizar conocimientos con respecto al desarrollo de aplicaciones en Kinect sobre esta plataforma.

- **Alternativa de Software Pagado**

Esta alternativa posee la ventaja de permitir al personal el trabajo dentro del entorno de Microsoft Windows, sistema operativo más cercano a las actividades diarias de éstos. Por lo tanto, se especifica que al trabajar sobre Windows, no se necesitará mayor conocimiento para el manejo sobre esta plataforma, no así como lo sería en la plataforma Linux. Por otra parte, la base de datos será Microsoft SQL Server Compact Edition. Y por último, el IDE sobre el cual se desarrollará la aplicación para Kinect será Microsoft Visual Studio, versión gratuita para estudiantes; éste será un desafío para el equipo de trabajo, pues serán los primeros acercamientos con él, por ende, habrá una gran curva de aprendizaje.

B:4 Factibilidad Legal

En este punto se tratará las dos alternativas en conjunto, permitiendo que se vaya haciendo una comparación entre éstas. Debido a que el proyecto que desarrollará el equipo de trabajo es un producto que será entregado a la comunidad, no cuenta con un cliente fijo al cual se le deba cumplir con plazos y entregas, pero de igual manera los integrantes del grupo de trabajo han definido sus tiempos y los momentos en los cuales harán entregas del proyecto, de acuerdo a la planificación de la asignatura de Proyecto 1 y la asignatura de Proyecto 2, para así tener en consideración las fallas que encuentren los usuarios y poder hacer actualizaciones al programa. Se considera que la aplicación utilizará datos otorgados por los usuarios al momento de la utilización de ésta, con el fin de conocer el progreso del usuario en la metodología de aprendizaje de piano otorgada por el software.

En cuanto al software y herramientas a utilizar deberá cancelar el uso de la alternativa de software libre. Debido a que estos no cuentan el respaldo de Microsoft, que es la empresa encargada de la fabricación del dispositivo Kinect. De esta manera, se puede contar con toda la documentación realizada por el fabricante.

B:5 Conclusión de Factibilidad

Después de haber sido analizada cada una de las posibilidades con detenimiento, evaluando los aspectos a favor y en contra de cada una de ellas, el equipo de trabajo señala que la opción que se debe implementar es la alternativa de software pagado. Esta afirmación se realiza debido a que se cuenta con la documentación necesaria para la realización del proyecto, además del respaldo de una empresa grande como lo es Microsoft. Pese a que esta alternativa presenta un costo adicional al equipo de trabajo ya que, hay que invertir en la adquisición del S.O.

También, la investigación realizada por los integrantes del grupo de trabajo, demuestra que la gran mayoría de proyectos realizados con esta tecnología se encuentran desarrollados con la tecnología de la alternativa de pago, ya que el dispositivo Kinect muestra un comportamiento favorable con la tecnología de Microsoft antes mencionada. Además, se encuentran librerías que podrían ser utilizadas por el proyecto.

C: Diseño de Interfaces

- Menú Principal: Es aquí donde el usuario encuentra las primeras opciones para interactuar con el sistema. Entre ellas, la lista de usuario, Piano Libre, Realizar Curso, la posibilidad de eliminar su usuario, entre otras que serán detalladas a continuación.



Figura B.1 Interfaz Menú Principal.

- Usuarios: El usuario podrá elegir, entre la lista de usuarios que se encuentran ingresados en el sistema su perfil u optar la posibilidad de crear un nuevo perfil en el sistema.



Figura B.2 Interfaz Usuarios.

- **Agregar Usuario:** Al presionar el botón “más” que se puede apreciar en la interfaz anterior, se despliega un dialogo en el que se solicita un nombre de usuario. Es importante destacar, que el equipo de desarrollo estimo que el uso de teclado para el ingreso de valores resultaba más cómodo para el usuario que realizar esta interacción con el dispositivo Kinect. Cuando se agrega un usuario, el sistema despliega un dialogo indicando que el usuario fue agregado con éxito. En caso contrario, se despliega otro dialogo indicando que el nombre de usuario no es válido.

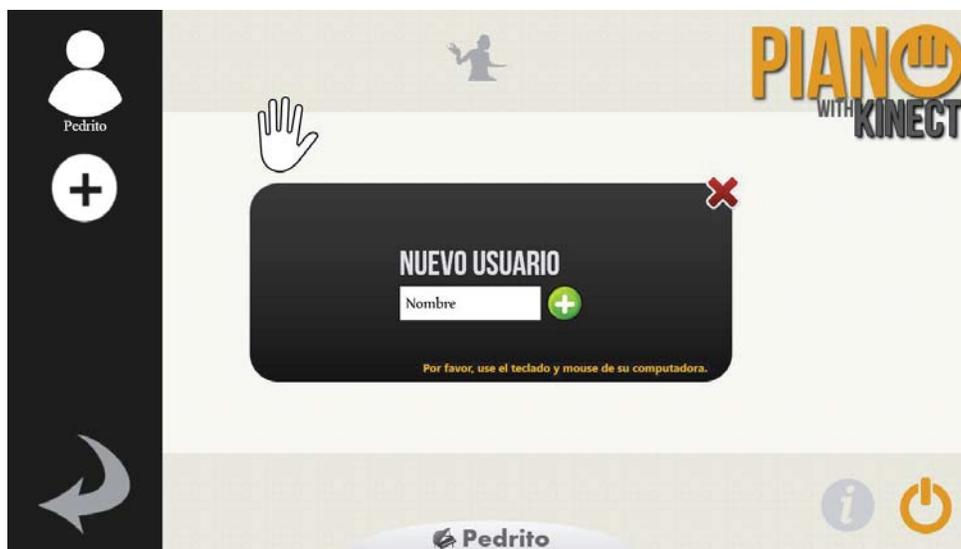


Figura B.3 Interfaz Agregar Usuario.

- **Selección de Usuario:** Cuando se selecciona un usuario de la lista, se despliega un cuadro seleccionable en la parte baja en la que se puede apreciar el nombre del usuario que fue seleccionado y el puntaje total que lleva en las lecciones que ha realizado.



Figura B.4 Interfaz Selección de Usuario.

- Eliminar Usuario: En el cuadro central nombrado de forma anterior, aparece la opción de eliminar el usuario que ha sido seleccionado. Aquí se despliega un dialogo en el que se pide confirmación para realizar esta acción.



Figura B.5 Interfaz Eliminar Usuario.

- Piano Libre: El usuario puede interactuar con el piano tocando las teclas sin seguir un orden preestablecido.

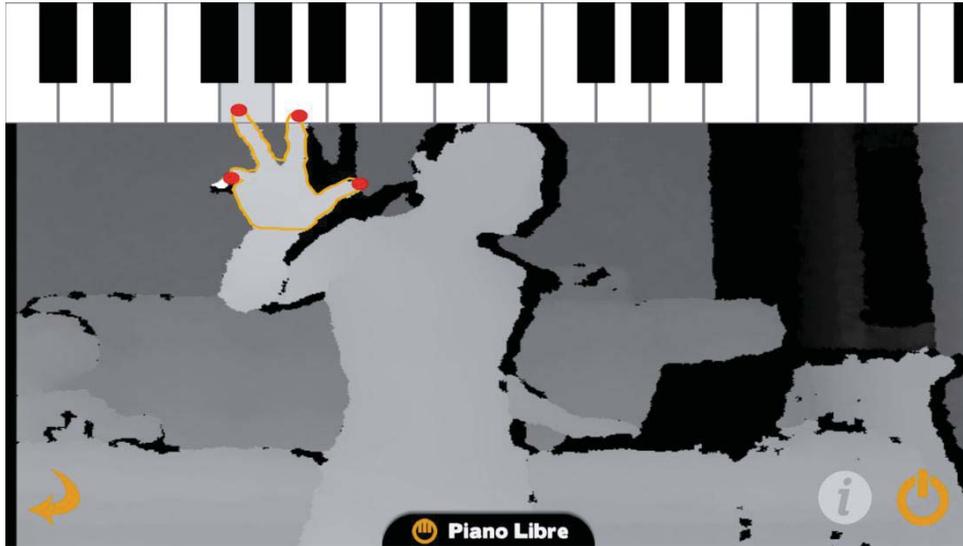


Figura B.6 Interfaz Piano Libre.

- Curso Piano: El usuario puede elegir qué lección quiere realizar. Además, se aprecia los puntajes de las lecciones logradas junto a un tick verde identificador, y en las que no se han realizado una X correspondiente.



Figura B.7 Interfaz Curso Piano.

- Lección 1: Una de las pantallas de la lección 1, que corresponde a ejercicios de posición de manos y numeración de dedos. En esta como en cada lección se puede apreciar las manos del usuario con un contorno naranja y puntos rojos en la cima de los dedos.



Figura B.8 Interfaz Curso Piano – Lección 1.

- Lección 2: Esta lección se divide en 2 pasos, la primera es sobre digitación y tonos sobre el piano, la segunda para practicar un ejercicio de ritmo.



Figura B.9 Interfaz Curso Piano – Lección 2 Paso 1.

Figura B.10 Interfaz Curso Piano – Lección 2 Paso 2.

- Lección 3: En esta lección se muestra al usuario las notas musicales sobre el piano e información básica acerca de este tema, además, como segundo objetivo, se pide realizar una pequeña canción de ejercicio.

Figura B.11 Interfaz Curso Piano – Lección 3 Paso 1.

Curso
Lección 3

DO RE MI FA SOL LA SI DO RE MI FA SOL LA SI DO RE

SOL - LA - SI - LA - SOL - LA - SI -

NOTAS MUSICALES

Canción práctica.
Los Globos

Sol La Si La Sol La Si Si La Sol La Si La Sol

2 3 4 3 2 3 4 4 3 2 3 4 3 2

Con sus ma-nos lan-za-rá ro-jos glo-bos a vo-lar! 2/2

¡TOQUE LA CANCIÓN CON LA MANO IZQUIERDA! (SI SE EQUIVOCA, COMIENCE NUEVAMENTE)

Pedrito

Figura B.12 Interfaz Curso Piano – Lección 3 Paso 2.

- Lección 4: La última lección consta de una evaluación, son 4 preguntas en relación a todo lo aprendido en las lecciones anteriores. En la siguiente imagen se puede apreciar la primera pregunta.

Curso
Lección 4

EVALUACIÓN

Revisión de los conocimientos. Diagnóstico de avance

1. ¿Cuál es la numeración correcta de las manos?

4 3 2 2 3 4 2 3 4 7 8 9 2 3 4 4 3 2 9 8 7 4 3 2
5 1 1 5 1 5 6 7 8 9 5 5 1 10 6 5 4 3 2

Pedrito

Figura B.13 Interfaz Curso Piano – Lección 4 Paso 1.

- Rehacer lección: El usuario, al ingresar nuevamente a una lección que ya realizó, debe rehacer la lección, por ende perderá su puntaje obtenido. Se muestra un dialogo de información con posibilidad de cancelarlo.



Figura B.14 Interfaz Curso Piano – Rehacer Lección.

- Interfaces de Ayuda: El usuario puede acceder a pantallas de ayuda desde distintas partes de la aplicación. Tiene ayuda en la pantalla principal, ayuda en piano libre, y por último información de ayuda en el curso. Las 3 pantallas se muestran a continuación en este mismo orden.

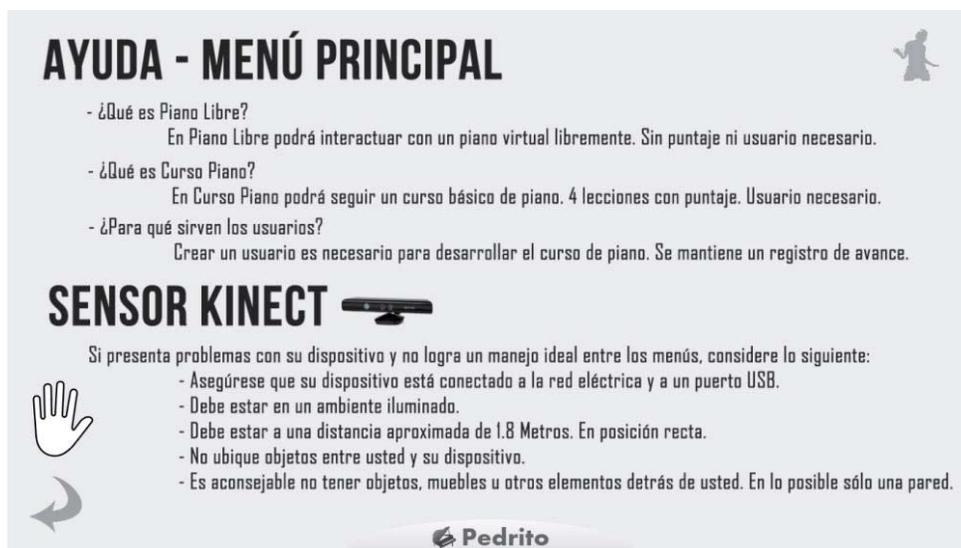


Figura B.15 Interfaz Menú Principal – Ayuda.

AYUDA - PIANO LIBRE

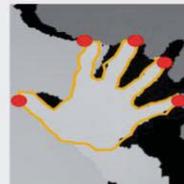


Para tener un funcionamiento idóneo del piano y la interacción con sus manos, considere los siguientes aspectos:

- El sensor Kinect debe estar enfocado de frente hacia la palma de su manos.
- Desde donde estaba, a 1.8 Mts. aprox., acérquese cautelosamente hasta que aparezca un contorno naranja de sus manos.
- Al aparecer el contorno de sus manos, identifique los puntos rojos en cada dedo. Busque la distancia ideal.
- El movimiento entre las teclas debe ser fino; lentamente y manteniendo la distancia. Sin movimientos bruscos.
- Como recomendación, reconozca el centro del teclado y ubique una mano a cada lado desde el centro.
- Interactúe con el piano libremente.



DISTANCIA IDEAL



MANO DETECTADA CORRECTAMENTE



Figura B.16 Interfaz Piano Libre – Ayuda.

AYUDA - CURSO PIANO



- En este apartado podrá seguir el curso de piano mediante 4 lecciones definidas.
- Las lecciones otorgarán un conocimiento básico del instrumento de piano.
- Puede acceder a cualquier lección, sin embargo es recomendable seguir el orden pre-determinado.
- Por cada lección obtendrá un puntaje, con éste verá reflejado su rendimiento.
- El máximo puntaje a aspirar es 100 pts.
- Para ingresar a una lección sólo presione en el cuadro correspondiente.



SENSOR KINECT



Si presenta problemas con su dispositivo y no logra un manejo ideal entre los menús, considere lo siguiente:

- Asegúrese que su dispositivo está conectado a la red eléctrica y a un puerto USB.
- Debe estar en un ambiente iluminado.
- Debe estar a una distancia aproximada de 1.8 Metros. En posición recta.
- No ubique objetos entre usted y su dispositivo.
- Es aconsejable no tener objetos, muebles u otros elementos detrás de usted. En lo posible sólo una pared.



Pedrito

Figura B.17 Interfaz Curso Piano – Ayuda.