



**Departamento de Ciencias de la Computación
y Tecnologías de la Información**



**Prototipo de Sistema de Educación Interactiva
para profesores de Lenguaje de Enseñanza Básica
utilizando Kinect® de Microsoft©**

**Memoria para optar al título de:
Ingeniero Civil en Informática**

Diciembre, 2016
Chillán, Chile

Autor
Cristopher Alejandro Alvear Candia
Profesor Guía
María Antonieta Soto Chico

Resumen.

Este proyecto, se presenta para dar conformidad a los requisitos exigidos por la **Universidad de Bío-Bío** en el proceso de titulación de la carrera de **Ingeniería Civil en Informática**, perteneciente a la Facultad de Ciencias Empresariales, en el campus Fernando May, ciudad de Chillán.

El proyecto titulado *“Prototipo de sistema de Educación Interactiva para profesores de Lenguaje de Enseñanza Básica utilizando Kinect de Microsoft”*, busca desarrollar un sistema prototipo, utilizando el sensor Kinect para los sistemas Microsoft Windows, que demuestre las capacidades de este para el desarrollo de aplicaciones que integren actividades didácticas en el aula de clases de la asignatura de Lenguaje.

El proyecto surge como una alternativa para mejorar el sistema educacional en Chile, enfocándose en las clases de Lenguaje de Enseñanza Básica, toda vez que se señala que la escasa innovación y adopción de nuevas tecnologías en el proceso de enseñanza-aprendizaje son una causa del estancamiento en el progreso de la educación chilena.

El prototipo hace uso de los sensores de audio e imagen de Kinect. Los micrófonos permiten realizar reconocimiento de voz para validar lecturas del estudiante, mientras que las cámaras permiten realizar términos pareados utilizando gestos.

A continuación, se define el marco del proyecto de desarrollo de software, los detalles de la problemática hacia la que apunta, las características funcionales y técnicas del sistema y los detalles de la implementación del prototipo y desarrollo del proyecto.

Palabras clave: Tecnología Educativa, Tecnología de Información y Comunicación, Kinect, Microsoft, Windows, Educación Interactiva, Enseñanza Básica, Lenguaje, Prototipo Educativo.

Índice de contenidos.

1. Introducción.	9
1.1. Estructura del informe.	12
2. Problemática.	14
2.1. Problemas específicos.	14
3. Definición del proyecto.	16
3.1. Objetivos del proyecto.	16
3.1.1. Objetivo general.	16
3.1.2. Objetivos específicos.	16
3.2. Ambiente de Ingeniería de Software.	17
3.2.1. Modelo de desarrollo.	17
3.2.2. Descripción de las etapas de desarrollo.	18
3.2.3. Desarrollo basado en componentes.	18
3.2.4. Plataforma de desarrollo.	19
3.2.5. Herramientas relevantes de apoyo al desarrollo.	20
3.3. Relevancia del proyecto.	20
4. Análisis.	22
4.1. Perfil de usuario.	22
4.2. Especificación de requerimientos.	22
4.2.1. Visión global.	22
4.2.2. Alcances.	22
4.2.3. Objetivos del prototipo.	22
4.2.4. Descripción global del prototipo.	23
4.2.4.1. Uso y funcionalidades básicas.	23
4.2.5. Requerimientos técnicos.	24
4.2.6. Definición de requerimientos.	25
4.2.6.1. Requerimientos funcionales.	26
4.2.6.2. Requerimientos no funcionales.	29
4.2.7. Casos de Uso.	31
4.2.7.1. Actores.	31
4.2.7.2. Diagrama de Casos de Uso.	31

4.2.7.3.	Descripción de Casos de Uso.	32
5.	Diseño.	38
5.1.	Arquitectura.	38
5.1.1.	Arquitectura del entorno.	38
5.1.2.	Arquitectura del prototipo.	41
5.2.	Estándar de implementación de los módulos.	42
5.3.	Interfaz gráfica de usuario (GUI).	43
5.3.1.	GUI de lectura.	43
5.3.2.	GUI de términos pareados.	44
5.4.	Especificación de reportes.	46
5.4.1.	Lectura.	46
5.4.2.	Términos pareados.	47
5.5.	Especificación de datos.	47
5.5.1.	Datos de entrada.	47
5.5.2.	Datos de salida.	49
6.	Implementación.	50
6.1.	Estructura de los módulos.	50
6.2.	Capa de control.	50
6.2.1.	KinectController.	53
6.2.2.	TimedThreadHandler.	54
6.2.3.	FileUtils.	54
6.2.4.	KinectCapabilities.	54
6.2.5.	Librerías nativas de Kinect.	55
6.2.6.	NAudio.	56
6.2.7.	iTextSharp.	56
6.2.8.	Módulo de reconocimiento de voz.	56
6.2.8.1.	VoiceRecognizer.	57
6.2.8.2.	Reading.	58
6.2.9.	Módulo de reconocimiento de gestos.	61
6.3.8.1.	Pair.	61
6.3.8.2.	PairingTerms.	61
6.3.8.3.	MouseEmulator.	62

6.3.8.4.	SkeletonTracker.	63
6.3.8.5.	JointCache.....	64
6.3.8.6.	HandControl.	65
6.4.	Flujo de navegación.	68
6.5.	Interfaz gráfica de usuario.	69
6.5.1.	GUI de lectura.	71
6.5.2.	GUI de términos pareados.....	73
6.5.3.	GUI guardar reporte.....	75
6.6.	Características del prototipo.....	76
7.	Pruebas.....	77
7.1.	Especificación de las pruebas.	77
7.1.1.	Pruebas funcionales.	77
7.2.	Resultados de las pruebas.	78
7.2.1.	Pruebas funcionales.	78
7.2.2.	Índices de mantenibilidad.	78
7.2.3.	Prueba preliminar.	79
7.2.4.	Prueba de aceptación.....	80
7.3.	Análisis de los resultados.	83
8.	Síntesis y conclusiones.	85
9.	Trabajos Futuros.....	89
10.	Definiciones, siglas y abreviaciones del proyecto.....	91
11.	Referencias.....	92

Índice de tablas.

Especificaciones técnicas requeridas del computador para ejecutar Kleng.	24
Especificaciones técnicas requeridas del sensor Kinect.	25
Especificaciones técnicas de software requeridas para ejecutar Kleng.	25
Definición de secciones en la especificación de requerimientos.	26
Requerimientos funcionales.	29
Requerimientos no funcionales.	31
UCA01. Usuario.....	31
UCA02. Kinect.	31
UC01. Iniciar aplicación.....	33
UC02. Finalizar aplicación.....	33
UC03. Ver menú lectura.....	33
UC04. Ver menú términos pareados.....	34
UC05. Realizar lectura.....	35
UC06. Realizar términos pareados.....	36
UC07. Cargar lectura.....	37
UC08. Cargar términos pareados.....	37
Datos de entrada de Kleng.....	48
Datos salida de Kleng.....	49
Características Kleng Alpha 3.1.12.....	76
Definición de pruebas funcionales.....	78
Clasificación de complejidad ciclomática.....	79
Lista de estudiantes que probaron Kleng.....	81

Índice de figuras.

Resultados de Lectura SIMCE 2004-2014 según grupo socio-económico. (Gobierno de Chile, Ministerio de Educación, 2015).	9
Resultados Lectura, SIMCE 2014. (La Tercera, Paulina Salazar, 2015).	10
Kinect para Microsoft Windows. (www.engadget.com).	12
Modelo iterativo de desarrollo de software utilizado en el proyecto.	17
Diagrama de interacción básica entre el usuario y el prototipo.	24
Diagrama de Casos de Uso (UCD).	32
Arquitectura global (de entorno) de Kleng.	39
Arquitectura de Kleng.	41
Diseño de GUI de la Lectura.	43
Diseño de GUI de la visualización de la lectura.	44
Diseño de GUI de Términos Pareados.	45
Diseño de GUI de la visualización de los términos pareados.	46
Especificación reporte de Lectura.	47
Especificación reporte de Términos Pareados.	47
Ejemplo lectura predefinida en un archivo de texto.	48
Ejemplo de términos pareados predefinido en un archivo de texto.	49
Simbología de íconos clases UML (Visual Studio).	50
Diagrama UML de clases (sin detalle de clases) de la capa lógica.	51
Especificación de clase UML de KinectController.	53
Especificación de clase UML de TimedThreadHandler.	54
Especificación de clase UML de FileUtils.	54
Logo de librería NAudio.	56
Logo de librería iTextSharp.	56
Especificación de clase UML de VoiceRecognizer.	58
Especificación de clase UML de Reading.	59
Especificación de clase UML de Pair.	61
Especificación de clase UML de PairingTerms.	62
Especificación de clase UML de MouseEmulator.	63

Especificación de clase UML de SkeletonTracker.....	64
Especificación de clase UML de JointCache.....	65
Especificación de clase UML de HandControl.....	67
Logo de Kleng.....	68
Diagrama de flujo de navegación por Kleng.....	69
GUI menú de inicio Kleng.....	70
GUI ícono estado de Kinect.....	71
GUI menú de lectura Kleng.....	71
GUI ícono de idioma Kleng.....	72
GUI visualización de lectura Kleng.....	72
GUI menú de términos pareados Kleng.....	73
GUI visualización de términos pareados Kleng.....	74
GUI estados de la barra inferior de términos pareados.....	75
GUI guardar reporte Kleng.....	75
Resultados generales pruebas de sistema.....	78
Kleng Alpha 2.1.....	80
Estudiante realizando una lectura en Kleng.....	81
Estudiante realizando términos pareados en Kleng.....	82
Encuesta de aceptación Kleng.....	82

1. Introducción.

En Chile, el sistema educacional y las metodologías de enseñanza han estado sometidas a duras críticas durante la última década. Y es que este sistema, está muy alejado de proporcionar una cultura de innovación, lo cual se debe principalmente a la falta de motivación y compromiso [1], que fomenten la participación activa y voluntaria del estudiante en las diversas actividades que se desarrollan en el aula, estancando así el progreso en la educación, y su esencia, el aprendizaje.

Sin ir más allá, los resultados históricos de la SIMCE¹ [2], medidos entre los años 2004 y 2014, demuestran que en la última década, la mejora en los resultados se ha visto afectada en asignaturas específicas, como Lenguaje (ver Figura 1).

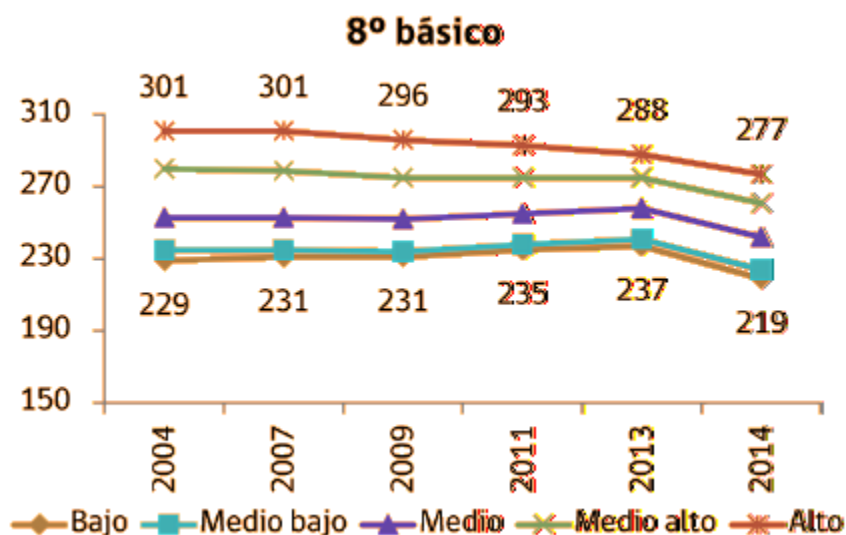


Figura 1. Resultados de Lectura SIMCE 2004-2014 según grupo socio-económico. (Gobierno de Chile, Ministerio de Educación, 2015).

Por segregación socioeconómica, se observa que la baja se produce en cada uno de los grupos, y es mayormente abrupta el 2014, donde una pequeña alza hasta el 2013 de los grupos socioeconómicos bajos y medio, cae.

Los resultados a nivel general de los 8° básicos en la SIMCE de lectura, registran una baja de 11 puntos, luego de un alza sostenida (ver Figura 2).

¹ Sistema de Medición de la Calidad de Educación.

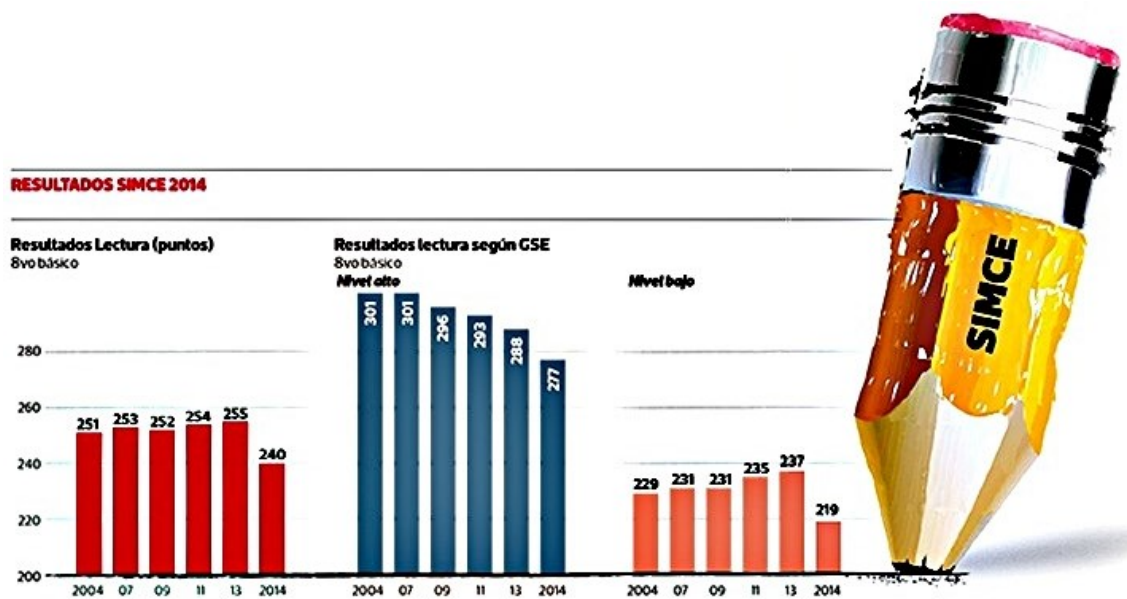


Figura 2. Resultados Lectura, SIMCE 2014. (La Tercera, Paulina Salazar, 2015).

¿Podrá la incorporación de las TIC² en el aula mejorar estos resultados?

Según [3], donde se analizan los factores rendimiento académico de los estudiantes y uso de las TIC en el aula, *“Existe una relación significativa entre logro escolar y frecuencia de uso de computadora en la escuela. Efectivamente, cuanto más usan la computadora en su escuela los estudiantes obtienen mejores puntuaciones [...]”*.

Aunque, en los últimos años el uso de computadores, como editor de texto, presentaciones y apoyo audiovisual (junto a proyectores y, el uso de las redes, en especial la Internet, para la comunicación académica se ha extendido rápidamente, su uso no va más allá. Según [4], un estudio realizado en Colombia, *“[...] las TIC se están usando en la educación como herramientas neutrales para hacer las mismas actividades de enseñanza y aprendizaje que se pueden hacer sin ellas.”* y, en muchas ocasiones los estudiantes entienden que los computadores sirven para investigar, dibujar, buscar programas o navegar a través de la Internet, sin más.

Chile no está lejos de esa realidad, ya que, en [3] se provee de los resultados de una encuesta realizada a 7025 estudiantes y 263 docentes en 165 establecimientos de Chile, que muestran relevantes estadísticas:

² Tecnologías de la Información y la Comunicación.

- El 100% de los establecimientos cuenta con computadores.
- La media de computadores por establecimiento es de 16.81.
- El 50.39% de los estudiantes posee un computador en su hogar.
- El 42.7% de los estudiantes usa su computador 1 o más veces a la semana.
- El 43.82% de los docentes usa un computador en su colegio.
- El 58.49% de los profesores sólo utiliza el computador para buscar información, comunicarse, o como herramienta de ofimática.

En [5], se muestra un análisis de la educación chilena en 20 años, donde un factor positivo es el alto mejoramiento que ha sufrido la infraestructura e equipamiento educacional, contrastado con el estancamiento de los resultados en desempeño académico.

Esto permite ver que, aunque la mayor parte de los establecimientos educacionales tienen un potencial tremendo en su estructura tecnológica, y la posesión de computadores es muy alta, más de la mitad de los docentes sólo utiliza los computadores para buscar y presentar información, lo que, en consecuencia, genera un recurso desaprovechado como medio para mejorar la calidad y dinamismo en la educación.

A partir de todo este panorama, es que nace la propuesta de la creación de un prototipo de sistema basado en TI, llamado Kleng, que impulse el desarrollo de aplicaciones que ofrezcan a los profesores de enseñanza básica, específicamente de la asignatura Lenguaje, actividades didácticas que mejoren el aprendizaje y la participación de sus estudiantes en el aula de clases, y más aún, establezca la base para sistema educativos utilizando esta tecnología, en futuros proyectos dentro de la Universidad del Bío-Bío.

El prototipo, se implementará a través del uso de Kinect de Microsoft³ (ver Figura 3), un dispositivo con diversos sensores que hacen posible, entre otras cosas, el procesamiento de imagen, sonido y profundidad de campo.



Figura 3. Kinect para Microsoft Windows. (www.engadget.com).

No es común en Chile el desarrollo de aplicaciones que utilicen Kinect, y mucho menos orientadas a la educación. Algunos proyectos relacionados, se pueden consultar en el Anexo A.

El objetivo del presente trabajo, es describir el desarrollo del prototipo de sistema informático propuesto, que intenta impulsar la innovación educativa y, subsanar en alguna medida, el estancamiento en las metodologías de enseñanza actuales, enfocándose en las clases de Lenguaje.

1.1. Estructura del informe.

El presente capítulo, es introductorio y es donde se presentan de manera general la problemática y el enfoque del proyecto.

El segundo capítulo, aborda el problema que motivó la propuesta del proyecto, las causas identificadas y sus consecuencias.

El tercer capítulo, establece los objetivos del proyecto y el ambiente de desarrollo junto a las técnicas y herramientas utilizadas.

El cuarto capítulo, muestra el análisis de la problemática y a los requerimientos, estableciendo las funcionalidades y alcances del prototipo a desarrollar.

³ Corporación multinacional de origen estadounidense, fundada por Bill Gates y Paul Allen, dedicada al desarrollo de sistemas de información, en especial su sistema operativo Windows.

El quinto capítulo, presenta el diseño de la arquitectura del prototipo y su entorno.

El sexto capítulo, describe los detalles de la implementación del prototipo, desde la preparación del entorno y los dispositivos hasta la estructura de los componentes programados.

El séptimo capítulo, define y muestra los resultados de las pruebas del prototipo.

El octavo capítulo, entrega la síntesis y las conclusiones respectivas, de manera de plasmar las apreciaciones del desarrollo y los resultados del proyecto.

El noveno capítulo, sugiere algunas ideas para trabajos futuros con el prototipo u otros sistemas similares.

El décimo capítulo, define los conceptos y siglas relevantes, entregando una mayor comprensión de aspectos claves abordados en el informe.

El undécimo capítulo, presenta las referencias utilizadas para el enfoque y definición del proyecto.

2. Problemática.

El problema abordado en el proyecto se enfoca a las clases de Lenguaje de enseñanza básica, la cual posee bajos rendimientos académicos, como se describió en el capítulo anterior.

La frase *"Estamos en una época en la que el lenguaje de la imagen está muy distante del lenguaje de nuestra generación, que es el del libro"* [6], describe con exactitud la realidad actual, en que las lecturas, los libros, ya no son el medio más importante de comunicar conocimiento: ahora las imágenes, audios y vídeos, conocido en el mundo de la tecnología como multimedia, es la nueva y eficaz forma de comunicación.

Las clases de Lenguaje en Chile, se han mantenido con una simplicidad monótona desde décadas [7]. A pesar de la implantación de nuevas tecnologías de la información en otras áreas educativas [8], que han mejorado la mayoría de los procesos de enseñanza, todavía es costoso encontrar tecnologías que ayuden al aprendizaje de nuestro lenguaje, y que utilicen estos nuevos medios de comunicar el conocimiento, y aún más, que los profesores las adopten [9].

Una importante causa de la poca evolución de las metodologías de enseñanza es que los profesores por naturaleza enseñan de la misma forma en que fueron enseñados: *"Se sabe de manera general que los docentes reproducen en sus prácticas la manera como fueron enseñados por sus propios profesores en la escuela y en la educación superior"* [9].

Se debe tratar de innovar en las salas de clase, "No vale la pena ir a una clase en la que el profesor se limita a seguir al pie de la letra unos apuntes o un texto, simplemente leyéndolo. Se trata de evitar el aburrimiento, la rutina. Que cada clase sea una aventura nueva." [7].

2.1. Problemas específicos.

A continuación, se describen los problemas específicos identificados, en orden descendente por trascendencia.

- a. Decadencia de las metodologías tradicionales de enseñanza en las clases: las clases de Lenguaje se han mantenido sin cambios desde hace décadas, centrándose en la enseñanza del código, según [9] y [10], lo que disminuye la motivación y con ello sesga el aprendizaje del estudiante [11].
- b. Escasa innovación y uso de tecnologías en las clases: el mundo está avanzando rápido y constantemente, y por sobre todo los avances tecnológicos han significado un cambio en la forma de vida, pero las clases, en especial las de Lenguaje se han quedado atrasadas en este proceso de cambio.

Según [12], donde se realizó una clasificación por etapas de integración tecnológica en la educación en América Latina, Chile está en una fase de Integración, lo que significa que existen programas de inserción masiva de tecnologías, pero no se han incorporado las TIC de manera sistemática e integral en el proceso de enseñanza-aprendizaje, lo cual se logra en la siguiente etapa definida en este estudio, llamada etapa de Transformación.

- c. Resistencia de los profesores a la adopción de nuevas tecnologías: el rápido avance de las tecnologías y la mala experiencia con herramientas informáticas muy complicadas de usar, han resultado en una gran resistencia por parte de los profesores a adoptar nuevas tecnologías que permitan mejorar sus clases.

“La experiencia y la teoría nos muestran que toda apuesta por el cambio implica también la exposición a la resistencia.” [13].

- d. Desmotivación de los estudiantes: la desmotivación es un problema muy importante en la efectividad del aprendizaje en la educación. *“No hay alumnos flojos, sólo alumnos desmotivados”* [14]. Según [15], la desmotivación en los estudiantes tiene un fuerte impacto en las metas académicas, la efectividad de las estrategias de aprendizaje y la autoeficacia del estudiante.
- e. Bajo rendimiento académico en Lenguaje: los malos resultados académicos, como demuestran los resultados de la SIMCE abordados en el capítulo anterior, son un problema preocupante.

3. Definición del proyecto.

En este capítulo se describen los detalles de la solución propuesta, sus objetivos y metodologías de trabajo.

3.1. Objetivos del proyecto.

En esta sección se indica el objetivo general del proyecto y sus objetivos específicos.

3.1.1. Objetivo general.

Diseñar e implementar un prototipo de sistema interactivo educativo, bajo el nombre de Kleng, utilizando las características sensoriales que ofrece el dispositivo Kinect para Microsoft Windows (ver Anexo B), específicamente la captura y procesamiento de audio (voz), y la captura y procesamiento de imagen (gestos, movimientos), con la misión de apoyar las actividades pedagógicas de la asignatura de Lenguaje de enseñanza básica (1° a 4° básico) en la sala de clases.

3.1.2. Objetivos específicos.

- a. Investigar el uso del SDK de Kinect, mediante la revisión de bibliografía complementaria y tutoriales, para lograr la implementación del prototipo.
- b. Implementar un módulo de reconocimiento de voz (palabras y frases) a través del reconocimiento de palabras de *Microsoft Speech Platform*, que permitirá reconocer lecturas, no sólo palabras o comandos, por parte del usuario.
- c. Implementar un módulo de reconocimiento gesticular de una mano a través de las utilidades de *Microsoft Skeleton Tracking*, que permitirá apuntar objetos en la visualización digital (monitor, proyector).
- d. Implementar actividades didácticas utilizando los sensores de audio (voz) del sensor Kinect, específicamente medir el tiempo tardado y la dicción del estudiante al realizar una lectura.

- e. Implementar actividades didácticas utilizando los sensores de imagen (movimiento) del sensor Kinect, específicamente actividades de términos pareados.

3.2. Ambiente de Ingeniería de Software.

En esta sección se definirá el ambiente de trabajo en el que se desarrolló el proyecto.

3.2.1. Modelo de desarrollo.

Para el desarrollo completo de este proyecto se utilizó un modelo de desarrollo de software iterativo e incremental (ver Figura 4).

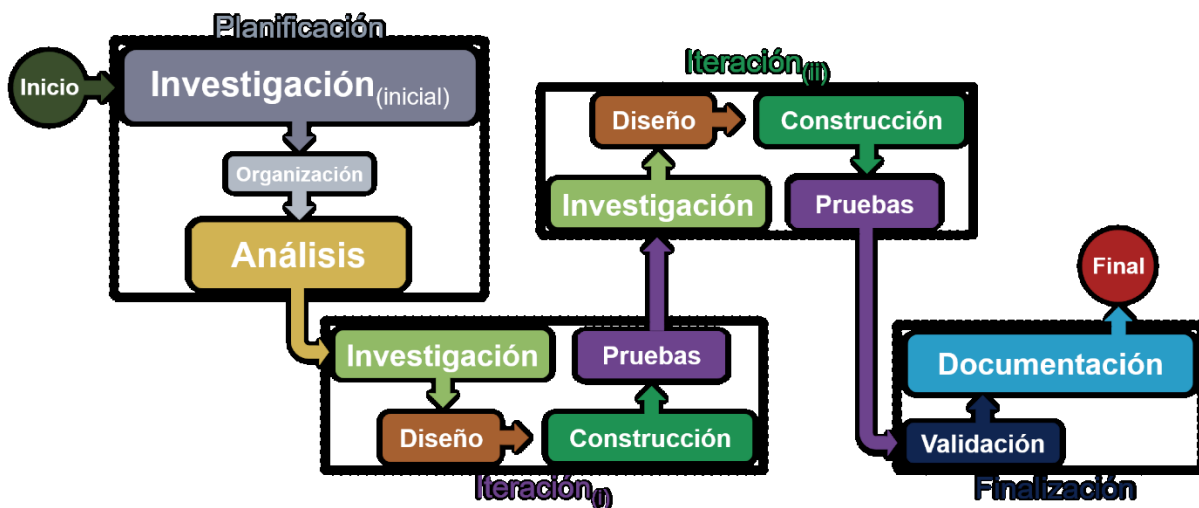


Figura 4. Modelo iterativo de desarrollo de software utilizado en el proyecto.

La ventaja de utilizar este modelo de desarrollo de software, es que permite agrupar tareas en iteraciones, generando al final de cada una de estas, software funcional (incrementos), y de esta forma, haciendo posible probar su funcionalidad y corregir posibles errores de planificación o construcción en la siguiente iteración, lo que además le entrega flexibilidad al desarrollo.

Además, por la naturaleza del proyecto, donde se diferencian claramente la construcción del módulo de procesamiento de voz y el módulo de procesamiento

gesticular de una mano, se pueden distribuir los esfuerzos de implementación según la funcionalidad entregada por cada módulo, en cada iteración.

La inclusión de la etapa de Investigación al inicio de cada iteración, se justifica debido a que es necesario adquirir conocimientos antes de diseñar el módulo correspondiente y, entender gradualmente el uso de las tecnologías asociadas para una implementación eficaz y robusta.

3.2.2. Descripción de las etapas de desarrollo.

El modelo presentado en la sección anterior se divide en 4 etapas, siendo la primera de estas inductiva, las 2 siguientes iteraciones de desarrollo, y una última etapa de cierre.

El proyecto se inicia con la etapa de Planificación, donde se realiza una investigación inductiva a las tecnologías relacionadas al proyecto, de manera de evaluar la factibilidad técnica, organizar las tareas del proyecto y, finalmente, establecer los requerimientos y detalles de implementación del sistema.

Luego se llevan a cabo las 2 iteraciones, que dan como resultado el desarrollo del software, siendo la primera iteración la que genera el módulo de reconocimiento de voz, y la segunda, el módulo de reconocimiento gesticular de una mano. Al final de la segunda iteración se realiza la integración de los 2 módulos, y se prueban las actividades didácticas, en un aula.

Por último, para cerrar el proyecto, la etapa de Finalización, en la cual se corrigen errores y se realiza el cierre de la documentación.

La descripción detallada de las etapas del modelo se realiza en el Anexo C.

3.2.3. Desarrollo basado en componentes.

Debido a que las funcionalidades que requiere Kleng son diferenciables, y el desarrollo de alguna de ellas no es precondition para el desarrollo de la otra, en específico el reconocimiento de voz y el de gestos, se tomó la decisión de separar las 2 funcionalidades requeridas en módulos.

Se realizó principalmente para facilitar la programación, mejorar la reutilización de código y generar una documentación más clara y fácil de utilizar.

3.2.4. Plataforma de desarrollo.

Kinect, construido por Microsoft, está diseñado en consecuencia para su propio sistema operativo, Windows, y sus herramientas de desarrollo oficiales también funcionan sobre él.

- a. Microsoft Windows 10 Pro version 1607 x64. Sistema operativo en el cual se instalaron los kits de desarrollo y se implementó el prototipo del proyecto.
- b. Kinect SDK 1.8 para Windows: conjunto de librerías que permiten la programación para el sensor Kinect en lenguajes de alto nivel.
- c. Microsoft Visual Studio 2015 (Service Pack 3): IDE⁴ requerido para programar el sensor Kinect, junto a la plataforma .NET en el lenguaje C#.

A nivel de hardware, se contó con un computador compatible con los productos de Microsoft y el dispositivo sensorial, descritos a continuación.

- d. Portátil Asus X556UQ: computador en el cual se desarrolló Kleng y se llevaron a cabo las pruebas principales de software.
 - Procesador Intel i7 6500u doble núcleo a 2.5 GHz.
 - Memoria Principal 8 GB a 2133 MHz.
 - Puerto USB 2.0.
- e. Dispositivo Kinect modelo 1517: sensor parte del kit de desarrollo Microsoft Kinect.
 - Cable de alimentación.
 - Cable conexión USB.

⁴ Entorno de Desarrollo Integrado (*del inglés Integrated Development Environment*). Entorno para el desarrollo de aplicaciones.

Todos estos sistemas y tecnologías son requeridas oficialmente como base para programar para Kinect, aunque existen excepciones del software en sus versiones libres (www.openkinect.org) para sistemas Linux y MacOS.

3.2.5. Herramientas relevantes de apoyo al desarrollo.

Las principales tecnologías que apoyaron la implementación de Kleng fueron el motor de reconocimiento de audio Microsoft Speech Platform, la librería de audio NAudio y la librería para la generación de documentos en PDF iTextSharp. Por último, el conjunto de estilos Material Design in XAML Toolkit permitió implementar una mejor interfaz gráfica.

La especificación completa de las herramientas de desarrollo, se pueden consultar en el Anexo D.

3.3. Relevancia del proyecto.

Un proyecto informático con las características descritas en este capítulo, supone un impulso importante para la región y el país en el ámbito de innovación educativa, dando un paso en la toma de acciones para intentar revertir la problemática desarrollada en capítulos anteriores (ver capítulo 2, p. 14).

Aunque el proyecto no consiste en desarrollar un sistema completo para liberar al mercado, este tiene la ambición de impulsar la construcción de futuros sistemas, enfocados a actividades de procesamiento de voz y de control de elementos sin mandos físicos, sobre todo a proyectos en la Universidad del Bío-Bío y la región.

Según un estudio realizado por los investigadores de la Universidad Carnegie Mellon [16], también citado en [17], los juegos educativos Kinect son cinco veces más eficaces que sus versiones estándar, lo que le da un nuevo impulso motivacional a los proyectos basados en esta tecnología.

Las ventajas de usar estas tecnologías en la educación han sido de tal impacto, que incluso los colegios internacionales SEK⁵, han comenzado a incorporar Kinect (la versión de Xbox 360) en su sistema educativo [18], de manera de aportar con la formación integral del estudiante.

Además, según el portal Educación 3.0 [18], la incorporación de Kinect en los colegios permitirá potenciar 2 factores claves en el aula de clases: el entretenimiento y la motivación del estudiante. La relación con la problemática de estos factores se relata en capítulos anteriores (ver sección 2.1, p. 14).

Sin duda son muchos los motivos y ventajas para trabajar con sistemas de este tipo, es por eso que, mientras más desarrolladores e instituciones educativas se interesen en proyectos similares, mayor es la probabilidad de que el cambio en el proceso de enseñanza-aprendizaje se produzca con una menor resistencia de los profesores y con mayor diversidad de innovaciones, aportando así una importante parte que hace falta en la educación actual en Chile.

⁵ Institución Internacional, fundada en el año 1892, que entrega a sus estudiantes programas de estudio bilingüe y que comienza desde la educación preescolar.

4. Análisis.

En este capítulo se entrega una perspectiva general de la funcionalidad del prototipo.

4.1. Perfil de usuario.

El desarrollo de Kleng se enfoca en la implementación de actividades didácticas para niños de enseñanza básica, que cursen los grados entre 1° y 4° básico de la clase de Lenguaje, con una edad comprendida entre los 5 a 10 años. No obstante, otro usuario directo es el operador del prototipo propiamente tal, el profesor de Lenguaje, por lo tanto, el enfoque de usabilidad también apunta hacia él.

4.2. Especificación de requerimientos.

En esta sección se definen las cualidades del prototipo.

4.2.1. Visión global.

El prototipo tiene como fin demostrar las capacidades del sensor Kinect, para el desarrollo de aplicaciones que utilicen el reconocimiento de voz (palabras, oraciones, textos) y reconocimiento de gestos de una mano para el control de elementos en la visualización por pantalla (o proyector). Además, motivará a los desarrolladores a crear soluciones basadas en esta tecnología orientadas a la educación.

4.2.2. Alcances.

El prototipo busca impulsar la creación de sistemas de apoyo didáctico a las clases de Lenguaje en enseñanza básica, mediante el uso de las capacidades de Kinect en actividades demostrativas para los niños.

El prototipo podrá usarse de manera experimental, pero de ninguna forma, supondrá un apoyo a todas las actividades realizadas por el profesor y, mucho menos, reemplazará su función en el aula.

4.2.3. Objetivos del prototipo.

- Impulsar el interés en el desarrollo de aplicaciones con Kinect para la educación.

- Demostrar las capacidades de Kinect para el desarrollo de aplicaciones didácticas.
- Aportar con un módulo de reconocimiento de lecturas con evaluación de modulación (dicción), contador de errores y tiempo demorado y, un módulo de reconocimiento de gestos de una mano para seleccionar opciones o elementos en pantalla, junto a la implementación de actividades experimentales que utilicen estas 2 principales funciones.

Las funcionalidades básicas establecidas respecto al reconocimiento de lecturas serán:

- a. Medir el tiempo de lectura.
- b. Evaluar la modulación o dicción.
- c. Contar las palabras correctas e incorrectas en un tiempo determinado de lectura.

Las actividades básicas definidas respecto al reconocimiento de gestos serán:

- a. Seleccionar elementos en pantalla para realizar términos pareados.

4.2.4. Descripción global del prototipo.

En esta sección se describen las especificaciones generales del prototipo y su entorno, definiendo las interacciones de usuario, hardware y requerimientos técnicos para su ejecución.

4.2.4.1. Uso y funcionalidades básicas.

Kleng debe permitir al estudiante realizar una lectura, la cual debe ser reconocida palabra por palabra, con un cronómetro que mide el tiempo que demora, y contadores que deben mostrar la cantidad de palabras correctas, incorrectas, y su puntuación en modulación, con una nota en el rango 1.0 a 7.0.

Kleng debe permitir al estudiante realizar términos pareados y seleccionar elementos en pantalla, mediante movimientos y gestos de una mano, apuntando y trazando la trayectoria de estos.

La interacción básica (ver Figura 5) entre el usuario principal (el estudiante) y Kleng, consiste en el sensor Kinect conectado al computador, que recibe la voz o capta los

gestos del usuario y los envía al computador para ser procesados. Además, la interfaz de visualización debe ser la pantalla o monitor del computador, o alternativamente, un proyector compatible.

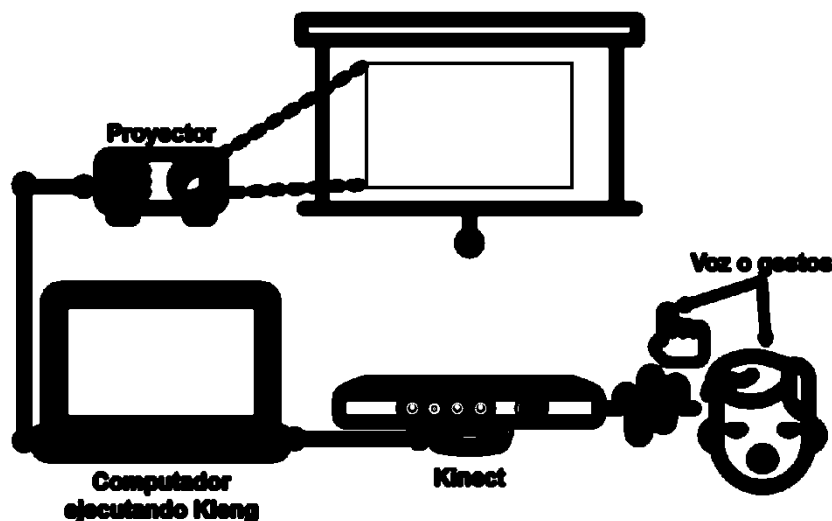


Figura 5. Diagrama de interacción básica entre el usuario y el prototipo.

4.2.5. Requerimientos técnicos.

Para ejecutar adecuadamente Kleng, es necesario contar con un computador con características técnicas mínimas (ver Tabla 1).

Procesador.	Doble núcleo 2.0 GHz con arquitectura 32 bits (x86) o 64 bits (x64).
Memoria Principal.	2 GB.
Tarjeta de Vídeo.	Soporte para DirectX ⁶ 9.0c.
Monitor (LCD o proyector).	Resolución XGA ⁷ (1024x768) escala 4:3.
Teclado y Ratón.	

Tabla 1. Especificaciones técnicas requeridas del computador para ejecutar Kleng.

Además, es indispensable poseer el dispositivo Kinect adecuado (ver Tabla 2).

⁶ API gráfica diseñada para los sistemas Microsoft Windows.

⁷ Extended Graphics Array: estándar para visualización de gráficos creado por IBM en 1990. Define una resolución de 1024 píxeles de ancho por 768 píxeles de alto.

Sensor.	Kinect modelo 1473 o 1517.
	Cable de alimentación.
	Cable conexión USB.

Tabla 2. Especificaciones técnicas requeridas del sensor Kinect.

Finalmente, Kleng solo será compatible y ejecutado en un sistema operativo que cumpla los requerimientos mínimos y con un conjunto de librerías instaladas en él (ver Tabla 3).

Sistema Operativo.	Microsoft Windows 7 o superior.
Librerías.	DirectX 9.0c.
	Microsoft .NET Framework 4.0.
	Microsoft Speech Runtime, idioma español es-ES o es-MX.

Tabla 3. Especificaciones técnicas de software requeridas para ejecutar Kleng.

4.2.6. Definición de requerimientos.

En esta sección, se definen los requerimientos del prototipo.

Los significados de los términos utilizados para definir cada requerimiento, se establecen a continuación (ver Tabla 4).

ID.	Identificador del requerimiento. Permite trazarlo en etapas posteriores.
Nombre.	Texto breve que permite referirse a un requisito de manera simple
Descripción.	Texto que describe el requerimiento.
Cumplimiento.	Estado de cumplimiento del requerimiento. <u>Cumple</u> : implementación conforme a lo requerido. <u>Cumple Parcialmente</u> : implementación cumple sólo una parte de la funcionalidad requerida. <u>No Cumple</u> : implementación no existe o no cubre lo requerido.
Prioridad.	Relevancia de la implementación del requisito. <u>Esencial</u> : característica indispensable. <u>Condicional</u> : característica relevante. <u>Adicional</u> : característica de menor relevancia.
Verificación.	Método de verificación del requisito en la etapa de pruebas <u>Prueba</u> : el requisito se verificará o validará mediante la ejecución de un

	<p>procedimiento de prueba caracterizado por un criterio de éxito/fracaso claramente definido y objetivo.</p> <p><u>Demostración</u>: el requisito se verificará o validará demostrando que el sistema puede desarrollar la función o capacidad establecida en el requisito.</p> <p><u>Inspección</u>: el requisito se verificará o validará mediante inspección visual de un artefacto producido durante el ciclo de desarrollo o el sistema mismo, para confirmar que se satisface la característica establecida en el requisito.</p> <p><u>Análisis</u>: se desarrollará un análisis que demuestre que el sistema es capaz de realizar la funcionalidad descrita.</p>
Observación.	(Opcional). Observaciones a la descripción o al estado de cumplimiento.

Tabla 4. Definición de secciones en la especificación de requerimientos.

4.2.6.1. Requerimientos funcionales.

Descripción de las funcionalidades del prototipo (ver Tabla 5). Los requerimientos funcionales comienzan con el identificador KLFR.

ID:	KLFR01
Nombre:	Mostrar tiempo
Descripción:	El prototipo mostrará un cronómetro con el tiempo transcurrido desde el inicio hasta el fin de la lectura.
Cumplimiento:	Cumple ✓
Prioridad:	Esencial
Verificación:	Demostración
ID:	KLFR02
Nombre:	Seleccionar lectura
Descripción:	El prototipo permitirá cargar el archivo de texto correspondiente a la lectura que se desea realizar.
Cumplimiento:	Cumple ✓
Prioridad:	Esencial
Verificación:	Prueba
ID:	KLFR03
Nombre:	Iniciar lectura
Descripción:	El prototipo permitirá iniciar una nueva lectura, durante la cual el estudiante leerá el texto predefinido en voz alta hacia el sensor Kinect.
Cumplimiento:	Cumple ✓
Prioridad:	Esencial
Verificación:	Demostración
ID:	KLFR04
Nombre:	Finalizar lectura
Descripción:	El prototipo permitirá finalizar la lectura en curso, mostrando los resultados de esta (tiempo, modulación, palabras correctas e incorrectas).
Cumplimiento:	Cumple ✓

Prioridad:	Esencial
Verificación:	Demostración
ID:	KLFR05
Nombre:	Mostrar palabras correctas
Descripción:	El prototipo mostrará la cantidad de palabras correctas que el estudiante lee, al finalizar la lectura.
Cumplimiento:	Cumple Parcialmente *✓
Prioridad:	Esencial
Verificación:	Demostración
Observación:	Si la palabra es leída por el estudiante, pero el sensor no la reconoce adecuadamente, no se contabilizará. Esto es un problema sin solución para la actual arquitectura física, que depende de la buena modulación, ruido y la eficacia acústica de la sala. Además, se debe tomar en cuenta el rango de funcionamiento del sensor (ver Anexo B).
ID:	KLFR06
Nombre:	Mostrar palabras incorrectas
Descripción:	El prototipo mostrará la cantidad de palabras incorrectas que el estudiante lee, al finalizar la lectura.
Cumplimiento:	Cumple Parcialmente *✓
Prioridad:	Condicional
Verificación:	Demostración
Observación:	Si la palabra es leída por el estudiante, pero el sensor no la reconoce adecuadamente, se contabilizará como incorrecta. Esto es un problema sin solución para la actual arquitectura física, que depende de la buena modulación, ruido y la eficacia acústica de la sala. Además, se debe tomar en cuenta el rango de funcionamiento del sensor (ver Anexo B).
ID:	KLFR07
Nombre:	Mostrar nota modulación
Descripción:	El prototipo mostrará la nota de modulación del estudiante, en un rango desde 1.0 a 7.0, al finalizar la lectura.
Cumplimiento:	Cumple ✓
Prioridad:	Esencial
Verificación:	Demostración
ID:	KLFR08
Nombre:	Seleccionar término pareado
Descripción:	El prototipo permitirá cargar el archivo de texto correspondiente al término pareado que se desea realizar.
Cumplimiento:	Cumple ✓
Prioridad:	Esencial
Verificación:	Prueba
ID:	KLFR09
Nombre:	Iniciar términos pareados
Descripción:	El prototipo permitirá iniciar una actividad de términos pareados, durante la cual el estudiante apuntará con su dedo índice para mover un cursor que trazará la línea de pareo, y empuñará su mano para seleccionar los términos en la ubicación actual del cursor, y de esta forma poder unirlos.
Cumplimiento:	Cumple Parcialmente *✓
Prioridad:	Esencial

Verificación:	Demostración
Observación:	El “empuñar” la mano es un gesto costoso de detectar computacionalmente, y difícil con el hardware del proyecto, por lo tanto, se reemplazó por el gesto de “empujar”. Además, surgió la necesidad de implementar gestos para subir y bajar a través de listas, debido a que no todos los términos a unir pueden ser visualizados simultáneamente por pantalla, por lo que se implementaron los gestos de deslizar hacia arriba o abajo la mano secundaria (mano alternativa) para subir o bajar respectivamente.
ID:	KLFR10
Nombre:	Finalizar términos pareados
Descripción:	El prototipo permitirá finalizar la actividad de términos pareados en curso, mostrando los resultados de esta por pantalla (buenas, malas).
Cumplimiento:	Cumple ✓
Prioridad:	Esencial
Verificación:	Demostración
ID:	KLFR11
Nombre:	Mostrar términos pareados buenos
Descripción:	El prototipo mostrará la cantidad de términos pareados correctos, al finalizar la lectura.
Cumplimiento:	Cumple ✓
Prioridad:	Esencial
Verificación:	Demostración
ID:	KLFR12
Nombre:	Mostrar términos pareados malos
Descripción:	El prototipo mostrará la cantidad de términos pareados incorrectos, al finalizar la lectura.
Cumplimiento:	Cumple ✓
Prioridad:	Condicional
Verificación:	Demostración
ID:	KLFR13
Nombre:	Estado de conexión
Descripción:	El prototipo mostrará el estado de conexión del sensor Kinect (Conectado, Inicializando, Desconectado).
Cumplimiento:	Cumple ✓
Prioridad:	Adicional
Verificación:	Demostración
ID:	KLFR14
Nombre:	Generar reporte lectura
Descripción:	El prototipo generará automáticamente un reporte con los resultados (ver sección 7.5.1, p. 66) de la lectura realizada por el estudiante, al final de esta, en formato PDF.
Cumplimiento:	Cumple ✓
Prioridad:	Condicional
Verificación:	Demostración
ID:	KLFR15
Nombre:	Generar reporte términos pareados
Descripción:	El prototipo generará automáticamente un reporte con los resultados (ver sección 7.5.2, p. 67) de la actividad de términos pareados realizada por el

	estudiante, al final de esta, en formato PDF.
Cumplimiento:	Cumple ✓
Prioridad:	Condicional
Verificación:	Demostración
ID:	KLFR16
Nombre:	Menú lectura
Descripción:	El prototipo mostrará un menú (GUI) con opciones para acceder a Iniciar, Cargar y Finalizar la lectura.
Cumplimiento:	Cumple ✓
Prioridad:	Condicional
Verificación:	Demostración
ID:	KLFR17
Nombre:	Menú términos pareados
Descripción:	El prototipo mostrará un menú (GUI) con opciones para acceder a Iniciar, Cargar y Finalizar la actividad de términos pareados (ver sección 5.3.1, p. 43).
Cumplimiento:	Cumple ✓
Prioridad:	Condicional
Verificación:	Demostración
ID:	KLFR18
Nombre:	Menú principal
Descripción:	El prototipo mostrará un menú (GUI) con opciones para acceder a al menú de lectura y al menú de términos pareados (ver sección 5.3.1, p. 43).
Cumplimiento:	Cumple ✓
Prioridad:	Condicional
Verificación:	Demostración

Tabla 5. Requerimientos funcionales.

4.2.6.2. Requerimientos no funcionales.

En esta sección se describen las cualidades (no tangibles) del prototipo (ver Tabla 6). Las restricciones del proyecto comienzan con el identificador KLPR, mientras que los requerimientos de calidad con el identificador KLQR.

ID:	KLPR01
Nombre:	Sensor
Descripción:	El proyecto se desarrollará utilizando el sensor de Microsoft Kinect modelo 1517, ya que se dispone de este.
ID:	KLPR02
Nombre:	Sistema Operativo
Descripción:	El prototipo se desarrollará y ejecutará sobre el sistema operativo Microsoft Windows 10, ya que se dispone de este y es compatible con las tecnologías utilizadas.
ID:	KLPR03
Nombre:	IDE
Descripción:	El proyecto se desarrollará en el IDE de Microsoft Visual Studio por su potencia e integración con el sensor Kinect.

ID:	KLPR04
Nombre:	Lenguaje de Programación
Descripción:	El prototipo se desarrollará en el lenguaje C#, sobre la plataforma .NET, ya que el SDK de Kinect lo requiere como lenguaje nativo.
ID:	KLPR05
Nombre:	Idioma
Descripción:	La interfaz del prototipo deberá visualizarse en idioma español, debido que se enfoca a clases de Lenguaje en Chile.
ID:	KLQR01
Nombre:	Mensajes explícitos
Descripción:	El prototipo mostrará mensajes con la información explícita correspondiente, ya sea un mensaje de información, advertencia, excepción o error.
Tipo:	Requerimiento de calidad. Usabilidad-Inteligibilidad
Cumplimiento:	Cumple ✓
Prioridad:	Condicional
Verificación:	Prueba
ID:	KLQR02
Nombre:	Modularidad
Descripción:	El prototipo será desarrollado en módulos con funciones específicas orientadas a un solo objetivo.
Tipo:	Requerimiento de calidad. Mantenibilidad.
Cumplimiento:	Cumple ✓
Prioridad:	Esencial
Verificación:	Análisis
ID:	KLQR03
Nombre:	Documentación código
Descripción:	El código fuente de los módulos del prototipo será documentado, definiendo cada funcionalidad, parámetros y salidas de las funciones y clases.
Tipo:	Requerimiento de calidad. Mantenibilidad-Reusabilidad
Cumplimiento:	Cumple ✓
Prioridad:	Esencial
Verificación:	Inspección
ID:	KLQR04
Nombre:	Convención de código C#
Descripción:	El código fuente de los módulos del prototipo será escrito de acuerdo a los estándares del lenguaje C#, proporcionados por Microsoft en [19].
Tipo:	Requerimiento de calidad. Mantenibilidad.
Cumplimiento:	Cumple ✓
Prioridad:	Condicional
Verificación:	Inspección
ID:	KLQR05
Nombre:	Logo Kleng
Descripción:	La interfaz gráfica del prototipo contará con un personaje infantil que diferencia a Kleng. El personaje aparecerá en la interfaz de inicio.
Tipo:	Requerimiento de calidad. Calidad en uso-Satisfacción.
Cumplimiento:	Cumple ✓
Prioridad:	Condicional
Verificación:	Demostración

Tabla 6. Requerimientos no funcionales.

4.2.7. Casos de Uso.

Representación conceptual de las situaciones en las cuales los actores interactúan con Kleng. Los actores pueden ser sistemas externos o personas (usuarios).

4.2.7.1. Actores.

En esta sección se definen los actores que interaccionan con Kleng, mediante una breve descripción de sus funciones.

En las interacciones está presente el profesor de Lenguaje de enseñanza básica y el estudiante: sus roles sobre el prototipo se pueden intercambiar, por eso se representa a ambos como Usuario.

Actor	Usuario.	ID	UCA01
Descripción	Usuario de Kleng.		
Funciones	<ol style="list-style-type: none"> 1. Iniciar la aplicación. 2. Establecer la actividad (lectura o términos pareados) a realizar. 3. Realizar la actividad. 4. Finalizar la actividad. 5. Visualizar los resultados. 6. Finalizar la aplicación. 		

Tabla 7. UCA01. Usuario.

Actor	Kinect.	ID	UCA02
Descripción	Dispositivo que captura la información sensorial (voz y gestos del estudiante) y los comunica a Kleng.		
Funciones	<ol style="list-style-type: none"> 1. Capturar la información sensorial. 2. Transferir la información a la aplicación Kleng. 		

Tabla 8. UCA02. Kinect.

4.2.7.2. Diagrama de Casos de Uso.

Diagrama que muestra las operaciones de interacción entre los actores y el prototipo, permitiendo así agrupar y ordenar los requerimientos en las situaciones en las cuales entidades externas interactuarán con Kleng.

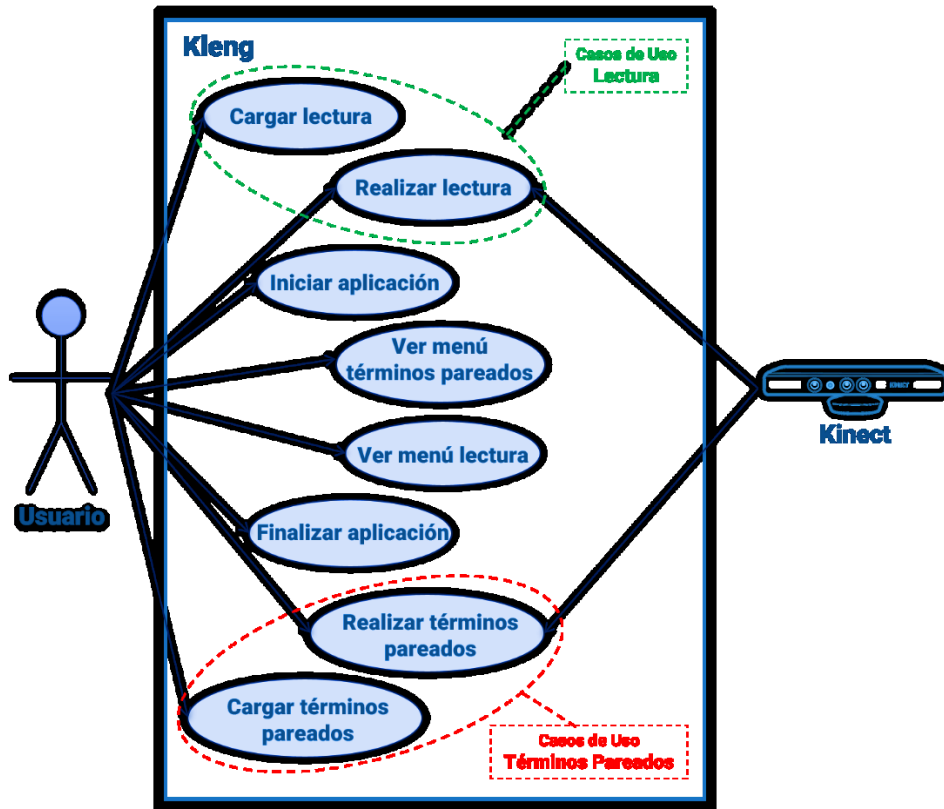


Figura 6. Diagrama de Casos de Uso (UCD).

4.2.7.3. Descripción de Casos de Uso.

En esta sección, se describe cada caso de uso (óvalo) del diagrama anterior (ver Figura 6). Aunque Kleng es un prototipo, se hará referencia a él como sistema, por formalidad.

Cada especificación posee una descripción, en lenguaje natural y de manera general, de la interacción del o los actores con el sistema (prototipo).

En el flujo básico se listan secuencialmente dichas acciones y sus consecuencias. Los flujos alternativos describen las acciones que suceden luego de excepciones, en etapas del flujo básico, que pueden generar más de un flujo de continuación; se dice que es fuera del caso común o ideal.

Finalmente, las pre-condiciones y post-condiciones indican las condiciones necesarias para que respectivamente inicie o finalice el flujo.

Iniciar aplicación.			
ID.	UC01	Referencias.	KLFR18, KLFR13
Descripción.	El usuario inicia la aplicación.		
Actores.	Usuario UCA01	Actores secundarios.	-
Pre-condiciones.	-	Post-condiciones.	-
Flujo Básico.	<p>[1] Usuario ejecuta la aplicación.</p> <p>[2] Sistema muestra una pantalla de inicio con el logo de Kleng.</p> <p>[3] Sistema ingresa al menú principal.</p>	Flujos Alternativos.	-

Tabla 9. UC01. Iniciar aplicación.

Finalizar aplicación.			
ID.	UC02	Referencias.	KLFR18
Descripción.	El usuario cierra la aplicación.		
Actores.	Usuario UCA01	Actores secundarios.	-
Pre-condiciones.	UC01 (aplicación iniciada)	Post-condiciones.	-
Flujo Básico.	<p>[1] Usuario solicita al sistema finalizar ejecución.</p> <p>[2] Sistema muestra un cuadro de confirmación de salida.</p> <p>[3] Usuario confirma la finalización del sistema.</p> <p>[4] Sistema finaliza su ejecución.</p>	Flujos Alternativos.	<p><u>Cancelar salida:</u></p> <p>[3.1] Usuario cancela la finalización del sistema.</p> <p>[3.2] Sistema regresa a menú principal.</p>

Tabla 10. UC02. Finalizar aplicación.

Ver menú lectura.			
ID.	UC03	Referencias.	KLFR16
Descripción.	El sistema despliega el menú de lectura mediante la visualización digital.		
Actores.	Usuario UCA01	Actores secundarios.	-
Pre-condiciones.	UC01 (aplicación iniciada)	Post-condiciones.	-
Flujo Básico.	<p>[1] Usuario solicita al sistema mostrar menú de lectura.</p> <p>[2] Sistema muestra el menú de lectura con las opciones "Iniciar Lectura", "Cargar Lectura" y "Finalizar Lectura".</p>	Flujos Alternativos.	<p><u>Volver a menú principal:</u></p> <p>[1.1] Usuario solicita al sistema cerrar menú de lectura.</p> <p>[1.2] Sistema regresa a menú principal.</p>

Tabla 11. UC03. Ver menú lectura.

Ver menú términos pareados.			
ID.	UC04	Referencias.	KLFR16
Descripción.	El sistema despliega el menú de términos pareados mediante la visualización digital.		
Actores.	Usuario UCA01	Actores secundarios.	-
Pre-condiciones.	UC01 (aplicación iniciada)	Post-condiciones.	-
Flujo Básico.	<p>[1] Usuario solicita al sistema mostrar menú de términos pareados.</p> <p>[2] Sistema muestra el menú de lectura con las opciones “<i>Iniciar Términos Pareados</i>”, “<i>Cargar Términos Pareados</i>” y “<i>Finalizar Términos Pareados</i>”.</p>	Flujos Alternativos.	<p>Volver a menú principal:</p> <p>[1.1] Usuario solicita al sistema cerrar menú de lectura.</p> <p>[1.2] Sistema regresa a menú principal.</p>

Tabla 12. UC04. Ver menú términos pareados.

Realizar lectura.			
ID.	UC05	Referencias.	KLFR01, KLFR03, KLFR04, KLFR05, KLFR06, KLFR07, KLFR14
Descripción.	El usuario inicia al sistema para que escuche la lectura.		
Actores.	Usuario UCA01, Kinect UCA02	Actores secundarios.	-
Pre-condiciones.	UC01 (aplicación iniciada), UC03 (menú lectura), UC07 (lectura cargada)	Post-condiciones.	-
Flujo Básico.	<p>[1] Usuario solicita al sistema iniciar la lectura.</p> <p>[2] Sistema inicia el cronómetro y comienza a escuchar (activa los micrófonos de Kinect) al estudiante.</p> <p>[3] Usuario lee el texto en</p>	Flujos Alternativos.	<p><u>Lectura no cargada:</u></p> <p>[2.1] Sistema muestra mensaje de excepción “<i>Ninguna lectura cargada</i>”.</p> <p>[2.2] Usuario confirma mensaje.</p> <p>[2.3] Sistema regresa a menú lectura.</p>

	<p>voz alta hacia el sensor Kinect.</p> <p>[4] Kinect envía la información sensorial al sistema.</p> <p>[5] Sistema procesa la información sensorial.</p> <p>[6] Usuario solicita al sistema finalizar la lectura.</p> <p>[7] Sistema desactiva los micrófonos de Kinect.</p> <p>[8] Sistema detiene el cronómetro.</p> <p>[9] Sistema muestra los resultados (nota modulación, palabras correctas e incorrectas y duración de la lectura).</p> <p>[10] Sistema pregunta si desea guardar los resultados.</p> <p>[11] Usuario confirma guardar los resultados.</p> <p>[12] Sistema pregunta datos del estudiante (nombre, RUN).</p> <p>[13] Usuario ingresa los datos requeridos.</p> <p>[14] Usuario confirma el ingreso de datos.</p> <p>[15] Sistema valida el RUN.</p> <p>[16] Sistema genera un reporte (documento) con los resultados (nota modulación, palabras correctas e incorrectas y duración de la lectura), los datos del estudiante (nombre, RUN) y la fecha actual.</p>		<p>RUN inválido:</p> <p>[15.1] Sistema verifica que el RUN es inválido.</p> <p>[15.2] Sistema muestra mensaje de excepción “<i>RUN inválido</i>”.</p> <p>[15.3] Usuario confirma mensaje.</p> <p>[15.4] Sistema regresa a paso [10] del flujo básico.</p>
--	--	--	--

Tabla 13. UC05. Realizar lectura.

Realizar términos pareados.			
ID.	UC06	Referencias.	KLFR09, KLFR10,

			KLFR11, KLFR12, KLFR15
Descripción.	El usuario inicia la actividad de términos pareados para unirlos con la gesticulación de una de sus manos.		
Actores.	Usuario UCA01, Kinect UCA02	Actores secundarios.	-
Pre-condiciones.	UC01 (aplicación iniciada), UC04 (menú términos pareados), UC08 (términos pareados cargados)	Post-condiciones.	-
Flujo Básico.	<p>[1] Usuario solicita al sistema iniciar términos pareados.</p> <p>[2] Sistema muestra términos a la izquierda con los términos a unir a la derecha.</p> <p>[3] Sistema rastrea (activa la cámara de Kinect) los gestos de la mano del estudiante.</p> <p>[4] Kinect envía la información sensorial al sistema.</p> <p>[5] Sistema procesa la información sensorial.</p> <p>[6] Sistema mueve el cursor acorde al gesto de apuntar del usuario y selecciona el término si este empuña su mano.</p> <p>[7] Usuario utiliza gestos de su mano para unir los términos (empuñar para seleccionar el término y apuntar para trazar la ruta con un cursor).</p> <p>[8] Usuario solicita al sistema finalizar la actividad.</p> <p>[9] Sistema desactiva la cámara de Kinect.</p> <p>[10] Sistema muestra los resultados (uniones buenas y malas).</p> <p>[11] Sistema pregunta si desea guardar los resultados.</p> <p>[12] Usuario confirma guardar los resultados.</p> <p>[13] Sistema pregunta datos del estudiante (nombre, RUN).</p> <p>[14] Usuario ingresa los datos requeridos.</p> <p>[15] Usuario confirma el ingreso de datos.</p> <p>[16] Sistema valida el RUN.</p> <p>[17] Sistema genera un reporte (documento) con los resultados (uniones buenas y malas), los datos del estudiante (nombre, RUN) y la fecha actual.</p>	Flujos Alternativos.	<p><u>Términos pareados no cargados:</u></p> <p>[2.1] Sistema muestra mensaje de excepción "Ningún término pareado cargado".</p> <p>[2.2] Usuario confirma mensaje.</p> <p>[2.3] Sistema regresa a menú de términos pareados.</p> <p><u>RUN inválido:</u></p> <p>[16.1] Sistema verifica que el RUN es inválido.</p> <p>[16.2] Sistema muestra mensaje de excepción "RUN inválido".</p> <p>[16.3] Usuario confirma mensaje.</p> <p>[16.4] Sistema regresa a paso [10] del flujo básico.</p>

Tabla 14. UC06. Realizar términos pareados.

Cargar lectura.			
ID.	UC07	Referencias.	KLFR02
Descripción.	El usuario carga la lectura al sistema para que pueda realizarse.		

Actores.	Usuario UCA01	Actores secundarios.	-
Pre-condiciones.	UC01 (aplicación iniciada), UC03 (menú lectura)	Post-condiciones.	-
Flujo Básico.	<p>[1] Usuario solicita al sistema cargar la lectura.</p> <p>[2] Sistema muestra una interfaz de selección de lectura.</p> <p>[3] Usuario selecciona la lectura.</p> <p>[4] Usuario confirma la carga de la lectura seleccionada.</p> <p>[5] Sistema valida el formato de la lectura.</p> <p>[6] Sistema carga la lectura.</p> <p>[7] Sistema muestra mensaje "<i>Lectura cargada satisfactoriamente</i>".</p>	Flujos Alternativos.	<p>Fichero inválido:</p> <p>[6.1] Sistema muestra mensaje de error "<i>Formato de lectura inválido</i>".</p> <p>[6.2] Usuario confirma mensaje.</p> <p>[6.3] Sistema regresa a menú lectura.</p>

Tabla 15. UC07. Cargar lectura.

Cargar términos pareados.			
ID.	UC08	Referencias.	KLFR02
Descripción.	El usuario carga los términos pareados al sistema para que pueda realizarse.		
Actores.	Usuario UCA01	Actores secundarios.	-
Pre-condiciones.	UC01 (aplicación iniciada), UC04 (menú términos pareados)	Post-condiciones.	-
Flujo Básico.	<p>[1] Usuario solicita al sistema cargar los términos pareados.</p> <p>[2] Sistema muestra una interfaz de selección de términos pareados.</p> <p>[3] Usuario selecciona los términos pareados.</p> <p>[4] Usuario confirma la carga de los términos pareados seleccionados.</p> <p>[5] Sistema valida el formato de los términos pareados.</p> <p>[6] Sistema carga los términos pareados.</p> <p>[7] Sistema muestra mensaje "<i>Términos pareados cargados satisfactoriamente</i>".</p>	Flujos Alternativos.	<p>Fichero inválido:</p> <p>[6.1] Sistema muestra mensaje de error "<i>Formato de términos pareados inválido</i>".</p> <p>[6.2] Usuario confirma mensaje.</p> <p>[6.3] Sistema regresa a menú lectura.</p>

Tabla 16. UC08. Cargar términos pareados.

5. Diseño.

En este capítulo se presenta el diseño general del prototipo. La explicación se enfoca en la arquitectura, ya que no se realizó un diseño detallado de clases, las cuales se describen en el capítulo de implementación (ver Capítulo 6, p. 50).

5.1. Arquitectura.

En esta sección, se define la arquitectura o infraestructura lógica del prototipo, mediante modelos que representan los componentes, librerías y relaciones relevantes.

5.1.1. Arquitectura del entorno.

La arquitectura Kleng y su entorno, se basa en un desarrollo sobre Windows [20], dentro del cual se enlazan a Kleng diversas librerías, especialmente para el procesamiento de voz, a la vez que se comunica con el sensor Kinect a través de los controladores y librerías nativas de su SDK.

Los dos entornos generales más importantes de la arquitectura (ver Figura 7), son el sensor Kinect y el sistema operativo Windows los cuales se comunican mediante una conexión serial USB 2.0. El desencadenante del flujo de información entre estos dos entornos es la aplicación Kleng, dentro del sistema operativo.

El orden del esquema en la arquitectura definida, considera a los elementos base y de bajo nivel en la parte inferior de la figura, mientras que las librerías y la aplicación, a un alto nivel de abstracción, de encuentran en niveles superiores del esquema.

Se comenzará explicando el esquema desde el nivel inferior.

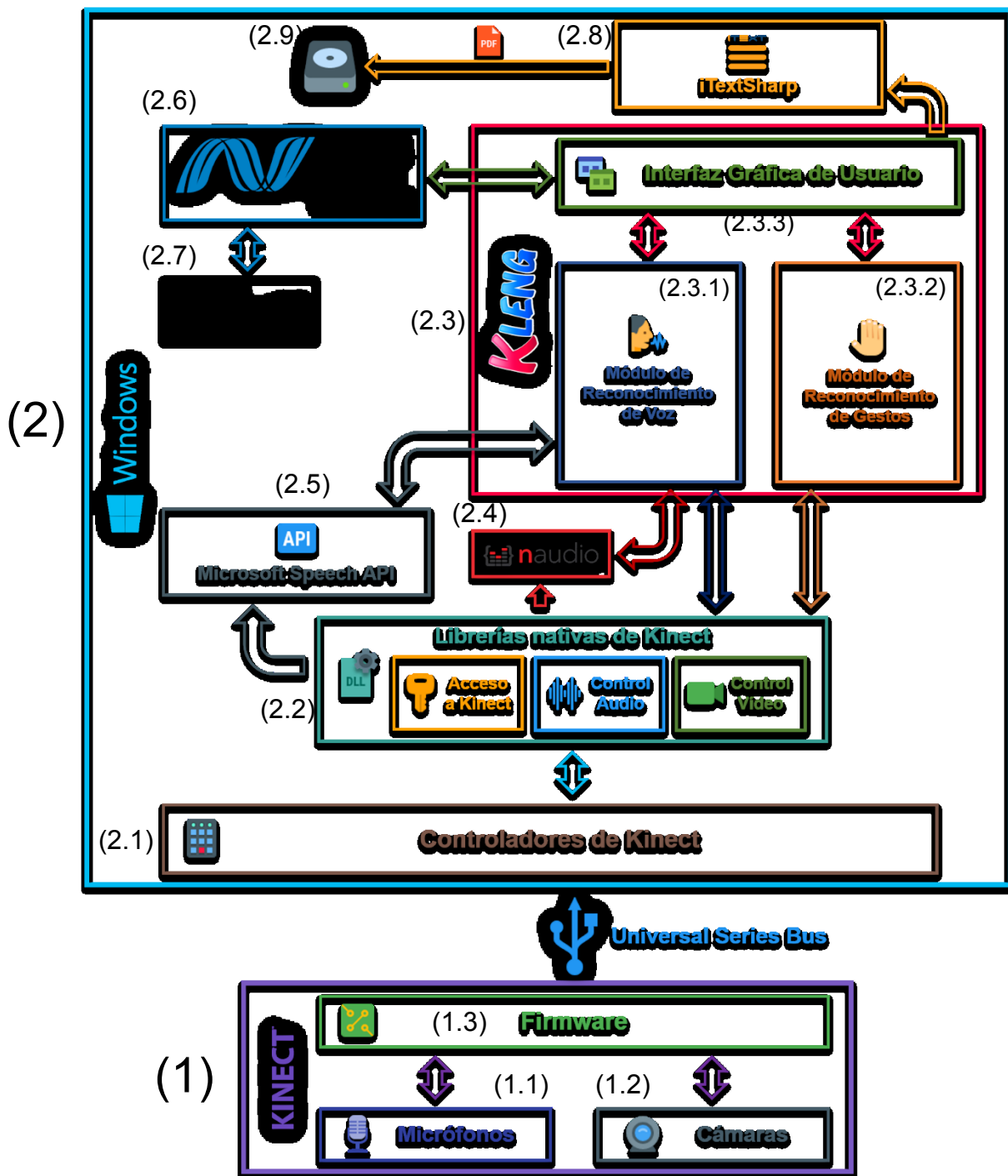


Figura 7. Arquitectura global (de entorno) de Kleng.

Kinect (1) representa todo el hardware de Kinect. Se representan los micrófonos (1.1) y las cámaras (1.2), los cuales reciben la información del entorno y se comunican internamente con el firmware (1.3) del dispositivo, el cual es el encargado de enviar y recibir la información al sistema operativo mediante conexión USB.

Windows (2), es el sistema operativo que permite la ejecución de todo el software relacionado al control y comunicación entre Kinect (1) y Kleng (2.3).

Los controladores de Kinect (2.1), permiten la comunicación del sistema operativo a bajo nivel con el sensor, funcionando como el puente lógico para su funcionamiento. Las librerías nativas de Kinect (2.2), pertenecientes al SDK, facilitan el proceso de comunicación con Kinect, controlando mediante lenguajes de alto nivel, como C#, sus características funcionales.

Kleng (2.3) es la aplicación que gatilla todos los flujos de control de información. Esta se comunica directamente con las librerías nativas de Kinect (2.2), y con librerías adicionales, NAudio (2.4) y Microsoft Speech (2.5), para cumplir su funcionamiento.

NAudio (2.4) es una librería controlada por el módulo de reconocimiento de voz, en Kleng, es utilizada para la grabación de audio digital, en este caso, en mono PCM a 16 kHz.

Microsoft Speech (2.5) es una API de procesamiento y reconocimiento de voz para sistemas Windows, además optimizada para el sensor Kinect. Es controlada también por el módulo de reconocimiento de voz (2.3.1) y recibe la información sensorial de las librerías de Kinect (2.2), entregando información al módulo de las palabras reconocidas en el audio.

El módulo de reconocimiento de gestos (2.3.2), se comunica directamente con las librerías nativas de Kinect (2.2), en donde Microsoft Skeleton Tracking realiza la simplificación de la conexión, que permite el control de las cámaras para reconocer los gestos.

La interfaz gráfica de usuario (2.3.3), hace uso de .NET Framework (2.6), el cual a su vez utiliza la librería gráfica DirectX (2.7) para dibujar los elementos en pantalla.

Finalmente, la interfaz gráfica (2.3.3) usa iTextSharp (2.8) para almacenar los resultados en un documento en formato PDF en el almacenamiento principal (2.9).

5.1.2. Arquitectura del prototipo.

Este apartado define la arquitectura específica de Kleng.

La arquitectura utilizada es por niveles o capas, que separa la lógica funcional (control) de la lógica del diseño (presentación o vista). La capa de persistencia, muy propia de esta arquitectura, se omitió debido a que el prototipo no presenta persistencia de datos, sólo generación de reportes.

La capas son descritas a continuación (ver Figura 8).

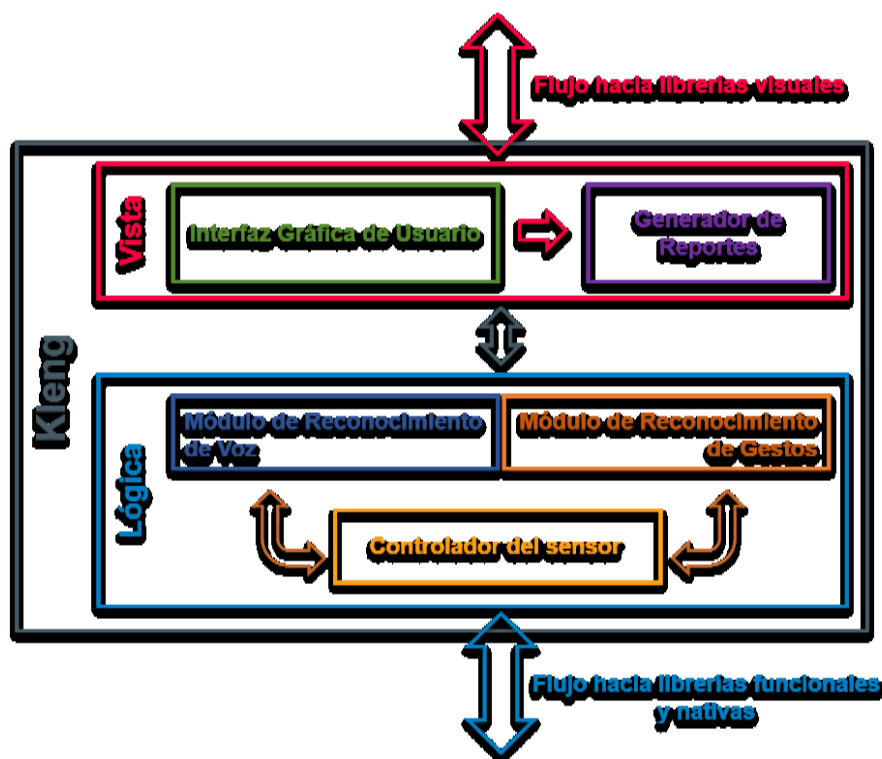


Figura 8. Arquitectura de Kleng.

- a. Capa Presentación (Vista): interfaz gráfica de usuario. Corresponde a todas las vistas que permiten que el usuario interactúe con el prototipo. Posee un pequeño módulo para generar los reportes, el cual utiliza iTextSharp. Este nivel también

cuenta con controladores por cada vista (requisito del estándar .NET), un diseño conocido como MVC⁸.

- b. Capa Lógica (Procesamiento): Corresponde a todas las funcionalidades de procesamiento de voz y gestos (los módulos), que permiten un comportamiento consistente a los requerimientos de Kleng. Existe un componente que permite el acceso y activación del sensor Kinect (controlador del sensor), disponible para los dos módulos.

5.2. Estándar de implementación de los módulos.

Se generó un estándar de construcción de los componentes clave relacionados con el sensor, con el fin de facilitar su posterior utilización. Se establecieron las siguientes reglas de comportamiento.

- Inicialización: las propiedades del componente son inicializadas para su uso y su conexión con componentes externos. Para esto, se debe implementar un método Initialize().
- Comienzo: comienzan las operaciones de procesamiento del componente. Para esto, se debe implementar un método Start().
- Procesamiento: los datos recibidos son procesados y enviados a componentes externos mediante eventos desencadenantes. Las tareas de recepción y envío de datos se realizan preferentemente en hilos de proceso sincronizados, los cuales se inician con métodos Start() y se finalizan con métodos Stop().
- Finalización: terminan las operaciones de procesamiento del componente. Para esto, se debe implementar un método Stop().
- Liberación memoria: cuando no se requiera más el componente en memoria principal, se liberan los recursos asociados a este y se destruyen sus

⁸ Modelo Vista-Controlador. Patrón de arquitectura de software, que se enfoca principalmente en separar las vistas de la lógica dentro de un sistema

referencias. Para esto, se utilizan métodos Destroy(), o Dispose() en caso de utilizar la interfaz IDisposable.

5.3. Interfaz gráfica de usuario (GUI).

En esta sección, se muestran los modelos de interfaz gráfica de usuario.

5.3.1. GUI de lectura.

La interfaz gráfica para la lectura, se enfoca en cargar, iniciar, finalizar y mostrar los resultados (ver Figura 9).

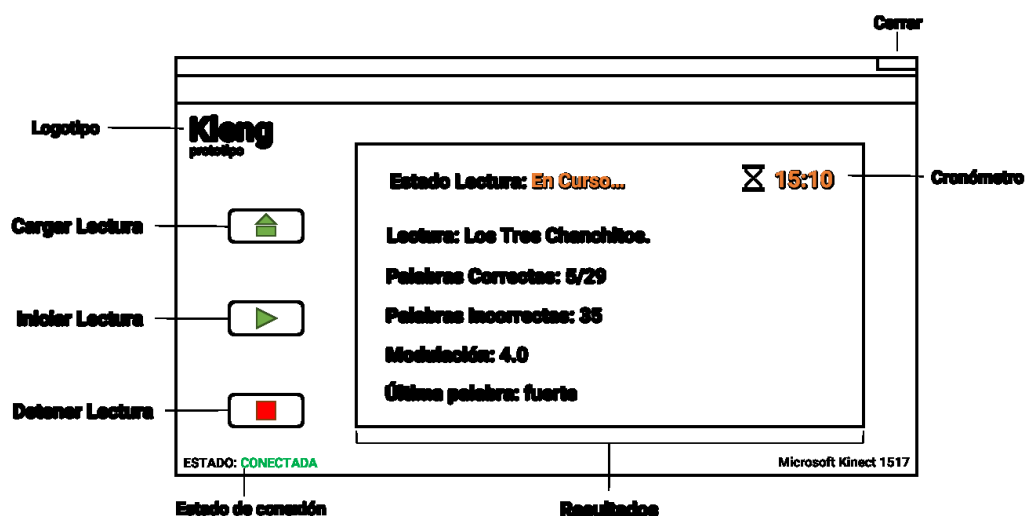


Figura 9. Diseño de GUI de la Lectura.

- Logotipo: logo de Kleng.
- Cargar Lectura: permite cargar la lectura desde un archivo de texto.
- Iniciar Lectura: activa los micrófonos del sensor Kinect y comienza a correr el tiempo en el cronómetro.
- Detener Lectura: desactiva los micrófonos de Kinect, detiene el cronómetro y despliega los resultados en la zona definida de la GUI.
- Resultados: resultados de la lectura.

- Estado de conexión: indica si el sensor Kinect está conectado, desconectado o inicializando.
- Cerrar: cierra la actividad de lectura.

Además, la acción de Cargar Lectura desplegará una interfaz con el nombre y el texto de la lectura (ver Figura 10).

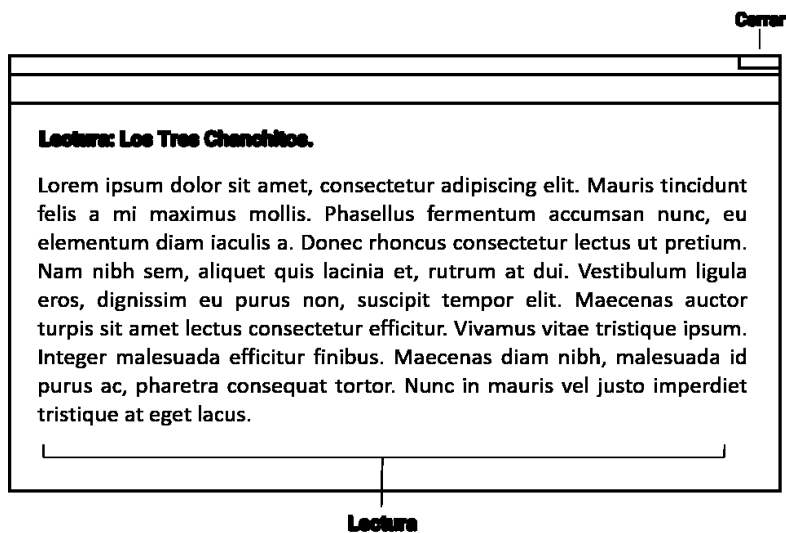


Figura 10. Diseño de GUI de la visualización de la lectura.

5.3.2. GUI de términos pareados.

La interfaz gráfica para la actividad de Términos Pareados se enfoca en cargar, iniciar, finalizar y mostrar los resultados de esta (ver Figura 11).

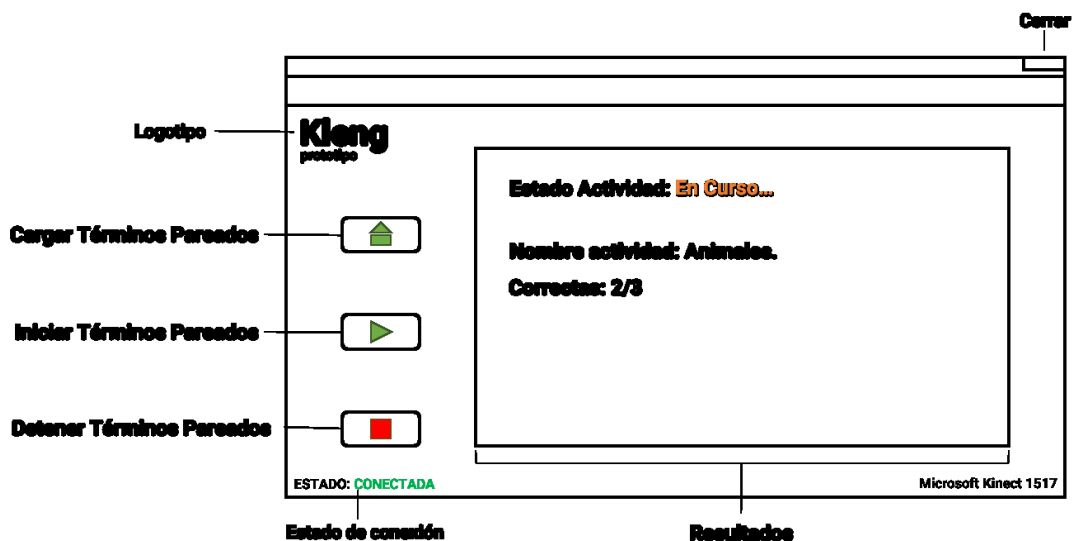


Figura 11. Diseño de GUI de Términos Pareados.

- Cargar Términos Pareados: permite cargar los términos desde un archivo de texto.
- Iniciar Términos Pareados: activa las cámaras del sensor Kinect y abre la GUI de la actividad de términos pareados a realizar.
- Parar Términos Pareados: desactiva las cámaras de Kinect y despliega los resultados en la zona definida de la GUI.

Observación: Algunos elementos ya se definieron en la GUI de lectura, por lo tanto, no se definen en la sección actual.

La opción de Iniciar Términos Pareados desplegará una interfaz con la actividad de términos para unir (ver Figura 12).

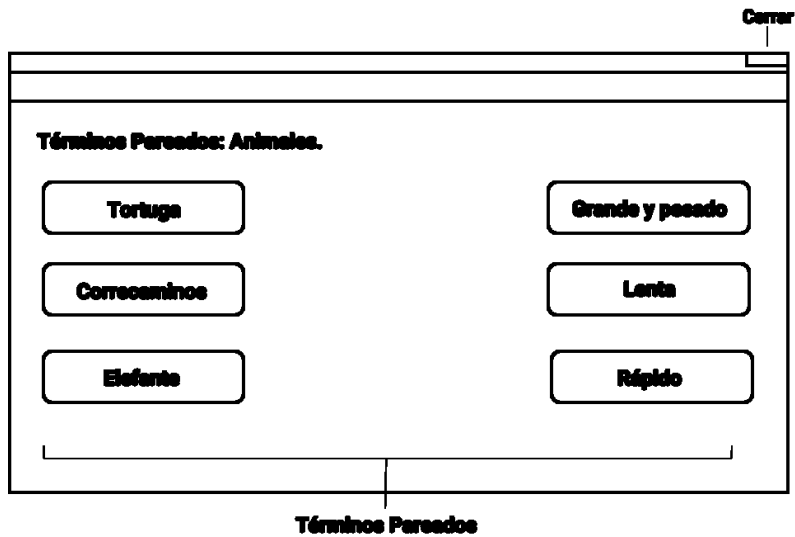


Figura 12. Diseño de GUI de la visualización de los términos pareados.

La visualización de los términos, se llevará a cabo en 2 columnas dispuestas verticalmente (izquierda y derecha), en las cuales se podrán seleccionar los términos de alguna de ellas para luego seleccionar su par en la otra columna.

5.4. Especificación de reportes.

En esta sección se especifica el formato de los reportes generados por Kleng.

5.4.1. Lectura.

El reporte de lectura permitirá identificar la fecha y actividad realizada, el estudiante relacionado y sus puntajes de modulación, tiempo y palabras correctas e incorrectas, como se muestra en la Figura 13.

31/12/9999 24:00
Nombre: Cristopher Alvear Candia. RUN: 12.454.338-k. Profesor: Juan Pérez Contreras.
Título: Mimo y su Mamá. Tiempo: 24:59:59. Modulación: 7.0. Palabras correctas: 99/99 Palabras incorrectas: 99.

Figura 13. Especificación reporte de Lectura.

5.4.2. Términos pareados.

El reporte de términos pareados permitirá identificar la fecha y actividad realizada, el estudiante relacionado y sus puntajes de términos correctamente pareados, como se muestra en la Figura 14.

31/12/9999 24:00
Nombre: Cristopher Alvear Candia. RUN: 12.454.338-k. Profesor: Juan Pérez Contreras.
Título: Animales. Términos correctos: 99/99

Figura 14. Especificación reporte de Términos Pareados.

5.5. Especificación de datos.

En esta sección se definen los grupos de datos que manejará el prototipo.

5.5.1. Datos de entrada.

Kleng utiliza los datos de entrada para generar los reportes. Además, para cada actividad deben cargar los datos de esta desde un archivo de texto, definidos en la sección anterior.

ID	Nombre Grupo	Datos	Fuente
KLID01	Datos Estudiante	R.U.N. (Rol Único Nacional). Nombre (ej. Christopher Alvear Candia).	Formulario.
KLID02	Datos Profesor	Nombre (ej. Juan Pérez Contreras).	Formulario.
KLID03	Lectura	Nombre (ej. Los 3 cerditos). Texto (lectura).	Archivo de texto (.txt).
KLID04	Términos Pareados	Nombre. Lista de términos (ej. Casa, Puerta).	Archivo de texto (.txt).

Tabla 17. Datos de entrada de Kleng.

La Tabla 17 muestra los grupos de datos de entrada para Kleng.

Un ejemplo del formato del texto que representa una lectura se puede observar en la Figura 15.

Las lecturas pueden incluir signos gramaticales (¿? ¡! ; : , .), múltiples líneas y párrafos.

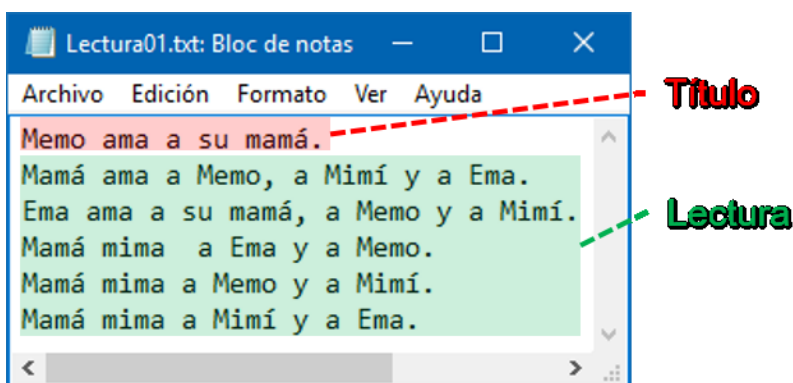


Figura 15. Ejemplo lectura predefinida en un archivo de texto.

Por último, un ejemplo del formato del texto que representa una actividad de términos pareados se puede observar en la Figura 16.

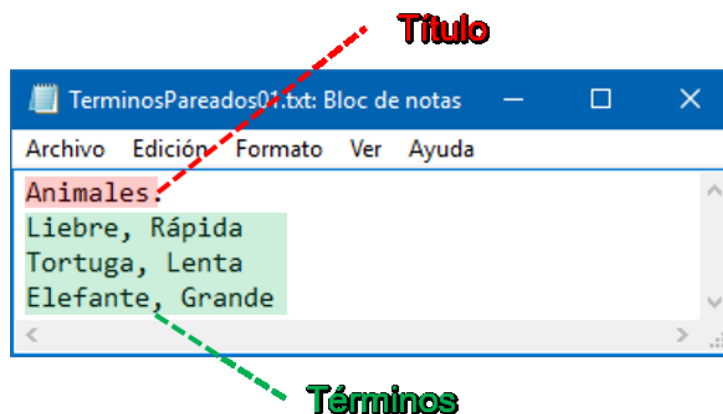


Figura 16. Ejemplo de términos pareados predefinido en un archivo de texto.

5.5.2. Datos de salida.

Kleng genera reportes opcionales para cada actividad realizada por el estudiante, con el objetivo de mantener un registro de los resultados obtenidos.

La Tabla 18 muestra los grupos de datos de salida, que básicamente, por su relevancia, pertenecen a la generación de reportes de las actividades.

ID	Nombre Grupo	Datos	Fuente
KLOD01	Datos Estudiante	R.U.N. (Rol Único Nacional). Nombre (ej. Christopher Alvear Candia).	GUI, Reporte (.pdf)
KLOD02	Datos Profesor	Nombre (ej. Juan Pérez Contreras).	Reporte (.pdf)
KLOD03	Lectura	Nombre (ej. Los 3 cerditos). Tiempo (ej. 2:34 minutos). Nota modulación de 1.0 a 7.0 (ej. 4.5). Palabras correctas (ej. 3/15). Palabras incorrectas.	GUI, Reporte (.pdf)
KLOD04	Términos Pareados	Nombre (ej. Animales). Uniones correctas. Uniones incorrectas.	GUI, Reporte (.pdf)
KLOD05	Reportes	Fecha (ej. 30-11-2016). Hora (12:30:50).	Reporte (.pdf)

Tabla 18. Datos salida de Kleng.

6. Implementación.

En este capítulo se explican la estructura de la capa de control y la interfaz gráfica final.

Las etapas de preparación para programar el sensor Kinect, con los detalles de instalación del dispositivo, sus controladores y uso básico se especifican en el Anexo E.

6.1. Estructura de los módulos.

Kleng procesa la entrada de audio mediante algoritmos multihilo, reconociendo una lectura previamente definida, mediante una técnica iterativa de reconocimientos parciales, es decir, analiza segmentos de la lectura continuamente comprobando si este se ajusta con el reconocimiento, de manera que permita identificar palabras o frases individuales sin considerar errores anteriores o futuros, y que esto afecte al progreso de la lectura.

Kleng procesa la entrada de imagen (vídeo) mediante algoritmos que indican la posición relativa dependiendo de la dirección en que el usuario apunte con su mano hacia la pantalla, y gestos para realizar acciones.

En esta sección, se describen las clases de mayor relevancia en Kleng. Los diagramas se simplificaron, de manera de mostrar sólo las relaciones, para posteriormente detallar cada clase con su estructura, atributos y operaciones.

La simbología de clases se establece en la Figura 17.

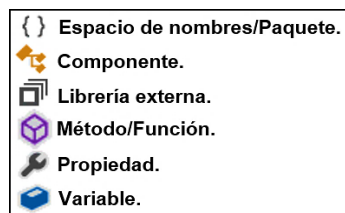


Figura 17. Simbología de íconos clases UML (Visual Studio).

6.2. Capa de control.

La capa de control o capa lógica de la aplicación, consta de los componentes de ejecución principales de Kleng.

Esta capa, fue implementada en el espacio de nombres (o paquete) llamado Components. Dentro, las clases se organizaron en los siguientes espacios:

- Raíz: es el espacio Components, donde se encuentra la clase de control principal de Kinect (KinectController), utilidades de programación multihilo (TimedThreadHandler) y funciones de entrada y salida a nivel de sistema operativo (FileUtils).
- VoiceRecognizer: espacio principal de la lógica de procesamiento de audio (VoiceRecognizer) y reconocimiento de la lectura (Reading).
- HandControl: espacio principal de la lógica de procesamiento de gestos de las manos (HandControl) y realización de términos pareados (PairingTerms).
- Exceptions: espacio donde se definen las excepciones de Kleng.
- Events: espacio donde se definen los eventos de Kleng.

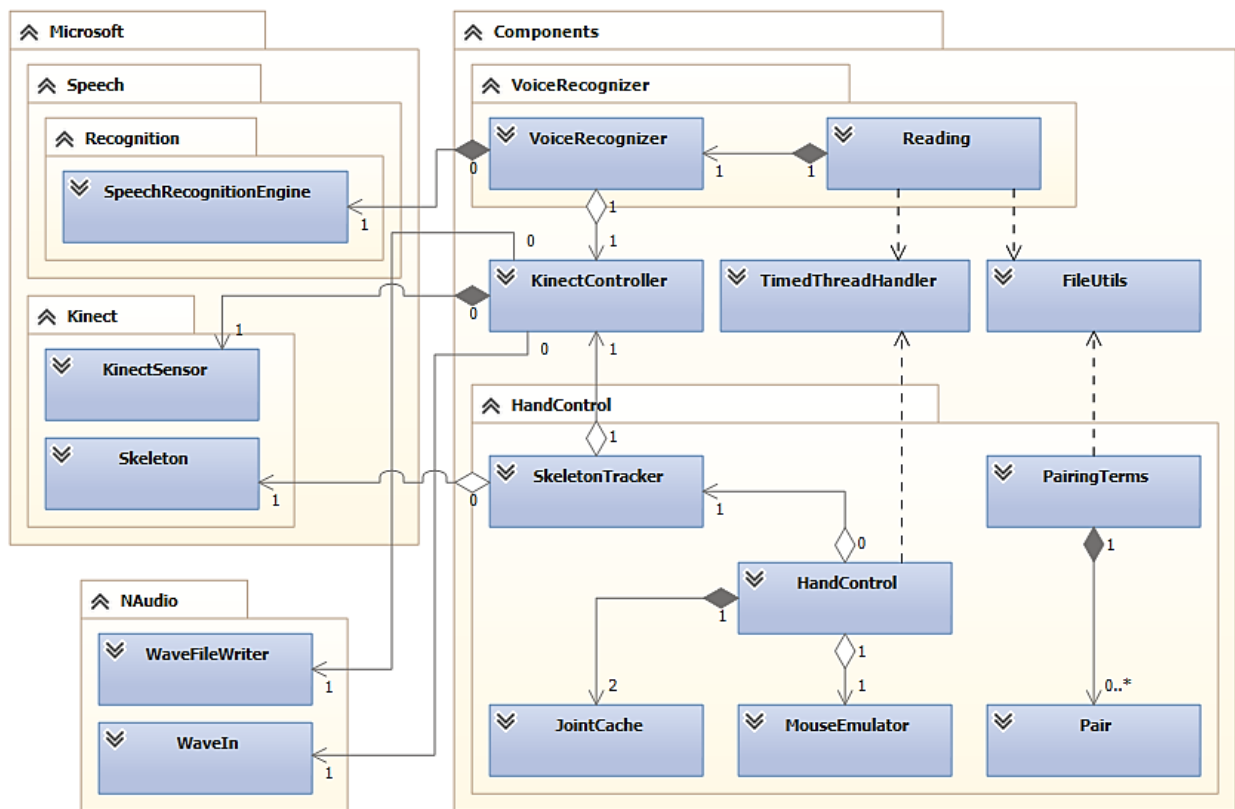


Figura 18. Diagrama UML de clases (sin detalle de clases) de la capa lógica.

El diagrama de clases UML de la capa lógica (ver Figura 18), muestra las relaciones entre cada una de las clases. Se presentan las clases relevantes a la lógica de Kleng, omitiendo los eventos y excepciones, y agregando los espacios de nombres y clases de las librerías nativas (Microsoft) y adicionales (NAudio) utilizadas, de manera de entender su importancia. A nivel general, los módulos se describen a continuación:

- a. VoiceRecognizer: reconoce la lectura del estudiante. Hace utilización de KinectController, el cual entrega acceso a los micrófonos, en específico, al flujo de audio desde Kinect. Además, su componente principal es la clase nativa SpeechRecognitionEngine, que permite el reconocimiento de comandos de voz.
- b. HandControl: reconocimiento de gestos de las manos. Al igual que el módulo anterior, hace utilización de KinectController, en este caso para acceder a las cámaras. Su componente principal es Skeleton, una clase que almacena la información recibida del flujo SkeletonStream desde Kinect para calcular las posiciones absolutas de las distintas partes de cuerpo en un plano tridimensional.

6.2.1. KinectController.

Clase de acceso principal a Kinect. Los atributos y funciones se observan a continuación (ver Figura 19):

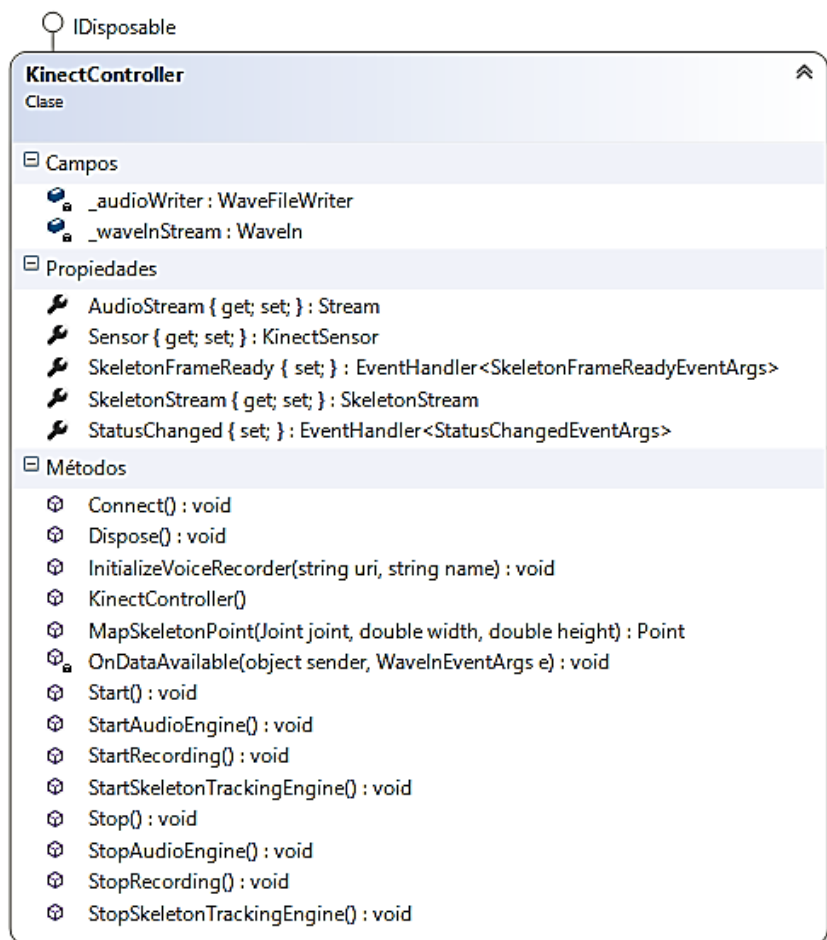


Figura 19. Especificación de clase UML de KinectController.

Sus 2 campos (_audioWriter y _waveInStream) son derivados de NAudio para la grabación del audio.

Las propiedades controlan el estado de Kinect y el acceso al flujo de audio y vídeo.

6.2.2. TimedThreadHandler.

Clase que permite la ejecución de hilos en intervalos de tiempo (ver Figura 20).

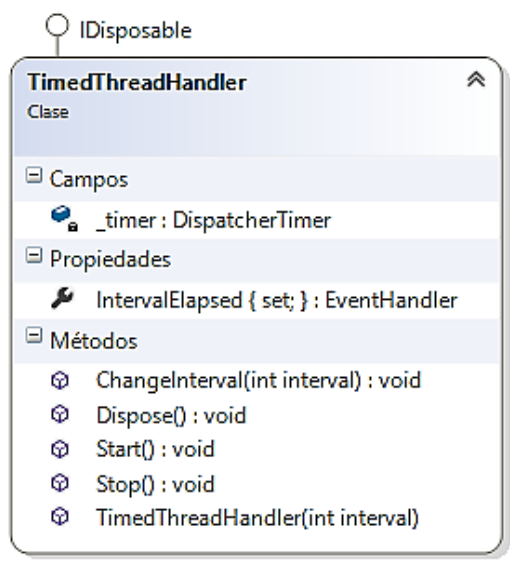


Figura 20. Especificación de clase UML de TimedThreadHandler.

6.2.3. FileUtils.

Conjunto de utilidades para la manipulación de archivos, directorios, lectura y escrituras en archivos de texto y reproducción de audio (ver Figura 21).

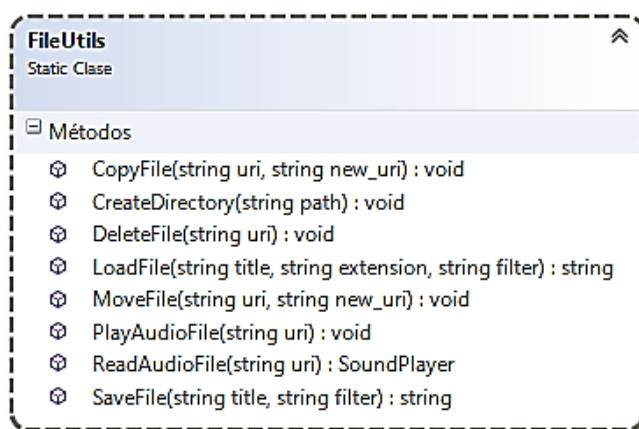


Figura 21. Especificación de clase UML de FileUtils.

6.2.4. KinectCapabilities.

Es una clase estática de configuración, donde se definen los valores de inicialización de Kleng.

Estos valores son utilizados para calibrar la sensibilidad del reconocimiento de voz y de gestos y, para modificar el rendimiento general del prototipo.

La especificación de las variables se realiza en el Anexo F.

6.2.5. Librerías nativas de Kinect.

Las librerías nativas de Kinect, permiten acceder al sensor y a sus funcionalidades de bajo nivel, en específico activar, desactivar sensores, controlar los flujos de audio y video y conocer el estado de este.

KinectSensor es la clase que permite activar o desactivar sensores de Kinect (cámara, sensor infrarrojo y micrófonos) y, conocer el estado general del sensor. Los estados más importantes se listan a continuación:

- Connected: sensor conectado y listo.
- Disconnected: sensor desconectado.
- Error: error en el sensor.
- Initializing: sensor conectado, pero todavía no listo.
- NotPowered: sensor conectado, pero sin fuente de alimentación.

La clase SpeechRecognitionEngine permite la utilización del motor de reconocimiento de voz creado por Microsoft, principalmente para sistemas Windows, pero que fue adaptado y optimizado para los dispositivos Kinect. Esta utilidad, esencial dentro del prototipo, permite definir gramáticas (conjunto de palabras o frases) para luego ser reconocidas desde el flujo de audio (la onda sonora). El principal problema que tiene esta clase, es que es lenta en el reconocimiento (lo que se solucionó en cierta medida con algoritmos multihilo), no es capaz de reconocer frases con rapidez y no permite la utilización del modo dictado, el cual realiza un reconocimiento de idiomas completos sin necesidad de definir gramáticas. El modo dictado está presente en el motor System Speech, pero este no está optimizado para el sensor Kinect, por lo tanto, los resultados son mediocres, aunque depende del objetivo de la aplicación, este podría ser útil.

Skeleton es una clase que almacena los datos en coordenadas tridimensionales capturados por las cámaras de Kinect, permitiendo conocer la posición absoluta de algún objeto delante del sensor.

6.2.6. NAudio.

Librería de código abierto que posee utilidades de procesamiento de audio destinadas a acelerar el desarrollo de aplicaciones en .NET. En Kleng, proporciona un mecanismo para capturar el flujo de entrada de audio y almacenarlo en un archivo con formato WAV⁹.



Figura 22. Logo de librería NAudio.

6.2.7. iTextSharp.

Librería de código abierto que posee utilidades de generación de documentos en formato PDF. En Kleng, permite la generación sencilla de reportes de los resultados de las actividades.



Figura 23. Logo de librería iTextSharp.

6.2.8. Módulo de reconocimiento de voz.

El módulo de reconocimiento de voz es el encargado de traducir la voz recibida por el sensor Kinect a texto, perteneciente a la lectura realizada por el estudiante, utilizando la librería Microsoft Speech, que posee un sistema de reconocimiento de palabras mediante una base de datos para el lenguaje español (de España o México).

El módulo, se conforma por 2 clases.

⁹ Formato de audio digital normalmente sin compresión de datos.

6.2.8.1. VoiceRecognizer.

Es la clase central del módulo, ya que contiene la lógica del reconocimiento de voz basada en Microsoft Speech. Maneja la gramática, que consiste en las palabras pertenecientes a la lectura que el motor de Microsoft Speech reconocerá e indicará si han sido reconocidas en el audio.

VoiceRecognizer usa un objeto de la clase KinectController con el objetivo de controlar y recibir información del flujo de audio desde el sensor Kinect.

El flujo de audio es redirigido hacia una instancia de la clase SpeechRecognitionEngine para que retorne las palabras y frases básicas reconocidas (llamados átomos, en el código).

La clase maneja los siguientes eventos derivados de SpeechRecognitionEngine:

- SpeechDetected: Evento que se lanza cuando la entrada de audio de Kinect recibe voz (cualquiera sea la palabra o frase).
- SpeechHypothesized: Evento que se lanza cuando el motor Microsoft Speech reconoce una sección de audio que probablemente pertenece a una palabra o frase de la gramática gestionada.
- SpeechRecognized: Evento que se lanza cuando la entrada de audio de Kinect recibe una palabra o frase que pertenece a la gramática.
- SpeechRejected: Evento que se lanza cuando la entrada de audio de Kinect recibe una palabra o frase que no pertenece a la gramática.

La estructura de VoiceRecognizer se observa a continuación (ver Figura 24):

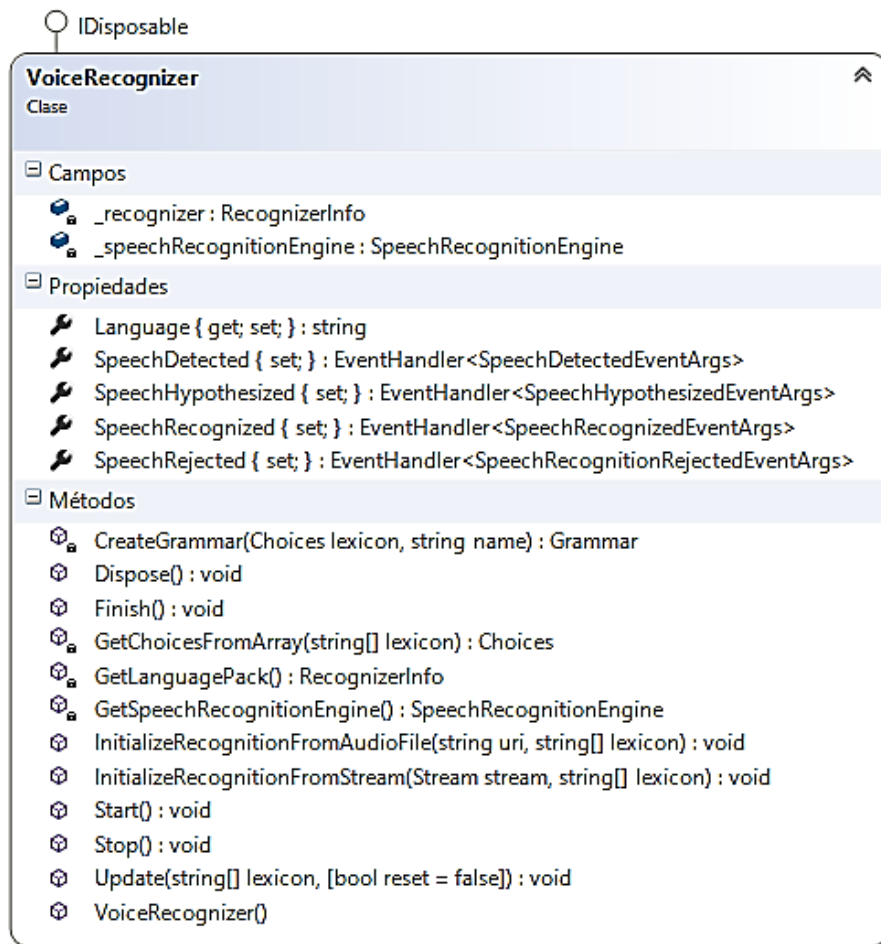


Figura 24. Especificación de clase UML de VoiceRecognizer.

6.2.8.2. Reading.

Reading es una clase que utiliza la información que entrega VoiceRecognizer, comparando en tiempo real el reconocimiento con el texto de la lectura y, despertar eventos asociados con el reconocimiento correcto, incorrecto o hipotético del flujo de audio. Estos eventos permiten avisar a las clases asociadas que estén utilizando a Reading (el Main por ejemplo), que algo ha pasado y necesita su atención.

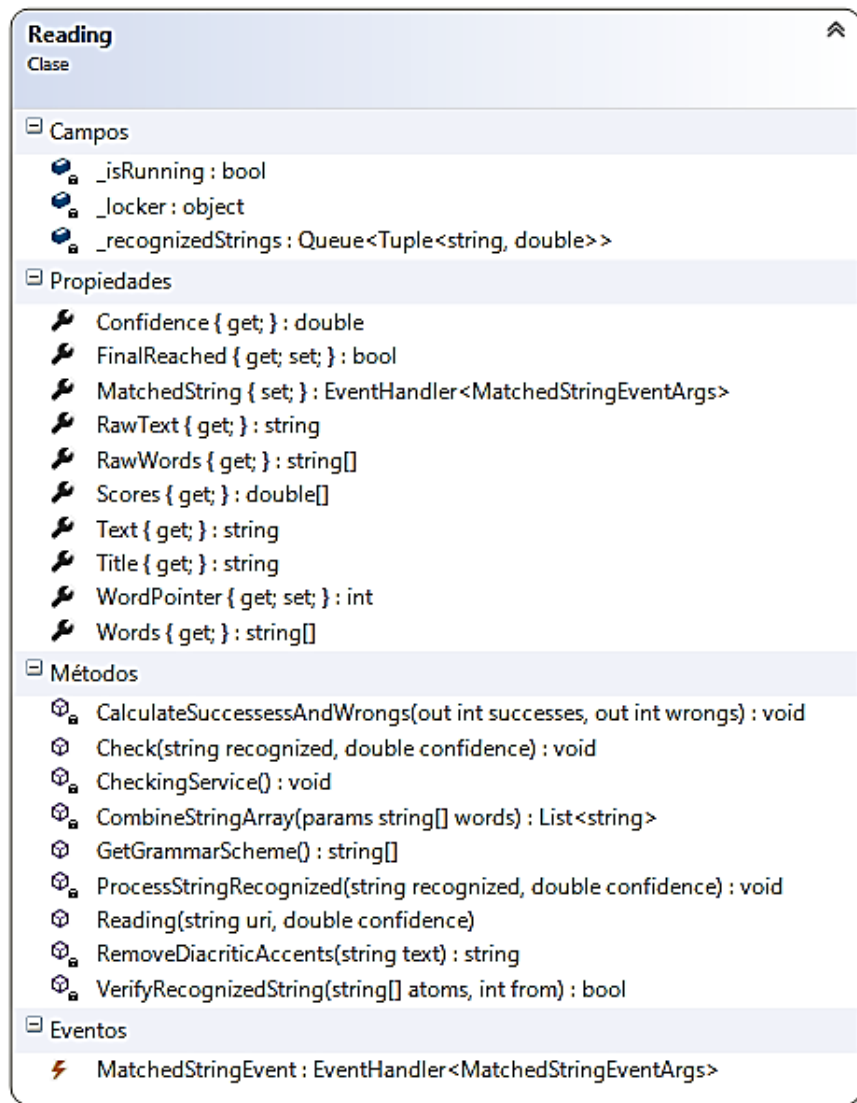


Figura 25. Especificación de clase UML de Reading.

El evento `MatchedStringEvent` se desencadena cuando una palabra o frase es reconocida con el nivel de `Confidence` (confiabilidad) mayor o igual a la establecida.

El sistema de reconocimiento posee una cola de espera para que un hilo encole los reconocimientos desde `Speech Recognition Engine`, y luego otro hilo, paralelamente, desencole y analice el reconocimiento con el texto de la lectura, verificando en qué parte de ésta encaja. Para esto, se maneja un índice que apunta a la última palabra reconocida.

La técnica algorítmica anterior, soluciona en parte un problema repetitivo en Microsoft Speech, el cual produce un retraso considerable en el reconocimiento de palabras y frases, lo cual implica el fracaso en el reconocimiento de frases de gran longitud, como las lecturas.

También, se maneja una colección donde se almacenan los puntajes de confiabilidad por palabra reconocida, para luego calcular la modulación a través del promedio de estos, y proyectar sus valores (ya que retorna decimales entre 0 y 1) hacia una escala de 1.0 a 7.0.

La clase Reading carga la lectura desde el fichero con un formato predefinido (ver Figura 15, p. 48) y realiza una normalización de las palabras en él, para generar un esquema adecuado y entregárselo a VoiceRecognizer. Esta normalización consiste en la eliminación de puntos, comas, u otros signos gramaticales de manera de añadir cada elemento y sus combinaciones a VoiceRecognizer por separado.

Respecto a las combinaciones realizadas por el algoritmo, éstas son parciales y sólo se ejecuta la concatenación de cadenas con las cadenas siguientes iterativa e incrementalmente (en forma piramidal).

Por ejemplo, si tenemos una cadena $S = A + B + C$, donde A, B, C son átomos (en general palabras) de S, entonces sus combinaciones serían: $\{\{A\}; \{A, B\}; \{A, B, C\}; \{B\}; \{B, C\}; \{C\}\}$. Pero se habría omitido $\{A, C\}$. Ésta omisión puede ser pequeña para una cadena de 3 palabras, pero para cadenas de mayor longitud estas omisiones aumentan considerablemente.

El problema de las combinaciones se trató de resolver con un algoritmo de solución completa, es decir, que calculaba todas las combinaciones (el método CombineStringArray todavía presente en la clase). El problema con esto es que utiliza mucha memoria principal, por sobre los 6 GB para textos de 30 palabras, en contraposición con los 200 MB del algoritmo anterior.

Finalmente, se optó por dejar el algoritmo liviano más en uso de memoria, debido a que el algoritmo completo se hacía inviable, aun así, el reconocimiento se realiza de una

manera adecuada, con la mayoría de errores producidos por el ruido del audio, y no por errores en la definición de gramática

6.2.9. Módulo de reconocimiento de gestos.

Es el encargado de procesar la información de coordenadas tridimensionales que captura la librería de Skeleton, luego de procesar el flujo de vídeo, con el objetivo de mover el cursor por pantalla guiado por una mano, apuntando, y realizar gestos de deslizar con la otra mano para bajar y subir por listas.

El módulo se conforma por 6 clases.

6.3.8.1. Pair.

Esta clase representa una pareja de términos que se corresponden y puede verificar si un conjunto de cadenas son un el par correcto correspondiente (ver Figura 26).

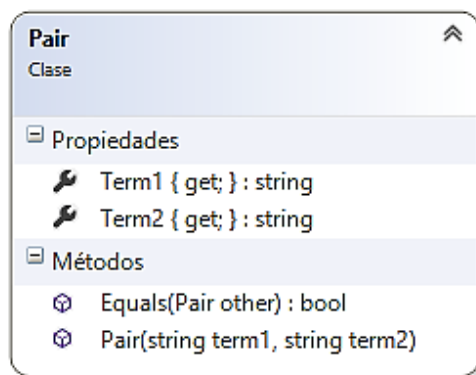


Figura 26. Especificación de clase UML de Pair.

6.3.8.2. PairingTerms.

Representa la actividad de términos pareados. Básicamente es una lista de objetos Pair, que posee métodos de mezcla de términos (Shuffle) y verificación de uniones de términos (ver Figura 27).

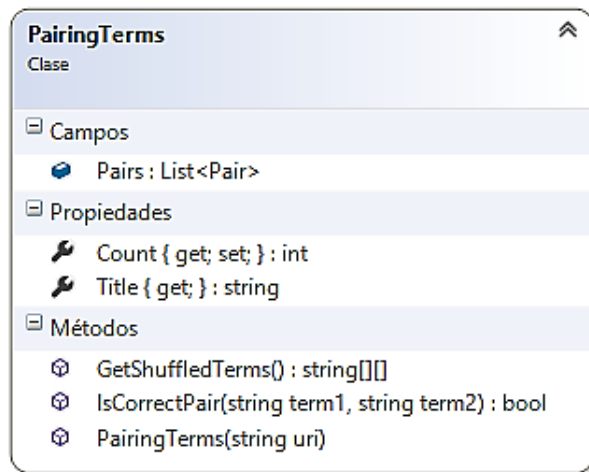


Figura 27. Especificación de clase UML de PairingTerms.

La mezcla de términos de utiliza para entregar a la capa visual una lista desordenada de términos para que esta la muestre.

6.3.8.3. MouseEmulator.

Clase que permite gatillar programáticamente eventos del ratón. Movimiento del cursor, clic principal, intermedio o secundario y el uso de la rueda, son posibles de emular. Además, es posible obtener la posición actual absoluta del cursor en la pantalla (ver Figura 28).

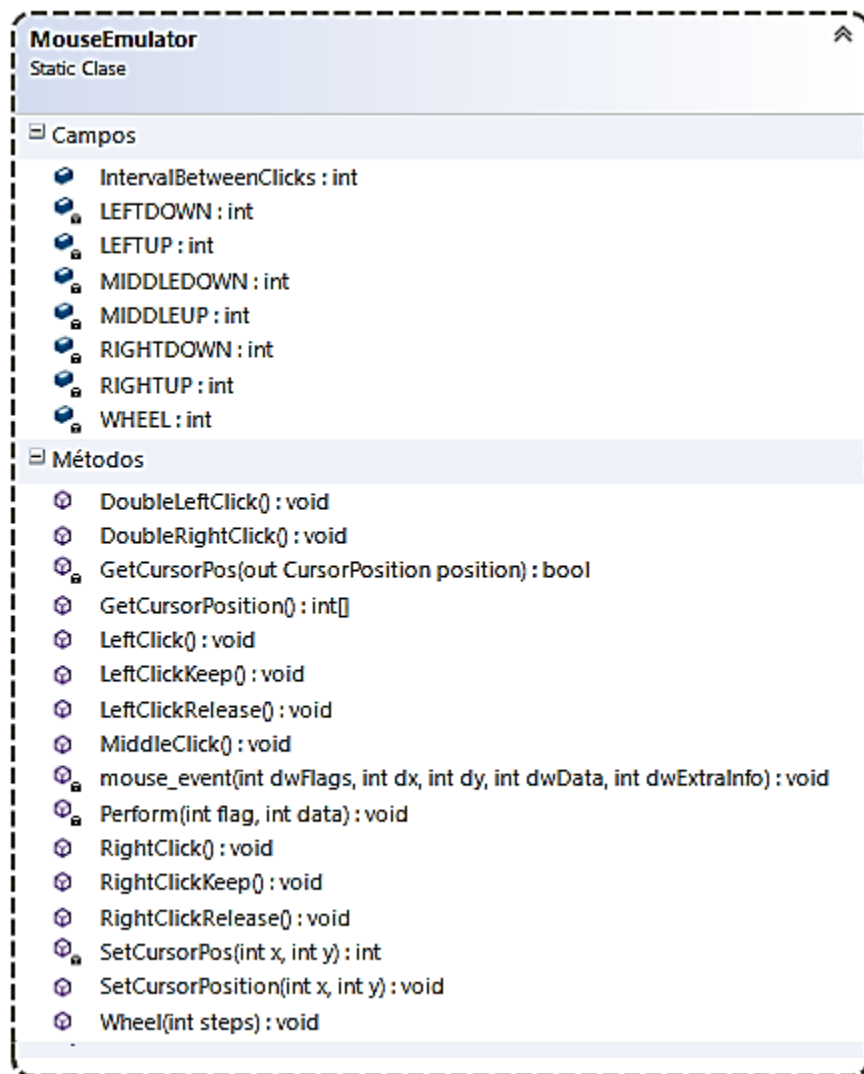


Figura 28. Especificación de clase UML de MouseEmulator.

El control de las acciones del ratón es posible gracias a la utilización de una librería de bajo nivel del sistema operativo, llamada User32.dll.

6.3.8.4. SkeletonTracker.

Gestiona la información recibida por el objeto Skeleton de la librería nativa de Kinect, realizando cálculos cartesianos que permiten obtener distancias y ángulos de los puntos identificados. La información entregada por esta clase es esencial para realizar un reconocimiento de gestos (ver Figura 29).

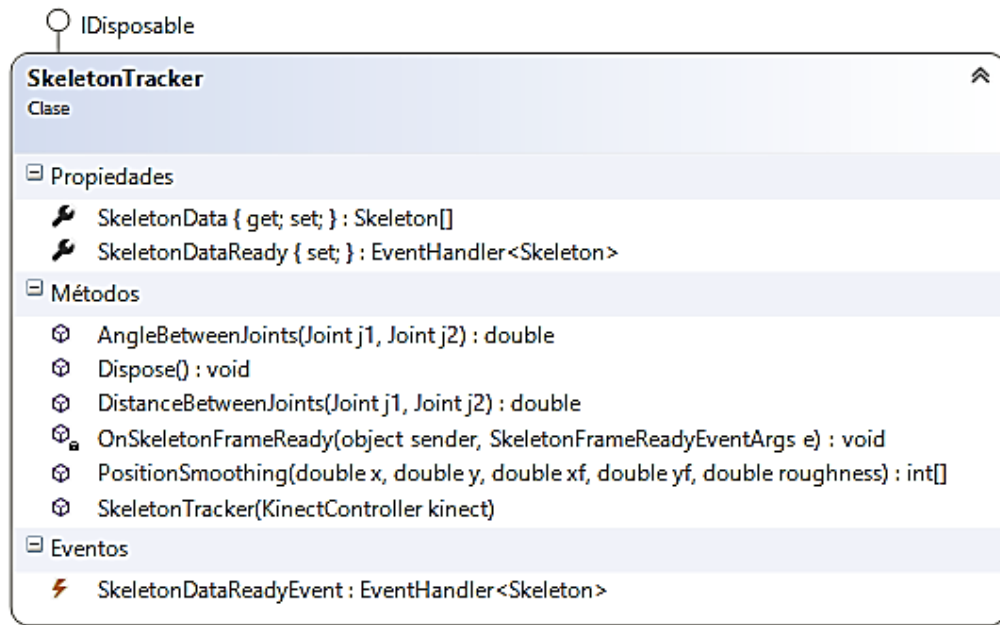


Figura 29. Especificación de clase UML de SkeletonTracker.

Gestiona el desencadenamiento del evento SkeletonDataReady, que indica que un joint del skeleton ha sido reconocido (y sus coordenadas están disponibles).

6.3.8.5. JointCache.

Clase que gestiona una lista de joints, y realiza cálculos de promedio y distancia para facilitar el reconocimiento de gestos en intervalos determinados de tiempo (ver Figura 30).

La cache funciona como una cola de tamaño limitado, el cual está determinado en las configuraciones de la aplicación (KinectCapabilites), en otras palabras, funciona como un historial de posiciones de un joint, de manera de calcular movimientos en el espacio tridimensional y reconocer gestos.

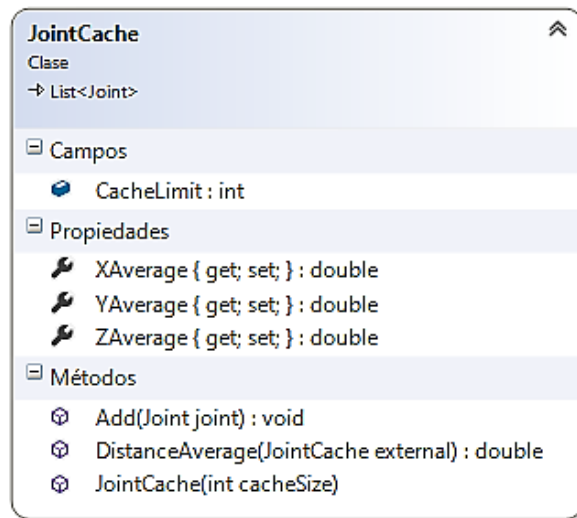


Figura 30. Especificación de clase UML de JointCache.

6.3.8.6. HandControl.

La clase más importante del reconocimiento de gestos. Maneja los cálculos principales para verificar si algún joint ha realizado algún gesto en particular (ver Figura 31).

Se encarga de realizar la verificación de los siguientes joints: manos, codos y hombros, derechos e izquierdos.

La clase posee un modo para zurdos, esto quiere decir que cambia automáticamente la mano principal, y todos los demás joints (por reflexión) si se activa.

Los gestos que han sido programados son:

- Push (empujar): movimiento de profundidad de atrás hacia adelante (eje Z).
- Pull (tirar): movimiento de profundidad de adelante hacia atrás (eje Z).
- Swipe Up (deslizar hacia arriba): movimiento vertical de abajo hacia arriba (eje Y).
- Swipe Down (deslizar hacia abajo): movimiento vertical de arriba hacia abajo (eje Y).
- Swipe Left (deslizar hacia la izquierda): movimiento horizontal de derecha a izquierda (eje X).

- Swipe Right (deslizar hacia la derecha): movimiento horizontal de izquierda a derecha (eje X).
- Closeness (cercanía): cercanía de 2 joints.

No obstante, las capacidades de esta clase van más allá. Se implementaron funciones que permiten conocer el ángulo y distancia entre 2 o más joints, lo que permitiría reconocer posiciones y posturas específicas. En el prototipo no se implementó este reconocimiento ya que no se ajustaba a los objetivos principales, pero las funciones permanecen.

HandControl implementa un sistema de estabilización de movimientos y de gestos, con el objetivo de mejorar la precisión de estos, como por ejemplo en el control del cursor, o en el apuntar hacia un determinado objeto. Este sistema de estabilización se basa en un tiempo de espera luego del reconocimiento de un gesto, tiempo en el cual se limita el movimiento.

Además, posee un sistema de resolución de conflicto en gestos opuestos, es decir, gestos que puedan generar problemas debido a su naturaleza y la continuidad de su ejecución. Un ejemplo de estos gestos es el Swipe Left y Swipe Right, cuando una persona realiza un gesto a la izquierda, luego, de manera natural, realiza un gesto a la derecha para volver a su posición inicial, pero ese último gesto no es necesariamente un gesto intencional para indicar una acción, por lo tanto, se deben establecer reglas de comportamiento para el reconocimiento correcto de estos gestos opuestos.

Para evitar estos conflictos, se utiliza un bloqueo selectivo de los gestos opuestos, por un determinado tiempo luego de que un gesto haya sido reconocido, gestionando un control de tiempo y de repetición de estos.

Internamente la clase maneja un Enum con los valores asociados a cada tipo de movimiento, que funcionan como flags para indicar los bloqueos y los estados del cronómetro de bloqueo para cada gesto.

El cálculo de posiciones, distancias y ángulos entre partes del cuerpo utilizan un porcentaje considerable de CPU, por lo que es necesario encontrar el equilibrio y realizar sólo los cálculos necesarios por frame, para evitar retrasos de procesamiento.

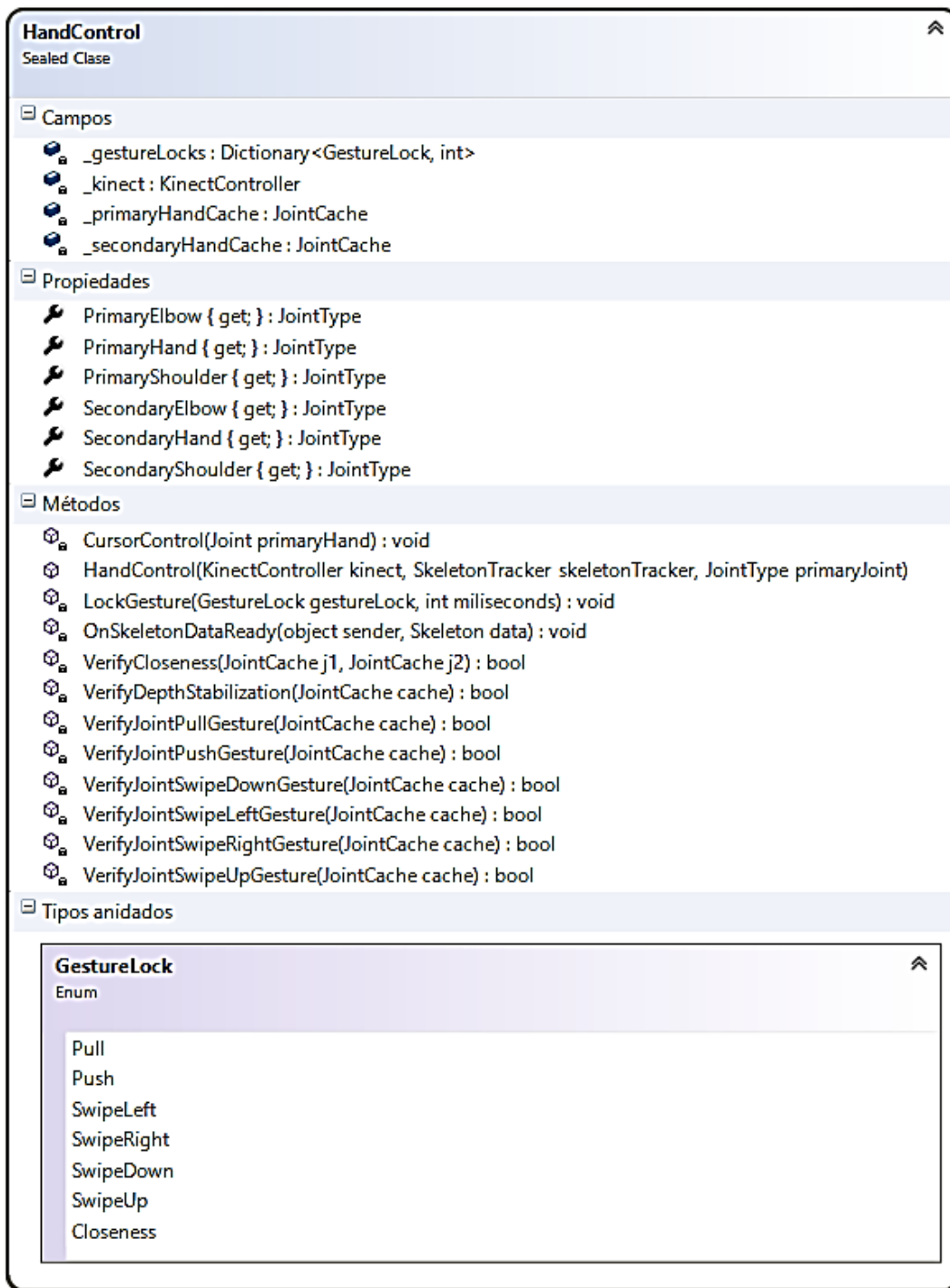


Figura 31. Especificación de clase UML de HandControl.

6.4. Flujo de navegación.

Kleng iniciará con la visualización del logo del proyecto (ver Figura 32) para posteriormente ingresar al menú principal, que permitirá seleccionar las actividades a realizar: lectura o términos pareados.



Figura 32. Logo de Kleng.

La selección de la opción de lectura implicará la transición desde el menú principal hasta el menú de lectura, el cual permitirá seleccionar entre cargar, iniciar o finalizar la lectura.

La selección de la opción de términos pareados implicará la transición desde el menú principal hasta el menú de términos pareados, el cual permitirá seleccionar entre cargar, iniciar o finalizar los términos pareados.

La transición desde el menú principal hacia el menú de lectura como también hacia el menú de términos pareados, se resume en el siguiente diagrama (ver Figura 33):

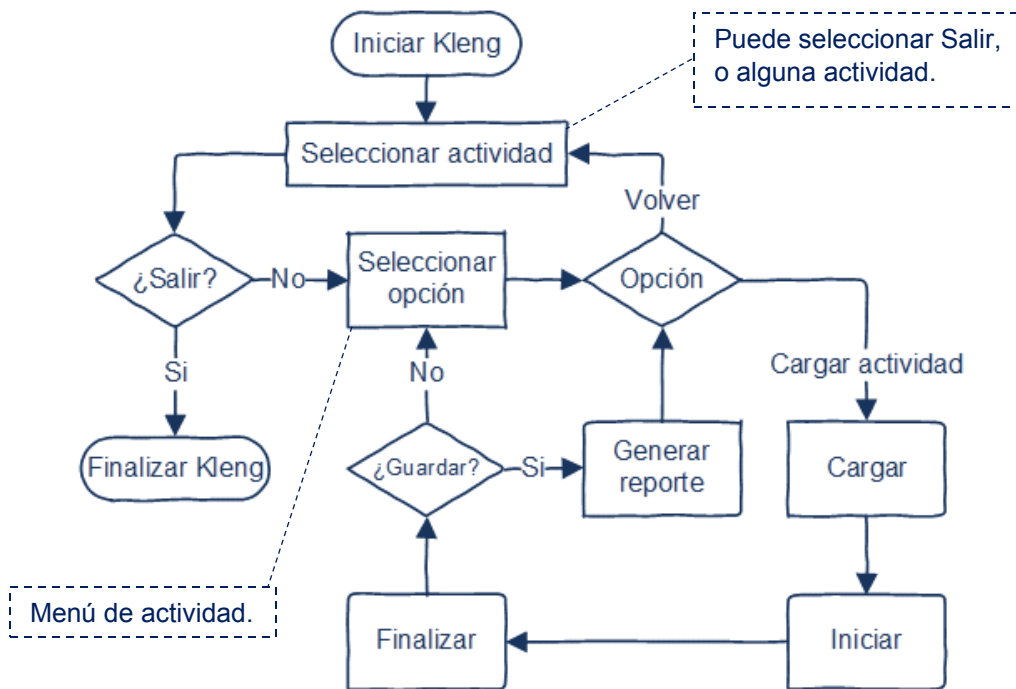


Figura 33. Diagrama de flujo de navegación por Kleng.

Seleccionar actividad, en el menú principal, permite entrar al menú de Lectura, Términos Pareados o salir de la aplicación.

Seleccionar opción, permite al usuario en el menú respectivo de la actividad, seleccionar alguna de las 3 opciones genéricas: Iniciar, Cargar o Finalizar.

Iniciar desencadena el inicio de la actividad respectiva.

Cargar despliega una interfaz de navegación por directorios para seleccionar el archivo de la actividad a cargar (ver sección 5.5.1, p. 47). Si el usuario cancela la carga, vuelve al menú anterior de la actividad.

Finalizar, termina la actividad en curso y despliega los resultados respectivos y activa la opción Guardar para generar el reporte de los resultados en un documento. Puede gatillarse manualmente, o de forma automática si Kleng comprueba que la actividad ha sido completada.

6.5. Interfaz gráfica de usuario.

La implementación final de los módulos y de la GUI, permitió incrementar la funcionalidad y mejorar la apariencia prevista en el capítulo de diseño (ver capítulo 5, p. 38).

Los módulos se programaron guiados por los estándares más estrictos de C# proporcionados por Microsoft, además, su alta modularidad permitirá su posterior utilización de manera muy simple.

La apariencia seleccionada se basa en Material Design de Google, gracias al conjunto de estilos "Material Design in XAML". Las animaciones y control de elementos de la GUI se programaron en un espacio de nombres de utilidades gráficas, con animaciones de todo tipo (las principales son Blink, Rotate y Zoom), para entregar mayor calidad visual.

En cada una de las interfaces se destaca la activación de los botones sólo cuando su función sea útil, con el objetivo de evitar errores por parte del usuario y omitir ese tiempo de mostrar un mensaje de advertencia al presionar algo innecesario.

La especificación de los cuadros de mensaje se realiza en el Anexo G.

El menú de inicio de Kleng se puede observar a continuación (ver Figura 34):

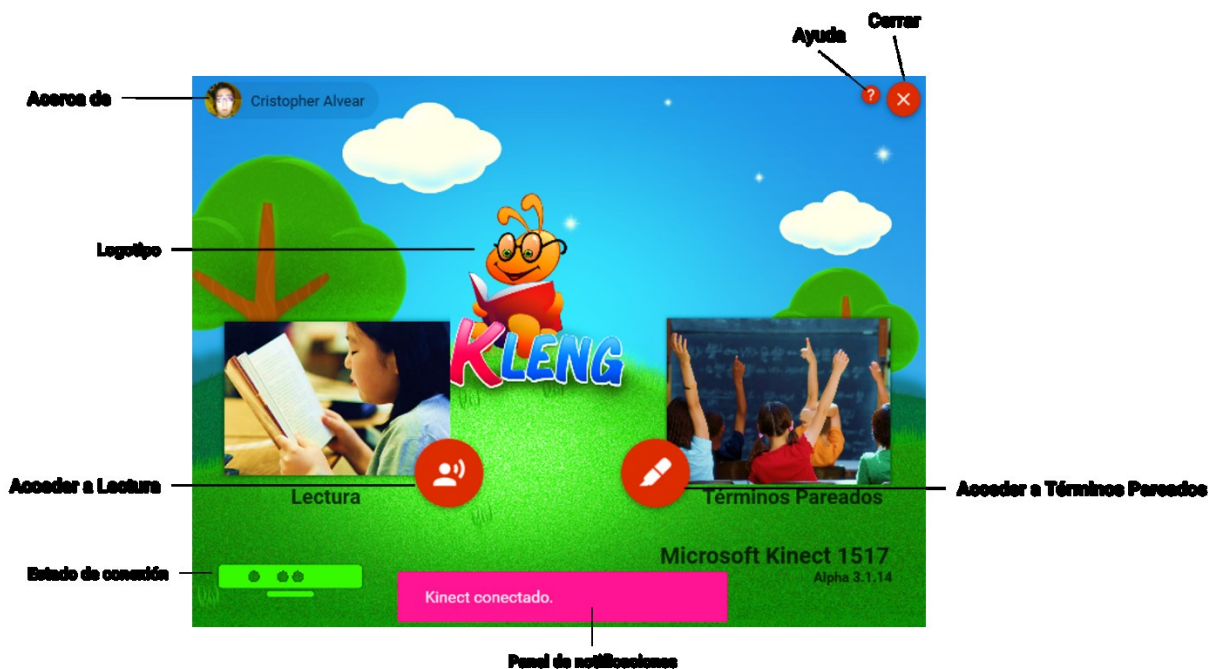


Figura 34. GUI menú de inicio Kleng.

De estos elementos, se destaca el botón Ayuda y el Panel de notificaciones. El primero, permite acceder al modo de ayuda, en el cual se reproduce una explicación auditiva si se realiza un clic en algún botón que el usuario desee conocer. El panel, es dinámico y sólo aparece por algunos segundos cuando existe un mensaje importante respecto al estado de Kinect.

El estado de Kinect se indica con los siguientes 3 colores del ícono en la zona inferior izquierda (ver Figura 35):

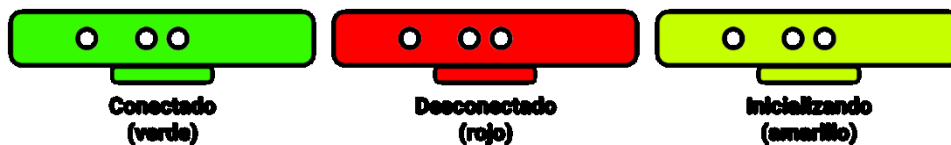


Figura 35. GUI ícono estado de Kinect.

6.5.1. GUI de lectura.

El menú de la lectura se puede observar a continuación (ver Figura 36):



Figura 36. GUI menú de lectura Kleng.

El control de Grabar audio permite activar la grabación del audio de la lectura realizada por el estudiante, junto al reporte. Si no se realiza el guardado del reporte al final de la lectura, el audio tampoco será guardado.

El ícono de idioma, muestra el idioma de reconocimiento de voz actual, que depende de la librería Microsoft Speech instalada en el sistema. La variedad de idiomas soportados por Kleng se muestra en la siguiente imagen (ver Figura 37):



Figura 37. GUI ícono de idioma Kleng.

Las palabras incorrectas indican la cantidad palabras erróneas u omitidas que el estudiante ha dicho. Las palabras erróneas no necesariamente pertenecen a la lectura.

Al cargar la lectura y, mientras está en curso, se muestra una ventana de visualización del texto (ver Figura 38):

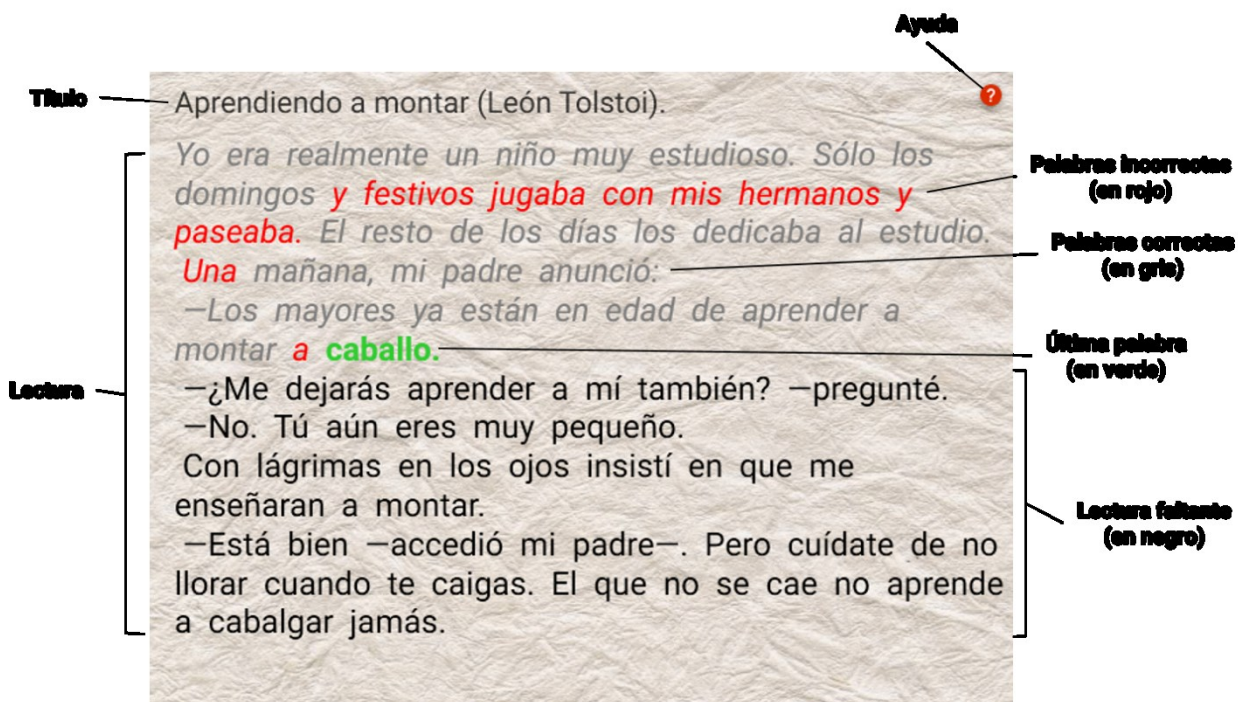


Figura 38. GUI visualización de lectura Kleng.

En la zona superior se muestra el título de la lectura.

El texto ya leído, de forma correcta, se muestra en color gris claro, mientras que las palabras leídas incorrectamente u omitidas, se muestran en rojo. El indicador de posición (última palabra leída), se muestra en verde, y el texto que falta leer se muestra en negro.

6.5.2. GUI de términos pareados.

El menú de términos pareados se muestra a continuación (ver Figura 39):



Figura 39. GUI menú de términos pareados Kleng.

Similar al menú de lectura, donde se destaca el control de Modo zurdo, que permite alternar la mano principal para personas zurdas, trasladando todos los joints asociados de forma reflexiva.

La visualización de los términos pareados se puede observar en la siguiente figura (ver Figura 40):

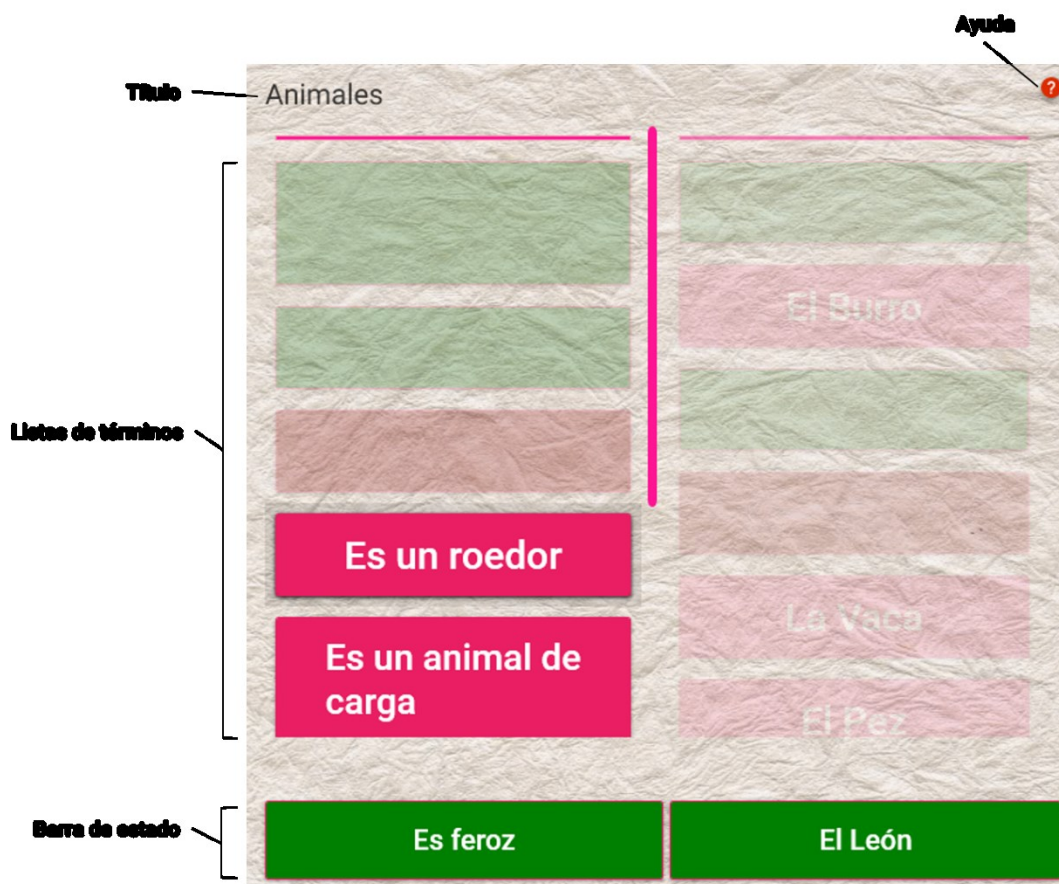


Figura 40. GUI visualización de términos pareados Kleng.

En la zona superior se muestra el título de la actividad de términos pareados.

La zona intermedia, correspondiente a las listas de términos, donde estos se listan en 2 columnas de forma aleatoria. La columna izquierda comienza activa, mientras que la segunda desactivada. A medida que se realizan clics en algún término de alguna columna, esta se desactiva y se activa la otra columna.

La zona inferior, llamada barra de estado, permite al estudiante saber cuál es el término que ha seleccionado o cuál es la última pareja que unió, indicándole en verde o rojo si ésta última ha sido correcta o incorrecta. Además, puede pulsarla si el estudiante ha seleccionado un término y desea cancelarlo.

Los diversos estados de la barra de estado se indican a continuación (ver Figura 41):



Figura 41. GUI estados de la barra inferior de términos pareados.

6.5.3. GUI guardar reporte.

El menú de guardar reporte se observa en la figura a continuación (ver Figura 42):

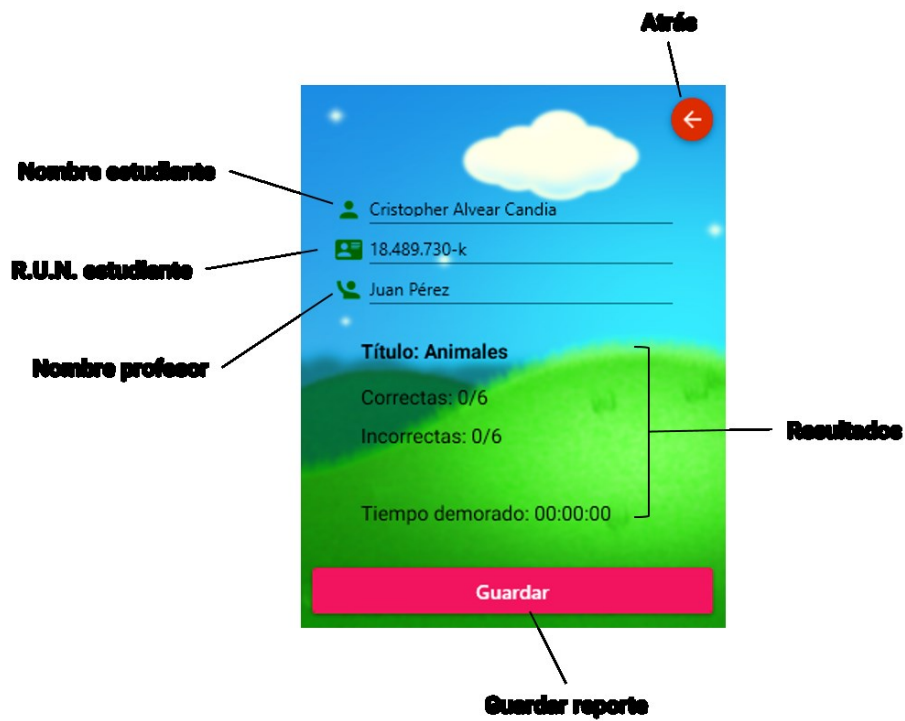


Figura 42. GUI guardar reporte Kleng.

6.6. Características del prototipo.

En la Tabla 19 se muestran las cualidades finales del prototipo.

Nombre	Kleng (Kinect para lenguaje).
Versión	Alpha 3.1 revisión 14 (2016-12-08)
Tipo de aplicación	Prototipo educativo interactivo
Características	<p>Evaluación de lecturas:</p> <ul style="list-style-type: none"> ✓ Reconocimiento de palabras y/o frases progresivamente. ✓ Cronómetro. ✓ Evaluación de la modulación, en escala de 1.0 a 7.0. ✓ Contador de palabras correctas e incorrectas. ✓ Énfasis en tiempo real de la última palabra reconocida. ✓ Énfasis en tiempo real de palabras correctas e incorrectas. ✓ Soporte de idioma español (España y México) e inglés. ✓ Grabación de audio en calidad PCM, formato WAV. ✓ Generación de reportes de los resultados en PDF. ✓ Carga de lecturas desde ficheros TXT. ✓ Lecturas de hasta 100 palabras. ✓ Lecturas con signos gramaticales ¿?!;:()- ✓ Finalización automática de la lectura. <p>Evaluación de términos pareados:</p> <ul style="list-style-type: none"> ✓ Control del cursor mediante la posición de la mano primaria. ✓ Reconocimiento de gestos push (selección) de mano primaria y, swipe up (subir lista) y swipe down (bajar lista) de mano secundaria. ✓ Estabilización y suavizado del movimiento del cursor. ✓ Estabilización del cursor en la ejecución de gestos. ✓ Control selectivo automático y bloqueo temporal de gestos conflictivos (como swipe up y swipe down). ✓ Modo zurdo. ✓ Contador de términos correctos e incorrectos. ✓ Generación de reportes de los resultados en PDF. ✓ Carga de términos desde archivos TXT. ✓ Términos de hasta 10 palabras o 60 caracteres. ✓ Redimensionado automático de términos según su longitud. ✓ Barra de estado indicando pares seleccionados, correctos e incorrectos. ✓ Revertir selección de un término presionando la barra de estado. ✓ Finalización automática de actividad. <p>Interfaz gráfica y calidad de uso:</p> <ul style="list-style-type: none"> ✓ Material Design de Google. ✓ Cursores de Cut the Rope. ✓ Variadas animaciones (Blink, Rotate y Zoom). ✓ Modo ayuda con instrucciones auditivas en español. ✓ Paleta de colores Material Design: Rosado, Naranja y Verde. ✓ Activación y desactivación inteligente de controles.

Tabla 19. Características Kleng Alpha 3.1.12.

7. Pruebas.

Este capítulo presenta las pruebas de software realizadas a Kleng, las que tienen como objetivo comprobar la correcta implementación de los módulos y las funcionalidades descritas en la etapa de Análisis (ver capítulo 4, p. 22) y Diseño (ver capítulo 5, p. 38).

Las pruebas de sistema se hicieron de forma manual, mientras que las pruebas de mantenibilidad mediante herramientas métricas de Visual Studio.

7.1. Especificación de las pruebas.

Primero se definirán las pruebas de sistema funcionales y, posteriormente, pruebas de calidad de código a través de herramientas de Visual Studio. No se hará mayor referencia a las pruebas unitarias ni de integración, ya que se realizaron a lo largo del desarrollo, sin especificación.

La especificación completa de las pruebas de sistema funcionales, se pueden consultar en el Anexo H.

7.1.1. Pruebas funcionales.

En esta sección se presenta la definición de las pruebas de sistema funcionales relevantes (ver Tabla 20). Estas pruebas no constan de la medición de la calidad de reconocimiento de voz, ni la calidad de reconocimiento de gestos.

ID:	KLST01
Nombre:	Realizar lectura
Objetivo:	Probar la realización de la lectura.
ID:	KLST02
Nombre:	Realizar términos pareados
Objetivo:	Probar la realización de términos pareados.
ID:	KLST03
Nombre:	Desconectar Kinect en menú actividad
Objetivo:	Probar el comportamiento del sistema al desconectar el sensor cuando se realiza una actividad, ya sea lectura o términos pareados.
ID:	KLST04
Nombre:	Iniciar actividad con Kinect desconectado
Objetivo:	Probar el comportamiento del sistema iniciar una actividad, ya sea lectura o términos pareados con Kinect desconectado.
ID:	KLST05
Nombre:	Ingresar R.U.N. con formato válido

Objetivo:	Probar el comportamiento del sistema al ingresar R.U.N. válidos en la vista de guardar reporte de actividad.
ID:	KLST06
Nombre:	Ingresar R.U.N. con formato inválido
Objetivo:	Probar el comportamiento del sistema al ingresar R.U.N. inválidos en la vista de guardar reporte de actividad.
ID:	KLST07
Nombre:	No completar nombres del formulario guardar reporte
Objetivo:	Probar el comportamiento del sistema al guardar el reporte sin completar (dejando vacío) el nombre del estudiante o profesor en el formulario.
ID:	KLST08
Nombre:	Cargar actividad inválida
Objetivo:	Probar el comportamiento del sistema al cargar una actividad inválida desde un archivo de texto, en la lectura o términos pareados.
ID:	KLST09
Nombre:	Modo zurdo
Objetivo:	Probar el modo zurdo para los términos pareados.

Tabla 20. Resumen definición de pruebas funcionales.

7.2. Resultados de las pruebas.

En esta sección, se muestran los resultados de las pruebas formales realizadas a Kleng.

7.2.1. Pruebas funcionales.

A continuación, se muestra una tabla resumen con los resultados de las pruebas de sistema funcionales realizadas el 29 de noviembre del 2016 (ver Tabla 22).

ID	Fallas	Estado	Observaciones
KSLT01	0	Ok ✓	
KSLT02	0	Ok ✓	
KSLT03	1	Ok ✓	El evento ChangedStatus no se lanzaba adecuadamente.
KSLT04	0	Ok ✓	
KSLT05	0	Ok ✓	
KSLT06	0	Ok ✓	
KSLT07	0	Ok ✓	
KSLT08	0	Ok ✓	
KSLT09	1	Ok ✓	Generó problemas con la lógica del cambio de joints.

Figura 43. Resultados generales pruebas de sistema.

7.2.2. Índices de mantenibilidad.

En esta sección se mostrarán y explicarán los índices de calidad de código más importantes proporcionados por las herramientas métricas de Visual Studio [21].

Kleng se compone de 34 clases: 21 de ellas de la capa de control y 13 de la capa de presentación.

El proyecto tiene 2.588 líneas de código (sin contar las librerías externas).

- a. Índice de mantenimiento: (83). Valor entre 0 y 100 que representa la facilidad relativa de mantenimiento del código. Un valor alto significa mayor facilidad de mantenimiento. En Kleng, 83 es un índice alto, que indica que el código tiene una mantenibilidad muy buena, a pesar de poseer clases de alta complejidad. La capa de control, marca un índice 88, una mantenibilidad excelente. La capa de presentación cae a un 71, debido a que las GUI siempre poseen mayor código boilerplate¹⁰.
- b. Complejidad ciclomática: (511). Complejidad estructural del código. Se calcula según el número de rutas de acceso del código diferentes del flujo del programa. La complejidad de Kleng total es alta, no obstante, no posee métodos que sobrepasen una complejidad de 20, y como se observa en la Tabla 21, esto es bastante bueno.

Complejidad ciclomática	Descripción
1-10	Métodos sencillos, sin mucho riesgo
11-20	Métodos medianamente complejos, con riesgo moderado
21-50	Métodos complejos, con alto riesgo
51 o más	Métodos inestables, de altísimo riesgo

Tabla 21. Clasificación de complejidad ciclomática.

7.2.3. Prueba preliminar.

¹⁰ Código que es repetitivo y declarativo. No tiene una lógica compleja y suele ser copiado y pegado.

Durante la versión Alpha 2 del prototipo (ver Figura 44), se realizó una prueba de aceptación en el colegio Arturo Mutizábal de Chillán, con 9 estudiantes de 1° básico, el 17 de agosto del 2016.

En esta etapa, la inmadura implementación del reconocimiento de la lectura, hizo difícil su uso. Se identificaron muchos problemas en el reconocimiento: palabras imposibles de reconocer, signos gramaticales conflictivos y lentitud en el procesamiento. Estos problemas se corrigieron en la versión Alpha 3 de Kleng, reestructurando la clase que manejaba el procesamiento de la lectura (Reading). Además, sirvió para guiar el desarrollo hacia una interfaz de mayor calidad, que fuera más atractiva para los niños.

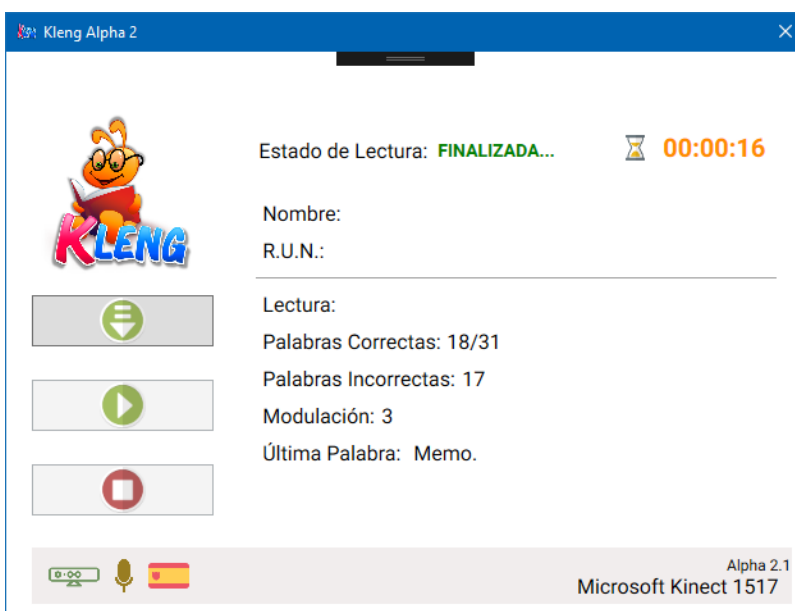


Figura 44. Kleng Alpha 2.1.

7.2.4. Prueba de aceptación.

Al final del proyecto, luego de las pruebas anteriores y sus correcciones pertinentes, se realizó la prueba final de aceptación.

Esta prueba tuvo lugar el día 30 de noviembre del 2016, en el Colegio Sagrado Corazón de Jesús (COSACO) de San Carlos. La profesional a cargo de la organización fue la profesora y jefe del Departamento de Básica, Sandra Navarrete Méndez, y la supervisora de los estudiantes Camila Eliza Canto, asistente de sala.

En total, fueron 14 estudiantes de 4° básico los que probaron la aplicación y dieron su opinión (ver Tabla 22).

Matías Vásquez
Maximiliano Montoya
Patricio Bustos
Javiera Sánchez
Génesis Salvo
Emilia Quezada
Diego Valenzuela
Fernando Toro
Leonardo Ortega
Juan Pablo Zapata
Sofía Barros
Rocío Cerda
Nicole Zúñiga
Martina López

Tabla 22. Lista de estudiantes que probaron Kleng.

Se probaron las 2 actividades de Kleng junto a las opciones Guardar audio y Modo zurdo.

Cada estudiante realizó una actividad de lectura (ver Figura 45) y una de términos pareados (ver Figura 46), aunque algunos querían jugar nuevamente con los términos pareados, pero el tiempo no lo permitió.



Figura 45. Estudiante realizando una lectura en Kleng.



Figura 46. Estudiante realizando términos pareados en Kleng.

Las pruebas se realizaron satisfactoriamente, la mayoría de los niños lo disfrutó y les gustó la experiencia.

Se realizó una pequeña encuesta a los niños, para conocer su apreciación general del prototipo. La encuesta se muestra a continuación (ver Figura 47).

<u>Prueba de aceptación Kleng</u>	
1. ¿Kleng es entretenido?	
Si <input type="checkbox"/>	No <input type="checkbox"/>
2. ¿Qué actividad te gustó más?	
Lectura	<input type="checkbox"/>
Términos pareados	<input type="checkbox"/>
Ambas	<input type="checkbox"/>
Ninguna	<input type="checkbox"/>

Figura 47. Encuesta de aceptación Kleng.

Los resultados fueron los siguientes:

- Todos los niños respondieron que les gustó Kleng y se entretuvieron con él.
- 2 niños indicaron que les gustó la actividad de términos pareados, mientras que el resto respondió que ambas les gustaron.

Finalmente, se identificaron los siguientes problemas:

- Reconocimiento erróneo en lecturas con voz muy baja.
- Algunos malos reconocimientos implicaban un salto imprevisto e incorrecto en el avance de la lectura.
- Márgenes máximos inadecuados para el movimiento de cursor con la mano.
- Dificultad para algunos niños en realizar correctamente los gestos.
- Modo zurdo no funciona.
- Luego de 1 hora aproximadamente, el sobrecalentamiento del sensor Kinect afecta notablemente el rendimiento en Kleng, respecto al reconocimiento de voz y gestos.

El funcionamiento del modo zurdo y los márgenes fueron solucionados, mientras que se establecieron sensibilidades (en KinectCapabilities) más altas para los gestos, de manera de facilitar su uso para los niños, aunque esto último no fue probado con ellos.

El sobrecalentamiento de Kinect es una variable incontrolable, puede que dependa de la complejidad del código, pero optimizar el código sobrepasaría los esfuerzos límites del proyecto.

El resto de los problemas dependen de la correcta acústica de la sala y el volumen de la voz, por lo tanto, dentro del proyecto, estos son difícilmente abordables.

7.3. Análisis de los resultados.

La prueba de aceptación permitió encontrar fallas en el prototipo. El uso intensivo hace que los sensores de Kinect funcionen mal. El ruido ambiental y el volumen bajo de la voz del estudiante hace difícil la tarea de reconocimiento de la lectura, esto es inevitable.

La aceptación de Kleng fue bastante buena, como siempre sucede en estas situaciones, había niños muy motivados y otros no tanto, pero ninguno tuvo una reacción negativa, todo lo contrario, todos querían participar y probar el prototipo.

A algunos estudiantes les fue difícil adaptarse a la mecánica del control de movimiento de los términos pareados, mientras que otros lo hicieron perfectamente. Este también fue un problema al principio, ya que se centraban más en el aprendizaje de cómo mover el cursor y realizar los gestos que en los términos que seleccionaban.

En conclusión, Kleng puede llegar a ser muy entretenido de usar por los niños durante un rato, permitiendo jugar con actividades que comúnmente en las clases de Lenguaje se hacen con lápiz y papel, lo cual se torna aburrido.

8. Síntesis y conclusiones.

Los métodos de enseñanza en todo el mundo, están experimentando un proceso de cambio, un cambio gradual que no sólo consta de la incorporación de tecnologías de la información y la comunicación en el aula, sino que también un cambio de mentalidad en los educadores, que les permita utilizar estas tecnologías de apoyo educacional adecuadamente.

Chile está atrasado en la innovación educacional, la utilización de computadores no garantiza en ninguna medida esta incorporación de tecnologías y el cambio de mentalidad, que mejoren el proceso de enseñanza-aprendizaje. Es el momento de realizar los esfuerzos para innovar y motivar a los desarrolladores a incursionar en el mundo de la educación interactiva en Chile.

Las aplicaciones con NUI están ampliando las formas de uso de los computadores. Hace poco más de 1 década, pocos creían posible controlar los sistemas con la voz o con gestos, sin utilizar algún mando físico.

Kinect para Microsoft Windows es uno dispositivo que permite la implementación de aplicaciones con NUI, en especial el reconocimiento de voz y gestos.

Kleng es un prototipo de sistema interactivo, orientado a niños desde 1° a 4° básico que cursen la asignatura de Lenguaje, que demuestra las capacidades del dispositivo Kinect y sus tecnologías de desarrollo asociadas, aplicadas a la educación, y que pretende motivar el uso de estas tecnologías en el área.

La investigación del SDK de Kinect fue una tarea recurrente a lo largo del proyecto, debido a que las tecnologías NUI usadas en Kleng, generan diversos problemas como consecuencia de la naturaleza dinámica e impredecible de los datos de entrada, por lo que se estudió exhaustivamente la manera de evitar errores por esto. El reconocimiento de palabras, aún utilizando librerías diseñadas por Microsoft, y el reconocimiento de gestos fueron los principales desafíos del proyecto, los cuales fueron completados con éxito.

El reconocimiento de voz, aplicado a la evaluación de lecturas, es complejo y aleatorio, debido a múltiples factores: el tono y rapidez de la voz, el ruido, calidad de micrófonos y acústica del ambiente son algunas de las variables que inciden en un reconocimiento exitoso. Además, el análisis de la onda sonora en tiempo real es un proceso costoso computacionalmente, por lo que los retardos en los reconocimientos, en especial de frases y textos, son difíciles de evitar.

La librería Microsoft Speech simplifica el proceso de reconocer palabras o frases, y su optimización para Kinect asegura un buen desempeño en estos dispositivos, pero no soluciona bien el problema de la lentitud. En consecuencia, el reconocimiento de lecturas se hizo muy complicado, especialmente al inicio, ya que al utilizar la librería Microsoft Speech sin una infraestructura algorítmica de apoyo, los resultados eran mediocres. Incluso, en la Alpha 2 del prototipo existían muchos problemas en la realización de la lectura, principalmente errores y lentitud en la identificación de algunas palabras y frases.

El uso de la programación multihilo permitió mejorar el tiempo de respuesta en el reconocimiento de frases, e implementar en tiempo real un reconocimiento progresivo de un texto predefinido, con muy pocos errores sin importar la rapidez de lectura (aunque sí importa mucho el volumen de la voz).

Por otra parte, el reconocimiento de gestos generó algunos desafíos a considerar:

- Los cálculos realizados para reconocer los gestos utilizan bastante CPU, por lo que se debió limitar la cantidad de estos por cada gesto reconocido.
- El flujo de vídeo, al igual que el sonoro, es sensible al ruido, por lo que las coordenadas calculadas por el SDK de Kinect pueden estar desviadas. En este sentido, el estabilizador de movimiento del cursor, que además sirve para gestos, funcionó de una manera más que correcta.
- El reconocimiento de gestos se realiza por un periodo determinado de tiempo, por lo tanto, fue necesario implementar un registro de posiciones de un tamaño específico (un historial).

- La ejecución de ciertos gestos opuestos, genera conflictos, que fueron solucionados por un sistema de bloqueo temporal de gestos.

El reconocimiento del gesto empuñar la mano, para seleccionar los términos a unir, no se logró con las librerías nativas de Kinect, por lo tanto, se reemplazó por el gesto de empujar, que en la práctica funciona muy bien.

También, se destaca que el desarrollo del modo zurdo no se planificó, pero se logró gracias a una implementación estructurada y visionaria. Aún así, en las pruebas se identificaron algunos problemas con este modo, que lo inhabilitaban, por lo tanto, se debió invertir tiempo en su reparación.

La actividad de términos pareados se integra muy bien con los gestos. La implementación de gestos para subir y bajar por las listas de términos, fue un desafío que se cumplió satisfactoriamente. Los botones donde están los términos, tienen un ajuste automático del tamaño según el contenido, lo que consigue bastante comodidad para su uso.

La interfaz gráfica lograda, fue mucho mejor que la planeada, ya que la incorporación de el énfasis en tiempo real de las palabras correctas e incorrectas en la lectura, y una barra de estado que permite ver el progreso de los términos pareados, le entregan mayor calidad de uso al software. Además, el uso del estándar Material Design de Google, usando colores fuertes y bien definidos, animaciones suaves para cada control, tamaño y estilo de fuente conveniente para niños pequeños, y un modo ayuda auditivo, le entrega una funcionalidad al prototipo bien recibida por el usuario.

Los indicadores de progreso en la lectura, permiten una retroalimentación propicia al estudiante, también la barra de estado en los términos pareados, que le ayuda a saber si lo está haciendo bien o mal, así como los variados mensajes y notificaciones que entregan información relevante al usuario.

La implementación de los módulos fue una tarea ardua, considerando que el desarrollador no poseía experiencia con las tecnologías utilizadas. En la práctica, se utilizaron rasgos de metodologías ágiles para el desarrollo, como un diseño ligero (poco detallado) y un enfoque en la implementación. Estos módulos fueron bien

documentados, en consecuencia, pueden ser utilizados por otros desarrolladores para agilizar la construcción de aplicaciones para Kinect.

Estos módulos proporcionan un reconocimiento robusto de palabras, frases o textos, y un reconocimiento de gestos, el cual puede ser ampliado con las diversas funciones de cálculo de distancias y ángulos, y un caché especializado para analizar el historial de posiciones. No obstante, las múltiples variables de calibración para estos módulos, junto a la complejidad de algunas funciones, hacen necesario un entendimiento completo de las capacidades de Kinect y su programación, antes de usarlos.

El prototipo recibió una buena aceptación por parte de los niños en las pruebas realizadas. El reconocimiento de las lecturas fue correcto, y la función de reconocimiento de gestos aplicados a los términos pareados fue la que los entretuvo mucho más tiempo.

Existe una dificultad al comienzo en el aprendizaje de los gestos para controlar los términos pareados, lo que tiende a desviar los esfuerzos de los estudiantes hacia aprender cómo ocupar Kleng, y no completar correctamente la actividad.

Luego de observar el comportamiento del prototipo en las pruebas, y el uso que le daban los niños, se concluye que este es perfectamente utilizable por ellos en sus casas, como un juego didáctico. Con Kleng, los niños podrán poner a prueba su modulación leyendo lecturas a gusto, y sus habilidades psicomotoras al unir términos pareados con los gestos.

Los objetivos se consiguieron, con creces, a pesar de las variadas complejidades que surgieron, y se espera que este trabajo pueda ser utilizado y motive a desarrolladores en la región y el país a crear proyectos educativos usando Kinect, o dispositivos similares.

9. Trabajos Futuros.

Luego de observar los resultados, se hace necesario proponer algunas ideas útiles para el uso de este trabajo y el prototipo, y futuros sistemas en base a este.

Respecto al desarrollo del prototipo:

- La iconografía pudiera no ser la adecuada, también las imágenes, las cuales fueron seleccionadas sin algún parámetro específico que tomara en cuenta la usabilidad.
- La función de Cargar podría causar confusión al utilizar Kleng, ya que obliga a recargar la actividad cada vez que se desee realizar, aunque sea la misma.
- La ayuda auditiva posee una voz sintética que no es apropiada en aplicaciones para niños.
- Se debió evaluar la incorporación de animaciones al personaje principal, la hormiga del logo de Kleng, durante la ejecución del software, de manera que sea más atractivo su uso para los niños.

Respecto a otro tipo de actividades:

- Con el reconocimiento de voz, se podría crear una actividad de “descripción de imágenes”, en la cual se mostraría una imagen y el estudiante podría describirla. El sistema reconocería algunas palabras claves por lo cual evaluaría qué tan bien el estudiante describió dicha imagen.
- Algo más complejo sería un sistema de “comprensión lectora” que, mediante la misma técnica anterior, pero con mayor diversidad gramatical, pudiera verificar si un relato posterior a una lectura del estudiante corresponde a una buena comprensión lectora.
- Con gestos, la implementación de una actividad de selección múltiple, donde el estudiante seleccione la palabra correctamente escrita.
- Incluso Kleng podría servir de base para sistemas de aprendizaje de inglés utilizando la voz, o matemáticas, o muchas otras asignaturas utilizando los términos pareados.

10. Definiciones, siglas y abreviaciones del proyecto.

A continuación, se presentan algunos significados y abreviaciones de relevancia para el proyecto, en orden alfabético:

- **FR**: Requerimiento Funcional (*del inglés Functional Requirement*).
- **Joint**: en el reconocimiento de gestos de Kinect, es una parte del skeleton identificada.
- **Kinésica**: viene del griego kinesis, que significa movimiento, y estudia el significado apelativo o comunicativo de los movimientos corporales y de los gestos aprendidos.
- **KL**: Kleng. Abreviación de “Kinect para lenguaje”.
- **PCM**: Modulación por Impulsos Codificados (*del inglés Pulse Code Modulation*). Audio digital sin compresión.
- **PDF**: Formato de Documento Portable (*del inglés Portable Document Format*).
- **PR**: Requerimiento del Proyecto, o Restricción (*del inglés Project Requirement*).
- **QR**: Requerimiento de Calidad (*del inglés Quality Requirement*).
- **Skeleton**: en el reconocimiento de gestos de Kinect, es el cuerpo identificado.
- **UC**: Caso de Uso (*del inglés Use Case*). Componente del UCD que representa una interacción de un Actor con el Sistema. Se representa con un óvalo.
- **UCA**: actor de los casos de uso (*del inglés Use Case Actor*).
- **UCD**: Diagrama de Casos de Uso (*del inglés Use Cases Diagram*). Diagrama UML que permite la representación gráfica de las funcionalidades del sistema.
- **UCS**: Especificación de Casos de Uso (*del inglés Use Case Specification*). Tabla que describe el proceso de uso, en lenguaje natural.
- **WPF**: Windows Presentation Foundation. Modelo de programación unificado, creado por Microsoft para facilitar la implementación de soluciones con GUIs sobre su sistema operativo Windows. Separa las vistas de la lógica (MVC).

11. Referencias.

- [1] Emol, «Estudio revela que sólo el 22% de los colegios en Chile tiene una alta cultura de innovación,» *El Mercurio*, Santiago, 2015.
- [2] Gobierno de Chile, Ministerio de Educación, «Resultados SIMCE 2014,» 2015.
- [3] M. Román y F. J. Murillo, «Disponibilidad y uso de TIC en escuelas latinoamericanas: incidencia en el rendimiento escolar,» *Educ. Pesqui.*, vol. 40, n° 4, pp. 879-895, diciembre 2014.
- [4] P. Jaramillo, «Uso de las Tecnologías de la Información en el aula. ¿Qué Saben Hacer los Niños con los Computadores y la Información?,» *Revista de Estudios Sociales*, pp. 27-44, junio 2005.
- [5] R. Cornejo Chávez, «EL Experimento Educativo Chileno 20 años después: una mirada crítica a los logros y falencias del sistema escolar,» *REICE*, vol. 4, n° 1, pp. 118-129, 2006.
- [6] La Nación, «¿Aburridos en el aula? La causa también está fuera de la escuela,» 15 mayo 2010. [En línea]. Available: <http://www.lanacion.com.ar/1264843-aburridos-en-el-aula-la-causa-tambien-esta-fuera-de-la-escuela>. [Último acceso: 4 septiembre 2016].
- [7] Educrea, «Técnicas de enseñanza para mejorar la motivación de los estudiantes,» 2015. [En línea]. Available: <http://educrea.cl/tecnicas-de-ensenanza-para-mejorar-la-motivacion-de-los-estudiantes/>. [Último acceso: 4 septiembre 2016].
- [8] Innovación, «BID destaca tres innovaciones chilenas en educación,» 7 julio 2014. [En línea]. Available: <http://www.innovacion.cl/2014/07/bid-destaca-tres-innovaciones-chilenas-en-educacion/>. [Último acceso: 24 octubre 2016].
- [9] Educarchile, «Cómo enseñamos Lenguaje y Matemáticas en el primer ciclo,» 3 marzo 2006. [En línea]. Available: <http://www.educarchile.cl/ech/pro/app/detalle?id=107341>. [Último acceso: 21 septiembre 2016].
- [10] Ministerio de Educación, «Implementación Curricular en el Aula. Lenguaje y Comunicación Primer Ciclo Básico (NB1 y NB2).,» Santiago, 2004.
- [11] J. Beltrán Llera y J. Bueno Álvarez, *Psicología de la Educación*, vol. 18, Barcelona: Boixareu Universitaria, 1995.

- [12] M. T. Lugo y V. Kelly, «Tecnología en educación ¿Políticas para la innovación?,» de *IIFE UNESCO*, Buenos Aires, 2010.
- [13] P. Guerrero, «Estudio de las Resistencias de los Profesores a una Estrategia Para el Desarrollo de la Creatividad en Tres Unidades Educativas,» *PSYKHE*, vol. 14, nº 1, pp. 31-45, mayo 2005.
- [14] R. Nahuelcura, «El Definido,» marzo 2013. [En línea]. Available: http://www.eldefinido.cl/actualidad/lideres/1150/Roxana_Nahuelcura_no_hay_alumnos_flojos_hay_alumnos_desmotivados/. [Último acceso: 17 abril 2016].
- [15] I. Ayala, S. Hernández, E. García y N. Rocha, «Desmotivacion Escolar, causas y posibles soluciones (Investigacion Educativa),» 27 octubre 2013. [En línea]. Available: <http://quintobinvestigaciones.blogspot.cl/>. [Último acceso: 31 octubre 2016].
- [16] N. Yannier , K. Koedinger y S. Hudson , «Learning from Mixed-Reality Games: Is Shaking a Tablet as Effective as Physical Observation?,» de *CHI 2015. Kids Haptic, Wearable, Tangible Learning.*, Seoul, 2015.
- [17] Á. M. Rodríguez, «TuWindowsMundo,» 2015. [En línea]. Available: <https://www.tuwindowsmundo.com/kinect-es-usado-como-un-juego-educativo-eficaz/>. [Último acceso: 26 septiembre 2016].
- [18] Educación 3.0, «Los colegios SEK incorporan Kinect en su curriculum educativo,» 8 noviembre 2011. [En línea]. Available: <http://www.educaciontrespuntocero.com/noticias/los-colegios-sek-incorporan-kinect-en-su-curriculum-educativo/3955.html>. [Último acceso: 26 septiembre 2016].
- [19] Microsoft, «Convenciones de código de C# (Guía de programación de C#),» [En línea]. Available: <https://msdn.microsoft.com/es-es/library/ff926074.aspx>. [Último acceso: 4 junio 2016].
- [20] Microsoft, «Kinect for Windows Architecture,» [En línea]. Available: <https://msdn.microsoft.com/en-us/library/jj131023.aspx>. [Último acceso: 31 octubre 2016].
- [21] Microsoft Software Development Network, «Valores de métrica de código,» [En línea]. Available: <https://msdn.microsoft.com/es-es/library/bb385914.aspx>. [Último acceso: 28 noviembre 2016].
- [22] J. Webb y J. Ashley, *Beginning Kinect Programming with the Microsoft Kinect SDK*, Apress, 2012.

- [23] G. Borenstein, Making Things See: 3D vision with Kinect, Processing, Arduino, and MakerBot, 1 ed., A. Odewahn, B. Jepson y H. Bauer, Edits., O'Reilly, 2012, p. 440.
- [24] I. Sommerville, Ingeniería del Software, 7 ed., M. M. Romo, Ed., Madrid: Pearson Educación S.A., 2005.
- [25] Microsoft, «Kinect for Windows Sensor Components and Specifications,» [En línea]. Available: <https://msdn.microsoft.com/en-us/library/jj131033.aspx>. [Último acceso: 2 julio 2016].
- [26] Microsoft, «Human Interfaces Guidelines for Kinect,» 2013.

Agradecimientos.

El desarrollo de este proyecto no hubiera sido posible sin el apoyo de muchas personas, que me entregaron su visión, conocimiento y motivación en diversos aspectos, directa o indirectamente relacionados al proyecto. No obstante, solo me remitiré a nombrar a las personas que directamente influyeron al desarrollo de este.

A Carola Figueroa Flores, profesora quien me ofreció la idea de desarrollar actividades didácticas utilizando Kinect como proyecto de título. Por su confianza en mí y su buena disposición siempre, muchas gracias.

A María Antonieta Soto, mi profesora guía, quien me orientó y corrigió mis avances a lo largo del proyecto. Sin ella no hubiera sido posible un enfoque formal del informe, con la información correctamente plasmada. Por su excelente disposición, ayudarme con su conocimiento y experiencia, y por su tiempo, que fue mucho, de verdad muchas gracias.

A Sandra Navarrete Méndez, profesora quien me ayudó con la realización de las pruebas en el colegio Sagrado Corazón en San Carlos. Por su tiempo y ayuda, muchas gracias.

A Alejandra Fuentes Lagos, profesora quien me apoyó y orientó en los inicios del proyecto. Muchas gracias.

Y, por último, muchas gracias a todos y cada uno de los docentes que formaron parte de mi experiencia en la carrera de Ingeniería Civil en Informática en la Universidad del Bío-Bío, ya que cada pequeño aprendizaje que adquirí gracias a ellos, perfeccionaron mis capacidades, me entregaron las competencias necesarias para abordar este proyecto y, por sobretodo, me hicieron crecer como persona, para lograr ser un profesional.