Herramienta Text-To-Speech para Open Document

Manuel Vélez Martínez
Ingeniería informática
Compiladores

Gerard Enrique Manonellas
Robert Clarisó Viladrosa

06/2016

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada 3.0 España de Creative Commons

Licencias alternativas (elegir alguna de las siguientes y sustituir la de la página anterior)

A) Creative Commons:

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-CompartirIgual 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-SinObraDerivada 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-CompartirIgual 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento 3.0 España de Creative Commons

B) GNU Free Documentation License (GNU FDL)

Copyright © AÑO TU-NOMBRE.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
A copy of the license is included in the section

entitled "GNU Free Documentation License".

C) Copyright

© (el autor/a)
Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Herramienta text-to-speech para open document*
Nombre del autor:	*Manuel Vélez Martínez*
Nombre del consultor/a:	*Gerard Enrique Manonellas*
Nombre del PRA:	*Robert Clarisó Viladrosa*
Fecha de entrega (mm/aaaa):	06/2016
Titulación::	*Ingeniería informática*
Área del Trabajo Final:	*Compiladores*
Idioma del trabajo:	***Castellano***
Palabras clave	*Tts, openDocument, audio libre*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.*

La finalidad del proyecto es la realización de una herramienta que analice y extraiga texto de un documento en formato open document y basándose en un fichero de configuración elaborado por el usuario genere uno o varios ficheros de audio en formato libre.

El desarrollo del proyecto tiene, por tanto, aplicaciones en diversas areas tales como la formación, el turismo, personas con diversidades funcionales, etc.

Las tecnologías utilizadas para el desarrollo del proyecto han incluido el formato XML y la validación mediante esquemas XSD, el formato OpenDocument (ODF), un estándar de ofimática creado por OASIS basado en XML, librerías y aplicaciones de text-to-speech (TTS) tales como freeTTS o eSpeak y librerías de audio tales como Jave.

El proyecto se ha desarrollado íntegramente en Java utilizando maven para la gestión de todas las librerías necesarias durante el proyecto.

Se ha generado, por tanto, como resultado del proyecto una herramienta que parsea documentos ODF, en concreto documentos de texto, hojas de calculo y presentaciones, y en base a la configuración seleccionada por el usuario genera audios en el formato libre ogg. Los idiomas soportados por la aplicación son, actualmente, Castellano, Catalán e Ingles siendo fácilmente extensible a todos los lenguajes soportados por el servicio TTS de google o el binario de eSpeak

Abstract (in English, 250 words or less):

The main purpose for this project is to create a tool that parses and extract all the text in a document in open document format and using the configuration stored in a config file created by the user generates one or several audio files in free audio format.

This project can be used in areas as training, tourism and for people with functional diversity.

Technologies used during the development phase include XML format and XSD schema, OpenDocument format (ODF), a XML based format created by OASIS, TTS libraries and applications as freeTTS or eSpeak and audio libraries as Jave.

Project is fully developed in java using Maven for managing all external dependencies needed for the project.

The project output is a tool that parses ODF documents, specifically text documents (ODT), spreadsheets (ODS) and presentations (ODP), and using the setup provided by the user in xml files, generates free audio files in ogg format. Spanish, Catalonian and English are the languages supported by the application being easily extensible to all languages supported by the google TTS Service of eSpeak application.

Índice

Capitulo 1 . Definición del TFC....6
1.1. Introducción....6
1.2. Descripción del trabajo....6
1.3. Objetivos....7
1.4. Estudio de productos existentes en el mercado....7
1.5. Análisis de necesidad del proyecto....8
Capitulo 2 . Plan de Trabajo....10
2.1. Tareas y Subtareas....10
2.1.1. Definición de las tareas....11
Tarea 1. Plan de trabajo. Entregable correspondiente a la PEC 1....11
Tarea 2. Análisis de requisitos, herramientas y entorno tecnológico. Entregable correspondiente a la PEC 2....11
Tarea 3. Diseño e implementación preliminar del proyecto. Entregable correspondiente a la PEC 3....12
Tarea 4. Realización de la memoria final y entrega del producto terminado....13
Tarea 5. Realización de una vídeo presentación....13
Hitos principales del proyecto....14
2.2. Productos a entregar....14
2.3. Planificación temporal del proyecto....16
2.4. Metodología de trabajo. Método de desarrollo....18
Capitulo 3 . Análisis de requisitos....19
3.1. Objetivos principales....19
3.2. Requisitos de almacenamiento de información....19
3.3. Requisitos funcionales....21
3.3.1. Requisitos de configuración....21
3.3.2. Requisitos del módulo de extracción de texto....21
3.3.3. Requisitos del módulo de producción de audio....22
3.3.4. Requisitos de la interfaz de usuario....23
3.4. Actores principales y perfiles de usuarios....23
3.4.1. Actores principales....23
3.4.2. Perfiles de usuario....24
Usuario....24
3.5. Diagramas de casos de uso....25
3.6. Requisitos no funcionales....26
3.6.1. Requisitos de rendimiento, interacción e integración....26
3.6.2. Restricciones de diseño....26
3.7. Descripción y justificación de las herramientas y del entorno tecnológico para el desarrollo....27
3.7.1. Lenguaje de programación....28
3.7.2. Entorno de programación (IDE)....30
3.7.3. Plataforma de desarrollo....32
3.7.4. Plataforma de ejecución....32
3.7.5. Librerías de tratamiento de formato Xml....33
3.7.6. Librerías de tratamiento de formato Open Document....34
3.7.7. Librerías TTS....38
3.7.8. Librerías de formatos de audio....40
Capitulo 4 . Diseño del sistema....43

4.1. Diseño de la interfaz del usuario....43
4.1.1. Diseño de los wireframes....45
01.Ventana principal de la aplicación:....45
4.1.2. Flujo de navegación....51
4.2. Diseño del sistema....52
4.2.1. Diagrama de clases del sistema....52
Reader....52
GuiReader....52
CliReader....53
ODFParser....53
ODTParser....53
ODSParser....53
ODPParser....53
TTS....53
OnLineTTS....54
OffLineTTS....54
EspeakTTS....54
AudioManager....54
4.3. Modelo dinámico....55
4.3.1. Diagramas de interacción....55
Diagrama de interacción 01. Conversión OffLine de fichero de texto en modo linea de comandos:....55
Diagrama de interacción 02. Conversión OnLine de fichero de texto en modo línea de comandos:....56
Diagrama de interacción 03. Conversión OffLine de fichero de texto en modo interfaz gráfica de usuario:....57
Diagrama de interacción 04. Conversión OnLine de fichero de texto en modo interfaz gráfica de usuario:....58
Diagrama de interacción 05. Conversión OffLine de hojas de cálculo en modo linea de comandos:....59
Diagrama de interacción 06. Conversión OnLine de hojas de cálculo en modo linea de comandos:....60
Diagrama de interacción 07. Conversión OffLine de hojas de cálculo en modo interfaz gráfico de usuario:....62
Diagrama de interacción 08. Conversión OnLine de hojas de cálculo en modo interfaz gráfico de usuario:....63
Diagrama de interacción 09. Conversión OffLine de presentación en modo linea de comandos:....64
Diagrama de interacción 10. Conversión OnLine de presentación en modo linea de comandos:....65
Diagrama de interacción 11. Conversión OffLine de presentación en modo interfaz gráfica de usuario:....66
Diagrama de interacción 12. Conversión OnLine de presentación en modo interfaz gráfica de usuarios:....67
Diagrama de interacción 13. Configuración inválida en modo linea de comandos:....68
Diagrama de interacción 14. Configuración inválida en modo interfaz de usuario:....70
Diagrama de interacción 15. Formato no soportado en modo linea de comandos:....71

Diagrama de interacción 16. Formato no soportado en modo interfaz de usuario:....72
4.3.2. Diagramas de actividades....73
Diagrama de actividades en modo GUI....73
Diagrama de actividades en modo línea de comandos....74
4.4. Diseño arquitectónico....75
4.5. Diseño de la configuración....78
4.5.1. Fichero de configuración xml....78
4.5.2. Fichero xsd....79
4.5.3. Fichero log4j.properties....79
Capitulo 5 . Resultados y análisis del proceso....81
5.1. Proceso de implantación....81
5.1.1. Windows....81
5.1.2. Linux....82
5.1.3. MacOSX....83
5.2. Manual de usuario....84
5.2.1. Precondiciones....84
5.2.2. Funcionalidades de la aplicación....84
5.2.3. Interfaz gráfica de usuario....85
Generación del audio en un solo fichero a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz offline....86
Generación del audio en un varios ficheros a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz online....87
Generación del audio en un varios ficheros a partir de un documento de texto de varias páginas modificando la configuración. Proveedor de voz offline....88
5.2.4. Línea de comandos....89
Generación del audio en un solo fichero a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz offline....89
Generación del audio en un varios ficheros a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz offline....90
Generación del audio en un solo fichero a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz online 90
Generación del audio en un varios ficheros a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz online....91
5.3. Procedimientos de pruebas....93
5.3.1. Tests realizados....94
5.3.2. Descripción detallada de los tests....97
Capitulo 6 . Objetivos conseguidos....114
Capitulo 7 . Mejoras y Trabajo futuro....115
Capitulo 8 . Conclusiones....118
Capitulo 9 . Agradecimientos....119
Capitulo 10 . Bibliografía....120

Capitulo 1 Definición del TFC

1.1 Introducción

Con el aumento del uso de las nuevas tecnologías ha aumentado la necesidad de transmitir información entre personas mediante diversos métodos. Dicha necesidad de distribución de la información y el conocimiento ha producido que en la actualidad existan una gran cantidad de formatos electrónicos en los que se pueda almacenar información para ser compartidas (ficheros de texto plano, documentos de texto enriquecido, páginas web, bases de datos, etc).

La transmisión de la información en formatos electrónicos entre personas ha señalado la necesidad de hacer disponibles estas informaciones a personas con diversidades funcionales. Está claro, además, que no es factible para una persona con una deficiencia funcional el acceso a la información almacenada en un fichero de texto o en una base de datos si no se le prestan los medios necesarios para que dicha información pueda ser percibida de una manera alternativa.

Por lo tanto, y teniendo en cuenta las posibles necesidades de cualquier tipo de usuario de un ordenador, es importante que exista la posibilidad de la traducción de documentos de cualquier tipo a medios tiempo-dependientes que faciliten a cualquier tipo de usuario la posesión, utilización y distribución de los datos profesionales, docentes o personales generados y almacenados en ellos.

1.2 Descripción del trabajo

El presente proyecto fin de carrera pretende, por tanto, desarrollar una herramienta text-to-speech[1] que sintetice documentos de ofimática y produzca uno o varios documentos de audio que puedan ser oídos por el usuario que utilice la herramienta.

Para realizar el trabajo se pretende aprovechar todas las posibilidades abiertas disponibles como son los formatos de audio libre, herramientas y librerías tts de código abierto, ficheros de configuración Xml [2] y documentos con formato open document [3] [11].

Por otro lado, es importante que la aplicación sea configurable de manera que se permita al usuario especificar puntos importantes como la división de los audios en capítulos, páginas o no dividirlo. Se podrá especificar el lenguaje del documento o autodetectarlo y algunas otras configuraciones facilitadas por los documentos open document.

1.3 Objetivos

Una vez expuestos los datos de la introducción y de la descripción se puede asegurar que se ha encontrado una necesidad desde la que se comience con el estudio, el diseño y el posterior desarrollo de una herramienta que permita a usuarios con posibles deficiencias de visión obtener y almacenar ficheros de audio con el contenido sintetizado de ficheros con información como texto, formulas, o cualquier otro dato de interés del usuario.

Una vez identificada la idea y la necesidad del proyecto, el desarrollo de una herramienta text-to-speech, se debe profundizar en cuales serán los elementos que formarán parte del proyecto. De esta manera se pretende que el tipo de fichero que se use como entrada de la herramienta sea un documento OpenDocument Format (texto, hojas de cálculo, presentaciones, etc), la configuración se almacene en un fichero Xml y los audios producidos tengan un formato de audio abierto.

Así pues, los objetivos del proyecto de desarrollo de una herramienta text-to-speech para documentos open document son los siguientes:

- Estudio de la disponibilidad actual de herramientas TTS para el formato ODF
- Estudio del formato ODF y selección de los elementos mas importantes de dicho formato
- Definición de un documento XSLT [4] que permita transformar el contenido del fichero ODF en un formato utilizable por la herramienta TTS seleccionada
- Estudio de las herramientas de síntesis TTS existentes
- Estudio de los formatos de audio Libre
- Definición de un documento de configuración en formato XML
- Creación de una aplicación en java con los siguientes módulos:
    - Módulo de configuración
    - Módulo de extracción de texto del documento odf
    - Módulo de generación de audio mediante herramientas TTS
    - Módulo de conversión de audio a formato de audio libre

1.4 Estudio de productos existentes en el mercado

En la actualidad existen una gran cantidad de aplicaciones que convierten texto a voz. Existen diversas páginas web, plugins para navegadores o incluso editores de texto.

Muy pocas de estas aplicaciones están orientadas a la generación de audio a partir de un documento de texto en formato open document, concretamente solo hay dos herramientas que realizan esta tarea:

- **Read Text:** [5] Es una extensión de open office que convierte en voz documentos en formato open office. Está disponible para cualquier plataforma en la que se pueda instalar open office y cuenta con un servicio de síntesis propio que puede ser extendido mediante la instalación de otros sistemas como eSpeak.
- **OO Text To Speech:** [6] Es una macro TTS disponible para open office con un motor de lectura que puede leer cualquier documento de texto. No ha sido actualizado desde 2006

Existen algunas alternativas que pueden ser adaptadas al fin que se persigue con el presente proyecto pero que o bien no son herramientas libres o bien requieren de la conversión del documento a otro formato:

- **Word Talk:** [7] Es un plugin gratuito para microsoft word (solo permite leer documentos de texto) que lee un texto y lo resalta al mismo tiempo. Permite exportar a formatos de audio que no son libres como Wav o Mp3.
- **Zamzar:** [8] Es una web que permite subir ficheros PDF, DOC o TXT y los convierte a MP3. Es simple pero requiere de conversiones previas y no se puede exportar a ningún tipo de audio librería.

Además, podemos contar en esta sección aplicaciones nativas para cada sistema operativo que leen lo que se muestra en pantalla. Ejemplos de estas aplicaciones son **JAWS** screen reader [9] o el propio screen reader de Ubuntu (**Orcas**) [10] que son sistemas dependientes del sistema operativo y que no permiten exportar a formatos de audio

1.5 Análisis de necesidad del proyecto.

Una vez expuestos ciertos antecedentes y productos similares se puede ver que el proyecto que se pretende desarrollar podría ocupar algunos de los huecos disponibles entre las demás aplicaciones del mercado.

El proyecto podría ser utilizable para usuarios o empresas con alguno de los siguientes perfiles:

- Usuarios que por su puesto de trabajo no pueden mirar una pantalla, como por ejemplo conductores que podrían recibir ciertas instrucciones de rutas o detalles técnicos que les podrían ayudar en el desarrollo de su función.
- Usuarios que desempeñan sus tareas en condiciones de poca visibilidad y que necesitan acceder a ciertos elementos de información como pudieran ser manuales de instrucciones o referencias técnicas
- Usuarios que están aprendiendo un idioma y quieren conocer como es la pronunciación de un determinado texto docente.

- Usuarios que, por condiciones de diversidad funcional no pueden leer documentos, o que incluso pretenden escribir sus pensamientos para que sean traducidos a voz y así puedan mejorar su comunicación.
- Empresas con puntos de información automatizados en los que el usuario selecciona un ítem de datos y el sistema se lo lee en voz alta. Por ejemplo puntos de información de un centro comercial.
- Instituciones culturales y museos que pretendan generar audioguias en diversos idiomas sin la necesidad de contar con un locutores que hablen los idiomas de las audio-guías. Por ejemplo, un museo que cuenta con descripciones de todos los cuadros en varios idiomas podría grabar y distribuir dichos audios a sus visitantes .

Capitulo 2 Plan de Trabajo

2.1 Tareas y Subtareas

La planificación temporal del proyecto fin de carrera está claramente marcada por cada uno de los entregables (Que se definirán en el punto 2.2 del presente documento) durante el semestre en el que se desarrollará dicho proyecto.

De esta manera, tenemos 6 tareas principales (Cada una de las entregas) que se han dividido en subtareas mas pequeñas:

1. PEC 1 (plan de trabajo)
    1.1. Redacción del plan de trabajo [Capítulo 2 de la memoria]
2. PEC 2 (Análisis de requisitos, herramientas y entorno tecnológico)
    2.1. Estudio del formato Open Document, Xml y Xslt
        2.1.1. Análisis de los formatos
        2.1.2. Redacción del estudio [Anexo A]
    2.2. Estudio de formatos de audio abiertos
        2.2.1. Análisis de los formatos de audio abiertos
        2.2.2. Redacción del estudio [Anexo B]
    2.3. Estudio de alternativas TTS
        2.3.1. Análisis de las alternativas TTS
        2.3.2. Redacción del estudio [Anexo C]
    2.4. Análisis de requisitos [Capitulo 3 de la memoria]
3. PEC 3 (Diseño e implementación del sistema)
    3.1. Diseño [Capitulo 4 de la memoria]
        3.1.1. Módulo de configuración de la aplicación
        3.1.2. Módulo de transformación del documento odf
        3.1.3. Módulo de generación de ficheros de audio
    3.2. Desarrollo
        3.2.1. Módulo de configuración
        3.2.2. Módulo de transformación
        3.2.3. Módulo de generación de ficheros de audio
    3.3. Pruebas y bugfixing
4. Memoria Final y productos

4.1. Mejoras del producto

4.2. Redacción de la introducción [Capítulo 1 de la memoria]

4.3. Redacción de los resultados obtenidos y el análisis del proceso [Capítulo 5]

4.4. Redacción de los objetivos conseguidos [Capítulo 6]

4.5. Redacción de Mejoras y Trabajo futuro [Capitulo 7]

4.6. Redacción de las conclusiones [Capitulo 8]

4.7. Redacción de Agradecimientos [Capítulo 9]

4.8. Redacción de la Bibliografía [Capítulo 10]

5. Vídeo presentación
6. Tribunales virtuales

2.1.1 Definición de las tareas

Tarea 1. Plan de trabajo. Entregable correspondiente a la PEC 1.

En esta tarea se definirá y redactará el plan de trabajo que contendrá todas las tareas y subtareas necesarias para la realización del proyecto fin de carrera. También se incluirán todos los hitos del proyecto organizados por fecha, se detallarán los productos a entregar, la metodología de trabajo que se seguirá durante el proyecto y, finalmente, la planificación temporal de tareas del proyecto.

De esta tarea se realizará un documento en el que se redacten todos los puntos enumerados en el párrafo anterior y que corresponderá a la PEC 1.

Tarea 2. Análisis de requisitos, herramientas y entorno tecnológico. Entregable correspondiente a la PEC 2.

A la entrega de la PEC 2 se pretende analizar y definir los requisitos necesarios para el diseño y la posterior codificación de la aplicación que se pretende construir en el presente proyecto fin de carrera. Además, tras la realización de esta tarea se pretende haber tomado una decisión sobre las herramientas a utilizar para el desarrollo así como el entorno tecnológico necesaria para el desarrollo y la explotación de los resultados del proyecto.

Para la realización de esta tarea se ha decidido realizar las siguientes subtareas:

- **Estudio de los formatos Open Document, Xml y Xslt necesarios para el diseño y el desarrollo de los módulos de configuración y transformación de documentos de la aplicación.** A la finalización de esta tarea se habrá producido un estudio sobre varios puntos: el formato Open Document [11] y los elementos mas importantes que se tendrá en cuenta para la

transformación, sobre el formato Xml a utilizar en los ficheros de configuración y sobre el fichero o los ficheros Xslt de transformación a utilizar así como diversas pruebas de concepto que ayuden a la toma de decisiones sobre las herramientas y el entorno tecnológico.

- Estudio de los formatos de audio abierto necesarios para el diseño y el desarrollo del módulo de generación de ficheros de audio de la aplicación. Este estudio contendrá una comparativa de los formatos de audio libres mas comunes y sus ventajas y desventajas a la hora de utilizarlos como solución para la aplicación text-to-speech que se pretende realizar. Este estudio incluirá la realización de diversas pruebas de concepto que ayudarán a clarificar puntos importantes tales como el rendimiento de la aplicación durante la generación de los ficheros de audio
- Estudio de las alternativas Text-To-Speech existentes en la actualidad y disponibles como herramientas a utilizar durante el desarrollo de los módulos de transformación y generación de ficheros de audio. Al termino del estudio se obtendrá el estado de herramientas TTS existentes para open document y una comparativa de diversas herramientas y librerías que se podrían integrar en la solución propuesta para el proyecto fin de carrera. Este estudio incluirá la realización de pruebas de concepto que ayudará a la clarificación de un punto tan crucial en la aplicación como es la disponibilidad de diversos idiomas en las voces o la claridad con la que la voz sintetizada puede ser percibida por un humano dependiendo de características culturales o geográficas del propio usuario.

Los tres estudios mencionados anteriormente contribuirán a la toma de una decisión razonada sobre la elección de una alternativa de diseño e implementación para cada uno de los módulos en los que se dividirá la aplicación. Además, estos mismos estudios delimitarán el análisis y la producción de los requisitos de la aplicación a desarrollar, por lo que el resultado de esta tarea será el capítulo 3 de la memoria (Análisis de requisitos) así como los anexos de los estudios y las comparativas mencionadas anteriormente

Tarea 3. Diseño e implementación preliminar del proyecto. Entregable correspondiente a la PEC 3.

En este hito del proyecto se entregará el diseño y una implementación mayoritariamente funcional (a la espera de mejoras o solución de bugs) de cada uno de los tres módulo analizados y especificados en la Tarea 2. El diseño y el desarrollo realizados en esta tarea se realizarán teniendo en cuenta los datos y requisitos obtenidos y especificados en la tarea 2 del proyecto.

Al finalizar esta tarea se habrá redactado completamente el módulo 4 de la memoria del proyecto (diseño del sistema) que incluye el diseño del interfaz del usuario, el modelo conceptual de la aplicación, el modelo dinámico y el diseño arquitectónico de la aplicación.

Además, esta fase del proyecto incluirá la definición y ejecución de un proceso

formal de pruebas en el que se detecten errores y desviaciones sobre las especificaciones iniciales de requisitos. Dichos errores conllevarán la necesidad de la realización de una fase de bugfixing.

Tarea 4. Realización de la memoria final y entrega del producto terminado.

Esta fase del proyecto comprenderá las siguientes sub tareas:

- Mejoras del producto: En esta sub tarea se realizará todo el código necesario para solucionar los errores encontrados en la fase de pruebas y para paliar las desviaciones detectadas sobre la especificación inicial. Al finalizar esta fase, el producto debe cumplir las especificaciones al 100% e incluir mejoras sugerida. Todas las desviaciones presentes al finalizar esta fase se especificarán en Capitulo 7 de la memoria. Por contra, el capítulo 6 de la memoria contendrá un listado con todos los objetivos que se han cumplido sobre lo especificado inicialmente.
- Redacción de la introducción del proyecto. Aunque la entrega de la PEC 1 ya incluye parte de la introducción de la memoria, se requiere ampliar dicha introducción con información extra sobre el estudio de productos existentes, o de la situación actual.
- Redacción de los resultados obtenidos y el análisis del proceso. En este capítulo se redactará el manual de usuario y de implantación, la estimación de costes y se incluirá el informe final de pruebas realizadas a la aplicación.
- Objetivos conseguidos. Tal y como se ha explicado anteriormente, en este punto se enumerará todos los elementos de la aplicación que cumplen con los objetivos y requisitos que se plantearon inicialmente y que se han cumplido al final de la fase de desarrollo.
- Mejoras y trabajo futuro. Por el contrario, en este capítulo de la memoria del proyecto se enumeraran las funcionalidades y elementos que, a pesar de estar especificadas inicialmente, no se han podido conseguir por limitaciones tecnológicas o cualquier otro motivo.
- Conclusiones. En este punto se realizará una retrospectiva del proyecto, se analizarán todos los hitos y se identificarán cuales han sido los fallos, los puntos fuertes y las lecciones aprendidas durante el desarrollo
- Agradecimientos del proyecto.
- Bibliografía en formato APA.

Tarea 5. Realización de una vídeo presentación.

Esta tarea incluirá la realización de una presentación en diapositivas que contengan una síntesis del trabajo desarrollado durante el proyecto así como una demostración de la aplicación desarrollada. Dicha presentación se grabará en vídeo que será el

resultado de esta tarea.

Hitos principales del proyecto

Una vez identificadas y desarrolladas cada una de las taras en el punto anterior del presente documento se pueden equiparar los hitos principales del proyecto con cada una de las entregas que se realizarán durante el semestre de desarrollo.

Así pues, los hitos principales del proyecto y sus fechas serán los siguientes:

Título	Entrega	Descripción
PEC 1	09/03/2016	Realización del plan de trabajo del proyecto.
PEC 2	06/04/2016	Creación del análisis de requisitos del proyecto, estudio del análisis de las herramientas existentes y especificación del entorno tecnológico necesario para el desarrollo del proyecto.
PEC 3	18/05/2016	Diseño de los diferentes módulos de la aplicación e implementación preliminar de las cuestiones principales del proyecto.
Memoria final y producto	15/06/2016	En este hito del proyecto se habrá entregado la memoria del proyecto incluyendo los entregables de todas las tareas anteriores y la totalidad de las funcionalidades de la aplicación.
Vídeo presentación	23/06/2016	En este hito se entregará una presentación para la defensa del citado proyecto y un vídeo con una síntesis del trabajo hecho así como una demostración del funcionamiento general de la aplicación.

2.2 Productos a entregar

A la finalización de la fase de desarrollo del proyecto se entregarán los siguientes elementos:

- **Memoria final del proyecto** que incluirá los estudios previos en los que se ha basado el desarrollo, el análisis y especificación de requisitos realizados durante el ciclo del proyecto, el diseño de cada uno de los módulos de la aplicación basado en la toma de decisiones a partir de los estudios realizados así como la justificación de todas las decisiones tomadas durante cualquiera de las fases del proyecto. Además, incluirá el manual de usuario, las conclusiones, los objetivos no alcanzados, posibles mejoras futuras y un informe de las pruebas realizadas a la aplicación
- **Herramienta Text-to-speech para open document** que incluirá la

distribución de un ejecutable de la propia aplicación, el código fuente y la configuración necesaria para el correcto funcionamiento de la aplicación.

- **Vídeo presentación** que incluye una presentación con la defensa del proyecto y una demostración de las funcionalidades implementadas durante la fase de desarrollo del proyecto.

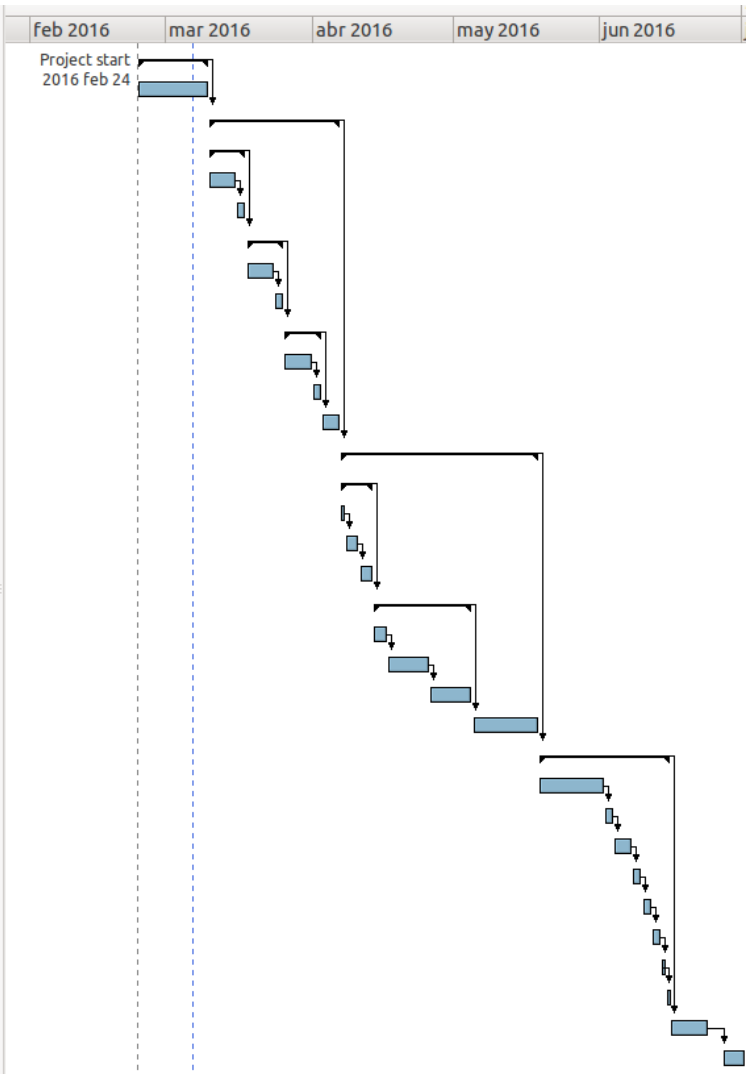
2.3 Planificación temporal del proyecto

Basada en la distribución de los hitos principales del proyecto (punto 2.2 del presente documento) y en la disponibilidad de un único alumno para el desarrollo de la aplicación y la generación de toda la documentación pertinente se ha generado la siguiente lista de tareas con sus esfuerzos asignados:

WBS	Name	Start	Finish	Work
1	**PEC1 (Plan de trabajo)**	**feb 24**	**mar 9**	**15d**
1.1	Redacción del plan de trabajo [Cap.2]	feb 24	mar 9	15d
2	**PAC2 (Analisis de requisitos, herramientas y entorno tec**	**mar 10**	**abr 6**	**28d**
2.1	**Estudio del formato Open Office, Xml y Xslt**	**mar 10**	**mar 17**	**8d**
2.1.1	Analisis del formato Open Office	mar 10	mar 15	6d
2.1.2	Redacción del estudio sobre el formato Open Office	mar 16	mar 17	2d
2.2	**Estudio de formatos de audio abiertos**	**mar 18**	**mar 25**	**8d**
2.2.1	Analisis de los formatos de audio abiertos	mar 18	mar 23	6d
2.2.2	Redacción del estudio sobre formatos de audio abiertos	mar 24	mar 25	2d
2.3	**Estudio de alternativas TTS**	**mar 26**	**abr 2**	**8d**
2.3.1	Analisis de las herramientas TTS	mar 26	mar 31	6d
2.3.2	Redacción del estudio sobre herramientas TTS	abr 1	abr 2	2d
2.4	Analisis de Requisitos [Cap.3]	abr 3	abr 6	4d
3	**PAC3 (Diseño e implementación del sistema)**	**abr 7**	**may 18**	**42d**
3.1	**Diseño [Cap.4]**	**abr 7**	**abr 13**	**7d**
3.1.1	Diseño del módulo de configuración	abr 7	abr 7	1d
3.1.2	Diseño del módulo de transformación del documento odf	abr 8	abr 10	3d
3.1.3	Diseño del módulo de generación de ficheros de audio	abr 11	abr 13	3d
3.2	**Desarrollo**	**abr 14**	**may 4**	**21d**
3.2.1	Módulo de configuración	abr 14	abr 16	3d
3.2.2	Módulo de transformación	abr 17	abr 25	9d
3.2.3	Módulo de generación de ficheros de audio	abr 26	may 4	9d
3.3	Pruebas y bugfixing	may 5	may 18	14d
4	**Memoria Final Y producto**	**may 19**	**jun 15**	**28d**
4.1	Mejoras en el producto	may 19	jun 1	14d
4.2	Redacción de la introducción [Cap.1]	jun 2	jun 3	2d
4.3	Redacción de los resultados y el análisis del proceso [Cap.5	jun 4	jun 7	4d
4.4	Redacción de Objetivos condeguidos [Cap.6]	jun 8	jun 9	2d
4.5	Redacción de Mejoras y Trabajo futuro[Cap.7]	jun 10	jun 11	2d
4.6	Redacción de Conclusiones [Cap.8]	jun 12	jun 13	2d
4.7	Redacción de Agradecimientos [Cap.9]	jun 14	jun 14	1d
4.8	Redacción de la Bibliografía [Cap.10]	jun 15	jun 15	1d
5	Video presentación	jun 16	jun 23	8d
6	Tribunales virtuales	jun 27	jul 1	5d

Teniendo en cuenta la estimación de trabajo expuesta anteriormente, el diagrama de Gantt queda de la siguiente manera:

WBS	Name	Wor
1	**PEC1 (Plan de trabajo)**	**15d**
1.1	Redacción del plan de trabajo [Cap.1	15d
2	**PAC2 (Analisis de requisitos, herrar**	**28d**
2.1	**Estudio del formato Open Office,**	**8d**
2.1.1	Analisis del formato Open Office	6d
2.1.2	Redacción del estudio sobre el for	2d
2.2	**Estudio de formatos de audio abi**	**8d**
2.2.1	Analisis de los formatos de audio	6d
2.2.2	Redacción del estudio sobre form	2d
2.3	**Estudio de alternativas TTS**	**8d**
2.3.1	Analisis de las herramientas TTS	6d
2.3.2	Redacción del estudio sobre herra	2d
2.4	Analisis de Requisitos [Cap.3]	4d
3	**PAC3 (Diseño e implementación de**	**42d**
3.1	**Diseño [Cap.4]**	**7d**
3.1.1	Diseño del módulo de configuraci	1d
3.1.2	Diseño del módulo de transforma	3d
3.1.3	Diseño del módulo de generación	3d
3.2	**Desarrollo**	**21d**
3.2.1	Módulo de configuración	3d
3.2.2	Módulo de transformación	9d
3.2.3	Módulo de generación de ficheros	9d
3.3	Pruebas y bugfixing	14d
4	**Memoria Final Y producto**	**28d**
4.1	Mejoras en el producto	14d
4.2	Redacción de la introducción [Cap.1	2d
4.3	Redacción de los resultados y el aná	4d
4.4	Redacción de Objetivos condeguido	2d
4.5	Redacción de Mejoras y Trabajo futu	2d
4.6	Redacción de Conclusiones [Cap.8]	2d
4.7	Redacción de Agradecimientos [Cap	1d
4.8	Redacción de la Bibliografía [Cap.10	1d
5	Video presentación	8d
6	Tribunales virtuales	5d



2.4 Metodología de trabajo. Método de desarrollo.

Durante el presente plan de trabajo del proyecto se han desgranado las tareas, los hitos así como las diversas entregas derivadas de la metodología de pruebas de evaluación continua de la Universitat Oberta de Catalunya. De esta manera, se establece como fecha de entrega final de la memoria y del producto el día 15 de Junio de 2016 y como fecha de inicio del mismo el día 24 de Febrero de 2016.

Esta planificación dividida en entregables produce que el trabajo sea dividido fácilmente en cada uno de los contenidos anteriormente especificados para cada uno de los hitos existentes en el proyecto.

Por otro lado, teniendo en cuenta el los recursos de los que se dispone para el desarrollo del producto (un sólo alumno) y todas las tareas que son necesarias para llevar a cabo el proyecto, se ha decidido optar por un ciclo de vida clásico o en cascada en la que cada tarea se realizará cuando se haya completado la tarea anterior.

Para finalizar, se debe señalar que es absolutamente necesario que el sistema esté especificado en su mayoría para poder comenzar con el diseño, esté diseñado en su mayoría para poder comenzar con el desarrollo y esté desarrollado en su mayoría para poder comenzar con la fase de pruebas. De lo contrario se corre el riesgo de acumular desviaciones sobre las especificaciones y el diseño previsto para dicho proyecto.

Capitulo 3 Análisis de requisitos

3.1 Objetivos principales

El fin principal del proyecto consiste en el desarrollo de una tarea que permita convertir documentos escritos formato open document (ya sean documentos de texto, hojas de cálculo, presentaciones, etc) a un formato de audio abierto. Este proyecto por tanto deberá cumplir los siguientes objetivos

- Realización de un estudio del panorama actual de las herramientas TTS para documentos Open Document.
- Realizar un análisis del formato Open Document para identificar los elementos mas importantes de cara a la extracción de información
- Realizar un análisis de herramientas y librerías TTS de cara a integrarlas en el diseño del proyecto
- Realizar un estudio de los parámetros de configuración necesarios para proporcionar diversas alternativas de funcionamiento a los usuarios de la aplicación
- Diseñar una herramienta utilizando todo el conocimiento obtenido de los cuatro objetivos principales
- Implementar la herramienta utilizando el diseño generado en el objetivo anterior.

3.2 Requisitos de almacenamiento de información

Debido a la naturaleza del proyecto desarrollado hay un número pequeño de requisitos a la información que es necesaria almacenar para el funcionamiento del sistema:

Id.	Descripción
INFO-001	La configuración de la aplicación se almacenará en un fichero Xml que será uno de los parámetros de entrada para el funcionamiento del software
INFO-002	El documento de entrada de la aplicación será un fichero en formato ODT que estará almacenado en el sistema.
INFO-003	La aplicación producirá como salida tantos ficheros de audio como se hayan especificado inicialmente en el fichero de configuración mediante un campo habilitado al efecto.
INFO-004	La aplicación utilizará un espacio de trabajo temporal que estará almacenado en el mismo directorio donde el fichero de entrada.
INFO-005	Tras la finalización normal de la ejecución de la aplicación el directorio

	temporal de trabajo se eliminará del sistema de ficheros.

3.3 Requisitos funcionales

3.3.1 Requisitos de configuración

El módulo y los ficheros de configuración de la aplicación cumplirán los siguientes requisitos:

Id.	Descripción
CONF-001	El fichero de configuración tendrá formato Xml
CONF-002	Existirá un fichero XSD con el que se validará el fichero de configuración XML
CONF-003	La aplicación tendrá dos modos de ejecución: • Linea de comandos. • Entorno gráfico.
CONF-004	El fichero de configuración Xml tendrá un nodo para la configuración de los detalles del procedimiento TTS
CONF-005	El fichero de configuración Xml tendrá un nodo para la configuración de los detalles del reader
CONF-006	El nodo de configuración TTS contendrá el tag <online> con valores true o false que permitirán que la aplicación utilice un servicio online u offline.
CONF-007	En nodo de configuración TTS contendrá el tag <url> en el que se especificará la url del servicio de voz online.
CONF-008	El nodo de configuración TTS contendrá el tag <language> en el que se configurará el idioma.
CONF-009	El nodo de configuración reader contendrá el tag <split-by> que podrá almacenar los valores PAGE-BREAK o UNIQUE
CONF-010	El nodo de configuración reader contendrá el tag <alternative> con valores true o false.
CONF-011	El nodo de configuración reader contendrá el tag output-path en el que se especificará el path en el que se almacenarán los ficheros de salida.
CONF-012	El nodo de configuración Reader contendrá el tag output-pattern en el que se especificará el formato de los ficheros de salida.

3.3.2 Requisitos del módulo de extracción de texto

Id.	Descripción
EXT-001	El módulo extraerá todo el texto de un documento open document
EXT-002	La extracción de información del documento Open Document se

	realizará de acuerdo a la configuración de la aplicación
EXT-003	En caso de seleccionar PAGE-BREAK en el campo <split-by>, el módulo producirá una cadena de texto por cada salto de página encontrado en el documento de texto
EXT-004	En caso de seleccionar UNIQUE en el campo <split-by>, el módulo producirá una cadena de texto con todo el texto del documento de entrada
EXT-005	En caso de configurar true en el tag <alternative> se incluirá en el texto extraído el texto alternativo de todos los objetos distintos al texto (Formulas, dibujos y cualquier otro elemento no textual del documento)
EXT-006	En caso de configurar false en el tag <alternative> se extraerá el texto alternativo de todos los objetos distintos al texto (Formulas, dibujos y cualquier otro elemento no textual del documento)
EXT-007	En caso de utilizarse almacenamiento temporal para la extracción, este estará almacenado en el directorio que contenga el fichero de entrada.
EXT-008	El modulo de extracción comprobará que tiene permiso de escritura en en todas las rutas necesarias
EXT-009	La aplicación contará con un logger configurable

3.3.3 Requisitos del módulo de producción de audio

Id.	Descripción
AUD-001	El modulo de producción de audio producirá ficheros de sonido que contengan la información extraída de ficheros open document
AUD-002	Los ficheros producidos por la aplicación tendrán formato de audio libre
AUD-003	En caso de configurar la aplicación para utilizar el servicio online TTS (tag <online> con valor YES), se utilizará un servicio TTS disponible en internet
AUD-004	En caso de configurar la aplicación para utilizar la librería offline TTS (tag <online> con valor NO), se utilizará una librería TTS empaquetada en la propia aplicación
AUD-005	En caso de utilizarse almacenamiento temporal para la producción, este estará almacenado en el directorio que contenga el fichero de entrada.
AUD-006	El modulo de producción comprobará que tiene permiso de escritura en en todas las rutas necesarias
AUD-007	El resultado de la producción de audio quedará almacenado en el directorio especificado en el campo <output-path> de la configuración
AUD-008	En caso de producirse solo un fichero de salida, este tendrá el nombre

	especificado en el campo output-pattern mas la extensión del fichero
AUD-009	En caso de producirse varios ficheros de salida, cada uno de ellos tendrá el nombre especificado en el campo output-pattern, un numero y la extensión del fichero

3.3.4 Requisitos de la interfaz de usuario

Id.	**Descripción**
INT-001	Toda la funcionalidad de la aplicación estará disponible desde linea de comandos
INT-002	Toda la funcionalidad de la aplicación estará disponible desde un interfaz de usuario
INT-003	El uso de ambos tipos de interfaz utilizando el mismo fichero de configuración y fichero de entrada producirá exactamente la misma salida.
INT-004	La linea de comandos aceptará dos parámetros obligatorios: • Fichero de configuración • Fichero de entrada en formato open document
INT-005	La interfaz gráfica de usuario mostrará un formulario con los siguientes elementos: • Campo fichero de configuración • Campo fichero de entrada en formato open documentación • Botón ejecutar.

3.4 Actores principales y perfiles de usuarios

La estructura de la aplicación no permite distinguir entre diferentes tipos de usuarios, por lo que, a efectos de documentación, solo existirá un tipo de usuario que se denominará “Usuario”.

3.4.1 Actores principales

Usuario

Este tipo de usuario realizará todas las operaciones que se pueden realizar con la aplicación:

Id	Descripción
AC-001	Conversión de un documento de texto en un solo fichero de audio
AC-002	Conversión de un documento de texto en varios ficheros divididos en secciones

AC-003	Conversión de un documento de texto en varios ficheros divididos en saltos de página
AC-004	Conversión de una presentación
AC-005	Conversión de una hoja de cálculo
AC-006	Conversión de un documento de texto en modo online
AC-007	Conversión de una presentación en modo online
AC-008	Conversión de una hoja de cálculo en modo online
AC-009	Conversión de un documento de texto desde la linea de comandos
AC-010	Conversión de una hoja de cálculo desde la linea de comandos
AC-011	Conversión de una presentación desde la linea de comandos

3.4.2 Perfiles de usuario

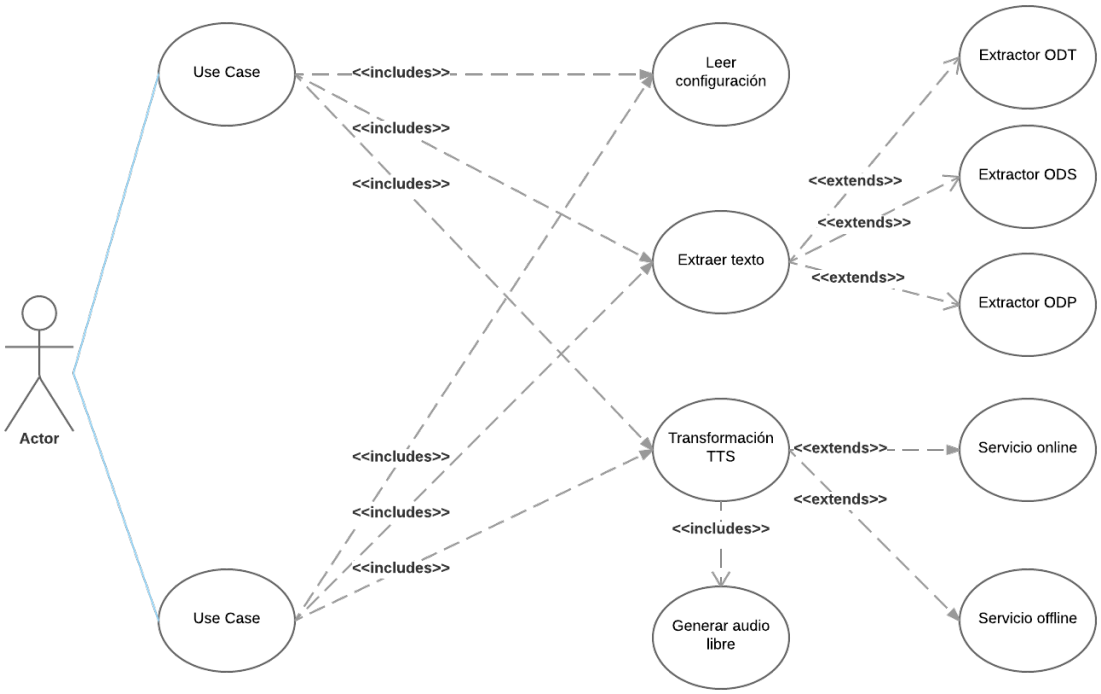
Usuario

Este es el único tipo de usuario que tendrá la aplicación. El perfil de este tipo de usuario cumplirá alguno de los siguientes punto:

- Usuarios con problemas de visión.
- Usuarios que, por su rol profesional, no tienen acceso a una pantalla a una distancia apropiada para leer un documento.
- Usuarios que estudian idiomas y desean oír un texto en el idioma en el que está escrito y que tratan de aprender

3.5 Diagramas de casos de uso

El diagrama de casos de uso que se desprende de los puntos anteriores es muy simple debido a que, como ya se ha explicado, solo tenemos un usuario que puede ejecutar la aplicación de dos maneras que, independientemente de los parámetros, siempre va a llamar a los mismos métodos.



3.6 Requisitos no funcionales

3.6.1 Requisitos de rendimiento, interacción e integración

Rendimiento

Id.	Descripción
PERF-001	El proceso de conversión no consumirá mas del 50% de la cpu
PERF-002	El rendimiento de la aplicación en entorno gráfico y en modo consola no diferirá en mas de un 5% de tiempo de generación de ficheros
PERF-003	El espacio consumido por los ficheros generados no influirá en el rendimiento de la aplicación
PERF-004	El tiempo de ejecución de la aplicación será proporcional al tamaño de los ficheros a convertir. (No es aceptable que el tiempo aumente exponencialmente con el tamaño de los ficheros de entrada)

Interacción

Id.	Descripción
INT-001	El usuario siempre conocerá el estado de la aplicación
INT-002	Todos los errores de la aplicación se comunicarán mediante un popup en el caso de la ejecución en modo gráfico
INT-003	Todos los errores de la aplicación se comunicarán mediante un mensaje en consola en el modo texto.
INT-004	La aplicación tendrá un modo ordenado de recuperarse de los errores producidos durante cualquier procesamiento

Integración

Id.	Descripción
ITG-001	La aplicación comprobará que el endpoint del servicio de Google está disponible antes de comenzar el procesamiento
ITG-002	La comunicación con el endpoint de Google se realizará mediante protocolo http

3.6.2 Restricciones de diseño

Atendiendo a las conclusiones sacadas durante el estudio previo realizado al diseño de la aplicación (Explicadas todas en el punto 3.7 del presente documento) se han

establecido las siguientes restricciones de diseño que afectarán al desarrollo de la aplicación:

Id.	Descripción
RD-001	Java es el lenguaje de programación seleccionado para el desarrollo de la aplicación
RD-002	La aplicación se desarrollará utilizando el IDE Intellij Idea
RD-003	La plataforma de desarrollo será Ubuntu Linux 15.10
RD-004	Las plataformas de ejecución podrán ser Linux, windows y Mac
RD-005	Los ficheros XML de configuración se validarán y parsearán utilizando la librería jdom
RD-006	Los fichero Open Document se parsearán utilizando la librería ODFDOM
RD-007	Se utilizará eSpeak para la producción offline de audio
RD-008	Se utilizará el servicio Google TTS para la producción online de audio
RD-009	Se utilizará la librería Jave para la producción de ficheros de audio
RD-010	El formato de salida seleccionado será OGG
RD-011	El proceso de empaquetado de la aplicación se realizará con Maven
RD-012	El control de versiones seleccionado es Git
RD-013	Se utilizará jenkins como servidor de integración continua

3.7 Descripción y justificación de las herramientas y del entorno tecnológico para el desarrollo

El desarrollo del presente proyecto requiere la toma de ciertas decisiones relativas al entorno tecnológico que se utilizará durante el diseño, la codificación y la explotación del software resultante.

En concreto, las decisiones tomadas son relativas a los siguientes puntos:

- Lenguaje de programación
- Entono de programación
- Plataforma de desarrollo
- Plataforma de ejecución
- Librerías de tratamiento de formato Xml
- Librerías de tratamiento de formato Open Document
- Librerías de TTS

- Librerías de formatos de audio

En todos estos puntos se ofrece un estudio de las alternativas y de las razones que han llevado a seleccionar la alternativa a utilizar durante el desarrollo del proyecto

3.7.1 Lenguaje de programación

Para el desarrollo de este proyecto, teniendo en cuenta los requisitos y las necesidades anteriormente especificados se puede encontrar una gran cantidad de alternativas que permitan llevar a cabo los objetivos planteados. Algunos de dichos lenguajes son Go, Python, Java o C#

GO

Go [12] es un lenguaje de programación concurrente y compilado con una sintaxis parecida a C que ha sido desarrollado por Google y que está disponible para Windows, Linux y MacOS

Ventajas:

- Sintaxis parecida a C (lenguaje muy familiar para cualquier ingeniero informático)
- Es un lenguaje muy eficiente que ayudaría a conseguir requisitos de rendimiento de la aplicación.

Desventajas:

- Falta de experiencia en el uso de dicho lenguaje
- Desconocimiento de IDEs y entornos de programación

Python

Python [13] es un lenguaje de programación interpretado con orientación a objetos, de tipado dinámico y multiplataforma

Ventajas:

- Es un lenguaje muy fácil de aprender en el que el desarrollador posee una importante experiencia profesional.
- Existe una gran cantidad de alternativas para cualquiera de las necesidades descritas en el presente proyecto.
- Multiplataforma

Desventajas:

- Al ser un lenguaje interpretado el rendimiento de la aplicación podría plantear ciertos problemas.

- Dado que ciertas librerías de Python están diseñadas e implementadas para ser utilizadas en un sistema operativo concreto se pueden encontrar problemas en la necesidad de hacer que la aplicación sea multi-plataforma

Java

Java [14] es un lenguaje de programación de propósito general, concurrente y orientado a objetos diseñado para permitir la ejecución de un mismo programa en múltiples sistemas operativos

Ventajas:

- Es un lenguaje de programación muy extendido con una comunidad muy extensa por lo que es muy fácil obtener información para el proyecto
- Existe una gran cantidad de alternativas para cualquiera de las necesidades descritas en el presente proyecto independientes de la plataforma
- El lenguaje es multiplataforma de manera que la compilación del proyecto puede ser ejecutada en cualquier sistema con una maquina virtual de java
- El desarrollador del proyecto posee experiencia, tanto en la universidad como en el mundo laboral, en desarrollos en java y otros lenguajes basados en la JVM como Groovy [15] o Kotlin [16]

Desventajas:

- La propia arquitectura del lenguaje de java así como el compilador que traduce el código fuente a bytecode [17], que finalmente es interpretado, puede hacer que se presenten ciertos problemas de rendimiento

C#

C# [18] es un lenguaje de programación de orientado a objetos desarrollado por microsoft como parte de la plataforma .NET [19].

Ventajas:

- Es un lenguaje de programación muy extendido.
- La sintaxis deriva de c/c++ con la que el desarrollador del proyecto está muy familiarizado.
- Existen herramientas que posibilitarán el cumplimiento de todas las necesidades del proyecto

Desventajas:

- C# no es un lenguaje de programación orientado a múltiples plataformas (Aunque en el marco del proyecto Mono -Dot GNU ya se ha implementado un compilador para varias plataformas), que aun puede presentar ciertos

problemas de portabilidad.

- La experiencia del desarrollador del proyecto con esta plataforma es muy reducida y en proyectos testimoniales por lo que utilizar este lenguaje incrementaría la dificultad del desarrollo.

Conclusiones

Teniendo en cuenta todo lo que se ha explicado anteriormente y, teniendo en cuenta que es importante que la aplicación sea multiplataforma y que se puedan cumplir todos los requisitos funcionales y no funcionales se ha optado por seleccionar **Java** como lenguaje de programación a utilizar durante el desarrollo del producto.

Una vez tomada la decisión del lenguaje de programación, teniendo en cuenta las fechas de fin de soporte de las versiones disponibles de java (la versión 7 no cuenta con soporte ni actualizaciones desde abril de 2015) se ha decidido utilizar **java 8** [20] para el desarrollo del proyecto.

Teniendo en cuenta la decisión de utilizar java, se pretende realizar un empaquetado de la aplicación resultante incluyendo todas las dependencias en un fichero Jar que finalmente pueda ser ejecutado en cualquier máquina con **java 8** instalado. Este empaquetado se realizará con la herramienta **Maven.** [21]

Por otro lado, es importante señalar que se ha decidido utilizar un sistema control de versiones para el mantenimiento del repositorio de código fuente de la aplicación. Este sistema de control de versiones es **Git [22]** utilizando un repositorio en el servicio **GitHub [23].** Este sistema permitirá crear ramas en las que se puedan solucionar bugs o agregar nuevas features.

Finalmente, se debe señalar también que debido al uso de un sistema de control de versiones se ha decidido que es interesante incluir un servidor de **integración continua** [24] que compile, integre y mergée periódicamente el código de las diversas ramas de la aplicación. Dicho sistema será **Jenkins** [25], inicialmente desplegado en la plataforma de desarrollo pero que podría ser migrado a un servidor privado en internet.

3.7.2 Entorno de programación (IDE)

Una vez seleccionado **Java** como lenguaje de programación para el desarrollo del proyecto, se debe seleccionar un IDE [26] apropiado que reúna las cualidades suficientes para facilitar en la medida de lo posible la obtención de un software que cumpla con los requisitos del proyecto.

Las posibilidades que se han tenido en cuenta a la hora de encontrar una alternativa han sido Eclipse, NetBeans e Intellij IDEA

Eclipse

Es un IDE de código abierto multiplataforma desarrollado por la Fundación Eclipse

[27]

Ventajas:

- Es, posiblemente, el IDE mas extendido entre los desarrolladores en Java por lo que la comunidad con la que cuenta es muy importante
- Existe un gran número de plugins que permiten añadir funcionalidades al propio entorno.
- Es compatible con proyectos Maven

Desventajas:

- Gran consumo de recursos del sistema

NetBeans

Es un IDE para java gratuito y libre desarrollado inicialmente por SunMicrosistems (actualmente parte de Oracle) [28]

Ventajas:

- Es un IDE muy extendido por lo que cuenta con una gran comunidad (seguramente menor que la de Eclipse)
- Existe un gran número de plugins que permiten añadir funcionalidades al propio entorno.
- Es compatible con proyectos Maven
- Facilita el diseño de interfaces gráficas

Desventajas:

- Gran consumo de recursos del sistema

IntelliJ Idea

Es un IDE para java desarrollado por JetBrains disponible en dos versiones, una versión comercial y una versión Community (Gratuita) [29]

Ventajas:

- Es un IDE muy cohesionado debido a que ha sido desarrollado por una sola empresa.
- El consumo de recursos es mucho mejor que eclipse y netbeans
- Es compatible con proyectos Maven.
- Es la herramienta con la que el desarrollador del proyecto tiene mayor

experiencia por lo que se eliminan riesgos en el aprendizaje.

Desventajas:

- La comunidad es menor que la de eclipse o la de netbeans

Conclusiones

Una vez expuestas las ventajas y desventajas de cada uno de los IDEs tenidos en cuenta para el estudio, se puede ver claramente que no hay ninguna razón técnica que pudiera decantar la elección.

Es por esto que, teniendo en cuenta la experiencia del desarrollador, se ha decidido adoptar **Intellij Idea** como IDE a utilizar durante la fase de desarrollo del software del proyecto.

3.7.3 Plataforma de desarrollo

Atendiendo a razones de versatilidad y comodidad del desarrollador se ha decidido que el sistema operativo utilizado durante el desarrollo de la aplicación sea **Ubuntu Linux 15.10** debido, principalmente, a que todas las herramientas a utilizar durante el desarrollo son multiplataforma (como se pretende que sea el entregable final) y por tanto, están disponibles tanto para la plataforma de desarrollo como para la plataforma de explotación,

Por tanto, las características de la plataforma de desarrollo son las siguientes

Lenovo UltraBook ThinkPad X250	
Sistema Operativo	Ubuntu Linux 15.10 (Wily Werewolf)
CPU	Intel Core i7-5600U @ 2.60GHz
Cores	4
RAM	16 GB
HD	512 GB
Gráfica	Intel HD 5500
Pantalla	Full HD (1920 x 1080)

3.7.4 Plataforma de ejecución

Tal y como se ha especificado en los requisitos de la aplicación es muy importante **que la aplicación pueda ejecutarse en plataformas** windows, Linux y Mac de 64 bits sin necesidad de la instalación de ningún software a excepción de la máquina virtual de java 8.

3.7.5 Librerías de tratamiento de formato Xml

Tal y como se ha descrito en la documentación del presente proyecto, la aplicación utilizará un fichero Xml en el que se podrán especificar ciertos items de configuración que modificarán el comportamiento del software.

Debido a esto, y, aunque en realidad no es una tarea demasiado compleja por la simpleza del fichero de configuración, se han manejado ciertas alternativas para la elección de una librería para el manejo de ficheros con formato Xml y la validación utilizando un DTD. Dichas alternativas son Xerces2, JDOM y JAXP.

Xerces2

Es una colección de librerías para la manipulación de documentos XML creada por la Apache software foundation [30]

Ventajas:

- Soporta el estándar de parseo DOM, recomendado para documentos XML pequeños.
- Soporta validación DTD

Desventajas:

- Dado que el fichero Xml a parsear y validar no es demasiado grande no hay ninguna desventaja razonable en esta librería

JDOM

Es una librería de código abierto para manipulación de XML optimizado para JAVA. [31]

Ventajas:

- Soporta el estándar de parseo DOM, recomendado para documentos XML pequeños.
- Creado especialmente para usarse con Java y beneficiarse de sus características.
- Soporta Validación DTD
- Esta es una librería conocida por el desarrollador del proyecto.

Desventajas:

- Dado que el fichero Xml a parsear y validar no es demasiado grande no hay

ninguna desventaja razonable en esta librería

JAXP

Es un API de java para la manipulación y el procesamiento de Xml desarrollada bajo el Java Community Process. [32]

Ventajas:

- Soporta el estándar de parseo DOM, recomendado para documentos XML pequeños.
- Soporta Validación DTD [33]

Desventajas:

- Dado que el fichero Xml a parsear y validar no es demasiado grande no hay ninguna desventaja razonable en esta librería

Conclusiones

El parseo y validación de pequeños ficheros de configuración Xml no es una tarea que pueda producir especiales problemas ni para la que se puedan encontrar una gran diferencia de procesamiento en cada una de las alternativas.

De hecho, las tres alternativas mostradas en este punto incluyen el uso de DOM como método de parseo pero no ha sido posible encontrar una ventaja significativa a favor de ninguna de las las alternativas presentadas.

Dicho esto, se ha optado por incluir **jdom** como librería a utilizar para el manejo y validación de ficheros Xml utilizando como hecho diferencial la experiencia del desarrollador con dicha herramienta.

3.7.6 Librerías de tratamiento de formato Open Document

La primera restricción con la que hay que contar para la elaboración de esta comparativa es el lenguaje de programación utilizado, que como ya se ha explicado anteriormente es **Java.**

Según la propia web del open document format existen cierto número de alternativas para la el parseo de ficheros ODF (ODFDOM, Aspose.Words, Jodf, JopenDocument, JODReports, JODconverter, ODFXSLTRunner, MystiqueXML y ODF Easy).

Además, una última alternativa planteada es la posibilidad de realizar transformaciones XLST para extraer los elementos seleccionados del fichero ODF y que después puedan ser transformados a audio.

ODFDOM

Es un framework OpenDocument creado por la Apache Software Foundation con el objeto de manipular ficheros ODF sin tener un conocimiento extenso de la especificación. [34]

Ventajas:

- Permite realizar operaciones tanto con el paquete ODF (todos los recursos almacenados dentro del fichero) como con las características del documento en si.
- El acceso al contenido se realiza en dos niveles. Low level que da acceso a las características elementales del esquema y el High level que permite manipular el contenido escondiendo los detalles de implementación
- Permite manipular el documento a tratar sin necesidad de extraer el contenido almacenado en su interior.

Desventajas:

- No permite el acceso a todos los elementos del documento como podría ser el texto alternativo de una fórmula, etc.

Aspose.Words

Es una librería creada por aspose que permite procesar documentos de una gran variedad de tipos (Office, Epub, Pdf, OpenDocument, etc). [35]

Ventajas:

- Esta librería está diseñada para procesar una gran cantidad de tipos de documentos de ofimática.

Desventajas:

- No es software libre
- No se puede utilizar el api de manera gratuita en un entorno de explotación

Jodf

Es una api de java que permite la manipulación y el parseo de documentos y hojas de cálculo creada por Independentsoft. [36]

Ventajas:

- Permite extraer texto de los documentos de una manera muy simple.

Desventajas:

- Solamente soporta odt y ods.

- No es libre ni gratuita.

jOpenDocument

Es una librería open source java para la creación de documentos odf. [37]

Ventajas:

- Es una api muy simple.

Desventajas:

- Está orientada a la creación de documentos y no al manejo y extracción de datos de un fichero existente por lo que no se adapta a las necesidades del proyecto.

JODReports

Es una solución para crear informes en formato ODF basándose en plantillas creadas en OpenOffice. [38]

Ventajas:

- No hay ninguna ventaja reseñable en este producto

Desventajas:

- No se ajusta a ninguna de las necesidades que presenta el proyecto.

JODConverter

Esta herramienta creada por Art of solving permite convertir documentos OpenOffice a una gran cantidad de formatos. [39]

Ventajas:

- Es un conversor entre formatos muy versátil y que podría resultar útil en algunos casos concretos.

Desventajas:

- No se ajusta a ninguna de las necesidades que presenta el proyecto.

ODFXSLTRunner

Es una aplicación que permite aplicar transformaciones XSLT a los ficheros Xml que forman parte de un fichero ODF [40]

Ventajas:

- Permite hacer transformaciones Xslt a los documentos

Desventajas:

- Es una aplicación y no una librería por lo que no se puede integrar en la aplicación salvo usando llamadas a sistema.

MystiqueXML

Es un framework para la creación de ficheros ODF creado por Yacme. [41]

Ventajas:

- No existe ninguna ventaja aparente en este framework

Desventajas:

- Permite la creación de ficheros pero no la extracción de información

ODFEasy

Es una capa alternativa para el api ODFDOM. Pretende facilitar la creación de ODF. [42]

Ventajas:

- Está pensada para simplificar el api de ODFDOM

Desventajas:

- A pesar de ser muy versátil es habitual tener que recurrir al api de ODFDOM para realizar algunas tareas concretas.
- Está disponible inicialmente solo para documentos ods (Hojas de cálculo)

Transformaciones XSLT

Dado que el formato ODF se compone entre otras cosas de varios ficheros XML una solución propuesta para la extracción de datos sería la creación de documentos XSLT y, mediante el uso de una librería de Xml, ejecutar la transformación para obtener streams de texto plano que finalmente transformar en audio

Ventajas:

- No requiere ninguna librería extra

Desventajas:

- Requiere la creación de tantos Xslt como alternativas de ejecución basada en

la configuración pueda presentar la aplicación.

- Requiere la creación de ficheros Xslt para cada formato a convertir
- Requiere la descompresión de los ficheros almacenados dentro del documento a transformar.

Conclusiones

Dado el estado de desarrollo de todos los productos que se han tenido en cuenta para el desarrollo del proyecto es posible concluir que ninguno de ellos presenta una solución que se ajuste totalmente a las necesidades

Dicho esto, la elección final de una librería es la de **ODFDOM** que es la única que puede realizar un parseo de la mayoría de los elementos presentes en el documento proporcionado como entrada a la aplicación.

En caso de que ciertos elementos no se puedan parsear mediante **ODFDOM** podrán ser extraídos realizando un parseo al fichero Xml mediante el acceso al DOM del documento.

3.7.7 Librerías TTS

El siguiente paso en el proyecto es la selección de una librería que permita ejecutar el proceso de conversión del texto extraído del documento ODF a un formato de audio.

Para ello se cuenta con las siguientes librerías como alternativas: eSpeak, iSpeech, Java Speech API, servicio TTS de Google, FreeTTS y MaryTTS.

eSpeak

Es un sintetizador de voz de código abierto disponible para Linux y window que utiliza el método de síntesis de formantes. [43]

Ventajas:

- Está disponible como librería compartida por lo que puede ser usado en varios sistemas operativos
- Incluye diversos tipos de voces y puede incluir una gran cantidad de voces mediante MBROLA Diphone.
- Se pueden modificar las características de las voces incluidas

Desventajas:

- Necesita la instalación del propio software eSpeak para poder ser utilizado

iSpeech

Es un api TTS comercial que dispone de un SDK para varios lenguajes de programación. No se conocen demasiados detalles del proceso de conversión ni del método de síntesis. [44]

Ventajas:

- Es un servicio online
- Cuenta con una gran cantidad de voces

Desventajas:

- Es un servicio de pago para plataformas no móviles

Servicio TTS de Google

Aunque existió un API para la utilización de este servicio, actualmente ya no es posible utilizarlo debido a que Google decidió retirarlo. De cualquier manera, es posible realizar peticiones a una url de Google para simular que es un navegador web el que hace la petición [45]

Ventajas:

- Es un servicio online
- Cuenta con una gran cantidad de voces. Todas las disponibles en Google translate.
- La calidad de las voces es muy alta en cualquier idioma

Desventajas:

- Es un servicio que solo permite traducir 100 caracteres por petición.
- La salida se produce en formato MP3

Java Speech API

Es una API para java de síntesis y reconocimiento de voz que produce audio siguiendo el análisis de estructura, preprocesamiento de textos, conversión de texto a fonema, análisis de prosodia y producción de onda. [46]

Ventajas:

- Se le pueden añadir multitud de voces MBROLA

Desventajas:

- Es bastante compleja la creación de sintetizadores para producir ficheros de

audio.

Free TTS

Es una implementación de la API Java Speech API que simplifica algunas de las complejidades de la propia API. Las características técnicas en cuanto a tipo de procesamiento y análisis son las mismas que en Java Speech API. [47]

Ventajas:

- Se le pueden añadir multitud de voces MBROLA por lo que no es un problema la generación de audio en diversos idiomas.
- La creación de ficheros de audio a partir del texto extraído es una labor sencilla

Desventajas:

- Para utilizar Free TTS añadiendo nuevas voces se necesitaría la instalación de MBROLA en el sistema de explotación.

Conclusiones

Dado que no hay ningún sistema que se ajuste plenamente a las necesidades, se ha decidido adoptar dos de los sistemas descritos en el presente apartado.

Por un lado se ha seleccionado el servicio de **Google TTS** dado la gran cantidad de voces que presenta y que es un servicio online que podría ser muy versátil para textos pequeños.

Y por otro lado se ha decidido usar **eSpeakTTS** para la conversión offline debido a la posibilidad de añadir nuevas voces, a pesar de necesitar MBROLA, y a que con este sistema se pueden convertir textos de cualquier longitud.

Por tanto, tendremos un servicio que puede usarse tanto de manera online con **Google TTS** y que producirá ficheros **MP3** (que habrá que convertir a un formato libre) y otro servicio offline mediante **eSpeak** que producirá ficheros **WAV** (también convertibles a formato libre)

3.7.8 Librerías de formatos de audio

Llegados a este punto sabemos que las librerías TTS seleccionadas nos van a producir ficheros de audio **WAV** y ficheros de audio **MP3** por lo que necesitaremos una librería de audio que pueda convertir ambos a un formato de audio libre.

Las alternativas tenidas en cuenta durante el estudio de los formatos son Java Sound API, tritonus y jave

Java Sound Api

Es una API incluida en el JDK que permite el manejo y la conversión de ficheros de audio. Soporta AIFF, AU y WAV [48]

Ventajas:

- Está incluida en la propia distribución de java
- Produce audio en ficheros de audio libre como AIFF y AU

Desventajas:

- No tiene soporte para MP3 por lo que habría que que incluir algún conversor extra para poder utilizar esta librería.
- El formato de audio de salida disponible es un formato sin pérdida y por lo tanto ocupará mucho espacio en el disco duro.

Tritonus

Es una implementación del java sound api que permite la inclusión de diversos plugins para manejar algunos tipos extras de ficheros de audio produciendo salidas y aceptado entradas en multitud de formatos. [49]

Ventajas:

- Tiene una gran cantidad de formatos de audio disponibles.
- Produce audio en ficheros de audio libre distintos a los formatos sin perdida por lo que se podría ahorrar espacio en disco duro

Desventajas:

- Es una librería extra que habría que integrar en la aplicación

JAVE

Es un wrapper del proyecto ffmpeg que transcribe entre un gran númeor de formatos. [50]

Ventajas:

- Tiene una gran cantidad de formatos de audio disponibles.
- Produce audio en ficheros de audio libre distintos a los formatos sin perdida por lo que se podría ahorrar espacio en disco duro

Desventajas:

- No está disponible en casi ningún repositorio de maven

Conclusiones

Dado que no es posible obtener una audio que no sea en MP3 desde el servicio de Google es necesario descartar el uso del Java Api por si solo, por lo que la elección de una librería de audio recae en **jave** que permitirá generar ficheros **ogg**.

Capitulo 4 Diseño del sistema

4.1 Diseño de la interfaz del usuario

Tal y como se ha descrito en el presente documento, la aplicación que se pretende desarrollar presentará dos tipos de interacción diferentes:

- Interfaz gráfica de usuarios
- Interfaz por uso de comandos

Desde el punto de vista de la entrada de datos de la aplicación, solo se necesitan dos ficheros para que esta funcione correctamente:

- Documento odf a convertir.
- Fichero de configuración Xml.

Por un lado, la interfaz gráfica del usuario de la aplicación necesitará los siguientes elementos para permitir al usuario obtener información del estado de la aplicación así como una interacción fluida y eficaz:

- Campo de texto para escribir el path del documento Odf.
- Botón que al ser pulsado muestre un formulario donde seleccionar el documento Odf dentro del árbol de directorios.
- Formulario de selección del documento Odf.
- Campo de texto que mostrará el path del fichero de configuración (no es editable).
- Botón que al ser pulsado muestre un formulario donde seleccionar el fichero de configuración dentro del árbol de directorios.
- Formulario de selección del fichero de configuración.
- Botón de ejecución de la tarea.
- Checkbox “online processing” que permitirá al usuario seleccionar procesamiento online u offline
- Grupo de radioButtons Choose language en el que el usuario podrá seleccionar español, catalan o inglés como lenguaje del fichero.
- Grupo de radioButtons “Split-by strategy” en el que el usuario seleccionará si desea dividir el resultado en tantos ficheros como páginas tenga el documento inicial o si desea obtener todo el resultado en un solo fichero de audio.
- Ventana de diálogo en la que mostrar feedback al usuario (errores, warnings, etc)

Es necesario puntualizar que los campos “online Processing”, “Choose language” y “Split-by Strategy” se encuentran deshabilitados en la interfaz hasta que se seleccione un fichero de configuración. Una vez ocurra esto, se habilitarán dichos campos conteniendo los valores del fichero xml.

Finalmente, la interfaz en modo linea de comandos de la aplicación será la facilitada por cada uno de los diversos sistemas operativos en los que pueda funcionar y hará uso de dos parámetros mutuamente incluyentes –config y –doc. El primero de dichos parámetros indicará el path del fichero de configuración y el segundo de ellos el path del fichero a convertir.

Como ya se ha señalado, dichos parámetros son mutuamente incluyentes por lo que el uso de uno de ellos implica que hay que usar el segundo. En caso contrario se producirá un error y se saldrá del programa.

Si ninguno de los dos parámetros es utilizado se ejecutará la aplicación en modo gráfico.

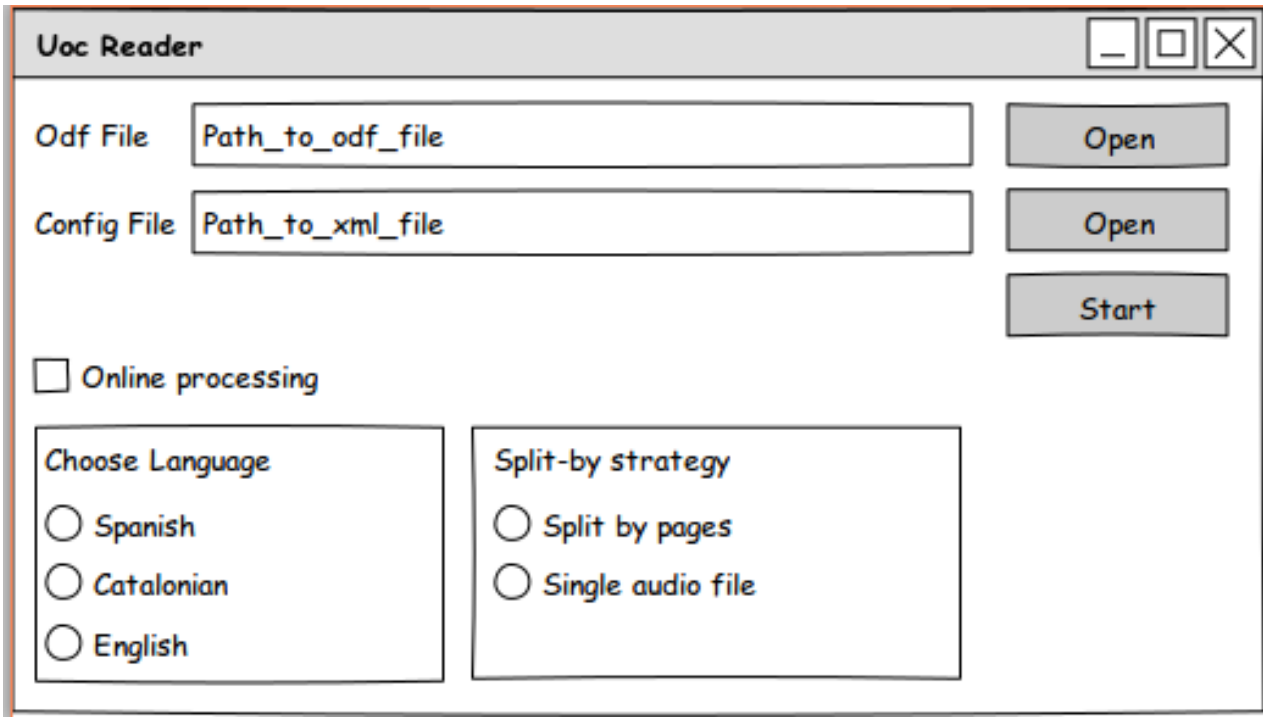
4.1.1 Diseño de los wireframes

01.Ventana principal de la aplicación:

En la ventana principal de la aplicación el usuario dispondrá de dos campos de texto en los que podrá introducir manualmente los paths del fichero de configuración y el fichero ODF a convertir.

Alternativamente tendrá habilitados dos botones para seleccionar dichos ficheros en el árbol de directorios del ordenador en el que se esté ejecutando la aplicación,

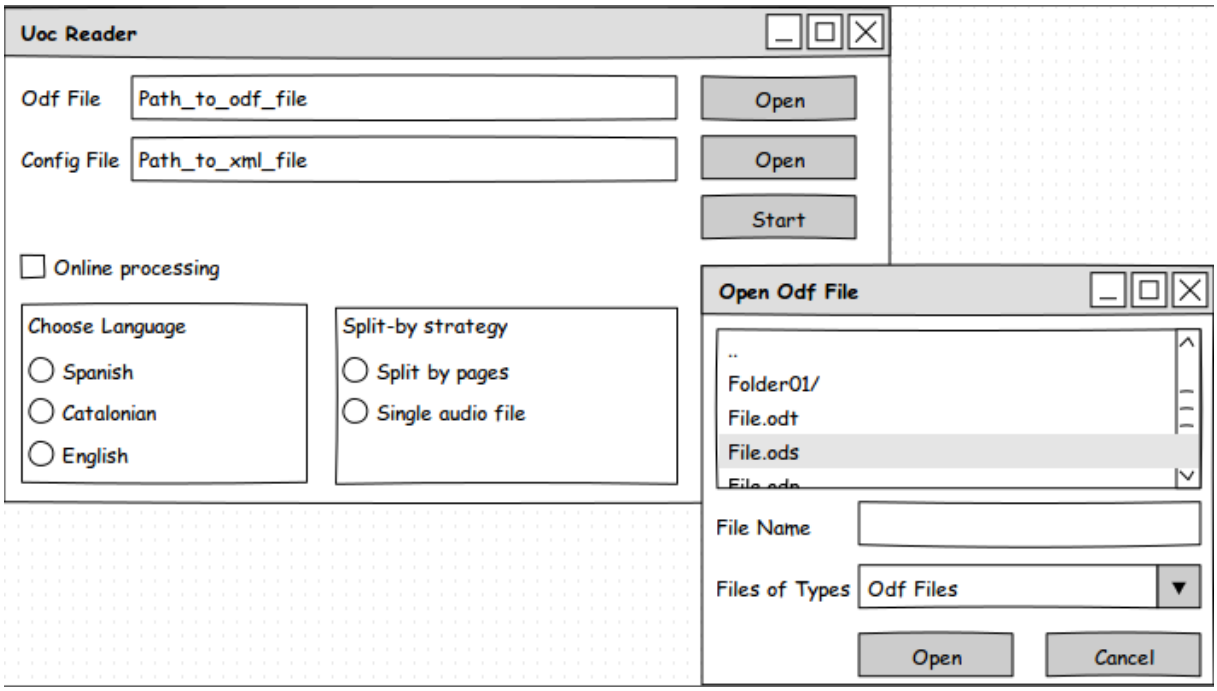
También se muestra un botón **Start** que lanza el proceso.



02.Selección de fichero ODF

Al pulsar el botón Open relativo al fichero ODF se mostrará un menú desplegable en el que el usuario podrá seleccionar el fichero ODF disponible en cualquier directorio del ordenador que ejecute la aplicación.

En caso de seleccionar un fichero y pulsar **open**, se cerrará el desplegable y el campo Odf File contendrá el path completo del fichero a convertir.



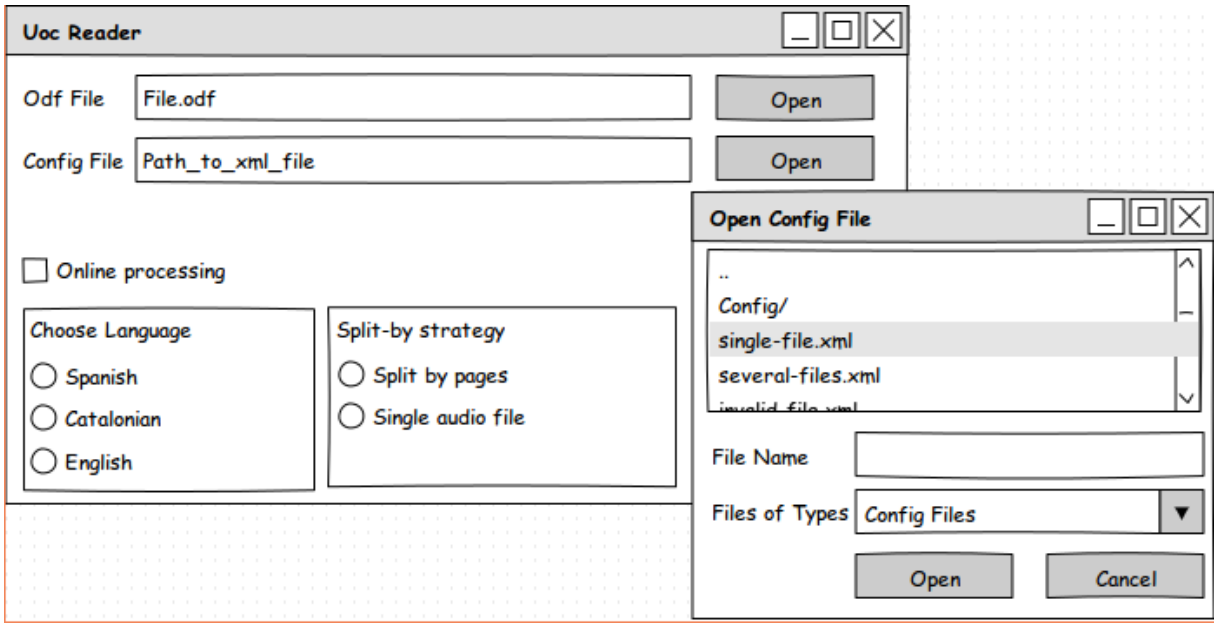
En caso de que el usuario decida pulsar el botón **cancel**, se volverá a la pantalla principal de la aplicación y no se registrará ningún cambio.

03.Selección de fichero de configuración

Al pulsar el botón Open relativo al fichero de configuración se mostrará un menú desplegable en el que el usuario podrá seleccionar el fichero **XML** disponible en cualquier directorio del ordenador que ejecute la aplicación.

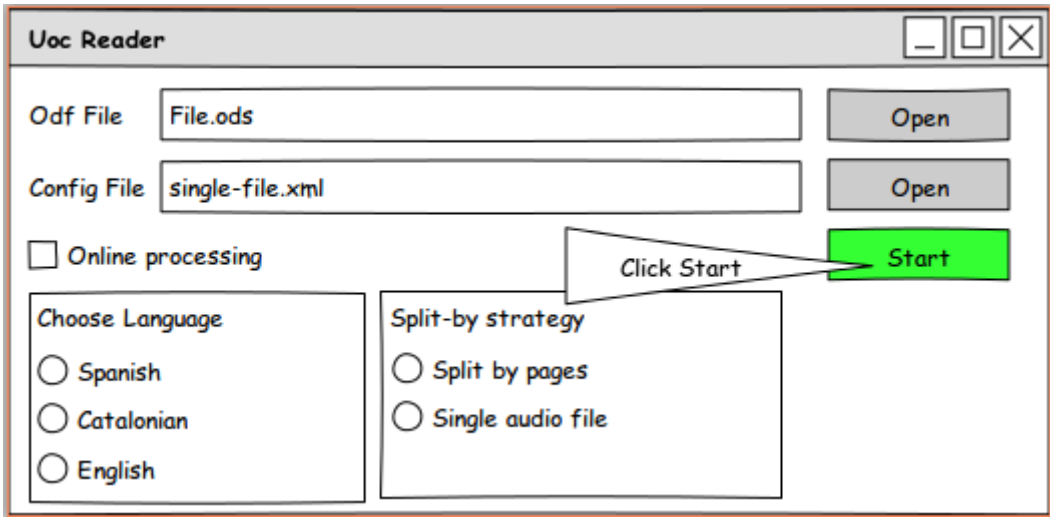
En caso de seleccionar un fichero y pulsar **open**, se cerrará el desplegable y el campo **Config File** contendrá el path completo del fichero de configuración.

En caso de que el usuario decida pulsar el botón **cancel**, se volverá a la pantalla principal de la aplicación y no se registrará ningún cambio.



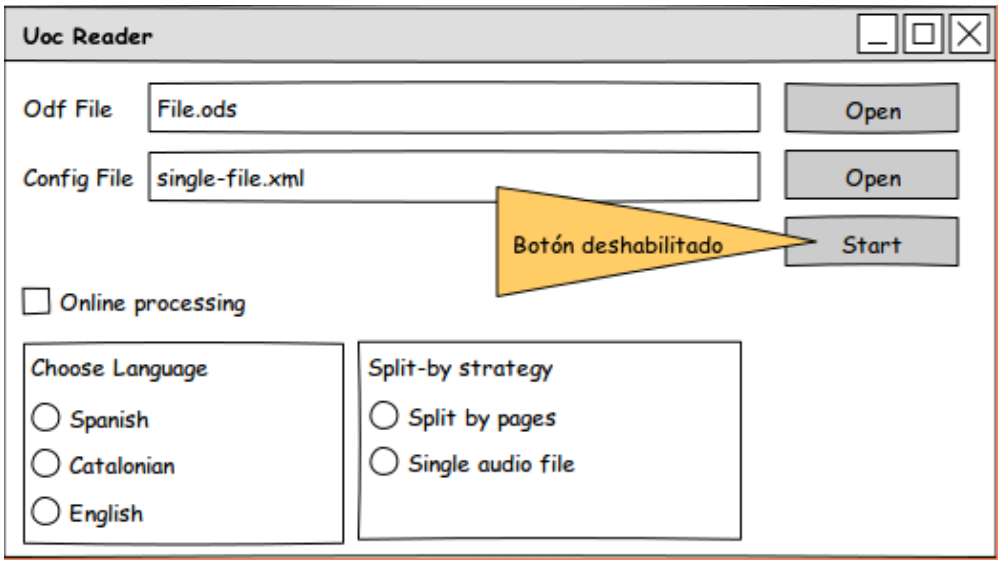
04.Inicio del proceso

Tal y como se ha descrito anteriormente, la pantalla principal contiene un botón Start (Botón de inicio de proceso). Al pulsar dicho botón, el usuario ejecutará el proceso de conversión que implica la lectura de ambos ficheros pudiendo resultar este proceso en éxito o en fallo, dependiendo del contenido de los ficheros o de otros factores del ordenador que ejecuta el programa



05.Progreso de conversión

Una vez ejecutado el proceso de conversión en la pantalla anterior, se deshabilitará el botón start hasta que el proceso de conversión haya terminado independientemente del resultado final.



Tras finalizar la ejecución se pueden dar dos casos:

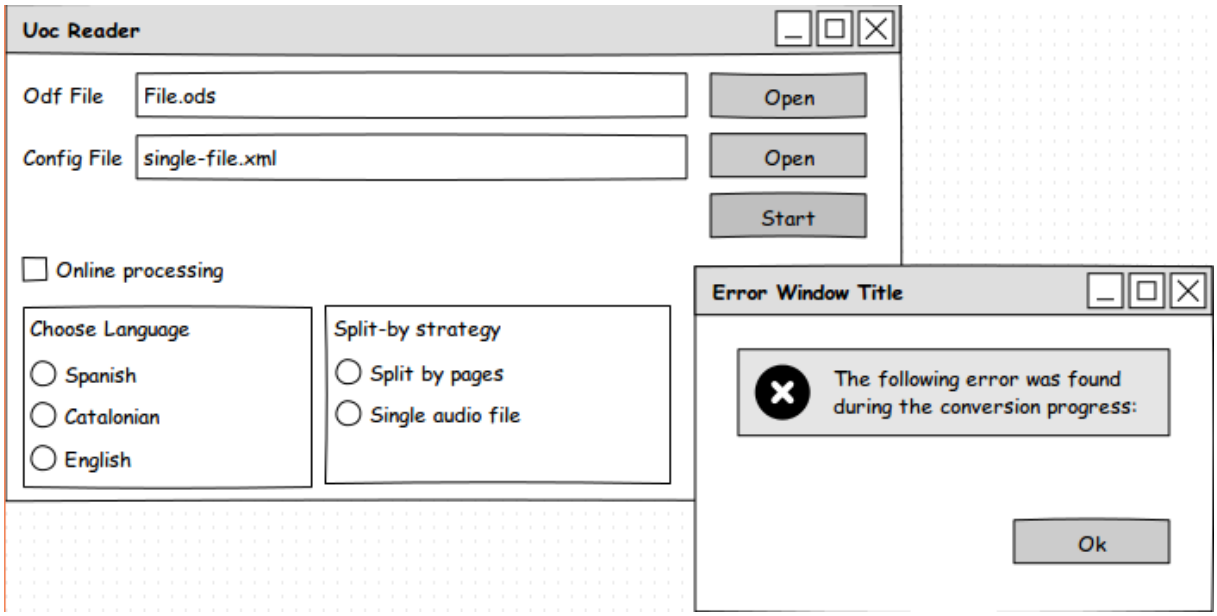
- Éxito en la ejecución del proceso (07. Éxito en el progreso)

- Error en la ejecución del proceso (06. Error en el progreso)

06.Error en el progreso

En el caso en el que el progreso haya ido mal se mostrará al usuario una ventana con un mensaje de error que contendrá información suficiente sobre el problema de modo que el usuario pueda solucionar el error e intentarlo de nuevo.

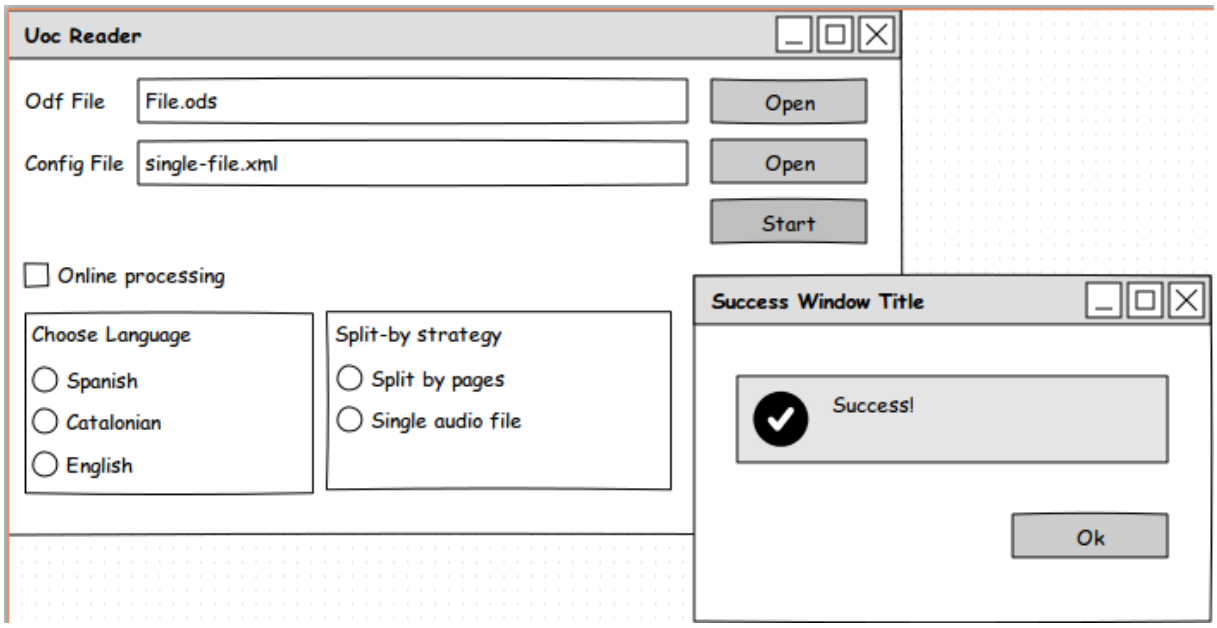
Ejemplos de estos problemas pueden ser errores en la configuración del fichero xml de configuración, ficheros ODF con un formato no aceptado, problemas de espacio en disco, problemas con el conversor o cualquier otra excepción lanzada por la aplicación



Al cerrar el mensaje se mostrará de nuevo la ventana principal de la aplicación con la configuración de la última ejecución (es decir, la que ha causado el problema)

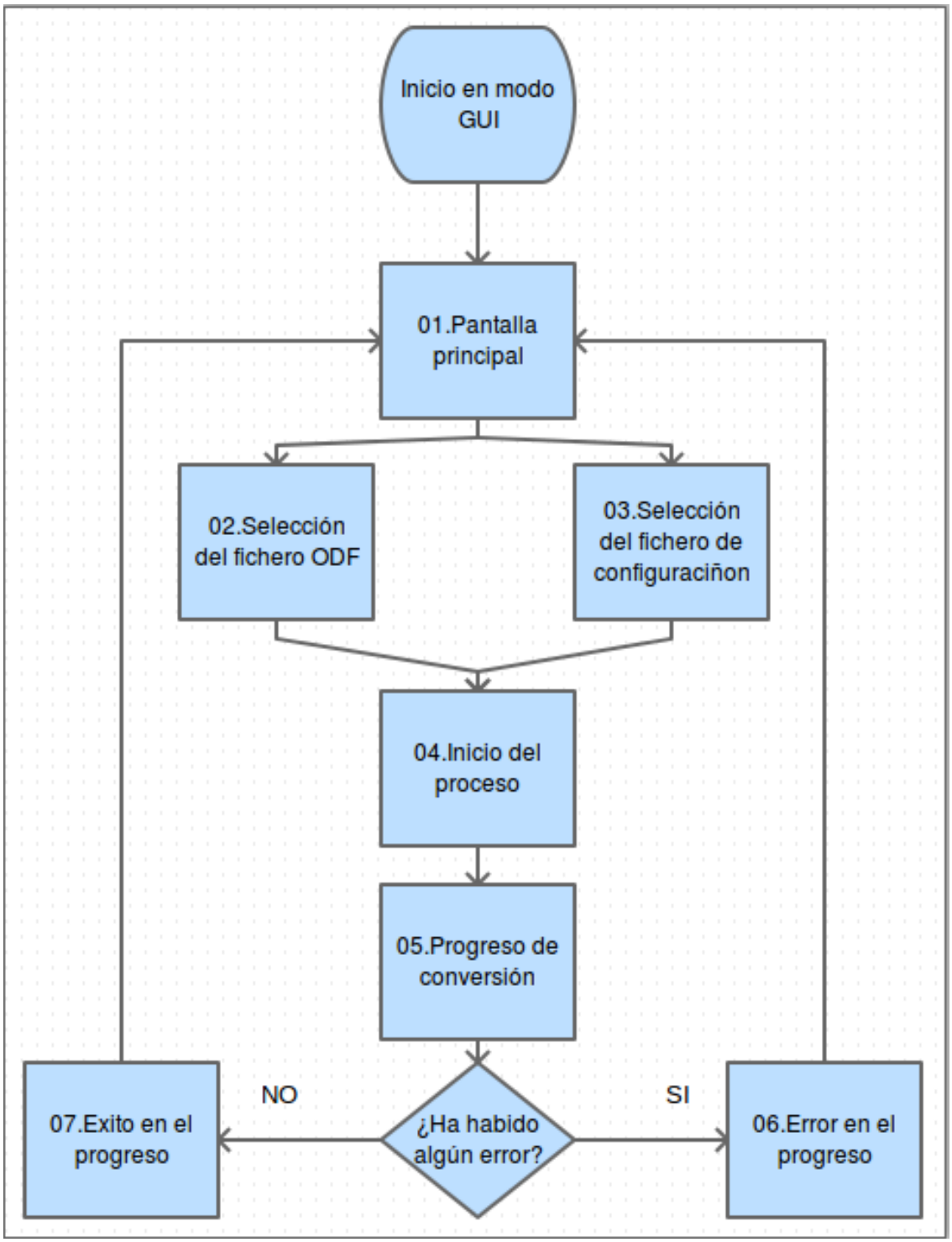
07.Éxito en el progreso

Finalmente, cuando el proceso ha concluido con éxito se mostrará al usuario un mensaje de éxito. Al cerrar este mensaje se volverá a la pantalla principal que incluirá la configuración ya seleccionada durante el proceso.



4.1.2 Flujo de navegación

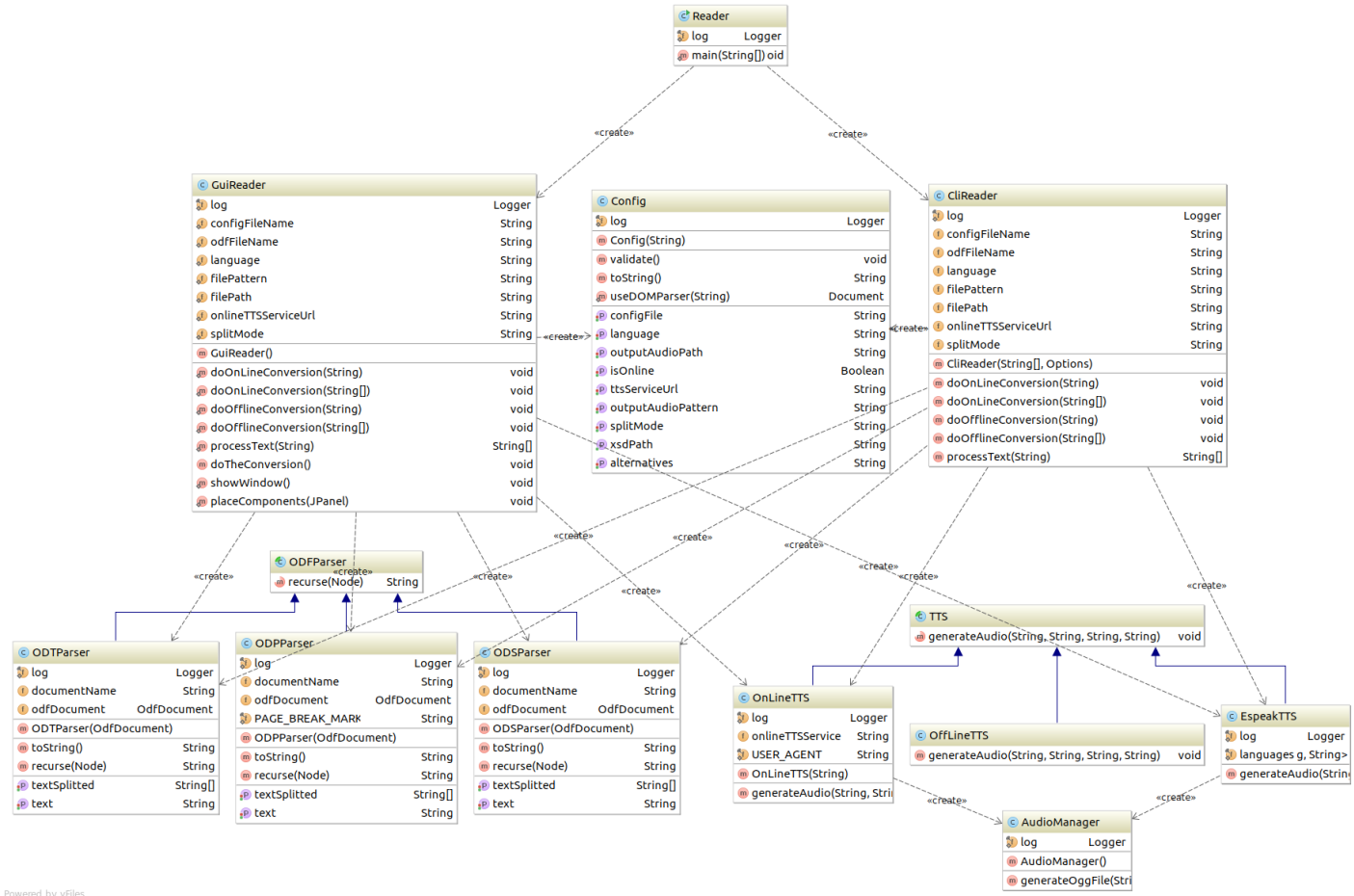
El siguiente esquema recoge el proceso de navegación del usuario por la interfaz gráfica de la aplicación.



4.2 Diseño del sistema

4.2.1 Diagrama de clases del sistema

Teniendo en cuenta todos los requisitos y las prioridades definidas durante la presente memoria del proyecto, el diagrama del clases del sistema es el siguiente:



Una vez se han mostrado relacionadas entre si cada una de las clases del sistema, se definirá cada una de ellas.

Reader

Es la clase principal y el punto de entrada de la aplicación. Esta clase llama a métodos de la clase GuiReader o CliReader dependiendo de los parámetros que se hayan pasado en la ejecución del programa.

GuiReader

Esta clase es la encargada de gestionar la ejecución de la aplicación en modo interfaz gráfica de usuario. Desde ella se invocan metodos de la factoria ODFParser y de la factoría TTS que extraen y convierten el texto a ficheros de audio.

CliReader

Esta clase es la encargada de gestionar la ejecución de la aplicación en modo linea de comando. Al igual que la clase anterior invoca métodos de la factoría ODFParser y de la factoría TTS que extraen y convierten el texto a ficheros de audio.

ODFParser

Es una factoría cuya funcionalidad principal es proporcionar una clase abstracta para el parseo de ficheros ODF. Basándose en esta clase abstracta se han creado tres clases concretas ODTParser (para parsear documentos de texto), ODSParser (para parsear hojas de cálculo) y ODPParser (para parsear presentaciones). La ventaja del uso del patrón factoria en este caso es que se podrán añadir nuevas clases sin tener que modificar clases de parseo existentes por lo que se cumple el principio SOLID “ABIERTO/CERRADO” de desarrollo por el que una entidad software debe estar abierta para extenderse pero cerrada para modificarse.

ODTParser

Ees la clase dedicada a parsear documentos de texto que extiende de la clase ODFParser. Contiene una implementación concreta para extraer el texto de este tipo de documentos.

ODSParser

Es la clase dedicada a parsear hojas de cálculo que extiende de la clase ODFParser. Contiene una implementación concreta para extraer el texto de este tipo de documentos.

ODPParser

Es la clase dedicada a parsear presentaciones que extiende de la clase ODFParser. Contiene una implementación concreta para extraer el texto de este tipo de documentos.

TTS

Es una factoría cuya funcionalidad principal es proporcionar una clase abstracta para la producción de audio a partir de texto. Basándose en esta clase abstracta se han creado tres clases concretas OnLineTTS (que utiliza el servicio TTS de google), OffLineTTS (Que utiliza la librería FreeTTS. Se encuentra en desuso debido a limitaciones de extensión de voces) y EspeakTTS (que genera ficheros de audio utilizando el ejecutable espeak tts). La ventaja del uso del patrón factoría en este caso es que se podrán añadir nuevas clases sin tener que modificar clases de

parseo existentes por lo que se cumple el principio SOLID “ABIERTO/CERRADO” de desarrollo por el que una entidad software debe estar abierta para extenderse pero cerrada para modificarse.

OnLineTTS

Es la clase dedicada a producir ficheros de audio que extiende de la clase TTS y que utiliza el servicio TTS De google. Contiene una implementación concreta para generar ficheros de audio en MP3. Utiliza la clase AudioManager para generar ficheros de audio libre en formato OGG.

OffLineTTS

Es la clase dedicada a producir ficheros de audio que extiende de la clase TTS y que utiliza la librería FreeTTS. Contiene una implementación concreta para generar ficheros de audio en WAV. Debido a limitaciones de la librería se ha decidido abandonar y utilizar otra solución.

EspeakTTS

Es la clase dedicada a producir ficheros de audio que extiende de la clase TTS y que utiliza el ejecutable de Espeak. Contiene una implementación concreta para generar ficheros de audio en WAV. Utiliza la clase AudioManager para generar ficheros de audio libre en formato OGG.

AudioManager

Esta clase gestiona la conversión de ficheros de audio de cualquier tipo a ficheros del tipo OGG.

4.3 Modelo dinámico

4.3.1 Diagramas de interacción

Diagrama de interacción 01. Conversión OffLine de fichero de texto en modo linea de comandos:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	Línea de comandos
Servicio TTS	Offline
Formato	ODT (Texto)

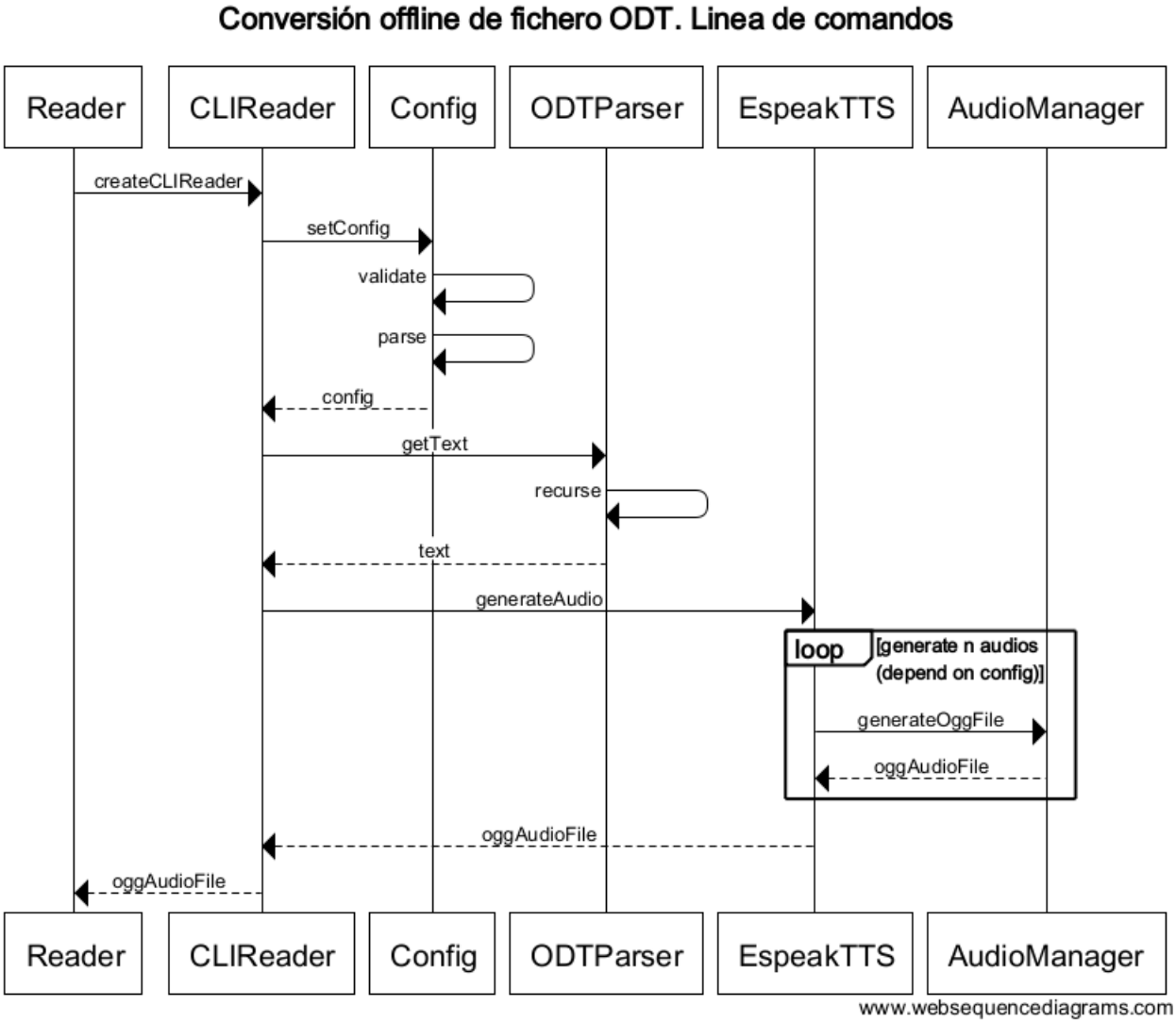


Diagrama de interacción 02. Conversión OnLine de fichero de texto en modo línea de comandos:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	Línea de comandos
Servicio TTS	Online
Formato	ODT (Texto)

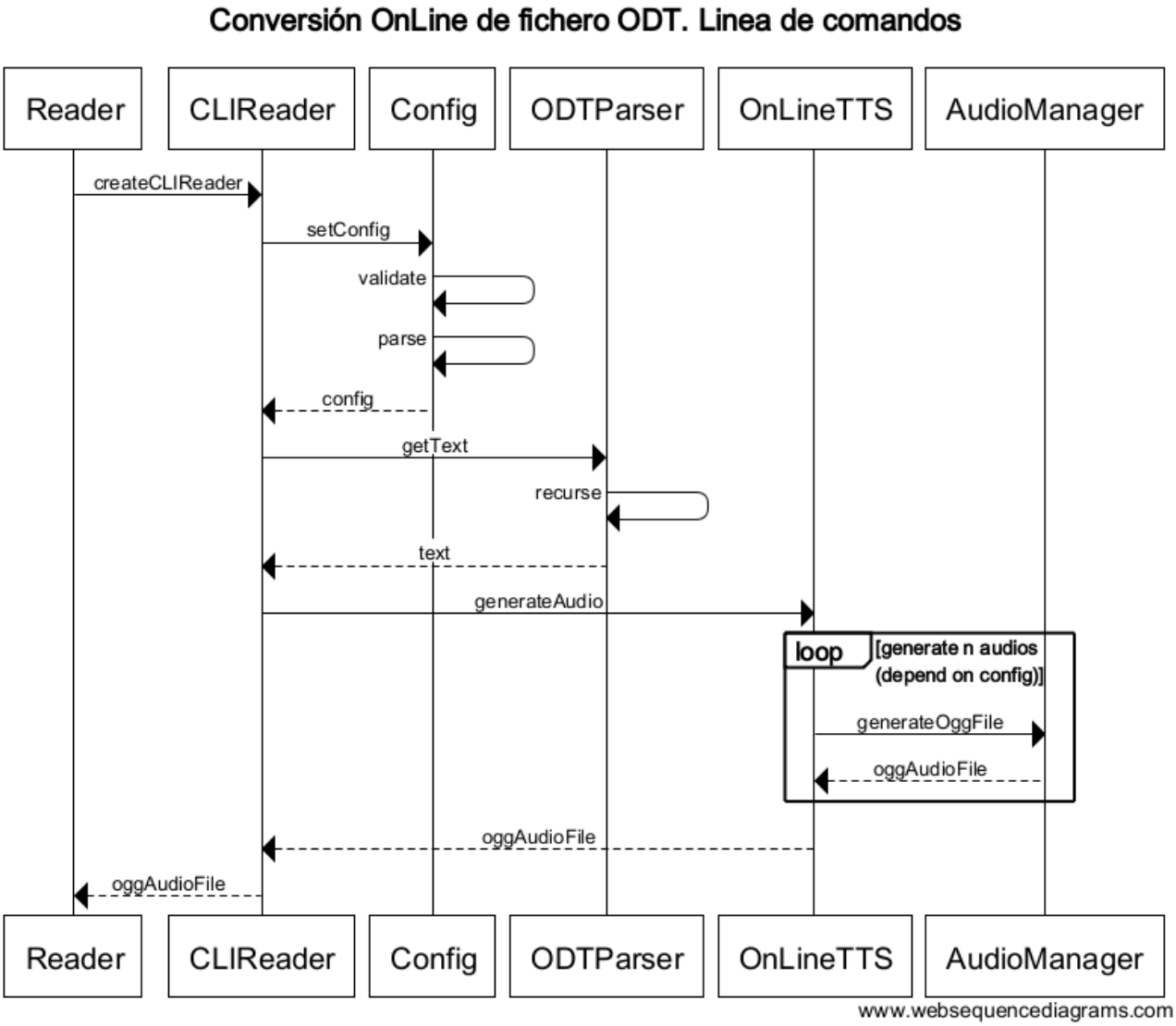


Diagrama de interacción 03. Conversión OffLine de fichero de texto en modo interfaz gráfica de usuario:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	GUI
Servicio TTS	Offline
Formato	ODT (Texto)

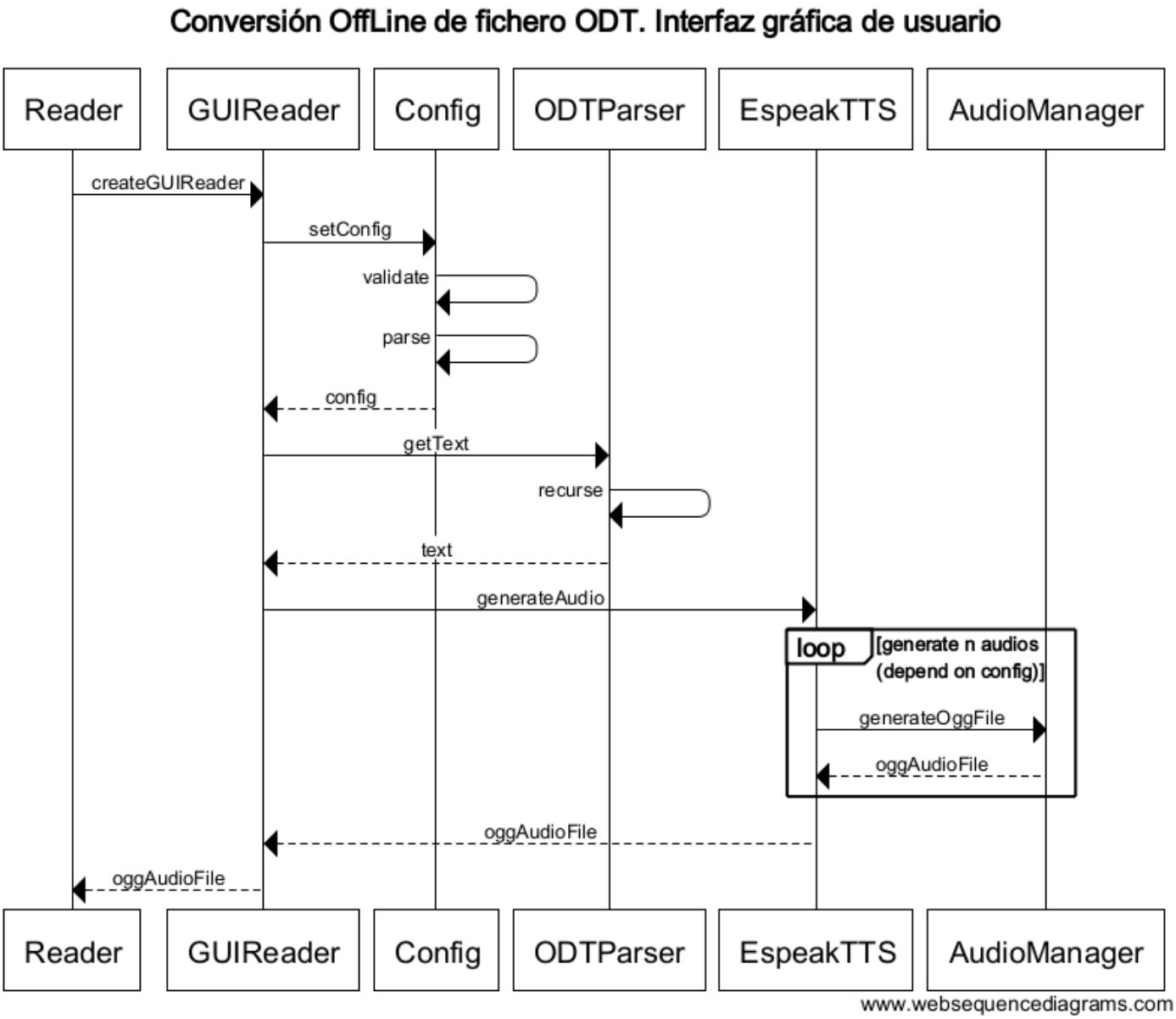


Diagrama de interacción 04. Conversión OnLine de fichero de texto en modo interfaz gráfica de usuario:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	GUI
Servicio TTS	Online

Formato	ODT (Texto)

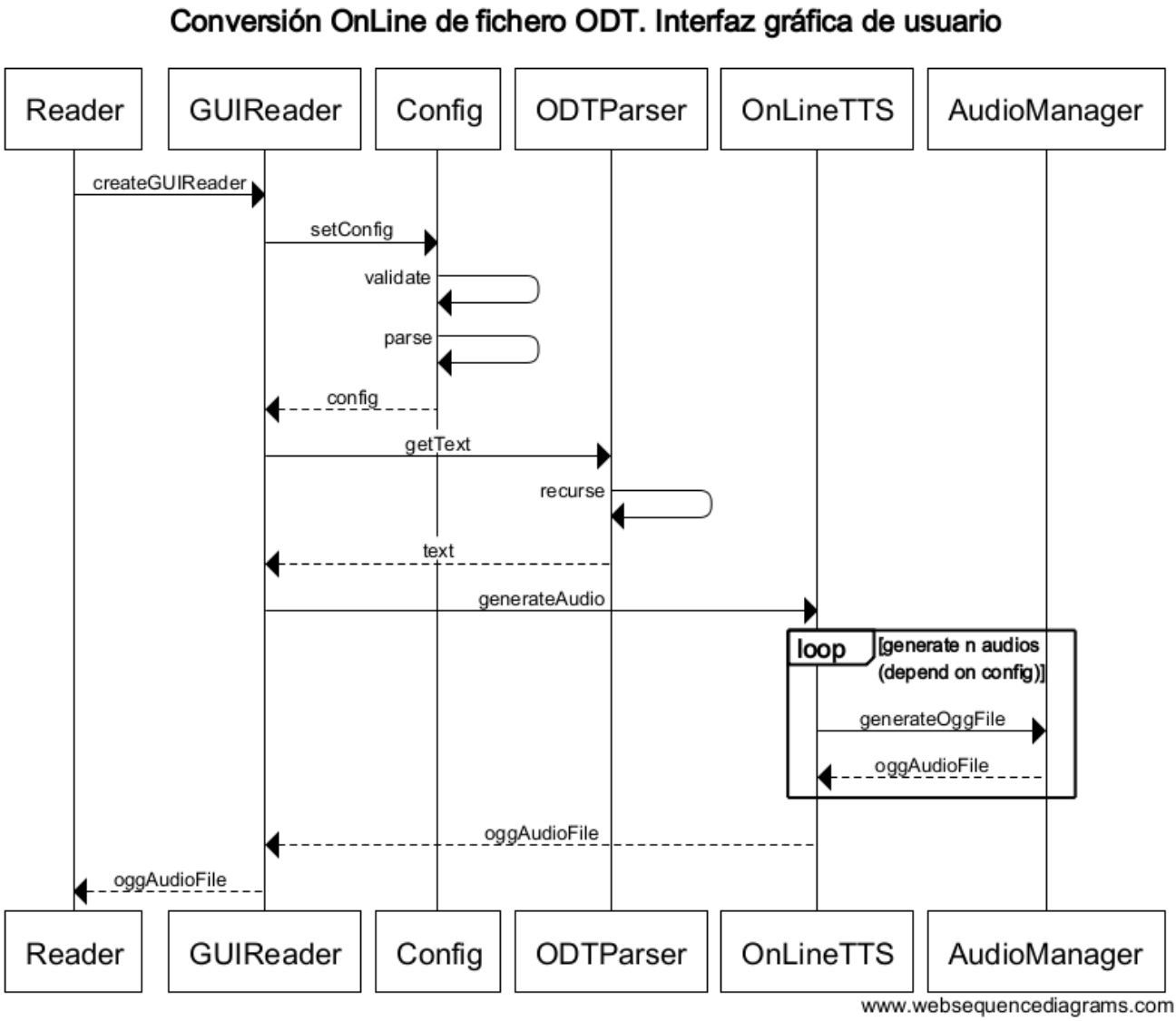


Diagrama de interacción 05. Conversión OffLine de hojas de cálculo en modo linea de comandos:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	Linea de comandos
Servicio TTS	Offline
Formato	ODS (Hojas de cálculo)

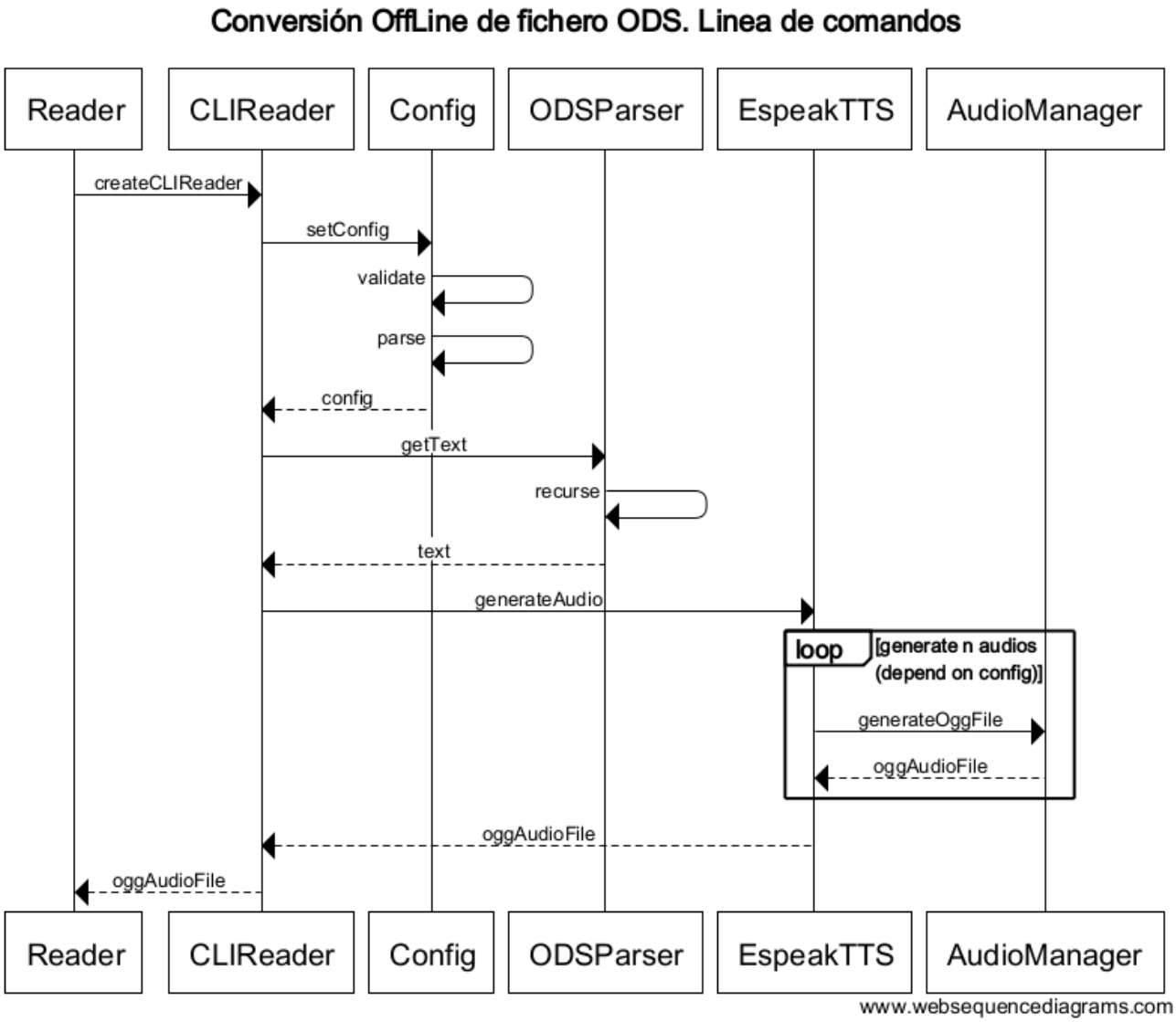


Diagrama de interacción 06. Conversión OnLine de hojas de cálculo en modo linea de comandos:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el

siguiente modo:

Modo	Línea de comandos
Servicio TTS	Online
Formato	ODS (hojas de calculo)

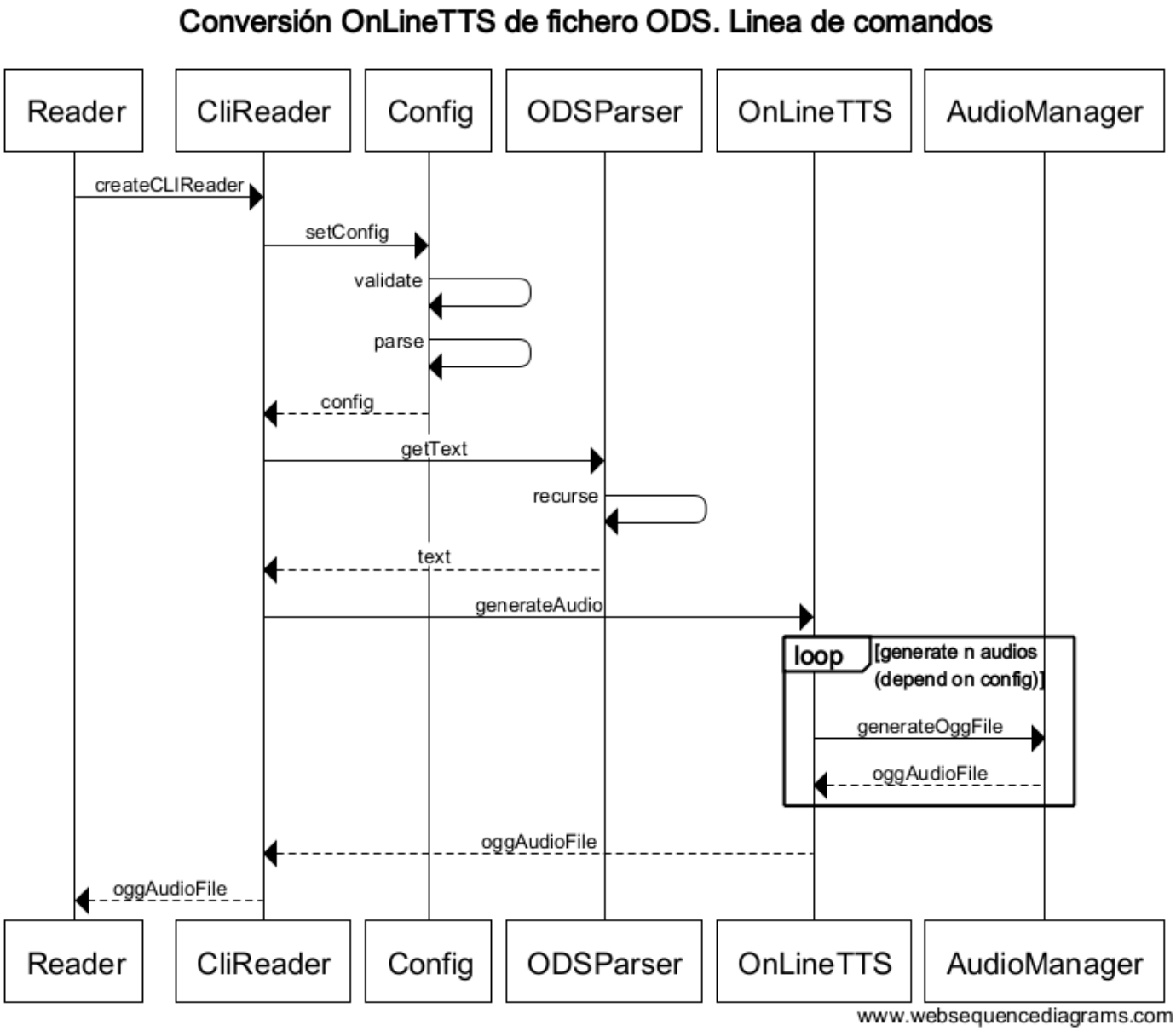


Diagrama de interacción 07. Conversión OffLine de hojas de cálculo en modo interfaz gráfico de usuario:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	GUI
Servicio TTS	Offline
Formato	ODS (hojas de calculo)

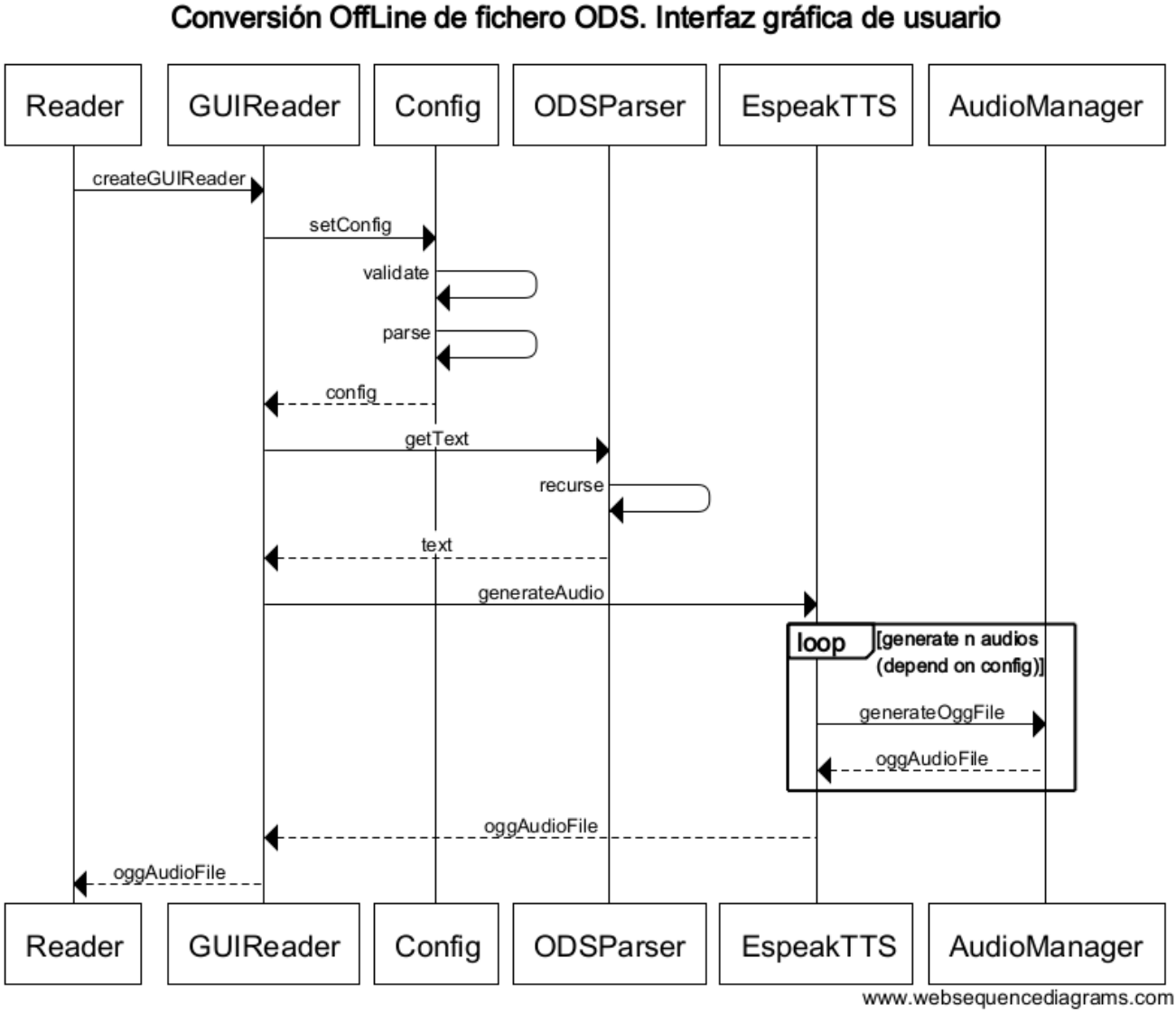


Diagrama de interacción 08. Conversión OnLine de hojas de cálculo en modo interfaz gráfico de usuario:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	GUI
Servicio TTS	Online
Formato	ODS (hojas de calculo)

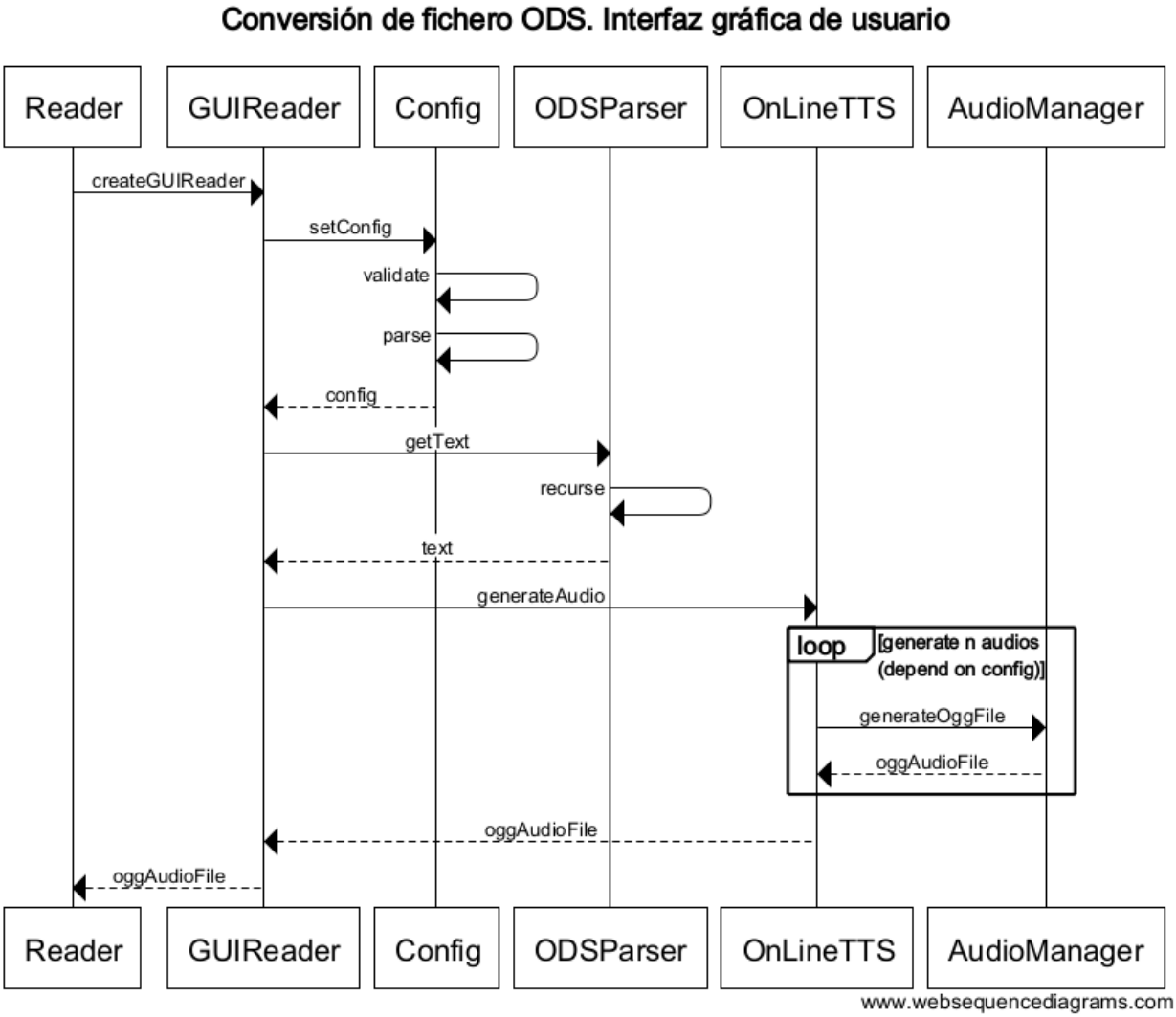


Diagrama de interacción 09. Conversión OffLine de presentación en modo linea de comandos:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	Línea de comandos
Servicio TTS	Offline
Formato	ODP (Presentación)

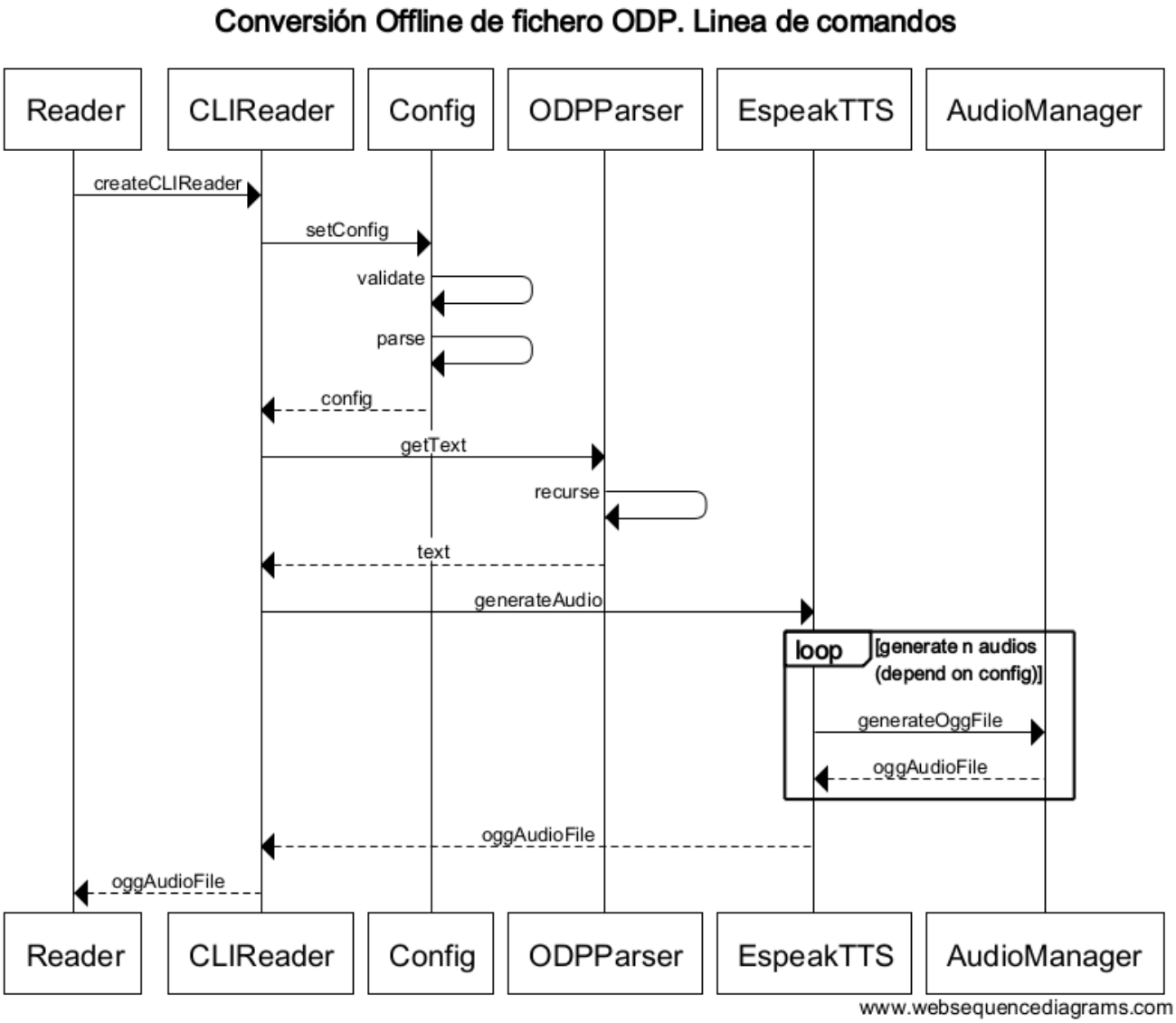


Diagrama de interacción 10. Conversión OnLine de presentación en modo linea de comandos:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	Línea de comandos
Servicio TTS	Online
Formato	ODP (Presentación)

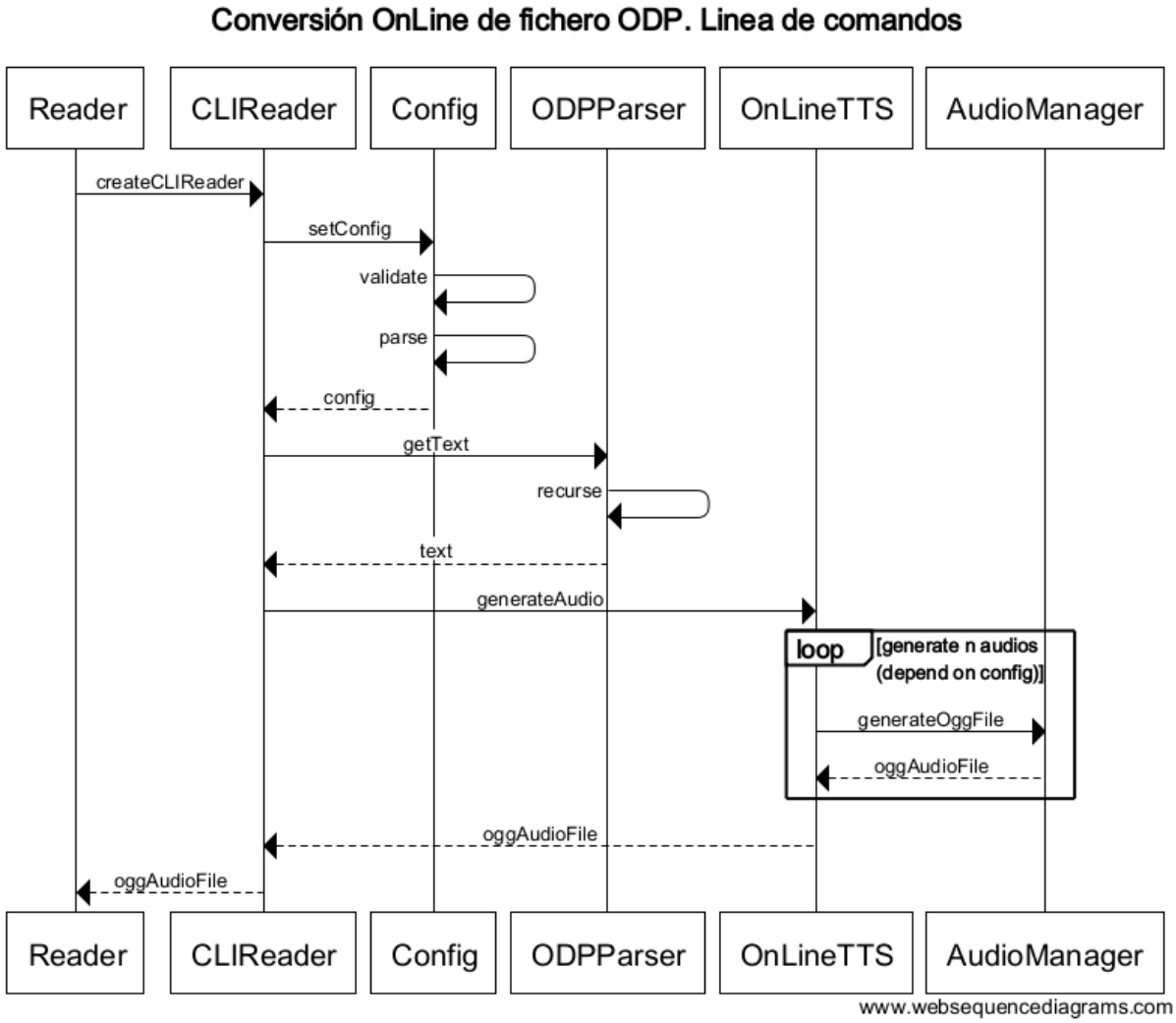


Diagrama de interacción 11. Conversión OffLine de presentación en modo interfaz gráfica de usuario:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	GUI
Servicio TTS	Offline
Formato	ODP (Presentación)

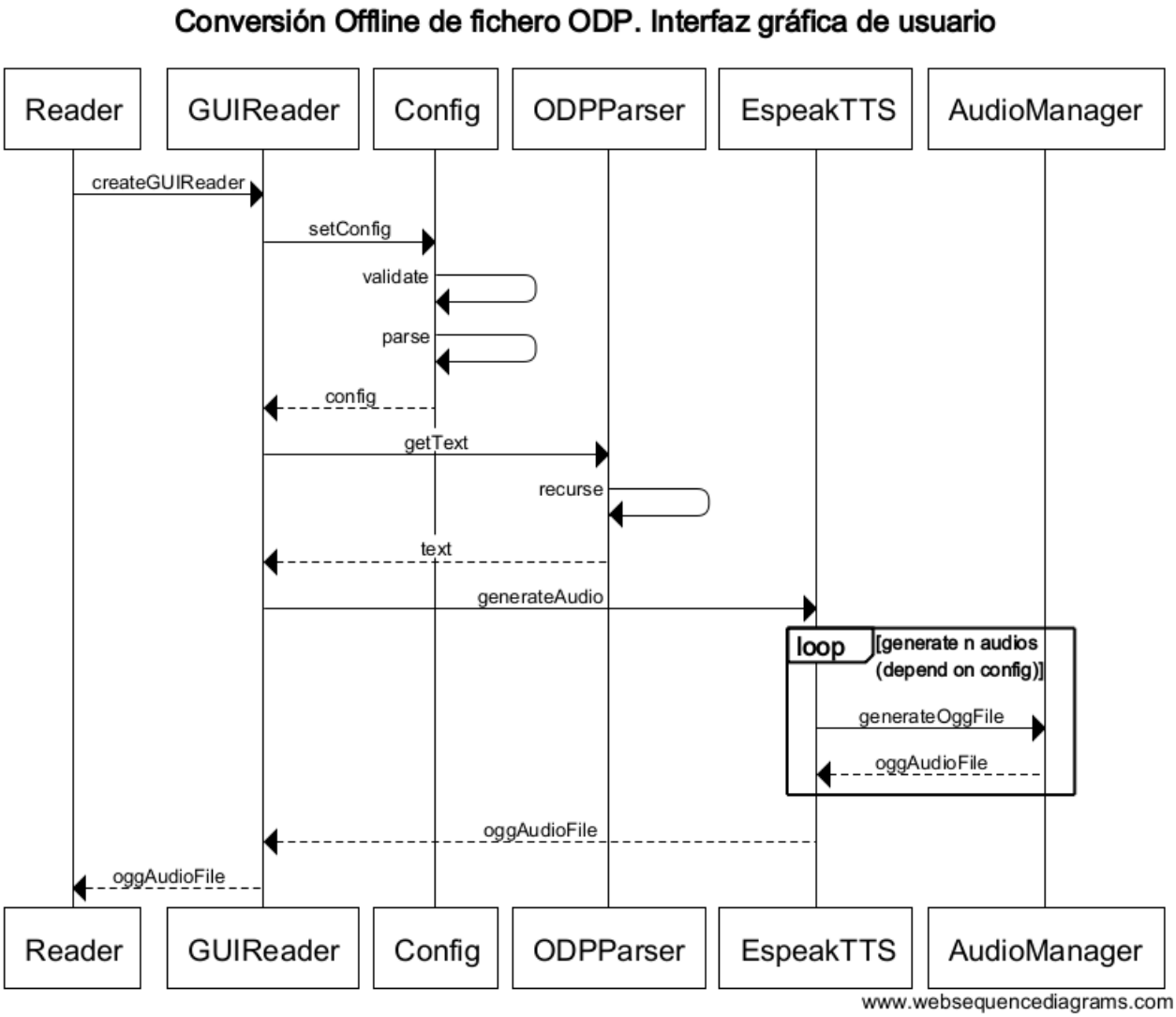


Diagrama de interacción 12. Conversión OnLine de presentación en modo interfaz gráfica de usuarios:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	GUI
Servicio TTS	Online

Formato	ODP (Presentación)

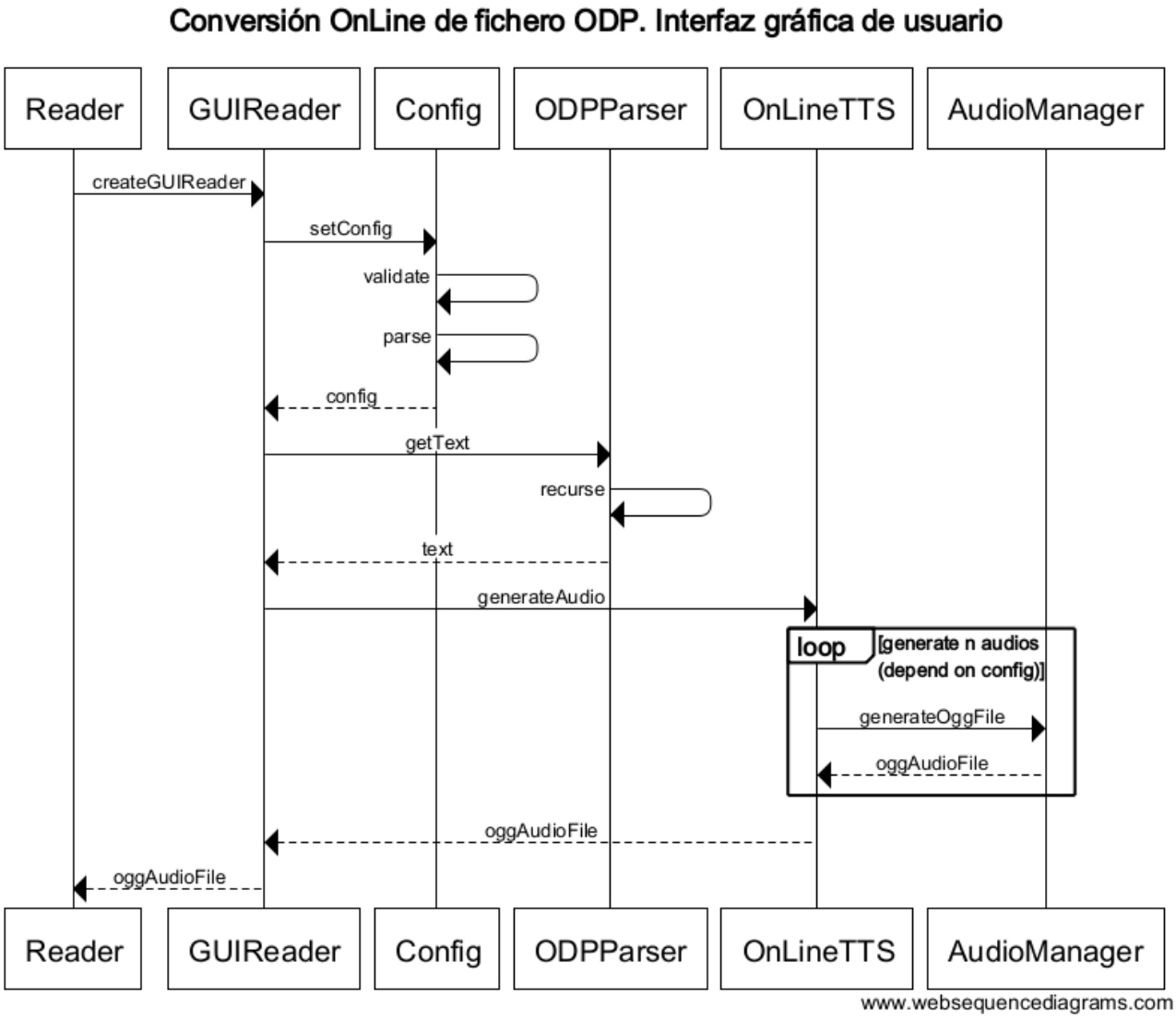


Diagrama de interacción 13. Configuración inválida en modo linea de comandos:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	Línea de comandos
Servicio TTS	Offline/Online
Formato	Indiferente
Configuración	Inválida

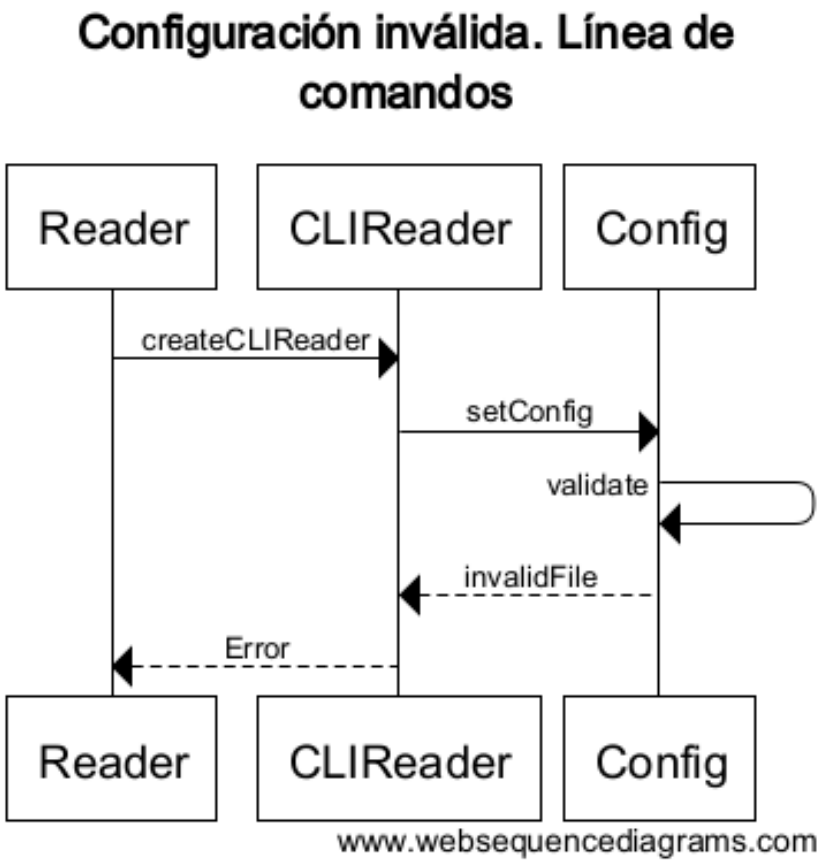


Diagrama de interacción 14. Configuración inválida en modo interfaz de usuario:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	GUI
Servicio TTS	Offline/Online
Formato	Indiferente
Configuración	Inválida

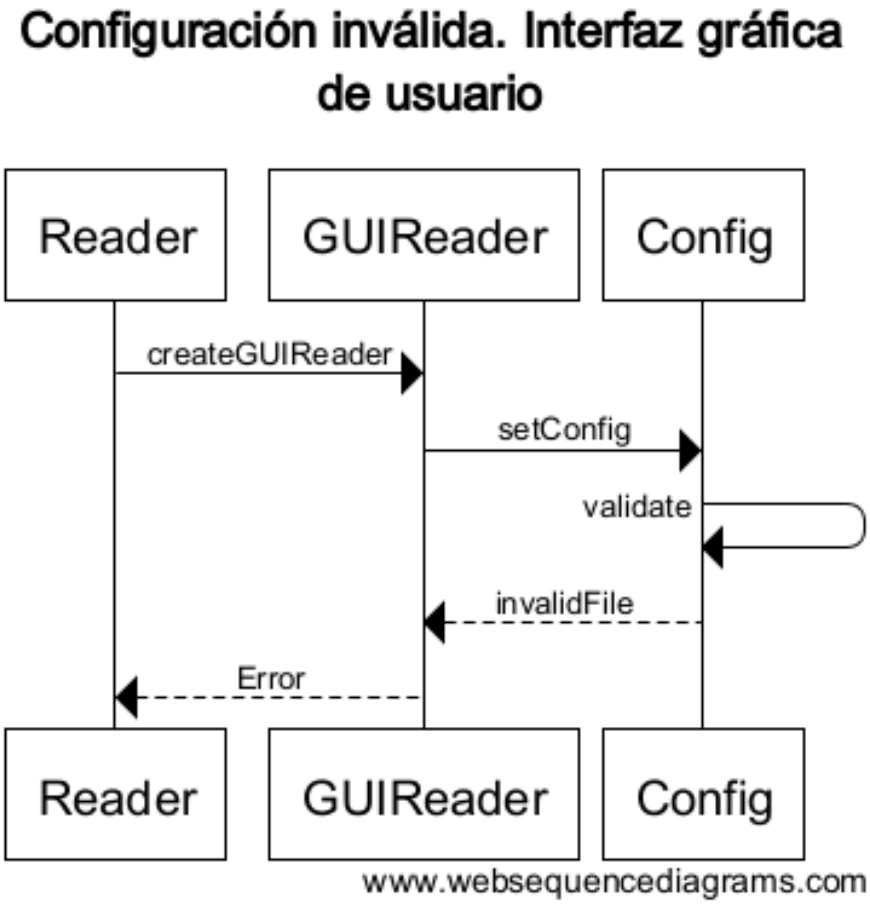


Diagrama de interacción 15. Formato no soportado en modo linea de comandos:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	Línea de comandos
Servicio TTS	Offline/Online
Formato	Inválido

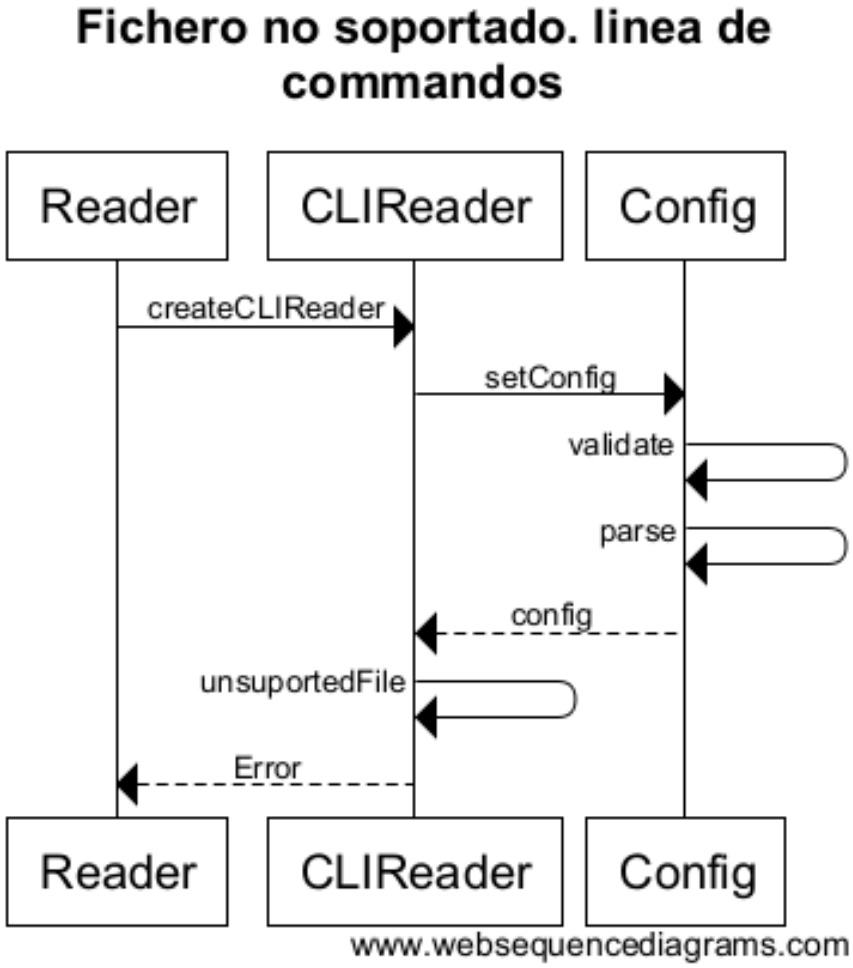
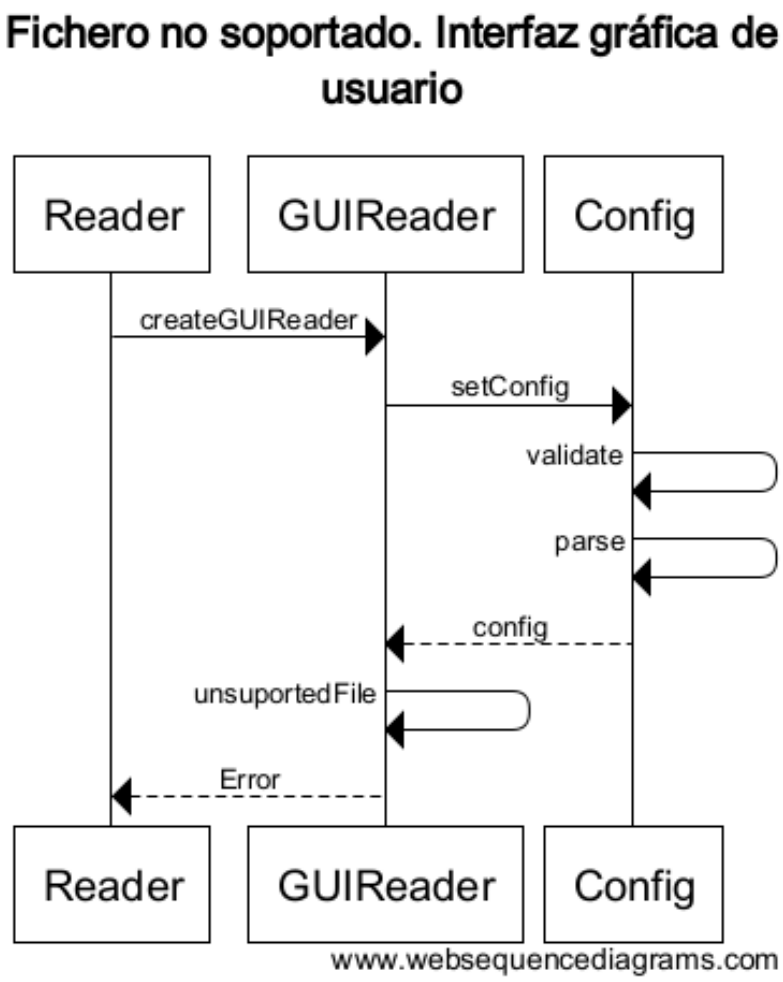


Diagrama de interacción 16. Formato no soportado en modo interfaz de usuario:

Este es el diagrama de interacción del sistema en la ejecución teniendo en cuenta el siguiente modo:

Modo	GUI
Servicio TTS	Offline/Online
Formato	Inválido



4.3.2 Diagramas de actividades

En este apartado se muestra el proceso de trabajo de la aplicación mediante una serie de tareas realizadas por el software tras la interacción del usuario en ambos modos de funcionamiento (GUI y Línea de comandos)

Diagrama de actividades en modo GUI.

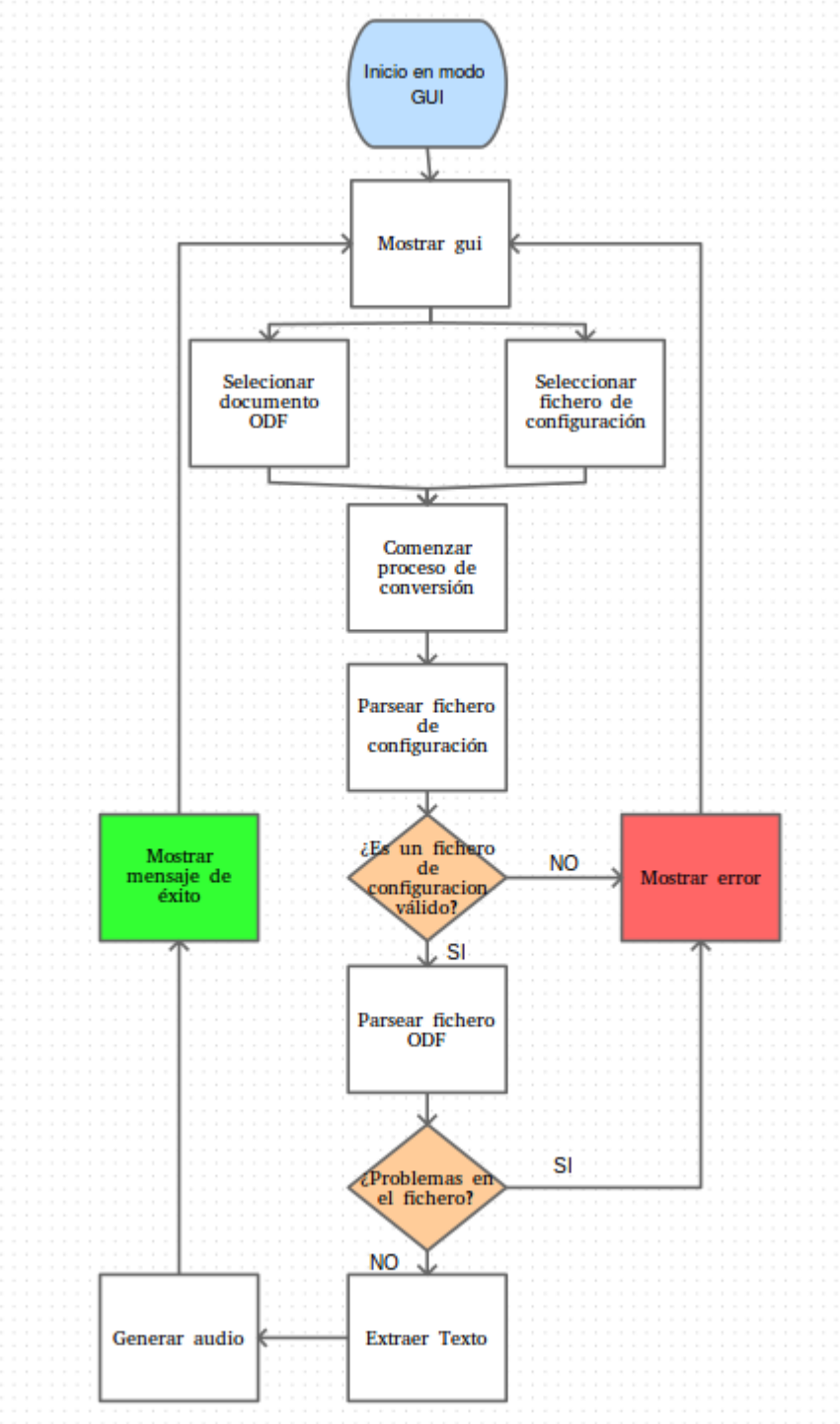
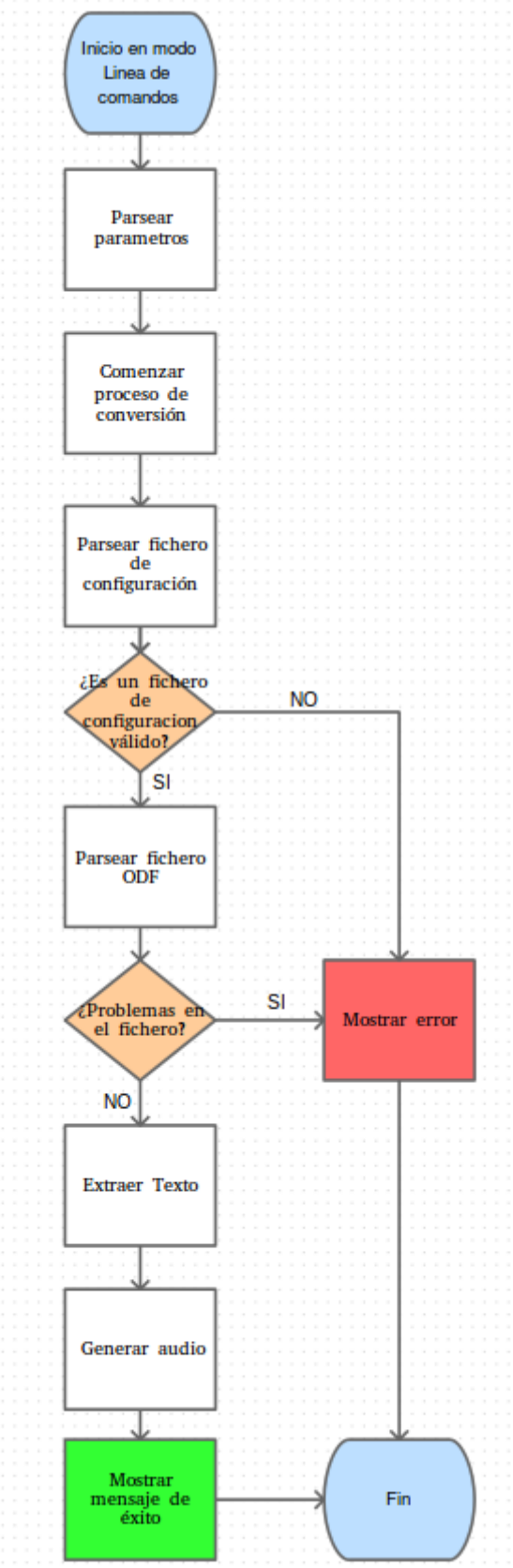


Diagrama de actividades en modo línea de comandos



4.4 Diseño arquitectónico.

Una vez expuestos todos los detalles del diseño dinámico y estático de nuestro sistema podemos llegar a la conclusión de que el sistema a implementar se puede representar por una arquitectura de dos capas con un solo nivel dado que residirán en una sola aplicación ejecutable en un ordenador.

La principal razón por la que se ha seleccionado una arquitectura de dos capas es que, tal y como se puede ver en el diseño, no hay una separación real entre lo que podría ser capa de presentación y capa de negocio. De hecho, la comunicación con los servicios externos, si bien es manejada por una clase de la capa de servicios externas, es ejecutada desde una clase que puede ser identificada como perteneciente a la capa de presentación.

Por tanto, dado el acoplamiento entre capa de negocio y capa de presentación podemos concluir que la capa de **Aplicación** será la encargada de mostrar los diversos eventos al usuario y contener la lógica de negocios de la aplicación.

Y por otro lado la capa de **Datos y servicios externos** será la encargada de manejar las conexiones con el servicio Online TTS y la ejecución de la aplicación externa del servicio Offline TTS.

Una vez realizada la especificación concreta del diseño propuesto, se puede clasificar cada una de las clases del diseño en una de las dos capas del sistema:

- **Capa de Aplicación**: Esta capa se puede dividir, a fin de mejorar la claridad de la documentación, a su vez en dos subcapas:
  - **Presentación**:
    - **Reader**. Esta clase instancia el tipo de reader que se va a ejecutar dependiendo de los parámetros pasados a la aplicación. En caso de no recibir parámetros se instanciará el reader con interfaz gráfica de usuario. En caso de ejecutarse con los parámetros –doc y –conig (mutuamente incluyentes) se instanciará el reader para linea de comandos.
    - **GUIReader.** Esta clase gestiona toda la infraestructura necesaria para presentar el interfaz gráfico de usuario, recoger los eventos generados por el usuario, ejecutar todas las tareas necesarias, recoger el feedback y mostrárselo al usuario.

      Instancia un objeto de la clase Config. Tras esto instancia un objeto especifico de la clase abstracta ODFParser (ODTParser, ODPParser u ODSParser) del que obtendrá el contenido del documento por medio de las operaciones disponibles en la clase ODFParser. Finalmente instanciará un objeto de la clase abstracta TTS (OnLineTTS o EspeakTTS) y por medio de los métodos de dicha clase ejecutará la conversión del texto a fichero de audio.

- **CLIReader**. Esta clase gestiona toda la infraestructura necesaria para ejecutar la aplicación en modo linea de consola, recoger los eventos generados por el usuario, ejecutar todas las tareas necesarias, recoger el feedback y mostrárselo al usuario (Errores, mensajes, etc).

  Instancia un objeto de la clase Config. Tras esto instancia un objeto especifico de la clase abstracta ODFParser (ODTParser, ODPParser u ODSParser) del que obtendrá el contenido del documento por medio de las operaciones disponibles en la clase ODFParser. Finalmente instanciará un objeto de la clase abstracta TTS (OnLineTTS o EspeakTTS) y por medio de los métodos de dicha clase ejecutará la conversión del texto a fichero de audio.

- **Negocio**.
  - **Config.** Esta clase gestiona la configuración de la aplicación. Se encarga del parseo del fichero de configuración y la validación de este contra el fichero XSD distribuido con la aplicación. En caso de no cumplir el formato del documento de configuración se levantará una excepción de parseo.
  - **Factoría ODFParser.** Esta factoría abstracta define una jerarquia de clases para el parseo de diversos tipos de formatos de ODF. De esta manera cuando sea necesario parsear otro tipo distinto de fichero solo habrá que añadir una nueva clase con la implementación concreta de los métodos de la clase
    - **Implementación ODTParser.** Esta es la implementación concreta para ficheros de tipo texto. El método getText de la clase recorre la lista de nodos del fichero a partir del nodo <office:text> de manera recursiva y extrae el contenido configurado.
    - **Implementación ODSParser.** Esta es la implementación concreta para ficheros de tipo hoja de cálculo. El método getText de la clase recorre la lista de nodos del fichero a partir del nodo <office:spreadsheet> de manera recursiva y extrae el contenido configurado.
    - **Implementación ODPParser.** Esta es la implementación concreta para ficheros de tipo presentación. El método getText de la clase recorre la lista de nodos del fichero a partir del nodo <draw:page> de manera recursiva y extrae el contenido configurado.

- **Capa de Datos y Servicios externos**:
  - **Factoría TTS.** Define la jerarquía necesaria para la transformación del texto extraído anteriormente. Cada una de las implementaciones definirá de manera concreta cada uno de los métodos necesarios para convertir el texto en audio dependiendo del sistema utilizado.

        - **Implementación OnLineTTS.** Que gestiona las conexiones con el servicio externo de conversión TTS. Realiza una conexión con el servicio web de conversión y genera ficheros en formato MP3. Llama a métodos de la clase AudioManager para convertir dichos ficheros a formato OGG.
        - **Implementación EspeakTTS.** Que gestiona la ejecución del binario espeak para producir ficheros offline. Realiza una llamada a la aplicación Espeak (que ddebe estar disponible en el sistema) y genera ficheros en formato WAV. Llama a métodos de la clase AudioManager para convertir dichos ficheros a formato OGG.
    - **AudioManager**. Esta clase gestiona la conversión de los streams de audio de las clases de manejo TTS y su escritura a disco en formato de audio libre.

4.5 Diseño de la configuración.

Tal y como se ha especificado en los requisitos, la aplicación tendrá un fichero de configuración para el funcionamiento del conversor en formato xml y con su correspondiente fichero xsd y un formato para configurar el logger de la aplicación.

4.5.1 Fichero de configuración xml

La información que se deberá mantener en este fichero se divide en dos bloques diferentes. La configuración del servicio TTS (tag <tts>) y la configuración de la aplicación en si (tag <reader>)

El tag **tts** cotendrá tres nodos hijos que regularán la siguiente configuración:

- **online:** Es un campo de tipo booleano (true o false). Con valor true la aplicación utilizará el servicio TTS configurado en el campo url. Con valor false utilizará el servicio TTS offline.
- **url:** Contendrá la url del servicio online tts. Campo obligatorio
- **language:** Es una cadena de dos caracteres que denota el idioma del fichero de entrada (ES para español, CA para catalán, EN para inglés, etc). Obligatorio

El tag **reader** que contendrá los siguientes nodos:

- **split-by:** Es un campo obligatorio en el que podrá almacenar los valores PAGE-BREAK o UNIQUE. obligatorio
- **alternatives:** Campo booleano. El valor de true hará que se lea el valor del campo alternative del documento. En caso contrario no se leerá.
- **output-path**: Path absoluto o relativo en el que se almacenarán los ficheros. Campo obligatorio.
- **output-pattern:** Nombre del fichero de salida al que se añadirá un timestamp. En caso de que se seleccione PAGE-BREAK en el campo split-by, se le añadirá un contador.

Por tanto, un ejemplo del fichero de configuración quedará de la siguiente forma:

```
<?xml version="1.0" encoding="UTF-8"?>
<config>
   <tts>
      <online>false</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>PAGE-BREAK</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

4.5.2 Fichero xsd

Tal y como se ha descrito en los requisitos, el fichero de configuración tendrá un esquema contra el que se validará cada vez que se ejecute la aplicación.

Cumpliendo toda la información descrita en la especificación de requisitos, tenemos que el fichero de xsd será el siguiente:

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="config">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="tts" >
          <xs:complexType>
            <xs:sequence>
              <xs:element type="xs:boolean" name="online"/>
              <xs:element type="xs:anyURI" name="url"/>
              <xs:element name="language">
                <xs:simpleType>
                  <xs:restriction base="xs:string">
                    <xs:enumeration value="ES"/>
                    <xs:enumeration value="EN"/>
                    <xs:enumeration value="CA"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="reader">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="split-by">
                <xs:simpleType>
                  <xs:restriction base="xs:string">
                    <xs:enumeration value="PAGE-BREAK"/>
                    <xs:enumeration value="UNIQUE"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>
              <xs:element type="xs:boolean" name="alternatives" default="true"/>
              <xs:element type="xs:string" name="output-path" />
              <xs:element type="xs:string" name="output-pattern" />
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

4.5.3 Fichero log4j.properties

Se ha añadido un logger [51] a la aplicación para obtener de manera fácil y ordenada algunos logs interesantes del funcionamiento de la aplicación, facilitando la obtención de diversos logs en diversos niveles.

Por ejemplo, algunas configuraciones permitirán obtener logs de DEBUG o incluso de nivel TRACE en momentos en los que se estén buscando errores en el sistema.

También se pueden configurar diversas localizaciones de logs, por lo que se puede,

por ejemplo enviar logs a la salida estándar y a un fichero localizado.

El contenido del fichero log4j.properties es el siguiente:

```
# Root logger option
log4j.rootLogger=INFO, stdout, file

# Redirect log messages to console
log4j.appender.stdout=org.apache.log4j.ConsoleAppender
log4j.appender.stdout.Target=System.out
log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
log4j.appender.stdout.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n

# Redirect log messages to a log file, support file rolling.
log4j.appender.file=org.apache.log4j.RollingFileAppender
log4j.appender.file.File=./log/file.out
log4j.appender.file.MaxFileSize=5MB
log4j.appender.file.MaxBackupIndex=10
log4j.appender.file.layout=org.apache.log4j.PatternLayout
log4j.appender.file.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n
```

Capitulo 5 Resultados y análisis del proceso

5.1 Proceso de implantación

Una vez finalizado el proceso de desarrollo del proyecto, comienza la fase de implantación en el usuario. En el caso del presente proyecto dicho proceso de implantación no es demasiado costoso teniendo en cuenta que ha sido diseñado y desarrollando utilizando herramientas multi-plataforma.

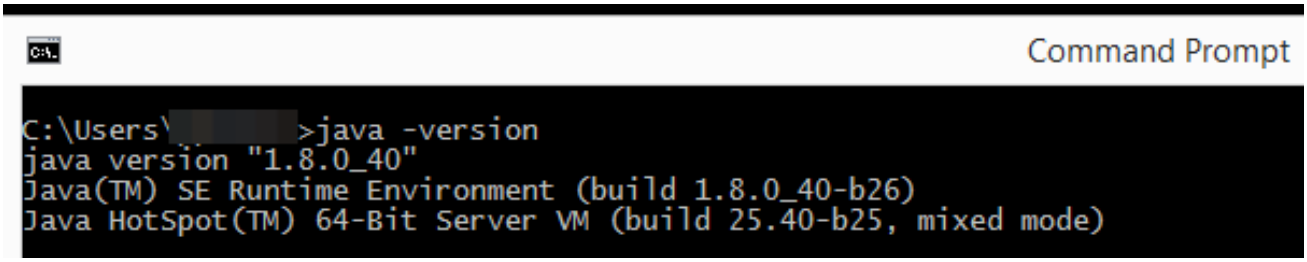
De esta manera, la única necesidad para la ejecución del entregable del proyecto es una máquina virtual de java 8 (no hay ninguna restricción sobre números de build no soportados). Una ve instalada la versión de la máquina virtual de java el proyecto puede ejecutarse en las versiones de 64 bits de Windows, Linux y en MacOS (aunque sin la posibilidad de utilización del procesamiento offline)

Una vez enumerados los requisitos se describe el proceso de implantación para cada una de las plataformas anteriormente:

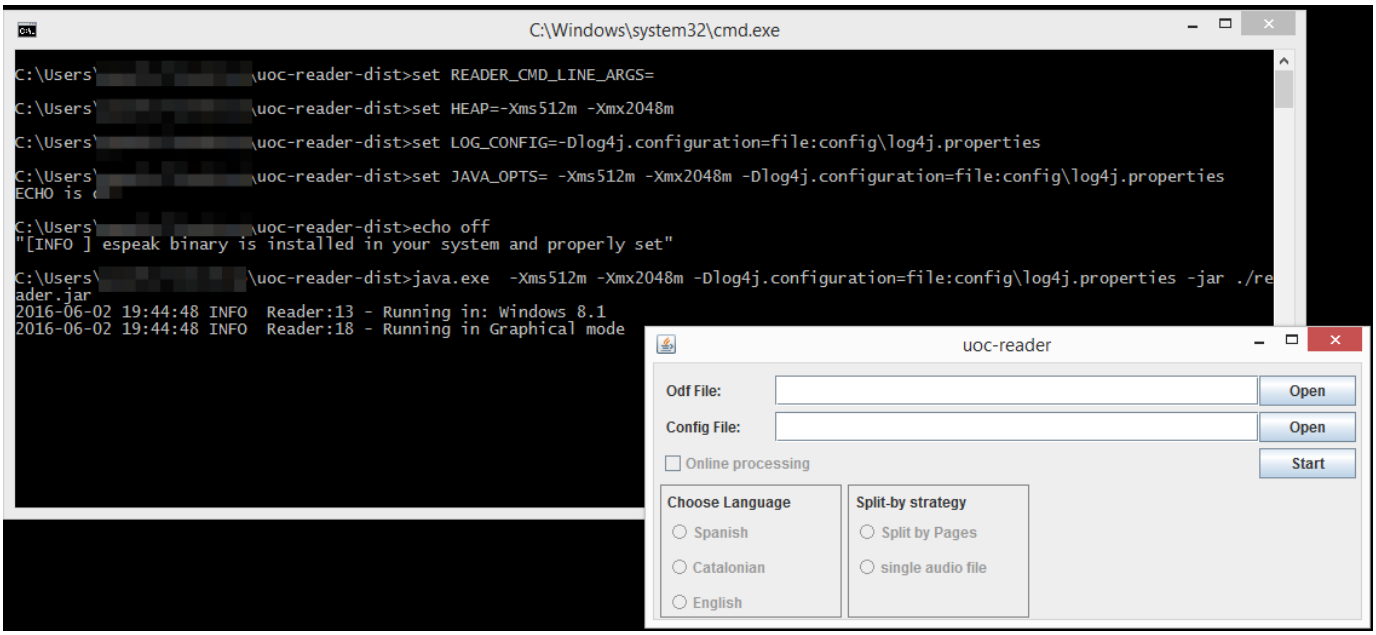
5.1.1 Windows

Pasos para instalar y ejecutar la aplicación en sistemas operativos windows.

1. **Instalación de java 8**. Para ello será suficiente con acceder a la siguiente url del sitio de descargas de oracle jre-8-win-x64.exe Tras ejecutar el fichero .exe que se descargará, se iniciará el proceso de instalación. No hay ningún requisito especial para dicha instalación con lo que valdrá con aceptar todos los valores por defecto. Para comprobar que java 8 se encuentra correctamente instalado solo habrá que ejecutar el siguiente comando en una consola de linea de comandos: **java -version** y la salida por pantalla debería ser parecido a la siguiente:



2. **Instalación de la aplicación.** La aplicación se encuentra disponible empaquetada en formato zip. De esta manera se podrá descomprimir utilizando cualquier herramienta gratuita para windows (7z, winrar, winzip, el propio explorador de ficheros de windows). Una vez descomprimido, el usuario verá una carpeta llamada uoc-reader-dist que contendrá, entre otras cosas, un fichero llamado **reader.bat**. Al hacer doble clic sobre el fichero se abrirá una linea de comandos y el interfaz principal de la aplicación tal y como se puede ver en la siguiente captura de pantalla:



5.1.2 Linux

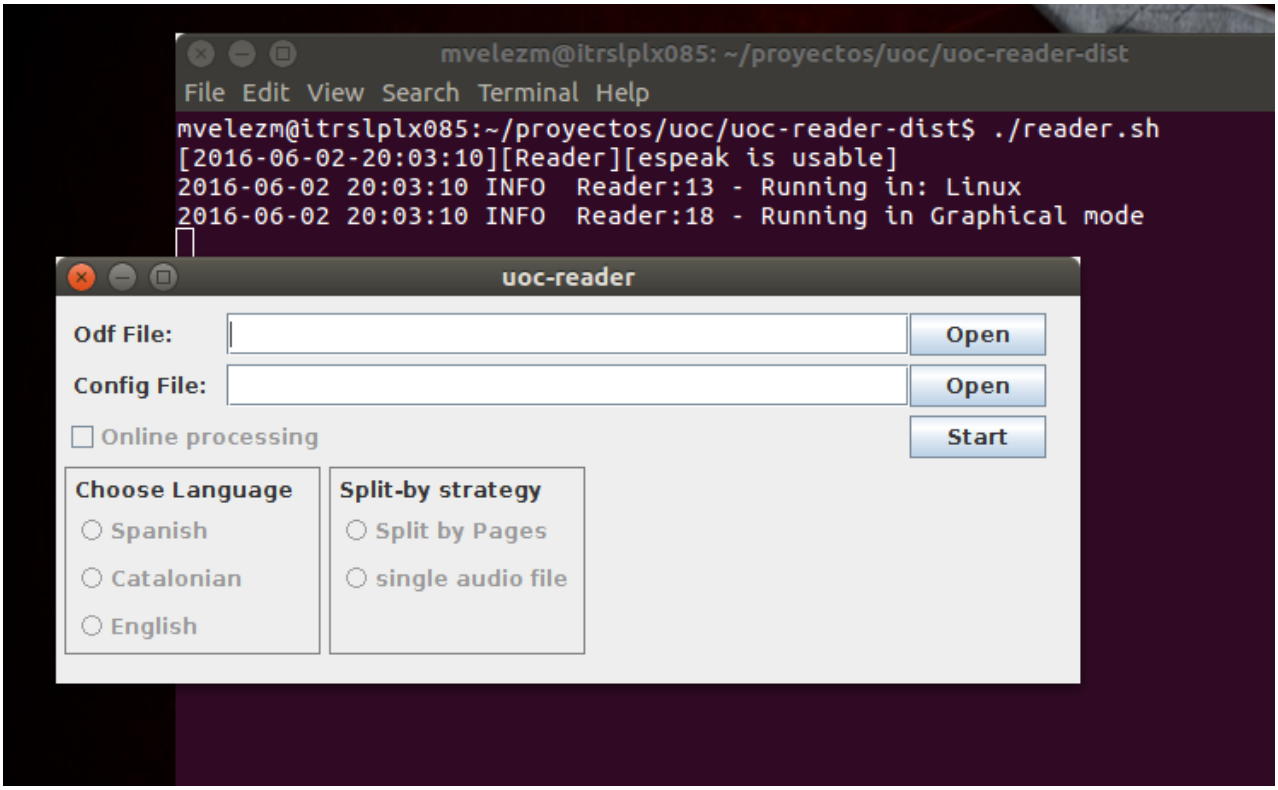
Pasos para instalar y ejecutar la aplicación en sistemas operativos Linux:

1. **Instalación de java 8**. Para la Para ello será suficiente con acceder a la siguiente url del sitio de descargas de oracle jre-8-linux-x64.tar.gz. En este caso la ejecución es algo mas complicada. Se debe descomprimir el fichero descargado utilizando el siguiente comando **tar -zxvf jre-8u91-linux-i586.tar.gz** que creará un directorio llamado **jre1.8.0_91.** Suponiendo que la distribución de java se hubiera descomprimido en /example tendriamos el ejecutable de java en **/example/jre1.8.0_91/bin/java** por lo que dicho directorio tendría que se añadido al path del sistema o del usuario (Es posible obtener mas información sobre la adición de directorios al path en la siguiente url: variables de entorno). Otra posibilidad es la instalación del openjdk 8, instalable fácilmente mediante el gestor de paquetes de cualquier distribución. Para comprobar que java 8 se encuentra correctamente instalado solo habrá que ejecutar el siguiente comando en una consola de linea de comandos: **java -version** y la salida por pantalla debería ser parecido a la siguiente:

2. **Instalación de la aplicación.** La aplicación se encuentra disponible empaquetada en formato zip. De esta manera se podrá descomprimir utilizando el comando **unzip**. Una vez descomprimido, el usuario verá una carpeta llamada uoc-reader-dist que contendrá, entre otras cosas, un fichero

```
mvelezm@itrslplx085:~/CloudStation/Downloads$ java -version
java version "1.8.0_51"
Java(TM) SE Runtime Environment (build 1.8.0_51-b16)
Java HotSpot(TM) 64-Bit Server VM (build 25.51-b03, mixed mode)
```

llamado reader.sh. Al ejecutar el fichero se abrirá una linea de comandos y el interfaz principal de la aplicación tal y como se puede ver en la siguiente captura de pantalla:



5.1.3 MacOSX

Pasos para instalar y ejecutar la aplicación en sistemas operativos Linux:

1. **Instalación de java 8**. Para la Para ello será suficiente con hacer clic en el siguiente link del sitio de descargas de oracle jre-8-macosx-x64.dmg . La instalación comenzará al hacer doble clic sobre el fichero dmg descargado. Para comprobar que java 8 se encuentra correctamente instalado solo habrá que ejecutar el siguiente comando en una consola de linea de comandos: **java -version** y la salida por pantalla debería ser parecido a la siguiente:
2. **Instalación de la aplicación.** La aplicación se encuentra disponible empaquetada en formato zip. De esta manera se podrá descomprimir utilizando el comando **unzip**. Una vez descomprimido, el usuario verá una carpeta llamada uoc-reader-dist que contendrá, entre otras cosas, un fichero llamado reader.sh. Al ejecutar el fichero se abrirá una linea de comandos y el interfaz principal de la aplicación.

5.2 Manual de usuario

Una vez instalada la aplicación en el sistema siguiendo los pasos explicados en el punto anterior, se describirá el manual de usuario de la aplicación.

El primer punto a tratar en este manual de usuario son las precondiciones necesarias para la ejecución de la aplicación.

5.2.1 Precondiciones

El funcionamiento normal de la aplicación viene delimitado por la existencia de dos tipos de ficheros que deben existir en el momento de la ejecución de la aplicación:

- **Fichero XML de configuración**. Es un fichero xml que cumple un esquema definido en un documento xsd. En dicho fichero de configuración el usuario definirá la ciertas variables de configuración relativas al proceso de extracción de texto y de conversión de dicho texto a formato de audio.
- **Documento ODF**. Los formatos permitidos por la aplicación son ODT (OpenDocument Text – Documentos de texto), ODS (OpenDocument Spread sheet – Hojas de cálculo) y ODP (OpenDocument Presentation – Presentaciones)

Para poder extraer texto de un documento es necesaria la existencia de un fichero de configuración y, naturalmente, un fichero ODF que procesar.

5.2.2 Funcionalidades de la aplicación

Con la aplicación diseñada e implementada para el presente proyecto fin de carrera el usuario podrá realizar las siguientes acciones.

- Conversión de un fichero ODF en un solo fichero de audio.
- Conversión de un fichero ODF en tantos ficheros como páginas tenga el documento.

Dichas acciones se pueden realizar mediante la interfaz gráfica de usuario (GUI) y mediante una ejecución utilizando parámetros en la linea de comandos (CLI).

La ejecución mediante interfaz gráfica permite al usuario la modificación de ciertos parámetros antes de cada ejecución mientras que la ejecución por linea de comandos utilizará la configuración leída del fichero Xml indicado sin posibilidad de realizar modificaciones durante la ejecución

Además es importante señalar que la aplicación puede generar el audio utilizando un servicio de lectura online u otro de lectura offline dependiendo de la configuración seleccionada.

Con el fin de simplificar lo máximo posible los procedimientos descritos en el presente manual de usuario de la aplicación se ha decidido dividir la funcionalidad en dos grandes bloques: Interfaz gráfica del usuario (GUI) y Linea de comandos

(CLI) cada una con una sección en el manual.

5.2.3 Interfaz gráfica de usuario

Una vez instalada la aplicación en el sistema de explotación, para iniciarla en modo gráfico bastará con ejecutar sin parámetros el script apropiado para el sistema operativo del usuario:

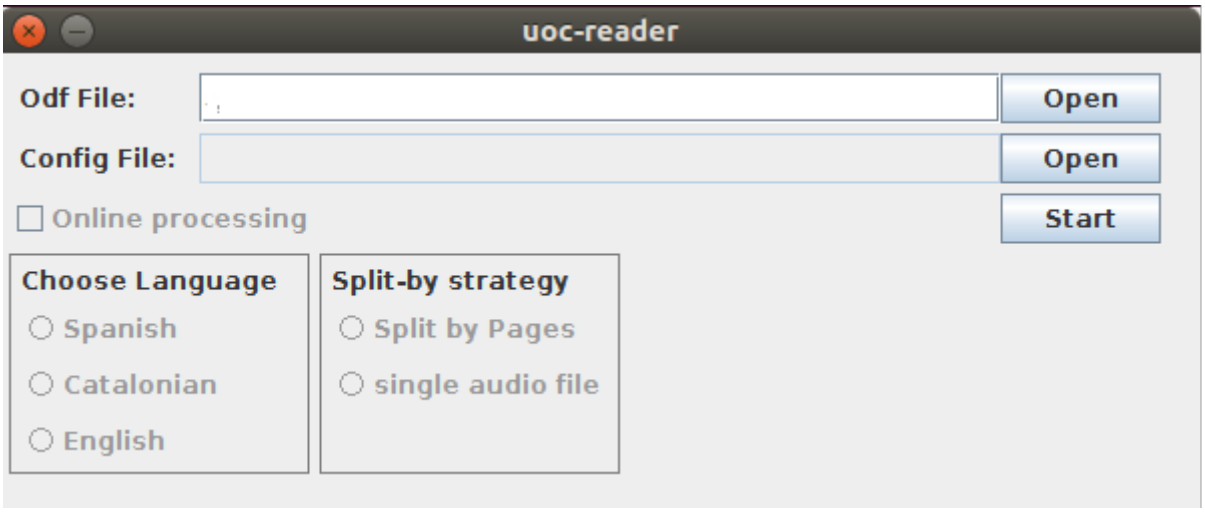
- **Windows**

reader.bat

- **Linux / MacOSX**

reader.sh

Una vez ejecutado el comando, se mostrará la pantalla principal de la aplicación que tendrá el siguiente aspecto:



El estado inicial de la aplicación muestra los siguientes elementos.

- Linea Odf file que contiene un campo de texto y un botón open. Al pulsar el botón Open, el usuario seleccionará un fichero odf (Formatos ODT, ODS u ODP) del que se extraerá el texto.
- Linea Config File que contiene un campo de texto y un botón open. Al pulsar el botón Open, el usuario seleccionará un fichero xml de configuración. Al seleccionar el fichero, la aplicación leerá la configuración y habilitará las opciones de configuración de interfaz.
- CheckBox Online Processing que estará inicialmente deshabilitada. Cuando el usuario seleccione un fichero, el CheckBox se habilitará y tendrá valor seleccionado o deseleccionado dependiendo de la configuración almacenada en dicho fichero. Si el tag <online> tiene valor True, el checkbox estará

habilitado y en caso contrario estará deshabilitado.

- Cuadro de configuración Choosen Language que mostrará tres radio buttons deshabilitados con valores Español, Catalán, Inglés. Cuando el usuario seleccione un fichero, los radioButtons se habilitarán y mostrará el idioma incluido en el campo language de la configuración almacenada en dicho fichero. Los valores posibles son ES para español, CA para catalán y EN para inglés.
- Cuadro de configuración Split-by Strategy que mostrará dos radio buttons deshabilitados con valores Split by Pages y single audio file. Cuando el usuario seleccione un fichero de configuración, los radioButtons se habilitarán y mostrará la configuración de fragmentado de ficheros. Los valores posibles son PAGE-BREAK para Split by Pages y UNIQUE para single audio file.

Generación del audio en un solo fichero a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz offline

Para ello se utilizarán los siguientes documentos:

- Fichero ODT example.odt (un fichero con 4 páginas)
- Fichero de configuración single-audio-offline.xml con el siguiente contenido:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>false</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>UNIQUE</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

El procedimiento para generar un fichero de audio a partir de un documento de cuatro páginas incluye los siguientes pasos:

1. Ejecución de la aplicación mediante la utilización del script apropiado al sistema operativo sin utilizar parámetros.
2. Pulsado del botón open de la linea Odf File para seleccionar el fichero example.odt en su emplazamiento.
3. Pulsado del botón Open de la linea Config File para seleccionar el fichero single-audio-offline.xml en su emplazamiento.

4. Pulsado del botón Start. El botón Start permanecerá deshabilitado hasta que se termine el procesamiento. En dicho momento se mostrará un mensaje en consola especificado que la aplicación ya ha terminado el procesamiento.
5. Tras esto se habrá generado un fichero de audio que se habrá almacenado en el path especificado en el tag <output-path> de la configuración con el nombre especificado en el tag <output-pattern> seguido de la fecha en formato YYYYMMDD-Hhmmss y la extensión ogg

Generación del audio en un varios ficheros a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz online

Para ello se utilizarán los siguientes documentos:

- Fichero ODT example.odt (un fichero con 4 páginas)
- Fichero de configuración multi-audio-online.xml con el siguiente contenido:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>true</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>PAGE-BREAK</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

El procedimiento para generar un fichero de audio a partir de un documento de cuatro páginas incluye los siguientes pasos:

1. Ejecución de la aplicación mediante la utilización del script apropiado al sistema operativo sin utilizar parámetros.
2. Pulsado del botón open de la linea Odf File para seleccionar el fichero example.odt en su emplazamiento.
3. Pulsado del botón Open de la linea Config File para seleccionar el fichero multi-audio-online.xml en su emplazamiento.
4. Pulsado del botón Start. El botón Start permanecerá deshabilitado hasta que se termine el procesamiento. En dicho momento se mostrará un mensaje en consola especificado que la aplicación ya ha terminado el procesamiento.
5. Tras esto se habrán generado tantos ficheros de audio como páginas tiene el documento (4 en este caso) y se habrán almacenado en el path especificado en el tag <output-path> de la configuración teniendo todos ellos como nombre el especificado en el tag <output-pattern> seguido de la fecha en formato YYYYMMDD-Hhmmss y un guión bajo mas el número de página y la extensión ogg

Generación del audio en un varios ficheros a partir de un documento de texto de varias páginas modificando la configuración. Proveedor de voz offline

Para ello se utilizarán los siguientes documentos:

- Fichero ODT example.odt (un fichero con 4 páginas) en inglés
- Fichero de configuración multi-audio-offline.xml con el siguiente contenido:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>false</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>PAGE-BREAK</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

El procedimiento para generar un fichero de audio a partir de un documento de cuatro páginas incluye los siguientes pasos:

1. Ejecución de la aplicación mediante la utilización del script apropiado al sistema operativo sin utilizar parámetros.
2. Pulsado del botón open de la linea Odf File para seleccionar el fichero example.odt en su emplazamiento.
3. Pulsado del botón Open de la linea Config File para seleccionar el fichero multi-audio-online.xml en su emplazamiento.
4. Una vez leída la configuración, se mostrarán los valores del fichero que se modificarán manualmente. Para ello se dejará desmarcado el checkbox online processing, se seleccionará el idioma ingles y como split-by strategy se seleccionará Split by Pages
5. Pulsado del botón Start. El botón Start permanecerá deshabilitado hasta que se termine el procesamiento. En dicho momento se mostrará un mensaje en consola especificado que la aplicación ya ha terminado el procesamiento.
6. Tras esto se habrán generado tantos ficheros de audio como páginas tiene el documento (4 en este caso) y se habrán almacenado en el path especificado en el tag <output-path> de la configuración teniendo todos ellos como nombre el especificado en el tag <output-pattern> seguido de la fecha en formato YYYYMMDD-Hhmmss y un guión bajo mas el número de página y la extensión ogg. (Tal y como se puede comprobar, el resultado final no sigue la configuración inicial si no la configuración modificada manualmente mediante el interfaz.

5.2.4 Línea de comandos

Una vez instalada la aplicación en el sistema de explotación, para iniciarla en modo gráfico bastará con ejecutar sin parámetros el script apropiado para el sistema operativo del usuario:

- **Windows**

reader.bat -config fichero_de_configuración -doc fichero_odf

- **Linux / MacOSX**

reader.sh -config fichero_de_configuración -doc fichero_odf

Una vez ejecutado el comando, se iniciará automáticamente la conversión del fichero odf en audio siguiendo la configuración almacenada en el fichero de configuración pasado por línea de parámetros.

Los parámetros posibles son los siguientes:

- -config (obligatorio) en el que se especifica el path al fichero de configuración a utilizar
- -document (obligatorio) en el que se especifica el path al documento odf a procesar.

Generación del audio en un solo fichero a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz offline

Para ello se utilizarán los siguientes documentos:

- Fichero ODT example.odt (un fichero con 4 páginas)
- Fichero de configuración single-audio-offline.xml con el siguiente contenido:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>false</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>UNIQUE</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

El procedimiento para generar un fichero de audio a partir de un documento de cuatro páginas incluye los siguientes pasos:

1. ejecución del comando apropiado para el sistema operativo:
    - windows: reader.bat -config single-audio-offline.xml -doc example.odt

- Linux: reader.sh -config single-audio-offline.xml -doc example.odt

2. Tras esto se habrá generado un fichero de audio que se habrá almacenado en el path especificado en el tag <output-path> de la configuración con el nombre especificado en el tag <output-pattern> seguido de la fecha en formato YYYYMMDD-Hhmmss y la extensión ogg. El proveedor utilizado será el proveedor offline

Generación del audio en un varios ficheros a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz offline

Para ello se utilizarán los siguientes documentos:

- Fichero ODT example.odt (un fichero con 4 páginas)
- Fichero de configuración multi-audio-offline.xml con el siguiente contenido:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>false</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>PAGE-BREAK</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

El procedimiento para generar un fichero de audio a partir de un documento de cuatro páginas incluye los siguientes pasos:

1. ejecución del comando apropiado para el sistema operativo:
   - windows: reader.bat -config multi-audio-offline.xml -doc example.odt
   - Linux: reader.sh -config multi-audio-offline.xml -doc example.odt
2. Tras esto se habrán generado tantos ficheros de audio como páginas tiene el documento (4 en este caso) y se habrán almacenado en el path especificado en el tag <output-path> de la configuración teniendo todos ellos como nombre el especificado en el tag <output-pattern> seguido de la fecha en formato YYYYMMDD-Hhmmss y un guión bajo mas el número de página y la extensión ogg. El proveedor utilizado será el proveedor offline

Generación del audio en un solo fichero a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz online

Para ello se utilizarán los siguientes documentos:

- Fichero ODT example.odt (un fichero con 4 páginas)
- Fichero de configuración single-audio-online.xml con el siguiente contenido:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>true</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>UNIQUE</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

El procedimiento para generar un fichero de audio a partir de un documento de cuatro páginas incluye los siguientes pasos:

1. ejecución del comando apropiado para el sistema operativo:
    - windows: reader.bat -config single-audio-online.xml -doc example.odt
    - Linux: reader.sh -config single-audio-online.xml -doc example.odt
2. Tras esto se habrá generado un fichero de audio que se habrá almacenado en el path especificado en el tag <output-path> de la configuración con el nombre especificado en el tag <output-pattern> seguido de la fecha en formato YYYYMMDD-Hhmmss y la extensión ogg. El proveedor utilizado será el proveedor online

Generación del audio en un varios ficheros a partir de un documento de texto de varias páginas sin modificación de la configuración. Proveedor de voz online

Para ello se utilizarán los siguientes documentos:

- Fichero ODT example.odt (un fichero con 4 páginas)
- Fichero de configuración multi-audio-online.xml con el siguiente contenido:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>true</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>PAGE-BREAK</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

El procedimiento para generar un fichero de audio a partir de un documento de cuatro páginas incluye los siguientes pasos:

1. ejecución del comando apropiado para el sistema operativo:
   - windows: reader.bat -config multi-audio-online.xml -doc example.odt
   - Linux: reader.sh -config multi-audio-online.xml -doc example.odt
2. Tras esto se habrán generado tantos ficheros de audio como páginas tiene el documento (4 en este caso) y se habrán almacenado en el path especificado en el tag <output-path> de la configuración teniendo todos ellos como nombre el especificado en el tag <output-pattern> seguido de la fecha en formato YYYYMMDD-Hhmmss y un guión bajo mas el número de página y la extensión ogg. El proveedor utilizado será el proveedor online.

5.3 Procedimientos de pruebas

Como es natural en un proceso de desarrollo de software es imprescindible la realización de diversas pruebas para confirmar que se cumplen todos los requisitos descritos durante la fase de especificación de la aplicación.

Para ello, se han diseñado un procedimiento de test que cubre toda la funcionalidad desarrollada.

Cada uno de los tests a ejecutar utilizaran un fichero de configuración y un fichero en formato odf con el siguiente contenido:

- Documentos ODF:
  - single-page.odt: un fichero odt con una sola página
  - multi-page.odt: un fichero odt con 4 páginas
  - 70-pages.odt: un fichero odt con 70 páginas
  - spreadsheet.ods: una hoja de calculo
  - presentation.odp: una presentación con varias páginas
- Configuraciones:
  - single-audio-offline.xml:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>false</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>UNIQUE</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

  - single-audio-online.xml:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>true</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>UNIQUE</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

- multi-audio-offline.xml:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>false</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>PAGE-BREAK</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

- multi-audio-online.xml:

```
<?xml version="1.1" encoding="UTF-8"?>
<config>
   <tts>
      <online>true</online>
      <url>http://translate.google.com/translate_tts</url>
      <language>ES</language>
   </tts>
   <reader>
      <split-by>PAGE-BREAK</split-by>
      <alternatives>true</alternatives>
      <output-path>./output</output-path>
      <output-pattern>testName</output-pattern>
   </reader>
</config>
```

5.3.1 Tests realizados

En la siguiente tabla se incluyen todos los tests realizados incluyendo el identificador del test, el título, la fecha de ejecución y el resultado del test.

En la siguiente sección se añadirá además la descripción de cada uno de los tests.

Identificador	Titulo	Fecha	Resultado
GUI-001	Conversión de un fichero de una página en un solo fichero de audio. Offline	12/06/2016	OK
GUI-002	Conversión de un fichero de una página en un solo fichero de audio. Online	12/06/2016	OK
GUI-003	Conversión de un fichero de varias páginas en un solo fichero de audio. Offline	12/06/2016	OK
GUI-004	Conversión de un fichero de varias páginas en un solo fichero de audio. Online	12/06/2016	OK
GUI-005	Conversión de un fichero de varias páginas en varios ficheros de audio. Offline	12/06/2016	OK
GUI-006	Conversión de un fichero de varias páginas en varios ficheros de audio. Online	12/06/2016	OK
GUI-007	Conversión de un fichero de una sola página	12/06/2016	OK

	utilizando configuración para ficheros de varias páginas. Offline		
GUI-008	Conversión de un fichero de una sola página utilizando configuración para ficheros de varias páginas. Online	12/06/2016	OK
GUI-009	Conversión de un fichero de varias páginas cambiando de configuración en el interfaz	12/06/2016	OK
GUI-010	Conversión de un fichero de 70 páginas en un solo fichero de audio	12/06/2016	OK
GUI-011	Conversión de un fichero de 70 páginas en varios ficheros de audio	12/06/2016	OK
GUI-012	Error en la conversión. Fichero ODF no disponible	12/06/2016	OK
GUI-013	Error en la conversión. Configuración no disponible	12/06/2016	OK
GUI-014	Conversión de un fichero ODS	12/06/2016	OK
GUI-015	Conversión de un fichero ODP	12/06/2016	OK
CLI-001	Conversión de un fichero de una página en un solo fichero de audio. Offline	12/06/2016	OK
CLI-002	Conversión de un fichero de una página en un solo fichero de audio. Online	12/06/2016	OK
CLI-003	Conversión de un fichero de varias páginas en un solo fichero de audio. Offline	12/06/2016	OK
CLI-004	Conversión de un fichero de varias páginas en un solo fichero de audio. Online	12/06/2016	OK
CLI-005	Conversión de un fichero de varias páginas en varios ficheros de audio. Offline	12/06/2016	OK
CLI-006	Conversión de un fichero de varias páginas en varios ficheros de audio. Online	12/06/2016	OK
CLI-007	Conversión de un fichero de una sola página utilizando configuración para ficheros de varias páginas. Offline	12/06/2016	OK
CLI-008	Conversión de un fichero de una sola página utilizando configuración para ficheros de varias páginas. Online	12/06/2016	OK
CLI-009	Conversión de un fichero de 70 páginas en un solo fichero de audio	12/06/2016	OK
CLI-010	Conversión de un fichero de 70 páginas en varios ficheros de audio	12/06/2016	OK

CLI-011	Error en la conversión. Configuración no disponible	12/06/2016	OK
CLI-012	Fichero ODF no disponible	12/06/2016	OK
CLI-013	Conversión de un fichero ODS	12/06/2016	OK
CLI-014	Conversión de un fichero ODP	12/06/2016	OK

5.3.2 Descripción detallada de los tests

En cada una de las siguientes tablas se describirán cada uno de los tests incluidos en la sección anterior.

<table>
<tr><th>ID</th><th colspan="2">Titulo</th></tr>
<tr><td rowspan="5">GUI-001</td><td colspan="2">Conversión de un fichero de una página en un solo fichero de audio. Offline</td></tr>
<tr><td>Procedimiento</td><td>Resultado</td></tr>
<tr><td>1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat</td><td>1. Se muestra la pantalla principal de la aplicación.</td></tr>
<tr><td>2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero single-page.odt del juego de test. Pulsar el botón open</td><td>2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario</td></tr>
<tr><td>3. Pulsar el botón Open relativo al Config file y seleccionar el fichero single-audio-offline.xml del juego de test. Pulsar el botón open</td><td>3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores:
Check online procesing desmarcado.
Choose language con el lenguaje especificado en el fichero de configuración
Split by strategy con el valor single audio file</td></tr>
<tr><td>4. Pulsar el botón Start.</td><td>4. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.</td></tr>
</table>

ID	Titulo
GUI-002	Conversión de un fichero de una página en un solo fichero de audio. Online

	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero single-page.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero single-audio-online.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing marcado. • Choose language con el lenguaje especificado en el fichero de configuración • Split by strategy con el valor single audio file
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
GUI-003	Conversión de un fichero de varias páginas en un solo fichero de audio. Offline	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero multi-page.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario

	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero single-audio-offline.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing desmarcado. • Choose language con el lenguaje especificado en el fichero de configuración • Split by strategy con el valor single audio file
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
GUI-004	Conversión de un fichero de varias páginas en un solo fichero de audio. Online	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero multi-page.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero single-audio-online.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing marcado. • Choose language con el lenguaje especificado en el

		fichero de configuración • Split by strategy con el valor single audio file
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
GUI-005	Conversión de un fichero de varias páginas en varios ficheros de audio. Offline	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero multi-page.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero multi-audio-offline.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing desmarcado. • Choose language con el lenguaje especificado en el fichero de configuración • Split by strategy con el valor split by pages
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrán producido tantos ficheros de audio como páginas tenga el documento odt en el path especificado en la configuración, con el nombre

		especificado en la configuración incluyendo la fecha de ejecución y un contador que indicará el número de página al que corresponde cada fichero.

ID	Titulo	
GUI-006	Conversión de un fichero de varias páginas en varios ficheros de audio. Online	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero multi-page.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero multi-audio-online.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing marcado. • Choose language con el lenguaje especificado en el fichero de configuración • Split by strategy con el valor split by pages
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrán producido tantos ficheros de audio como páginas tenga el documento odt en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución y un contador que indicará el número

		de página al que corresponde cada fichero.

ID	Titulo	
GUI-007	Conversión de un fichero de una sola página utilizando configuración para ficheros de varias páginas. Offline	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero single-page.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero multi-audio-offline.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing desmarcado. • Choose language con el lenguaje especificado en el fichero de configuración • Split by strategy con el valor split by pages
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
GUI-008	Conversión de un fichero de una sola página utilizando configuración para ficheros de varias páginas. Online	
	Procedimiento	Resultado

	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero single-page.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero multi-audio-online.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing desmarcado. • Choose language con el lenguaje especificado en el fichero de configuración • Split by strategy con el valor split by pages
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
GUI-009	Conversión de un fichero de varias páginas cambiando de configuración en el interfaz	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero muti-page.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero	3. Se muestra la pantalla principal que incluye el path del fichero de

	single-audio-offline.xml del juego de test. Pulsar el botón open	configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing desmarcado. • Choose language con el lenguaje especificado en el fichero de configuración • Split by strategy con el valor single audio file
	4. Seleccionar el check online procesing y cambiar la splity by strategy a split by pages. Pulsar el botón start	4. Se ejecutará el procedimiento de conversión y al finalizar se habrán producido tantos ficheros de audio como páginas tenga el documento odt en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución y un contador que indicará el número de página al que corresponde cada fichero.

ID	Titulo	
GUI-010	Conversión de un fichero de 70 páginas en un solo fichero de audio	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero 70-pages.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero single-audio-offline.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing desmarcado. • Choose language con el

		lenguaje especificado en el fichero de configuración • Split by strategy con el valor single audio file
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
GUI-011	Conversión de un fichero de 70 páginas en varios ficheros de audio	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero 70-pages.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero multi-audio-offline.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing desmarcado. • Choose language con el lenguaje especificado en el fichero de configuración • Split by strategy con el valor single audio file
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre

		especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
GUI-012	Error en la conversión. Fichero ODF no disponible	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón start sin seleccionar ningún fichero odf.	2. Se mostrará un popup con un mensaje de error

ID	Titulo	
GUI-013	Error en la conversión. Configuración no disponible	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero single-page.odt del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Start sin seleccionar ningún fichero de configuración	3. Se mostrará un popup con un mensaje de error.

ID	Titulo	
GUI-014	Conversión de un fichero ODS	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero spreadsheet.ods del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario

	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero single-audio-offline.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing desmarcado. • Choose language con el lenguaje especificado en el fichero de configuración • Split by strategy con el valor single audio file
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
GUI-015	Conversión de un fichero ODP	
	Procedimiento	Resultado
	1. Ejecutar la aplicación sin ningún parámetro mediante el script reader.sh o reader.bat	1. Se muestra la pantalla principal de la aplicación.
	2. Pulsar el botón Open relativo al Odf File y seleccionar el fichero presentation.odp del juego de test. Pulsar el botón open	2. Se muestra la pantalla principal que incluye el path del fichero odt seleccionado en la linea Odf File del formulario
	3. Pulsar el botón Open relativo al Config file y seleccionar el fichero multi-audio-offline.xml del juego de test. Pulsar el botón open	3. Se muestra la pantalla principal que incluye el path del fichero de configuración seleccionado en la linea config File del formulario. La sección de configuración se habilitará con los siguientes valores: • Check online procesing desmarcado. • Choose language con el lenguaje especificado en el

		fichero de configuración • Split by strategy con el valor Split by pages
	4. Pulsar el botón Start.	4. Se ejecutará el procedimiento de conversión y al finalizar se habrán producido tantos ficheros de audio como páginas tenga el documento odt en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución y un contador que indicará el número de página al que corresponde cada fichero.

ID	Titulo	
CLI-001	Conversión de un fichero de una página en un solo fichero de audio. Offline	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config single-audio-offline.xml -doc single-page.odt o reader.sh -config single-audio-offline.xml -doc single-page.odt	1. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
CLI-002	Conversión de un fichero de una página en un solo fichero de audio. Online	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config single-audio-online.xml -doc single-page.odt o reader.sh -config single-audio-online.xml -doc single-page.odt	1. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
CLI-003	Conversión de un fichero de varias páginas en un solo fichero de audio. Offline	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config single-audio-offline.xml -doc multi-page.odt o reader.sh -config single-audio-offline.xml -doc multi-page.odt	1. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
CLI-004	Conversión de un fichero de varias páginas en un solo fichero de audio. Online	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config single-audio-online.xml -doc multi-page.odt o reader.sh -config single-audio-online.xml -doc multi-page.odt	1. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
CLI-005	Conversión de un fichero de varias páginas en varios ficheros de audio. Offline	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config multi-audio-offline.xml -doc multi-page.odt o reader.sh -config multi-audio-	1. Se ejecutará el procedimiento de conversión y al finalizar se habrán producido tantos ficheros de audio como páginas tenga el documento odt en el path especificado en la configuración, con el nombre especificado en la configuración

	offline.xml -doc multi-page.odt	incluyendo la fecha de ejecución y un contador que indicará el número de página al que corresponde cada fichero.

ID	Titulo	
CLI-006	Conversión de un fichero de varias páginas en varios ficheros de audio. Online	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config multi-audio-online.xml -doc multi-page.odt o reader.sh -config multi-audio-online.xml -doc multi-page.odt	Se ejecutará el procedimiento de conversión y al finalizar se habrán producido tantos ficheros de audio como páginas tenga el documento odt en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución y un contador que indicará el número de página al que corresponde cada fichero.

ID	Titulo	
CLI-007	Conversión de un fichero de una sola página utilizando configuración para ficheros de varias páginas. Offline	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config multi-audio-offline.xml -doc single-page.odt o reader.sh -config multi-audio-offline.xml -doc single-page.odt	1. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
CLI-008	Conversión de un fichero de una sola página utilizando configuración para ficheros de varias páginas. Online	
	Procedimiento	Resultado

	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config multi-audio-online.xml -doc single-page.odt o reader.sh -config multi-audio-online.xml -doc single-page.odt	1. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
CLI-009	Conversión de un fichero de 70 páginas en un solo fichero de audio	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config single-audio-offline.xml -doc 70-pages.odt o reader.sh -config single-audio-offline.xml -doc 70-pages.odt	1. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
CLI-010	Conversión de un fichero de 70 páginas en varios ficheros de audio	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config multi-audio-offline.xml -doc 70-pages.odt o reader.sh -config multi-audio-offline.xml -doc 70-pages.odt	1. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
CLI-011	Error en la conversión. Fichero ODF no disponible	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la	1. Se mostrará un mensaje de error por consola y no habrá ningún tipo

	siguiente manera: reader.bat -config single-audio-offline.xml -doc fichero_no_existente.odf o reader.sh -config single-audio-offline.xml -doc fichero_no_existente.odf	de salida

ID	Titulo	
CLI-012	Error en la conversión. Configuración no disponible	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config config_no_existente.xml -doc single-page.odt o reader.sh -config config_no_existente.xml -doc single-page.odt	1. Se mostrará un mensaje de error por consola y no habrá ningún tipo de salida

ID	Titulo	
CLI-013	Conversión de un fichero ODS	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la siguiente manera: reader.bat -config single-audio-offline.xml -doc spreadsheet.ods o reader.sh -config single-audio-offline.xml -doc spreadsheet.ods	1. Se ejecutará el procedimiento de conversión y al finalizar se habrá producido un fichero ogg de audio en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución.

ID	Titulo	
GUI-015	Conversión de un fichero ODP	
	Procedimiento	Resultado
	1. Ejecutar la aplicación mediante el script correspondiente de la	1. Se ejecutará el procedimiento de conversión y al finalizar se habrán

	siguiente manera: reader.bat -config multi-audio-offline.xml -doc presentation.odp o reader.sh -config multi-audio-offline.xml -doc presentation.odp	producido tantos ficheros de audio como páginas tenga el documento odt en el path especificado en la configuración, con el nombre especificado en la configuración incluyendo la fecha de ejecución y un contador que indicará el número de página al que corresponde cada fichero.

Capitulo 6 Objetivos conseguidos

Tal y como se ha señalado en la introducción de la memoria del presente proyecto se han indicado los objetivos que se habían especificado para el proyecto eran los siguientes:

- Estudio de la disponibilidad actual de herramientas TTS para el formato ODF
- Estudio del formato ODF y selección de los elementos mas importantes de dicho formato
- Definición de un documento XSLT que permita transformar el contenido del fichero ODF en un formato utilizable por la herramienta TTS seleccionada
- Estudio de las herramientas de síntesis TTS existentes
- Estudio de los formatos de audio Libre
- Definición de un documento de configuración en formato XML
- Creación de una aplicación en java con los siguientes módulos:
  - Módulo de configuración
  - Módulo de extracción de texto del documento odf
  - Módulo de generación de audio mediante herramientas TTS
  - Módulo de conversión de audio a formato de audio libre

En principio, todos los objetivos se han cumplido en una gran medida a excepción del objetivo de la definición de un documento XSLT que ha sido descartado debido a que se han encontrado diversas librerías para extraer elementos de documentos odf mucho mas versátiles y apropiadas para el desarrollo de la aplicación.

Capitulo 7 Mejoras y Trabajo futuro.

Aunque todos los objetivos especificados para el proyecto se han cumplido, ciertos detalles inesperados encontrados durante la fase de desarrollo del proyecto han producido que sean necesarias mejoras o que algunos puntos no hayan podido ser desarrollados en la profundidad esperada en un principio. Dichos puntos de mejora son los siguientes:

- **Integración de una librería nativa de TTS.** La idea inicial para el proveedor TTS offline era la utilización de la librería freeTTS que, entre otras funcionalidades, permite integrar voces MBROLA. A la hora de integrar el desarrollo se descubrió que freeTTS tiene disponibles e integradas solo 3 voces en inglés y que para añadir nuevas voces había que hacer una modificación bastante simple pero que, dado que el proyecto estaba abandonado, requería decompilar la librería. Además del trabajo necesario, se incumplía claramente la licencia de la librería original. Estos problemas llevaron a la adaptación de eSpeak.
- **Ejecución de la aplicación en MacOSX.** Inicialmente se había propuesto la creación de una aplicación que fuera completamente funcionar el Linux, Windows y MacOSX. Solo se ha cumplido parcialmente dado que en MacOSX no funciona el proveedor offline debido a que ha sido imposible integrar eSpeak.
- **Desactivación de la extracción del texto alternativo de los objetos.** Todas las imágenes y elementos de un documento de texto tienen la posibilidad de contener un texto alternativo. La aplicación extrae ese texto y lo convierte en texto pero no es posible desactivarlo.
- **Extracción de texto de las formulas.** Las formulas matemáticas insertadas en un documento ODF se gestionan mediante referencias a otros documentos xml con formato MathML. Debido a una limitación de la librería, no es posible obtener información de dichos elementos MathML sin extraer los ficheros del empaquetado del documento que se está tratando por lo que no ha sido posible obtener ninguna información relativa a formulas.
- **Signos de puntuación, acentuación y entonación.** En general, se aprecia una falta de corrección a la hora de leer frases que contienen signos de puntuación tanto en el proveedor offline como en el proveedor online siendo este problema bastante mas notable en el proveedor online. Un ejemplo claro de este problema son las siguientes frases:
  - Me enfadé y se echó a reír; le amenacé y dio dos pasos atrás.
  - Me enfadé y se echó a reír le amenacé y dio dos pasos atrás.
  - ¡Me enfadé y se echó a reír le amenacé y dio dos pasos atrás!

  Tal y como se puede observar hay diferencias de puntuación entre cada una

de ellas hasta el punto de que la segunda y la tercera son frases incorrectas debido a la falta del símbolo de puntuación apropiado.

A la hora de convertirlas a audio, utilizando el proveedor online no es fácil encontrar alguna diferencia de pronunciación entre la primera y la segunda frase y la lectura de la tercera frase comienza leyéndose el literal “Símbolo de exclamación” y finaliza con el mismo literal sin modificar la entonación de la frase. Además, el proveedor lee la palabra “punto” al final de cada frase.

Utilizando el proveedor offline es fácil apreciar las diferencias de pronunciación entre la primera y la segunda, pero no hay cambio de entonación al introducir los símbolos de exclamación.

- **Problemas de entonación en las voces offline.** Tal y como se ha descrito ya en el documento, las voces del proveedor TTS offline son las incluidas por defecto en el empaquetado de eSpeak. La entonación en español y catalán es bastante robótica y con tendencia a no ser demasiado preciso en la acentuación. Una posible tarea futura es la investigación de este punto. Es posible que creando una configuración específica para cada voz se pueda paliar en cierta manera el problema. También se pueden integrar voces MBROLA o crear una o varias voces específicas para la aplicación.
- **Problemas de entonación en las voces online.** Tal y como se ha descrito ya en el documento, las voces del proveedor TTS online son las publicadas por el servicio TTS de google. Para este tipo de voces no es posible realizar ninguna mejora e incluye el problema del límite de conversión de 100 caracteres. Es, por tanto, recomendable encontrar otra solución online entre todas las disponibles en el mercado.
- **Número de lenguajes disponibles.** Aunque tanto eSpeak como Google TTS proveen un gran número de idiomas el proyecto se ha centrado en la utilización de solo tres de ellos (español, catalán e inglés) dado que para tratar algunos elementos como, por ejemplo, las tablas se necesita configuración extra para cada idioma. Es decir, cuando la aplicación encuentra una tabla introduce la palabra TABLA antes de comenzar a leer, la palabra CELDA cuando comienza a leer una celda nueva y la palabra FILA cuando comienza una fila nueva. Esta necesidad hace que haya que crear configuraciones especiales para cada idioma, por lo que en el futuro se deberían crear nuevos idiomas y nuevas configuraciones.
- **Interfaz de la aplicación.** Actualmente la interfaz está desarrollada usando java Swing que es un biblioteca gráfica disponible desde 1995 y que aporta todos los elementos necesarios para dotar de cierta usabilidad a la aplicación pero que en ciertas facetas de interacción presenta algunas lagunas. Sería importante investigar nuevas posibilidades de desarrollo de la interfaz.
- **Aplicación móvil.** Dado que tanto eSpeak como el servicio TTS de google están disponible para aplicaciones móviles Android, sería deseable portar el código de la aplicación a una aplicación nativa para terminales android.

- **Reproducción de audio desde la interfaz.** Sería deseable contar con una funcionalidad que reprodujera los ficheros en un pequeño reproductor integrado o los abriera en un reproductor externo.
- **Creación de una herramienta para crear/modificar configuraciones.** Aunque en la actualidad la aplicación permite modificar la configuración sin tener que modificar el fichero xml, una herramienta visual que permitiera crear configuraciones facilitaría el trabajo.

Capitulo 8 Conclusiones

El proyecto text-to-speech para open document supone un ejemplo de la importancia de la existencia de herramientas libres para la transmisión del conocimiento en diversos ámbitos dado que este proyecto no habría sido posible sin la utilización de estándares y formatos abiertos.

Text-to-speech para open document tiene implicaciones que transcienden al ámbito de la universidad que en principio podría parecer el propio del proyecto. Este, es aprovechable para tareas propias del aprendizaje y la enseñanza, para personas discapacitadas, instituciones turísticas, etc.

Desde el punto de vista profesional, el autor del proyecto ha visto incrementados sus conocimientos en diversas herramientas, entornos, estándares y formatos que resultan útiles para el desarrollo laboral como son java, xml, librerías de audio, parseado de documentos, etc.

Desde el punto de vista personal, el autor del proyecto ha profundizado en formatos de audio y en programación multi-plataforma lo que, junto con big data, la materia en la que trabaja actualmente, le ha llevado a comenzar un proyecto de reconocimiento y comparación de patrones en la música comercial.

El uso de la tecnología de voz y Text-to-Speech ha acercado al autor del proyecto a proyecto relacionados con el IoT (Internet of things) tales como IoT surfboard que utiliza dicha tecnología como apoyo para desarrollos y prototipos relacionados con el mundo de la domótica.

Además, se pretende comenzar a poner en marcha diversos proyectos que, basándose en el conocimiento generado por el mismo, permitan ampliar, mejorar o adaptar a nuevas tecnologías el desarrollo del proyecto original incluyendo nuevas funcionalidades o alguna de las sugeridas en la sección de mejoras

En definitiva, el proyecto ha facilitado la adquisición de ciertos conocimientos que han incrementado la valía profesional del autor y que le han animado a comenzar nuevos retos en el campo de la tecnología de los que espera obtener un desarrollo importante tanto personal como profesional.

Capitulo 9 Agradecimientos

Llegados al momento de la entrega del proyecto fin de carrera me gustaría mostrar mi agradecimiento a todas las personas que me han ayudado o se han interesado por mi durante estos tres largos, pero interesantes, años.

Quiero comenzar agradeciendo a Vanessa Gálvez Sillero, toda la luz de mi vida, que me haya iluminado el camino y lo haya hecho mucho mas fácil a pesar de todo el esfuerzo que hemos necesitado. También quiero mencionar en este punto a Manuel Vélez Gálvez, Nuestro hijo. Aunque no has nacido aun, ya eres parte de mi vida. Espero que algún día leas esto y te sientas orgulloso.

Me gustaría continuar con mi familia. Manuel Vélez De Tena, Dolores Martínez Carmona, María Dolores Vélez Martínez, Elisa Vélez Martínez y Sergio Molina Garcia. Muchas gracias por todo el apoyo y toda la motivación que he recibido de vosotros. Muchas gracias. Os quiero.

También quiero agradecer a Gerard Enrique Manonellas toda la ayuda, los consejos y las indicaciones con las que me ha facilitado el desarrollo del proyecto

No me quiero olvidar de mis amigos Juan Ignacio Jaen Mariñas, por haberme animado a hacer este segundo ciclo aunque no le hiciera mucho caso, Antonio Luis Lora Silva y Santiago Martín Palomo García por todo el tiempo que hemos pasado hablando del proyecto y todos los ánimos que me han dado.

Capitulo 10 Bibliografía

[1] Sintesis del Habla – TTS (s,f) En Wikipedia. Consultado el 25 de Febrero de 2016 en la dirección: https://es.wikipedia.org/wiki/S%C3%ADntesis_de_habla

[2] XML (s,f) En Wikipedia. Consultado el 25 de Febrero de 2016 en la dirección: https://es.wikipedia.org/wiki/Extensible_Markup_Language

[3] Formato OpenDocument (s,f) En Wikipedia. Consultado el 25 de Febrero de 2016 en la dirección: https://es.wikipedia.org/wiki/OpenDocument

[4] XSLT (s,f) En Wikipedia. Consultado el 25 de Febrero de 2016 en la dirección: https://es.wikipedia.org/wiki/Extensible_Stylesheet_Language_Transformations

[5] James H. Read Text Extension for Open Office. Open Office extensions. Consultado el 15 de Marzo de 2016 en la dirección: http://extensions.services.openoffice.org/en/project/read-text

[6] Angel White. Open Office Text To Speech. Freecode webpage. Consultado el 15 de Marzo de 2016 en la dirección: http://freecode.com/projects/ootts/

[7] University of Edinbugh. WordTalk: Free text-to-speech plugin for Microsoft word. Consultado el 15 de Marzo de 2016 en la dirección: http://www.wordtalk.org.uk/home/

[8] Zamzar, Free Online file conversion. Consultado el 15 de Marzo de 2016 en la dirección: http://www.zamzar.com/

[9] Blindness solutions, JAWS. Windows Screen Reader. Fredomscientific. Consultado el 15 de Marzo de 2016 en la dirección: http://www.freedomscientific.com/Products/Blindness/JAWS

[10] Orca Project, Gnome project. Gnome Wiki, Consultado el 15 de Marzo de 2016 en la dirección: https://wiki.gnome.org/Projects/Orca

[11] Michael Brauer, Open Document Format For Office applications. Oasis open format. Consultado el 15 de Marzo de 2016 en la dirección: https://www.oasis-open.org/committees/download.php/12572/OpenDocument-v1.0-os.pdf

[12] Go Language Web Page. Goland.org. Consultado el 18 de Marzo de 2016 en la dirección:https://golang.org/

[13] Python WebPage, Python Software Foundation. Python.org. Consultado el 18 de Marzo de 2016 en la dirección: https://www.python.org/

[14] Java Language Webpage, Oracle Corporation, java.com. Consultado el 18 de Marzo de 2016 en la dirección: https://www.java.com/

[15] Groovy, A multi-faceted language for the java platform. Apache software foundation. Consultado el 18 de Marzo de 2016 en la dirección: http://www.groovy-lang.org/

[16] Kotlin, a Statically typed programming language for the JVM, Android and the

browser. JetBrains. Consultado el 18 de Marzo de 2016 en la dirección: https://kotlinlang.org/

[17] Bytecode (s,f) En Wikipedia. Consultado el 19 de Marzo de 2016 en la dirección: https://en.wikipedia.org/wiki/Bytecode

[18] C Sharp (Programming language) (s,f) En Wikipedia. Consultado el 19 de Marzo de 2016 en la dirección: https://en.wikipedia.org/wiki/C_Sharp_(programming_language)

[19] .NET platform. Microsoft corporation. Consultado el 21 de Marzo de 2016 en la dirección: https://www.microsoft.com/net

[20] What's new in JDK 8. Oracle Corporation. Consultado el 21 de Marzo de 2016 en la dirección: http://www.oracle.com/technetwork/java/javase/8-whats-new-2157071.html

[21] Apache maven project, Apache Software Foundation, Consultado el 21 de Marzo de 2016 en la dirección: https://maven.apache.org/

[22] GIT Documentation page. Software freedom conservancy, Git-scm.com, Consultado el 21 de Marzo de 2016 en la dirección: https://git-scm.com/doc

[23] Git Hub, How People build software, github.com, Consultado el 21 de Marzo de 2016 en la dirección: https://github.com/

[24] Integración continua (s,f) En Wikipedia. Consultado el 19 de Marzo de 2016 en la dirección: https://es.wikipedia.org/wiki/Integraci%C3%B3n_continua

[25] Jenkins documentation page. Jenkins.io, Consultado el 21 de Marzo de 2016 en la dirección: https://jenkins.io/doc/

[26] Entorno de desarrollo integrado (IDE) (s,f) En Wikipedia. Consultado el 23 de Marzo de 2016 en la dirección: https://es.wikipedia.org/wiki/Entorno_de_desarrollo_integrado

[27] Eclipse IDE. The eclipse foundation, eclipse.org. Consultado el 23 de Marzo de 2016 en la dirección: https://eclipse.org/ide/

[28] NetBeans IDE. Oracle Corporation. Netbeans.org. Consultado el 23 de Marzo de 2016 en la dirección: https://netbeans.org/

[29] IntelliJIdea, Capable and Ergonomic Java IDE. Jet Brains. Consultado el 23 de Marzo de 2016 en la dirección: https://www.jetbrains.com/idea/

[30] Xercers2, Java Parser. Apache Software Foundation. Apache.org. Consultado el 21 de Marzo de 2016 en la dirección: https://xerces.apache.org/xerces2-j/

[31] JDOM, a java-based solution for accessing, and outputting XML, JDOM Project. Consultado el 22 de Marzo de 2016 en la dirección: http://www.jdom.org/

[32] GlassFish >> Project JAXP. Jaxp library a java.net project. Java.net. Consultado el 22 de Marzo de 2016 en la dirección: https://jaxp.java.net/

[33] XML DTD, w3schools.com Consultado el 25 de Marzo de 2016 en la

dirección:http://www.w3schools.com/xml/xml_dtd.asp

[34] ODFDOM – The opendocument api. Apache Incubator. Apache.org. Consultado el 22 de Marzo de 2016 en la dirección: https://incubator.apache.org/odftoolkit/odfdom/

[35] Aspose.words Developer guide. Aspose corporation. Aspose.com. Consultado el 25 de Marzo de 2016 en la dirección: http://www.aspose.com/docs/display/wordsnet/Developer+Guide

[36] JODF. Independentsoft.de. Consultado el 23 de Marzo de 2016 en la dirección: http://www.independentsoft.de/jodf/

[37] JopenDocument a pure java library for OASIS Open Document files manipulation. JopenDocument.org. Consultado el 25 de Marzo de 2016 en la dirección: http://www.jopendocument.org/documentation.html

[38] JODReports, Java OpenDocument Reports, sourceforge.net, Consultado el 25 de Marzo de 2016 en la dirección: http://jodreports.sourceforge.net/?q=node/22

[39] ArtOfSolving jodconverter, sourceforge.net. Consultado el 25 de Marzo de 2016 en la dirección: http://www.artofsolving.com/opensource/jodconverter

[40] ODFXSLTRunner, Simplify the application of XSLT stylesheets to ODF documents. Apache OpenOffice Wiki. Openoffice.org. Consultado el 25 de Marzo de 2016 en la dirección: https://wiki.openoffice.org/wiki/ODFXSLTRunner

[41] mystiqueXML, Framework open source per la generazione automatica di documenti, yacme.com, Consultado el 27 de Marzo de 2016 en la dirección: http://mystiquexml.yacme.com/

[42] ODFEasy, an alternative layer for the ODFDOM API. Sourceforge.net. Consultado el 27 de Marzo de 2016 en la dirección: https://sourceforge.net/projects/odfeasy/

[43] eSpeak document. Espeak text to speech. Sourceforge.net. Consultado el 30 de Marzo de 2016 en la dirección: http://espeak.sourceforge.net/docindex.html

[44] Ispeech, text to speech and speech recognitions in your apps. Ispeech.org. Consultado el 30 de Marzo de 2016 en la dirección: https://www.ispeech.org/developers

[45] The unofficial Google Text-To-Speech API. Techcrunch.com. Consultado el 30 de Marzo de 2016 en la dirección: http://techcrunch.com/2009/12/14/the-unofficial-google-text-to-speech-api/

[46] Java Speech API FAQ, Oracle Corporation. Oracle.com. Consultado el 30 de Marzo de 2016 en la dirección: http://www.oracle.com/technetwork/java/jsapifaq-135248.html

[47] FreeTTS. A speech Synthesizer writter entirely in java. Sourceforge.net. Consultado el 30 de Marzo de 2016 en la dirección: http://freetts.sourceforge.net/docs/index.php

[48] Tail Sound: The java tutorials. Oracle Corpotation. Consultado el 23 de Marzo de 2016 en la dirección: https://docs.oracle.com/javase/tutorial/sound/

[49] Tritonus, an implementation of the java sound api. Consultado el 24 de Marzo de 2016 en la dirección: http://www.tritonus.org/

[50] JAVE, a java wrapper of the ffmpeg project. Sauronsoftware.it. Consultado el 24 de Marzo de 2016 en la dirección: http://www.tritonus.org/ http://www.sauronsoftware.it/projects/jave/

[51] Apache Log4J 2. Apache Logging Services. Apache software foundations. Apache.org. Consultado el 22 de Mayo de 2016 en la dirección: http://logging.apache.org/log4j/2.x/