



**ESCUELA POLITÉCNICA
SUPERIOR DE CÓRDOBA**
Universidad de Córdoba



TRABAJO FIN DE GRADO

Grado en Ingeniería Informática

Aplicación Conversor de Cuestionarios entre XML Test de Moodle y QTI/IMS de CANVAS

Autor

Diego Alejandro Higuera Grisales

Director(es)

Prof. D. Cristóbal Romero Morales

Mayo, 2023



UNIVERSIDAD DE CÓRDOBA



Índice general

1. Introducción	1
1.1. MoodleXML	2
1.2. QTI/IMS	2
1.3. Partes del proyecto	3
2. Definición del problema	5
2.1. Identificación del problema real	5
2.2. Identificación del problema técnico	5
2.2.1. Funcionamiento	6
2.2.2. Entorno	6
2.2.3. Vida esperada	7
2.2.4. Ciclo de mantenimiento	7
2.2.5. Competencia	8
2.2.6. Aspecto externo	8
2.2.7. Estandarización	8
2.2.8. Calidad y fiabilidad	9
2.2.9. Programa de tareas	9
2.2.10. Seguridad	10
3. Objetivos	11
4. Antecedentes	13
4.1. Respondus	13
4.2. Plugins Moodle	13
4.3. QTIVIEWER/CCReader	14
4.4. GetMarked.ia	14
4.5. Código abierto	14
5. Restricciones	17
5.1. Factores Dato	17
5.2. Factores Estratégicos	17
5.2.1. Lenguaje de programación	18
5.2.2. Entorno de desarrollo	18
5.2.3. Desarrollo de la documentación	19
6. Recursos	21
6.1. Humanos	21
6.2. Hardware	21
6.3. Software	21

Capítulo 1

Introducción

Los grandes cambios que hemos vivido en la sociedad y las personas en los últimos tiempos se han visto reflejados tanto en nuestros hábitos, como formas de consumir formación. En concreto el impulso del e-learning y la formación online, superando los obstáculos de tiempo y espacio, donde se utilizan diferentes plataformas como herramientas de aprendizaje, las cuales han revolucionado la forma en que educadores y estudiantes interactúan y colaboran en entornos educativos virtuales. Entre las plataformas online para el aprendizaje más extendidas a nivel mundial se encuentran Moodle y Canvas LMS.

En este contexto, aunque la tecnología ha evolucionado rápidamente facilitándonos hacer varias cosas de nuestro día a día, existen los desafíos de (*la evaluación para el aprendizaje*) y cómo implementarla en un ambiente online. Las evaluaciones son usadas para determinar el conocimiento de los estudiantes y su dominio en un tema, así como también para identificar áreas de mejora y así poder adaptar la clase o hacer cursos personalizados.

Una de las prácticas mas usadas para la evaluación online son *los cuestionarios*, esta evaluación se realiza de forma dinámica gracias a los ordenadores que pueden realizar la tarea de selección de preguntas y la posterior evaluación en un instante. Una característica única de los cuestionarios en línea es que, el orden de las preguntas y las opciones de respuestas se pueden presentar de forma aleatoria, para que los estudiantes no reciban el mismo cuestionario. Cualquier cuestionario usado en una clase tradicional puede ser fácilmente adaptado en un cuestionario en línea. Las actividades de calificar los cuestionarios y comparar el desempeño de los estudiantes son muy fáciles de hacer, ya que se llevan a cabo de forma automática por el sistema. Además, los educadores tienen la opción de proporcionar aclaraciones adicionales, instrucciones o explicaciones detalladas en forma de comentarios junto a cada pregunta del cuestionario o cada opción de respuesta. Estos comentarios personalizados pueden ayudar a los estudiantes a comprender mejor el propósito de la pregunta, brindar orientación adicional sobre cómo abordarla y ofrecer

explicaciones claras sobre las opciones de respuesta. [? ?]

Las preguntas de tipo cuestionario pueden tener diferentes estructuras atendiendo a la forma de solicitar la respuesta del usuario y a como e muestra la información o datos de la pregunta. Existen diferentes tipos de cuestionarios, entre los mas utilizados son, verdadero o falso, identificar una o varias respuestas correctas, rellenar espacios en blanco , unión de columnas (emparejamiento), respuestas numéricas entre otros tipos. Los usuarios pueden organizar estas preguntas en bancos de preguntas y utilizarlos para crear pruebas personalizadas. Estos cuestionarios son creados para ser usados en una plataforma o sistema educativo de forma que en muchas ocasiones no pueden ser reutilizados en otras plataformas, ya que no existe un único estándar. Es este el caso vamos hablar de *MoodleXML* de Moodle y *QTI/IMS* usado en Canvas LMS.

1.1. MoodleXML

Moodle XML es un formato de archivo utilizado en el sistema de gestión del aprendizaje Moodle. Moodle es un sistema de gestión del aprendizaje de código abierto que brinda una amplia gama de herramientas para facilitar la enseñanza y el aprendizaje en línea, principalmente en entornos educativos. El formato Moodle XML se utiliza para importar y exportar datos de Moodle, lo que permite a los usuarios compartir contenido, actividades y configuraciones entre diferentes instalaciones de Moodle o con otros sistemas compatibles con el formato XML.

El archivo Moodle XML contiene una estructura jerárquica de datos que representa el contenido de un curso o elementos específicos dentro de un curso, como actividades, cuestionarios, recursos y configuraciones. Puede incluir información sobre el nombre, descripción, valoraciones y otros atributos relacionados con los elementos del curso. Los archivos Moodle XML se pueden crear y editar utilizando herramientas específicas de Moodle, como el editor de actividades de Moodle o mediante herramientas de creación de contenido compatible con Moodle XML. [?]

1.2. QTI/IMS

El formato QTI/IMS es un estándar de intercambio de datos utilizados en sistemas de evaluación educativa en línea. QTI significa Interoperabilidad de Pruebas y Cuestionarios.^{en} inglés, e IMS (IMS Global Learning Consortium) es una organización sin fines de lucro que desarrolla estándares abiertos para la educación y la tecnología. Canvas es compatible con el formato QTI/IMS (IMS QTI), lo que significa que importa y exporta preguntas y pruebas en este formato. Canvas es un sistema de gestión del aprendizaje (LMS) utilizado

por muchas instituciones educativas y organizaciones para la creación, entrega y gestión de cursos en línea.

El formato QTI/IMS utiliza XML (Extensible Markup Language) como lenguaje de marcado para describir la estructura y los atributos de las preguntas y pruebas. Esto permite que el contenido se presente de manera clara y estructurada, lo que facilita su comprensión y reutilización en otras herramientas compatibles con el formato QTI/IMS sin perder su estructura y características. Canvas cuenta con su herramienta de banco de preguntas (Question Bank), que permite a los usuarios crear preguntas y pruebas. [?]

1.3. Partes del proyecto

El proyecto en su totalidad estará formado por 3 manuales: el manual técnico, un manual de usuario y el manual de código.

Este, en concreto, es el manual técnico, documento encargado de explicar detalladamente desde el problema en cuestión, hasta la solución que se le va a dar, pasando por los recursos que se van a utilizar para ello. Su estructura se puede dividir en cinco partes, que a su vez se segmentarán en diversos capítulos:

- **Parte I: Introducción al proyecto.** Aspectos generales del proyecto.
 - *Capítulo 1: Introducción.* Presentación breve del problema que se pretende solucionar y su origen, mostrando las definiciones de aquellos términos que se utilizan frecuentemente a lo largo del manual.
 - *Capítulo 2: Definición del problema.* Definición de los problemas real y técnico de forma clara y concisa.
 - *Capítulo 3: Objetivos.* Relación de los objetivos que se pretenden alcanzar en la aplicación, explicando las funciones que debe cumplir lo que se diseña y apuntando las posibles vías de solución.
 - *Capítulo 4: Antecedentes.* Información relativa a aquellos métodos que tratan de resolver el problema de distribución en planta y los proyectos realizados anteriormente.
 - *Capítulo 5: Restricciones.* Limitaciones en el ámbito del diseño que condicionan la elección de una u otra alternativa, distinguiendo entre los factores dato y los estratégicos.
 - *Capítulo 6: Recursos.* Recursos humanos que intervienen en la elaboración del proyecto, y recursos materiales de hardware y software disponibles que se utilizan para llevarlo a cabo.

- **Parte II: Análisis.** Especificación de lo que debe hacer la aplicación desarrollada.
 - *Capítulo 7: Análisis del sistema.* Análisis del funcionamiento que deberá tener el sistema, la información a manejar, lo que deberá ofrecer la interfaz y los requisitos a satisfacer.
- **Parte III: Diseño.**
 - *Capítulo 8: Diseño del sistema*
- **Parte IV: Pruebas y resultados**
 - *Capítulo 9: Pruebas.* Pruebas a las que ha sido sometido el software durante su desarrollo y pruebas de rendimiento cuando se ha finalizado.
- **Parte V: Conclusiones y Futuras Mejoras**
 - *Capítulo 10: Conclusiones.* Mirada hacia atrás para determinar qué objetivos se han cumplido, cuáles no, y el porqué.
 - *Capítulo 11: Futuras mejoras.* A pesar de que inicialmente todo proyecto trata de ser ambicioso y conseguir un software inmejorable, esto casi nunca se cumple, por lo que este capítulo intentará ser una crítica constructiva para mejorar lo realizado y ampliarlo en el futuro.
 - *Capítulo 12: Distribución temporal.* Plan de trabajo y calendario seguido para el desarrollo del proyecto.

Capítulo 2

Definición del problema

En este apartado se va definir con claridad el problema a resolver. En primer lugar, se identificará el *Problema Real* al que nos enfrentamos desde un punto de vista externo (de un usuario) y de manera superficial. Y, en segundo lugar, se identificará el *Problema Técnico* desde una perspectiva más centrada en su desarrollo mediante la técnica conocida como PDS (*Product Desing Specification*).

2.1. Identificación del problema real

Por un lado Canvas exporta sus cuestionarios en formatos IMS/QTI, pero Moodle no acepta la importación de este formato. Moodle por su lado, recomienda el Formato Moodle XML para importar preguntas, ya que éste permite que se importen la mayor cantidad de datos de la pregunta [?]. Por lo que se necesita una herramienta de transformación entre estos formatos. Así éste Trabajo de Fin de Grado se centrará en el desarrollo de una *aplicación conversora de formatos*, que permita a cualquier profesor exportar e importar los cuestionarios online entre ambas plataformas de aprendizaje y así poder llevar a cabo la reutilización de los bancos de preguntas de otros formatos, así como las preguntas generadas en diferentes formatos (y por lo consiguiente en diferentes plataformas), de forma que se almacene toda la información necesaria para su reutilización.

2.2. Identificación del problema técnico

Siguiendo el esquema de la PDS, a continuación, se expondrán los aspectos técnicos del proyecto con la intención de obtener una descripción formal del problema a resolver.

2.2.1. Funcionamiento

Nos vamos a centrar únicamente en el funcionamiento del módulo de representación. Así pues, se comportará de la siguiente manera:

1. Una vez la aplicación haya generado la conversión, podrá importar directamente el cuestionario en la plataforma correspondiente.
2. Cada cuestionario tendrá su estructura distinta según el tipo, con sus respectivos nombres, marcas y flujo de respuestas.
3. Se guardará las conversiones en una carpeta que estará en la misma localización donde se encuentra el fichero a convertir.
4. Se dará al usuario la opción de elegir el cuestionario a convertir, y cual será el formato final, de QTI/IMS a MoodleXML ó de MoodleXML a QTI/IMS.
5. Debe realizar un tratamiento de errores que evite salidas inesperadas del sistema, evitando la generación de ficheros de salida incoherentes.
6. Y debe ser fácil de usar.

2.2.2. Entorno

En el análisis del entorno de la aplicación que se pretende desarrollar se tendrá en cuenta los siguientes puntos de vista principalmente: entorno de programación, entorno software, entorno de usuario y entorno físico o de trabajo.

- *Entorno de programación:* La aplicación será accesible desde el escritorio, es decir, no hará falta el acceso a Internet ni a navegador. El lenguaje de programación que se utilizará en la implementación es Python haciendo uso de librerías simples y eficientes para analizar y crear datos XML, como lo son `xml.etree.ElementTree` ó `lxml.objectify`.
- *Entorno Software:* Para el correcto funcionamiento de la aplicación serán necesarios los siguientes componentes software:
 - Python 3.7
 - Se recomienda la utilización como sistema operativo cualquier distribución estable del sistema operativo Linux ya que es donde se ha desarrollado el programa, aunque se podrá también ejecutar en Windows y Mac.

- *Entorno de Usuario:* La aplicación que se desarrollará deberá ser lo más intuitiva posible, de modo que pueda ser utilizada por todo tipo de usuarios, aunque no posean conocimientos informáticos.
- *Entorno físico o de trabajo:* Para ejecutarlas, los usuarios deben descargar sus archivos e instalarlos en su máquina. Su instalación no requiere grandes requisitos ya que no ocupa demasiado espacio, pero para su ejecución debe de tenerse en cuenta que debe tener espacio suficiente para guardar los ficheros a convertir y los ficheros finales, convertidos.

2.2.3. Vida esperada

Como se ha comentado, al tratarse de formatos de cuestionarios dependientes de las plataformas que los usan, la vida esperada estará ligada a la aceptación de la estructuras de dichos formatos en las plataformas o a futuras versiones de estos. Por tanto hemos de tener en cuenta que mientras se siga avanzando el tema en cuestión, nos puede llevar a cambiar la metodología que ha sido empleada en el desarrollo de este proyecto, en un plazo de tiempo que puede ser breve o extenso.

2.2.4. Ciclo de mantenimiento

En caso de que el módulo se considere de utilidad durante un período largo de tiempo este deberá ser llevado a cabo por programadores informáticos y se hará atendiendo a las necesidades generadas, pudiendo ser de tres tipos:

- *Perfeccionamiento:* Se basará en mejorar los aspectos que se consideren oportunos o crear nuevas funcionalidades a la aplicación, requeridas por los usuarios del sistema o por los autores de cursos. Estas mejoras pueden consistir en aumentar tipos de cuestionarios a convertir, facilitar el mantenimiento del sistema para posibles cambios futuros, etc.
- *Adaptación:* Conjunto de actividades que se realizarán para adaptar la aplicación al entorno tecnológico, como se comentaba en el apartado anterior, adaptarlo a posibles cambios en el formato de las estructuras que usa para generar las conversiones.
- *Corrección:* Corregir errores que puedan aparecer en su uso diario, no descubiertos hasta el momento, como pueden ser fallos de procesamiento o de implementación.

2.2.5. Competencia

El problema que abordamos es conocido por aquellos usuarios de ambas plataformas que han tenido que rehacer los cuestionarios ya que existen pocas soluciones efectivas ó de pago. Una máxima en el desarrollo de software es no reinventar la rueda, por eso nuestra idea no yace en aportar algo que ya existe; nuestra idea es ofrecer algo que destaque y cumplir con los requisitos y expectativas del usuario de manera efectiva y eficiente. Pensamos que los cuestionarios online son un elemento muy potente para la evaluación de un alumno, eso unido a la posibilidad de crear varios tipos de preguntas y poderlas usar en diferentes plataforma se reflejará en una experiencia de usuario que marcará la diferencia.

En el punto de antecedentes se mencionarán algunas herramientas similares que hayan servido para llevar a cabo este proyecto.

2.2.6. Aspecto externo

En cuanto a la interfaz gráfica de usuario (GUI) se pretende que tenga un diseño atractivo, intuitivo y sencillo de manejar, de forma que el usuario no tenga que leer el manual de usuario ya que los botones seguirán un patrón coherente en toda la aplicación de escritorio, expresando así cada uno su función clara y concisamente, para poder obtener las conversiones sin muchas complicaciones, que es donde radica la importancia de este proyecto.

2.2.7. Estandarización

El visualizador está programado en Python haciendo uso de la librería *tkinter*. Librería bastante extendida y documentada, que no debe plantear, en principio, ningún problema. El código fuente será lo suficientemente claro y legible, estando correctamente documentado para que en un futuro cualquier programador pueda entenderlo y realizar los cambios y/o mejoras que crea convenientes.

Los formatos que vamos a convertir, ambos están estandarizados, por un lado MoodleXML, este formato se basa en XML (eXtensible Markup Language) este lenguaje está completamente estandarizado por el W3C (World Wide Web Consortium), por lo que tampoco deberá suponer ningún problema para la ejecución. Además se aplican otros como: SCORM (Sharable Content Object Reference Model) y IMS Common Cartridge. Por otro lado, QTI es un estándar de IMS (Interoperability Standards for Educational Technology) que se utiliza para describir preguntas y pruebas en formato electrónico. QTI permite la creación y entrega de evaluaciones en línea y su posterior evaluación automática. Se aplican otros como IMS Content Packaging o IMS Metadata.

2.2.8. Calidad y fiabilidad

El objetivo es garantizar un alto nivel de excelencia y confiabilidad llevando a cabo exhaustivas pruebas, con diferentes tipos de preguntas, para identificar posibles puntos o componentes críticos, con el fin de minimizar la probabilidad de errores durante el funcionamiento del software. En caso de que los usuarios introduzcan errores (por ejemplo, con ficheros no validos), se busca gestionarlos adecuadamente.

2.2.9. Programa de tareas

Fase Inicial

A lo largo de esta fase se llevarán a cabo las actividades relacionadas con el estudio para la elaboración del proyecto, que son:

1. Estudio del lenguaje de desarrollo python.
2. Estudio del estándar XML.
3. Estudio del estándar QTI
4. Estudio de las librerías xml.etree.ElementTree y objectify para el manejo de ficheros de lenguajes de marca.
5. Estudio de sistemas educativos y aplicaciones gestoras de cuestionarios.

Fase de Diseño y Desarrollo

En esta fase de ingeniería se procederá al diseño y desarrollo de la solución software, y entre las tareas se encuentran:

1. Estudio de las variables que participan en el sistema:
 - a) Variedad de estructuras XML.
 - b) Variedad de tipos de preguntas.
 - c) Parámetros modificables.
 - d) Verificación y conservación de la información de los cuestionarios.
2. Diseño y desarrollo del sistema.
3. Programación de las funcionalidades.
4. Integración de las funcionalidades en la interfaz
5. Comprobar que el formato resultante es válido

Fase de Prueba y Documentación

En esta fase se llevarán a cabo las actividades correspondientes a las pruebas de la solución desarrollada. Estas pruebas se pueden clasificar en dos categorías.

1. Prueba de Unidad o de Caja blanca: Este tipo de pruebas se realizará durante el periodo que se dedique a la codificación de la aplicación, en la que irán surgiendo diversos errores y se deberán ir corrigiendo mediante la aplicación de pruebas de este tipo.
2. Prueba Funcional o de Caja negra: Este tipo de pruebas se centran en el estudio de la especificación del software, análisis de las funciones que debe realizar, de las entradas y de las salidas.

El resultado de las pruebas nos permitirá la depuración de la solución para obtener una versión final y depurada del prototipo. Finalmente, se desarrollará la documentación correspondiente a la Memoria del Proyecto.

2.2.10. Seguridad

Para el apartado de seguridad se aplican los siguientes criterios

- El programa se podrá ejecutar desde cualquier ordenador con sistema operativos como Windows, Linux y Mac.
- Al ser una aplicación de escritorio, los ficheros a convertir y los convertidos, estarán almacenados localmente en el dispositivo que ha ejecutado el programa, por lo que el nivel de privacidad de los datos será establecido por el propio usuario, quien se encargará de proteger el acceso a ficheros o directorios bajo su juicio.
- Incluye una validación de ficheros de entradas de datos, de extensión xml y verificación de permisos de lectura.
- Actualizaciones y parches, es decir, la seguridad de la aplicación también implica mantenerla actualizada con las últimas correcciones de seguridad y parches. Esto implica seguir buenas prácticas de gestión de versiones y poder permitir futuras actualizaciones periódicas para corregir vulnerabilidades conocidas.
- Se obviarán los medios de seguridad contra copias, ya que se trata de software de libre distribución.

Capítulo 3

Objetivos

De acuerdo a la identificación real y técnica del problema, que se ha realizado en el capítulo anterior, a continuación se expondrán todos los objetivos funcionales que se pretenden alcanzar con el desarrollo de este proyecto. El objetivo principal de este proyecto es realizar una aplicación de escritorio que convierta un tipo de cuestionario de un formato XML, el mismo tipo de cuestionario pero en otro formato XML, (MoodleXML a QTI/IMS, ó viceversa) favoreciendo así el uso de cuestionarios en las plataformas que usan estas estructuras. A continuación, vamos a especificar dicho objetivo descomponiéndolo en una serie de subobjetivos que será necesario cumplir para llevar a cabo el proyecto:

- Estudio del estándar MoodleXML y QTI/IMS.
- Estudio de la librería *xml.etree.ElementTree* y *objectify*.
- Estudio de otras herramientas software para el desarrollo del prototipo. Esto incluye para el desarrollo de la interfaz.
- Desarrollar una aplicación sencilla que permita cubrir la necesidad de conversión de cuestionarios en los formatos mencionados.
- La aplicación deberá generar y entender los cuestionarios bien formados de las diferentes estructuras a los que convierta.
- Permitir que el usuario pueda elegir el cuestionario a convertir, cual será el formato final y que el archivo se guarde en el dispositivo.
- Se busca crear una aplicación con una estructura modular que facilite la incorporación de nuevas funcionalidades en el futuro. La aplicación debe ser diseñada de manera escalable, de modo que sea posible añadir módulos adicionales que resulten útiles para el proceso de conversión, y que no hayan sido contemplados en este proyecto.

- Creación de una documentación completa y clara, en la cual se editará el manual de usuario del software desarrollado, así como las memorias y documentación técnica necesaria.

Capítulo 4

Antecedentes

Aunque existen en el mercado multitud de aplicaciones relacionadas con la conversión de formatos, en este apartado, se mencionarán diferentes aplicaciones comerciales relacionadas con el tema a tratar, que han servido de ayuda para concebir una idea más clara del resultado final que se quiere obtener, si nos centramos principalmente en la característica de conversión de los formatos MoodleXML y QTI/IMS, en ambos sentidos. A continuación, se mostrará un breve resumen de su funcionalidad:

4.1. Respondus

Respondus es una compañía que ofrece software y soluciones para la creación, administración y seguridad en exámenes en línea. Es un programa de pago con 30 días gratuitos, pero cuando se probó, solo se obtuvieron resultados insuficientes, ya que convertía un formato QTI a un formato *.doc*, y no especificaba cual era la respuesta correcta. [?]]

4.2. Plugins Moodle

Existen algunos plugins en la comunidad Moodle que realizan la importación de QTI, que desgraciadamente están ahora obsoletos y ya no se mantienen [?]], como son los siguientes:

- *Questionmark QML Importer (Alpha)*: Es una extensión que permite importar cuestionarios y evaluaciones creados en dicho formato, se mantuvo hasta 2018. En nuestras pruebas con varias preguntas de opción múltiple en el estándar QTI 1.2, ninguna de estas preguntas pudo ser importada sin errores con la ayuda de este plugin. Es importante destacar que el término *Alpha* indica que el plugin se encuentra en una fase inicial de desarrollo y es posible que no cuente con todas las características completas o pueda presentar algunos errores.

- *moodle-qformat_imsqti21plugin*: Es una extensión diseñada para el sistema de gestión de aprendizaje Moodle. Este plugin permite la importación y exportación de cuestionarios y evaluaciones en el formato IMS QTI 2.1. Cuando probamos este plugin con las preguntas del estándar QTI 2.1, ni la importación ni la exportación de Moodle XML a QTI 2.1 funcionaron.

4.3. QTIViewer/CCReader

Es una herramienta de software que proporciona una interfaz la cual permite visualizar y leer contenido en formato QTI, lo que facilita la revisión y la interacción con el contenido evaluativo, pero no permite hacer una conversión entre formatos. [?]

4.4. GetMarked.ia

Es una plataforma y servicio en línea que se especializa en la creación, administración y calificación de exámenes y evaluaciones en entornos educativos y empresariales. Utilizando tecnología de inteligencia artificial y análisis de datos, GetMarked.ai automatiza y optimiza el proceso de evaluación, proporcionando una solución eficiente y precisa. Sus características y funcionalidades principales se centran en:

- La creación y diseño de exámenes personalizados.
- La administración de exámenes en línea.
- La corrección y calificación automática de respuestas.
- La generación de informes y análisis detallados.
- La integración con sistemas de gestión del aprendizaje (LMS) y otras herramientas educativas.

GetMarked.ai se enfoca en brindar a educadores, instituciones académicas y organizaciones una forma eficiente de evaluar el aprendizaje, reduciendo la carga administrativa y mejorando la calidad y la eficacia de las evaluaciones. Esta es una permite también la conversión entre formatos, pero es de pago. [?]

4.5. Código abierto

Durante la investigación de herramientas nos encontramos con códigos en el repositorio online GitHub, y otros en páginas web, los cuales sirvieron de guía para realizar nuestro código.

- Generador de preguntas XML de Moodle: Es un módulo para Python. Con este módulo, podemos generar fácilmente un conjunto de preguntas de opción múltiple e importarlas a Moodle. [?]
- text2qti: Convierte archivos de texto sin formato basados en Markdown en cuestionarios en formato QTI (versión 1.2), que pueden ser importados por Canvas y otro software educativo, usando el lenguaje Python. [?]
- moodle2qti [?]
- Procesamiento de datos QTI en Python [?]

Capítulo 5

Restricciones

En este capítulo se expondrán todas las restricciones, o factores limitativos, existentes en el ámbito del diseño que condicionarán el desarrollo de nuestro proyecto. Estos factores limitativos, según su tipo, se pueden clasificar en dos grupos: *Factores Dato* y *Factores Estratégicos*.

5.1. Factores Dato

El problema que se ha planteado apenas presenta barreras que limiten las posibles soluciones, como pudieran ser las relativas al hardware, el software, los plazos, etc. Esto no quiere decir que sean inexistentes:

- Limitaciones en el plazo de entrega. El presente proyecto tiene el plazo máximo de entrega el 10 de Junio de 2023. Para esa fecha, se espera que el proyecto esté totalmente finalizado.
- Limitaciones técnicas. Estas restricciones pueden estar relacionadas con las capacidades técnicas del equipo o las especificaciones, debido a que anteriormente desconocía totalmente el problema presentado.
- Limitaciones económicas. El coste para llevar a cabo el proyecto es mínimo; debido a la utilización de software estándar disponible gratuitamente y hardware propio.

5.2. Factores Estratégicos

Siguiendo los objetivos descritos en el Capítulo 3, a continuación, identificaremos los factores estratégicos que, si bien condicionarán las distintas propuestas alternativas, pueden ser objeto de modificación o elección en uno u otro sentido.

5.2.1. Lenguaje de programación

Se ha optado por usar el lenguaje de programación Python, que a día de hoy, es un lenguaje de programación versátil y potente ya que cuenta con una gran cantidad de bibliotecas y módulos que facilitan la tarea de convertir formatos XML, por ejemplo, se hará uso de las siguientes bibliotecas estándares:

- `xml.etree.ElementTree` ya que proporciona una forma fácil de analizar y manipular documentos XML. [?]
- Biblioteca `lxml` la cual contiene el módulo `objectify`, el cual es altamente eficiente en el procesamiento de XML y XHTML.[?]

Además, Python es multiplataforma, escalable y tiene una comunidad activa [?]. Todo ello irá integrado en una aplicación programada en Python haciendo uso del módulo `tkinter`.

5.2.2. Entorno de desarrollo

El entorno de trabajo será un PC con Ubuntu 20.04.5 LTS, la versión de este sistema operativo cuenta con soporte a largo plazo, lo que la convierte en una opción estable y confiable. [?]

Para creación y modificación del código fuente se ha utilizado la herramienta *Visual Studio Code* para desarrollar la aplicación de escritorio, debido a que ofrece una amplia gama de características, herramientas y extensiones que pueden mejorar la eficiencia y la productividad en el proceso de desarrollo. Además VS Code tiene una excelente integración con Python, lo que proporciona soporte para la depuración de código Python directamente desde el editor, lo que facilita la identificación y solución de problemas en la aplicación de escritorio. [?]

Por último, para la aplicación de escritorio se ha usado *Tk*, una biblioteca de código abierto escrita en C, la cual Python incluye en su librería estándar, en el módulo `tkinter`, que permite interactuar con Tk para desarrollar aplicaciones de escritorio en Python de forma rápida, de modo que cualquier programador con una mínima base de Python puede comenzar rápidamente a crear aplicaciones gráficas profesionales y luego distribuirlas vía herramientas como `cx_Freeze` o `PyInstaller`, que se integran muy bien con Tk. [?]

En su momento, se decidió usar Python debido a que es el mismo lenguaje en el que estábamos programando las funcionalidades del conversor, y facilitaba enlazar la interfaz gráfica.

5.2.3. Desarrollo de la documentación

La documentación será creada utilizando LaTeX a través de la herramienta en línea Overleaf. Inicialmente, se consideró utilizar un editor de texto convencional como Microsoft Word, pero esta opción fue descartada al darnos cuenta de que no se podría alcanzar un nivel de control tan avanzado sobre el texto ni obtener resultados de la misma calidad que con el sistema elegido.

Para crear los diagramas necesarios que respalden el análisis y diseño de la aplicación bajo UML (Lenguaje Unificado de Modelado), utilizaremos la herramienta Draw.io en su versión web. Esta herramienta fue elegida debido a que es gratuita y proporciona la mayoría de los tipos de diagramas necesarios. Con ella, podremos representar gráficamente la estructura y las relaciones de la aplicación de manera efectiva. [1]

Capítulo 6

Recursos

Para llevar a cabo el desarrollo del proyecto se utilizarán una serie de recursos humanos, hardware y software los cuáles se detallarán a continuación

6.1. Humanos

El proyecto será realizado por el alumno de Ingeniería Informática Diego Alejandro Higuera Grisales, dirigido y coordinado por: Cristóbal Romero Morales, profesor perteneciente al departamento de Informática y Análisis Numérico de la Universidad de Córdoba.

Dado que este proyecto es un trabajo de fin de carrera, el estudiante encargado del desarrollo asumirá los roles de analista y programador. El director del proyecto tendrá la responsabilidad de orientar al estudiante durante su desarrollo y revisar periódicamente su trabajo para asegurar que avance de manera adecuada y correcta.

6.2. Hardware

El proyecto será desarrollado en los equipos propiedad del alumno.

- **Portátil:** Intel Core i7 8550U, up to 2000 MHz, 8 GB de RAM, Intel(R) UHD Graphics 620, SSD 256 GB.
- **Impresora:** Hp DeskJet 2720.

6.3. Software

En cuanto a los recursos software que se utilizarán para la realización del proyecto:

- Sistemas operativos usados:

Windows 10.

Linux (Ubuntu 20.04.4 LTS).

- Herramientas de desarrollo de la aplicación: Python 3, API de objectify y API de ElementTree.
- Herramienta para la edición de código: editor de código fuente Visual Studio Code (VS Code).
- Herramientas para el desarrollo de la documentación y diagramas:
 - \LaTeX . Sistema de composición de textos usado para realizar la documentación, bajo la herramienta de edición en línea Overleaf.
 - Draw.io. Aplicación web para la realización de diagramas (UML, ER, etc.).