



Universidad de Jaén

Escuela Politécnica Superior de Linares

Trabajo Fin de Grado

HERRAMIENTA PARA ANOTACIÓN MANUAL DE ASPECTOS EN CORPUS TEXTUALES

Alumno: Antonio Jesús López Ros

Tutor: Prof. Dña. M^a Dolores Molina González

Depto.: Ingeniería de Telecomunicación

Julio, 2020

ÍNDICES DE CONTENIDO

Resumen	7
Abstract	8
1 Introducción	9
1.1 Antecedentes	10
1.2 Breve introducción al proyecto	11
1.3 Motivación	11
1.4 Breve resumen de productos obtenidos	12
1.5 Estructura del proyecto	13
2 Objetivos	14
3 Estado del arte	15
3.1 Introducción a la minería de datos	15
3.2 Procesamiento de lenguaje natural	16
3.3 Análisis de Sentimientos	17
3.4 RDFaCE	18
3.5 BRAT	18
3.6 TextPro	19
4 Materiales y Herramientas de desarrollo	21
4.1 Materiales Utilizados	21
4.1.1 Software alternativo a usar	21
4.1.2 Lenguaje de programación JAVA	23
4.1.3 MariaDB	24
4.1.4 SQL	25
4.1.5 XML	25
4.2 Herramientas de desarrollo	27
4.2.1 Netbeans	27
4.2.2 Workbench	27
5 Descripción y desarrollo	29

5.1 Autenticación del sistema	29
5.2 Entrada de texto por base de datos o por ventana principal	30
5.3 Funcionamiento de la pantalla principal	32
5.3.1 Procesamiento de frases	33
5.3.2 Targets	33
5.3.3 Entities/Aspect category	34
5.3.4 Polaridad - Intensidad	35
5.4 Bases de datos	35
5.5 Exportación a XML	37
5.6 Pruebas de funcionamiento del software	38
5.7 Problemas encontrados en el desarrollo	39
5.8 Líneas de futuro	40
6 Conclusiones y limitaciones	42
7 Anexos	44
7.1 Anexo I: Manual de instalación del entorno de trabajo NetBeans	44
7.2 Anexo II: Manual de instalación para el MySQL Workbench	51
7.3 Anexo III: Ejemplo práctico y manual de usuario del programa	60
7.4 Anexo IV: Flujos de datos del programa	69
8 Bibliografía	70

ÍNDICE DE IMÁGENES

Figura 3.1	Ejemplo de anotación con RDFaCE	18
Figura 3.2	Ejemplo de anotación con BRAT	19
Figura 3.3	Ejemplo de anotación de normalización con Wikipedia	19
Figura 3.4	Arquitectura de TextPro	20
Figura 4.1	Arquitectura Cliente-Servidor	21
Figura 4.2	Arquitectura MVC	22
Figura 4.3	Ejemplo estructura XML	26
Figura 4.4	Ejemplo 1 de conexión de Workbench con BBDD	28
Figura 4.5	Ejemplo 2 de conexión de Workbench con BBDD	28
Figura 5.1	Tabla usuario en BBDD	29
Figura 5.2	Ventana de Autenticación	30
Figura 5.3	Credenciales válidas	30
Figura 5.4	Texto en la pantalla principal	30
Figura 5.5	BBDD donde se encuentran los textos	31
Figura 5.6	Ventana de confirmación	31
Figura 5.7	Desplegable con diferentes títulos	31
Figura 5.8	Secciones de la ventana principal	32
Figura 5.9	Procesamiento de sentences	33
Figura 5.10	Carga de un target	33
Figura 5.11	Ejemplo de selección de varios Targets	34
Figura 5.12	Ejemplo de despegable de entitys de medicamentos	34
Figura 5.13	Ejemplo de despegable de polaridad-intensidad	35
Figura 5.14	Ejemplo de texto etiquetado y colores de polaridad-intensidad	35
Figura 5.15	Tabla usuario	36
Figura 5.16	Tabla texto	37
Figura 5.17	Estructura de salida XML	38
Figura 7.1.1	Anexo I: Página principal de NetBeans	44

Figura 7.1.2 Anexo I: Página principal de NetBeans, descargas	45
Figura 7.1.3 Anexo I: Download versión 8.2	45
Figura 7.1.4 Anexo I: Diferentes paquetes de NetBeans 8.2	46
Figura 7.1.5 Anexo I: Barra de instalador	46
Figura 7.1.6 Anexo I: Instalador de NetBeans 8.2	47
Figura 7.1.7 Anexo I: Licencia de Producto	47
Figura 7.1.8 Anexo I: Instalación de NetBeans	48
Figura 7.1.9 Anexo I: Crear nuevo proyecto en NetBeans	48
Figura 7.1.10 Anexo I: Creación de nuevo proyecto	49
Figura 7.1.11 Anexo I: Estructura del proyecto en NetBeans	49
Figura 7.1.12 Anexo I: Importar librería MariaDB	50
Figura 7.2.1 Anexo II: Enlace de descarga para MySQL installer	51
Figura 7.2.2 Anexo II: Descarga para MySQL installer	52
Figura 7.2.3 Anexo II: Aceptación de términos de licencias	52
Figura 7.2.4 Anexo II: Elección de producto a instalar	53
Figura 7.2.5 Anexo II: Selección de Workbench y MySQL server	53
Figura 7.2.6 Anexo II: Selección de carpeta de instalación	54
Figura 7.2.7 Anexo II: Requerimientos	54
Figura 7.2.8 Anexo II: Instalación completada	55
Figura 7.2.9 Anexo II: Selección de 'Group Replication'	55
Figura 7.2.10 Anexo II: Apartado 'Type and Networking'	56
Figura 7.2.11 Anexo II: Apartado de elección de pass del root user	57
Figura 7.2.12 Anexo II: Finalización de la configuración del producto	58
Figura 7.2.13 Anexo II: Finalización de instalación	58
Figura 7.2.14 Anexo II: Pantalla de inicio de MySQL Workbench	59
Figura 7.3.1 Anexo III: Conexiones en MySQL Workbench	60
Figura 7.3.2 Anexo III: Tabla usuario en Workbench	61
Figura 7.3.3 Anexo III: Ventana de autenticación	61
Figura 7.3.4 Anexo III: Credenciales validas	61

Figura 7.3.5 Anexo III: Ventana principal	62
Figura 7.3.6 Anexo III: Tabla texto	63
Figura 7.3.7 Anexo III: Ventana para cargar texto por BBDD	63
Figura 7.3.8 Anexo III: Despegable para cargar texto BBDD	63
Figura 7.3.9 Anexo III: Procesar sentences	64
Figura 7.3.10 Anexo III: Seleccionamos un target en el texto	64
Figura 7.3.11 Anexo III: Etiquetamos el target elegido	65
Figura 7.3.12 Anexo III: Ventana de dar de alta	65
Figura 7.3.13 Anexo III: Aspect Category	65
Figura 7.3.14 Anexo III: Polaridad-Intensidad	66
Figura 7.3.15 Anexo III: Palabra de ejemplo etiquetada	66
Figura 7.3.16 Anexo III: Ejemplo 1 salida XML	67
Figura 7.3.17 Anexo III: Ejemplo 2 salida XML	68

RESUMEN

En este trabajo se ha desarrollado una aplicación web para que los usuarios de esta puedan etiquetar las palabras de un documento sobre las características o aspectos de un cierto producto o servicio, etiquetándose con una categoría y dentro de una polaridad (muy positiva, positiva, neutra, negativa y muy negativa).

Esta aplicación se inscribe dentro de la disciplina de análisis de sentimientos (AS). Se analizan partes del texto estudiado para que el usuario de la aplicación pueda etiquetar aquellas características que definen el producto o servicio. Posteriormente con este texto etiquetado lo exportamos a formato XML para que sea estructurado y se pueda utilizar en una web o una base de datos.

En la memoria se van a introducir términos necesarios para entender el programa, como son el procesamiento del lenguaje natural (PLN), la minería de datos (MO) y el AS.

La arquitectura de la aplicación Web siguen un patrón modelo-vista-controlador (MVC), en el que hemos utilizado el lenguaje de programación JAVA en el entorno de desarrollo NetBeans, y para la base de datos (BBDD), se ha utilizado SQL en el gestor Workbench, en la conexión del programa con la BBDD, se usa la librería MariaDB. Para la salida de datos se utiliza el lenguaje de marcas XML.

El programa cuenta con autenticación para que solo pueda acceder el usuario autenticado, y no permite el acceso a usuarios que no sean definidos con anterioridad en la BBDD.

ABSTRACT

In this project, a software application has been developed so that its users can label the words of a document about the characteristics or aspects of a certain product or service, it is labeling with a category and within a polarity (very positive, positive, neutral, negative and very negative).

This application is a part of the discipline of sentiment analysis (AS). Parts of the studied text are analyzed so that the user of the application can label those characteristics that define the product or service. Later with this tagged text we export it to XML format so that it is structured and can be used on a website or a database.

The memory will introduce terms necessary to understand the program, such as natural language processing (PLN), data mining (MO) and AS.

The architecture of the Web application follows a model-view-controller (MVC) pattern, in which we have used the JAVA programming language in the NetBeans development environment, and for the database (BBDD), SQL has been used in the Workbench manager, in the connection of the program with the DDBB, the MariaDB library is used. XML mark language is used for data output.

The program has authentication so that only the user can access, and does not allow access to users that are not previously defined in the BBDD.

1 INTRODUCCIÓN

El análisis de texto a través de la opinión de una persona es un tema que ha levantado un interés alto en la comunidad científica en estos años. Cada vez se utiliza más por parte de Gobiernos, empresas privadas, incluso por organizaciones públicas, como pueden ser los órganos sanitarios o educativos.

Esta información es muy importante y valorada, ya que la podríamos utilizar a la hora de comprar un producto o servicio, para saber la opinión que genera entre los usuarios, o incluso, cuando decidimos poner en venta o promocionar un servicio o producto podemos hacer pruebas y conocer de primera mano el grado de satisfacción de los clientes antes de que el producto salga a la venta. Son muchas las posibilidades gracias al AS en un texto.

El AS es el estudio computacional de las opiniones, sentimientos y emociones que se expresan en un texto, en nuestro caso será el estudio de las opiniones del usuario sobre las características que tiene un cierto producto, como puede ser una película o un medicamento. Para entender mejor el AS tenemos que estudiar qué tipos de opiniones pueden existir en un texto. Existen 2 tipos, la opinión directa que es cuando calificamos un objeto o atributo, y la opinión comparativa que como su nombre indica, es cuando se intenta establecer una relación entre varios objetos de orden cualitativo. También las opiniones pueden ser explícitas cuando se expresan en el texto tal cual o implícitas y no aparecen en la oración y se tienen que deducir.

En definitiva, como definición formal de opinión, se puede decir que es una valoración hecha por el usuario, y esta valoración está polarizada positivamente, neutra o negativamente sobre un objeto o un atributo del objeto.

Por eso este AS se utiliza en nuestra sociedad más a menudo, ya que son muchos los interesados en conocer las múltiples opiniones que los usuarios tienen acerca de servicios, temas de interés, productos, organizaciones, sucesos, etc.

Hoy en día cada vez existen y se producen más y más datos en Internet a través de las páginas web, las redes sociales, los blogs, incluso opiniones de usuarios que

realizan cuando compran un producto o un servicio en Internet, en definitiva, vivimos en un mundo en el que la opinión de las personas es muy importante, y también se produce de manera masiva en la red. Que haya tanta cantidad de datos y tantos usuarios expresando su opinión en un texto hace muy difícil, por no decir imposible, que los datos puedan ser analizados y procesados de forma manual.

En este punto, es cuando introducimos la utilidad de herramientas que sean capaces de analizar los sentimientos que hay en un texto, en la actualidad existen herramientas capaces de hacer esto, y permiten analizar varios textos de manera totalmente automática.

Normalmente en el AS se delimita a nivel de documento, párrafo o frase, y tiene como objetivo estudiar una entidad en el texto e identificar sus características y propiedades, para después determinar la polaridad que tiene, si esta entidad es positiva, negativa o neutra para el usuario que analiza el corpus de texto.

En cuanto a la investigación de herramientas de AS, nos encontramos con el gran problema de que dichas investigaciones se hacen en la mayoría de las ocasiones utilizando el inglés como idioma principal. Lo que en el caso de que, si el texto está escrito en español, hace que sea difícil el tratamiento de las palabras negativas o el análisis a nivel de aspecto, dado que el español es un idioma con mucha riqueza léxica, y en el que una sola palabra puede tener varios significados distintos para la persona que lo está analizando.

En este punto, dado lo descrito anteriormente, nos encontramos con la necesidad de desarrollar sistemas que sean capaces de extraer, clasificar y hacer una presentación de las diferentes opiniones del texto, para esto surgen los sistemas automáticos de AS o el MO. Esta última, es un conjunto de técnicas que se utilizan para la extracción y explotación de repositorios de datos o BBDD. [1,2,3]

1.1 Antecedentes

El mundo del AS es muy amplio, y se han ido descubriendo diferentes formas de extraer los diferentes aspectos en una opinión. Desde el conteo de frases o palabras que sean positivas y negativas para después hacer un recuento general, una lista de aspectos que sea predeterminada, hasta modelos estadísticos usando similitud semántica o el

conocido modelo estadístico de 'Latent Dirichlet Allocation' (LDA). Este modelo estadístico consiste por ejemplo en que, si hay un conjunto de observaciones, estas observaciones se pueden sustituir por palabras en documentos, y cada documento es una mezcla de un número de categorías o tópicos, entonces la aparición de cada palabra del documento se debe a una de la categorías o tópicos a la que el documento pertenece. [4]

En nuestra herramienta para analizar los aspectos de una opinión debemos de etiquetar estos aspectos y después darle una polaridad e intensidad a cada uno de ellos. Así podremos definir los aspectos con una etiqueta que defina el valor o la categoría en la que se está agrupando, y con la polaridad e intensidad nos permite saber el grado de positividad o negatividad de este. Por ejemplo, vamos hablar de un teléfono móvil, el teléfono puede tener varios aspectos con valoración propia como puede ser la batería del móvil, la pantalla, el procesador o el sistema operativo, cada uno de estos aspectos tendrá su propia polaridad e intensidad.

1.2 Breve introducción al proyecto.

Como hemos mencionado en el apartado anterior una de las maneras más efectiva de extraer aspectos de un texto es etiquetarlos para poder darles un significado real y posteriormente valorarlo mediante una opinión encontrada en el documento o texto.

En este proyecto hemos realizado un programa que le permite al usuario hacer esto y vamos a dedicar gran parte del proyecto a analizar el programa. En breves palabras, dado un texto o frase cualquiera, la interfaz gráfica nos permite seleccionar palabras clave, a las cuales encasillamos con una etiqueta y posteriormente con una polaridad e intensidad, de esta forma el software extrae la información de aspectos del texto y estos datos que obtenemos podemos estructurarlos.

1.3 Motivación

En esta sección se va a exponer las razones para que sea tan importante el estudio y desarrollo del análisis de sentimientos a nivel de aspectos y por ende, este proyecto.

La motivación que nos lleva a que cada vez se esté investigando más sobre este tema, es que el mundo de los datos se está volviendo más extenso y denso, los datos que

dejamos en internet crecen más rápidamente conforme van pasando los años, y necesitamos poder almacenar estos datos y estructurarlos, a su vez la información por parte de los usuarios de estos datos también es muy importante, ya que vivimos en una era en la que la opinión de las personas respecto a un producto o un servicio es lo que la mayorías de empresas u organizaciones tienen en cuenta para vender su producto o servicio o para elaborar un plan de marketing acerca de este.

Nos podemos encontrar empresas que quieran hacer un estudio sobre su producto o servicio, gobiernos que quieran estudiar a la población para hacer campaña electoral, hasta recomendaciones de los usuarios acerca de películas, libros o videojuegos. En nuestro proyecto, nos centramos en el análisis de las características de un producto u objeto, como puede ser una película o un medicamento.

Por esta razón el análisis de sentimientos basado en aspectos (ABSA) es hoy en día una de los campos de investigación con más futuro en el panorama del AS.

Por eso, es importante el análisis de los aspectos de una opinión. En este proyecto nos hemos centrado en la elaboración de un software para facilitar el etiquetado de textos para el usuario del programa, y posteriormente exportarlo en una estructura de datos XML. [1,2,3]

1.4 Breve resumen de productos obtenidos

En este proyecto se presentan 3 productos diferentes:

- El código fuente del programa para el etiquetado de corpus. Este estará formado por un conjunto de scripts en JAVA, necesarios para el funcionamiento del programa, entregándose un archivo ejecutable .jar.
- Fichero con extensión SQL con la BBDD MariaDB creada para almacenar los textos y usuarios del programa.
- La memoria del trabajo. En este documento se presentará toda la documentación y la exposición detallada del programa y todo lo relacionado con el desarrollo de la aplicación.

1.5 Estructura del proyecto

En este apartado se va a detallar los diferentes capítulos que se van a tratar en este trabajo de fin de grado:

- **RESUMEN**: aquí se va a explicar brevemente cual es el objeto de este trabajo.
- **1 INTRODUCCIÓN**: se hace una pequeña introducción del trabajo, antecedentes y la motivación que ha llevado a desarrollo de esta herramienta.
- **2 OBJETIVOS**: aquí se definen los objetivos que tiene que cumplir el proyecto para que se puede conseguir desarrollar la herramienta de forma completa.
- **3 ESTADO DEL ARTE**: se definen algunos conceptos clave para entender el análisis a nivel de aspecto, y también se proponen las herramientas mas famosas para este análisis.
- **4 MATERIALES Y HERRAMIENTAS DE DESARROLLO**: se describen los materiales usados para el desarrollo de la herramienta, así como el entorno y los lenguajes de programación usados.
- **5 DESCRIPCION Y DESARROLLO**: se trata del diseño y la descripción de la herramienta para el análisis a nivel de aspectos. También se mencionan los problemas obtenidos durante el desarrollo y las líneas de futuro.
- **6 CONCLUSIONES Y LIMITACIONES**: se exponen las ideas finales respecto a los resultados obtenidos del desarrollo de la herramienta, así como las limitaciones de la misma.
- **7 ANEXOS**: se detallan los manuales de instalación de NetBeans y de MySQL Workbench. Además, recoge un manual de usuario siguiendo un ejemplo de la herramienta. También hay un Anexo con el flujo del programa.
- **8 BIBLIOGRAFIA**: se muestran las reseñas de las fuentes bibliográficas que hemos empleado en la memoria.

2 OBJETIVOS

En este proyecto se propone desarrollar una herramienta que facilite el proceso de etiquetado manual por parte de un anotador a nivel de aspectos. Esta es una tarea que es muy costosa de realizar en el tiempo.

La entrada de datos se obtiene por 2 vías: seleccionando y cargando el texto mediante la BBDD con la tabla texto o directamente copiándolo en la pantalla principal.

Hay que tener en cuenta que le estamos introduciendo como datos de entrada un texto de opinión sobre las características de un cierto producto o servicio, el cual el usuario del programa selecciona los TARGETS y se van etiquetando según su propia opinión con ENTITYS. El usuario puede decir si la característica o aspecto es muy positivo, positivo, neutro, negativo o muy negativo en base a sus propios criterios. Por ejemplo, la palabra 'dolor de cabeza' es un target elegido y su ASPECT CATEGORY o ENTITY es dolencia, para un usuario puede ser 'negativo' y para otro usuario 'muy negativo'.

A través de la herramienta se pueden definir o crear los distintos aspectos que se vayan necesitando para etiquetar las palabras del documento. En el programa solo se pueden etiquetar palabras que aparezcan en el texto, no se pueden introducir targets que no estén en el documento.

Otro de los objetivos claros del proyecto es que el programa sea intuitivo y fácil de manejar para cualquier usuario, y como se puede observar solo habría que marcar un target del texto que queremos analizar y etiquetarlo con un aspect category o entity y su polaridad-intensidad.

Un aspecto que hemos considerado importante es la autenticación del usuario para entrar al programa, para que un usuario que no esté autorizado no pueda acceder a este, y cualquier usuario que haya sido dado de alta en la BBDD si pueda entrar sin problemas.

Uno de los objetivos principales que queríamos con este software era la salida de datos en formato XML, para que después se pueden almacenar de forma más ordenada y estructurada. Con este formato de salida conseguimos que el usuario del programa pueda obtener los datos etiquetados del texto para posteriormente si quiere utilizarlos como él quiera, ya sea en páginas web o para BBDD.

3 ESTADO DEL ARTE

En este apartado vamos a definir algunos términos que necesitamos entender para el desarrollo de nuestra herramienta como son la MO, el PLN y el AS, y también vamos a mostrar algunas aplicaciones o plugins que existen actualmente en el mercado y que desarrollan una actividad parecida a la de este programa. He encontrado varias aplicaciones como BRAT, RDFaCE o TextPro que explicaremos a continuación.

3.1 Introducción a la minería de datos

Nos encontramos con el problema de que los antiguos sistemas de almacenamiento no son capaces de procesar ni gestionar tanta cantidad de datos, y se necesitan sistemas y técnicas que nos faciliten la extracción de datos con mayor rapidez, procesamiento y gestión.

Para explorar y extraer la información de las grandes BBDD y repositorios nace la idea del MO o datamining. Para hacerlo de forma automática o semiautomática, se necesitan encontrar patrones repetitivos o tendencias en los repositorios para analizar los datos más rápidamente, aunque para cada caso el datamining puede ser distinto, a gran escala se observan cuatro etapas principales:

- Delimitación de los objetivos: esto trata de decidir qué objetivos y datos queremos recoger en la BBDD, y como los delimitamos.
- Procesado de datos: esta es la etapa en la que se gasta más tiempo, en esta, seleccionamos, reducimos y minimizamos las BBDD para que sea más fácil la extracción.
- Determinación del modelo: aquí sacamos una primera visualización gráfica a través de las estadísticas que nos proporcionan los algoritmos de Inteligencia Artificial (IA).
- Análisis de los resultados: esta es la última etapa, hay que verificar que los resultados se corresponden a los objetivos que teníamos delimitados, y si estos son coherentes y correctos, son los que se le envían al usuario o cliente.

Es muy importante que se desarrolle la MO ya que según algunos estudios de la corporación internacional de datos con siglas IDC (el último se llama 'Big Data, Bigger Digital Shadows and Biggest Growth in the Far East', patrocinado por EMC) [6], presentan que cada dos años se duplica el volumen de datos en Internet, por poner un ejemplo, esta cantidad equivale a que cualquier usuario envíe 3 tweets por minuto durante 181.061 años, y alrededor de un 90% de los datos que existen en la red no siguen ninguna estructura.[5,6]

3.2 Procesamiento de lenguaje natural

El PLN abarca las técnicas que se utilizan para comunicar a los ordenadores con las personas mediante lenguajes naturales existentes, usando técnicas de inteligencia artificial. Hay técnicas para procesar textos, audios, incluso lenguaje de signos.

Sin embargo, el PLN se encuentra con un problema, y es que el lenguaje natural es localmente ambiguo, esto quiere decir que una expresión se puede interpretar de distintas maneras dentro del contexto donde se escriba esta expresión. Además, el lenguaje humano es muy complejo léxicamente ya que una misma palabra puede tener varios significados dependiendo del contexto de la oración o del texto. Por eso es computacionalmente complejo el estudio del PLN, ya que son tareas que aún no han sido sistematizadas, porque no se ha comprendido aún cómo se comporta el lenguaje natural humano, y esto hace muy complejo a día de hoy, diseñar un algoritmo universal para el procesamiento del lenguaje.

Para ello, se utilizan técnicas para el análisis de la expresión de lenguaje natural, estas técnicas de análisis se agrupan en 4 niveles:

- Análisis Morfológico: consiste en el análisis de cada una de las palabras dentro de una oración, con este análisis conseguimos extraer unidades léxicas compuestas, rasgos de flexión y otros.
- Análisis sintáctico: este es el análisis de la estructura sintáctica de una frase o texto, esto se consigue gracias al estudio gramatical de la oración.
- Análisis semántico: se hace a través de modelos estadísticos o métodos de representación del conocimiento como las ontologías, y consiste en obtener el significado de la oración o frase.

- Análisis pragmático: es el análisis de lo que no se incluye en la oración, el contexto que tiene la expresión.

Estos procesos tienen multitud de beneficios en la sociedad, como puede ser la agilización del trabajo, los conocidos 'chatbots', traducciones automáticas y en el que nos centramos en este proyecto, el AS. [7]

3.3 Análisis de Sentimientos

El AS o minería de opiniones, se refiere al término que se ha explicado en el apartado 1 de introducción, viene dado por el análisis de opiniones, sentimientos o emociones que se expresan en un texto.

En el AS tenemos el problema de que el lenguaje humano es complejo y sobre todo en español, es difícil automatizar un sistema o un proceso para que distinga entre los diferentes matices gramaticales, variaciones culturales, el tono de la oración o frase. Por esto cuando se hace AS, en la mayoría de los casos automáticos se tienen algunas limitaciones, y no es 100% fiable, no es una tarea simple que las expresiones humanas entren en una serie de categorías, como pueden ser, negativo-positivo, tienen sus propios matices.

En definitiva, el AS nos facilita que se pueda analizar mejor una opinión, pero también tiene sus limitaciones. Esta es un área dentro del PLN que hemos descrito anteriormente.[1,2]

3.4 RDFaCE

Es una herramienta para etiquetar la información contenida en un texto, basándose en el paradigma WYSIWYG (what you see is what you get, 'lo que ves es lo que tienes'),

en su interfaz vienen definidas un listado de etiquetas disponibles, las cuales se usan para clasificar los datos e información del texto según el usuario lo desee. Como curiosidad también pueden etiquetarse imágenes del texto. En la imagen inferior se puede encontrar la ventana para etiquetar. [8]

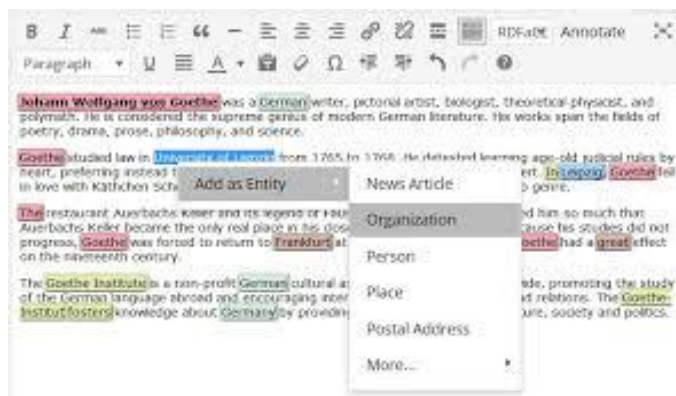


Figura 3.1 Ejemplo de anotación con RDFaCE

3.5 BRAT

BRAT es una herramienta en web que sirve para la anotación de textos, de hecho, utiliza la anotación o etiquetado estructurado como es en nuestro caso. Esta es una plataforma open source y se debe ejecutar de forma local para cargar los documentos y trabajar sobre ellos. La web donde se encuentra esta herramienta gratuita es [‘http://brat.nlplab.org/index.html’](http://brat.nlplab.org/index.html)

En esta plataforma se pueden cargar los ficheros TXT que queramos seleccionándolos en nuestro ordenador. Esta herramienta trabaja de forma parecida a nuestro programa ya que nos permite analizar un texto a través de etiquetas. En el archivo de salida obtenemos el identificador de la anotación, el tipo de etiqueta, la posición inicial, la posición final y el texto que contiene, este fichero se guarda en una extensión .ann. Las etiquetas se establecen en un listado que crea el usuario en un fichero aparte llamado ‘annotation.conf’. Nosotros por ejemplo no tenemos este fichero en nuestro programa y podemos definir las etiquetas en la propia ventana principal.

Hay 2 tipos de anotaciones en BRAT: la anotación text span y la anotación por relación. La primera de ellas consiste en etiquetar una palabra o conjunto de palabras con una anotación simple, y la anotación por relación lo que hace es crear una relación entre varias anotaciones simples.

También existe la posibilidad de anotaciones de normalización que asocian otras anotaciones con entradas en recursos como puede ser la Wikipedia, como se puede ver en la imagen inferior. [9,10]

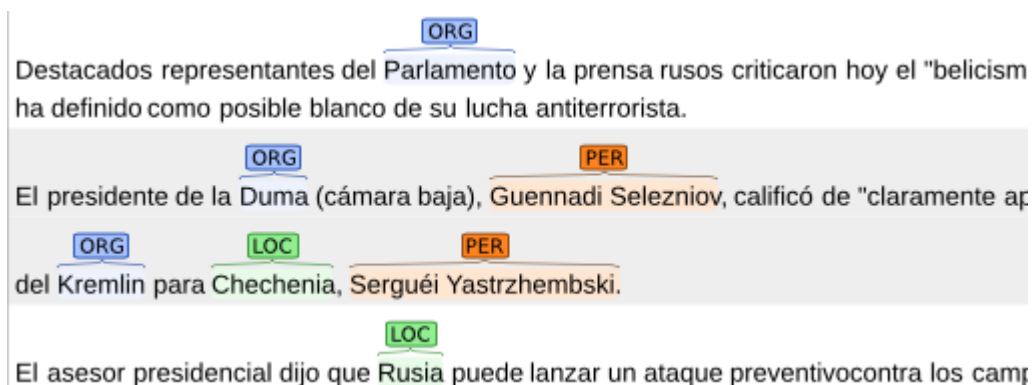


Figura 3.2 Ejemplo de anotación con BRAT

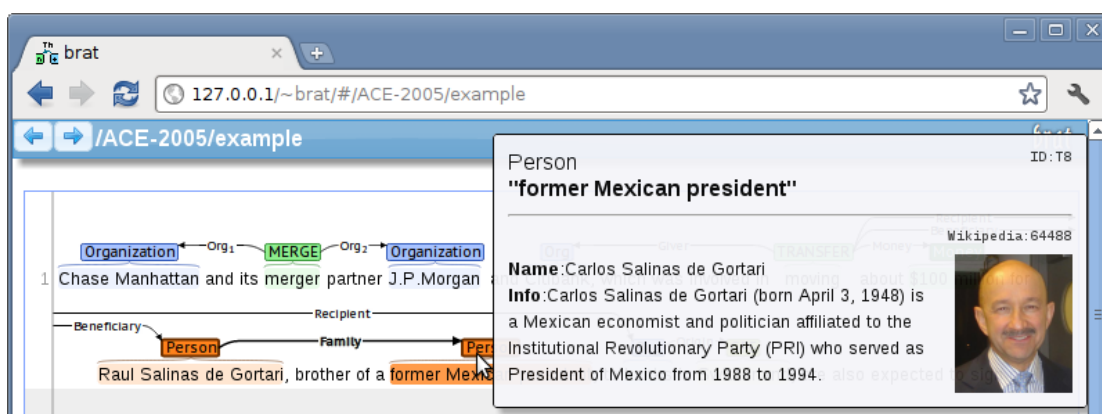


Figura 3.3 Ejemplo de anotación de normalización con Wikipedia

3.6 TEXTPRO

TextPro es un conjunto de herramientas que han sido diseñadas y orientadas a las tareas de PLN, estas son el etiquetado y división de oraciones en un texto, el análisis morfológico de las oraciones, la etiquetación de palabras. La plataforma de TextPro es <http://textpro.fbk.eu/home>

La mayoría de estas herramientas son ofrecidas por programas independientes, pero lo bueno de TextPro es que las reúne en un entorno integrado y además con este

software somos capaces de agregar y crear nuevos componentes. Estas herramientas son simples, modulares y tienen portabilidad a los distintos sistemas operativos.

La arquitectura de TextPro se basa en pipeline (túnel) de procesadores, cada uno de estos procesadores coge los datos de una entrada inicial o del procesador anterior, y envía los datos a la salida del procesador.

TextPro se basa en 9 componentes o herramientas básicas para poder completar su funciones, estas son: cleanPro (limpieza de páginas web), tokenPro (utiliza un token basado en reglas para analizar el flujo de caracteres en el texto), sentencePro (división de frases), morphoPro (análisis morfológico), tagPro (etiquetación), chunkPro (trocear las frases), entityPro (para el reconocimiento de entidades), lemmaPro (dado los análisis morfológicos de morphoPro y las etiquetación de tagPro produce una serie de semantizaciones) y multiwordPro (para el reconocimiento de múltiples palabras). [11]

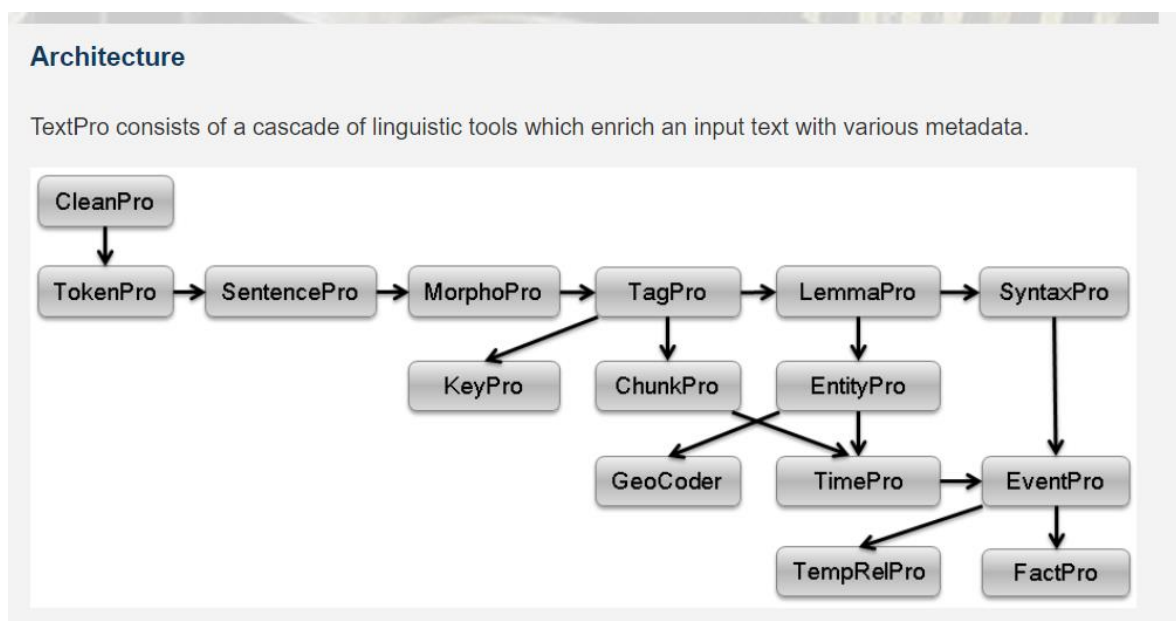


Figura 3.4 Arquitectura de TextPro

4 MATERIALES Y HERRAMIENTAS DE DESARROLLO

En este apartado vamos a explicar los materiales y herramientas de desarrollo que hemos utilizado para poder desarrollar este programa. El lenguaje de programación usado ha sido Java, y para conectarse a la BBDD lo hacemos a través de la librería MariaDB que sirve para gestionar BBDD. El formato elegido para la salida ha sido el lenguaje de etiquetas o de marcas XML, y los entornos que hemos utilizado para construir el programa han sido por un lado el NetBeans como compilador de código y el Workbench para construir las BBDD con lenguaje SQL. A continuación, en los siguientes apartados se va a explicar brevemente en qué consiste cada una de estos materiales y herramientas de desarrollo.

4.1 Materiales Utilizados

4.1.1 Software alternativo a usar.

En este apartado se comentará las posibles arquitecturas y tecnologías con las que desarrollar esta herramienta.

En el caso de las arquitecturas, nuestro proyecto está montado de forma local, y no tiene una arquitectura web, pero sigue el patrón de MVC. Hay 2 posibles arquitecturas web con las que desarrollar la herramienta estas son:

- Cliente-servidor: esta consiste en que el cliente realiza peticiones a otro programa o servidor, y este le responde, esta arquitectura es correcta si queremos que tener nuestra herramienta corriendo en una red de ordenadores.

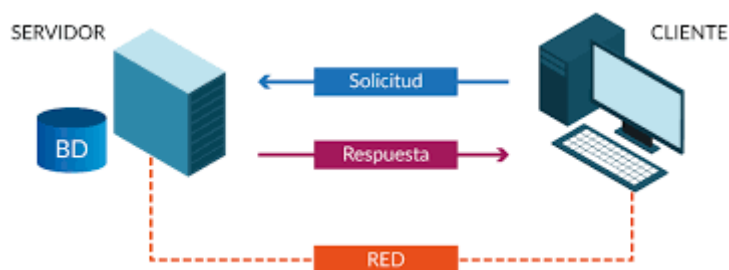


Figura 4.1 Arquitectura Cliente-Servidor

- El modelo MVC mencionado en apartado anteriores, es una arquitectura del programa, que utiliza 3 componentes (modelo, vista y controlador). Esto

permite separar de forma independiente la parte visual de la parte lógica del programa como en nuestra herramienta, la capa de modelo tiene sirve para la gestión de los datos, la capa de vista tiene como función la de representar los datos a través de la interfaz de usuario, y por último la capa de controlador sirve como conexión entre las 2 capas anteriores.

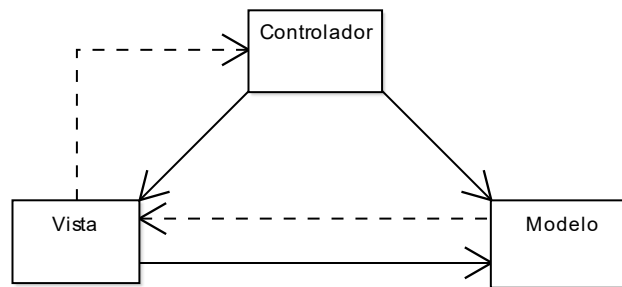


Figura 4.2 Arquitectura MVC

Se podría haber optado por programar en Python, PHP, C++ o C# y hubiese sido igual de válido, pero las librerías de estos eran menos conocidas, y estuve investigando hasta que encontré la librería MariaDB. Esta se adapta muy bien a los requerimientos de mi programa, además de que había manejado con anterioridad el lenguaje JAVA con apache tomcat o NetBeans y me resultaba más fácil de programar, una de las razones de peso por la que se usó JAVA fue que es multiplataforma y corre en Linux/Unix o cualquier SO.

Las tecnologías con las que se podría haber implementado la herramienta eran:

- PHP: es un lenguaje de programación que sobre todo se usa para arquitecturas de cliente-servidor en el lado del servidor, tiene una similitud con la sintaxis utilizada a en C. El nombre PHP es un acrónimo de Hypertext preprocessor, y está pensado para manejar páginas web ya que permite a paginas estáticas convertirse en dinámicas.
- Python: es un lenguaje orientado a objetos como JAVA y se puede utilizar en muchos entornos de programación como NetBeans, es un lenguaje que no necesita compilar el código fuente para su ejecución, funciona a nivel de comandos, con este lenguaje se pueden crear todo tipo de páginas web y de programas que no sean tipo web.

Para la BBDD, se usó el entorno Workbench porque es de los principales gestores de BBDD y además es opensource y es fácil crear las tablas.

4.1.2 *Lenguaje de programación JAVA*

JAVA es una plataforma de desarrollo y lenguaje de programación que sirve para desarrollar cualquier programa que queramos realizar. Una ventaja muy grande de utilizar JAVA es que es multiplataforma, esto significa que funciona en cualquier servidor, dispositivo o sistema operativo, esto en gran medida es gracias a su máquina virtual (JVM) que es la que ejecuta el programa, en mi caso utilizo la versión 8 de esta máquina virtual. En nuestro programa hemos utilizado la librería JDK 1.8 (java development kit) que es un conjunto de utilidades, herramientas, ejemplos y documentación que podemos utilizar para desarrollar una aplicación o proyecto en JAVA, también incluye el compilador y las APIs de JAVA.

Otra ventaja de JAVA es su robustez y que es capaz de gestionar la liberación de memoria de manera automática. Su principal atractivo es que JAVA es un lenguaje de programación que está orientado a objetos, con lo cual es fácil crear cualquier aplicación modulable y reutilizar partes del código. También este lenguaje de programación lleva ya muchos años utilizándose para la implementación de infinidad de programas, por lo cual hay bastantes librerías, bibliotecas y código abierto subido por otros programadores, que se puede utilizar en cualquier momento.

Una de las bibliotecas que se ha usado para la interfaz gráfica de nuestra herramienta ha sido la librería de JAVA llamada Swing incluida con el entorno JDK. Esta es un framework que sigue un patrón MVC para el desarrollo de interfaces graficas de usuario (GUI) con despegables, tablas, botones, cajas de texto, etc. Tiene las siguientes características:

- Se puede personalizar ya que usa diferentes estilos de apariencia al gusto del programador.
- Es extensible a otras clases que se estén desarrollando en el software.
- Se puede utilizar en cualquier desarrollador de programación siempre que tenga el entorno JDK de JAVA.

Para utilizar el Swing tenemos que crear la clase JFrame, contenedor principal, proporciona poder operar y manipular las diferentes ventanas, por eso nuestra clase 'EntradaVentana.java' que es la clase principal de la herramienta y la que se ejecuta, hereda de JFrame.

Con Swing se consigue tener los siguientes componentes, utilizados a la hora de desarrollar la herramienta:

- JButton: sirve para crear un botón simple en la GUI.
- JLabel: crea una etiqueta que salga en el programa.
- JTextField: permite introducir texto en una ventana, pero es solo de una línea.
- JComboBox: es una lista de elementos despegable donde el usuario elige entre las opciones que le demos.
- JTextArea: tiene la misma finalidad que JTextField pero permite varias líneas.
- JScrollBar: es una barra de desplazamiento, que el usuario puede mover.

Para la posición y el tamaño de los componentes descritos anteriormente se utilizan una serie de layout managers, en nuestra herramienta se han utilizado BorderLayout.NORTH, BorderLayout.SOUTH, BorderLayout.EAST y BorderLayout.WEST para indicar dentro del panel toolBarPanel donde se distribuyen las distintas ventanas.

Muchas de estas ventajas las tienen otros lenguajes de programación, pero JAVA es un lenguaje muy utilizado y esto lo convierte en un lenguaje del que se encuentra bastante información de errores o de programación respecto a otros lenguajes.

4.1.3 MariaDB

Este software de código abierto, lo hemos utilizado como sistema gestor de BBDD de nuestro programa, en cuestión, en nuestro programa tenemos la librería mariadb-java-client-2.6.0 que podemos descargar de la página oficial. Este gestor de BBDD surge como sustitución de MySQL, y de hecho cualquier software que antes utilizará MySQL se puede

intercambiar para utilizar este MariaDB. Una ventaja grande de este gestor es que es de código abierto y actualmente se está investigando y desarrollando haciendo que las BBDD tengan mejor rendimiento o simplemente optimizando estas BBDD.

Se utiliza la librería mariaDB en el NetBeans para que el programa pueda conectarse a la BBDD que está creada en el entorno de Workbench.

En el anexo 8.1 que es el manual de instalación del entorno NetBeans, también viene como descargar e importar la librería mariaDB que hemos utilizado.

4.1.4 SQL

El lenguaje SQL sirve para administrar, sustraer, recuperar, consultar, borrar la información y datos de una estructura de BBDD. Este lenguaje de programación permite hacer una gran cantidad de operaciones con las estructuras que tenemos creadas, podemos definir, borrar o manipular los objetos de la estructura de una BBDD.

Una vez creadas las estructuras y la BBDD, este lenguaje nos permite que hagamos consultas y vistas de los objetos almacenados en las tablas, para esto lo queremos en nuestro programa. Por ejemplo, si queremos hacer una consulta para ver que hay en la tabla 'usuario' de nuestro programa, que utilizamos para la autenticación, tenemos la siguiente línea de código que nos permite seleccionar estos datos.

```
'SELECT * FROM proyecto.usuario;'
```

En la tabla 'texto' tenemos la siguiente línea de código SQL para extraer datos de nuestra tabla.

```
'SELECT * FROM proyecto.texto;'
```

4.1.5 XML

El XML es un lenguaje de marcas extensible, y lo hemos utilizado como salida de datos de nuestro programa, en realidad no es un lenguaje en sí mismo, sino que sirve para estructurar y representar los datos que nos interesen mostrar. Una de las cosas que nos interesan en nuestra herramienta, es utilizar este formato de salida como entrada para

sistemas de machine learning para que aprendan los datos. Los sistemas que utilizan machine learning tienen como fin el aprendizaje automatizado y que se desarrolle de manera autónoma, es un campo dentro de la inteligencia artificial.

Existen 2 partes en el formato XML, la parte del prólogo donde se define los metadatos administrativos como las instrucciones de procesamiento y el tipo de documento, y la parte principal o del cuerpo, esta parte principal a su vez se divide en 2 tipos, la parte estructural donde se encuentra la organización del XML y la otra parte es el contenido de esta estructura.

El diseño de XML es simple y fácil de usar, además es genérico y se puede utilizar para gran cantidad de aplicaciones como servicios web. Su uso principal es el de representar información estructurada en la web, de forma que esta información o datos puedan ser transmitidos, almacenados, procesados, visualizada por distintos dispositivos y aplicaciones. Otras aplicaciones de XML son intercambiar y publicar contenidos de BBDD de manera estructurada, utilizándose para el formato de mensajes intercambiados por aplicaciones, y para la descripción de metadatos.

Algunos de los componentes de un documento de XML son los siguientes: los elementos que son las piezas claves para la estructura lógica del documento y se cierran con etiquetas, teniendo dentro de los elementos los atributos que nos dan una información adicional del elemento, y también existen las entidades que es la unidad virtual donde se almacena la información.

Vamos a poner un ejemplo de archivo XML para ver sus componentes y su estructura:

```
Documento XML con prólogo
<?xml version="1.0" encoding="ISO-8859-1"?>
<nota>
  <para>Pedro</para>
  <de>Laura</de>
  <titulo>Recordatorio</titulo>
  <contenido>A las 7:00 pm en la puerta del teatro</contenido>
</nota>
```

Figura 4.3 Ejemplo estructura XML.

4.2 Herramientas de desarrollo

4.2.1 NetBeans

En nuestro programa tenemos como entorno de desarrollo el NetBeans IDE 8.2, estando pensado principalmente para utilizarlo con JAVA y existiendo bastantes módulos para extenderlo según nuestros intereses. Este entorno de desarrollo es un software gratuito y se utiliza bastante en el mundo de la programación. El NetBeans IDE soporta el desarrollo de todo tipo de aplicación JAVA, ya sean web, EJB, aplicaciones móviles o J2SE.

Es un framework que simplifica bastante el desarrollo de aplicaciones en JAVA. Con el paquete NetBeans IDE si tiene todo lo necesario para el desarrollo de plugins y aplicaciones sin necesidad de un SDK (kit de desarrollo de software) adicional, además es multiplataforma y lo soportan los diferentes sistemas operativos.

En el anexo 8.1 hay un manual para la descarga e instalación de este entorno de desarrollo que hemos utilizado para nuestro programa.

4.2.2 Workbench

Esta es una herramienta de carácter visual para el diseño de BBDD, que además nos ofrece administración, gestión y mantenimiento de BBDD, siendo bastante intuitivo y fácil de manejar y existiendo la posibilidad de crear tus tablas a través de un componente gráfico.

En la pantalla principal nos permite ver esquema de entidad-relación, esquemas físicos de BBDD, y para cada esquema podemos ver las tablas, las vistas o las rutinas. Para conectarnos a una BBDD definida, necesitamos el hostname que es la dirección local del servidor (en nuestro caso la dirección IP de loopback 127.0.0.1), el puerto que por defecto es el 3306, el nombre de usuario que tenemos creado (en nuestro caso en root), posteriormente nos pedirá una contraseña que hemos definido anteriormente y ya nos conectamos. En las imágenes inferiores se puede observar este proceso de conexión a la BBDD desde workbench.

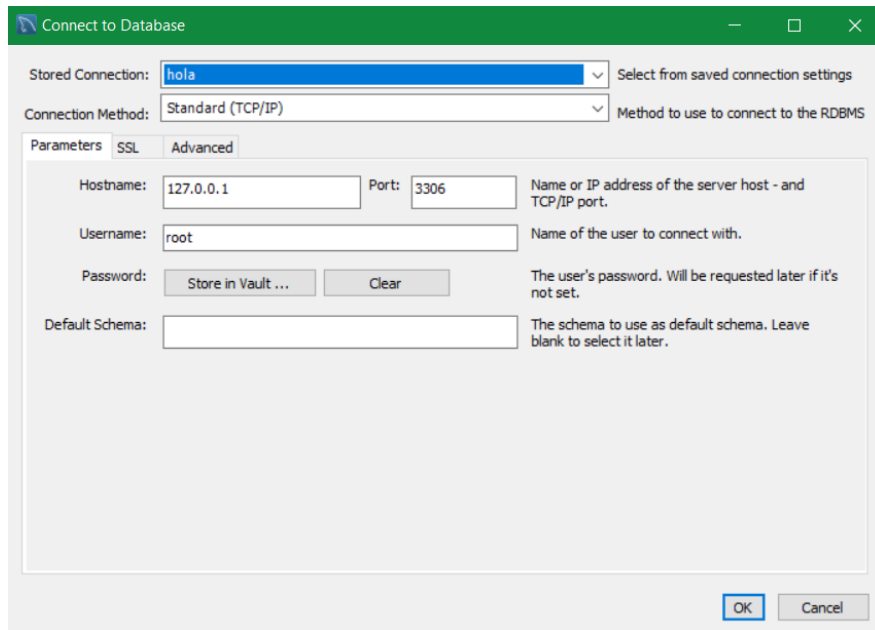


Figura 4.4 Ejemplo 1 de conexión de Workbench con BBDD.

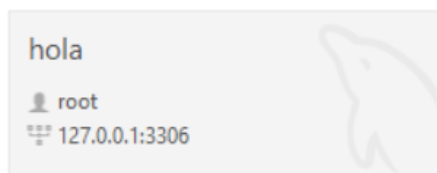


Figura 4.5 Ejemplo 2 de conexión de Workbench con BBDD

5 DESCRIPCIÓN Y DESARROLLO

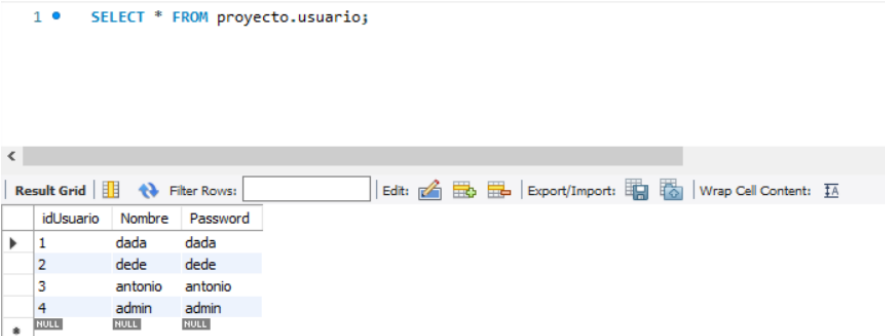
El proyecto que se está tratando en esta memoria, consiste en un software en el que el administrador de este puede elegir o cargar un texto desde una BBDD en una pantalla para posteriormente analizarlo.

Para analizarlos contamos con tres aspectos fundamentales como son el TARGET, el ASPECT CATEGORY y la POLARIDAD-INTENSIDAD. Posteriormente, y con el texto analizado por parte del administrador procederemos a exportarlo en formato XML para que sea más fácil de leer por parte de otros usuarios u otros programas.

5.1 Autenticación del sistema.

Para acceder al programa se necesita autenticarse con un usuario y con una contraseña correspondiente. Estos se crean en la BBDD del programa, y puedes crear tantos usuarios como se necesiten con sus correspondientes contraseñas.

Al iniciar el software, te saltara una ventana como la que se puede ver en la imagen inferior de ejemplo, para introducir un usuario y una contraseña y si estos son válidos te da acceso al programa.



```
1 • SELECT * FROM proyecto.usuario;
```

idUsuario	Nombre	Password
1	dada	dada
2	dede	dede
3	antonio	antonio
4	admin	admin
NULL	NULL	NULL

Figura 5.1 Tabla usuario en BBDD.



Figura 5.2 Ventana de Autenticación.

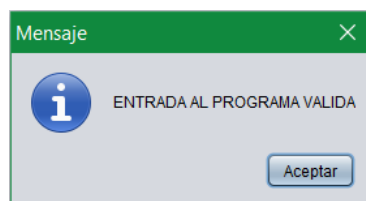


Figura 5.3 Credenciales válidas.

5.2 Entrada de texto por BBDD o por ventana principal.

En este programa hay varias maneras de cargar un texto o una frase, tenemos la opción de copiar y pegarlo en la ventana principal del programa como se puede ver en la imagen inferior en la que hemos pegado en texto en la sección blanca del programa.

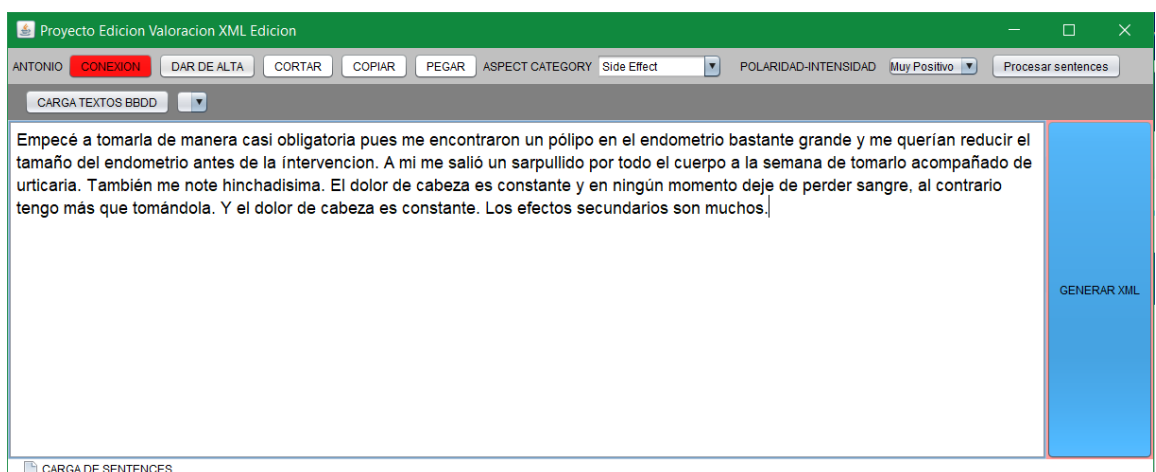


Figura 5.4 Texto en la pantalla principal

La segunda opción que hay es la de que la entrada del texto sea a través de la BBDD. Para cargar el corpus con esta opción, debemos de tener definidos los textos en nuestra BBDD como podemos observar en la imagen inferior. En el ejemplo hemos definido varios textos de medicamentos con sus títulos.

idTexto	Texto	Textolong	Título	ViaForma	FechaCreacion	Fe
0	Excelente medicamento, muy eficaz para solventar crisis de pánico y de angustia de manera rápida. Sus efectos se per...	2	Alprazolam	3	NULL	NULL
1	Mi odontólogo me la recetó luego de extraerme dos muelas del juicio, el período de toma era de 7 días, al cabo del 3er ...	2	Amoxicilina	3	NULL	NULL
2	Empecé a tomar sertralina por ansiedad, a los 2 días de tomarla empecé a sentir problemas estomacales, diarrea, náus...	2	Sertralina	3	NULL	NULL
3	Tomo Citalopram desde ayer por la noche. La primera vez, a las 3 horas sentía mucho mareo, veía todo más lento, me ...	2	Citalopram	3	NULL	NULL

Figura 5.5 BBDD donde se encuentran los textos.

Quando queramos cargar varios textos desde la BBDD tenemos que pulsar en el botón 'CARGAR TEXTOS BBDD', y te saldrá un desplegable con los títulos de los diferentes textos que hemos metido con anterioridad en nuestra BBDD. En la imagen adjunta podemos observar cómo podemos seleccionar los diferentes textos.

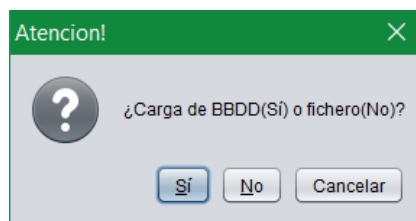


Figura 5.6 Ventana de confirmación.

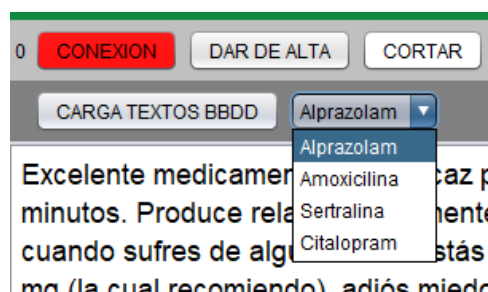


Figura 5.7 Desplegable con diferentes títulos.

5.3 Funcionamiento de la pantalla principal

La pantalla principal del programa está compuesta por varios elementos como el botón de conexión que aparece en rojo y sirve para conectar el programa con la BBDD anteriormente descrita. Otros elementos a considerar son los botones de 'ASPECT CATEGORY', 'POLARITY-INTENSITY' estos son desplegables y se pueden crear en el instante o la sección de 'GENERAR XML'.

El menú principal del programa se divide en cuatro secciones fundamentales, la primera sección es la superior en color gris donde nos encontramos con los botones principales para la conexión, cargar textos de la BBDD, y lo necesario para etiquetar los fragmentos de texto, incluidos los botones de pegar, cortar y pegar por si nos hicieran falta.

A la parte derecha de la pantalla principal en color azul nos encontramos con el 'GENERAR XML' que es la salida del programa una vez tenemos los textos etiquetados, este botón nos muestra los datos en formato XML para que sean más cómodo de tratar y almacenar.

En la sección central se encuentra el cuadro con el texto que estamos analizando, el cual vamos a ir etiquetando los diferentes fragmentos.

En la parte inferior de la pantalla principal nos encontramos con las frases del texto que han sido procesadas para poder analizarlas una por una. En la imagen inferior podemos ver cómo se dividen en cuatros secciones.

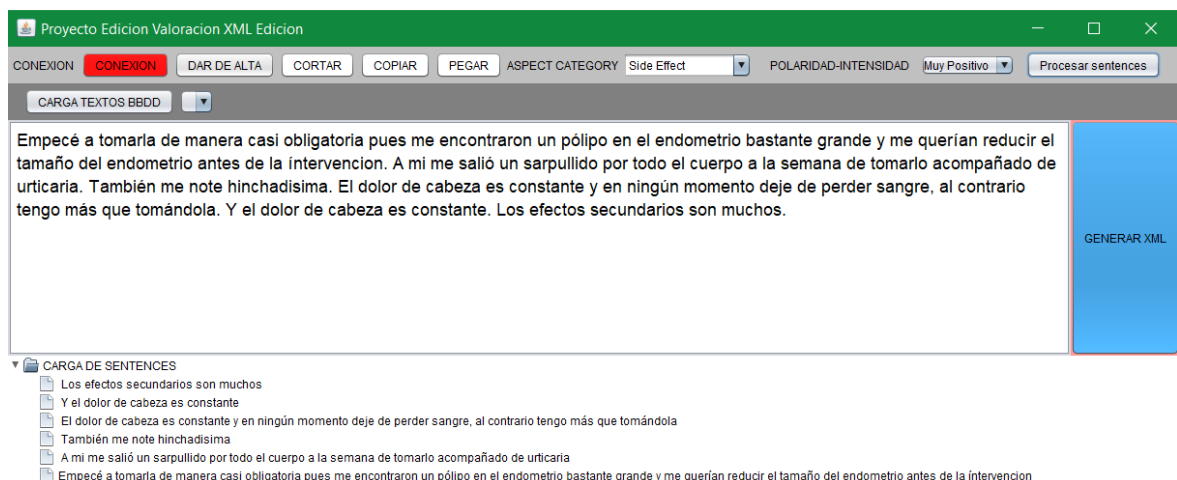


Figura 5.8 Secciones de la ventana principal.

5.3.1 Procesamiento de frases.

Como hemos indicado anteriormente en la parte inferior de la pantalla se muestran las frases que tiene procesadas del texto el programa. Para distinguir una frase de otra, las delimitamos a través de los puntos, por lo tanto, una frase sería los hay entre dos puntos en el texto, excepto la primera frase que no empieza con un punto. En la siguiente figura se puede ver un ejemplo de procesamiento de frases.

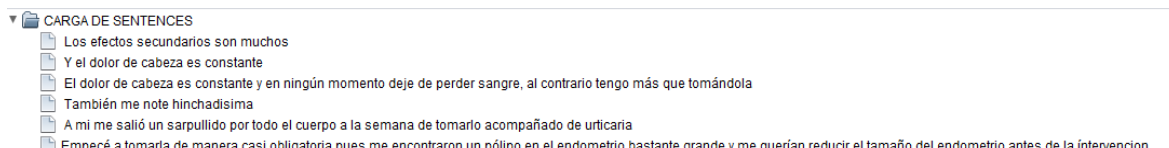


Figura 5.9 Procesamiento de sentences.

5.3.2 Targets

Los TARGETS o ASPECT TERM son las palabras o conjunto de palabras del texto las cuales utilizamos para analizar y las tomamos como palabra clave para posteriormente etiquetarlas con un ENTITY o ASPECT CATEGORY, y su correspondiente polaridad e intensidad, por ejemplo, en el texto utilizamos 'dolor de cabeza', 'urticaria' o 'sarpullido'.

Es importante seleccionar los TARGETS sobre el texto que queremos analizar. En la imagen podemos ver cómo se ha seleccionado algunos.

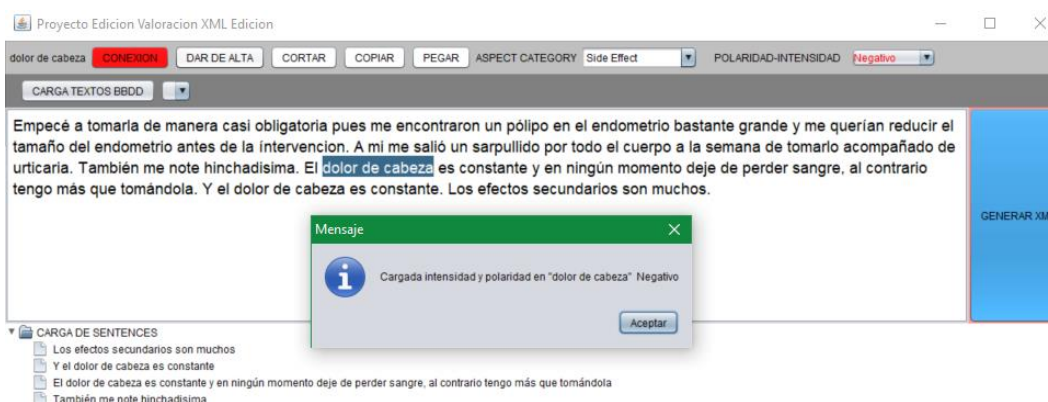


Figura 5.10 Carga de un target.

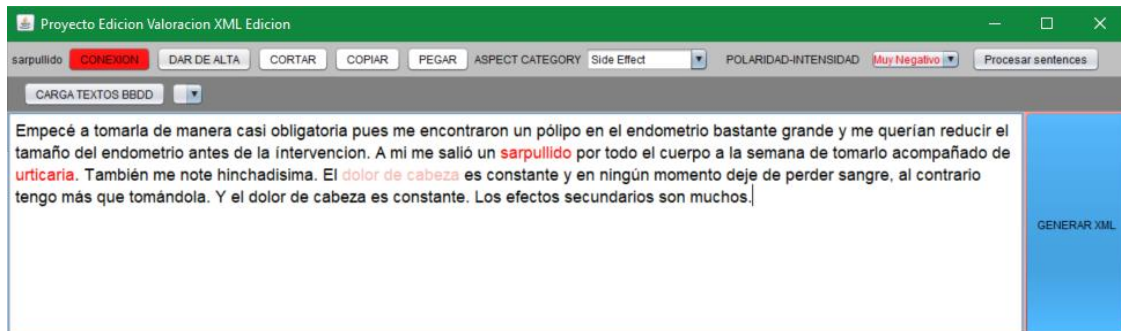


Figura 5.11 Ejemplo de selección de varios Targets.

5.3.3 Entitys/Aspect category.

Los ASPECT CATEGORY, la información que se quiere sacar del texto, son palabras o conjuntos de palabras definidos por el usuario los cuales se utilizan para etiquetar u obtener información sobre un target, en definitiva, sirven para identificar un target elegido del texto. Como podemos ver en la imagen inferior es un desplegable y se pueden crear tantos como sean necesarios.

Por ejemplo, en el ámbito de los medicamentos, estos entitys podrían ser 'eficacia', 'efectos secundarios', 'facilidad de ingerir', 'opinión del usuario'.

Posteriormente de etiquetar los targets con estos entitys habría que proceder a asignarle un grado de polaridad e intensidad como se explica en el próximo punto.

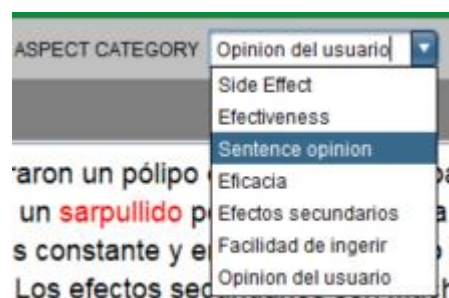


Figura 5.12 Ejemplo de desplegable de entitys de medicamentos.

5.3.4 Polaridad – Intensidad

También es un desplegable como en el caso anterior de los ENTITYS, en el cual se nos indica que polaridad de opinión tiene el target seleccionado y la intensidad que le da el usuario que está etiquetando el texto. Tenemos 5 categorías definidas: muy positivo (color verde oscuro), positivo (verde claro), neutro (amarillo), negativo (rojo claro) y muy negativo (rojo oscuro). De esta manera se podrá distinguir en el texto una vez etiquetado las diferentes polaridades con estos colores.

En la imagen se pueden ver estas 5 categorías y un ejemplo de cómo quedaría en el texto.

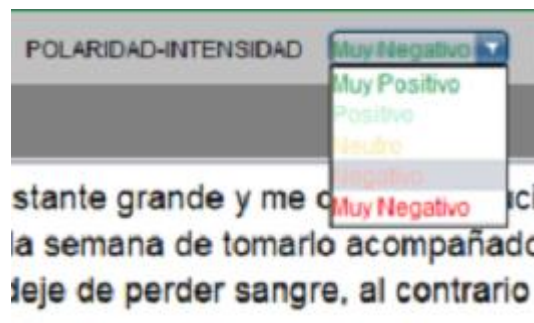


Figura 5.13 Ejemplo de desplegable de polaridad-intensidad

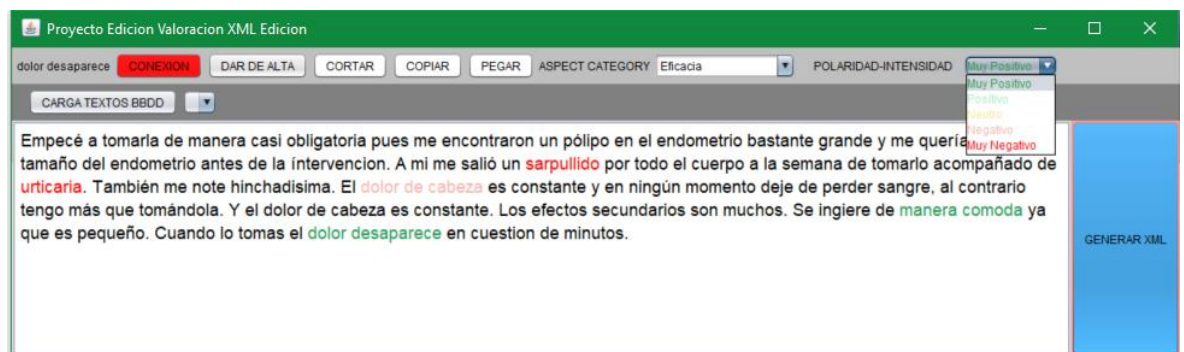


Figura 5.14 Ejemplo de texto etiquetado y colores de polaridad-intensidad.

5.4 Bases de Datos.

En este programa hemos utilizado el programa Workbench para montar las BBDD en formato MySQL. Este programa y formato se explicará en el apartado de herramientas de desarrollo.

En nuestro programa hay creadas 2 estructuras de BBDD, una para la autenticación del usuario para acceder al programa, y la otra que existe es para poder cargar los textos a través de una BBDD.

La primera de estas consiste en una tabla muy sencilla llamada usuario que contiene una idUsuario (para identificar y enumerar los distintos usuarios), el nombre que queremos crear para que el usuario pueda autenticarse y una contraseña asociada a este nombre, como se puede ver es una tabla muy sencilla como la que se muestra a continuación.

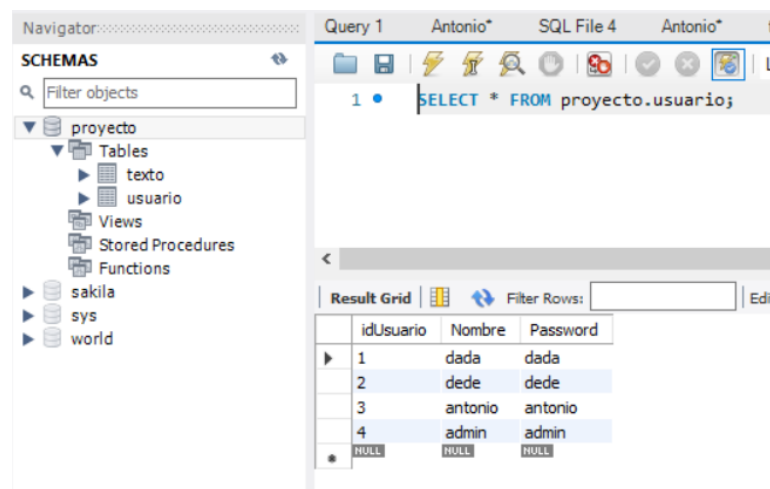


Figura 5.15 Tabla usuario.

La segunda tabla que sirve para cargar los textos desde la BBDD, con el botón de 'CARGAR TEXTOS BBDD' de la ventana principal consiste también en una tabla llamada 'texto' muy simple que contiene los siguientes campos: idTexto (como en el caso de la tabla usuario para numerar), texto (contiene el cómputo de palabras), textoLong (un número entero con la longitud del texto), título, y después tenemos campos que le pueden ser útil al usuario como son la fecha de creación, la fecha de la última modificación, o la primera palabra del texto. Se adjunta una imagen de la tabla texto a continuación.

idTexto	Texto	Textolong	Titulo	FechaCreacion	FechaUltimomod	TextoIni
0	Excelente medicamento, muy eficaz para solventar crisis de pánico y de angustia de manera rápida. Sus efectos se per...	63	Alprazolam	NULL	NULL	Excelente
1	Mi odontólogo me la recetó luego de extraerme dos muelas del juicio, el período de toma era de 7 días, al cabo del 3er ...	25	Amoxicilina	NULL	NULL	Mi
2	Empecé a tomar sertralina por ansiedad, a los 2 días de tomarla empecé a sentir problemas estomacales, diarrea, náus...	36	Sertralina	NULL	NULL	Empecé
3	Tomo Citalopram desde ayer por la noche. La primera vez, a las 3 horas sentía mucho mareo, veía todo más lento, me ...	80	Citalopram	NULL	NULL	Tomo

Figura 5.16 Tabla texto.

5.5 Exportación a XML.

La salida de nuestro programa es transformar el texto analizado por el usuario a formato XML. Este formato XML presenta de forma más estructurada y ordenada los datos para que posteriormente el usuario los utilice para lo que necesite.

La estructura de nuestro XML sería la siguiente: voy a introducir un ejemplo siguiendo el tema de los medicamentos, en este ejemplo se pueden observar las diferentes sentences con sus respectivos 'id', se pueden ver en negrita y subrayados los targets que hemos seleccionado, y para el ejemplo he marcado los entities en color rojo.

También se pueden encontrar con la polaridad 'polarity' que lo hemos marcado de color azul para distinguirlo en el ejemplo. Otro aspecto importante que se puede apreciar es que está señalada la posición inicial y la posición final de los targets señalados dentro del texto.

Se ha adjuntado la imagen en el programa de donde se ha sacado esta estructura XML a continuación.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><text><sentence id="0"
text="Empecé a tomarla de manera casi obligatoria pues me encontraron un pólipo en el
endometrio bastante grande y me querían reducir el tamaño del endometrio antes de la
intervención"/><sentence id="1" text=" A mi me salió un sarpullido por todo el cuerpo a la
semana de tomarlo acompañado de urticaria"><opinion entity="Side Effect" from="206"
polarity="Negativo" target="sarpullido" to="216"/><opinion entity="Side Effect"
from="272" polarity="Muy Negativo" target="urticaria" to="281"/></sentence><sentence
id="2" text=" También me note hinchadísima"><opinion entity="Side Effect" from="302"
```

polarity="Muy Negativo" target="hinchadísima" to="314"/></sentence><sentence id="3" text=" El dolor de cabeza es constante y en ningún momento deje de perder sangre, al contrario tengo más que tomándola"><opinion entity="Side Effect" from="322" polarity="Negativo" target="dolor de cabeza" to="337"/></sentence><sentence id="4" text=" Los efectos secundarios son muchos"><opinion entity="Sentence opinion" from="452" polarity="Muy Negativo" target="Los efectos secundarios son muchos" to="486"/></sentence></text>

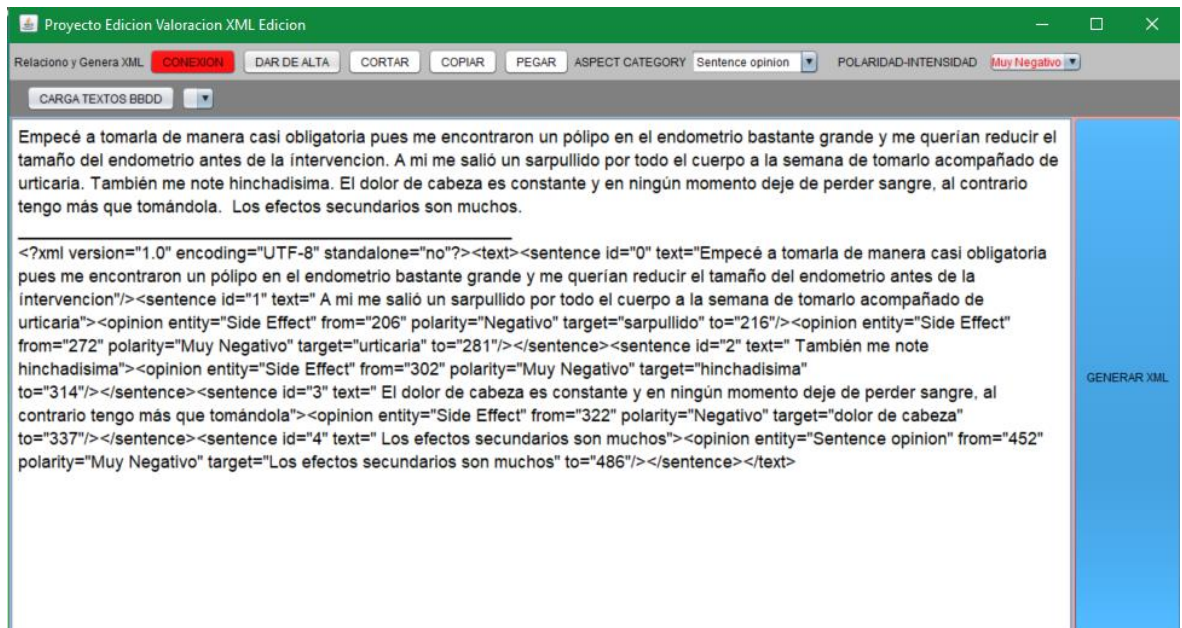


Figura 5.17 Estructura de salida XML

5.6 Pruebas de funcionamiento del software

Para el funcionamiento del programa se ha utilizado un texto de ejemplo de una opinión sobre un medicamento sacado del web '<https://www.mimedicamento.es/>', en nuestro caso hemos cogido un texto de prueba sobre el tratamiento de pólipos endometriales. Las pruebas han consistido en etiquetar el texto con diferentes targets elegidos por el usuario, y crear entitis según se necesite para el etiquetado de los targets. Se ha comprobado que funcionan los despleables bien y sin problema, y que los colores del desplegable de polaridad-intensidad, se actualizan en el texto analizado al momento de poner la etiqueta.

Otro punto que he testeado bastante ha sido la carga a través de la BBDD de textos, para ello han sido definidos varios acerca de los medicamentos con sus títulos, ID y los demás campos, y hemos comprobado que los carga correctamente y con un despegable podemos escoger el texto que queremos analizar.

También se han hecho pruebas con el usuario de acceso, definido con anterioridad en la tabla usuario de la BBDD, con su nombre y contraseña. Donde más pruebas se han realizado ha sido para montar el archivo XML, porque la estructura de este ha sido complicada de realizar, sobre todo la posición inicial y final de los targets dentro del texto

Con las BBDD en Workbench, al principio se creó una BBDD sencilla con solo ID, título y texto, y cuando se comprobó que funcionaba bien y el programa cargaba los textos se le añadieron los demás campos.

Por último, quiero destacar que se han realizado las pruebas con colores llamativos para poder diferenciar bien las distintas secciones del programa, y una vez que se testeo correctamente, se cambiaron los colores por unos más neutros y básicos para que el usuario no tenga problemas y quede más estético.

5.7 Problemas encontrados en el desarrollo

Han existido varios problemas durante el desarrollo de la aplicación. El primer problema que se encontró fue definir cómo iba a ser el flujo del programa (Anexo 7.4), una vez que se definió, nos encontramos con el problema de cómo iba a ser la componente visual del programa y la ubicación de los botones y el cuadro de texto, para ello se siguió el modelo MVC en JAVA, por lado se diseña la representación de la información y por otro lado donde interactúa el usuario.

Un gran problema que tuve y que me costó bastante tiempo arreglar fue que en mi PC tenía instalado el XAMPP que es básicamente un paquete que contiene un gestor MySQL, un servidor web apache, PHP, etc. El problema fue que no me funcionaba bien el gestor de MySQL con XAMPP y decidí utilizar el entorno Workbench, el error que tenía con este es que no podría conectarme al principio a la BBDD, hasta que conseguí ver que el gestor MySQL y el Workbench utilizaban el mismo puerto para la conexión 3306, entonces cuando vi esto solucione el problema desinstalando el XAMPP.

Otro problema que surgió fue elegir la release correcta de MariaDB ya que hay varias que funcionan como server y no iban correctamente. Actualmente en mi programa uso la librería de MariaDB: mariadb-java-client-2.6.0.

Cuando estaba haciendo pruebas del funcionamiento del programa me di cuenta que había contemplado que solo se pudiera analizar un target de una palabra por frase, esto después de darme cuenta lo cambie a nivel de código y ahora el software puede analizar tantos targets como quiera el usuario por frase, y se pueden incluir targets que contengan varias palabras.

Un problema que me surgió en mitad del desarrollo fue diseñar la estructura de la salida XML, ya que no tenía clara la estructura a seguir de los datos, al final se optó por seguir la estructura que se explica en el apartado 5.5.

5.8 Líneas de Futuro

Esta es la primera versión de esta herramienta, pero a continuación voy a indicar mejoras que se pueden hacer para las siguientes versiones:

- Una mejora puede ser que el usuario pueda definir los targets, aunque no aparezcan en el texto.
- Otra mejora puede ser añadir seguridad al sistema de autenticación mediante algún algoritmo de cifrado, crear un superadmin que pueda gestionar las contraseñas de los demás usuarios.
- Otra progresión que se puede hacer en el programa, sería añadir más intensidades a la polaridad, por ejemplo, que tenga (alto positivo, medio positivo, bajo positivo, alto neutro, medio neutro, bajo neutro, alto negativo, medio negativo o bajo negativo).

- Se le puede añadir al programa una mejora para la entrada de la herramienta que busque en tu pc y cargue archivos .txt como es el caso de BRAT.
- Si el usuario se equivoca, la herramienta no deja editar y retroceder para cambiar el error, el usuario se tendría que salir del programa y volver a acceder a él. Esto es un aspecto muy importante a mejorar para permitir al usuario corregir o editar sus errores.
- Desarrollar una aplicación móvil para los sistemas operativos más utilizados como son Android o IOS.

6 CONCLUSIONES Y LIMITACIONES

En este apartado se van a exponer las conclusiones que se han alcanzado al finalizar este trabajo fin de grado.

Respecto al punto de vista de los objetivos en el apartado 2 y analizando los resultados obtenidos previamente, se puede considerar que se han alcanzado los objetivos iniciales para el desarrollo de la herramienta.

Se han considerado acertadas las tecnologías y materiales empleados para el desarrollo del programa, sobre todo se destacan el lenguaje de programación JAVA con sus librerías de SWING que han permitido hacer una interfaz gráfica más vistosa y fácil de programar. Cabe también destacar las tablas de BBDD creadas en Workbench que contienen todos los campos necesarios para el procesamiento y etiquetado de corpus de texto.

La herramienta sigue el patrón de MVC separando la parte visual de la parte lógica, así es más sencillo solucionar cualquier problema que se nos plantee en el código, con esto conseguimos también que el código sea modulable y flexible.

Por otra parte, el usuario del programa obtiene una herramienta sencilla e intuitiva de controlar a base de clics en botones y con despegables, además cuando se etiqueta un target en el texto este se pone del color de la polaridad, con lo que visualmente al usuario le permite distinguir lo que ya tiene etiquetado.

Por último, se obtiene de salida un formato XML con los datos etiquetados estructurados, se podría haber optado por utilizar otro lenguaje con JSON, pero con XML los datos se pueden utilizar para una web o para una base de datos.

Por otro lado, durante el desarrollo de la herramienta se han detectado algunos condicionantes que podrían limitar el funcionamiento de nuestro programa.

Estas limitaciones pueden ser:

- Que el usuario no coja ningún target mientras está analizando el texto, esto haría que el programa no funcionara correctamente y no generaría bien la salida XML, ya que solo se pueden analizar los targets que vienen en el texto, esto como en el punto anterior indica sería una línea de futura para mejorar.
- Otra limitación sería que el programa no te saca la estructura XML tabulada, sino que se la muestra en un texto continuo, para ello se podría dejar estructurada dentro de la herramienta para que cuando la muestres ya salga tabulada con los distintos padres e hijos de un XML.
- Otra limitación que hay que mejorar es la seguridad del programa, ya que ahora mismo solo contiene el usuario y la contraseña en la BBDD, pero esta tabla puede ser vulnerable, y habría que cifrar los datos y proteger esta BBDD.
- Una limitación de la herramienta es que no puede leer archivos TXT, solo puede cargar desde la BBDD o directamente copiándolo en la ventana principal.
- Por último, cabe destacar que si el usuario, se equivoca y no le da al botón de conectar o al de procesar sentencias, el programa no avanza y tiene que cerrarlo y volverse a acceder. Otro problema es que no hay ninguna forma de editar si el usuario se equivoca eligiendo el target o etiquetando. Esto es sin duda una línea de futuro a mejorar.

7 ANEXOS

En este apartado se muestran una serie de anexos en los que se desarrolla algunos aspectos que se conformado en otros puntos anteriores.

7.1 Anexo I: Manual de instalación del entorno de trabajo Netbeans

En este anexo se explica con minitutorial de la descarga e instalación del entorno utilizado para desarrollar la herramienta Netbeans.

En primer lugar, tenemos que tener acceso a Internet para descargarnos el NetBeans IDE 8.2, esta sería la pagina principal de esta plataforma. <https://netbeans.org/>



Figura 7.1.1 Anexo I: Página principal de NetBeans

Posteriormente, hacemos un clic en download, en la parte derecha de la pantalla, entonces saldrá una página donde aparecen las diferentes opciones y releases en las que podemos descargar la plataforma. Nos vamos a 'other releases', y pinchamos en 'Find out more' como se muestra en la siguiente imagen.

Apache NetBeans 12 LTS (NB 12.0)

Latest LTS version of the IDE, released on June 4, 2020.

Features

Download

Older releases

Older Apache NetBeans releases and pre-Apache NetBeans releases can still be downloaded, but are no longer supported.

Find out more

Daily builds and building from source

All Apache NetBeans source code is freely available to build yourself, or you can download (unsupported) daily development builds.

Find out more

Figura 7.1.2 Anexo I: Página principal de NetBeans, descargas

Una vez estamos en 'other releases' bajamos al final de la página y nos encontramos con la versión IDE 8.2, que es la que hemos utilizado para desarrollar nuestra herramienta.

Pre-Apache NetBeans versions

- <https://netbeans.org/downloads/old/8.2/>
- <https://netbeans.org/downloads/old/8.1/>
- <https://netbeans.org/downloads/old/8.0/>
- <https://netbeans.org/downloads/old/7.4/>

Figura 7.1.3 Anexo I: Download versión 8.2

Hacemos clic en esta url <https://netbeans.org/downloads/old/8.2/> y nos aparecerá una página con todas las opciones de paquetes de descarga que tiene esta versión, podemos coger el paquete Java SE, Java EE, HTML5/JavaScript, PHP, C/C++ y all (en nuestro caso escogimos la opción de all que incluía todos los paquetes), ocupa 221 MB.

En la imagen inferior se puede ver los diferentes paquetes que oferta para descargar netbeans. Cuando hagamos clic en all se procederá a la descarga del ejecutable 'netbeans-8.2-windows.exe'. Cabe destacar que para otros sistemas operativos la descarga se realizara clicando sobre el sistema operativo que se desee.

NetBeans IDE Download Bundles

Supported technologies *	Java SE	Java EE	HTML5/JavaScript	PHP	C/C++	All
NetBeans Platform SDK	•	•				•
Java SE	•	•				•
Java FX	•	•				•
Java EE		•				•
Java ME						•
HTML5/JavaScript		•	•	•		•
PHP			•	•		•
C/C++					•	•
Groovy						•
Java Card™ 3 Connected						•
Bundled servers						
GlassFish Server Open Source Edition 4.1.1		•				•
Apache Tomcat 8.0.27		•				•

Download	Download	Download x86	Download x86	Download x86	Download
		Download x64	Download x64	Download x64	
Free, 95 MB	Free, 197 MB	Free, 108 - 112 MB	Free, 108 - 112 MB	Free, 107 - 110 MB	Free, 221 MB

Figura 7.1.4 Anexo I: Diferentes paquetes de NetBeans 8.2

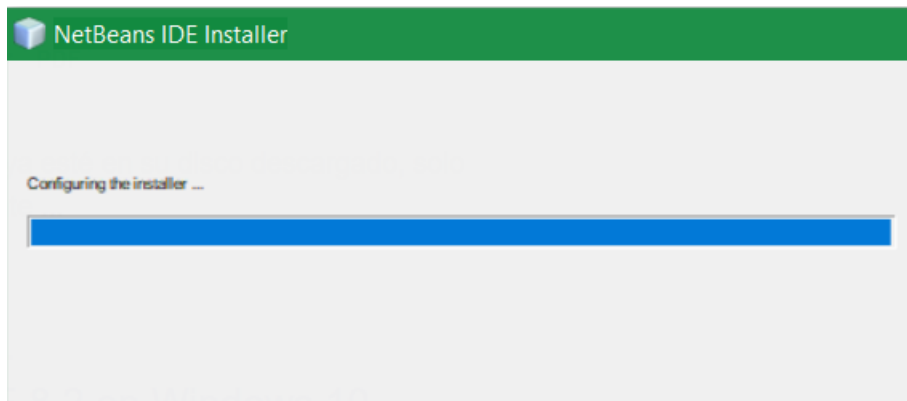


Figura 7.1.5 Anexo I: Barra del instalador.

Luego de esto nos saldrá el instalador de NetBeans 8.2, en nuestro caso hemos hecho una instalación básica y por defecto, por lo que hemos seleccionado en las distintas pestañas 'next'. También se acepta la licencia de producto como viene en las siguientes imágenes.

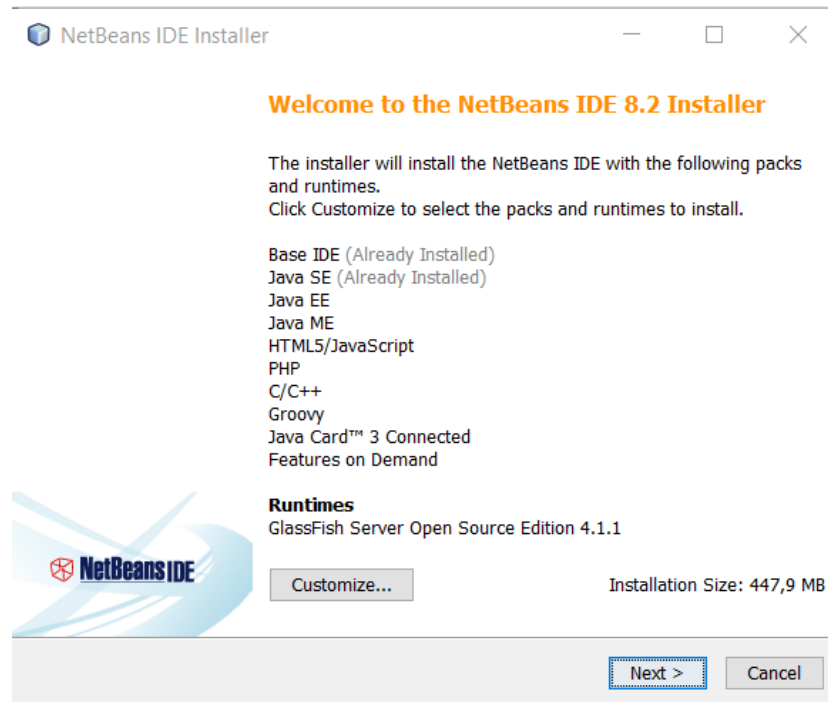


Figura 7.1.6 Anexo I: Instalador de NetBeans 8.2.

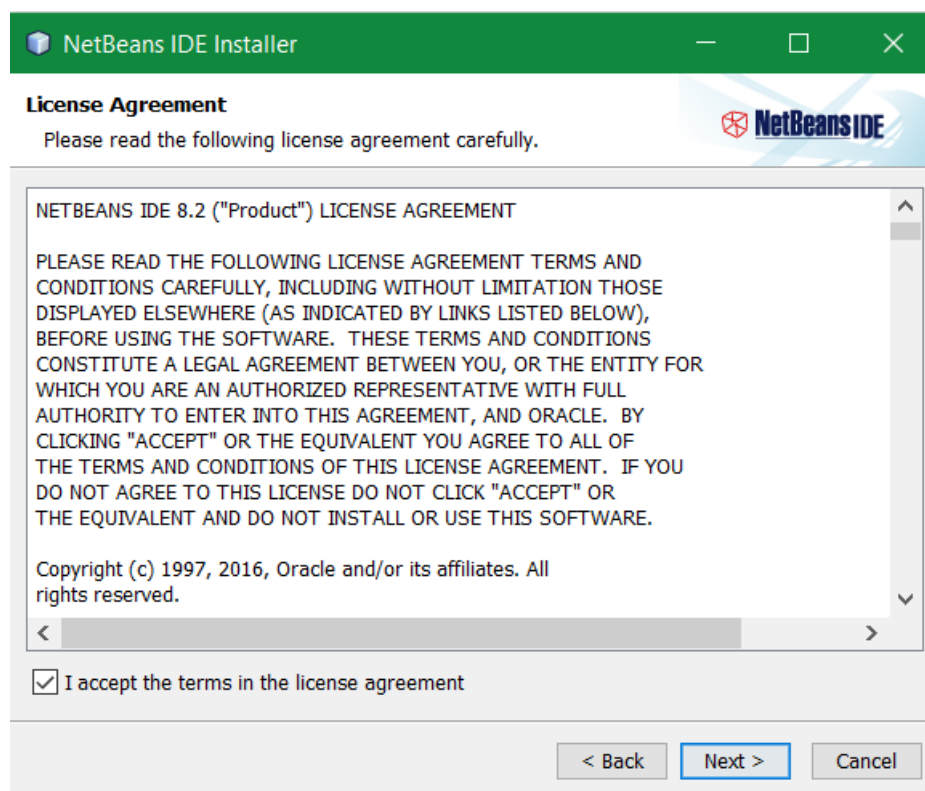


Figura 7.1.7 Anexo I: Licencia de Producto.

Se indica donde se quiere guardar el ejecutable del programa, por defecto lo guarda en C:\Program Files, el JDK 1.8.0 se guarda dentro de la carpeta \Java, este es el kit de desarrollo de Java que provee de las herramientas necesarias para la creación de un programa. Como se muestra en la imagen. En total nos va a ocupar los archivos de instalación 447,9 MB.

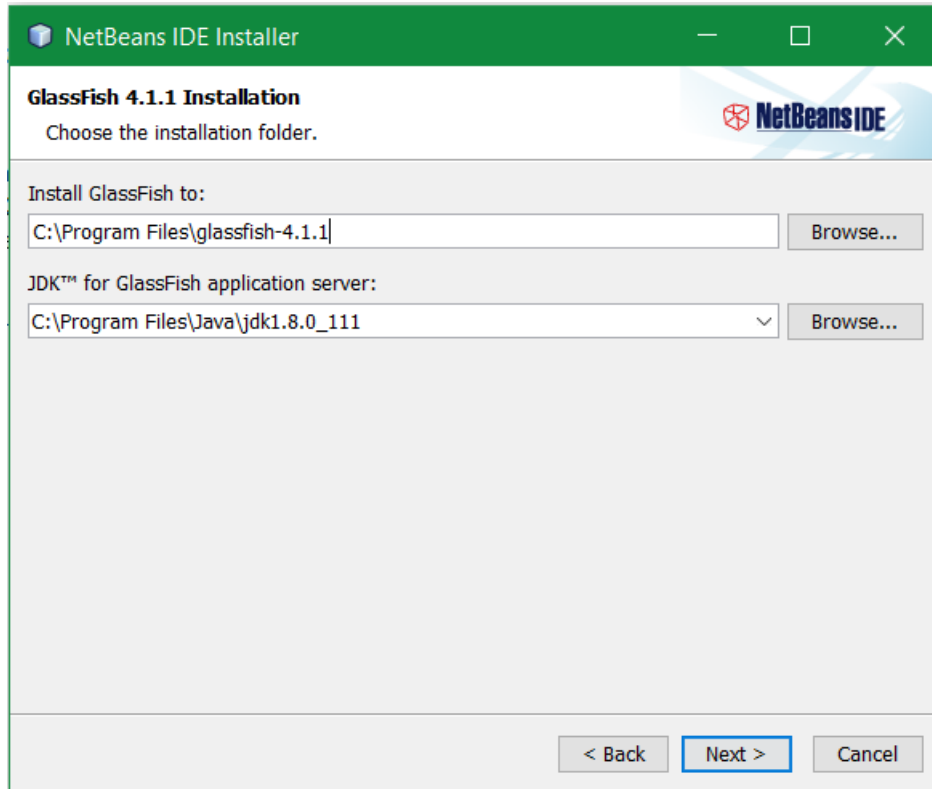


Figura 7.1.8 Anexo I: Instalación de NetBeans

Por último, le damos al botón de 'install', esperamos y cuando termine para finalizar la instalación, pulsamos el botón 'Finish'.

Una vez instalado el entorno de NetBeans, lo iniciamos y para crear nuestro proyecto tenemos que darle a 'File', y posteriormente a 'New Project'.

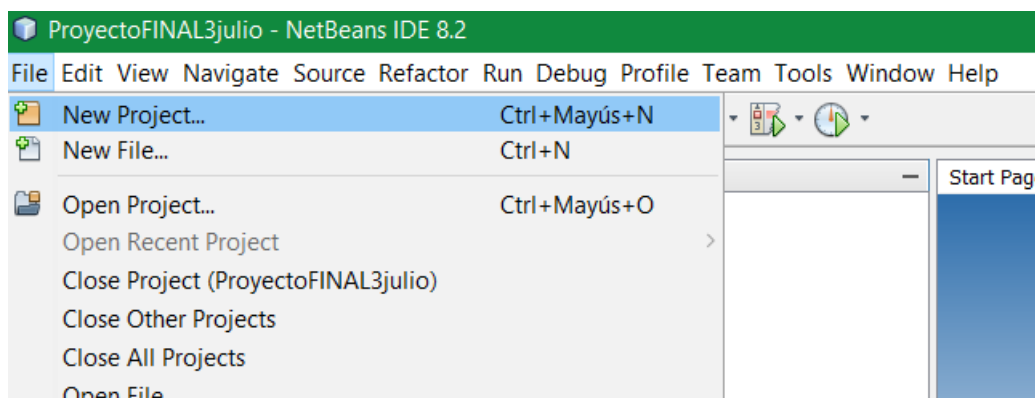


Figura 7.1.9 Anexo I: Crear nuevo Proyecto en NetBeans

Ahora procedemos a ponerle nombre y elegir qué tipo de proyecto queremos crear, en nuestro caso 'Java Aplicación', es muy importante elegir el nombre del proyecto ya que cuando creamos la clase .java nos la crea con el mismo nombre.

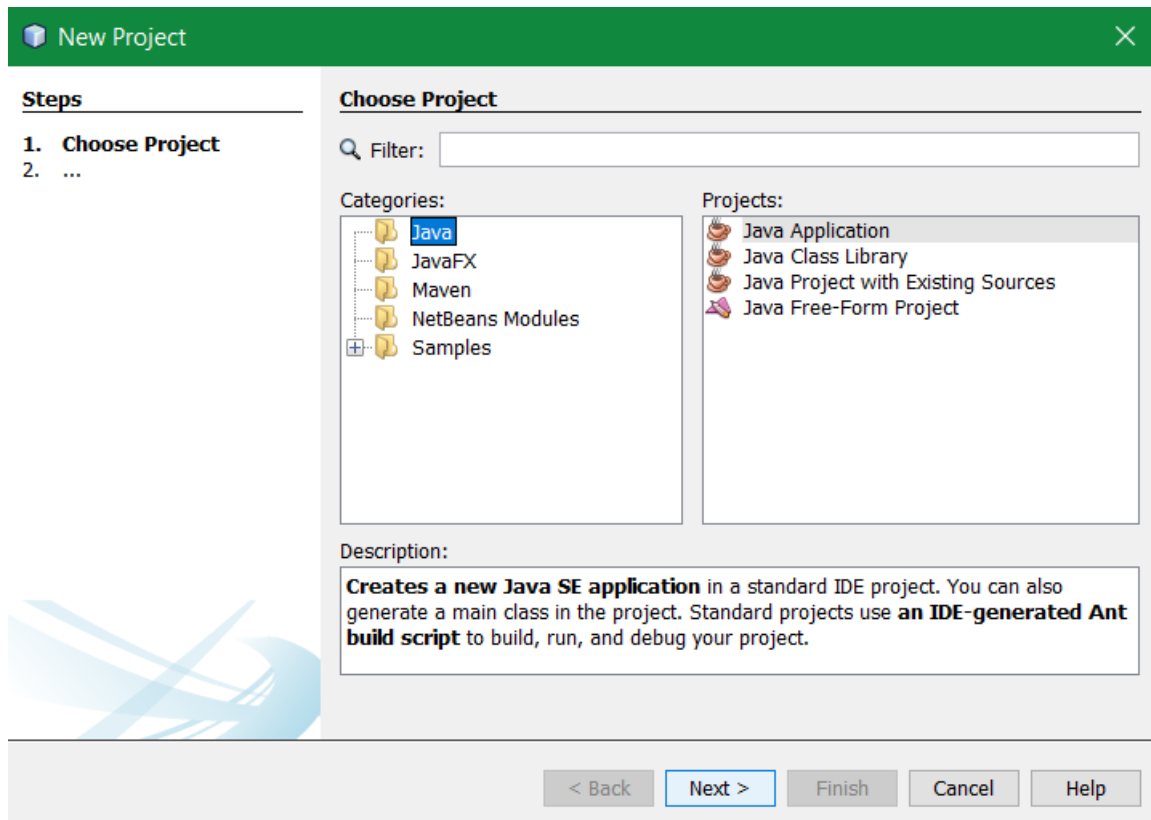


Figura 7.1.10 Anexo I: Creación de nuevo Proyecto

Finalmente, así es como ha quedado nuestro proyecto, es importante resaltar que hay que importar manualmente las librerías de mariadb dentro de 'Libraries'.

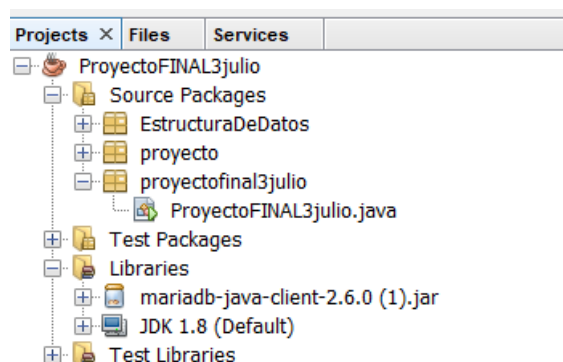


Figura 7.1.11 Anexo I: Estructura del proyecto en NetBeans

Para descargarnos la librería 'mariadb-java-client-2.6.0.jar' que se ha utilizado en nuestra herramienta, se puede descargar desde esta url <https://downloads.mariadb.org/connector-java/2.6.0/> , para importar esta librería dentro de NetBeans, se pincha con el botón derecho del ratón el paquete Libraries, y seleccionamos ' Add JAR/Folder' como se puede ver en la imagen inferior.

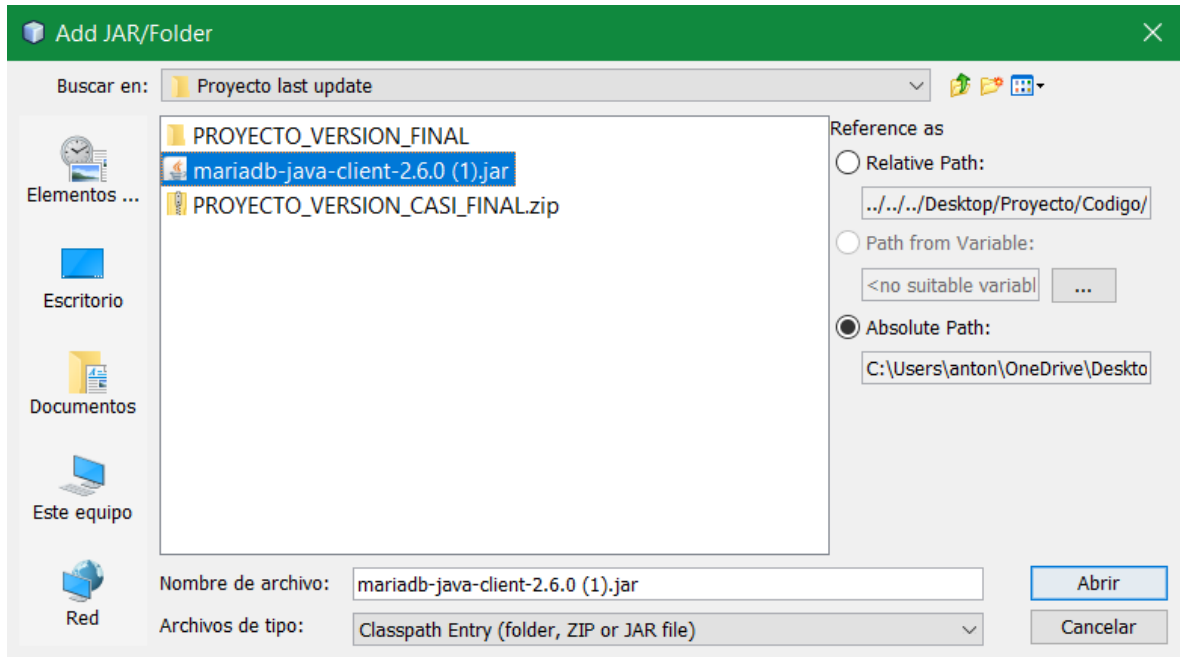


Figura 7.1.12 Anexo I: Importar librería MariaDB

7.2 Anexo II: Manual de instalación para el MySQL Workbench

Las versiones de Windows admitidas para el MySQL Workbench son Microsoft Windows server 2012 y 2016, y Microsoft Windows 8 y 10.

Los requisitos mínimos o recomendados para poder instalar el gestor de BBDD y poder manejarlo sin problemas son:

- CPU con mínimo 2 núcleos y la recomendación son 4 núcleos.
- 2 GB de RAM como mínimo y 8 GB de RAM recomendados en el PC.
- Subsistema de E/S de disco aplicable a una BBDD de escritura intensiva.

Para la descarga correcta del archivo de instalación MySQL usamos el enlace <https://dev.mysql.com/downloads/installer/> y desde aquí nos descargamos la versión que pesa menos 'mysql-installer-web-community-8.0.21.0.msi' como se puede ver en la imagen inferior.

General Availability (GA) Releases Archives ⓘ

MySQL Installer 8.0.21

Select Operating System: [Looking for previous GA versions?](#)

Windows (x86, 32-bit), MSI Installer <small>(mysql-installer-web-community-8.0.21.0.msi)</small>	8.0.21	24.5M	Download
Windows (x86, 32-bit), MSI Installer <small>(mysql-installer-community-8.0.21.0.msi)</small>	8.0.21	427.6M	Download

ⓘ We suggest that you use the [MD5 checksums](#) and [GnuPG signatures](#) to verify the integrity of the packages you download.

Figura 7.2.1 Anexo II: Enlace de descarga para MySQL installer

Hay que hacer clic en el vínculo que pone 'No thanks, just start my download' como se muestra en la imagen. Entonces se te descarga el MySQL installer

MySQL Community Downloads

Login Now or Sign Up for a free account.

An Oracle Web Account provides you with the following advantages:

- Fast access to MySQL software downloads
- Download technical White Papers and Presentations
- Post messages in the MySQL Discussion Forums
- Report and track bugs in the MySQL bug system

Login »
using my Oracle Web account

Sign Up »
for an Oracle Web account

MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can sign up for a free account by clicking the Sign Up link and following the instructions.

No thanks, just start my download.

Figura 7.2.2 Anexo II: Descarga para MySQL installer

Ahora, una vez descargado el archivo indicado anteriormente, se procede con su instalación, se ejecuta el archivo 'mysql-installer-web-community-8.0.21.0.msi', y marcamos la casilla para aceptar los términos de licencias.

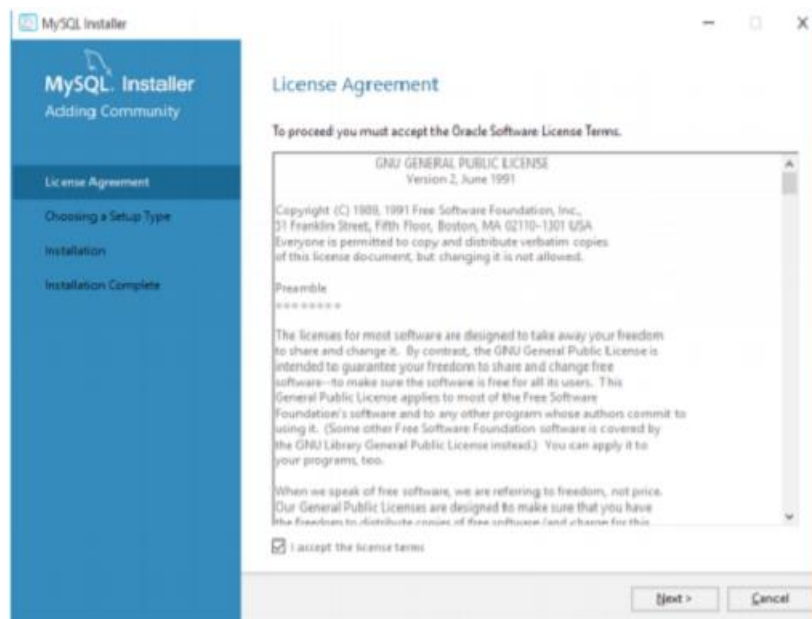


Figura 7.2.3 Anexo II: Aceptación de términos de licencias

Posteriormente seleccionamos la opción de custom para elegir lo que nos interese para nuestro programa, en nuestro caso vamos dentro de MySQL Server, seleccionamos 'MySQL Server8.0.12 -X64' el procesador de 32bits o 64bits lo seleccionamos según el PC. También seleccionamos el MySQL Workbench 8.0.12 – X64.

Como se puede ver en la imagen tenemos ambos seleccionados.

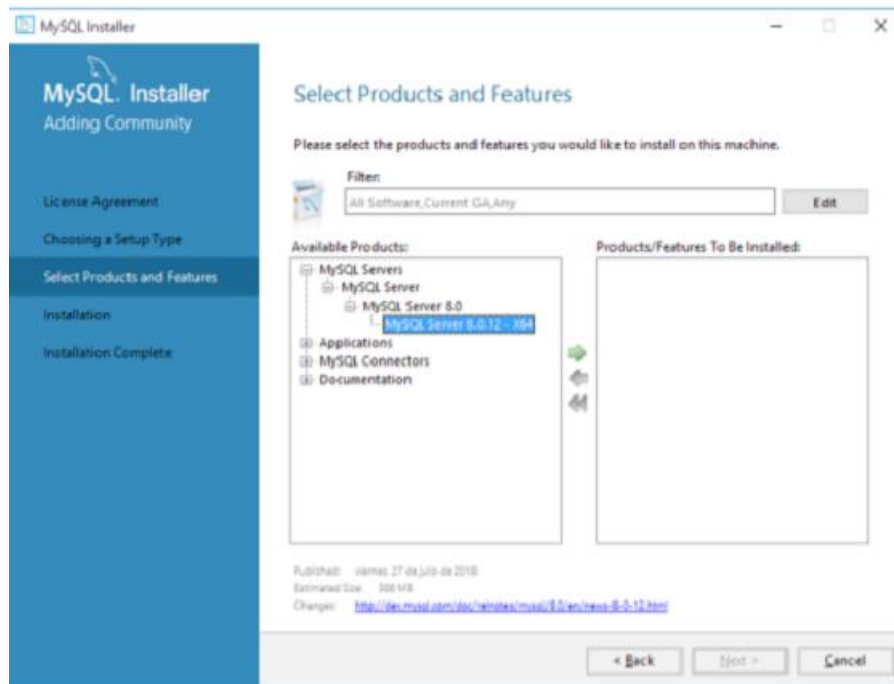


Figura 7.2.4 Anexo II: Elección del producto a instalar

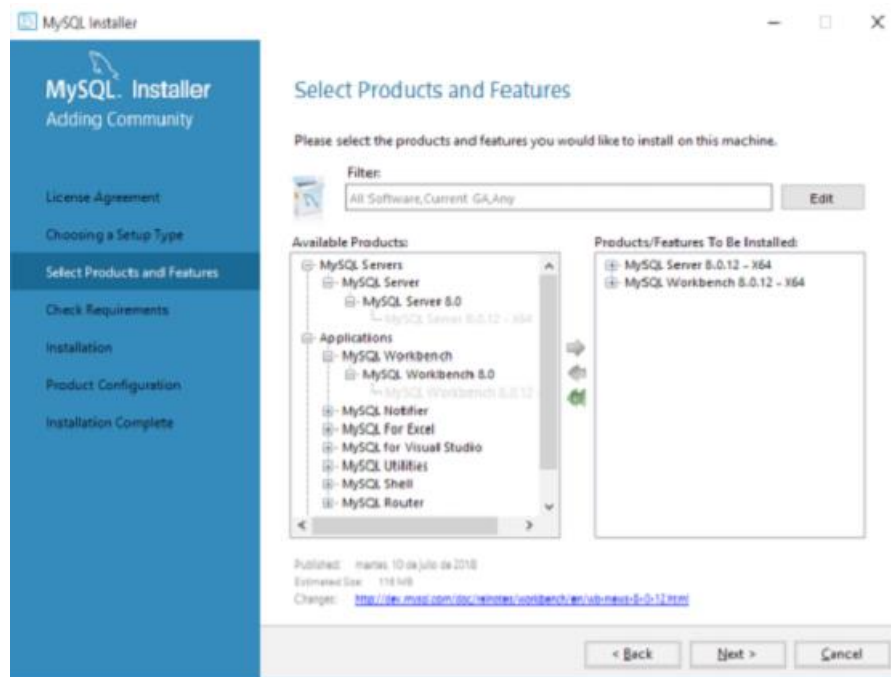


Figura 7.2.5 Anexo II: Selección de Workbench y MySQL Server

El próximo paso es escoger la ruta donde va a instalar el directorio, por defecto esta carpeta se crea en 'C:\Program Files\MySQL\MySQL Server 8.0', aquí también se instala el directorio del Workbench.

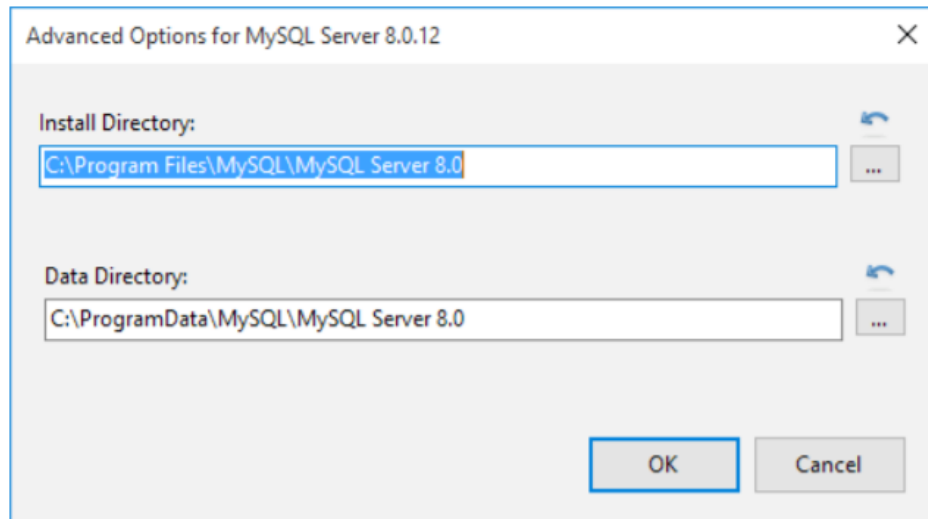


Figura 7.2.6 Anexo II: Selección de carpeta de instalación

Posteriormente te hace un chequeo de los requisitos para que se instale correctamente y normalmente te pide que instale el Microsoft Visual C++, el cual te sale en una ventana emergente. Una vez hecho esto deberían de salirte 2 tics en verde.

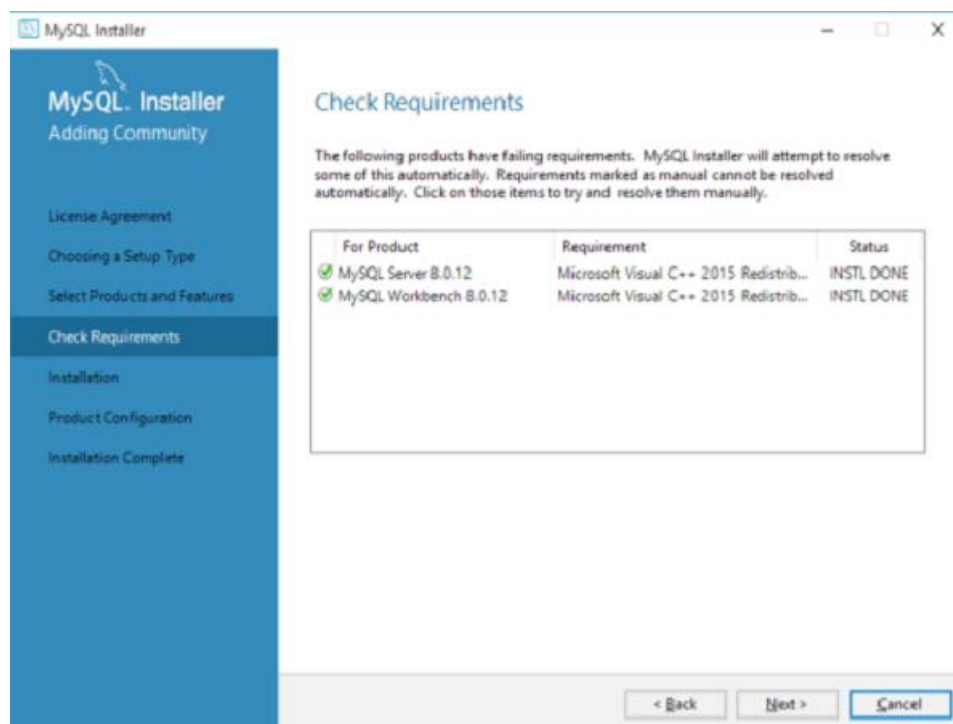


Figura 7.2.7 Anexo II: Requerimientos

Una vez se completa la instalación de ambas cosas, el 'status' saldrá como completado, y habrá que hacer clic en 'next'.

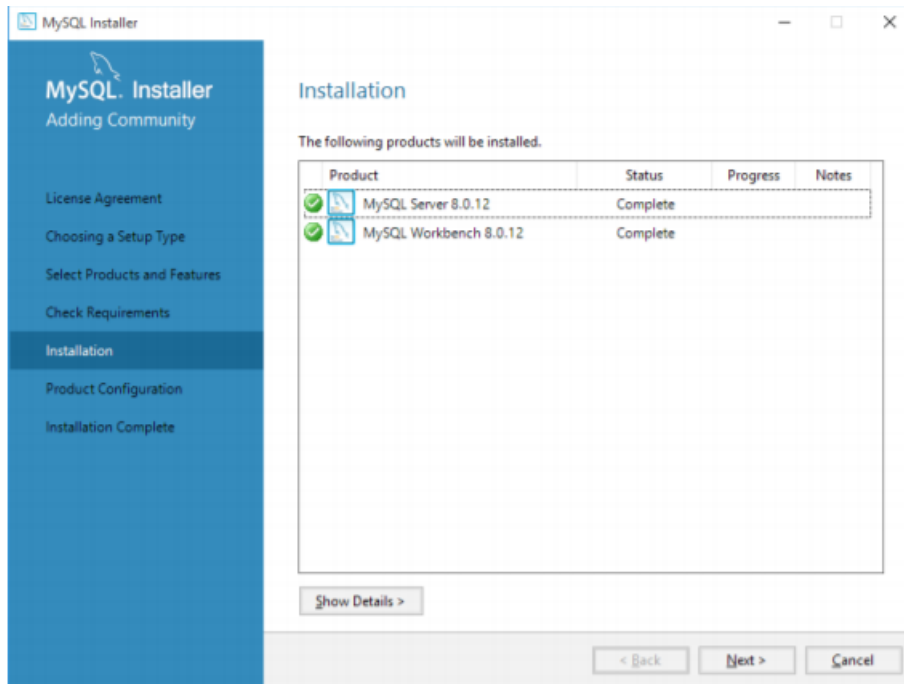


Figura 7.2.8 Anexo II: Instalación completada

Posterior a la instalación se configurará el producto para su utilización, para ello se abrirá la ventana 'Group Replication' en la que habrá que seleccionar la opción 'Standalone MSQL Server/Classic MySQL Replication' y hacer clic en 'next' para continuar.

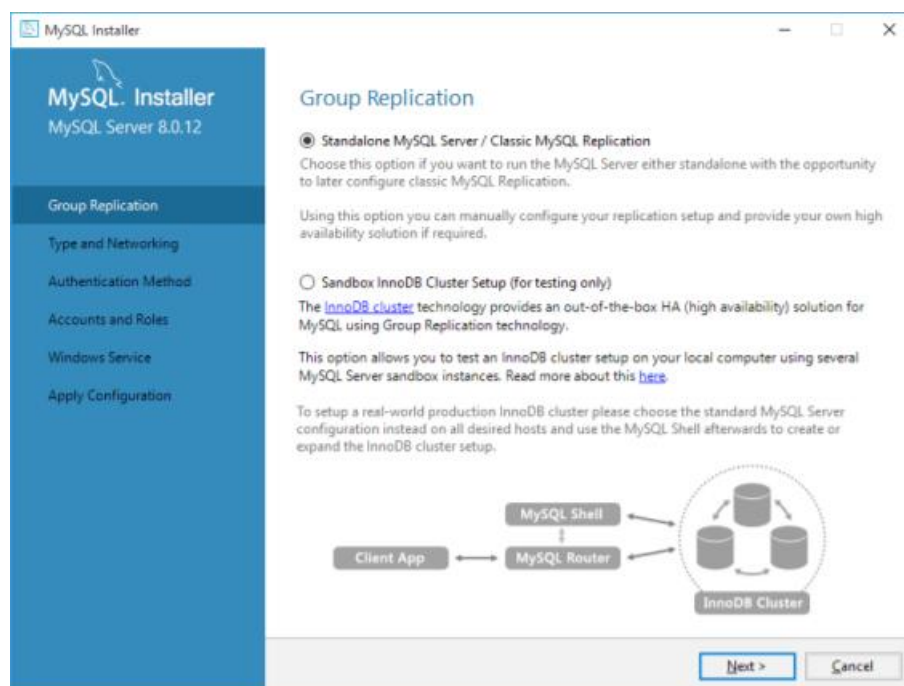


Figura 7.2.9 Anexo II: Selección de 'Group Replication'

En la siguiente ventana 'Type and networking' en el desplegable 'Config Type' se mostrarán 3 opciones a elegir:

- Development Machine: se selecciona esta opción si se quiere que el equipo tenga otras funciones además de funcionar como un servidor de BBDD, tiene un consumo de memoria bajo. En nuestro caso se ha seleccionado esta opción.
- Server Machine: esta opción es para si el equipo realizara otras funciones o servicios, ya que tiene un consumo de memoria medio.
- Dedicated Machine: se instala en una maquina cuya funciona únicamente es la de alojar el servidor de la BBDD, tiene un consumo alto.

Hay que asegurarse que en el apartado 'connectivity' se marca la casilla TCP/IP, el puerto deberá ser el 3306, y hay que marcar la opción 'Open Firewall port for network access'. Un punto importante es asegurarse que en nuestro PC no hay nada conectado ni escuchando en el puerto 3306 que pueda impedir su uso.

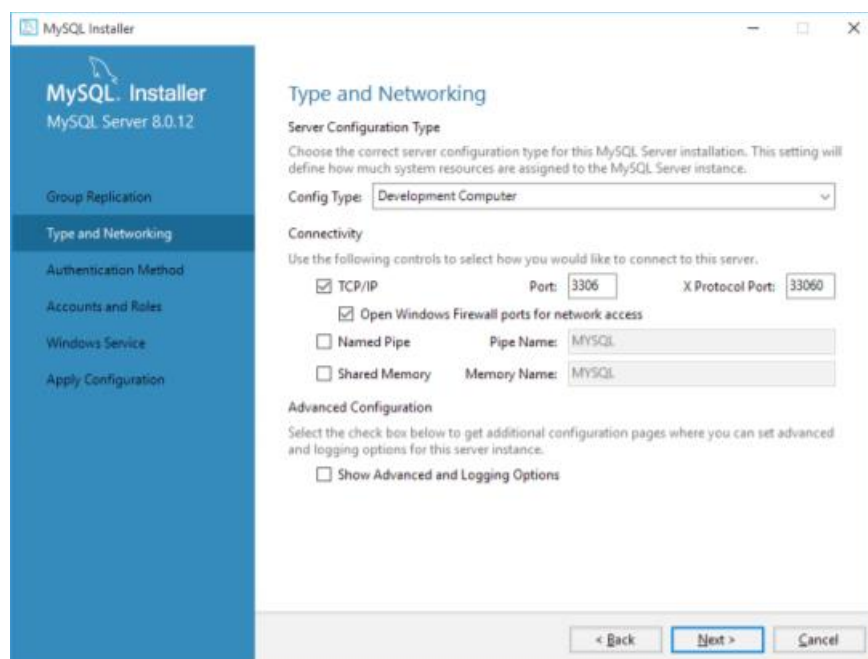


Figura 7.2.10 Anexo II: Apartado 'Type and Networking'

En la siguiente pestaña que aparece marcar la opción 'Use Legacy Authentication Method (Retain MySQL 5.x Compatibility)', y hacemos clic en 'next' para pasar a la ventana para dar de alta a la cuenta de usuario 'Root', esta cuenta será utilizada para la administración y es necesario poner una contraseña segura para esta cuenta.

En la imagen inferior se muestra la ventana para dar de alta y la elección de contraseña para el usuario 'Root'.

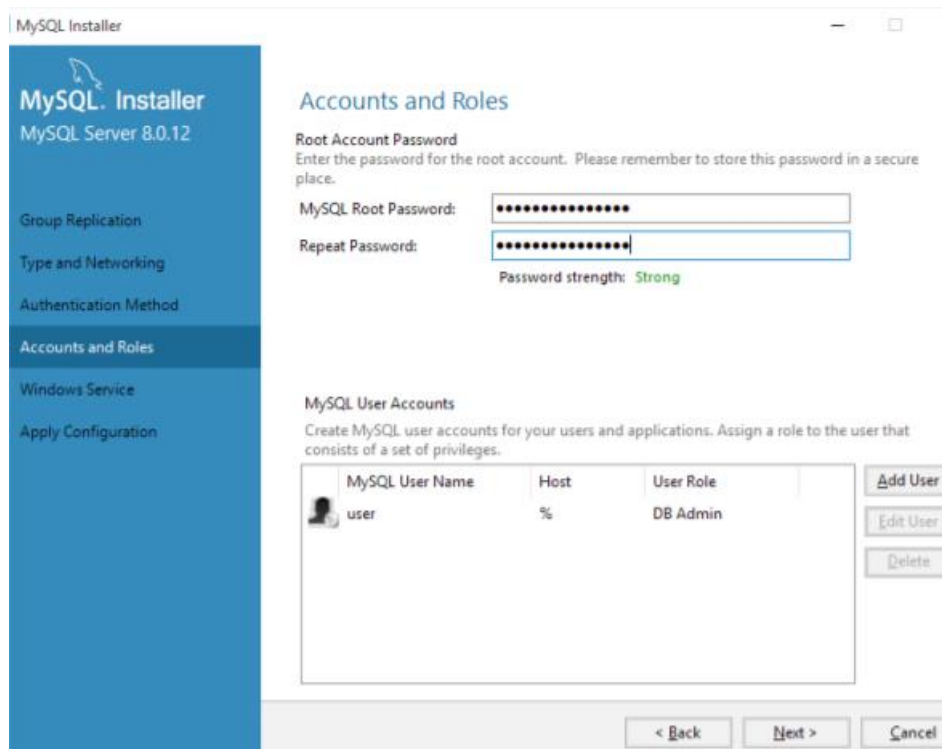


Figura 7.2.11 Anexo II: Apartado para elección de Pass del root user

En el siguiente apartado hay que revisar que las siguientes opciones se encuentren marcadas:

- La casilla 'Configure MySQL Server as a Windows Service'.
- El nombre de Windows server name, MySQLxx.
- Marcar 'Start the MySQL server at System Startup'
- Por último, tiene que estar añadida la opción de 'Standard System Account'

A continuación, en la siguiente ventana se aplicaran todas las opciones seleccionadas y se hará clic en 'Execute', al terminar todas las acciones se pulsará 'Finish' para terminar con la configuración del producto como se muestra en la siguiente imagen.

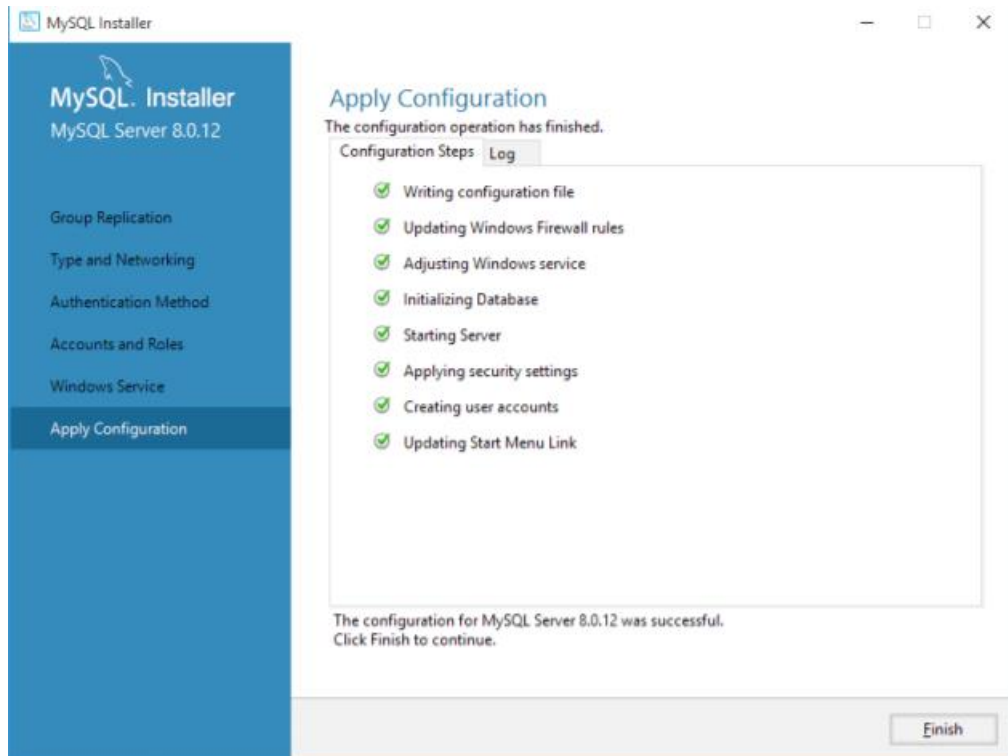


Figura 7.2.12 Anexo II: Finalización de la configuración del producto

Por último se mostrará el 'status' completado, saldrá una ventana llamada 'Installation Complete', marcamos la casilla para inicializar el Workbench después de la instalación y finalizamos el proceso.

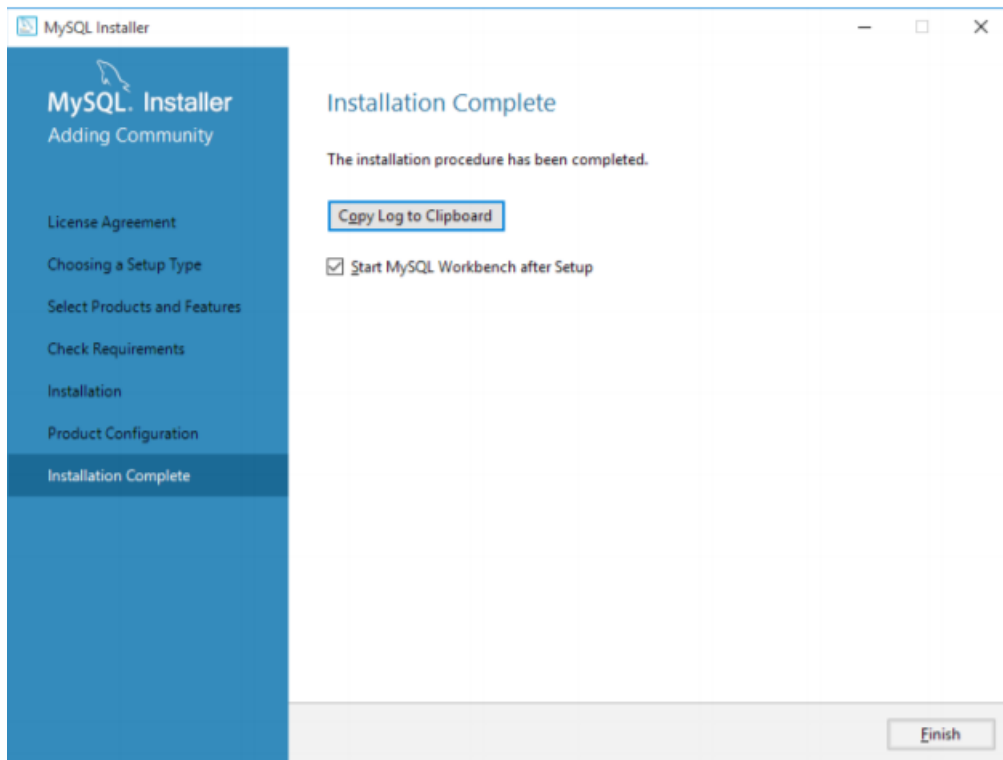


Figura 7.2.13 Anexo II: Finalización de la instalación

Esta sería la pantalla de inicio del MySQL Workbench en la que tenemos la conexión de usuario root que administra el gestor de BBDD.

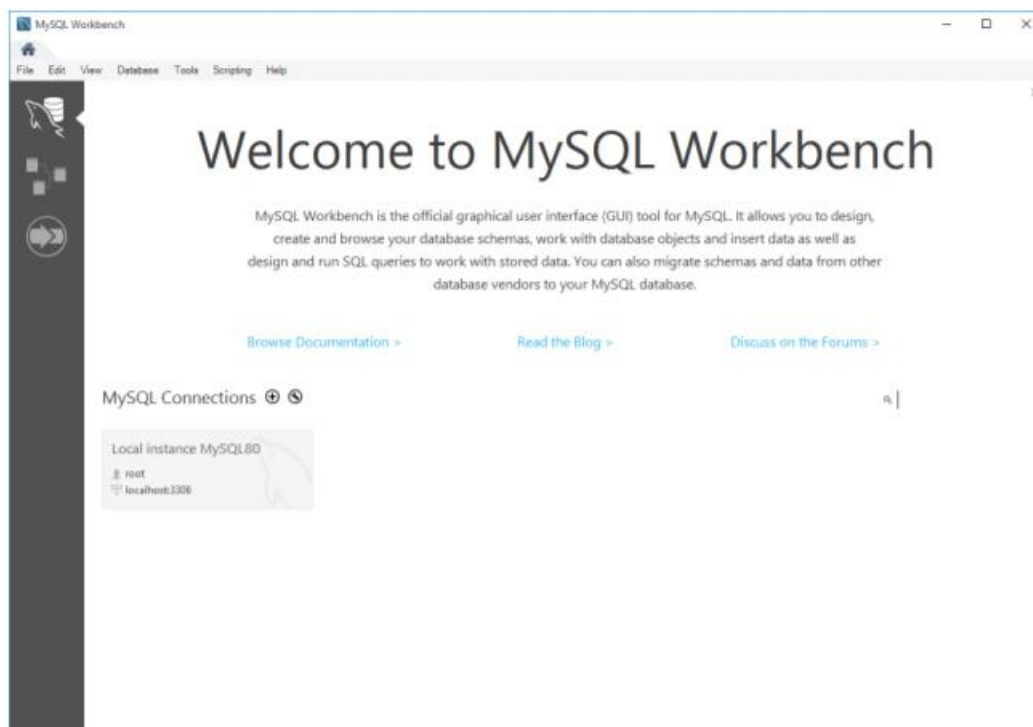


Figura 7.2.14 Anexo II: Pantalla de inicio de MySQL Workbench

7.3 Anexo III: Ejemplo práctico y manual de usuario del programa.

Una vez instalados el NetBeans y el MySQL Workbench, cuyos manuales de instalación se encuentran en el Anexo I y el Anexo II respectivamente, procederemos a la inicialización de la herramienta. Para explicar este manual para el usuario, se utilizará un ejemplo sacado de una opinión sobre un medicamento sacado del web [‘https://www.mimedicamento.es/’](https://www.mimedicamento.es/) es una web que es producto de Insigth Pharma Service BV, cualquier paciente o usuario puede dar su opinión respecto a un medicamento.

Para inicializar el programa se tiene que ejecutar el .jar que hemos entregado, otra manera de iniciar el programa es a través de NetBeans correr directamente dentro del paquete Proyecto el ejecutable EntradaVentana.java. Antes de empezar a ejecutar nada debemos de tener en cuenta la conexión con la BBDD MySQL Workbench, y que las tablas estén bien creadas.

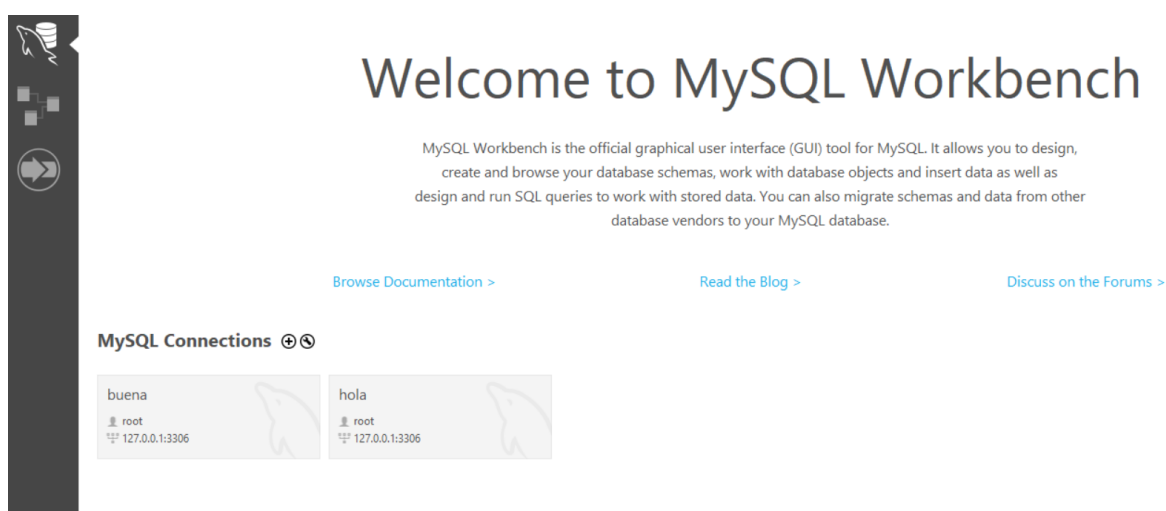


Figura 7.3.1 Anexo III: Conexiones en MySQL Workbench

Una vez tengamos esto en orden podemos proceder a iniciar el programa, la primera pestaña que le va a salir al usuario es la de autenticación, en la que debe de poner el usuario y contraseña correctos y previamente definidos en la tabla usuario de la BBDD, si el usuario se equivoca con las credenciales, el programa volverá a pedir otra vez usuario y contraseña. En la imagen inferior podemos ver un ejemplo de autenticación con el usuario admin y contraseña admin, además también podemos apreciar la tabla usuario de la BBDD, donde se tienen que definir los usuarios.

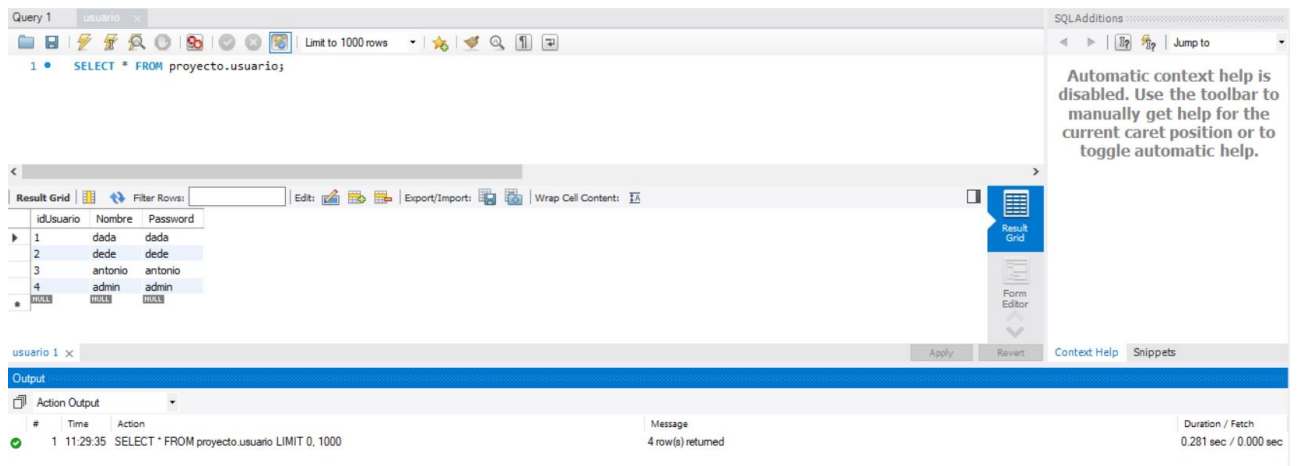


Figura 7.3.2 Anexo III: Tabla Usuario en Workbench

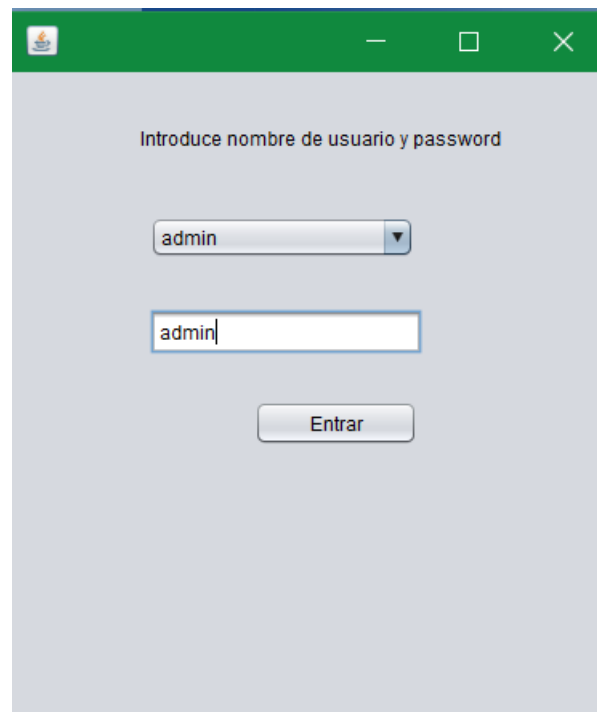


Figura 7.3.3 Anexo III: Ventana de autenticación

Si las credenciales son válidas aparecerá la siguiente ventana y se podrá entrar al programa.

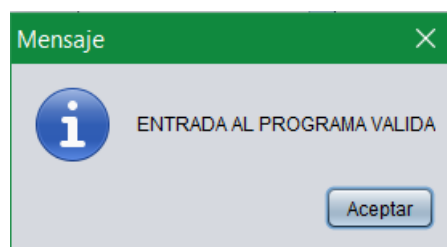


Figura 7.3.4 Anexo III: Credenciales validas

Una vez dentro del programa, lo primero que nos aparece será la ventana principal, esta ventana viene con un texto por defecto ya cargado, en la imagen inferior se puede ver como es el diseño de la ventana principal de la herramienta.

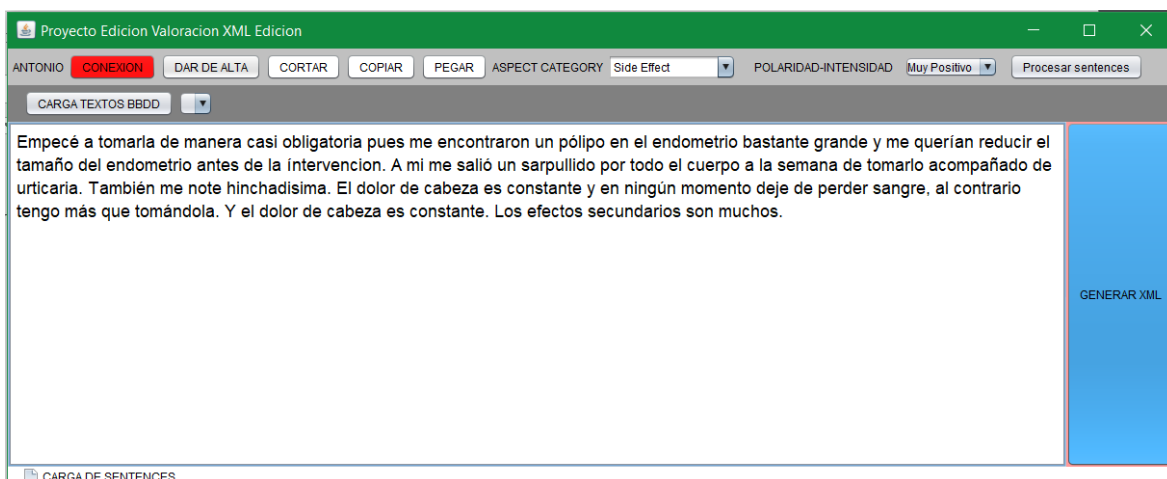


Figura 7.3.5 Anexo III: Ventana principal

Lo primero que tenemos que hacer será darle al botón rojo de 'CONEXIÓN' para poder conectarnos a la BBDD y al programa, una vez hecho esto, la siguiente acción será cargar un texto en la ventana principal, para ello tenemos 2 opciones:

- Cargar el texto copiándolo del cualquier sitio y pegándolo directamente en la ventana, hay también 2 opciones para copiar y pegar, desde el teclado con ctrl+c y ctrl+v, o también lo podemos hacer con los botones que aparecen en la ventana superior de la herramienta: 'CORTAR', 'COPIAR' y 'PEGAR'.
- La otra opción para cargar el texto sería a través de la BBDD, para ello los textos a cargar tienen que estar definidos en la tabla 'texto' del MySQL Workbench, después se selecciona el botón 'CARGAR TEXTOS BBDD' y se abre un desplegable con los distintos títulos de los textos que tenemos almacenados. La tabla 'texto' contiene la id del texto, el texto, el título, la fecha de creación, la fecha de modificación y la longitud del texto.

En la parte inferior se presentan imágenes de como cargar un texto de la BBDD.

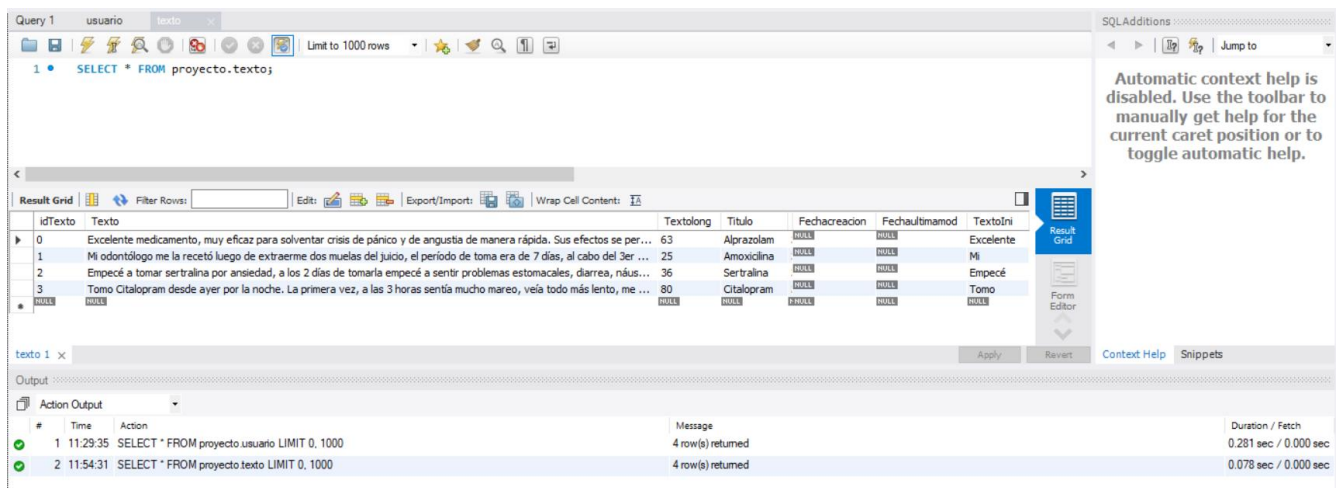


Figura 7.3.6 Anexo III: Tabla texto

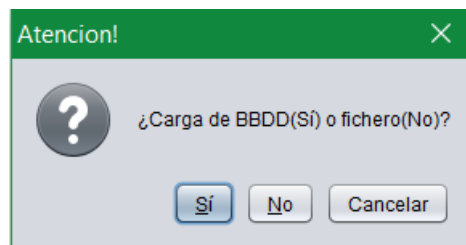


Figura 7.3.7 Anexo III: Ventana para cargar texto por BBDD

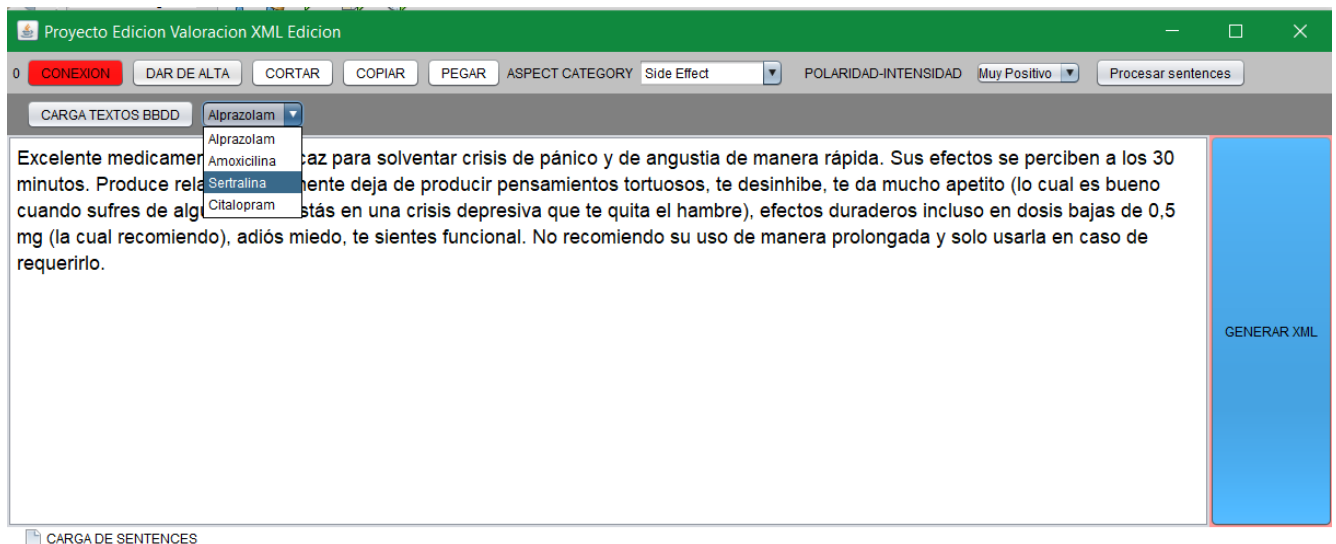


Figura 7.3.8 Anexo III: Despegable para cargar Texto BBDD

Una vez elegido el texto que se desea analizar procedemos a procesar las frases que tiene el texto para que se así sea más fácil de analizar y ya este preparado para la salida de XML, para esto tenemos el botón 'Procesar sentences' arribar a la derecha de la herramienta, cuando pulsamos este botón, las distintas frases se nos cargaran en la parte inferior del programa. En la imagen adjuntada se puede observar esto.

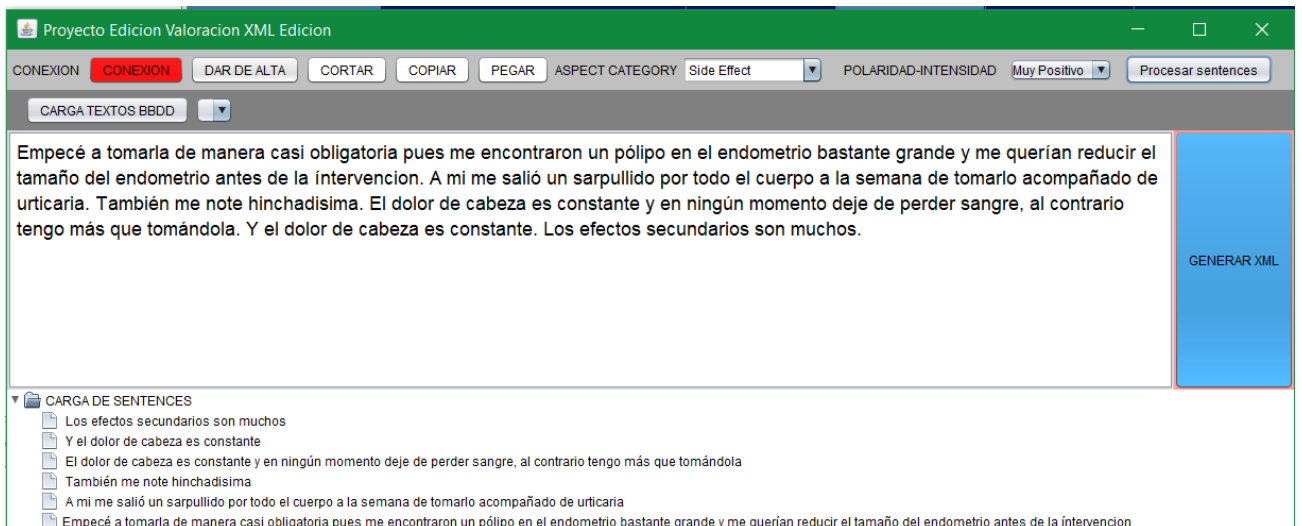


Figura 7.3.9 Anexo III: Procesar sentencias

Gracias a que podemos analizar el texto por frases, el usuario puede diferenciar cuando esta etiquetando las distintas frases.

Para que el usuario empiece a etiquetar es muy simple seleccionamos una palabra o varias palabras con el ratón como se ve en la imagen inferior.

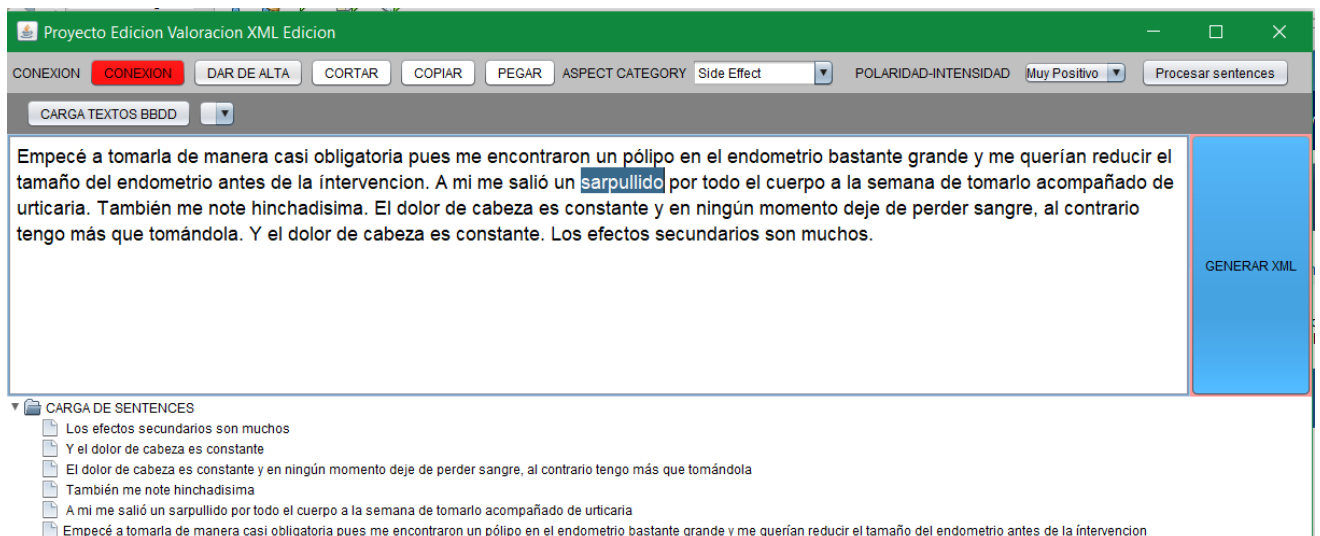


Figura 7.3.10 Anexo III: Seleccionamos un target en el texto

En este caso hemos seleccionado la palabra 'sarpullido', para etiquetar esta palabra es tan sencillo como una vez marcada seleccionar un 'ASPECT CATEGORY' y 'POLARIDAD-INTENSIDAD'. Como podemos ver en la siguiente imagen, en nuestro ejemplo hemos seleccionado como aspecto category 'Efecto secundario' y como polaridad 'muy negativo'.

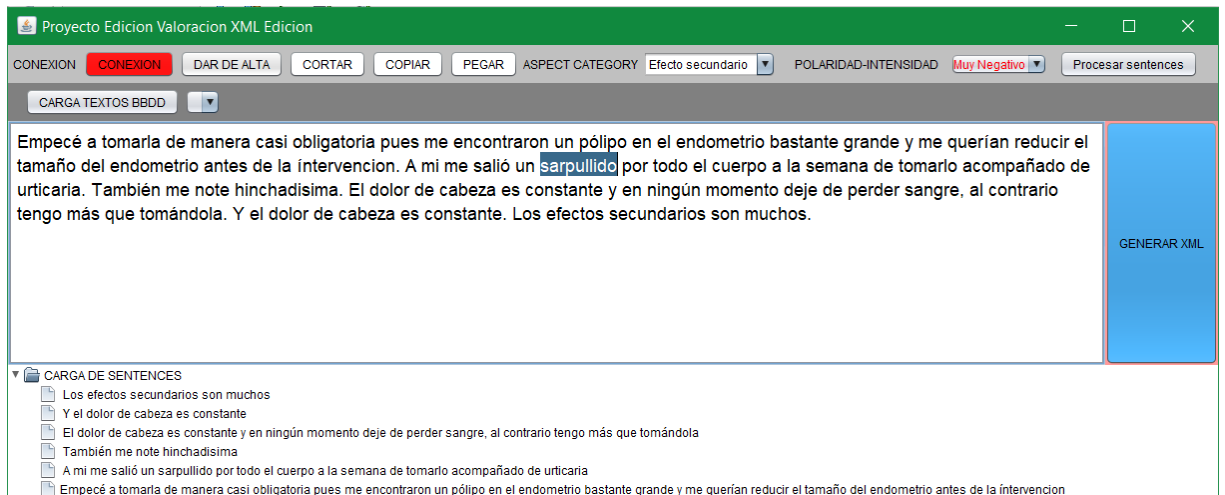


Figura 7.3.11 Anexo III: Etiquetamos el target elegido.

Para finalizar el proceso de etiquetar ese target, le damos al botón 'DAR DE ALTA', entonces ese target se guardará con su respectivo aspect category y su polaridad.

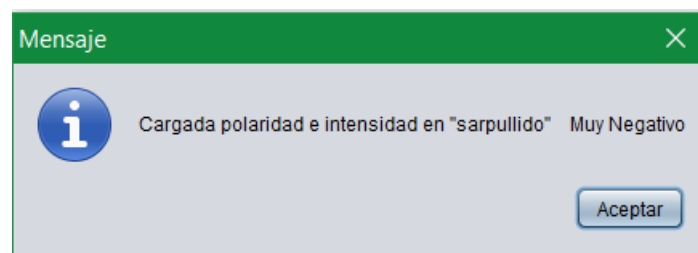


Figura 7.3.12 Anexo III: Ventana de dar de alta

Algunos aspectos importantes a tener en cuenta es que tanto Aspect Category, como Polaridad-intensidad son dos desplegables, en los que el usuario puede elegir el que desee. En el caso de Aspect category el usuario puede crear tantos como necesite para etiquetar el texto. En la imagen aparecen algunos ejemplos que hemos creado.

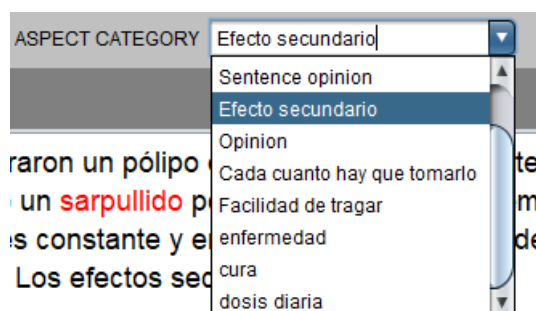


Figura 7.3.13 Anexo III: Aspect Category

En el caso de Polaridad-intensidad, también es un desplegable, pero solo tiene definidos 5 polaridades: muy negativo, negativo, neutro, positivo y muy positivo. Cada una de las polaridades tiene un color dependiendo del grado de polaridad. Por ejemplo, para 'muy negativo' el color es rojo oscuro. Dependiendo cual se escoja, en el texto aparecerá el target de ese color, en nuestro caso el target 'sarpullido' aparece en rojo oscuro porque esta etiquetado como muy negativo.

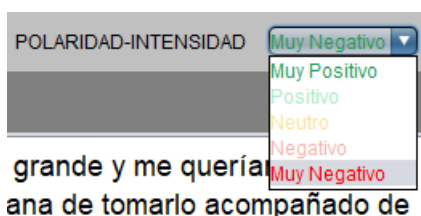


Figura 7.3.14 Anexo III: POLARIDAD-INTENSIDAD

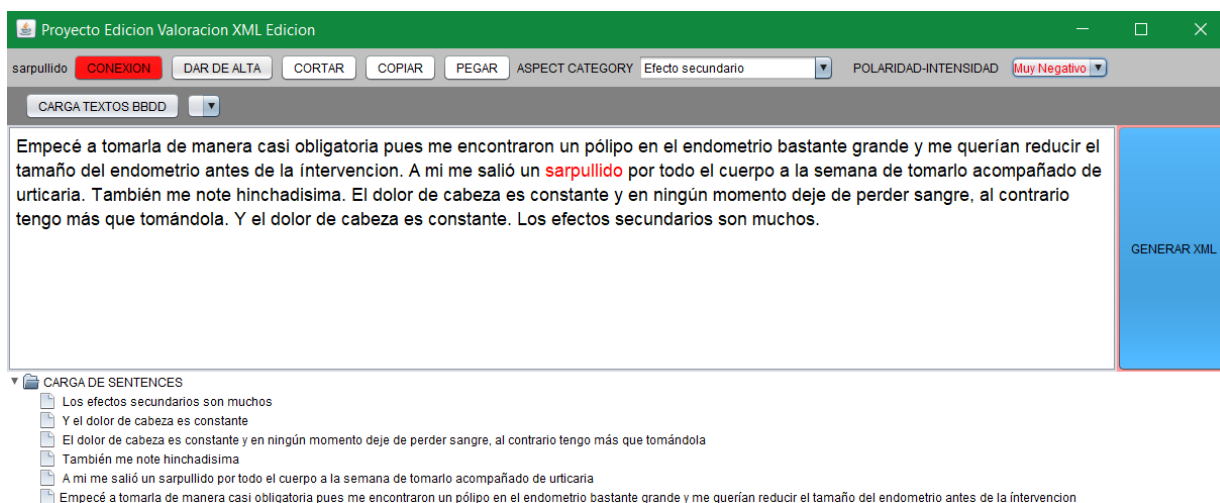


Figura 7.3.15 Anexo III: Palabra de ejemplo etiquetada

Posteriormente si queremos seguir etiquetando targets del texto repetimos este proceso hasta que finalicemos de analizar el documento, una vez que lo tengamos finalizado, pulsaremos el botón azul 'GENERAR XML', por ejemplo, para el target etiquetado anteriormente nos saldrá esta estructura XML.

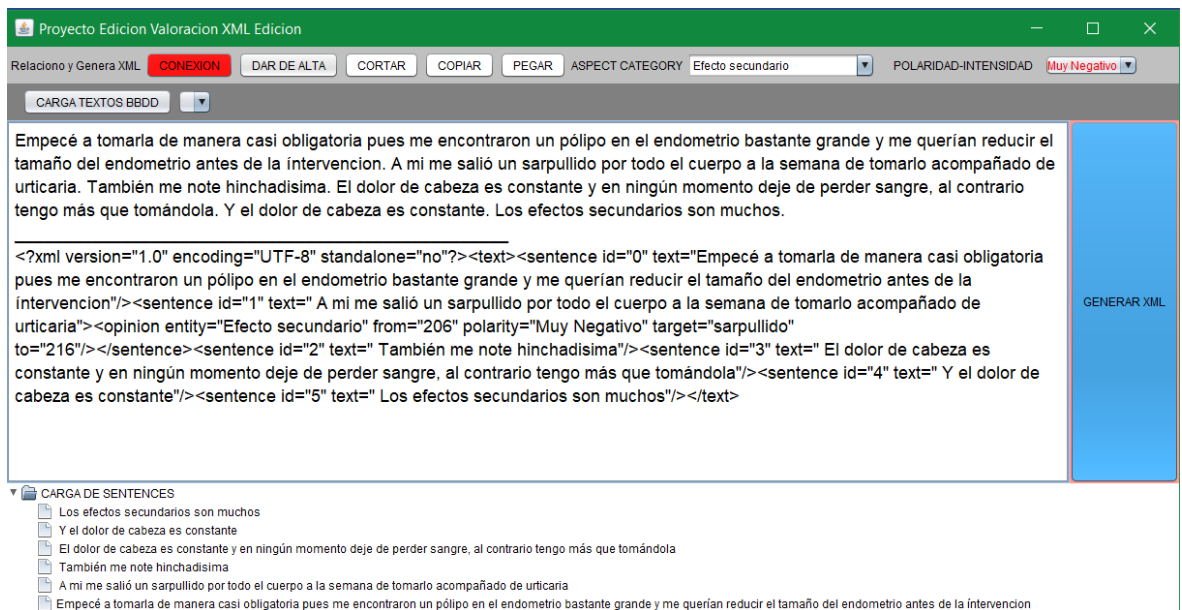


Figura 7.3.16 Anexo III: Ejemplo 1 salida XML

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><text><sentence id="0"
text="Empecé a tomarla de manera casi obligatoria pues me encontraron un pólipo en el
endometrio bastante grande y me querían reducir el tamaño del endometrio antes de la
intervención"/><sentence id="1" text=" A mí me salió un sarpullido por todo el cuerpo a la
semana de tomarlo acompañado de urticaria"><opinion entity="Efecto secundario"
from="206" polarity="Muy Negativo" target="sarpullido" to="216"/></sentence><sentence
id="2" text=" También me note hinchadísima"/><sentence id="3" text=" El dolor de cabeza
es constante y en ningún momento deje de perder sangre, al contrario tengo más que
tomándola"/><sentence id="4" text=" Y el dolor de cabeza es constante"/><sentence id="5"
text=" Los efectos secundarios son muchos"/></text>
```

En el anterior formato XML se puede deducir que tenemos 'id=1' para identificar las frases, 'text=" A mí me salió un sarpullido por todo el cuerpo a la semana de tomarlo acompañado de urticaria" aquí identificamos la frase que se ha analizado, dentro de opinión, tenemos target=" sarpullido" , también tenemos la posición inicial y final de target from="206" y to="216", se encuentra también la parte etiquetada por el usuario como es aspect category o entity= "Efecto secundario" y la polarity= "Muy Negativo"

Ahora se va a poner un ejemplo completo de salida XML con varios targets etiquetados en el texto.

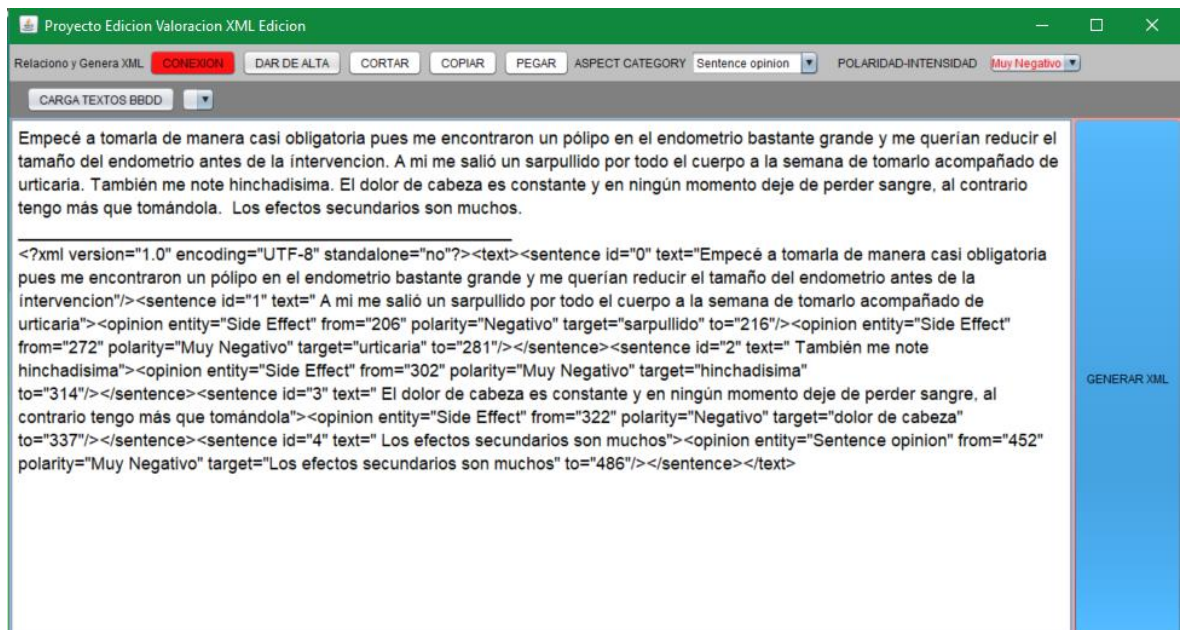
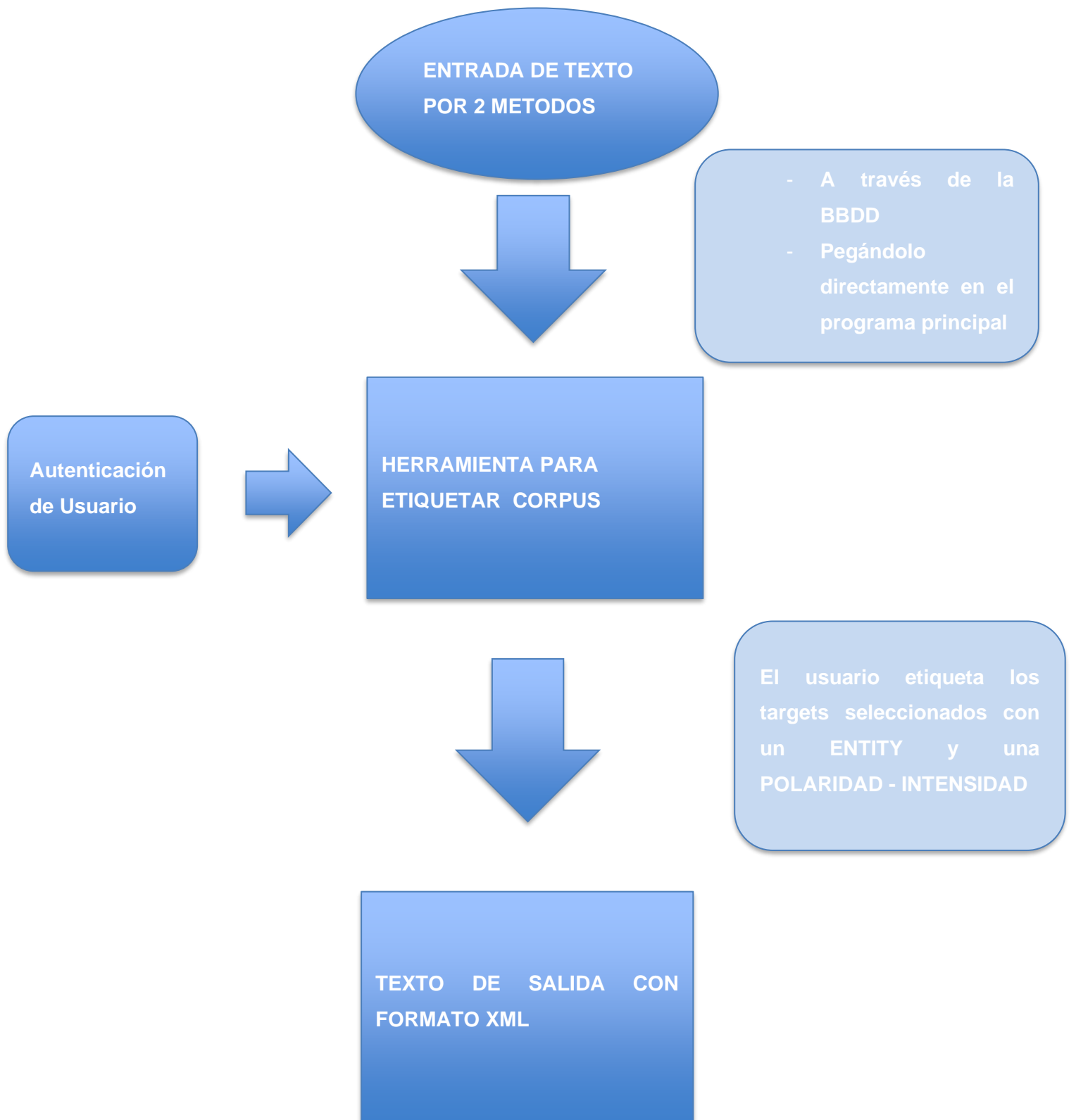


Figura 7.3.17 Anexo III: Ejemplo 2 salida XML

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?><text><sentence id="0"
text="Empecé a tomarla de manera casi obligatoria pues me encontraron un pólipo en el
endometrio bastante grande y me querían reducir el tamaño del endometrio antes de la
intervención"/><sentence id="1" text=" A mi me salió un sarpullido por todo el cuerpo a la
semana de tomarlo acompañado de urticaria"><opinion entity="Side Effect" from="206"
polarity="Negativo" target="sarpullido" to="216"/><opinion entity="Side Effect"
from="272" polarity="Muy Negativo" target="urticaria" to="281"/></sentence><sentence
id="2" text=" También me note hinchadísima"><opinion entity="Side Effect" from="302"
polarity="Muy Negativo" target="hinchadísima" to="314"/></sentence><sentence id="3"
text=" El dolor de cabeza es constante y en ningún momento deje de perder sangre, al
contrario tengo más que tomándola"><opinion entity="Side Effect" from="322"
polarity="Negativo" target="dolor de cabeza" to="337"/></sentence><sentence id="4"
text=" Los efectos secundarios son muchos"><opinion entity="Sentence opinion"
from="452" polarity="Muy Negativo" target="Los efectos secundarios son muchos"
to="486"/></sentence></text>
```

Con esto concluiría el manual de uso de usuario, cabe destacar que, si el usuario se equivoca etiquetando o saltándose algún paso anterior, debe de cerrar la herramienta y volverla a abrir.

7.4 Flujo de datos de Programa.



8 BIBLIOGRAFÍA

[1] Carlos Henríquez, Ferran Pla, Lluís-F. Hurtado, Jaime Guzmán. Análisis de sentimientos a nivel de aspecto usando ontologías y aprendizaje automático. Universidad Nacional de Colombia, disponible en

<http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/5492/3251>

[2] S. Jiménez Zafra. Análisis de sentimientos a nivel de aspecto y estudio de la negación en opiniones escritas en español. Departamento de informática, ESPL, Universidad de Jaén, disponible en

<https://gplsi.dlsi.ua.es/sepln15/sites/gplsi.dlsi.ua.es.sepln15/files/attachments/paper19.pdf>

[3] Análisis del sentimiento. Instituto de ingeniería del conocimiento, disponible en

<https://www.iic.uam.es/soluciones/entorno-digital/lynguo/analisis-del-sentimiento/>

[4] David M. Blei, Andrew Y. Ng, Michael I. Jordan. Latent Dirichlet Allocation (LDA), disponible en <http://www.jmlr.org/papers/v3/blei03a>

[5] Isidro Peñalver, Rafael Valencia, Francisco García. Minería de opiniones basada en características guiada por ontologías. Sociedad Española para el Procesamiento del Lenguaje Natural, disponible en <https://core.ac.uk/reader/16369218>

[6] Big Data, Bigger Digital Shadows and Biggest Growth in the Far East, sponsored by EMC and published by IDC, disponible en <https://www.emc.com/leadership/digital-universe/2012iview/index.htm>

[7] Procesamiento del lenguaje natural (PLN). Instituto de ingeniería del conocimiento, disponible en <https://www.iic.uam.es/inteligencia/que-es-procesamiento-del-lenguaje-natural/>

[8] RDFace: RDFa Content Editor, disponible en

<https://nextweb.gnoss.com/recurso/rdface-rdfa-content-editor/f09c650e-3128-48ed-a99d-a8cd075d927a>

[9] BRAT, disponible en <http://brat.nlplab.org/index.html>

[10] Jon Saklofske. Brat Rapid Annotation Tool. Early Modern Digital Review, vol.2, disponible en <https://jps.library.utoronto.ca/index.php/renref/article/view/32993>

[11] TextPro – Text Processing Tools. Annotation layers, disponible en <http://textpro.fbk.eu/annotation-layers>