

Universidad de las Ciencias Informáticas



**Título: Módulo de traducción automática para el motor de
búsqueda ORION**

**Trabajo de Diploma para optar por el título de
INGENIERO EN CIENCIAS INFORMÁTICAS**

Autores: Yudeikis Caridad Orta Rodríguez
Samuel Pérez Guerrero

Tutores: Lic. Karel Antonio Verdecia Ortiz
Ing. Eyeris Rodríguez Rueda

La Habana, junio 2013

Declaración de Autoría

Declaramos ser autores del presente trabajo de diploma y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales del mismo, con carácter exclusivo. Autorizamos a dicho centro para que haga el uso que estime pertinente de este trabajo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Yudeikis Caridad Orta Rodríguez

Autor

Samuel Pérez Guerrero

Autor

Lic. Karel Antonio Verdecia Ortiz

Tutor

Ing. Eyeris Rodríguez Rueda

Tutor

Agradecimientos

Yudeikis

Mamita linda, quiero agradecerte por siempre brindarme tu amor, apoyo, preocupación y sabiduría. Gracias por darme todo en la vida y hacerme la persona que soy hoy. Eres lo más grande que tengo y espero que lo sepas.

A mi hermano Lesdy, quien siempre ha estado a mi lado, cuidando de mí desde pequeña y dándome consejos para ser cada vez una mejor persona. Más que un hermano, has sido para mí un amigo, un compañero y sobre todo, un padre.

A mi amiga de la infancia Vily, por apoyarme en momentos buenos y malos, por tantos años compartiendo estudios, discusiones, historias, secretos, en fin, miles de cosas que te hacen una amiga especial y una hermana para mí.

A mi novio Yasef, con quien he aprendido tantas cosas y que ha hecho de mí una persona más madura ofreciéndome su amor, cariño, apoyo y esos consejos tan acertados que siempre tengo que escuchar. Eres esa persona especial que espero nunca se aleje de mi vida.

A mi suegra Jorgelina, por su ayuda, cariño y reflexiones que también me han ayudado mucho.

A mi amigo Javier, gracias por tus cariños, preocupaciones y risas, gracias por esta amistad tan linda desde que nos conocimos hace cuatro años, sin interés alguno y siempre ayudándome cuando más lo he necesitado.

A todos aquellos que he conocido en estos años, gracias a todos por permitirme compartir con ustedes tantos momentos.

Gracias a todos.

Agradecimientos

Samuel

Agradezco en primer lugar a mis padres. Gracias por entregarlo todo para construir mi futuro, por educarme de la forma que lo han hecho, por darme todo su amor y cariño.

Gracias a mis hermanos Sandro y Zenia, que sé que un día seguirán mis pasos.

A mi abuela Vilma que siempre ha soñado con el día de ver a su nieto convertido en un profesional.

Gracias a mis tíos por ayudarme en todo momento, en especial a mi tía Maribel, Mary como le decimos cariñosamente que ha sido una segunda madre para mí.

A mis amigos por los buenos momentos que pasamos juntos y los recuerdos que quedaron grabados para siempre, por todo el apoyo y la ayuda brindada.

Agradezco a nuestros tutores por su apoyo incondicional, dedicación, preocupación e interés de ayudarnos en todo momento.

A todos los que de una forma u otra han contribuido a la realización de este sueño.

Dedicatoria

Yudeikis

A la memoria de mi abuela querida, que siempre soñó con verme graduada y convertida en una profesional.

A mi madre que tanto se esforzó por convertir ese sueño en realidad, por ser siempre mi guía y ejemplo a seguir, por su paciencia y dedicación y por su apoyo incondicional.

Samuel

Dedico este trabajo a mi abuela Vilma por siempre apoyarme y demostrarme su cariño, por todo el tiempo y amor que me dedica.

A mis padres por siempre estar ahí para mí, por ser siempre mis guías y ejemplo a seguir, por su paciencia y dedicación y por su apoyo incondicional en todo.

A mi hermano que lo adoro, espero siempre ser para ti más que un hermano, un amigo.

A mi hermanita por ser la persona más linda y angelical que existe en el mundo.

Resumen

La lingüística computacional es una disciplina que en los últimos años se ha dedicado, en gran medida, al estudio e investigación para automatizar el proceso de traducción, realizado por traductores humanos hasta la primera década del pasado siglo.

A pesar de que en la actualidad muchos son los servicios disponibles que permiten realizar traducciones automáticas, Cuba trabaja hoy en pos de alcanzar la soberanía tecnológica. Como parte de las transformaciones que se realizan, actualmente se desarrolla, en la Universidad de las Ciencias Informáticas, el buscador ORION, cuyo objetivo fundamental es brindar servicios para la recuperación eficiente de información. Sin embargo, gran parte de las páginas web indexadas por el buscador se encuentran en idioma español, pues están comprendidas bajo el dominio punto cu.

Partiendo de ello, en el presente trabajo se realiza un estudio de herramientas de traducción automática que brindan servicios en la actualidad, así como de diferentes motores de traducción que apoyan y complementan su funcionamiento. Teniendo en cuenta el análisis realizado, se lleva a cabo el diseño e implementación de un módulo para el motor de búsqueda ORION, que permite realizar traducciones automáticas, tanto de textos, como de páginas web, entre los idiomas inglés y español. Asimismo permite a los usuarios emitir su valoración acerca de la calidad de las traducciones realizadas.

Palabras claves: motor de traducción, ORION, traducción automática.

Índice

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA.....	5
1.1 Introducción.....	5
1.2 Conceptos asociados al dominio del problema.....	5
1.3 Clasificación de los sistemas de traducción automática.....	7
1.4 Sistemas informáticos vinculados al campo de acción.....	10
1.4.1 Altavista.....	10
1.4.2 Google.....	11
1.4.3 Bing.....	13
1.4.4 Servicio de traducción automática del Instituto Cervantes.....	15
1.5 Motores de traducción automática.....	17
1.5.1 PROMT.....	17
1.5.2 SYSTRAN.....	18
1.5.3 Reverso.....	19
1.5.4 Apertium.....	20
1.6 Metodología de desarrollo.....	21
1.6.1 Open Unified Process (OpenUP).....	21
1.7 Lenguaje de modelado.....	23
1.7.1 Lenguaje Unificado de Modelado.....	23
1.8 Herramienta CASE a utilizar.....	24
1.8.1 Visual Paradigm for UML 6.4.....	24
1.9 Lenguajes de programación.....	25
1.9.1 Lenguaje de Marcado de Hipertexto HTML.....	26
1.9.2 CSS.....	26
1.9.3 JavaScript.....	26
1.9.4 PHP.....	27
1.10 Marco de trabajo.....	27
1.10.1 Symfony.....	28
1.11 Entorno de Desarrollo Integrado (IDE).....	28
1.11.1 NetBeans.....	29
1.12 Servidor Web.....	29
1.12.1 Apache 2.....	29
1.13 Sistema Gestor de Base de Datos (SGBD).....	30
1.13.1 MongoDB.....	31
1.14 Conclusiones.....	31
CAPÍTULO 2. ANÁLISIS Y DISEÑO DEL SISTEMA.....	32
2.1 Introducción.....	32
2.2 Propuesta de sistema.....	32
2.3 Modelado del Dominio.....	33

Índice

2.3.1 Descripción de Clases del Modelo del Dominio	33
2.3.2 Diagrama de Clases del Modelo del Dominio	34
2.4 Especificación de los Requisitos del Software.	35
2.4.1 Requisitos funcionales	35
2.4.2 Especificación de requisitos funcionales	36
2.4.3 Requisitos no funcionales.....	38
2.5 Modelo de Casos de Uso del Sistema	39
2.5.1 Diagrama de Casos de Uso del Sistema	39
2.5.2 Patrones de casos de uso utilizados	40
2.5.3 Especificación de casos de uso.....	41
2.6 Estilo arquitectónico.....	44
2.7 Patrones de diseño.....	45
2.8 Modelo de Diseño.....	47
2.8.1 Diagrama de clases del diseño.....	47
2.9 Diagrama de Interacción.....	49
2.9.1 Diagramas de secuencia.....	49
2.10 Diseño de la Base de Datos.....	51
2.11 Modelo de Despliegue	51
2.12 Conclusiones	52
CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBA	53
3.1 Introducción	53
3.2 Diagrama de componentes.....	53
3.3 Estándar de codificación.....	54
3.4 Interfaces principales de la aplicación.....	56
3.5 Validación del sistema.	58
3.5.1 Pruebas funcionales.....	58
3.5.2 Pruebas de integración	61
3.5.3 Pruebas de carga y estrés.....	62
3.6 Conclusiones	63
CONCLUSIONES GENERALES.....	64
RECOMENDACIONES	65
REFERENCIAS BIBLIOGRAFICAS	66
BIBLIOGRAFÍA.....	70
GLOSARIO DE TÉRMINOS	71
ANEXOS.....	72

Índice de tablas

ÍNDICE DE TABLAS

Tabla 1: Conceptos asociados al dominio.	34
Tabla 2: Especificación del RF “Gestionar la traducción de texto de un idioma origen a un idioma destino”	37
Tabla 3: Especificación del RF “Permitir la introducción de la dirección URL de la página web a traducir”	37
Tabla 4: Descripción del CUS “Gestionar traducción de textos”	41
Tabla 5: Descripción del CUS “Permitir introducción de dirección URL”	43
Tabla 6: Caso de prueba para el CUS “Gestionar traducción de textos”	59
Tabla 7: Caso de prueba para el CUS “Permitir introducción de dirección URL”	61

Índice de figuras

ÍNDICE DE FIGURAS

Figura 1.1: Funcionamiento de un sistema basado en reglas.....	8
Figura 1.2: Interfaz del buscador Altavista	11
Figura 1.3: Interfaz del traductor BabelFish integrado al buscador Altavista.....	11
Figura 1.4: Interfaz del buscador Google	13
Figura 1.5: Interfaz de Google Translate	13
Figura 1.6: Interfaz del buscador Bing.....	14
Figura 1.7: Interfaz de Bing Translator	14
Figura 1.8: Interfaz del servicio de traducción del Instituto Cervantes	16
Figura 2.1: Modelo de dominio.....	34
Figura 2.2: Diagrama de casos de uso del sistema.....	39
Figura 2.3: Patrón de caso de uso: Reutilización	41
Figura 2.4: Patrón de caso de uso: Adición.....	41
Figura 2.5: Patrón arquitectónico MVC	45
Figura 2.6: Diagrama de clases del diseño con estereotipos web del CUS “Gestionar traducción de textos”	48
Figura 2.7: Diagrama de clases del diseño con estereotipos web del CUS “Permitir introducción de URL”	49
Figura 2.8: Diagrama de secuencia del CUS “Gestionar traducción de textos”	50
Figura 2.9: Diagrama de secuencia del CUS “Permitir introducción de dirección URL” para la interfaz “Traducir”	50
Figura 2.10: Modelo físico de datos	51
Figura 2.11: Modelo de despliegue	52
Figura 3.1: Modelo de componentes	53
Figura 3.2: Interfaz para la traducción de textos	57
Figura 3.3: Interfaz de una página web traducida.....	57
Figura 3.4: Porcentaje de No Conformidades detectadas	61

Introducción

INTRODUCCIÓN

Desde su propio nacimiento, la informática ha constituido un pilar esencial en el desarrollo de la sociedad mundial. Aunque el concepto “Informática” no aparece hasta mediados del siglo XX, desde el siglo IV a.C., con la invención del ábaco, comienza a nacer y desarrollarse. En la actualidad es una de las actividades más aplicativas en el mundo, debido a la gran cantidad de esferas a las cuales puede moldearse y adaptarse. La comunicación es uno de estos campos que ha experimentado un gran avance gracias a la informática, y precisamente dentro de esta área, se encuentra la traducción, la cual ha constituido un desafío importante para la humanidad a lo largo de su historia.

Con el devenir del tiempo, se fueron formando las lenguas madre, que dieron origen a los diferentes dialectos y lenguas actuales, existiendo unos 6.500 idiomas aproximadamente en el mundo [1]. Con esta diversidad de idiomas y dada la laboriosidad del proceso de traducción, desde la década de 1940 se han intentado encontrar medios para automatizar dicho proceso.

En el caso particular de Cuba, este asunto se amplía con los retos que enfrenta por su tránsito hacia la “Sociedad de la Informatización”, donde el idioma se afianza como Industria, Ciencia y Tecnología, en buena medida gracias al avance de la información, la computación y las telecomunicaciones.

Actualmente existen 68 instituciones de nivel superior que incluyen 3150 sedes universitarias municipales [2]. Como parte del desarrollo científico alcanzado en dichas instituciones se ha generado un gran volumen de conocimiento, en muchos casos documentado en artículos científicos publicados en la intranet cubana. Específicamente en el plano de la salud, Cuba se ubica hoy entre los ocho primeros países de Latinoamérica que más publicaciones científicas ha realizado, junto a Brasil, Argentina, México, Chile y Venezuela. Desde el año 1980 y hasta el 2002 se habían publicado alrededor de 3650 documentos de corte científico [3].

Parte de dicho conocimiento generado en las instituciones cubanas es publicado en revistas seriadas o especializadas accesibles desde la intranet cubana, de forma tal que pueda aprovecharse el resultado de la misma en otros lugares donde existan condiciones para su utilización. Sin embargo, de los 3650 documentos publicados, 3640 se encuentran en idioma español, 4 en portugués, y solo 6 en inglés, lo cual limita su comprensión por parte de los usuarios angloparlantes que puedan acceder a ellos.

En la esfera del periodismo existen alrededor de 200 blogs, según la Unión de Periodistas y Escritores de Cuba [4], que presentan la realidad cubana de forma amable o mediante la crítica constructiva y no solo

Introducción

tocan temas de actualidad política, sino que hablan de cultura, educación, etc. Dentro de esta comunidad bloguera solo un blog se encuentra en idioma inglés.

Otro elemento que atenta contra la comprensión de la información generada en Cuba, lo constituye la utilización de sistemas para la gestión de contenidos en la creación de portales e intranets. Según datos estadísticos publicados por el Centro Cubano de Información de Red (CUBANIC), se muestra que la red cubana presenta un total de 2339 dominios de primer nivel, o sea, punto cu, hasta noviembre de 2012. Muchos de los sitios web publicados bajo estos dominios se crean a partir de los sistemas ya mencionados, los cuales no ofrecen la posibilidad de traducción automática de sus contenidos a otros idiomas, pues este proceso debe hacerse manualmente por los administradores de dichos portales, por lo que en ocasiones no son de utilidad para los usuarios de diferentes países que puedan visitar estas páginas.

Todas las limitaciones antes mencionadas, impuestas por la barrera del idioma, traen consigo disímiles consecuencias tanto para la economía como para el desarrollo científico del país. Resulta evidente que no puede existir un aumento del desarrollo social e ideológico si el avance de la ciencia está detenido por barreras en la comunicación. Por tanto, se hace imprescindible internacionalizar toda la información científica generada en el país que se encuentra publicada en la intranet cubana, con el fin de lograr un ambiente colaborativo y de intercambio, en pos de alcanzar un mayor desarrollo científico y social.

Al mismo tiempo, como parte de la guerra mediática, la verdad sobre Cuba necesita ser difundida ampliamente, con el fin de contrarrestar los efectos que causan los grandes medios de prensa internacionales, encargados de especular y tergiversar la situación existente. Los medios cubanos difunden la realidad a través de pruebas contundentes que lo demuestran, pero hoy en día dicha información se encuentra restringida, por la barrera del idioma, solo a usuarios de habla hispana, teniendo en cuenta la cantidad de blogs que se encuentran en idioma español. Por tanto, la falta de comprensión de los contenidos publicados en el país conlleva al desconocimiento de la verdadera situación existente, por parte de los usuarios que navegan en la red de redes.

En la era de la comunicación y de las nuevas tecnologías, la difusión de información no debería tener obstáculos y, por lo tanto, la diversidad de lenguas existentes no debería constituir una barrera. Las alternativas existentes en la actualidad en Internet para la traducción automática de contenidos web, suponen varios riesgos y dificultades para Cuba. Algunos sistemas como Google Translate han restringido la utilización de sus servicios, convirtiéndolo en un servicio comercial, y en otros casos, como la herramienta de traducción gratuita que ofrecen buscadores como Bing, Yahoo, Altavista, suponen un

Introducción

riesgo de soberanía tecnológica, ya que los usuarios cubanos requerirían de un acceso total a Internet para poder traducir desde el sitio web donde se encuentren así como que en algún momento estos sistemas puedan restringir el acceso de Cuba a sus servicios.

En la Universidad de las Ciencias Informáticas (UCI) actualmente se encuentra en desarrollo el motor de búsqueda ORION, cuyo principal objetivo es brindar servicios para la recuperación eficiente de información en fuentes de datos estructuradas y no estructuradas. El buscador será accesible para usuarios de todas las regiones del mundo y los contenidos web, resultados de las búsquedas mediante el mismo, serán páginas web comprendidas bajo el dominio punto cu, por lo que la gran mayoría se encuentran en idioma español, sin posibilidad de traducción automática.

Sobre la base de los elementos expuestos anteriormente se formula el siguiente **problema a resolver**:
¿Cómo contribuir a la traducción automática de los contenidos publicados en la intranet cubana?

Teniendo como **objeto de estudio** de la investigación el proceso de traducción automática.

Delimitando como **campo de acción**: La traducción automática de textos y páginas web.

En pos de resolver el problema antes mencionado se plantea el siguiente **objetivo general**: Desarrollar un módulo de traducción automática de textos y de las páginas web indexadas por el motor de búsqueda ORION, a los idiomas inglés y español, para contribuir a la traducción automática de los contenidos publicados en la intranet cubana.

Objetivos específicos:

- ✓ Caracterizar el marco teórico conceptual y el estado del arte respecto a las tecnologías actuales para el desarrollo de herramientas de traducción.
- ✓ Diseñar un módulo que permita la traducción automática de textos y de las páginas web indexadas por el motor de búsqueda ORION.
- ✓ Implementar un módulo que permita la traducción automática de textos y de las páginas web indexadas por el motor de búsqueda ORION.
- ✓ Validar el correcto funcionamiento del módulo de traducción automática.

Teniendo en cuenta lo anterior, se plantea la siguiente **idea a defender**: El desarrollo de un módulo de traducción automática, a los idiomas inglés y español, permitirá una mejor comprensión de las páginas web cubanas a los usuarios.

Introducción

Para dar cumplimiento al objetivo general se plantean las siguientes **Tareas de Investigación:**

- ✓ Realización de un levantamiento bibliográfico para estudiar los principales elementos teóricos y conceptos que permitan analizar el estado actual del desarrollo de tecnologías y herramientas de traducción automática.
- ✓ Estudio de sistemas homólogos existentes en la actualidad.
- ✓ Estudio de las herramientas de traducción automática seleccionadas.
- ✓ Definición de la metodología a utilizar en el desarrollo del módulo de traducción automática.
- ✓ Identificación de los requisitos funcionales y no funcionales.
- ✓ Documentación de los artefactos generados durante los procesos de diseño e implementación del módulo de traducción automática según la metodología seleccionada.
- ✓ Implementación del módulo que permita la traducción automática de textos, así como de páginas web.
- ✓ Integración del módulo con el motor de búsqueda ORION.
- ✓ Selección de las técnicas de validación útiles para el módulo de traducción.
- ✓ Aplicación de pruebas funcionales, de integración y de rendimiento al módulo de traducción automática.

Los **Métodos Científicos** en los cuales se basa la investigación son los métodos teóricos. Estos permiten organizar el trabajo, lo cual posibilita el desarrollo de un amplio conocimiento referente al estado del arte tratado, su evolución en etapas determinadas, así como adquirir de diversas bibliografías elementos importantes para el desarrollo del mismo.

Métodos Teóricos:

- ✓ Histórico – Lógico: Permite constatar teóricamente cómo ha evolucionado el proceso de traducción automática desde su surgimiento, así como la evolución y desarrollo de las principales herramientas existentes en la actualidad.
- ✓ Analítico – Sintético: Permite el análisis de documentos y teorías acerca del proceso de traducción automática y la extracción de los elementos más importantes relacionados a este proceso.
- ✓ Método Inductivo-Deductivo: Se emplea para que luego de adquirir una serie de elementos referentes al proceso de traducción automática, fuese posible arribar a razonamientos que puedan ser aplicables al problema a resolver.

Introducción

El documento está estructurado de la forma siguiente:

Capítulo 1. Fundamentación Teórica:

Se hace un estudio del estado del arte sobre las herramientas existentes en la actualidad con características similares en cuanto a objetivos de uso y de funcionamiento. Se efectúa un análisis de las metodologías de desarrollo, los lenguajes de modelado y herramientas, seleccionando las que serán utilizadas en el desarrollo del presente trabajo.

Capítulo 2. Análisis y Diseño del sistema:

Se determinan las capacidades o funciones que el sistema debe cumplir (requisitos funcionales) y las propiedades o cualidades que el producto debe tener (requisitos no funcionales). Se realiza, además, una descripción de los casos de uso y se muestra el diagrama de casos de uso del sistema. También se realiza el diseño de la aplicación, el cual queda reflejado en el diagrama de clases del diseño y los diagramas de secuencia; además se describe la distribución del sistema propuesto a través del diagrama de despliegue.

Capítulo 3. Implementación y prueba:

Se realiza el diagrama de componentes y su descripción, con el fin de comprender los elementos de software del módulo y sus relaciones. Se muestran las pantallas principales del módulo desarrollado donde se evidencian las funcionalidades implementadas. Además se realiza la validación del diseño realizado mediante un conjunto de casos de prueba con sus respectivos resultados, así como con la realización de pruebas de integración y pruebas de carga y estrés.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

La traducción automática ha evolucionado junto al desarrollo informático. A pesar de no haber alcanzado la perfección, actualmente es utilizada en los principales sectores de la sociedad, sobre todo en la red de redes, donde se ha integrado a diversos buscadores, medios de prensa e instituciones. En el presente capítulo se realiza un estudio de estas aplicaciones web en busca de un mejor entendimiento del estado actual y desarrollo de dichos sistemas. Para realizar un análisis exhaustivo del tema se tiene en cuenta el estudio de algunas herramientas de traducción automática presentes en la web, logrando así una mejor comprensión de su funcionamiento. Tomando como base el análisis realizado se define la metodología de desarrollo, tecnologías y herramientas que se usarán en el desarrollo de la solución.

1.2 Conceptos asociados al dominio del problema

Traducción

Cinco son los enfoques principales que se han sucedido a la hora de buscar una definición completa:

✓ **Actividad entre lenguas:**

Según el filólogo y traductor español Valentín García Yebra, la traducción consiste en enunciar en otra lengua lo que ha sido enunciado en una lengua fuente, conservando las equivalencias semánticas y estilísticas. [5]

Asimismo para los teóricos J. P. Vinay y J. Darbelnet no es más que pasar de una lengua A a una lengua B para expresar la misma realidad. [6]

✓ **Actividad entre textos:**

Dentro de los exponentes de este enfoque se encuentran Danica Selescovitch y Marianne Lederer, quienes sentaron las bases de una teoría interpretativa de la traducción y la definieron como la sustitución de un texto en lengua de partida por un texto semántica y pragmáticamente equivalente en lengua meta. [7]

✓ **Acto de comunicación:**

Fundamentación teórica

De acuerdo a la lingüista estadounidense Eugene Nida, la traducción se basa en reproducir en la lengua terminal el mensaje de la lengua original por medio del equivalente más próximo y más natural, primero en lo que se refiere al sentido, y luego en lo que atañe al estilo. [8]

Basil Hatim y Ian Mason, destacados lingüistas, la definen como un proceso comunicativo que tiene lugar en un contexto social. [9]

✓ **Proceso:**

Según el teórico y escritor Jean Delisle la traducción es una operación que consiste en determinar la significación de los signos lingüísticos en función de un querer decir concretizado en un mensaje, y restituir después ese mensaje íntegramente mediante los signos de otra lengua. [10]

✓ **Actividad textual, comunicativa y cognitiva:**

Para la traductora y académica española Amparo Hurtado Albir es un proceso interpretativo consistente en la reformulación de un texto con los medios de una lengua que se desarrolla en un contexto social y con una finalidad determinada. [11]

A partir de los conceptos expuestos anteriormente se determinó la traducción como una actividad que consiste en comprender el significado de un texto en un idioma, llamado *texto origen*, para producir un texto con un significado semántico equivalente, en otro idioma, llamado *texto traducido* o *de salida*.

Motor de búsqueda

Un motor de búsqueda se define como un sistema informático que busca archivos almacenados en servidores web, a partir de palabras claves ingresadas por el usuario, y muestra como resultado un listado de direcciones web en los que se mencionan temas relacionados con las palabras clave buscadas. [12]

Traducción automática

La filóloga española Pilar Hernández Mercedes define la traducción automática como el proceso por el cual una máquina traduce un texto de una lengua a otra, subdividiendo la sintaxis, identificando las partes del discurso, intentando resolver eventuales ambigüedades y, por último, traduciendo los componentes y la estructura en la lengua de destino. [13]

De acuerdo a los reconocidos lingüistas Daniel Jurafsky, James H. Marti y Wayne Ward consiste en utilizar ordenadores para automatizar algunos o todos los procesos que intervienen en la traducción de una lengua a otra. [14]

Fundamentación teórica

Partiendo de estas definiciones se puede decir que la traducción automática es el proceso mediante el cual se utiliza software informático para traducir textos de un lenguaje natural a otro, teniendo en cuenta la estructura gramatical de cada idioma y haciendo uso de reglas para transformar la estructura gramatical del texto a traducir al texto traducido.

1.3 Clasificación de los sistemas de traducción automática.

De acuerdo a su funcionamiento, los sistemas de traducción automática se pueden clasificar teniendo en cuenta diferentes criterios. [13]

Según el número de lenguas

- ✓ Bilingües, diseñados para traducir de una lengua de origen a una lengua destino.
- ✓ Multilingües, diseñados para traducir de varias a varias lenguas.

Según la dirección

- ✓ Unidireccional, si trabaja en una sola dirección.
- ✓ Bidireccional, si trabaja en ambas direcciones.

Según el grado de automatización

- ✓ Sistemas de traducción completamente automática: realizan la traducción sin intervención del traductor humano.
- ✓ Sistemas de traducción automática asistida: la intervención del traductor humano es constante.
- ✓ Sistemas de traducción humana asistida por ordenador: el ordenador se utiliza como herramienta de ayuda al traductor, facilitándole el uso de, por ejemplo, procesadores de textos con diccionarios en línea, información gramatical y morfológica de las palabras, verificación ortográfica, etc.

Según el método de obtención de la información

Las estrategias para llevar a cabo un proceso de traducción automática se engloban en tres categorías generales [15]:

- ✓ **Traducción automática basada en reglas**

Fundamentación teórica

En la traducción automática basada en reglas se considera la traducción como una tarea en la que el objetivo es lograr equivalencias semánticas. Este método proyecta el análisis de los textos de origen en un tipo de representación a partir del cual se genera el texto meta.

A continuación se muestra una imagen donde se pone de manifiesto el funcionamiento de la traducción automática basada en reglas

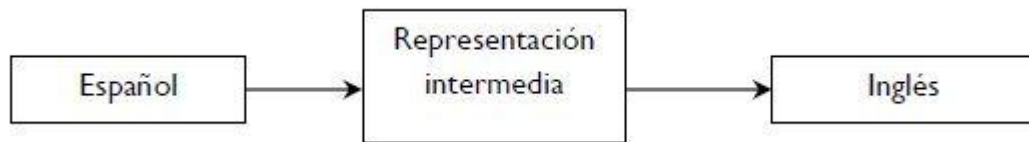


Figura 1.1: Funcionamiento de un sistema basado en reglas.

En este método se pueden distinguir dos variantes: traducción automática por lenguaje intermedio y la traducción automática por transferencia.

Traducción automática por lenguaje intermedio

El método interlingua concibe la traducción en dos fases, pasando por una única representación intermedia de carácter universal, común a las lenguas entre las que se va a traducir. La idea inicial es representar el texto origen en una lengua intermedia abstracta, la cual sería independiente a la lengua inicial y final, para posteriormente traducirlo a la lengua destino. Dicha lengua posee todas las características morfológicas, sintácticas y semánticas de los idiomas a los que serán traducidos, en lugar de traducir directamente de la lengua fuente a la lengua objetivo.

Traducción automática por transferencia

La base fundamental de la traducción automática por transferencia es la interlingua y es uno de los métodos de traducción más usados actualmente. Propone dos representaciones intermedias, una por cada lengua del par de traducción. Estas representaciones dependen de la lengua que caracterizan y su función es convertir las representaciones intermedias de la lengua de origen en representaciones intermedias de la lengua meta.

Este método realiza el proceso de traducción en tres fases:

Análisis: A partir del texto de la lengua de origen se obtiene una representación léxica, semántica y/o sintáctica.

Fundamentación teórica

Transferencia: Se establece equivalencias entre las representaciones del par de lenguas objeto de la traducción.

Generación: A partir del módulo de transferencia se produce el texto en lengua meta.

✓ **Traducción automática basada en corpus paralelos**

El funcionamiento de la traducción automática basada en corpus paralelos se basa principalmente en el análisis de muestras reales con sus respectivas traducciones. Este método ha experimentado su mayor desarrollo en una tecnología conocida como "memoria de traducción" que consiste en almacenar traducciones, realizadas manualmente y validadas por un traductor humano, para reutilizarlas posteriormente en la traducción de textos similares.

Entre los modelos que utilizan corpus se incluyen los métodos estadísticos y los basados en ejemplos.

Traducción automática basada en estadística

La traducción automática estadística es un paradigma de traducción automática que centra en el resultado. Su objetivo es generar traducciones a partir de modelos de traducción estadísticos cuyos parámetros emanan del análisis de corpus monolingües y bilingües.

La idea básica para traducir mediante técnicas estadísticas, es la de utilizar los denominados corpora paralelos (textos traducidos por un humano) para extraer de ellos la información estadística necesaria, con el objetivo de "entrenar" el sistema de traducción automática. En la actualidad se dispone de un conjunto considerable de estos textos, aunque solo para algunos idiomas.

Traducción automática basada en ejemplos

La traducción automática basada en ejemplos, se basa en el uso de un corpus bilingüe como principal fuente de conocimiento en tiempo real. Utiliza información estadística pero con métodos diferentes. O sea, el mismo no utiliza algoritmos estadísticos a la hora de traducir, sino para encontrar ejemplos en determinados contextos. Se trata de anotar y clasificar, mediante la alineación oracional de textos paralelos, las formas en que un fragmento de texto de la lengua de origen ha sido traducido a una lengua meta y usar esa información para elegir traducciones de esos fragmentos en otros textos.

✓ **Traducción automática basada en el contexto**

Fundamentación teórica

La traducción automática basada en el contexto es un método novedoso desarrollado por la compañía Meaningful Machines. Utiliza técnicas basadas en hallar la mejor traducción para una palabra, fijándose en el resto de palabras que la rodean. Básicamente este método se basa en tratar el texto en unidades de entre cuatro y ocho palabras, de manera que se traduce cada una de ellas por su traducción al idioma destino y se eliminan las traducciones que han generado una "frase" sin sentido, dejando solo las frases coherentes. Se repite dicho paso para todo el texto.

1.4 Sistemas informáticos vinculados al campo de acción

A nivel mundial se han desarrollado disímiles software que permiten la traducción tanto de textos y documentos como de páginas web. Aunque no se puede afirmar que han alcanzado la perfección, realmente algunos traducen con una extraordinaria exactitud, sobre todo si traducen dos pares de lenguas que sean comunes sintácticamente tales como el italiano y el español, sin embargo, cuando se trata de traducir lenguas que son muy diferentes entre sí, como el ruso y el español, es donde surgen ambigüedades.

En la actualidad existen en el mundo diversas aplicaciones web que permiten traducir automáticamente tanto textos, como documentos o páginas web, incluso, algunas de ellas permiten la integración de su herramienta de traducción a otros sitios web.

1.4.1 Altavista

Es el primer buscador completo en Internet que hizo su debut como motor de búsqueda en diciembre de 1995, permitiendo realizar búsquedas de texto sobre una base de datos. Ya en 1998 Altavista presenta su herramienta de traducción BabelFish, que constituye el primer servicio de traducción mecanizada de la Web que puede traducir palabras, frases o sitios web completos en tiempo real conforme el usuario va navegando, a través del motor de traducción automática SYSTRAN. [16]

Presenta una interfaz muy sencilla con un primer campo en el que se pueden insertar textos y un segundo campo para especificar la página web que se quiere traducir, elemento este a tener en cuenta en el desarrollo del módulo de traducción automática para el buscador ORION. En total, ofrece 19 pares de lenguas: del inglés al chino, francés, alemán, italiano, japonés, coreano, portugués y español; del chino al inglés; del francés al alemán y al inglés; del español al inglés; del alemán al inglés y al francés; del italiano al inglés; del japonés al inglés; del coreano al inglés; del portugués al inglés y del ruso al inglés.

Fundamentación teórica

Un elemento significativo del buscador Altavista es que permite integrar su servicio de traducción a otros sitios web, pero, independientemente de las ventajas que pueda propiciar dicha integración, solo brinda 12 pares de idiomas para traducir desde otra aplicación. Asimismo solo permite un límite de 150 palabras para realizar la traducción automática online, tanto desde el propio Altavista como desde cualquier otro sitio.

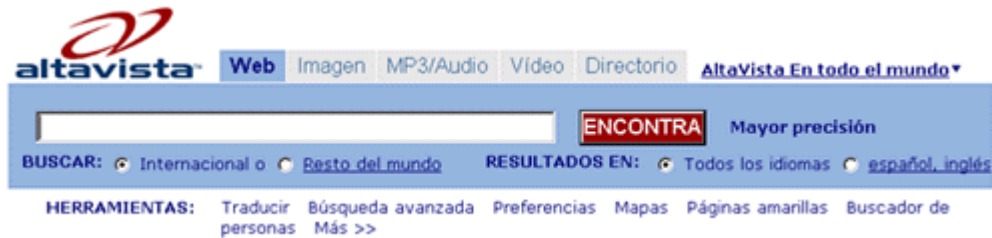


Figura 1.2: Interfaz del buscador Altavista



Figura 1.3: Interfaz del traductor BabelFish integrado al buscador Altavista

1.4.2 Google

Google es un motor de búsqueda en la web propiedad de Google Inc. Entre sus principales servicios ofrece traducción automática gratuita, basada en datos estadísticos proporcionados por Google para traducir texto a 64 idiomas distintos. Este traductor permite traducir textos entre multitud de idiomas en unos pocos segundos, es muy fácil de utilizar y permite trabajar con él de múltiples formas para realizar traducciones, no solo de textos, también de documentos y páginas web.

Fundamentación teórica

La interfaz es muy fácil de manejar. Presenta un campo en el que se puede introducir texto, URL de una página web o subir un documento de diferentes formatos: PDF, TXT, DOC, PPT, XLS o RTF.

En el caso específico de que el usuario introduzca una dirección URL, el sistema muestra otra interfaz con la página web traducida; proceso similar ocurre cuando el usuario pulsa el enlace “Traducir esta página”, en alguna de las páginas indexadas durante una búsqueda. Para el desarrollo del módulo de traducción automática se tendrá en cuenta el proceso descrito anteriormente para la traducción de páginas web.

Actualmente se puede traducir tanto directa como inversamente con los 64 idiomas disponibles y algunos están en fase de prueba para poder ser incluidos. Este sistema puede realizar la traducción de manera simultánea o (si se inhabilita esta opción) hacerlo después pinchando en el botón azul «Traducir». Además, si el usuario mueve el cursor sobre el texto traducido y pincha en una determinada palabra, podrá ver las diferentes propuestas de traducción para la misma e incluso contribuir con una traducción que el usuario considere mejor y más apropiada. También hay que señalar que el sistema ofrece otras prestaciones como, por ejemplo: “Teclado virtual”, “Escuchar”, “Leer fonéticamente”, “Grabar y traducir la voz”, etc. [17]

También permite al usuario seleccionar un criterio de acuerdo a la calidad de la traducción de textos, característica que se tendrá en cuenta en el desarrollo del módulo de traducción, con el objetivo de ofrecerle al usuario un ambiente colaborativo.

La traducción del inglés es la mejor, ya que para traducir un texto entre dos idiomas cualquiera que no sea uno de ellos el inglés, primero hace la traducción al inglés y después a cualquier otro idioma. Para generar dicha traducción analiza cientos de millones de documentos en busca de patrones que le ayuden a elegir la mejor traducción. La búsqueda de patrones en documentos que hayan sido traducidos por traductores humanos permite al traductor de Google decidir cuál sería la traducción más adecuada. Por lo que la calidad de la traducción será mejor cuanto mayor sea el número de documentos traducidos por traductores humanos analizados.

El buscador Google también permite la integración de su herramienta a otras aplicaciones web, pero la distribución de su interfaz de programación de aplicaciones (API, por sus siglas en inglés) es comercial, por lo que las personas que quieran integrar este servicio de traducción a su sitio deben pagar para consumirlo.

Fundamentación teórica



Figura 1.4: Interfaz del buscador Google



Figura 1.5: Interfaz de Google Translate

1.4.3 Bing

Es un buscador web de Microsoft anteriormente conocido como *Live Search*, *Windows Live Search* y *MSN Search*. El motor de búsqueda Bing posee un potente traductor que permite traducir entre diversos idiomas, e incluso, una vez hecha la traducción es posible escucharla. [18] Bing Translator traduce textos y páginas web en 39 idiomas utilizando como estrategia la traducción estadística.

Su interfaz presenta dos campos: uno para introducir el texto origen y otro donde se muestra el texto una vez traducido, característica que se aprovechará en el diseño de la interfaz del módulo de traducción automática. Para llevar a cabo el proceso de traducción, el usuario puede introducir en el primer campo tanto un texto origen como la dirección de una página web concreta. En este último caso, cuando se indica la URL de una página, Bing Translator abre una ventana nueva en la que se muestra la traducción de la página escogida. Además, el resultado se puede visualizar en función de diferentes vistas: «En

Fundamentación teórica

paralelo», «Arriba/Abajo», «Original con traducción flotante» y «Traducción con original flotante». Asimismo, cabe decir que la plataforma detecta automáticamente la lengua de partida, pero si el usuario lo prefiere, puede definirla personalmente a partir de los 39 idiomas disponibles. Por último, cuando la traducción ya está hecha, el usuario puede mover el cursor sobre cualquier oración del texto meta y así se mostraría la oración correspondiente en el texto origen, además puede invertir fácilmente la dirección de traducción. [19]

El buscador Bing es otra de las aplicaciones web que permite integrar su servicio de traducción automática a sitios web, pero presenta las siguientes desventajas:

- ✓ Permite hasta 850 palabras para la traducción de textos.
- ✓ Al integrar Bing Translator a otros sitios web, limita el uso de su servicio a 12 millones de palabras por mes.



Figura 1.6: Interfaz del buscador Bing

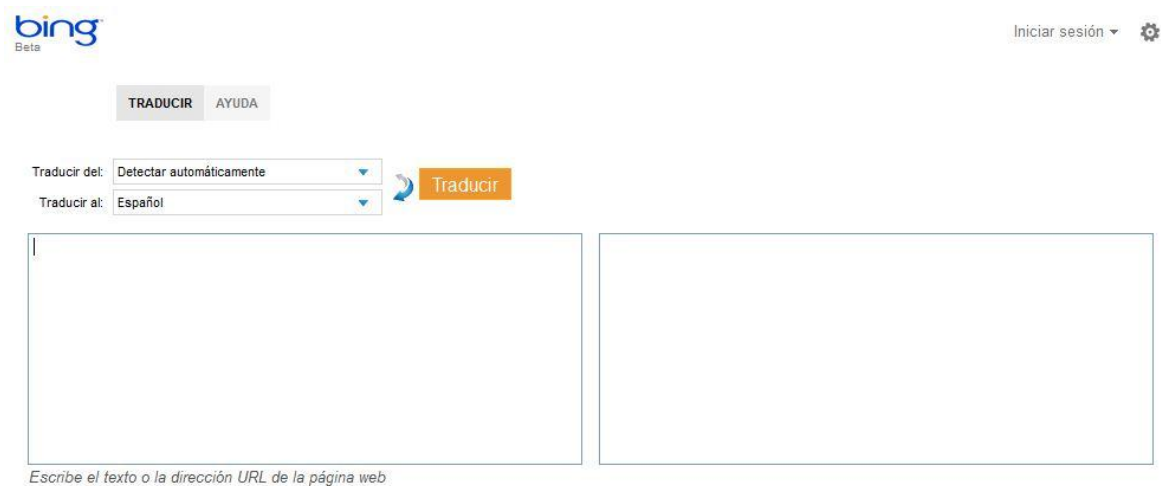


Figura 1.7: Interfaz de Bing Translator

Fundamentación teórica

1.4.4 Servicio de traducción automática del Instituto Cervantes

El Instituto Cervantes desde 2005 ofrece un servicio de traducción gratuito en la red, basado en el uso de la tecnología Opentrad, para traducir textos, documentos y páginas web sin restricción alguna. Opentrad es una plataforma de servicios de traducción automática desarrollada en código abierto, lo que permite a la comunidad utilizar, desarrollar y mejorar sus propias aplicaciones. El sistema se basa en dos motores de traducción de alta velocidad: uno de transferencia sintáctica parcial, Apertium, ideado para parejas de lenguas próximas y utilizado para todas las lenguas menos el euskera; y otro de transferencia sintáctica completa, Matxin, para parejas de lenguas divergentes, que es utilizado para el euskera. [20]

Presenta una interfaz muy intuitiva, la cual muestra un campo para introducir el texto origen y en el caso de la traducción de páginas web permite al usuario introducir la dirección URL que desea traducir en otro campo o seleccionarla. Al igual que la interfaz de la herramienta BabelFish, separa los campos para insertar textos y direcciones URL, elementos que se aprovecharán en el diseño del módulo de traducción en cuestión.

En el caso de la traducción de documentos soporta los formatos TXT, HTML, ODT, DOCX, PPTX y XLSX.

Desde su lanzamiento los usuarios han realizado más de 35 millones de traducciones. Actualmente, ejecuta una media de traducción de 800 millones de palabras al mes, lo que se convierte en 1,5 millones de traducciones.

Independientemente de las ventajas que pueda propiciar la aplicación, presenta una importante limitación ya que no permite integrar su servicio de traducción automática a otros sitios online.

The image shows the user interface of the Instituto Cervantes automatic translation service. At the top left is the logo, which consists of a stylized red 'I' and 'C' followed by the text 'Instituto Cervantes' and 'Servicio de traducción automática'. Below the logo, there is a red instruction: 'Para realizar una traducción gratuita* de una página web o de un texto, siga los siguientes pasos:'. Underneath, there is a 'Traducir de' label followed by a dropdown menu currently showing 'elegir idioma'. Below this, there are two main options: 'Opción WEB:' and 'Opción TEXTO:'. Under 'Opción WEB:', there is a sub-label 'Seleccione una página de la lista:' followed by a dropdown menu and a red 'Traducir' button. Below that is the sub-label 'O escribala en el siguiente cuadro:' followed by a text input field and another red 'Traducir' button. Under 'Opción TEXTO:', there is the sub-label 'Escriba el texto en el siguiente cuadro:' followed by a larger text input area and a red 'Traducir' button. At the bottom center, there is a red button labeled 'Borrar datos'.

Figura 1.8: Interfaz del servicio de traducción del Instituto Cervantes

Resumiendo...

A partir del análisis realizado se pueden resumir las principales limitaciones que presentan las aplicaciones web que ofrecen servicios de traducción automática en la actualidad, entre ellas se encuentran:

- ✓ Restringen el número de palabras a traducir.
- ✓ No ofrecen servicios externos, o sea, son solo sitios web de traducción automática o aplicaciones cuyos servicios no pueden ser integrados a otros sitios web.
- ✓ En caso de permitir la vinculación de su herramienta de traducción automática a otros sitios web, no es de manera gratuita.
- ✓ En caso de tener una versión gratuita limitan los pares de idiomas disponibles, así como la cantidad de palabras a traducir.

A pesar de dichas limitaciones, se han definido las características fundamentales que presentan algunas de estas aplicaciones web, identificando las buenas prácticas que se pueden aprovechar, principalmente los elementos relacionados con las interfaces de traducción tanto de textos como de páginas web para el desarrollo de un módulo de traducción automática que pueda ser integrado al motor de búsqueda ORION

Fundamentación teórica

de forma tal que pueda complementar el funcionamiento del mismo, teniendo como premisa la independencia tecnológica.

También resulta vital tener en cuenta la posibilidad de ofrecer a los usuarios un ambiente colaborativo, elementos estos que se pueden apreciar a nivel internacional.

1.5 Motores de traducción automática

Para el desarrollo del módulo de traducción automática es necesaria la utilización de un motor de traducción, el cual complementará el funcionamiento del módulo durante las traducciones, tanto de textos como de páginas web. Por tanto, a continuación se realiza un estudio de algunos de ellos, en pos de definirlo, teniendo en cuenta las características, ventajas y desventajas de cada uno de ellos.

1.5.1 PROMT

Con este motor de traducción automática se puede traducir el contenido de los sitios de Internet de un idioma a otro, incluidos los textos que llevan hipervínculos y los pies de las imágenes, respetando el formato HTML de la página de origen. Las imágenes, el contenido multimedia y los textos incluidos en JavaScript permanecen en el idioma de origen, sin traducir. Los pares de idiomas disponibles son: del español al inglés, alemán, francés y ruso (y viceversa). Además ofrece un "filtro" de traducción por área temática, lo que permite un refinamiento de la calidad de la traducción (por ejemplo: comunicación, negocios, informática, medicina, etc.).

Los requisitos de hardware necesarios para trabajar con este motor son:

Sistema operativo: Windows Server 2003 - 2000 Server - XP - 2000 Professional.

Procesador: Pentium III 500 o superior.

RAM: 128 MB (recomendado 256 MB).

Espacio en disco: 700 MB (según el número de combinaciones lingüísticas).

Basándose en la tecnología de PROMT, existen diversos traductores online que ofrecen servicios de traducción automática, algunos de ellos son:

- ✓ **Prompt Translation:** Permite la traducción de texto y páginas web, con diferentes pares de idiomas. Comprueba la ortografía, tiene un diccionario y permite imprimir los resultados. Tiene un límite de 3000 caracteres.

Fundamentación teórica

- ✓ **Free Translation:** Prácticamente con las mismas opciones, ofrece también las funciones de diccionario, de síntesis del habla y corrección ortográfica.

La principal limitación del motor de traducción PROMT es que es un software comercial y solo ofrece 30 días de prueba a las empresas e instituciones que deseen utilizarlo en su sitio web. Además se necesita de conexión a Internet para poder utilizar sus potencialidades. Por tanto, este motor de traducción automática no constituye una solución viable a la problemática en cuestión.

1.5.2 SYSTRAN

Diseñado en 1970 por Peter Toma, es uno de los sistemas comerciales de traducción automática más veteranos [21]. En los últimos años, la tecnología del sistema, tradicionalmente basada en la traducción por reglas, se ha renovado, con la incorporación fundamentalmente de un módulo para la gestión de memorias de traducción.

Es un sistema multiplataforma y es utilizado de manera gratuita en las múltiples instalaciones accesibles por Internet: Google, AOL, AltaVista. La empresa ofrece módulos preparados para más de 40 idiomas.

Además integra funciones multilingües muy útiles para aplicaciones como el comercio electrónico, CRM, bases de datos, correo electrónico, mensajería instantánea, SMS, WAP, etc. Cuenta con 50.000 palabras básicas y 250.000 términos científicos y las traducciones se procesan a una velocidad de 500.000 palabras por hora.

Para trabajar con este motor son necesarios algunos requisitos como: [22]

Sistema operativo: Windows Server 2003 SP2, R2 - 2008 SP2, R2 - Linux RedHat Enterprise 5, 6 - CentOS 5, 6.

Procesador: Dual-Core Intel Xeon o compatible (2 GHz).

RAM: 4 GB (recomendado 8 GB)

Espacio en disco: 5 GB (recomendado 50 GB o más)

Entre las ventajas que posee la utilización de SYSTRAN como motor de traducción automática se encuentran:

- ✓ Tiene en cuenta la morfosintaxis y semántica de cada palabra en cada idioma.
- ✓ Es un traductor que se moderniza día a día y ofrece tanto palabras básicas como términos científicos.

Fundamentación teórica

- ✓ Tiene en cuenta la estructura gramatical de cada idioma y usa diferentes reglas dependiendo del idioma.
- ✓ Traduce textos de un idioma a otro de manera instantánea y gratuita.

Con el motor de SYSTRAN hay varias páginas que ofrecen traducción automática:

- ✓ **Babelfish:** Ofrece todos los pares de idiomas que posee el motor, con una limitación de 150 palabras. También permite traducir páginas web.
- ✓ **AppliedLanguage:** Con los pares de idiomas de SYSTRAN, permite traducir pequeños textos, correos electrónicos y páginas web.
- ✓ **Bing Translator:** Es un servicio proporcionado por Microsoft como parte de sus servicios Bing que permiten a los usuarios traducir textos o páginas web en diferentes idiomas. Todas las traducciones son con tecnología de este motor de traducción automática.

A pesar de las características y ventajas mencionadas anteriormente, se decide no utilizar dicho sistema ya que, al utilizarlo de manera gratuita, sólo se puede traducir un máximo de 150 palabras. Por otra parte, no reconoce la palabra que el usuario desea traducir porque no tiene en cuenta el contexto del artículo y se confunde con los nombres propios y comunes, con el género y número de los sustantivos. Además, requiere de conexión a Internet para realizar las traducciones.

1.5.3 Reverso

Permite traducir textos, documentos, correos electrónicos y páginas web. La traducción de documentos puede ser total o parcialmente, directamente desde un documento o una hoja de cálculo. Además, este motor permite corregir y retocar los posibles errores de traducción de manera rápida y eficaz. Con Reverso puede conectar diccionarios especializados o configurar sus propios glosarios. También puede jerarquizar los diccionarios conectados según el orden de prioridad que usted decida. Reverso, al igual que PROMT analiza el contexto, o sea, no traduce palabra a palabra.

Para trabajar con este motor son necesarios algunos recursos como:

- ✓ Windows 98-NT-2000-ME-XP-Vista.
- ✓ 128 MB de memoria RAM.
- ✓ 300 Mb de disco duro por grupo de idiomas, alrededor de 2Gb para todos los idiomas.
- ✓ Word 2000, 2002 o Word XP, Internet Explorer v.5.5 o superior para la integración de las barras de herramientas.

Fundamentación teórica

Con el motor Reverso puede traducirse desde:

- ✓ **Su propia página:** Se pueden traducir breves textos, y cuenta con diccionarios.
- ✓ **El Mundo:** Con la posibilidad de traducir además páginas web.
- ✓ **Ya.com:** Para textos y páginas web.

La principal limitación de Reverso, al igual que los motores estudiados anteriormente, es que para utilizarlo se necesita de pleno acceso a la red de redes, por lo que se determina no utilizarlo para el desarrollo del módulo de traducción automática.

1.5.4 Apertium

Es un motor de traducción automática desarrollado en la Universidad de Alicante de España que se distribuye bajo la licencia GPL. Mediante las herramientas que ofrece se pueden desarrollar nuevos pares de lenguas o mejorar los pares de lenguas ya existentes. Es decir, además de distribuir todo lo necesario para poder usar el sistema de traducción automática también se distribuyen todas las herramientas necesarias para poder crear nuevos sistemas: [23]

- ✓ Un motor de traducción común para todos los pares de lenguas disponibles.
- ✓ Datos lingüísticos para varios pares de lenguas.
- ✓ Herramientas para gestionar los datos lingüísticos necesarios para construir un sistema de traducción automática para un determinado par de lenguas.

Los traductores que pueden desarrollarse con Apertium son sistemas de traducción automática por transferencia que procesan el texto en varias etapas: desformateo, análisis morfológico, desambiguación categorial, análisis sintáctico superficial, transferencia estructural superficial, transferencia léxica, generación morfológica y reformateo.

La principal ventaja de este motor de traducción automática con respecto a los ya mencionados anteriormente es, por un lado, que es un software de código abierto, y por otro, que no precisa de conexión a Internet para realizar las traducciones. Además se puede usar el sistema para construir sistemas de traducción entre cualquier par de lenguas, usando formatos fáciles de entender para quien programará el sistema.

Teniendo en cuenta las desventajas y limitaciones que presentan los motores antes estudiados, se concluye con la determinación del uso de Apertium como motor de traducción automática ya que, a diferencia del resto de los motores estudiados, Apertium se distribuye bajo la licencia GPL y no requiere

Fundamentación teórica

de conexión a Internet para llevar a cabo el proceso de traducción, premisas esenciales durante el desarrollo del módulo de traducción automática para el motor de búsqueda ORION.

1.6 Metodología de desarrollo

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. Indican, paso a paso, todas las actividades a realizar para lograr el producto informático deseado, así como las personas que deben participar en el desarrollo de las actividades y qué papel deben tener. Además se encargan de elaborar estrategias de desarrollo de software que promuevan prácticas centradas en las personas o los equipos, orientadas hacia la funcionalidad y la entrega, de comunicación intensiva y que requieren implicación directa del cliente.

Estas metodologías se dividen en dos grupos, las metodologías pesadas o robustas y las metodologías ligeras o ágiles. La diferencia entre estos dos tipos de metodologías es que mientras las robustas o pesadas intentan lograr el objetivo común por medio de la organización y la documentación, las ligeras o ágiles tratan de lograr la calidad del software por la comunicación inmediata y directa entre las personas que intervienen en el proceso. [24]

1.6.1 Open Unified Process (OpenUP)

Es una metodología ágil basada en Rational Unified Process (RUP), conservando las características esenciales de esta metodología, que incluye el desarrollo iterativo e incremental, casos de uso y escenarios de conducción de desarrollo, gestión de riesgos, y el enfoque centrado en la arquitectura. El resultado ha sido una metodología mucho más simple y ligera, definiéndose en el campo de proyectos pequeños que abarquen de 3 a 6 personas. [29]

Durante el proceso de desarrollo de software solo el contenido fundamental y necesario es incluido. La mayoría de los elementos de OpenUP están declarados para fomentar el intercambio de información entre los equipos de desarrollo y mantener un entendimiento compartido del proyecto, sus objetivos, alcance y avances.

Principios del OpenUP

- ✓ Colaborar para sincronizar intereses y compartir conocimiento.

Fundamentación teórica

Este principio promueve prácticas que impulsan un ambiente de equipo saludable, facilitan la colaboración y desarrollan un conocimiento compartido del proyecto.

- ✓ Equilibrar las prioridades para maximizar el beneficio obtenido por los interesados en el proyecto.

Este principio promueve prácticas que permiten a los participantes de los proyectos desarrollar una solución que maximice los beneficios obtenidos por los participantes y que cumple con los requisitos y restricciones del proyecto.

- ✓ Centrarse en la arquitectura de forma temprana para minimizar el riesgo y organizar el desarrollo.
- ✓ Desarrollo evolutivo para obtener retroalimentación y mejoramiento continuo.

Este principio promueve prácticas que permiten a los equipos de desarrollo obtener retroalimentación temprana y continua de los participantes del proyecto, permitiendo demostrarles incrementos progresivos en la funcionalidad.

OpenUP estructura el ciclo de vida de un proyecto en cuatro fases: concepción, elaboración, construcción y transición. El ciclo de vida del proyecto provee a los interesados un mecanismo de supervisión y dirección para controlar los fundamentos del proyecto, su ámbito, la exposición a los riesgos, el aumento de valor y otros aspectos.

Fases del OpenUP [30]

➤ **Concepción**

En esta fase se trabaja en el entendimiento del propósito y los objetivos, además de obtener suficiente información sobre el proyecto en cuestión.

➤ **Elaboración**

Durante esta fase se tratan los riesgos significativos para la arquitectura. Su propósito es establecer la base la elaboración de la arquitectura del sistema.

➤ **Construcción**

Esta fase está enfocada al diseño, implementación y prueba de las funcionalidades para desarrollar un sistema completo. El propósito de esta fase es completar el desarrollo del sistema basado en la arquitectura definida.

➤ **Transición**

Es la última fase, cuyo propósito es asegurar que el sistema es entregado a los usuarios, y evalúa la funcionalidad y rendimiento del último entregable de la fase de construcción.

Fundamentación teórica

Beneficios en el uso del OpenUP

- ✓ Disminuye las probabilidades de fracaso en los proyectos pequeños, incrementando las probabilidades de éxito.
- ✓ Permite la detección de errores tempranos por medio de un ciclo iterativo.
- ✓ Evita la elaboración de documentación, diagramas e iteraciones redundantes requeridos en la metodología RUP.
- ✓ Cuenta con un enfoque centrado al cliente y con iteraciones cortas.

Se ha decidido utilizar la metodología OpenUP para su aplicación en el proceso de desarrollo de software, debido a que el grupo de desarrollo es pequeño y el tiempo de construcción del módulo a desarrollar es poco. Además se tuvo en cuenta que OpenUP mantiene las principales características de RUP, obviando solamente las partes opcionales que RUP propone.

1.7 Lenguaje de modelado

El modelado constituye un mecanismo para facilitar la comprensión de las soluciones informáticas que cada vez tienen un nivel de complejidad más elevado. Los modelos permiten trabajar con sistemas mayores y más complejos, facilitando el proceso de codificación e implementación del sistema de forma distribuida y en distintas plataformas. [31]

1.7.1 Lenguaje Unificado de Modelado

Grady Booch, James Rumbaugh e Ivar Jacobson, empezaron a colaborar para combinar y recopilar las mejores características de cada uno de sus métodos de diseño y análisis orientado a objetos en un método unificado. El resultado, según Pressman, denominado **Lenguaje Unificado de Modelado (UML**, por sus siglas en inglés, Unified Modeling Language), se ha convertido en el método más utilizado por la industria.

Es un lenguaje gráfico para visualizar, construir y documentar los artefactos de un sistema de software Orientado a Objetos (OO) incluyendo su estructura y diseño. El objetivo final es una aplicación de software robusta, flexible y escalable. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como RUP), pero no especifica en sí mismo qué metodología o proceso usar. [33]

Fundamentación teórica

Permite la creación de los diferentes modelos que ofrecen las vistas necesarias para la construcción de un software de calidad y permite la comprensión del sistema que se quiere realizar tanto por parte de los usuarios finales, como de los desarrolladores que implementarán la solución. [34]

Dentro de sus características se pueden mencionar algunas tales como:

- ✓ Tecnología orientada a objetos.
- ✓ Viabilidad en la corrección de errores.
- ✓ Desarrollo iterativo e incremental.
- ✓ Modela estructuras complejas.
- ✓ Participación del cliente en todas las etapas del proyecto.
- ✓ Permite especificar todas las decisiones de análisis, diseño e implementación, construyéndose así modelos precisos, no ambiguos y completos.
- ✓ Puede conectarse con lenguajes de programación (Ingeniería directa e inversa).
- ✓ Modela el comportamiento del sistema mediante casos de uso, diagramas de secuencia y de colaboración, que sirven para evaluar el estado del mismo.
- ✓ Permite documentar todos los artefactos de un proceso de desarrollo.

Se seleccionó UML como Lenguaje de Modelado pues soporta la metodología de desarrollo OpenUP, además es orientado a sistemas y no a procesos, está consolidado como lenguaje estándar de análisis y diseño y es fácil de utilizar. Dentro de sus ventajas figura que la trazabilidad y la documentación del proyecto se realiza de una forma ordenada y guiada por casos de uso.

1.8 Herramienta CASE a utilizar

Las herramientas CASE (Ingeniería de Software Asistida por Computadoras), son aquellas aplicaciones que brindan ayuda al desarrollo de software, disminuyendo su costo en términos de tiempo y de dinero. Garantizan una correcta documentación, generación de código y reutilización del software.

1.8.1 Visual Paradigm for UML 6.4

Visual Paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Además es una herramienta de diseño multiplataforma. Soporta los principales estándares de la industria, tales como; UML, SysML, BPMN, XMI. Ofrece un conjunto completo de herramientas que los equipos de

Fundamentación teórica

desarrollo necesitan para la captura de requisitos, planificación de software, planificación de pruebas, modelamiento de clases, modelamiento de datos y otras actividades.

Con la utilización de la misma se obtienen una serie de ventajas como:

- ✓ Permite la sincronización entre el código fuente y el modelo en tiempo real.
- ✓ Cuenta con una abundante documentación, como son: tutoriales, demostraciones interactivas y proyectos UML.
- ✓ Es una herramienta colaborativa, es decir, permite múltiples usuarios trabajando sobre el mismo proyecto.
- ✓ Permite control de versiones.
- ✓ Proporciona a los desarrolladores una plataforma que les permite diseñar un producto rápidamente y con la calidad requerida.
- ✓ Facilita la interoperabilidad con otras herramientas CASE y se integra con múltiples herramientas de desarrollo, como Eclipse, Jbuilder, NetBeans IDE, Oracle Jdeveloper.
- ✓ Genera código y realiza ingeniería inversa para varios lenguajes de programación, Java, C++, CORBA IDL, PHP, XML Schema y ADA.
- ✓ Es una poderosa herramienta de generación de PDF/HTML a partir de diagramas UML.

Se decide utilizar Visual Paradigm como herramienta CASE debido a que el equipo de desarrollo cuenta con experiencia suficiente en el trabajo con la misma, lo que evita la necesidad de emplear tiempo en el estudio de otra herramienta de este tipo.

1.9 Lenguajes de programación

Actualmente existe una gran variedad de lenguajes de programación para el desarrollo web tales como: Perl, Java, PHP, Ruby y Python, los cuales permiten interactuar con los usuarios y utilizar sistemas de bases de datos.

Para el desarrollo del módulo de traducción automática para el buscador ORION se decidió utilizar los lenguajes del lado del cliente HTML, CSS y JavaScript y como lenguaje del lado del servidor PHP. A continuación se describen las características de cada uno de ellos.

Fundamentación teórica

1.9.1 Lenguaje de Marcado de Hipertexto HTML

El Lenguaje de Marcado de Hipertexto (HTML, por sus siglas en inglés, Hypertext Markup Language), fue creado en el año 1990 por Tim Berners-Lee y diseñado principalmente para mostrar información, animaciones en forma de hipertexto, es un lenguaje universal que todos los navegadores traen embebido por su gran sencillez y puede ser abierto y modificado con cualquier editor de texto. Algunas ventajas que presenta, son la facilidad con que se pueden actualizar los contenidos y, además, permite la creación de estilos en formato CCS (hojas de estilos en cascada) a las páginas para una mejor visibilidad. [37]

1.9.2 CSS

Una hoja de estilo CSS (Cascading Style Sheets) es un lenguaje creado por el W3C que permite a los desarrolladores controlar los aspectos de estilo y formato de múltiples páginas HTML simultáneamente, permitiendo separar el contenido de presentación en todo momento.

El empleo de hojas de estilo CSS aporta disímiles beneficios como:

- ✓ Aumento de la legibilidad y reducción del peso de las páginas Web.
- ✓ Mejora del mantenimiento y actualización de los sitios Web.
- ✓ Mejora de la accesibilidad: se pueden definir hojas de estilo locales en función de las necesidades o preferencias del usuario.
- ✓ Versatilidad: se ofrecen diferentes hojas de estilo para los diferentes tipos de medio existentes (sintetizadores de voz, dispositivos braille, pantallas de computador a color, televisión).

1.9.3 JavaScript

JavaScript es un lenguaje de scripts desarrollado por Netscape para incrementar las funcionalidades del lenguaje HTML. Dentro de sus características más importantes se pueden encontrar que es un lenguaje interpretado, es decir, no requiere compilación pues el navegador se encarga de interpretar las sentencias JavaScript contenidas en una página HTML y ejecutarlas adecuadamente; es un lenguaje orientado a eventos, o sea, que se pueden desarrollar scripts que ejecuten acciones en respuesta a eventos y que es un lenguaje orientado a objetos. Su modelo de objetos está reducido y simplificado, pero incluye los elementos necesarios para que los scripts puedan acceder a la información de una página y puedan actuar sobre la interfaz del navegador. [38]

Fundamentación teórica

Entre las acciones típicas que se pueden realizar en JavaScript se tienen dos vertientes. Por un lado los efectos especiales sobre páginas web, para crear contenidos dinámicos y elementos de la página que tengan movimiento, cambien de color o cualquier otro dinamismo. Por el otro, JavaScript nos permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que podemos crear páginas interactivas con programas como calculadoras, agendas, o tablas de cálculo.

1.9.4 PHP

PHP es un lenguaje de código abierto, interpretado en el lado del servidor, utilizado para la generación de páginas web dinámicas y que puede ser embebido en páginas HTML.

Entre sus principales ventajas se encuentran: [39]

- ✓ Velocidad: está escrito en C, por lo que se ejecuta rápidamente utilizando poca memoria.
- ✓ Estabilidad: utiliza su propio sistema de administración de recursos y dispone de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.
- ✓ Simplicidad: los usuarios con experiencia en C y C++ podrán utilizar PHP rápidamente, debido a que la sintaxis es similar.
- ✓ Puede interactuar con muchos motores de bases de datos relacionales tales como MySQL, MS SQL, Oracle, Informix, PostgreSQL y además bases de datos no relacionales como MongoDB.
- ✓ Lenguaje de código abierto, lo que posibilita la actualización y corrección de problemas por la amplia comunidad con que cuenta.
- ✓ Multiplataforma: permite ser ejecutado bajo varias versiones de Unix, Windows y Mac.

La principal ventaja que hace de PHP el lenguaje de programación idóneo para el desarrollo del módulo de traducción automática, es la posibilidad de programar orientado a objetos, lo cual proporciona mucha claridad al desarrollo. Además es uno de los lenguajes de programación web que más documentación posee, dispone de un sitio oficial que brinda toda la documentación que los desarrolladores necesiten, y cuenta con una gran comunidad de desarrollo. [40]

1.10 Marco de trabajo

La utilización de un marco de trabajo (framework, por su traducción al inglés) simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas

Fundamentación teórica

comunes. Además, proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. [41]

Teniendo en cuenta la integración del módulo que se desea desarrollar con el motor de búsqueda ORION, se decidió que en el desarrollo del mismo se utilizará como marco de trabajo para el apoyo a la implementación, Symfony, en su versión 2.0, además de que es el marco de trabajo para el lenguaje PHP que se encuentra aprobado por el centro CIDI al cual pertenece el presente trabajo de diploma.

1.10.1 Symfony

Para el desarrollo del módulo de traducción automática se emplea Symfony, un marco de trabajo para PHP con gran aceptación entre los desarrolladores de aplicaciones web. Esta tecnología hace un uso bastante eficiente de varios patrones de diseño. Elegir Symfony (versión 2) para el desarrollo de la solución facilitará el proceso de integración del módulo a desarrollar con la Interfaz del Traductor, además permitirá mantener cierta homogeneidad con los demás componentes del motor de búsqueda ORION. A continuación se relacionan algunas características de Symfony que refuerzan su elección:

- ✓ Fácil de instalar y configurar en la mayoría de las plataformas.
- ✓ Independiente del sistema gestor de bases de datos.
- ✓ Sencillo de usar en la mayoría de los casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
- ✓ Sigue la mayoría de las mejores prácticas y patrones de diseño para la web.
- ✓ Preparado para aplicaciones empresariales y adaptables a las políticas y arquitecturas propias de cada empresa, además es lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- ✓ Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros.

1.11 Entorno de Desarrollo Integrado (IDE)

Un Entorno de Desarrollo Integrado (IDE por sus siglas en inglés Integrated Development Environment) es un entorno de programación que ha sido empaquetado como un programa de aplicación; es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Además proveen un área de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, PHP, Python, Java, C#, Delphi, Visual Basic, etc.

Fundamentación teórica

Existen múltiples IDEs entre los que se encuentran Visual Studio, Eclipse, NetBeans, seleccionándose este último para el desarrollo de la solución ya que es un IDE que soporta varios lenguajes como Java, PHP, C++, entre otros, y que facilita la creación de interfaz gráfica de usuario para aplicaciones web y de escritorio, lo que permite ahorrar tiempo de desarrollo. Por otra parte se posee gran dominio y experiencia en el uso de este IDE para el desarrollo de aplicaciones web en PHP.

1.11.1 NetBeans

El IDE NetBeans es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Trae soporte para crear aplicaciones en otros lenguajes además de Java, como son: PHP, C++, JavaScript y Python. Es un producto libre y gratuito sin restricciones de uso. Se encuentran disponibles versiones de NetBeans para los sistemas operativos Windows, Linux, Mac OS y Solaris. [42]

Se recomienda utilizar la versión 7.3 del IDE. La base en la que se sustenta su elección es que permite desarrollar aplicaciones utilizando el marco de trabajo Symfony y ejecutar los comandos de este directamente desde la interfaz del IDE.

1.12 Servidor Web

Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente. Estos generan una respuesta en cualquier lenguaje entendible por el cliente. Para la transmisión de todos estos datos suele utilizarse generalmente el protocolo HTTP, perteneciente a la capa de aplicación del modelo OSI. [43]

1.12.1 Apache 2

Apache es el servidor web hecho por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. La licencia Apache es descendiente de la licencias BSD. Esta licencia te permite hacer lo que quieras con el código fuente. [43]

Razones para usar Apache:

- ✓ Corre en una multitud de sistemas operativos, lo que lo hace prácticamente universal.
- ✓ Es una tecnología gratuita de código fuente abierta.

Fundamentación teórica

- ✓ Es un servidor altamente configurable de diseño modular: Es muy sencillo ampliar las capacidades del servidor Web Apache. Actualmente existen muchos módulos para Apache que son adaptables a este.
- ✓ Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor: Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.
- ✓ Tiene una alta configurabilidad en la creación y gestión de logs: Apache permite la creación de ficheros de log a medida del administrador, de este modo se puede tener un mayor control sobre lo que sucede en el servidor.

Por las características anteriormente expuestas se selecciona Apache 2 como servidor web, pues es un servidor de código abierto para plataformas Unix (BSD, GNU/Linux) y viene en la instalación por defecto de muchos de estos sistemas operativos.

1.13 Sistema Gestor de Base de Datos (SGBD)

Los sistemas de gestión de bases de datos son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se define también como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Ayuda a realizar la definición, el control de la seguridad y la privacidad, la manipulación y el mantenimiento de la integridad de los datos dentro de la base de datos. [44]

Los almacenes de datos pueden ser relacionales y no relacionales, a estos últimos también se les conoce comúnmente como bases de datos No SQL. [45]

Las bases de datos relacionales cumplen con el modelo relacional, por lo que permiten establecer conexiones entre los datos que están guardados en tablas y a través de dichas conexiones relacionar los datos de ambas tablas. Dentro de los sistemas de gestión de bases de datos relacionales usados a nivel internacional se encuentran: PostgreSQL, MySQL, ORACLE y SQL-Server.

Por otro lado, los almacenes de datos no relacionales normalmente no tienen esquemas fijos de tablas. Estos sistemas responden a las necesidades de escalabilidad horizontal que tienen cada vez más empresas. Actualmente los que más sobresalen son: MongoDB, Cassandra y BigTable.

Fundamentación teórica

1.13.1 MongoDB

Un estudio enfocado en las base de datos no relacionales arroja que MongoDB constituye una solución escalable y de alto rendimiento de almacenes de datos No SQL. Es un sistema de código abierto y escrito en C++, orientado al almacenamiento de datos en documentos al estilo JSON con esquemas dinámicos, que ofrecen potencia y simplicidad. Se destaca por conservar los índices de todos los atributos y hacer mucho más flexible la agregación y procesamiento de datos. [47]

MongoDB se basa en colecciones, o sea, los datos se agrupan en conjuntos llamados "colecciones". Cada colección tiene un nombre único en la base de datos y puede contener un número ilimitado de documentos. Una colección es análoga a una tabla, excepto que no tienen un esquema definido.

Se decide utilizar MongoDB como SGBD por lograr un mejor manejo de la información frente a un volumen considerable de datos, pues se utilizará para el almacenamiento de una gran cantidad de criterios realizados por los usuarios, además de registrar la información generada al realizar traducciones. Permite una fácil integración con el lenguaje de programación seleccionado, además de que el marco de trabajo a utilizar, Symfony 2.0, posee soporte para MongoDB.

1.14 Conclusiones

En este capítulo se estudiaron las disímiles aplicaciones existentes a nivel internacional que ofrecen servicios de traducción automática, analizando características, ventajas y desventajas de cada una de ellas. Basado en dicho estudio se concluye que es necesario desarrollar un módulo de traducción automática que pueda ser integrado al buscador ORION, complementando así su funcionamiento. Dicho módulo de traducción será bilingüe, bidireccional, completamente automático y las traducciones se realizarán a través del motor de traducción Apertium cuya estrategia es la traducción automática por transferencia.

Para este proceso de desarrollo se utilizará como metodología OpenUP, recurriendo al Visual Paradigm como herramienta de modelado, con la cual se podrá realizar el análisis y diseño de la aplicación a través del lenguaje de modelado UML. Además utilizará el lenguaje de programación PHP, para la implementación del módulo en plataforma de desarrollo NetBeans.

CAPÍTULO 2. ANÁLISIS Y DISEÑO DEL SISTEMA

2.1 Introducción

Un flujo de trabajo es una relación de actividades que producen resultados observables. Dentro de los flujos de trabajo que propone OpenUP, se encuentra el de Requerimientos, en él se establece qué tiene que hacer exactamente el sistema que se construye, por lo que en el presente capítulo se realiza una descripción del módulo a desarrollar y se muestra el modelo de dominio para un mayor entendimiento del flujo de trabajo del mismo. Se capturan los requerimientos funcionales y no funcionales, lo que permite identificar los casos de uso y sus descripciones.

Por último se reflejan los artefactos modelados como resultado del flujo de Análisis y Diseño para la construcción de una aplicación web. Además de los diagramas de clases del diseño con estereotipos web y diagramas de secuencia, se tiene en cuenta, el modelo de despliegue, para tener una mayor visión de los elementos que interactúan en el funcionamiento de la aplicación.

2.2 Propuesta de sistema

Para dar solución a los objetivos expuestos se propone el desarrollo de un módulo de traducción automática, utilizando el marco de trabajo Symfony 2.0 para el trabajo con interfaces visuales y el motor de traducción automática Apertium para generar las traducciones, capaz de integrarse al motor de búsqueda ORION.

La aplicación estará compuesta por dos interfaces:

- ✓ Traducción de contenidos: Mediante esta interfaz el usuario puede traducir textos planos. Además le permite introducir una dirección URL para traducir páginas web si lo desea.
- ✓ Traducción de páginas web: Se muestra una vez que el usuario seleccione la opción “Traducir” en una de las páginas web indexadas por el motor de búsqueda ORION. También se muestra al introducir una dirección URL en la interfaz Traducción de contenidos. Además, el usuario puede insertar una dirección URL en esta interfaz si desea traducir otra página web, diferente a la que se le ha mostrado.

Una vez que el usuario inserte la información que necesita traducir, la aplicación debe interactuar con el motor de traducción automática Apertium enviándole los datos necesarios para realizar este proceso.

Análisis y diseño del sistema

Apertium es el responsable de generar el texto o la página web traducida según el idioma de salida seleccionado, proceso que culminará una vez que el sistema muestre la traducción al usuario. Al mostrarse la traducción, el usuario podrá emitir su opinión respecto a la calidad de la misma, información que persistirá en la base de datos de la aplicación.

La aplicación también registrará un conjunto de datos generados durante la traducción como son: fecha, dirección IP del usuario que está traduciendo, así como el tipo de navegador que ha utilizado.

Debido a que el proceso de sincronización de un gran número de usuarios al traducir páginas o texto son tareas que requieren un tiempo considerable de procesamiento, se considera necesaria la implementación de dichos mecanismos en forma de tareas de Symfony 2. Las tareas de Symfony 2 permiten ejecutar cierto código de forma independiente a la interfaz administrativa de un sistema, o sea, desde la línea de comandos. Con esto se logra evitar que la aplicación supere el tiempo límite de los navegadores en espera de una respuesta del servidor.

Al utilizar Symfony 2 y PHP en el desarrollo del módulo, se facilita su integración al motor de búsqueda ORION.

2.3 Modelado del Dominio

El modelo de dominio también conocido como Modelo Conceptual es una representación visual de los conceptos u objetos del mundo real, significativos para un problema o área de interés. [48]

Teniendo en cuenta que la definición de procesos y roles del negocio se hace difícil encontrarlos, se hace necesario describir el funcionamiento de la aplicación mediante una serie de conceptos, entidades y sus relaciones, agrupándose en un modelo de dominio con el fin de contribuir a la comprensión del contexto del sistema.

2.3.1 Descripción de Clases del Modelo del Dominio

Un modelo de dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos de dominio representan los principales conceptos. La modelación del dominio tiene como objetivo fundamental la comprensión y descripción de las clases más importantes en el sistema.

Análisis y diseño del sistema

Concepto	Descripción
Usuario	Usuarios comunes que necesitan realizar una traducción automática.
Aplicación web	Es la aplicación web a través de la cual el usuario puede acceder a la interfaz de un traductor automático.
Interfaz Traductor	Se refiere a las interfaces que se muestran una vez que el usuario seleccione las opciones que brinda la aplicación para realizar traducciones automáticas.
Motor de traducción automática	Sistema externo que se encarga de traducir los contenidos solicitados por los usuarios.
Traducción	Se refiere a los contenidos web traducidos que genera el motor de traducción automática.
Base de datos	Se refiere a la base de datos que almacena toda la información referente a la traducción.

Tabla 1: Conceptos asociados al dominio.

2.3.2 Diagrama de Clases del Modelo del Dominio.

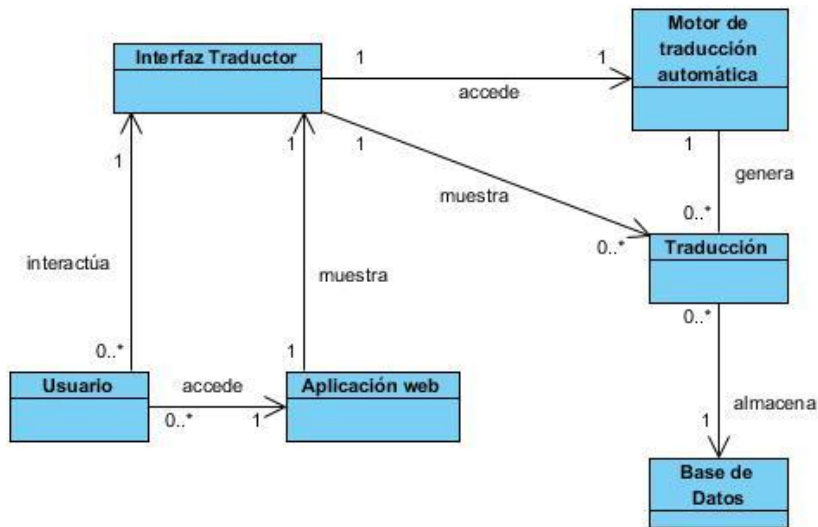


Figura 2.1: Modelo de dominio

El modelo de dominio representado anteriormente muestra cómo una aplicación web que brinda servicios de traducción automática, realiza dicho proceso. El funcionamiento comienza cuando un usuario que

Análisis y diseño del sistema

accede a la aplicación desea realizar traducciones de contenidos web. El sistema muestra la interfaz correspondiente, con la que los usuarios interactúan para llevar a cabo dichas traducciones. Para completar el proceso, la aplicación accede a un motor de traducción automática, sistema que genera las traducciones en cuestión, finalmente mostradas en las interfaces de la aplicación. Por último se registra la información generada durante el proceso en la base de datos de la aplicación web.

2.4 Especificación de los Requisitos del Software.

Hacer una identificación exhaustiva y correcta de los requisitos fundamentales que debe cumplir el sistema es muy importante. Un proyecto no puede ser exitoso sin una especificación correcta y exhaustiva de los requerimientos [49].

Una vez que se han definido los conceptos principales relacionados con el objeto de estudio y el dominio, se puede comenzar a analizar qué debe hacer la aplicación para que cumpla con los objetivos planteados al inicio de este trabajo. Para ello, se enumeran a través de requerimientos funcionales y no funcionales, las acciones que el sistema deberá ser capaz de realizar.

2.4.1 Requisitos funcionales

El propósito fundamental de la captura de los requisitos es guiar el desarrollo hacia el sistema correcto. Esto se consigue mediante una descripción de los requisitos del sistema suficientemente buena como para que pueda llegarse a un acuerdo entre el cliente (incluyendo a los usuarios) y los desarrolladores sobre qué debe y qué no debe hacer el sistema [50].

Requisitos funcionales del módulo de traducción automática:

- RF 1 Gestionar la traducción de texto de un idioma origen a un idioma destino.
 - RF 1.1 Permitir la introducción del texto a traducir.
 - RF 1.2 Permitir la selección del idioma original.
 - RF 1.3 Validar que se encuentren los datos necesarios para realizar la traducción.
 - RF 1.4 Mostrar texto traducido.
- RF 2 Permitir la introducción de la dirección URL de la página web a traducir.
 - RF 2.1 Permitir la selección del idioma de origen.
 - RF 2.2 Validar que la dirección URL sea correcta.
- RF 3 Permitir la traducción de páginas web indexadas por el motor de búsqueda ORION.

Análisis y diseño del sistema

RF 3.1 Identificar el idioma en el que se encuentra la página web a traducir.

RF 4 Gestionar la traducción de páginas web, conservando el diseño original.

RF 4.1 Descargar página web a traducir.

RF 4.2 Obtener, del motor de búsqueda ORION, el idioma en el que se encuentra la página web a traducir.

RF 4.3 Actualizar las direcciones de las imágenes, enlaces, css y scripts.

RF 4.4 Mostrar la página web traducida.

RF 4.5 Eliminar los ficheros generados durante el proceso de traducción.

RF 5 Consumir servicio de traducción automática del motor Apertium.

RF 6 Registrar la información de la traducción realizada.

RF 7 Registrar elección del criterio de calidad de la traducción.

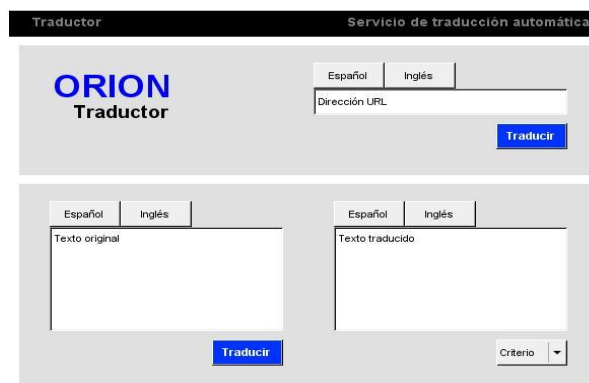
RF 8 Registrar motivos de la elección del criterio de calidad de la traducción.

2.4.2 Especificación de requisitos funcionales

A continuación se realiza una descripción de los principales requisitos de la aplicación, el resto de las descripciones se encuentran en el **anexo # 2**.

Nombre	Descripción	Complejidad	Prioridad para cliente
Gestionar la traducción de texto de un idioma origen a un idioma destino.	Realizar el proceso de traducción automática de textos introducidos por el usuario, entre los idiomas español e inglés.	Media	Alta

Prototipo



Análisis y diseño del sistema

Campos	Tipos de Datos	Reglas o Restricciones
txt_textoOrigen	Text	Insertar obligatoriamente un texto.
bt_espanolOrigen	String	No procede
bt_inglesOrigen	String	No procede
txt_textoSalida	Text	No procede

Tabla 2: Especificación del RF “Gestionar la traducción de texto de un idioma origen a un idioma destino”


Nombre	Descripción	Complejidad	Prioridad para cliente
Permitir la introducción de la dirección URL de la página web a traducir	Los usuarios pueden introducir la dirección URL de una página web que deseen traducir.	Media	Alta
Prototipo			
			
Campos	Tipos de Datos	Reglas o Restricciones	
txt_url	Text	Insertar obligatoriamente la dirección URL a traducir.	
bt_inglesUrl	String	No procede	
bt_espannoUrl	String	No procede	
Observaciones	La dirección URL puede ser introducida desde la interfaz de traducción de textos o desde la interfaz que se muestra al traducir una página web.		

Tabla 3: Especificación del RF “Permitir la introducción de la dirección URL de la página web a traducir”

2.4.3 Requisitos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto usable, rápido o confiable. Los requerimientos no funcionales aunque no aportan funcionalidades propiamente dichas dentro de la aplicación, son de vital importancia para una puesta en marcha exitosa del software, y para lograr que este responda a las expectativas del usuario.

- **Requerimientos de software:**
 - ✓ Se requiere como Sistema Operativo, GNU/Linux.
- **Requerimientos de hardware:**
 - ✓ Procesador Pentium IV.
 - ✓ 2 Gigabytes (GB) de memoria RAM (Random Access Memory, por sus siglas en inglés) como mínimo.
 - ✓ 10 GB de espacio en disco duro.
- **Restricciones de Diseño e implementación:**
 - ✓ Se utilizará el marco de trabajo Symfony en su versión 2.0 para el desarrollo de la aplicación.
 - ✓ Empleo del paradigma de Programación Orientada a Objetos (POO).
- **Requerimientos de apariencia o interfaz externa:**
 - ✓ Debe poseer una interfaz amigable al usuario, basada en paneles con formularios, con una navegabilidad intuitiva para el usuario.
 - ✓ Para la realización del diseño se tendrá en cuenta la identidad del motor de búsqueda ORION, por lo cual se utilizarán los colores azul, blanco y gris.
- **Requerimientos de Seguridad:**
 - ✓ Confidencialidad: La información manejada por el sistema estará protegida de acceso no autorizado y divulgación.
 - ✓ Integridad: La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma será considerada igual a la fuente o autoridad de los datos.
 - ✓ Disponibilidad: Los dispositivos o mecanismos utilizados para lograr la seguridad no ocultarán o retrasarán a los usuarios para obtener las traducciones deseadas en un momento dado.

Análisis y diseño del sistema

- **Requerimientos de Usabilidad:** La aplicación podrá ser utilizada por usuarios con conocimientos mínimos de computación.
- **Requerimientos de licencia:** Se utilizarán herramientas de software libre, de las cuales se posee licencia para su uso y distribución.

2.5 Modelo de Casos de Uso del Sistema

El modelo de casos de uso es un modelo del sistema que contiene actores, casos de uso y sus relaciones. El actor es una entidad externa del sistema que de alguna manera participa en la historia del caso de uso. Por lo regular estimula el sistema con eventos de entrada o recibe algo de él.

2.5.1 Diagrama de Casos de Uso del Sistema

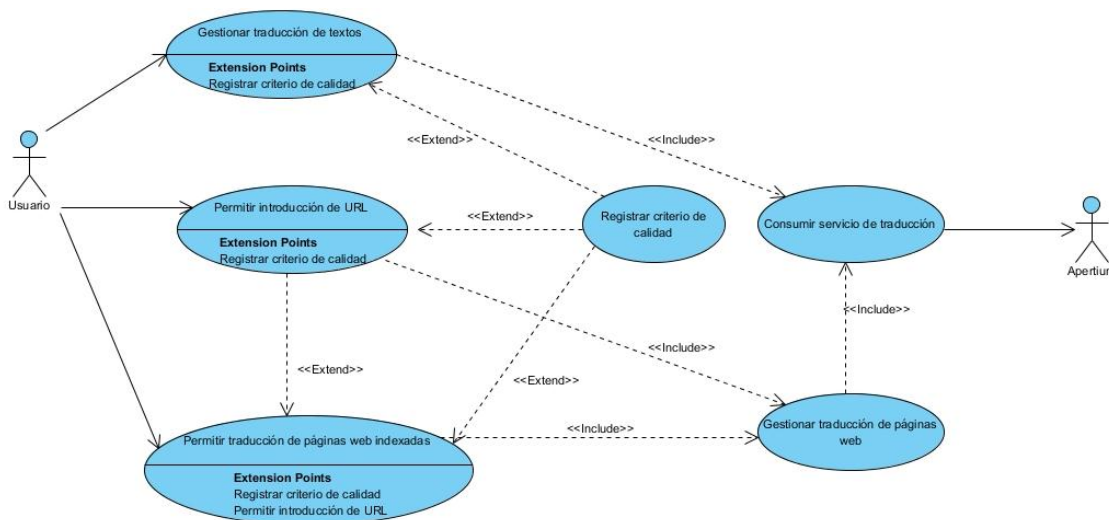


Figura 2.2: Diagrama de casos de uso del sistema

Muestra los seis casos de uso por los que está compuesto el sistema, dentro de los cuales existen tres casos de uso críticos, cuya importancia está determinada a partir de la prioridad que poseen para el módulo de traducción automática tal como se muestra en el **anexo # 1**:

- ✓ El caso de uso “Gestionar traducción de textos” engloba todo el proceso referido a la traducción automática de texto plano.
- ✓ El caso de uso “Permitir introducción de URL” se refiere al proceso mediante el cual el usuario introduce la dirección URL de una página web que desea traducir.

Análisis y diseño del sistema

- ✓ “Permitir traducción de páginas web indexadas”, caso de uso que abarca el proceso desde que el usuario selecciona una de las páginas web indexadas por el buscador después de realizada una búsqueda, hasta que se le muestra dicha página traducida.

Con el propósito de lograr una mayor reutilización, existen varios casos de uso incluidos, ejemplo de ellos son:

- ✓ “Gestionar traducción de páginas web”, incluido en los dos últimos casos de uso mencionados anteriormente.
- ✓ “Consumir servicio de traducción”, el cual comprende el proceso a partir de que la aplicación interactúa con el actor Apertium. Además, dicho caso de uso se encarga de registrar la información generada durante la traducción.

También se muestra el caso de uso extendido “Registrar criterio de calidad”, que abarca el proceso de selección de un criterio de calidad, así como la justificación de dicha selección.

2.5.2 Patrones de casos de uso utilizados

Los patrones de casos de uso son comportamientos que deben existir en el sistema describiendo el uso del mismo y cómo este interactúa con los usuarios. Estos patrones son utilizados generalmente como plantillas que describen como debería ser estructurados y organizados los casos de uso.

Entre los beneficios que trae consigo la aplicación de estos patrones se encuentran:

- ✓ Aumentar la productividad.
- ✓ Reutilizar elementos existentes.
- ✓ No invertir tiempo en resolver problemas ya resueltos.
- ✓ Aplicar la teoría al trabajo práctico.
- ✓ Habilitar las herramientas de soporte para modelar el desarrollo.

Los patrones de casos de uso utilizados fueron:

Concordancia: Extrae una sub-secuencia de acciones que aparecen en diferentes lugares del flujo de casos de uso y es expresado por separado. Este tiene dos variantes:

- ✓ Reutilización: Se compone de tres casos de uso, el primero llamado “Sub-secuencia Común”, modela una secuencia de acciones que aparecerán en múltiples casos de uso en el modelo; los otros casos de uso modelan el uso del sistema que comparte la sub-secuencia común de acciones, de manera que deben existir al menos dos de ellos.

Análisis y diseño del sistema

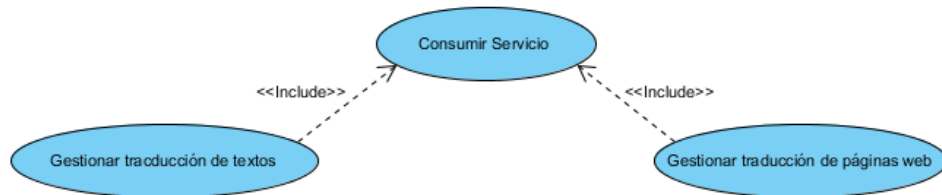


Figura 2.3: Patrón de caso de uso: Reutilización

- ✓ Adición: Este patrón es preferible usarlo cuando otros casos de uso se encuentran propiamente completos, o sea, que no requieren de una sub-secuencia común de acciones para modelar los usos completos del sistema.

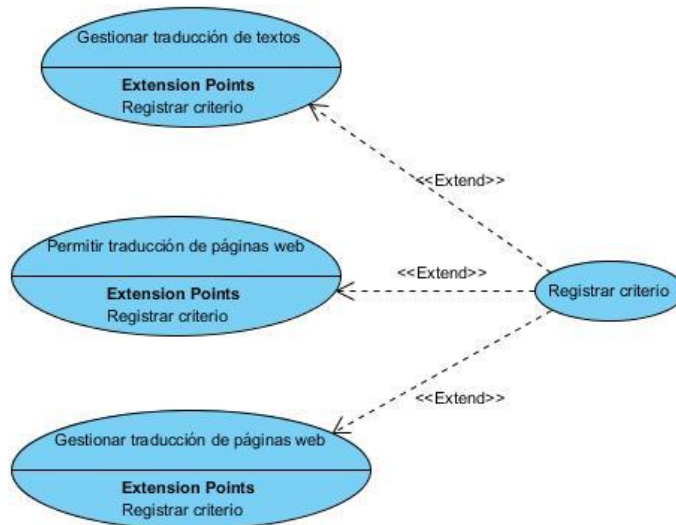


Figura 2.4: Patrón de caso de uso: Adición

2.5.3 Especificación de casos de uso

A continuación se realiza una descripción de los principales casos de usos del sistema, el resto de las descripciones se encuentran en el **anexo # 3**.

Tabla 4: Descripción del CUS “Gestionar traducción de textos”

Objetivo	Gestionar la traducción de textos introducidos por el usuario.
Actores	Usuario
Resumen	El caso de uso se inicia cuando el usuario introduce el texto que desea traducir y selecciona el

Análisis y diseño del sistema

	idioma en el que se encuentra. La aplicación traduce el texto a través del motor de traducción Apertium y lo muestra al usuario.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario ha pulsado la opción "Traductor" en el motor de búsqueda ORION.	
Postcondiciones	Se mostró el texto traducido al idioma de salida seleccionado por el usuario.	
Flujo de eventos		
Flujo básico Gestionar traducción de textos		
	Actor	Sistema
1.	Introduce el texto a traducir en el campo "Texto a traducir", selecciona el idioma del texto origen y pulsa el botón "Traducir".	
2.		Valida que se haya introducido el texto que se desea traducir.
3.		En caso de estar correcto, <u>Ver CU: "Consumir servicio de traducción".</u>
4.		Muestra el texto traducido.
5.		Elimina los ficheros locales generados durante la traducción.
6.		Termina el CU.
Flujos alternos		
1ª El usuario no introdujo el texto que desea traducir.		
	Actor	Sistema
2.1		Muestra el mensaje "Debe introducir el texto que desea traducir".
		Vuelve a la acción 1.
2º. El usuario no selecciona el idioma del texto original.		
	Actor	Sistema
2.2		Traduce el texto, por defecto, al idioma inglés.
Relaciones	CU Incluidos	<u>Ver CU: Consumir servicio de</u>

Análisis y diseño del sistema

		<u>traducción</u>
	CU Extendidos	
Requisitos funcionales	no	
Asuntos pendientes		

Tabla 5: Descripción del CUS “Permitir introducción de dirección URL”

Objetivo	Permitir la traducción de páginas web, a partir de una dirección URL introducida por el usuario.	
Actores	Usuario	
Resumen	El caso de uso se inicia cuando un usuario desea traducir una página web indicando su dirección URL, la cual es utilizada por la aplicación para gestionar posteriormente todo el proceso de traducción hasta mostrar la página web traducida.	
Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario ha seleccionado la opción “Traductor” en la interfaz del motor de búsqueda ORION. El usuario ha seleccionado la opción “Traducir” en una de las páginas web indexadas por el buscador y se ha mostrado la página web traducida.	
Postcondiciones	Se mostró la página web traducida al usuario.	
Flujo de eventos		
Flujo básico Permitir introducción de dirección URL		
	Actor	Sistema
1.	Introduce la dirección URL que desea traducir en el campo “Introducir dirección URL”, selecciona el idioma en el que se encuentra y pulsa el botón “Traducir”.	
2.		Valida que se haya introducido la dirección URL que se desea traducir.
3.		En caso de estar correcta, <u>Ver CU: “Gestionar traducción de páginas web”.</u>
4.		Termina el CU.
Flujos alternos		
1 ^{ra} . El usuario no introdujo la dirección URL que desea traducir.		

Análisis y diseño del sistema

	Actor	Sistema
2.1		Muestra el mensaje "Debe introducir la dirección URL que desea traducir"
		Vuelve a la acción 1.
2º. El usuario no selecciona el idioma en el que se encuentra la página web a traducir.		
	Actor	Sistema
		Traduce la página web, por defecto, al idioma inglés.
3º. La dirección URL introducida por el usuario no se encuentra.		
	Actor	Sistema
3.1		Muestra el mensaje "Página no encontrada".
		Vuelve a la acción 1.
Relaciones	CU Incluidos	<u>Ver CU: "Gestionar traducción de páginas web".</u>
	CU Extendidos	
Requisitos no funcionales		
Asuntos pendientes		

2.6 Estilo arquitectónico

Los estilos expresan la arquitectura en el sentido más formal y teórico, describen entonces una clase de arquitectura, o piezas identificables de las arquitecturas empíricamente dadas. Esas piezas se encuentran repetidamente en la práctica, trasuntando la existencia de decisiones estructurales coherentes. Una vez que se han identificado los estilos, es lógico y natural pensar en reutilizarlos en situaciones semejantes que se presenten en el futuro. [51]

Para el desarrollo de la aplicación se decide emplear el patrón arquitectónico Modelo - Vista - Controlador (MVC) ya que el marco de trabajo Symfony 2 brinda una estructura y un funcionamiento basado en la misma.

El patrón MVC se encuentra dentro de la clasificación de los estilos de llamada y retorno y tiene como objetivo separar la lógica de negocios de la interfaz gráfica de manera que cambios en la misma no afecten la lógica de negocios y viceversa. Está compuesto por tres componentes, el modelo que

Análisis y diseño del sistema

representa la información que tanto el usuario como la aplicación puede manipular, la vista que implica todos los elementos que componen la interfaz gráfica y el controlador que maneja la interacción y la comunicación entre el modelo y las acciones del usuario.

Algunas ventajas de este patrón arquitectónico son: [52]

- ✓ Facilita la agregación de múltiples representaciones de los mismos datos.
- ✓ Crea independencia de funcionamiento.
- ✓ Facilita el mantenimiento de errores.
- ✓ Alta escalabilidad pues se puede manejar muchas peticiones con el mismo rendimiento simplemente añadiendo más hardware.

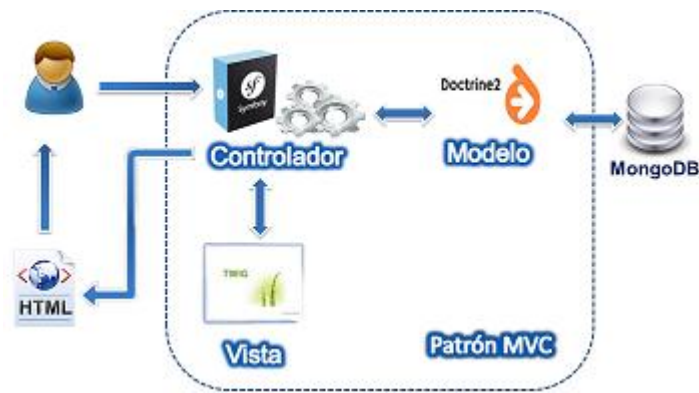


Figura 2.5: Patrón arquitectónico MVC

2.7 Patrones de diseño

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. Son parejas de problema/solución con un nombre, que codifican buenos principios y sugerencias relacionados generalmente con la asignación de responsabilidades. [53]

En el diseño de la herramienta se tuvieron en cuenta los patrones GRASP (Patrones Generales de Software para Asignación de Responsabilidades), los cuales describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones.

De los patrones GRASP se utilizan los siguientes:

Experto: Este patrón plantea que se debe asignar una responsabilidad al experto en información, o sea, a la clase que cuenta con la información necesaria para cumplir la responsabilidad.

Análisis y diseño del sistema

Beneficios del patrón Experto:

- ✓ Se conserva el encapsulamiento, ya que los objetos se valen de su propia información para hacer lo que se les pide. Esto posibilita tener sistemas de fácil mantenimiento.
- ✓ El comportamiento se distribuye entre las clases que cuentan con la información requerida para cumplir con la responsabilidad asignada.

Una de las clases donde se evidencia la utilización del patrón Experto es la clase Acceso_Datos pues esta es la clase que contiene la información necesaria (atributos, constructor y métodos) para cumplir con sus responsabilidades (conectarse a la base de datos y ejecutar consultas).

Creador: Este patrón plantea que se debe asignar a una clase X la responsabilidad de crear una instancia de una clase Y. La creación de objetos es una de las actividades más frecuentes en un sistema orientado a objetos. Este patrón es el encargado de guiar la asignación de responsabilidades relacionadas con la creación de objetos.

Beneficios del patrón Creador: Se crean menos dependencias y existe mayor posibilidad de reutilización de código.

En la clase controladora Traductor se evidencia la utilización del patrón Creador pues esta crea un objeto de la clase Simple_HTML_DOM con el objetivo de poder acceder a sus métodos.

Bajo acoplamiento: Este patrón plantea que se debe asignar las responsabilidades de forma tal que las clases se comuniquen con el menor número de clases que sea posible. Una clase con bajo acoplamiento no depende de muchas otras.

Beneficios del patrón Bajo acoplamiento

- ✓ Fáciles de entender por separado.
- ✓ Fáciles de reutilizar.

Alta cohesión: Este patrón plantea que se debe asignar una responsabilidad de modo que la cohesión siga siendo alta (una clase tiene responsabilidades moderadas). Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas, que no realicen un trabajo enorme. Una clase con baja cohesión hace muchas cosas no afines o un trabajo excesivo.

Beneficios del patrón Alta cohesión

- ✓ Mejoran la claridad y la facilidad con que se entiende el diseño.
- ✓ Se simplifican el mantenimiento y las mejoras en funcionalidad.

2.8 Modelo de Diseño

El modelo de diseño es un modelo de objetos que describe la realización de los casos de uso, y sirve como una abstracción del modelo de implementación y el código fuente. Su objetivo fundamental es adquirir una comprensión en profundidad de los aspectos relacionados con los requerimientos no funcionales y restricciones relacionados con los lenguajes de programación.

2.8.1 Diagrama de clases del diseño

A continuación se muestran los diagramas de clases del diseño de los principales casos de uso del sistema en el cual se expresan las relaciones entre las clases, donde la única diferencia con los diagramas de clases tradicionales, es que cuentan con un grupo de estereotipos de UML adicionales que son específicos para los diagramas de clases web, que permiten modelar aplicaciones con esta arquitectura. El resto de los diagramas de clases del diseño se encuentra en el **anexo # 4**.

El siguiente diagrama muestra la relación existente entre las clases que componen el caso de uso "Gestionar traducción de textos". El mismo está compuesto por una clase cliente CP_Principal que envía una petición a la clase controladora Traductor. Este componente representa un conjunto de servicios que ofrece el marco de trabajo Symfony 2, mediante los cuales se procesa la petición del usuario. Teniendo en cuenta la respuesta que arrojó dicha petición, el controlador renderiza una vista según el motor de plantillas Twig, construyendo la clase CP_Traducir a partir de la plantilla utilizada.

La interfaz CP_Traducir está compuesta por el formulario Form_Texto, donde el usuario introduce la información necesaria para llevar a cabo el proceso de traducción. Los datos de este formulario son enviados a la clase controladora, la cual valida los datos y hace uso del modelo, mediante el repositorio de Doctrine, para interactuar con la base de datos y guardar la información de la traducción realizada en la entidad CE_Registro, y finalmente muestra los resultados de la traducción en el formulario Form_TextoTraducido de la clase cliente CP_Traducir.

Análisis y diseño del sistema

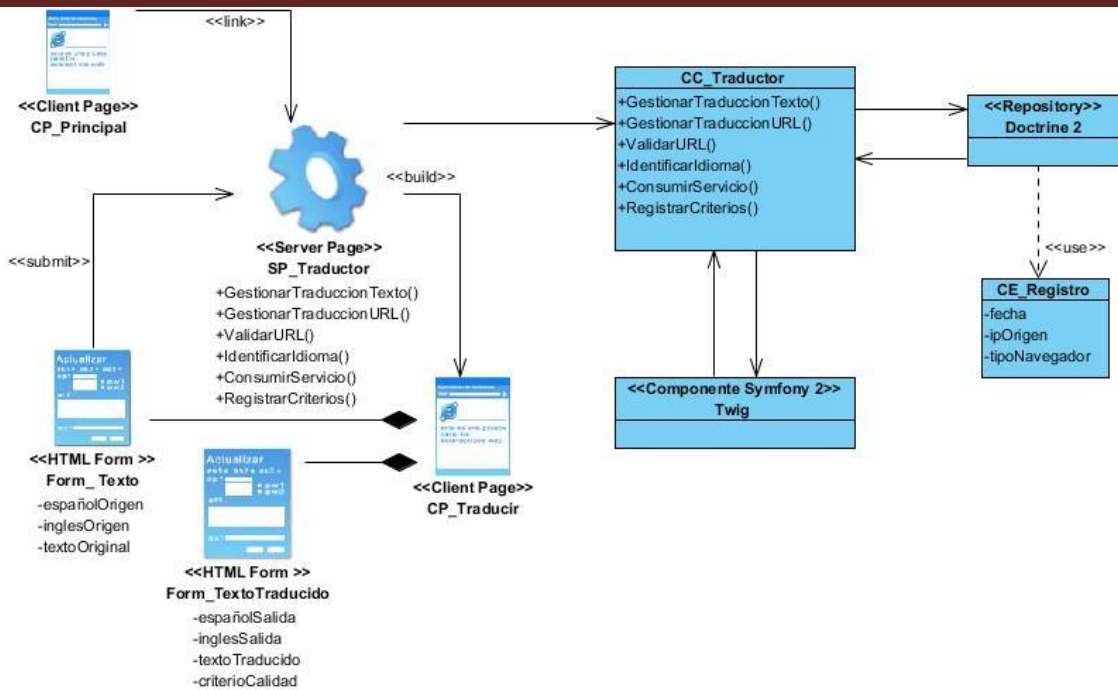


Figura 2.6: Diagrama de clases del diseño con estereotipos web del CUS “Gestionar traducción de textos”

El diagrama de clases del caso de uso “Permitir introducción de dirección URL” muestra la relación de una clase cliente CP_Principal con la clase controladora Traductor, a la cual le envía una petición, y como respuesta dicha clase controladora construye la interfaz CP_Traducir, integrada por el formulario Form_URL, donde el usuario ingresa la dirección URL de la página web que desea traducir. Los datos de este formulario se envían a la clase controladora, la cual valida los datos e interactúa con el modelo, mediante el repositorio de Doctrine, con el fin de acceder a la base de datos y almacenar la información generada durante el proceso de traducción en la entidad CE_Registro. Finalmente, esta clase controladora construye la clase cliente CP_PaginaTraducida para mostrar la página web traducida, además, dicha clase está compuesta por el formulario Form_DirURL, que permite introducir otra dirección URL que el usuario desee traducir. Las interfaces mostradas en este proceso se crean a partir del motor de plantillas Twig, componente que interactúa con el controlador para generar las vistas.

Análisis y diseño del sistema

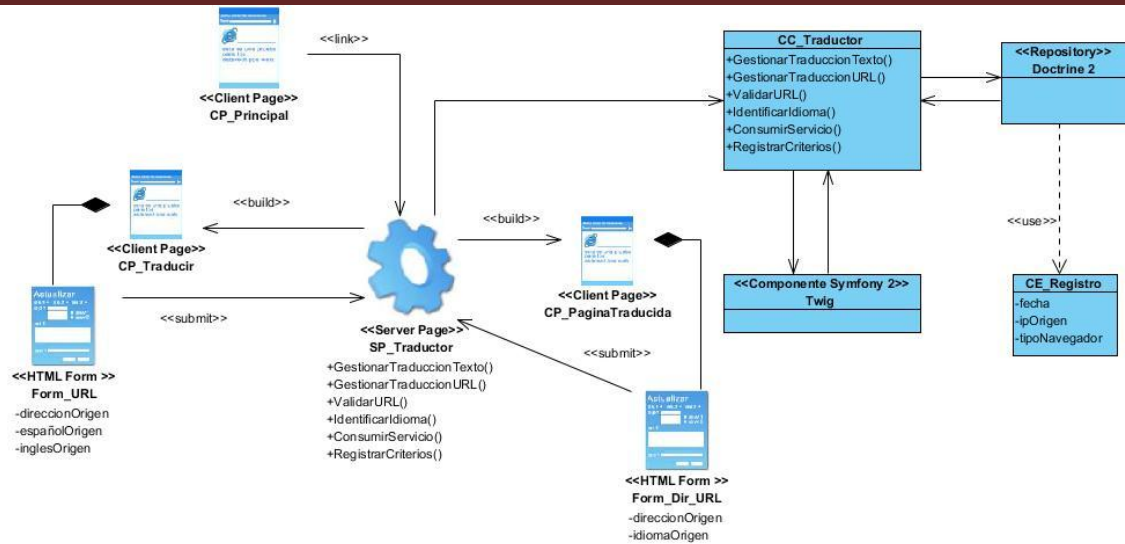


Figura 2.7: Diagrama de clases del diseño con estereotipos web del CUS “Permitir introducción de URL”

2.9 Diagrama de Interacción

Los diagramas de interacción se utilizan para modelar los aspectos dinámicos de un sistema y muestran un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos.

2.9.1 Diagramas de secuencia

Los diagramas de secuencia reflejan cómo se comunican los objetos en una interacción a través de los mensajes intercambiados entre las clases.

Con el objetivo de modelar los aspectos dinámicos del sistema, a continuación se presentan los diagramas de secuencia por cada escenario de los casos de uso críticos definidos anteriormente. Para consultar el resto de los diagramas de secuencia: **Ver anexo # 5**

Análisis y diseño del sistema

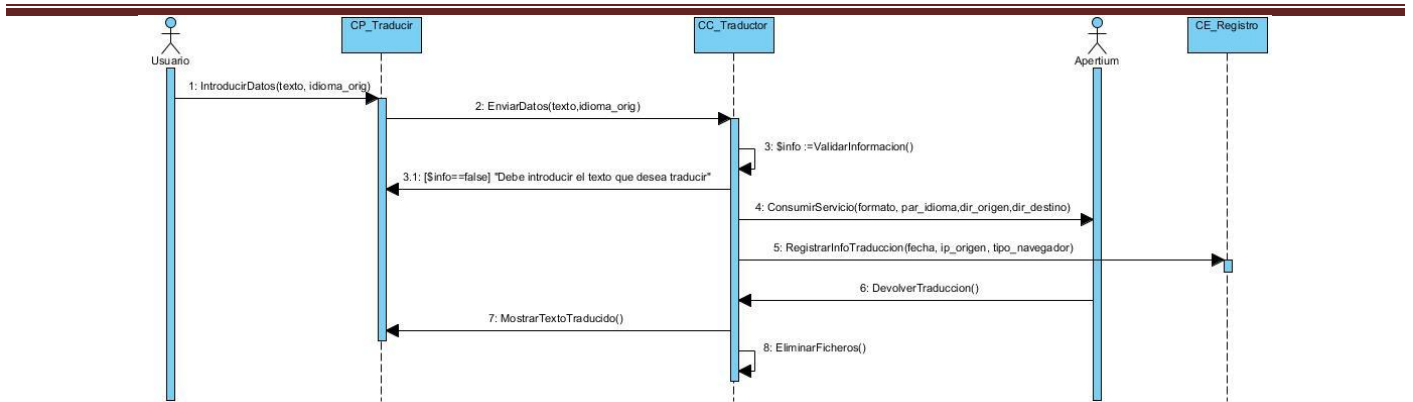


Figura 2.8: Diagrama de secuencia del CUS “Gestionar traducción de textos”

Nota: El CUS “Permitir introducción de dirección URL” es un caso de uso extendido del CUS “Permitir traducción de páginas web indexadas”, tal como se representa en el diagrama de CUS. Por tanto, el usuario puede insertar la dirección URL de una página, tanto desde la interfaz CP_Traducir, como desde la interfaz CP_PaginaTraducida, generada al traducirse una página web. Una vez introducida la URL, independientemente de la interfaz donde fue insertada, se realiza el mismo procesamiento para llevar a cabo la traducción, por ello, al realizar el diagrama de secuencia correspondiente a este caso de uso, intervienen dos interfaces distintas, desde donde el usuario puede insertar la dirección URL a traducir. Partiendo de esto, se han modelado dos diagramas de secuencia diferentes, uno para cada interfaz desde donde se inicia el caso de uso.

A continuación se muestra el diagrama de secuencia de este caso de uso para cuando el usuario introduce la dirección URL que desea traducir en la interfaz “Traducir”.

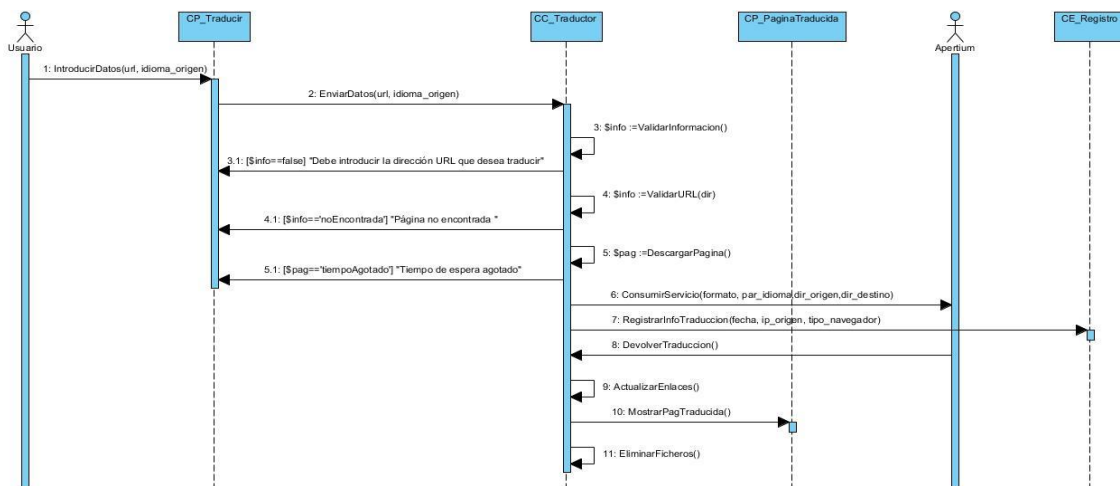


Figura 2.9: Diagrama de secuencia del CUS “Permitir introducción de dirección URL” para la interfaz “Traducir”

Análisis y diseño del sistema

2.10 Diseño de la Base de Datos

Uno de los pasos cruciales en la construcción de una aplicación que requiera manejar la persistencia de la información, es sin duda, el diseño de la base de datos, la cual, correctamente diseñada, permite obtener acceso a información exacta y actualizada. El sistema que se propone hace uso de una base de datos ya que es necesario persistir una cantidad considerable de información relacionada con las traducciones realizadas.

Como ya se ha definido anteriormente, el gestor encargado de almacenar la información capturada por la aplicación es MongoDB, una base de datos documental escalable y de alto rendimiento.

La base de datos del módulo de traducción estará compuesta por tres colecciones de datos:

- ✓ La colección criterioTexto que almacena la información referente a los criterios seleccionados por el usuario de acuerdo a la calidad de las traducciones de textos realizadas.
- ✓ La colección criterioPagina almacena los datos asociados a los criterios de calidad seleccionados por el usuario respecto a las traducciones de páginas web realizadas.
- ✓ La colección Registro guarda la información relativa a cada una de las traducciones realizadas.

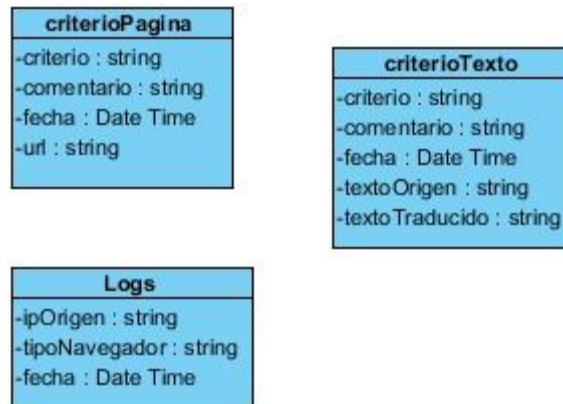


Figura 2.10: Modelo físico de datos

2.11 Modelo de Despliegue

Para una mejor comprensión entre la correspondencia de la arquitectura de software y la arquitectura de hardware se realiza el modelo de despliegue.

El modelo de despliegue que se presenta a continuación muestra el nodo "PC_Cliente", representado por un nodo ordenador el cual contiene un navegador para Internet. Este se encarga de comunicarse con el

Análisis y diseño del sistema

nodo que contiene la aplicación web a través del protocolo HTTP. Este proceso se realiza a través de los recursos que se le muestran al usuario en la página, lo que le permite establecer un sistema de comunicación con el servidor web Apache.

En el nodo “Servidor Web Apache2” se atienden las solicitudes del cliente, se analizan y se les da respuesta. En este nodo están contenidos los procesos de información que garantizan el funcionamiento de la aplicación logrando cumplir con los requerimientos funcionales del sistema. La capa del modelo se comunica con el nodo “Servidor de Bases de Datos MongoDB” a través del protocolo TCP/IP, donde se encuentra la información almacenada.

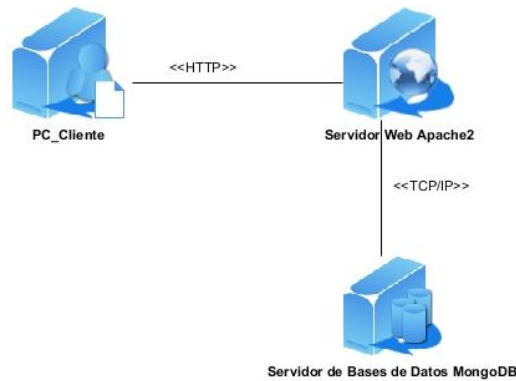


Figura 2.11: Modelo de despliegue

2.12 Conclusiones

En este capítulo se desarrolló la propuesta de solución, analizándose los requerimientos que debe tener el módulo a desarrollar, creando los casos de uso necesarios para satisfacer los mismos.

El modelo del diseño es la entrada principal a las actividades de la implementación brindando una profundización en las características de la aplicación y en las acciones que debe realizar la misma por lo que también se realizó un modelo detallado de la solución propuesta, a través del modelado de los diagramas de clases del diseño por cada realización de caso de uso del sistema, así como los diagramas de secuencia correspondiente a cada uno de los escenarios especificados. Además se construyó el modelo de despliegue con el objetivo de representar las relaciones entre los componentes de hardware que serán utilizados en la implementación.

CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBA

3.1 Introducción

La fase de implementación en el desarrollo de un producto de software, es el mecanismo donde se ponen en práctica todas las descripciones y arquitecturas propuestas en las fases de análisis y diseño, es el complemento del trabajo de las fases que lo preceden dentro del proceso de desarrollo de software. La implementación ofrece una materialización precisa de los requisitos.

Una de las últimas fases del ciclo de vida antes de entregar un programa para su explotación es la fase de pruebas, cuyo objetivo es comprobar si este cumple sus requisitos. Dentro de ella pueden desarrollarse varios tipos de pruebas en función de los objetivos de las mismas.

3.2 Diagrama de componentes

El diagrama de componentes describe cómo se implementan las clases en término de componentes, como pueden ser ficheros de código fuente, ejecutables, entre otros. Describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponible en el entorno de implementación y en el lenguaje de programación utilizado. Además, muestra las dependencias entre componentes. [54]

A continuación se muestra el diagrama de componentes propuesto, el cual está acorde con los requerimientos del sistema a implementar y con el patrón arquitectónico MVC propuesto por Symfony 2.

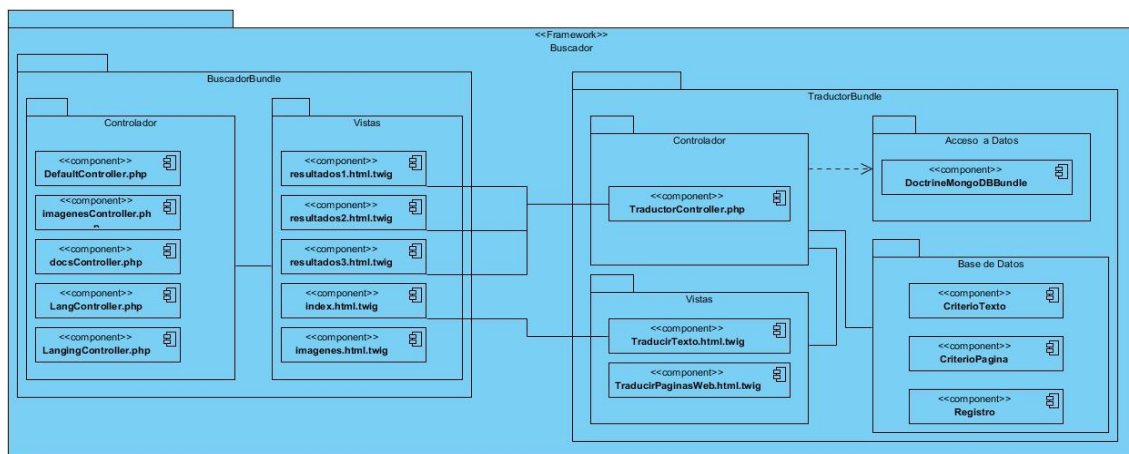


Figura 3.1: Modelo de componentes

Implementación y pruebas

El módulo desarrollado está empaquetado en un bundle principal denominado TraductorBundle, donde se encuentran distribuidos tres paquetes: modelo, vista y controlador, de tal forma que exista correspondencia con la arquitectura MVC que se ha empleado. Dentro del paquete Controlador se encuentra la clase TraductorController.php que maneja la interacción y la comunicación entre el modelo y las acciones del usuario, la cual, a su vez, está relacionada con las interfaces: resultado1.html.twig, resultado2.html.twig y resultado3.html.twig, del buscador ORION, ya que ellas son las encargadas de proporcionarle la información necesaria para realizar la traducción de las páginas web indexadas por el buscador. También existe relación entre las interfaces index.html.twig del buscador y TraducirTexto.html.twig del módulo de traducción pues, a través de dicha vista, el usuario accede desde el motor de búsqueda a la interfaz de traducción de texto que brinda el traductor. Además se representa la dependencia del sistema con el bundle DoctrineMongoDBBundle, utilizado para el acceso a datos y el trabajo con el gestor de base de datos MongoDB.

3.3 Estándar de codificación

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Si bien los programadores deben implementar un estándar de forma prudente, éste debe tender siempre a lo práctico. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez.

Para facilitar el entendimiento del código y fijar un modelo a seguir, se establecieron estándares de codificación. Los aspectos para los que generalmente se establecen estándares son los siguientes:

Identificadores

En el caso de los identificadores existen estilos definidos mundialmente como el lowerCamelCase y el UpperCamelCase. Cada palabra interna en identificadores compuestos comienza con mayúsculas para ambos estilos, además ocurre, que no se colocan caracteres de separación entre las palabras que conforman un identificador compuesto en ninguno de los dos casos. Para el primero, el identificador comienza con minúscula y para el segundo, el identificador comienza con mayúscula.

- ✓ Clase: para las clases se escogió el UpperCamelCase.
Ejemplo: `class TraductorController`
- ✓ Función: para las funciones se escogió el UpperCamelCase.

Implementación y pruebas

Ejemplo: `public function TraducirTextoAction ()`

- ✓ Variable: para las variables se determinó el empleo del estándar `lowerCamelCase`.

Ejemplo: `$saveDir;`

Indentación

La indentación es una práctica de programación que consiste en comenzar a escribir cada línea de código a diferentes distancias desde el borde izquierdo del área de texto del editor. Esta distancia está determinada por la jerarquía que se forma al introducir sentencias dentro de bloques de estructuras. En el caso del Netbeans (IDE utilizado en el desarrollo de la aplicación) se encarga de ajustar los espacios de forma automática.

Por otra parte se escribirá sólo una sentencia por línea de código y en el caso de cortar las líneas, se hará luego de una coma o antes de un operador. La sección de la derecha de la línea que se corte se ubicará en la línea siguiente indentada al nivel de la expresión correspondiente en la línea superior.

Llaves

Existe diversidad de criterios en cuanto a la ubicación de las llaves que delimitan el cuerpo de los bloques de código en los lenguajes que contienen este tipo de estructuras. Algunos programadores prefieren hacerlo ubicando la llave de apertura inmediatamente detrás de la línea cabecera del bloque, mientras otros apuestan por ubicarlas de forma solitaria en la línea siguiente a la línea cabecera. Para este último estilo existen además diferencias en cuanto al nivel de indentación de las mismas. Algunos lo hacen al nivel de la línea cabecera y otros al nivel de las líneas del cuerpo del bloque. Para este trabajo: Las llaves de apertura de las funciones se colocarán solitarias en la línea siguiente e indentadas al nivel de la línea cabecera del bloque y las llaves de aperturas de las estructuras `for`, `if`, `while`, `else`, se colocarán inmediatamente detrás de la línea cabecera del bloque. Las llaves de cierre se colocarán solitarias en la línea que sigue a la última línea dentro del bloque e indentadas al nivel de la línea cabecera del bloque.

Es prudente señalar que este estilo agrega más líneas de código al programa al ubicar las llaves solitarias en una línea, pero a su vez se gana en legibilidad del código.

Líneas en blanco

Para mejorar la legibilidad y organización del código muchas veces se utilizan líneas en blanco para separar segmentos de código que pueden corresponder a clases, funciones, declaraciones,

Implementación y pruebas

implementaciones, comentarios, bloques o sencillamente secciones críticas que se deseen despejar. En este trabajo se ha definido emplear líneas en blanco:

- ✓ Entre funciones.
- ✓ Entre declaraciones de variables e implementaciones dentro del cuerpo de las funciones.

Comentarios

Es necesario tener en cuenta algunos detalles al escribir comentarios:

- ✓ La capacidad de síntesis.
- ✓ El uso de lenguaje técnico.
- ✓ No repetir exactamente paso por paso lo que hace el algoritmo sino expresar un resumen de su propósito.
- ✓ Usar un estilo uniforme de comentario definido en estándares para todo el equipo.

Para el desarrollo del módulo de traducción automática se decidió lo siguiente:

- ✓ Debe incluirse un espacio simple entre el “//” y el texto.
- ✓ Si el comentario se aplica a un grupo de instrucciones debe estar seguido de una línea en blanco, si se refiere sólo a la siguiente instrucción, se suprime la línea.
- ✓ En los métodos se creará encima de cada función o método que se implemente una breve descripción de lo que realiza el mismo entre los comodines de comentarios /* */, además en esta descripción deberá aparecer quién creó el método y el tipo de dato que retorna.

3.4 Interfaces principales de la aplicación.

La aplicación cuenta con dos interfaces de usuario. Una de ellas es la interfaz correspondiente a la traducción de textos y páginas web, mediante la cual los usuarios podrán traducir un texto del idioma español al idioma inglés, y viceversa. Igualmente podrán insertar la dirección URL de una página web que se desee traducir. Además los usuarios tendrán la posibilidad de emitir su criterio con respecto a la calidad de la traducción realizada.

Implementación y pruebas

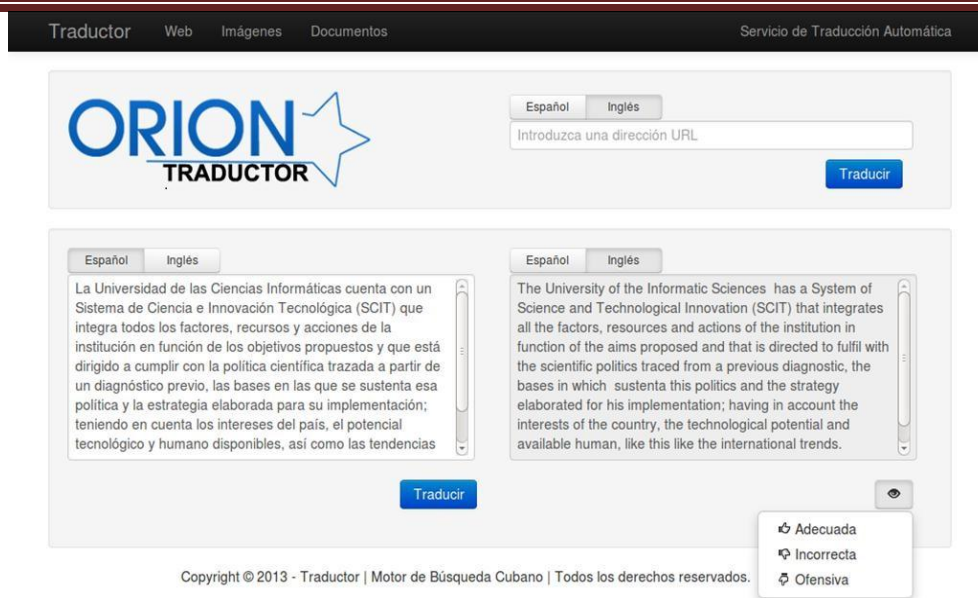


Figura 3.2: Interfaz para la traducción de textos

Otra de las interfaces con que cuenta el módulo de traducción automática es aquella que se muestra una vez que el usuario introduce una dirección URL en la interfaz mencionada anteriormente o seleccione la opción “Traducir”, en una de las páginas web indexadas por el buscador ORION. Después de mostrar la traducción de la página web brinda la posibilidad de insertar otra dirección URL y también permite emitir un criterio con respecto a la calidad de la traducción realizada.

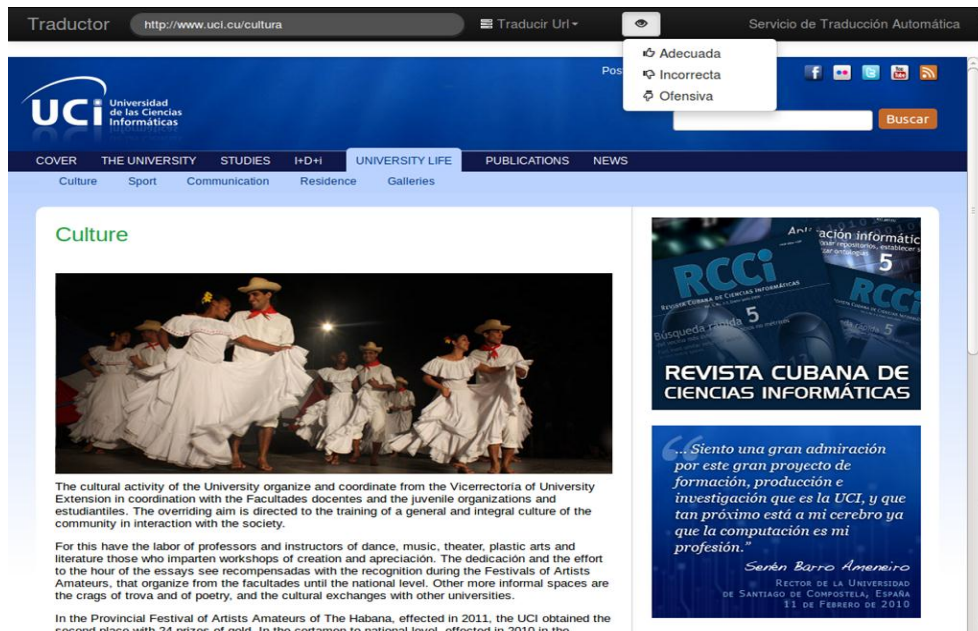


Figura 3.3: Interfaz de una página web traducida

Implementación y pruebas

3.5 Validación del sistema.

Una vez terminada la implementación del producto que se requiere es necesario realizarle pruebas con el objetivo de detectar errores en la aplicación y la documentación; este proceso resulta de gran importancia ya que da una medida de la calidad del mismo siempre que se lleve a cabo de la forma correcta.

3.5.1 Pruebas funcionales

Las pruebas funcionales, también denominadas pruebas de comportamiento o de caja negra se centran en los requisitos funcionales del software. O sea, la prueba de caja negra permite al ingeniero del software obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. [55] Por ello se denominan pruebas funcionales, y el probador se limita a suministrarle datos como entrada y estudiar la salida, sin preocuparse de lo que pueda estar haciendo el módulo por dentro.

El método de prueba que se utilizará será Partición equivalente, el mismo consiste en diseñar un caso de prueba por cada caso de uso, para los casos de uso incluidos no se diseñarán casos de pruebas, sino que estos se agregarán en el caso de uso que los contenga y se probarán cuántas veces aparezcan en el software.

A continuación se presentan los casos de prueba correspondientes a los principales casos de uso del sistema. Para consultar el resto: **Ver anexo # 6.**

Caso de prueba para el CUS “Gestionar traducción de textos”

❖ Descripción general

A partir de que el usuario introduce el texto que desea traducir y selecciona el idioma en el que se encuentra, la aplicación realiza la traducción del mismo a través del motor de traducción automática Apertium, para mostrarlo posteriormente al usuario.

❖ Condiciones de ejecución

El usuario ha pulsado la opción “Traductor” en el motor de búsqueda ORION.

Escenario	Descripción	Variable 1 Texto a traducir	Variable 2 Idioma del texto	Respuesta del sistema	Flujo central
-----------	-------------	--------------------------------	--------------------------------	-----------------------	---------------

Implementación y pruebas

			origen		
EC 1.1 Introducir los datos para la traducción correctamente.	El usuario introduce todos los datos para realizar la traducción automática.	V <i>“Mi nombre es Carmen”</i>	V <i>“español”</i>	Realiza la traducción automática del texto introducido según el idioma seleccionado, y muestra el texto traducido: <i>“My name is Carmen”</i> .	<ol style="list-style-type: none"> 1. Introducir el texto que se desea traducir. 2. Seleccionar el idioma en el que se encuentra dicho texto. 3. Presionar el botón “Traducir”.
EC 1.2 Introducir solo el texto a traducir.	El usuario solo introduce el texto que desea traducir.	V <i>“My name is Carmen”</i>	N/A	Realiza la traducción automática por defecto al idioma inglés del texto introducido, mostrando el resultado en la propia interfaz: <i>“My name is Carmen”</i> .	<ol style="list-style-type: none"> 1. Introducir el texto que se desea traducir. 2. Presionar el botón “Traducir”.
EC 1.3 Introducir datos para la traducción incorrectamente.	El usuario no introduce los datos necesarios para realizar la traducción automática.	I	V <i>“español”</i>	Muestra el mensaje de error “Debe introducir el texto que desea traducir”.	<ol style="list-style-type: none"> 1. No introducir el texto que se desea traducir. 2. Seleccionar un idioma origen. 3. Presionar el botón “Traducir”.
		I	N/A		<ol style="list-style-type: none"> 1. No introducir el texto que se desea traducir. 2. No seleccionar un idioma origen. 3. Presionar el botón “Traducir”.

Tabla 6: Caso de prueba para el CUS “Gestionar traducción de textos”

Implementación y pruebas

Caso de prueba para el CUS “Permitir introducción de dirección URL”

❖ Descripción general

A partir de que el usuario introduce la dirección URL de la página web que desea traducir, la aplicación se encarga de identificar en qué idioma se encuentra para llevar a cabo la traducción de la misma a través del motor de traducción automática Apertium, y mostrarla posteriormente al usuario.

❖ Condiciones de ejecución

Para que la aplicación ejecute el caso de uso en cuestión se deben cumplir alguna de las siguientes precondiciones:

- ✓ El usuario ha seleccionado la opción “Traductor” en la interfaz del motor de búsqueda ORION.
- ✓ El usuario ha seleccionado la opción “Traducir” en una de las páginas web indexadas por el buscador y se ha mostrado dicha página traducida.

Escenario	Descripción	Variable 1 Dirección URL a traducir	Respuesta del sistema	Flujo central
EC 1.1 Introducir los datos para la traducción correctamente.	El usuario introduce, correctamente, la dirección de la página web que desea traducir.	V “http://www.uci.cu”	Realiza la traducción automática de la página web, mostrándola en la interfaz correspondiente.	1. Introducir la dirección URL de la página web que se desea traducir. 2. Presionar el botón “Traducir”.
EC 1.2 Introducir los datos para la traducción de manera incorrecta.	El usuario no introdujo la dirección URL de la página web que desea traducir	I	Muestra el mensaje de error: “Debe introducir la dirección URL de la página web que desea traducir”.	1. No introducir la dirección URL de la página web que se desea traducir. 2. Presionar el botón “Traducir”.
	La dirección URL introducida por el usuario no se encuentra.	I “http://www.uci.cu52”	Muestra el mensaje de error: “Página no encontrada”.	1. Introducir una dirección URL incorrecta. 2. Presionar el botón “Traducir”.
	El tiempo de espera para descargar la	I	Muestra el mensaje de error: “Tiempo de espera	1. Introducir una dirección URL que supere el tiempo límite de

Implementación y pruebas

	página supera el tiempo establecido.		agotado”.	descarga establecido. 2. Presionar el botón “Traducir”.
--	--------------------------------------	--	-----------	--

Tabla 7: Caso de prueba para el CUS “Permitir introducción de dirección URL”

A partir del diseño y ejecución de los casos de prueba, introduciendo juegos de datos, tanto correctos como incorrectos, se detectaron errores tales como: funcionalidades ausentes, errores de interfaz, no se valida la entrada de datos incorrectos y funcionalidades incorrectas, para un total de siete no conformidades reflejadas en el **anexo # 7**.

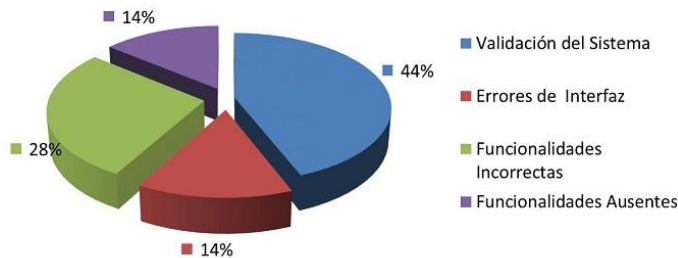


Figura 3.4: Porcentaje de No Conformidades detectadas

La gráfica que se muestra representa los resultados alcanzados en las pruebas funcionales, representando en porcentaje las no conformidades detectadas. El 14 % representa las funcionalidades ausentes, un 28 % simboliza a las funcionalidades incorrectas, el 14 % significa los errores de interfaz y el 44 % constituye la validación de datos. Al realizar un análisis de los resultados, se concluye que la mayor cantidad de no conformidades se refiere a la validación de datos.

Las no conformidades fueron corregidas, dándole paso a una segunda iteración de pruebas, en la cual se evaluaron nuevamente las funcionalidades, obteniendo resultados satisfactorios.

3.5.2 Pruebas de integración

La necesidad de realizar las pruebas de integración viene dada por el hecho de que los módulos que forman un programa suelen fallar cuando trabajan de forma conjunta, aunque previamente se haya demostrado que funcionan correctamente de manera individual. Con el uso de estas pruebas se consigue

Implementación y pruebas

ir formando el programa global a medida que se comprueba cómo interaccionan los distintos componentes y se comunican, libres de errores. [55]

A partir de la realización de las pruebas funcionales a cada uno de los componentes internos del módulo de traducción automática, de manera independiente, fue posible determinar su correcto funcionamiento mediante el estudio del flujo de datos entre ellos. Una vez realizada dichas pruebas resulta imprescindible la ejecución de pruebas de integración, con el objetivo de comprobar el intercambio de información en cada una de las interfaces expuestas por la aplicación y su interacción una vez que se haya integrado al buscador ORION.

Mediante la ejecución de las pruebas de integración se logró verificar la operación conjunta de cada uno de los componentes del módulo en cuestión, junto a los componentes que conforman el modelo de implementación del buscador ORION, haciendo énfasis en la interacción entre estos, lo que permitió la detección de incoherencias en el funcionamiento de la aplicación. También fue posible la detección de deficiencias en el manejo de la información ya que, en ocasiones, los datos obtenidos del buscador, como por ejemplo, los datos necesarios para registrar la información de la traducción, no eran los correctos. Dichas deficiencias fueron corregidas, logrando así, una correcta integración de los componentes del módulo de traducción automática entre sí y con el motor de búsqueda.

3.5.3 Pruebas de carga y estrés

Las pruebas de carga consisten en probar el funcionamiento del software bajo condiciones extremas. Estudia la especificación del software, las funciones que debe realizar, las entradas y las salidas, analizando los valores límites.

Las pruebas de estrés están diseñadas para enfrentar al programa a condiciones anormales. La prueba ejecuta un sistema, de manera que demande recursos en cantidad, frecuencia o volúmenes extremos.

Para el módulo de traducción automática es preciso realizar dichas pruebas pues resulta necesario comprobar el rendimiento del sistema soportando una cantidad máxima de usuarios que soliciten traducciones automáticas y su comportamiento al aumentar esta carga con los mismos recursos disponibles.

Con tal objetivo se desarrollaron las pruebas de rendimiento utilizando la herramienta JMeter, lo que arrojó importantes datos acerca del estrés que puede soportar el sistema. Se realizó una simulación de carga al sistema con un total de **1065** peticiones enviadas en **15** hilos de ejecución concurrentes, arrojando un **0.0071%** de error al atender dichas peticiones y un tiempo promedio de respuesta de **5.312** segundos.

Implementación y pruebas

Para realizar la validación de los resultados de las pruebas se verificó, además, la influencia que ejercen algunos parámetros sobre el tiempo de respuesta de la aplicación, elementos que pueden provocar cierto retraso en la generación de la carga en las pruebas e interferencia en la obtención de los resultados, dichos parámetros a tener en cuenta son:

- ✓ Utilización del CPU: Una utilización por debajo del 70% demuestra un rendimiento aceptable, sin embargo por encima de este valor demuestra que el sistema está sobrecargado.
- ✓ Utilización de la memoria: Un uso de la memoria física de la máquina menor al 75 % es aceptable, sin embargo si el valor es igual a este o está por encima de él, demuestra la invalidación de los resultados de las pruebas.

Para comprobar los parámetros mencionados, hay que tener en cuenta que la simulación fue realizada en un servidor de bajas prestaciones con las siguientes características: sistema operativo Ubuntu 12.04, microprocesador a 2.6 GHz y una memoria RAM de 2GB. Durante la ejecución de la prueba el CPU se mantuvo entre el 50 y el 60 % de utilización y la memoria física estuvo por debajo del 75 %.

3.6 Conclusiones

En este capítulo se realizó una estructuración del modelo de implementación en términos de subsistemas, con el objetivo de lograr una mayor comprensión de cómo está integrada la aplicación y sus dependencias. Se determinaron los estándares de codificación a emplear durante la implementación, lo cual permitió lograr una mayor legibilidad y limpieza del código. A través de la validación del sistema, aplicando pruebas funcionales, de integración y de rendimiento, se pudieron detectar errores y problemas para su posterior solución, y permitieron monitorear el funcionamiento de la aplicación en diferentes entornos de ejecución y carga.

Conclusiones

CONCLUSIONES GENERALES

El estudio del estado del arte permitió comprender cómo funcionan los servicios de traducción automática que se brindan actualmente en el mundo. Partiendo de dicho funcionamiento se logró diseñar y desarrollar una aplicación web para la traducción automática de textos y páginas web, basada en las buenas prácticas identificadas en los sistemas estudiados y encaminada a brindar una solución viable al problema identificado. Asimismo, se determinaron las herramientas y tecnologías adecuadas para su desarrollo. La aplicación de pruebas permitió validar el correcto funcionamiento de la aplicación desarrollada simulando ambientes reales en diferentes entornos de ejecución.

Por todo lo expuesto anteriormente se concluye que los objetivos propuestos para el presente trabajo han sido cumplidos satisfactoriamente. La aplicación desarrollada contribuirá de manera significativa a la traducción automática de los contenidos publicados en la intranet cubana.

Recomendaciones

RECOMENDACIONES

A pesar de haberse cumplido los objetivos de este trabajo, en la elaboración del mismo surgieron algunas ideas para enriquecerlo de forma que sea una aplicación de mayores prestaciones:

- Ampliar la traducción automática de textos y páginas web a otros idiomas.
- Permitir la traducción automática de documentos con diferentes formatos.

Referencias bibliográficas

REFERENCIAS BIBLIOGRAFICAS

1. *Resumen de estadísticas* Disponible en: <http://www.ethnologue.com/>
2. *Sistema Universitario Cubano* (2012). Ministerio de la Educación Superior: Disponible en: http://www.mes.edu.cu/index.php?option=com_content&task=view&id=14&Itemid=30.
3. MACÍAS-CHAPULA, C. A. (2005). Hacia un modelo de comunicación en salud pública en América Latina y el Caribe. *Revista Panamericana de Salud Pública*, 2005, nº p. 427-438.
4. RODRÍGUEZ, E. D. (2009). Blogs y periodismo en Cuba: entre el “deber ser” y la realidad. *Revista Latina de Comunicación Social*, 2009, vol. 64, nº p. 961-967. Disponible en: http://www.revistalatinacs.org/09/art/873_Habana/74_97_Elaine_Diaz.html.
5. YEBRA, V. G. (1997). *Teoría y práctica de la traducción*. 3 ed. 1997. vol. 53 de Biblioteca Románica Hispánica: Manuales, ISBN 9788424918408.
6. VINAY, J. P. y DARBELNET, J. (1987). *Stylistique comparée du français et de l'anglais: méthode de traduction*. 1987. ISBN 9782278008940.
7. SELESKOVITCH, D. y LEDERER, M. (2001). *Interpréter pour traduire*. 2001. ISBN 9782864606413.
8. NIDA, E. A. y TABER, C. R. (1986). *La Traducción: teoría y práctica*. Cristiandad, Ediciones, S.L., 1986. 267 p. ISBN 9788470573972.
9. HATIM, B. y MASON, I. (1990). *Discourse and the translator*. Longman, 1990. 258 p. ISBN 9780582021907.
10. DELISLE, J. (1980). *L'analyse du discours comme méthode de traduction: initiation à la traduction française de textes pragmatiques anglais: théorie et pratique*. Éditions de l'Université d'Ottawa, 1980. ISBN 9782760346529.
11. ALBIR, A. H. (2001). *Traducción y Traductología: Introducción a la Traductología*. Ediciones Cátedra, 2001. ISBN 9788437619415.
12. *Sitio oficial Softísimo* (2009). Disponible en: <http://www.softissimo.com/>.
13. HERNÁNDEZ, P. (2002). *En torno a la traducción automática*. 2002. vol. 2, 17 p.
14. JURAFSKY, D. y MARTIN, J. H. (2009). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. 2 ed. Pearson Prentice Hall/Pearson education international, 2009. 988 p. ISBN 9780131873216.

Referencias bibliográficas

15. ZAPATA, I. (1995). *Un recorrido por algunos principios de la Traducción Automática*. Enseñanza de lenguas y traducción con ordenadores, Ediciones pedagógicas, Madrid, 1995, 87 p-126 p.
16. *Sitio oficial del motor de búsqueda Altavista* (2000). [Consultado el: 15 de octubre de 2012]. Disponible en: <http://es.altavista.com/about/>.
17. *Sección de Ayuda* Sitio oficial del motor de búsqueda Google: [Consultado el: 24 de octubre de 2012]. Disponible en: <http://support.google.com/>.
18. *Sitio oficial del buscador Bing* (2005). Disponible en: [Bing Product Guide](#).
19. *Sección de Ayuda* (2000). Sitio oficial del motor de búsqueda Bing: [Consultado el: 27 de octubre de 2012]. Disponible en: <http://www.bing.com/translator/help/>.
20. *Sitio oficial del servicio de traducción automática del Instituto de Cervantes* Disponible en: http://traductor.cervantes.es/preguntas_frecuentes_traductor.htm.
21. *Sitio oficial del motor de traducción automática SYSTRAN* (2010). [Consultado el: 5 de diciembre de 2012]. Disponible en: <http://www.systran.es/>.
22. *Productos de traducción* (2010). Sitio oficial del motor de traducción automática SYSTRAN [Consultado el: 5 de diciembre de 2012]. Disponible en: <http://www.systran.es/productos-de-traducccion/servidor/systran-enterprise-server/requisitos-del-sistema>.
23. BORRELL, C. y OLIVER, A. (2005). *Gestionando Babel: un aula virtual multilingüe en tiempo real*. Universidad de Catalunya, 2005, 25 p. Disponible en: <http://lpg.uoc.edu/aoliverg/uploads/arxiu/borrell-oliver-moodlemoot.pdf>.
24. AMO, F. A.; NORMAND, L. A., *et al.* (2005). *Introducción a la ingeniería del software*. Delta Publicaciones, 2005. 542 p. ISBN 9788496477001.
25. FIGUEROA, R. G.; SOLÍS, C. J., *et al.* (2002). *Metodologías tradicionales vs. metodologías ágiles*. Universidad Técnica Particular de Loja.
26. *Metodología RUP. Metodología XP*. Universidad de San Martín, Disponible en: <http://www.usmp.edu.pe/publicaciones/boletin/fia/info49/articulos/RUP%20vs.20XP.pdf>.
27. *Sitio oficial de Extreme Programming* (2009). [Consultado el: 17 de enero de 2013]. Disponible en: <http://www.extremeprogramming.org/>.
28. *Reglas de Extreme Programming* (2009). Sitio oficial de Extreme Programming [Consultado el: 17 de enero de 2013]. Disponible en: <http://www.extremeprogramming.org/rules.html>.
29. ROSALES, Y.; LOBO, A., *et al.* (2010). *Plugin de la herramienta Visual Paradigm para la evaluación del diseño Orientado a Objetos*. Universidad de las Ciencias Informáticas, 2010, Disponible en:

Referencias bibliográficas

- <http://publicaciones.uci.cu/index.php/SC>.
30. HERRERA, G. (2006). PHP. 2006, nº Disponible en:
<http://www.vaslibre.org.ve/publicaciones/phpflisol2006.pdf>.
31. VALLECILLO, A. y FUENTES, L. (2004). Una introducción a los perfiles UML. *Revista Novática-Asociación de Técnicos de Informática-España*, 2004.
32. *Integration definition for function modeling IDEF0* (1993). IDEF Integrated Definition Methods: Disponible en: <http://www.idef.com/Downloads.htm>
33. BOOCH, G. A.; RUMBAUGH, J., et al. (2006). *UML: el lenguaje unificado de modelado: guía del usuario*. Pearson Addison Wesley, 2006. vol. 1, 527 p. ISBN 9788478290765.
34. STEVENS, P.; POOLEY, R. J., et al. (2007). *Utilización del UML en ingeniería del software con objetos y componentes*. 2 ed. Pearson Educación, 2007. 304 p. ISBN 9788478290864.
35. G.S.I. *Rational Rose Enterprise* Rational Rose Enterprise [Consultado el: 7 de febrero de 2013]. Disponible en:
<http://www.rational.com.ar/herramientas/roseenterprise.html>
36. TRAINING, A. O. (2005). *ALTOVA*. ALTOVA Inc.: Vervante, 2005. ISBN 9781933210056.
37. MUSCIANO, C. y KENNEDY, B. (1997). *HTLM: La guía completa*. 2 ed. O'Reilly, 1997. ISBN 9789701021415.
38. ALVAREZ, M. A. (2009). *Qué es JavaScript y las posibilidades que nos ofrece con respecto al HTML*. [Consultado el: 7 de febrero de 2013]. Disponible en:
<http://www.desarrolloweb.com/articulos/490.php>.
39. GUTMANS, A.; BAKKEN, S. S., et al. (2005). *PHP 5 Power Programming*. Ilustrada ed. Prentice Hall PTR, 2005. 689 p. ISBN 9780131471498.
40. *PHP* Sitio oficial de PHP: [Consultado el: 9 de febrero de 2013]. Disponible en:
<http://www.php.net/manual/es/intro-what-is.php>
41. EGUILUZ, J. (2011). *Desarrollo web ágil con Symfony2*. 1 ed. 2011. 576 p.
42. *Welcome to Netbeans* (2011). Sitio oficial del IDE Netbeans: [Consultado el: 13 de febrero de 2013]. Disponible en: <http://netbeans.org/>.
43. KABIR, M. J. (1999). *La Biblia Del Servidor Apache*. Anaya Multimedia, S.A., 1999. ISBN 9788441508071
44. *XtremWeb-CH* (2012). Disponible en: <http://www.xtremwebch.net/index.html>.
45. MARTÍNEZ, R. *Postgresql* [Consultado el: 13 de febrero de 2013]. Disponible en:
<http://www.postgresql-es.org/principal>.

Referencias bibliográficas

46. ANDERSON, D. P. (2004). *A system for public-resource computing and storage*. 2004.
47. MongoDB (2011). Sitio oficial del Gestor de Base de Datos MongoDB: [Consultado el: 13 de febrero de 2013]. Disponible en:
<http://www.mongodb.com>.
48. RÍOS, S. S. (2007). *Análisis y UML. Modelo conceptual*. 2007, Disponible en:
http://www.uvmsf.cl/~ssanchez/images/Metodologias/Unidad4_MAD.pdf.
49. LARMAN, C. (2003). *UML y patrones: una introducción al análisis y diseño orientado a objetos y al proceso unificado*. 2 ed. Pearson Prentice Hall, 2003. 590 p. ISBN 9788420534381.
50. RUMBAUGH, J. (1998). *Modelado y diseño orientado a objetos*. 2 ed. Prentice Hall, 1998. 675 p. ISBN 9780228191995.
51. REYNOSO, C. (2004). *Estilos y Patrones en la Estrategia de Arquitectura de Microsoft*. 2004.
52. GARCÍA, J.; RODRÍGUEZ, J., et al. *Un estudio comparativo de dos herramientas MDA: OptimaJ y ArcStyler*. Departamento de Informática y Sistemas, Universidad de Murcia.
53. WELICKI, L. (2010). *Patrones y Antipatrones: una introducción*. Microsoft Corporation, 2010, Disponible en: <http://msdn.microsoft.com/es-es/library/bb972242.aspx#M2>.
54. PRESSMAN, R. S. (2010). *Ingeniería de software*. 7 ed. McGraw-Hill Interamericana de España S.L., 2010. 959 p. ISBN 9786071503145.
55. MAÑAS, J. A. (1994). *Pruebas de Programas*. 1994, Disponible en:
<http://www.it.uc3m.es/tsps/testing.htm>.

Bibliografía

BIBLIOGRAFÍA

1. CARRASCO, O. M. F.; LEÓN, D. G., *et al.* (1995). *Un enfoque actual sobre la calidad del software*. 1995.
2. CASTILLO, R. A. (1990). *Traducción y terminología científica en Cuba: estado actual y algunas perspectivas*. Editorial Científico-Técnica, 1990. 103 p.
3. CRACIUNESCU, O. y GERDING-SALAS, C. *Traducción automática y asistida: ¿nuevas formas de traducir?* 17 p.
4. DURÁN, J. D. y RUIZ, A. (2007). *¿Por qué OMG ha elegido BPMN para modelar procesos de negocio si ya existe UML?* España: 2007.
5. FOWLER, M. A. y SCOTT, K. A. (1999). *UML gota a gota*. Pearson Educación, 1999. 203 p. ISBN 9789684443648.
6. FRANCIA, S. (2012). *MongoDB and PHP*. Oreilly & Associates Incorporated, 2012. 62 p. ISBN 9781449314361.
7. GESCHWINDE, E. y SCHÖNIG, H. J. (2002). *Php and Postgresql: Advanced Web Programming*. 3 ed. Sams Pub., 2002. ISBN 9780672323829.
8. GONZÁLEZ, A. O.; LÓPEZ, J. M., *et al.* (2011). *Traducción y tecnologías*. Editorial UOC, S.L., 2011. 316 p. Manuales (Universitat Oberta de Catalunya). ISBN 9788497883658.
9. MORENO, J. F. Q. y DE AMORES CARREDANO, J. G. (2000). *Diseño e Implementación de Sistema de Traducción Automática*. Universidad de Sevilla. Secretariado de Publicaciones, 2000. 564 p. Ciencias (Universidad de Sevilla). ISBN 9788447205851.
10. SÁNCHEZ, R. M. y ACEBAL, J. B. M. (2009). *Desarrollo de un traductor automático estadístico catalán/castellano*. 2009.
11. SANZ, L. F. y VILLALÓN, J. A. C. M. (1996). *Ingeniería del software: pruebas del software*. Universidad Politécnica de Madrid. Facultad de Informática, 1996. 79 p. ISBN 9788485632770.
12. SCHULZ, R. G. (2008). *Diseño Web con CSS*. Marcombo, S.A., 2008. 304 p. ISBN 9788426714701.
13. SURHONE, L. M.; TENNOE, M. T., *et al.* (2010). *Apertium*. VDM Publishing, 2010. 74 p. ISBN 9786131368196.
14. TERUEL, A. (2001). *El Plan de Pruebas*. 2001, Disponible en: <http://www ldc.usb.ve/~teruel/ci4713/clases2001/planPruebas.html>.

Glosario de términos

GLOSARIO DE TÉRMINOS

API: Conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Clasificación morfológica: Clasificación de una determinada palabra atendiendo a que parte de la oración representa, o sea, si es un adjetivo, verbo, pronombre personal, pronombre posesivo, adverbio, conjunción, pronombre interrogativo, pronombre relativo o preposición.

Corpus-monolingüe: Recurso lingüístico consistente un texto de una misma lengua (en algún formato electrónico adecuado).

Corpus-paralelo: Recurso lingüístico consistente en textos de dos lenguas (en algún formato electrónico adecuado) que están alineados a cierto nivel de granularidad, generalmente a nivel de párrafo, aunque también a nivel de sección, pagina o incluso a veces de palabra.

Con alineación se refiere a que dada frase o palabra en el texto origen, sea capaz de determinar cuál es su traducción correspondiente en el texto destino.

Multiplataforma: es un término usado para referirse a los programas, sistemas operativos, lenguajes de programación, herramientas informáticas, u otra clase de software, que puedan funcionar en diversas plataformas. Por ejemplo, una aplicación multiplataforma podría ejecutarse en Windows, GNU/Linux y Mac OS X.

Lingüística computacional: Campo multidisciplinar de la lingüística y la informática que utiliza la informática para estudiar y tratar el lenguaje humano. Para lograrlo, intenta modelar de forma lógica el lenguaje natural desde un punto de vista computacional.

Semántica: Se refiere a los aspectos del significado, sentido o interpretación del significado de un determinado elemento, símbolo, palabra, lenguaje o representación formal. En principio cualquier medio de expresión (lenguaje formal o natural) admite una correspondencia entre expresiones de símbolos o palabras y situaciones o conjuntos de cosas que se encuentran en el mundo físico o abstracto que puede ser descrito por dicho medio de expresión.

Anexos

ANEXOS

Anexo # 1: Prioridad y complejidad de los casos de uso del sistema

La complejidad y la prioridad de los casos de uso del sistema están determinadas a partir del artefacto "Evaluación de casos de uso", propuesto por la metodología de desarrollo de software OpenUP, utilizada para guiar el desarrollo del módulo de traducción automática.

Tabla 8: Determinación de la complejidad de los casos de uso del sistema

Casos de uso	Criterios de Complejidad						Complejidad resultante
	Transacciones	Entidades candidatas	Comunicación con actores	Requisitos no funcionales	Tecnología	Reutilización	
Gestionar traducción de textos	A	B	M	A	A	M	Media
Permitir introducción de dirección URL	A	B	M	A	A	M	Media
Permitir traducción de páginas web indexadas	A	B	M	A	A	M	Media
Gestionar traducción de páginas web	A	B	M	A	A	M	Media
Consumir servicio	M	B	B	M	A	B	Baja
Registrar	A	M	M	M	A	A	Media

Anexos

criterio de calidad						
---------------------	--	--	--	--	--	--

Tabla 9: Determinación de la prioridad de los casos de uso del sistema

Casos de uso	Criterios de prioridad				Prioridad resultante
	Criticidad	Dependencia	Estabilidad	Frecuencia	
Gestionar traducción de textos	A	M	A	A	Alta
Permitir introducción de dirección URL	A	A	A	A	Alta
Permitir traducción de páginas web indexadas	A	A	A	A	Alta
Gestionar traducción de páginas web	A	B	A	A	Media
Consumir servicio	A	B	A	A	Media
Registrar criterio de calidad	B	B	A	B	Baja

Anexo # 2: Especificación de requisitos funcionales

Especificación del RF “Permitir la traducción de páginas web indexadas por el motor de búsqueda ORION”

Nombre	Descripción	Complejidad	Prioridad para cliente
Permitir la traducción de páginas web indexadas por el motor de búsqueda ORION	El usuario puede seleccionar la opción “Traducir”, en alguna de las páginas web indexadas por el buscador ORION, después de realizada una búsqueda.	Media	Alta
Prototipo			

Anexos



Especificación del RF "Registrar elección del criterio de calidad de la traducción"

Nombre	Descripción	Complejidad	Prioridad para cliente
Registrar elección del criterio de calidad de la traducción.	El usuario puede seleccionar uno de los criterios de calidad disponibles una vez que se muestra un texto o una página web traducida.	Media	Baja
Prototipo			
Prototipo de interfaz que se muestra una vez que el usuario seleccione un criterio de calidad			
<p>The prototyping image shows a rectangular frame containing the text "Criterio de calidad Adecuada" in blue. Below this text is a blue button with the word "Enviar" in white.</p>			
Campos	Tipos de Datos	Reglas o Restricciones	
apropiada	String	No procede	
incorrecta	String	No procede	
ofensiva	String	No procede	

Anexos

Especificación del RF “Registrar elección del criterio de calidad de la traducción”

Nombre	Descripción	Complejidad	Prioridad para cliente
Registrar motivos de elección del criterio de calidad de la traducción	Si criterio de calidad seleccionado por el usuario es “Incorrecta” u “Ofensiva”, se le muestra un campo de texto donde pueden insertar comentarios que justifiquen dicha elección.	Media	Baja
Campos	Tipos de Datos	Reglas o Restricciones	
txt_opinion	Text	No procede	
Prototipo			
			
Observaciones	Este requisito se ejecuta solamente cuando el usuario selecciona como criterio de calidad: “Incorrecta” u “Ofensiva”.		

Anexo # 3: Especificación de casos de uso del sistema

Descripción del CUS “Permitir traducción de páginas web indexadas”

Objetivo	Permitir la traducción automática de páginas web indexadas por el buscador ORION.
Actores	Usuario
Resumen	El caso de uso se inicia cuando el usuario pulsa la opción “Traducir” en una de las páginas web indexadas por el buscador. A partir de la dirección URL seleccionada, la aplicación se encarga de gestionar, posteriormente, todo el proceso de traducción hasta mostrar la página web traducida.

Anexos

Complejidad	Media	
Prioridad	Alta	
Precondiciones	El usuario ha realizado una búsqueda en la interfaz del motor de búsqueda ORION.	
Postcondiciones	Se mostró la página web traducida al usuario.	
Flujo de eventos		
Flujo básico Permitir traducción de páginas web indexadas		
	Actor	Sistema
1.	En una de las páginas web indexadas por el buscador ORION, selecciona la opción "Traducir".	
2.		Obtiene la dirección URL de la página web a traducir.
3.		Identifica el idioma en el que se encuentra la página web a traducir.
4.		<u>Ver CU: "Gestionar traducción de páginas web"</u> .
5.		Termina el CU.
Flujos alternos		
	Actor	Sistema
Relaciones	CU Incluidos	<u>Ver CU: "Gestionar traducción de páginas web"</u> .
	CU Extendidos	
Requisitos funcionales	no	
Asuntos pendientes		

Descripción del CUS "Gestionar traducción de páginas web"

Objetivo	Gestionar el proceso de traducción de las páginas web.
Actores	Usuario
Resumen	El caso de uso se inicia cuando la aplicación obtiene la dirección URL que

Anexos

	el usuario desea traducir, utilizada para llevar a cabo el proceso de traducción. El caso de uso termina al mostrar la página web traducida.	
Complejidad	Media	
Prioridad	Media	
Precondiciones	El usuario ha seleccionado la opción "Traducir" en la interfaz del motor de búsqueda ORION o ha introducido la dirección URL que desea traducir.	
Postcondiciones	Se mostró la página web traducida.	
Flujo de eventos		
Flujo básico Gestionar traducción de páginas web		
	Actor	Sistema
1.		Descarga la página web a partir de la dirección URL.
2.		En caso de descargar la página web correctamente, <u>Ver CU: "Consumir servicio de traducción"</u> .
3.		Actualiza las direcciones de las imágenes, enlaces, css y scripts de la página web.
4.		Muestra la página web traducida.
5.		Elimina los ficheros locales generados durante la traducción.
6.		Termina el CU.
Flujos alternos		
1º. La dirección URL introducida por el usuario supera el tiempo de espera establecido para mostrarse.		
	Actor	Sistema
		Muestra el mensaje "Tiempo de espera agotado".
Relaciones	CU Incluidos	<u>Ver CU: Consumir servicio de traducción</u>
	CU Extendidos	
Requisitos funcionales	no	

Anexos

Asuntos pendientes	
--------------------	--

Descripción del CUS “Consumir servicio de traducción”

Objetivo	Interactuar con el motor de traducción automática Apertium para realizar la traducción.	
Actores	Usuario, motor de traducción automática Apertium	
Resumen	El caso de uso se inicia cuando la aplicación interactúa con el motor de traducción automática Apertium, el cual realiza la traducción solicitada y devuelve el texto traducido. Además, durante la ejecución del caso de uso se registra la información generada durante el proceso.	
Complejidad	Baja	
Prioridad	Media	
Precondiciones	Se ha realizado el proceso de gestión de la información a traducir.	
Postcondiciones	Se devolvió el texto o la página web traducida.	
Flujo de eventos		
Flujo básico Consumir servicio de traducción		
	Actor	Sistema
1.		Envía una solicitud de traducción con los parámetros: <ul style="list-style-type: none">• Idioma origen.• Idioma de salida.• Dirección local del fichero original.• Dirección local para guardar el fichero traducido.• Formato del fichero a traducir.
2.		Registra información relacionada con la traducción: <ul style="list-style-type: none">• Fecha en que se está realizando la traducción.• Dirección IP desde donde está

Anexos

		<p>accediendo el usuario a la aplicación.</p> <ul style="list-style-type: none"> Tipo de navegador usado por el usuario para acceder a la aplicación.
3.	Realiza la traducción solicitada.	
4.		Obtiene el fichero traducido.
5.		Termina el CU.
Flujos alternos		
	Actor	Sistema
Relaciones	CU Incluidos	
	CU Extendidos	
Requisitos funcionales	no	
Asuntos pendientes		

Descripción del CUS “Registrar criterio de calidad”

Objetivo	Registrar el criterio de calidad emitido por los usuarios sobre las traducciones realizadas.
Actores	Usuario
Resumen	El caso de uso se inicia cuando el sistema muestra un texto o página web traducida y el usuario selecciona uno de los criterios de calidad disponibles. Dicha opinión es registrada en la base de datos de la aplicación.
Complejidad	Media
Prioridad	Baja
Precondiciones	La aplicación ha mostrado un texto o una página web traducida al idioma deseado.
Postcondiciones	Se registró el criterio del usuario acerca de la calidad de la traducción.
Flujo de eventos	

Anexos

Flujo básico registrar criterio de calidad		
	Actor	Sistema
1.		Muestra la opción para elegir un criterio de calidad: <ul style="list-style-type: none"> • Adecuada. Ver Sección 1: "Adecuada". • Incorrecta. Ver Sección 2: "Incorrecta u Ofensiva". • Ofensiva. Ver Sección 2: "Incorrecta u Ofensiva".
2.		Termina el CU.
Flujo alterno		
1 ^{ra} El usuario no selecciona ninguna de las opciones.		
	Actor	Sistema
1.		Espera por una acción del usuario.
2.		
Sección 1: "Adecuada"		
Flujo básico		
	Actor	Sistema
1.	Selecciona la opción "Adecuada".	
2.		Registra: <ul style="list-style-type: none"> • Criterio de calidad seleccionado. • Fecha de traducción. En caso de que el criterio sea sobre un texto, registra además: <ul style="list-style-type: none"> • Texto original. • Texto traducido.
3.		Termina el CU.
Flujo alterno		
1 ^{ro} . Si el criterio emitido por el usuario es con respecto a una página web traducida.		
	Actor	Sistema
2.1		Registra además:

Anexos

		<ul style="list-style-type: none"> Dirección URL de la página web original.
Sección 2: "Incorrecta u Ofensiva"		
Flujo básico		
	Actor	Sistema
1.	Selecciona una de las opciones: <ul style="list-style-type: none"> Incorrecta Ofensiva 	
2.		Muestra campo de entrada de texto para introducir comentarios.
3.	Introduce comentarios que justifican su elección con respecto al criterio de calidad y pulsa el botón "Aceptar".	
4.		Registra: <ul style="list-style-type: none"> Criterio de calidad seleccionado. Comentario realizado. Fecha de traducción. En caso de que el criterio sea sobre un texto, registra además: <ul style="list-style-type: none"> Texto original. Texto traducido.
Flujos alternos		
1º. Si el criterio emitido por el usuario es con respecto a una página web traducida.		
	Actor	Sistema
4.1		Registra además: <ul style="list-style-type: none"> Dirección URL de la página web original.
2º. El usuario no desea introducir ningún comentario.		
	Actor	Sistema
3.1	Pulsa el botón "Aceptar".	
		Registra: <ul style="list-style-type: none"> Criterio de calidad seleccionado.

Anexos

		<ul style="list-style-type: none"> Fecha de traducción. <p>En caso de que el criterio sea sobre un texto, registra además:</p> <ul style="list-style-type: none"> Texto original. Texto traducido.
	Actor	Sistema
3.1.1		<p>Si el criterio es sobre una página web, registra además:</p> <ul style="list-style-type: none"> Dirección URL de la página web original.
Relaciones	CU Incluidos	
	CU Extendidos	<p><u>Ver CU: Gestionar traducción de textos</u></p> <p><u>Ver CU: Permitir introducción de URL</u></p> <p><u>Ver CU: Permitir traducción de páginas web indexadas</u></p>
Requisitos funcionales	no	
Asuntos pendientes		

Anexo # 4: Diagrama de clases del diseño

El diagrama de clases del caso de uso “**Permitir traducción de páginas web indexadas**” está compuesto por una clase cliente CP_Principal, la cual envía una petición a la clase controladora Traductor; posteriormente, esta procesa la petición del usuario y, a partir de la respuesta arrojada, el controlador renderiza una vista según el motor de plantillas Twig, construyendo la clase CP_PaginaTraducida a partir de la plantilla utilizada. En esta nueva interfaz se muestra la página web traducida, en caso de descargar la página original correctamente. Además, el controlador interactúa con el repositorio de Doctrine con el objetivo de guardar los datos de la traducción realizada en la entidad CE_Registro de la base de datos de la aplicación.

Anexos

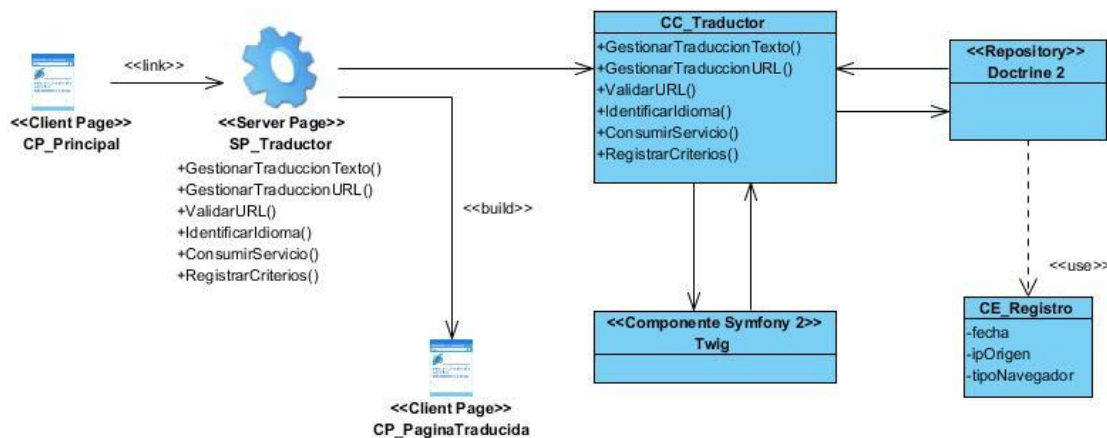


Diagrama de clases del diseño con estereotipos web del CUS "Permitir traducción de páginas web indexadas"

El siguiente diagrama presenta las relaciones entre las diferentes clases que componen el caso de uso "**Registrar criterio de calidad**". Dentro de las clases que lo integran existe una clase cliente `CP_Traducir`, en el que se muestran varios elementos. Uno de ellos es el formulario `Form_TextoTraducido`, donde el usuario selecciona uno de los criterios disponibles, considerando la calidad de la traducción realizada. También se muestra una clase JavaScript `Script_Criterio`, cuya función es mostrar el formulario `Form_Comentario`, si el criterio seleccionado fue "Incorrecta" u "Ofensiva", con el objetivo de que el usuario justifique su elección.

Este proceso es similar una vez que el sistema muestra una página web traducida, con el fin de que el usuario exponga sus criterios con respecto a la calidad de la traducción de dicha página. La información de los formularios es enviada a la clase controladora Traductor, la cual hace uso del modelo, a través del repositorio de Doctrine, para interactuar con la base de datos y guardar la información de la traducción realizada en la entidad `CE_CriterioTexto` o en la entidad `CE_CriterioPagina`, según la naturaleza de la información.

Anexos

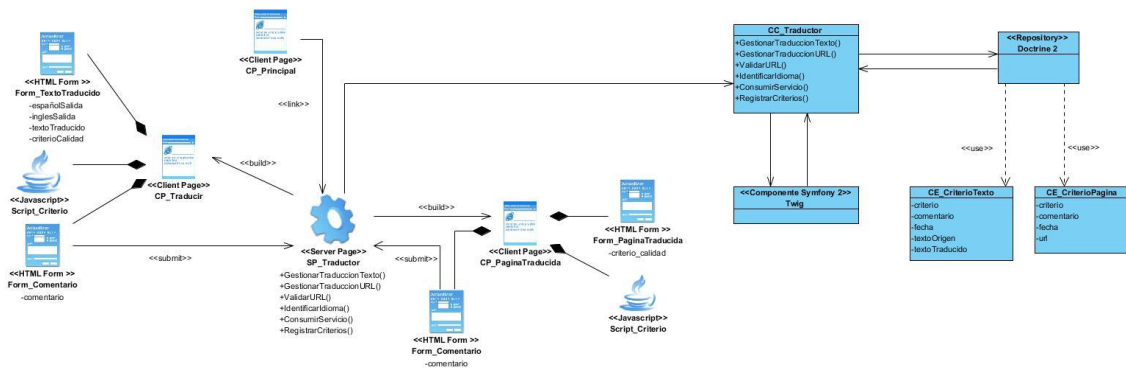


Diagrama de clases del diseño con estereotipos web del CUS “Registrar criterio de calidad”

Anexo # 5: Diagrama de secuencia

A continuación se muestra el diagrama de secuencia del caso de uso “Permitir introducción de dirección URL” para cuando el usuario introduce la dirección URL que desea traducir en la interfaz “PaginaTraducida”.

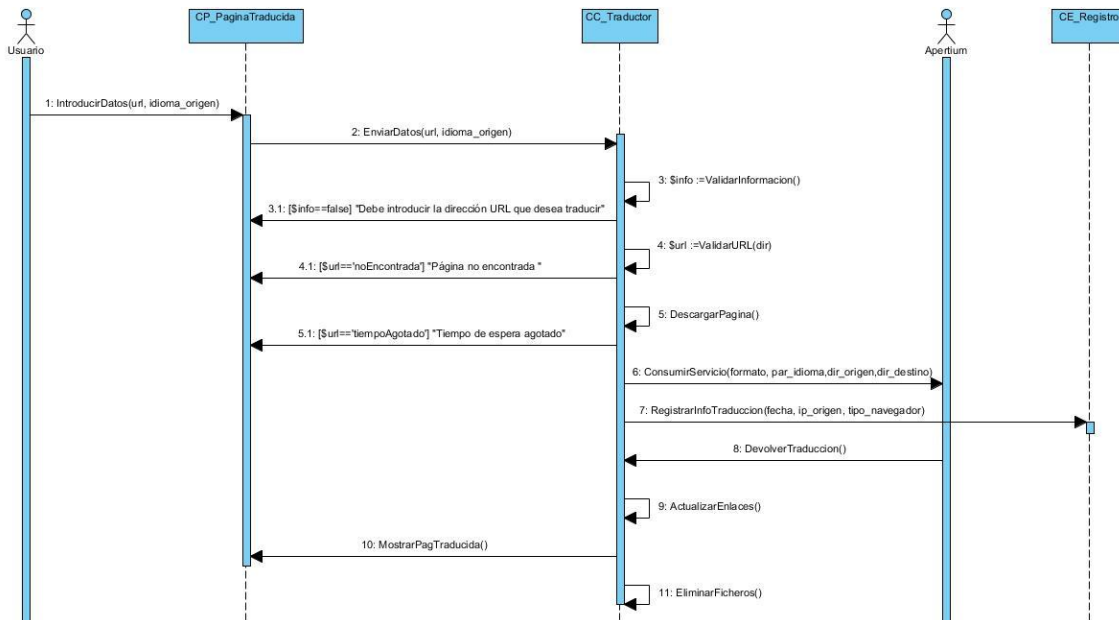


Diagrama de secuencia del CUS “Permitir introducción de dirección URL” desde la interfaz “PaginaTraducida”

Anexos

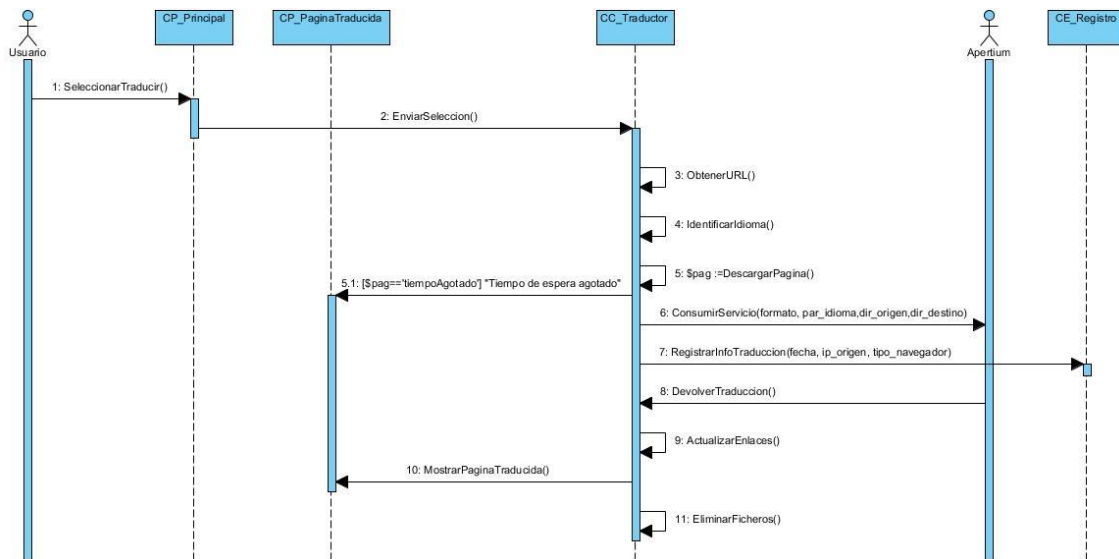


Diagrama de secuencia del CUS “Permitir traducción de páginas web indexadas”

Nota: El CUS “Registrar criterio de calidad” puede iniciarse desde la interfaz “Traducir” o desde la interfaz “PaginaTraducida”, generada al traducir una página web indexada por el buscador. En ambos casos el procesamiento del sistema es el mismo una vez que el usuario seleccione cualquiera de los criterios disponibles.

Para la realización del diagrama de secuencia correspondiente a cada uno de los escenarios que conforman el caso de uso, hay que tener en cuenta ambas interfaces. Por tanto se realizaron dos diagramas de secuencia para cada escenario, teniendo en cuenta cada una de las interfaces antes mencionadas.

A continuación se muestran los diagramas de secuencia de cada uno de los escenarios correspondientes a este caso de uso para cuando se inicia desde la interfaz “Traducir”.

Anexos

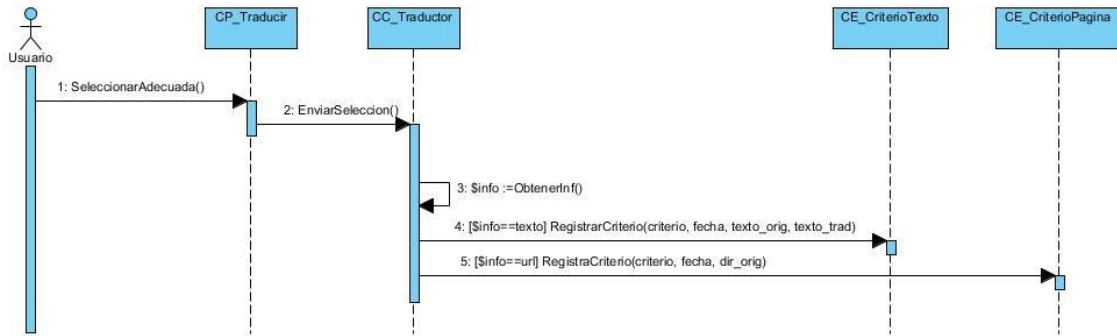


Diagrama de secuencia del escenario "Adecuada" del CUS "Registrar criterio de calidad" para la interfaz "Traducir"

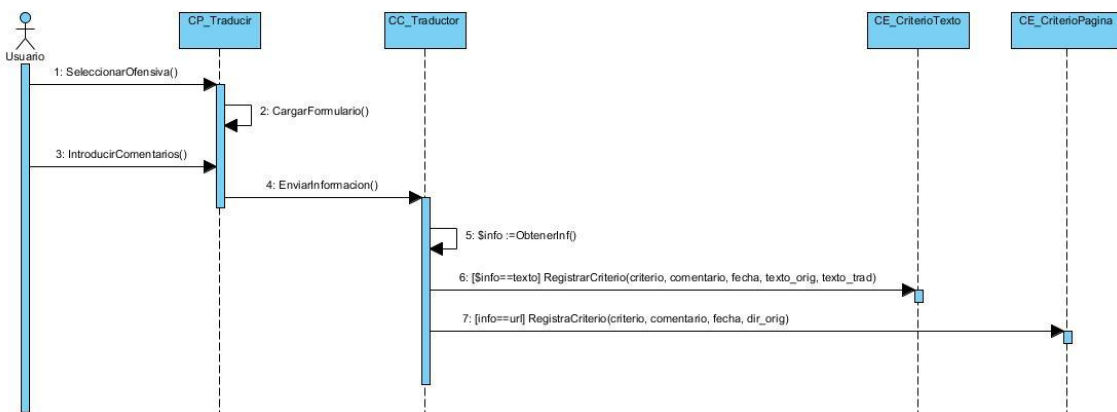


Diagrama de secuencia del escenario "Incorrecta u Ofensiva" del CUS "Registrar criterio de calidad" para la interfaz "Traducir"

A continuación se muestran los diagramas de secuencia de cada uno de los escenarios correspondientes a este caso de uso para cuando se inicia desde la interfaz "Pagina Traducida".

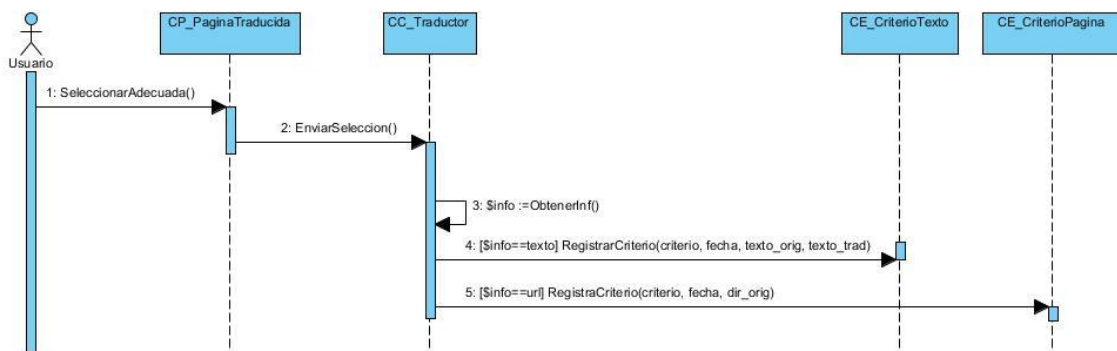


Diagrama de secuencia del escenario "Adecuada" del CUS "Registrar criterio de calidad" para la interfaz "Pagina Traducida"

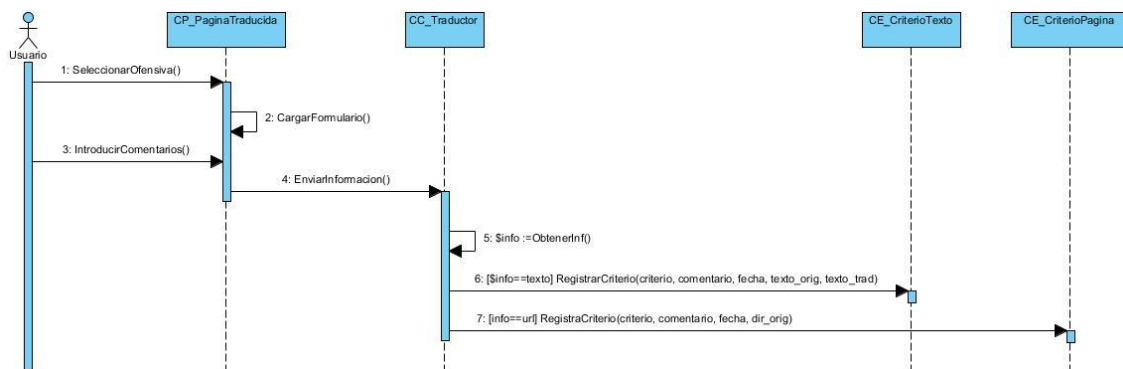


Diagrama de secuencia del escenario "Incorrecta u Ofensiva" del CUS "Registrar criterio de calidad" para la interfaz "PaginaTraducida"

Anexo # 6: Casos de prueba

Caso de prueba para el CUS "Permitir traducción de páginas web indexadas"

❖ Descripción general

El usuario pulsa la opción "Traducir" en una de las páginas web indexadas por el buscador. El módulo de traducción obtiene la dirección URL, la cual utiliza para gestionar posteriormente todo el proceso de traducción hasta mostrar la página web traducida.

❖ Condiciones de ejecución

El usuario ha realizado una búsqueda en la interfaz del motor de búsqueda ORION.

Escenario	Descripción	Variable 1 Criterio de calidad	Respuesta del sistema	Flujo central
EC 1.1 Traducir una página web indexada.	El usuario selecciona una de las páginas web indexadas por el buscador, después de realizada una búsqueda.	V "http://firefoxmania.uci.cu/"	Realiza la traducción de la página seleccionada, mostrándola en la interfaz correspondiente.	1. Seleccionar la opción "Traducir" en una de las páginas web indexadas por el buscador.

Anexos

	El usuario selecciona una de las páginas web indexadas por el buscador, la cual supera el tiempo de descarga establecido.	I	Muestra el mensaje de error: "Tiempo de espera agotado".	
--	---	---	--	--

Caso de prueba para el CUS "Registrar criterio de calidad"

❖ Descripción general

El usuario selecciona uno de los criterios de calidad disponibles, según su opinión respecto a la calidad de la traducción realizada. Si el criterio escogido es "Inadecuado" o "Malo", el usuario puede introducir comentarios que justifiquen dicha elección.

❖ Condiciones de ejecución

La aplicación ha mostrado un texto o una página web traducida al idioma deseado.

Escenario	Descripción	Variable 1 Criterio de calidad	Variable 2 Comentarios	Respuesta del sistema	Flujo central
EC 1.1 Elegir un criterio de calidad.	El usuario selecciona uno de los criterios de calidad que se muestran.	V "Adecuada"	<i>[Para este escenario la variable 2 no existe]</i>	Registra los datos de la traducción, además del criterio de calidad seleccionado.	1. Seleccionar el criterio de calidad "Adecuada".
		V "Incorrecta"	N/A		

Anexos

EC	1.2	El usuario selecciona el criterio de calidad "Incorrecta" u "Ofensiva", e introduce comentarios que justifican dicha elección.	V "Ofensiva"	V "Considero que la traducción es ofensiva porque..."	Registra los datos de la traducción, el criterio de calidad seleccionado, así como los comentarios realizados.	1. Seleccionar el criterio de calidad "Incorrecta" u "Ofensiva". 2. Introducir comentarios.
----	-----	--	-----------------	--	--	--

Anexo # 7: No conformidades detectadas

Funcionalidades ausentes

- ✓ Registrar la información de la traducción realizada.

Funcionalidades incorrectas

- ✓ No se guardan correctamente todos los datos al seleccionar un criterio de calidad y realizar un comentario.
- ✓ En la interfaz que se muestra al traducir una página web, no se muestra el formulario para insertar la dirección URL de otra página que se desee traducir.

Errores de interfaz

- ✓ Para la traducción de textos no se muestra el idioma inglés seleccionado, por defecto, como idioma de salida.

Validación de datos

- ✓ Al insertar una dirección URL incorrecta no muestra el mensaje "Página no encontrada".
- ✓ Si el usuario no inserta el texto que desea traducir y pulsa la opción "Traducir", no muestra un mensaje "Debe introducir el texto que desea traducir".
- ✓ Cuando una página demora cargando más del tiempo establecido, no se muestra el mensaje "Tiempo de espera agotado".

Anexos
