Universidad de las Ciencias Informáticas

Facultad 6

MÓDULO DE MINERÍA DE TEXTO PARA EL RSERVER 2.0

Trabajo de diploma para optar por el título de

Ingeniero en Ciencias Informáticas.

Autor:

Raúl Herrera Domínguez.

Tutor:

Ing. Rachíd Alí Grave de Peralta.

"Año 54 del Triunfo de la Revolución"

Junio 2012

La Habana, Cuba

"La posibilidad de realizar un sueño es lo que hace que la vida sea interesante."

Paulo Coelho

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Facultad 6 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

_____	_____
Raúl Herrera Domínguez.	Ing. Rachíd Alí Grave de Peralta.
Autor	Tutor

DATOS DE CONTACTO

AUTOR:

Raúl Herrera Domínguez .

Universidad de las Ciencias Informáticas,

Ciudad de la Habana, Cuba

E-mail: rhdominguez@estudiantes.uci.cu.

TUTOR:

Ing. Rachíd Alí Grave de Peralta.

Universidad de las Ciencias Informáticas,

Ciudad de la Habana, Cuba

E-mail: rali@uci.cu.

AGRADECIMIENTOS

A todos mis amigos, a Silvio y a Margarita por aguantarme y cargar conmigo durante estos 5 años (hermano yo se que no es fácil pero te quiero), gracias por enseñarme a confiar en un amigo y por demostrarme que los amigo de verdad están en las buenas y las malas y aunque se insultan, se quieren (así que cada ves que te yo te insulte mano, no es mas que cariño, no te preocupes), a mi amigo Alejandro ¨Yariny¨ por compartir conmigo tantos momentos, a Betty por aguantarme y soportarme todas mis malacrianzas con una sonrisa (lo malo no es lo que has aguantado, sino lo que te falta), a mis amigos de la casa, a Raúl, a Marquito ¨el militar¨, a Magdiel ¨el enamorado¨, a Ramón, a Noel, a Osmel, a Omarito, a Julito, a mi amigo Fox , a ¨Convell¨, a Héctor y a todos (que son muchos y no quiero dejar a nadie fuera), a todos los que han estado cada día ahí. Quiero que sepan que los quiero mucho y que nunca los voy a olvidar. A mi tutor por enseñarme a ¨aprender¨ por mí solo para lograr ser un verdadero profesional.

DEDICATORIA

Para mi mami querida, y mi eterna novia que con dedicación y paciencia me ha dedicado cada minuto de su vida, enseñándome en cada uno de ellos todo lo que necesito para vivir y ser mejor cada día. A mi segunda mamá que aunque su salud no la deja estar en este momento especial de mi vida ha sabido en todo momento mostrarme el camino correcto y enseñarme como ser mejor. A mi abuelo Roberto que ha sido más que un padre para mí, enseñándome como proceder en cada momento de mi vida y conduciéndome con su sabiduría por el mejor camino. A Rubén y a Robe por ser mi guía y mi inspiración para lograr cada cosa que quiero en mi vida, a Yuni por enseñarme parte de lo que se y por estar ahí cuando lo necesito y a toda mi familia. Doy gracias a Dios o al que me puso en tu vientre mamá por darme esta familia tan grande, tan unida y tan especial. Los quiero mucho a todos.

RESUMEN

La *minería de textos* es una técnica computacional que tiene como principal objetivo la búsqueda de elementos de interés en grandes colecciones de documentos no estructurados, dicho propósito se ha convertido en un reto teniendo en cuenta que aproximadamente un ochenta porciento de la información de las organizaciones están almacenadas en forma textual no estructurada. En consecuencia a lo anterior han surgido herramientas computacionales capaces de organizar y buscar la información de una manera eficiente. Se destaca la librería (TM, por sus siglas en inglés Text Mining) escrita en el lenguaje de programación R y considerada una de las herramientas cimeras en el análisis de textos. Por otro lado, las tecnologías web son cada vez más usadas en el ámbito empresarial y económico pero los lenguajes de programación que la soportan no son recomendables para hacer análisis matemáticos precisos. En el presente trabajo se propone el desarrollo de un módulo basado en la librería TM, capaz de brindar un conjunto de funciones de minería de texto integrables en proyectos web, específicamente escritos en el lenguaje de programación PHP. Uno de los principales resultados del trabajo es la incorporación al "Servidor de análisis estadísticos Rserver en su versión 2.0". Finalmente se validó el módulo mediante un conjunto de pruebas exploratorias y se conformó la documentación necesaria para el uso del mismo.

PALABRAS CLAVES: Módulo, TM, Minería de texto, Lenguaje de programación R.

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

Introducción

Desde los inicios de la humanidad, cuando el hombre empezaba a guardar información. Comenzó a hacerlo para que fuera usada por las generaciones futuras, con el cursar de los años dichas generaciones han realizado y sumado nuevos aportes al saber universal al usar nuevas técnicas para el guardado de la información, ya que a medida que avanza el desarrollo tecnológico aumenta la cantidad de información digitalizada.

Actualmente, se almacenan doscientos noventa y cinco trillones de bytes, lo que equivale a trescientas quince veces más información que el número de granos de arena que se estima hay en la Tierra. El noventa y cuatro porciento de la información que dispone la humanidad ha sido digitalizada. En menos de una década ha sucedido esta transición [1]. Uno de los retos consiste, precisamente, en encontrar segmentos específicos de dicha información. Debido a lo anterior se hacen necesarias herramientas computacionales capaces de organizar y buscar información eficientemente. Dicho problema se puede resolver mediante la minería de texto.

La minería de textos es una tecnología emergente cuyo objeto es la búsqueda de elementos de interés en grandes colecciones de documentos no estructurados. Se estima que aproximadamente un ochenta por ciento de la información de las organizaciones está almacenada en forma textual no estructurada: informes, correos, actas de reuniones, entre otros. La minería de textos opera sobre bases de datos textuales no estructuradas con el objetivo de detectar patrones no triviales e incluso información sobre el conocimiento almacenado en las mismas. Los sistemas de minería de textos pueden ayudar en la categorización de la información existente en una organización, en el filtrado de información por ejemplo de correos, en la detección de información similar o relacionada con otra existente, o para eliminar información duplicada [2]. Otro aspecto en el que las tecnologías de minería de textos encuentran una prometedora área para su aplicación es el de la web semántica, un ejemplo es el proyecto Lucene que implementa

motores de búsqueda de texto de alto rendimiento para la web realizando una búsqueda semántica, permitiendo así indexar cualquier archivo de texto, siempre que sea posible extraer información textual del mismo. Este nuevo modelo de Internet pretende construir toda una estructura de metadatos(información sobre la estructura y significado de los datos almacenado) e incluirlos en los documentos de forma que sean navegables, identificables y "comprensibles" por las máquinas.

En la actualidad existen herramientas que realizan minería de texto, entre ellas se encuentran: TextQuest y WordSmith, las cuales permiten una realización de análisis estadísticos desde la web.

Se destaca entre estas herramientas TM, un marco de trabajo para aplicaciones desarrolladas con el lenguaje de programación R [3]. TM es sumamente potente y contiene un conjunto de métodos para la importación de datos y el trabajo con información textual no estructurada.

La Universidad de las Ciencias Informáticas representa la avanzada del país en el desarrollo de la industria cubana del software. Cuenta con centros de desarrollo como el Centro de Tecnología de Gestión de Datos (DATEC), que tiene como objetivo principal crear bienes y servicios informáticos relacionados con la gestión de datos, este centro está compuesto por líneas de producción dentro de ellas se encuentra la línea de soluciones integrales. Esta última desarrolla componentes que serán reutilizados en su conjunto para formar productos; dichos productos como por ejemplo la Plataforma de Ayuda para la toma de decisiones de sistemas integrales (Patdsi) proveen servicios para la ayuda en la toma de decisiones para así contribuir a la solución de diversas disyuntivas. Dentro de este centro se encuentra en desarrollo un Servidor de Análisis Estadístico llamado RServer en su versión 2.0 que tiene como objetivo fundamental garantizar el análisis estadístico en arquitecturas web, el mismo no presenta en la actualidad un módulo con la capacidad de procesar rápidamente grandes cantidades de

datos textuales y la posibilidad de automatizar las laboriosas tareas de rutina sobre los textos.

En función de lo antes expuesto se identifica el **Problema Científico:** ¿Cómo realizar análisis de minería de texto en el Rserver 2.0?

Lo que determina como **Objeto de Estudio:** Aplicaciones para el análisis de la minería de texto.

Lo que precisa como **Campo de Acción**: Herramientas para el análisis de minería de texto.

Persiguiendo con ello el **Objetivo general**: Desarrollar el módulo de minería de texto para el Rserver. 2.0

Para su consecución se han planteado los siguientes **Objetivos específicos:**

- ✓ Diseño del módulo de minería de Texto para el RServer 2.0.
- ✓ Implementar las funcionalidades.
- ✓ Realizar pruebas que demuestren el correcto funcionamiento del módulo.

Para la realización de los objetivos propuestos se plantean las siguientes **Tareas de investigación:**

- ✓ Revisión de las aplicaciones de análisis estadísticos existentes para comprender el funcionamiento de las mismas.
- ✓ Definición de los requisitos funcionales y no funcionales para el correcto funcionamiento del módulo.
- ✓ Investigación de las herramientas para el diseño e implementación del software.
- ✓ Selección de patrones a emplear para el desarrollo de la aplicación.
- ✓ Realización del diseño de clases para facilitar la implementación de la aplicación.
- ✓ Estudio de los marcos de trabajo a utilizar para la implementación.

- ✓ Implementación del diseño realizado para satisfacer las funcionalidades definidas.
- ✓ Diseño de casos de pruebas basados en casos de uso para aplicarlos al módulo implementado.
- ✓ Realización de pruebas de caja negra para validar el correcto funcionamiento de la solución propuesta.

Para obtener como **Posibles resultados:**

- ✓ Versión funcional del módulo.
- ✓ Documentación de acuerdo con los lineamientos mínimos de calidad establecidos por la universidad y los artefactos generados a partir de la metodología empleada.

Estructura del Trabajo de Diploma

El Trabajo de Diploma quedó estructurado en tres capítulos.

Capítulo 1: Fundamento teórico:

Estudio del arte de la arquitectura de software y las herramientas que realizan minería de texto que existen en el mundo. Se realiza un análisis crítico y valorativo de los últimos avances en el tema. Se propone el ambiente de desarrollo y las tecnologías que serán usadas para el desarrollo del módulo evaluando las herramientas más óptimas para ello.

Capítulo 2: Análisis y diseño del sistema:

En este capítulo se presenta una descripción detallada sobre las características básicas que presenta la aplicación, en él se identifican las clases del dominio, los requisitos funcionales y no funcionales presentes, así como la identificación de actores que conforman el desarrollo. Los diagramas de casos de uso del sistema y una explicación detallada de los mismos. Los patrones de diseño usados, el modelo de diseño que recoge una definición de las clases del diseño y los diagramas de interacción que

compone cada modelo y la vista de despliegue donde se muestra el Hardware que se usará en la conformación del módulo.

Capítulo 3: Implementación y prueba:

En este capítulo se presentan los diagramas de componentes necesarios para la implementación total del módulo, presentando los diagramas de componentes que intervienen en él y las pruebas realizadas.

CAPÍTULO1. FUNDAMENTO TEÓRICO

Introducción

Este capítulo pretende brindar una descripción sobre los aspectos fundamentales que recoge la minería de textos, se encontrarán además datos de herramientas de minería de textos existentes dentro de Cuba y el mundo y se muestran las tecnologías que se usarán para el desarrollo del módulo, así como la demostración de su uso.

1.1- Estudio de Herramientas

Herramientas de Minería de Texto

Las herramientas de minería de texto se enfocan en descubrir, a partir de cantidades de texto, el conocimiento que no está literalmente escrito en cualquiera de los documentos. Se suele confundir la minería de textos con la minería de datos, pero se diferencian en que en la minería de datos la información se obtiene normalmente de bases de datos, en la que la información está estructurada, lo cual hace más sencilla la extracción de la información, contrario a lo que ocurre en la minería de textos donde la información se encuentra en formato no estructurado ya sea en bases de datos o en grupos de documentos.

En el mundo existen varias aplicaciones informáticas que realizan análisis de texto y minería de texto. A continuación se precisan las características de algunas de ellas.

1.1.1- TextAnalyst

Esta herramienta contiene diversas funciones de análisis textual, para crear una “red semántica” del contenido del texto. Se pueden obtener los párrafos del texto que se encuentran ligados con cada uno de los nodos de la red semántica. El sistema, determina que conceptos (palabras o combinaciones de ellas) son las más importantes en el contexto del texto bajo estudio. Cada concepto se etiqueta como un nodo, y se le asigna un “peso semántico numérico” (equivalente a la probabilidad de dicho concepto con relación al texto). Así mismo, TextAnalyst determina los pesos de las relaciones entre conceptos individuales en el texto. El sistema puede leer texto en diversos tipos de formato: HTML, archivos del Word, texto plano, entre otros. Un elemento importante, es la facilidad de poder realizar un “resumen” del

texto bajo estudio. El resumen que se obtiene, describe bastante bien al texto completo, existe la posibilidad de solicitar un resumen de menor o mayor tamaño al que se genera. Esta opción, se considera muy interesante para la revisión de documentos en el área de Educación a Distancia, pues permitiría tener una aproximación, facilitando la lectura, al tener una gran recopilación de artículos para su revisión. Los resultados se pueden exportar, para ser utilizados posteriormente en Excel o aplicaciones de bases de datos [4]. Tiene como principal desventaja que solo realiza análisis para textos en inglés lo cual impide su aplicación para el desarrollo del módulo.

1.1.2- T-LAB

Es una herramienta de análisis de texto desarrollada en Italia, se considera muy poderosa. Utiliza técnicas de análisis estadístico de texto, minería de texto y análisis multivariado (análisis de correspondencias, análisis de grupos de documentos, entre otros). T-LAB permite la extracción, la comparación y el mapeo de los contenidos de diversos tipos de textos: transcripciones de discursos, libros, artículos, notas periodísticas, documentos de internet y respuestas a cuestionarios de preguntas abiertas. Los análisis del sistema permiten tres tipos de aplicaciones:

- ✓ Minería de texto, para buscar y extraer información significativa y clasificarla.
- ✓ Mapeo de texto, para explorar de manera gráfica las relaciones entre temas y palabras claves.
- ✓ Utiliza técnicas multivariadas de análisis de correspondencias y análisis de clúster.
- ✓ Análisis de contenido, para realizar investigaciones con “plantillas” construidas por el usuario. [4]

Su interfaz es muy fácil de utilizar y los textos a analizar pueden ser de varios tipos: artículos de periódicos, transcripciones de entrevistas y discursos, respuestas a las preguntas abiertas, documentos empresariales, textos legislativos, libros, entre otros tipos. [5]. Es utilizado por muchos investigadores y profesionales. Es un software costoso por el grupo de ventajas que presenta y no permite una integración con el RServer 2.0, impidiendo así su uso para el desarrollo del módulo.

1.1.3- Searchopia

Es una aplicación que realiza una búsqueda en los contenidos de “carpetas” completas (incluyendo subcarpetas), de acuerdo con palabras y frases que estipula el usuario en un determinado criterio de búsqueda. El sistema obtiene todos los artículos que versen sobre la búsqueda estipulada. Puede

realizar la búsqueda en diversos tipos de archivos de manera simultánea (html, .txt, .doc,). Los archivos que se obtienen pueden clasificarse por fecha, tamaño y tipo. Se considera una aplicación adecuada y de gran poder [4]. Este software no tiene una integrabilidad y una modularidad visible que cumpla con el propósito que se requiere por lo que se dificulta su uso.

1.1.4- twURL

Es una poderosa herramienta para buscar páginas en Internet con alta relevancia, de acuerdo con un criterio de búsqueda. El sistema busca, colecciona, analiza y selecciona páginas relevantes de acuerdo a la dirección URL. Es importante mencionar que realiza búsquedas relevantes de acuerdo con las direcciones URL, aunque no realiza un análisis estadístico del contenido, ni tampoco realiza minería del texto. Realiza una “minería” pero de las páginas. Se considera que se puede utilizar en un primer inicio, para recuperar páginas relevantes en internet y posteriormente, aplicarles a dichas páginas un análisis estadístico y de minería de texto con alguna de las herramientas mencionadas anteriormente [4]. Con el uso de este software no se cumpliría con el objetivo trazado ya que realiza minerías sobre páginas web.

Herramientas de análisis de texto y contenido

1.2.1- Herramientas WordSmith

Conjunto de programas para el análisis textual, compatibles con el sistema operativo Windows y presenta tres herramientas:

- ✓ WordList. Realiza análisis estadístico de frecuencias de un texto.
- ✓ Concord. Realiza el análisis de concordancia de las palabras de un texto
- ✓ KeyWords. Realiza búsqueda de palabras clave en un texto.

La opción estadística de frecuencias obtiene entre otras cosas:

- ✓ Lista por orden alfabético de las frecuencias de las palabras.
- ✓ Lista por orden de frecuencia de aparición, con cálculo de la frecuencia absoluta y relativa.
- ✓ Tratamiento estadístico: longitud media de palabras, de la frase y del párrafo; número de palabras según su número de letras, etc.
- ✓ Posibilidad de comparación entre listas.

La opción de concordancias (Concord), obtiene:

- ✓ Lista por orden alfabético de todas las apariciones en el texto de una determinada palabra seleccionada, acompañada del contexto que la precede y que la sigue (concordancia).
- ✓ Identificación automática de las palabras que aparecen conjuntamente un determinado número de veces: colocaciones, grupos (clusters) y estructuras (patterns).

Por su parte, la opción de KeyWords proporciona:

- ✓ Comparación entre una lista de palabras y la lista de palabras de un texto de referencia.
- ✓ Identificación automática de las palabras que aparecen conjuntamente un determinado número de veces.

Las herramientas WordSmith, se consideran poderosas, en comparación con otras herramientas similares de análisis estadístico de texto [4]. Es una herramienta privada, lo que no posibilita su fácil acceso, por otro lado, no presenta grandes ventajas para su uso con el sistema operativo Linux y no presenta una arquitectura idónea para la integración con el RServer 2.0.

1.2.2- Concordance

Es un programa para análisis de concordancia de archivos de texto. Puede soportar diferentes lenguajes, elaborar listas de palabras y concordancias. Está en disponibilidad de realizar análisis de concordancias de documentos en la Web (WebConcordance). Puede analizar diversos archivos de texto de manera simultánea. Es también similar a WordSmith, aunque tiene la ventaja de realizar análisis de concordancia, estando conectado a Internet [4]. Es una herramienta que no realiza análisis estadístico de frecuencia, que necesita varios días de prueba para validar su correcto funcionamiento y necesita mucho de la conexión a internet para la realización de todas sus funcionalidades.

1.2.3- Wudz

Es un programa para analizar documentos de texto y desplegar información estadística del texto (frecuencias de palabras). Puede analizar el equivalente de un libro de dos cientos cincuenta páginas. Es similar a las herramientas WordSmith, pero más limitado, no realiza análisis de concordancia. Pero es fácil de utilizar para elaborar frecuencias de palabras. Es de uso gratuito [4]. Es una herramienta de propiedad privada y no permite una integración con el RServer 2.0.

1.2.4- TextQuest

TextQuest se diseñó originalmente para ofrecer apoyo informático al análisis de contenido. Los modelos de búsqueda en un sistema de categorías pueden ser palabras, partes de estas, secuencias de palabras, secuencias de caracteres, partes de palabras, llamadas cadena de raíces de palabras. Son cadenas, hasta seis, que deben suceder dentro de la misma unidad de texto, se puede especificar su secuencia y su distancia. También podemos emplear comodines de búsqueda. Se puede además codificar interactivamente en pantalla patrones de búsqueda ambiguos como de negación, el proceso de codificación se puede controlar mediante varios archivos de registro:

- ✓ Archivo de unidades de texto con modelos de búsqueda ambiguos.
- ✓ Archivo con unidades de texto no codificadas.
- ✓ Archivo de unidades de texto con negaciones.
- ✓ Control total sobre el proceso de codificación.

TextQuest funciona con etiquetas de categoría que nos obligan a documentar con detalles nuestras categorías, esto permite un empleo más cómodo de estas etiquetas durante la codificación interactiva, para las etiquetas de variable en la configuración de los programas informáticos de análisis de datos estadísticos (Paquete estadístico para ciencias Sociales, por sus siglas en inglés SPSS o Excel), y en los archivos de registro.

La diferencia entre codificación automática e interactiva se mide con ICRC o coeficiente interactivo de confianza en la codificación. Adicionalmente la generación de configuraciones para SPSS esclarece el análisis estadístico. El listado de palabras con signos de no codificación contiene todas que no se usaron durante la codificación, que podemos emplear para extender el sistema existente de categorías y de uso especial mientras codificamos preguntas abiertas [4]. Esta herramienta es bastante completa, pero no permite la integración con el Rserver 2.0 al no tener una arquitectura permisible para esto.

1.3- Lenguaje de Programación PHP.

PHP es un lenguaje de código abierto muy popular. Es muy usado desde el lado del servidor para el desarrollo web. Puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica. Se selecciona PHP como lenguaje de programación para realizar este módulo por las siguientes razones: Los productos de DATEC están escritos en dicho lenguaje y con el objetivo de prever una futura integración con estos u otros sistemas desarrollados en este lenguaje se recomienda su uso. Por otra parte Symfony 2.0 obliga a utilizar la versión 5.3 de dicho lenguaje u otra versión superior

1.4-Lenguaje de Programación R.

R es el lenguaje de programación más utilizado por un número creciente de analistas de datos dentro de las empresas y las academias [6]. Es gratuito y posee un amplio abanico de herramientas estadísticas dentro de los que se encuentran los modelos lineales y no lineales, pruebas estadísticas, análisis de series temporales y algoritmos de clasificación y agrupamiento, lo que lo hace un lenguaje muy potente en lo que a análisis estadístico se refiere. También provee una librería llamada TM que dentro tiene un grupo de funcionalidades muy sofisticadas para el manejo de textos, esta librería permite el análisis minucioso de los textos para así poder realizar las operaciones que se desean sobre los mismos. Se selecciona R como lenguaje de programación para la realización de análisis sobre los textos por presentar ese grupo de ventajas antes mencionadas.

1.5-Framework de desarrollo.

Un espacio de trabajo, también conocido como framework simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Además, un framework proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. Por último, un framework facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas.

1.5.1-Symfony 2.0.

Symfony es un framework diseñado para optimizar el desarrollo de las aplicaciones web. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. Symfony 2.0 es 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 mejores prácticas y patrones de diseño para la web. [7].

En esta versión integra el uso de bundles (que no son más que paquetes que contendrán todo lo referente a una determinada aplicación) lo que permite la integración con otras aplicaciones que se realicen posteriormente usando la misma versión del framework, de ahí la necesidad del uso de Symfony en su versión 2.0 para el desarrollo del módulo.

1.6- Metodología de desarrollo de software

Una metodología de desarrollo no es más que una colección de documentación formal referente a los procesos que se llevarán a cabo para lograr el desarrollo total de determinado software, esto define las políticas y los procedimientos que intervienen en el desarrollo del software. Su objetivo es garantizar la eficacia y la eficiencia en el proceso de generación de software.

OpenUP (Open Unified Process) es una metodología ágil que aplica un enfoque iterativo e incremental dentro de su estructura del ciclo de vida y que puede adaptarse para desarrollar diversos tipos de proyectos. Tiene como principios, colaborar para sincronizar intereses y compartir conocimiento. Equilibrar las prioridades para maximizar el beneficio obtenido por los interesados en el proyecto. Propiciar el desarrollo evolutivo para obtener retroalimentación y mejoramiento continuo. Además, al centrarse en la arquitectura de forma temprana minimiza el riesgo y organiza el desarrollo. Trayendo como beneficios que permite disminuir las probabilidades de fracaso, permite detectar errores tempranos a través de un ciclo iterativo, evita la elaboración de documentación, diagramas e iteraciones innecesarios requeridos en la metodología RUP, por ser una metodología ágil tiene un enfoque centrado al cliente y con iteraciones cortas [8].

Se escoge OpenUp como metodología, no solo porque se aprovechan las ventajas que brinda sino fundamentalmente porque garantiza la eficacia y la eficiencia en el proceso de desarrollo del módulo.

1.7-Herramientas Case.

Las Herramientas de Software Asistidas por Ordenador (CASE, por sus siglas en inglés de Computer Aided Software Engineering) consisten en una o varias herramientas que permiten organizar y manejar cierta información de un proyecto informático.

Las herramientas CASE representan una forma que permite modelar los procesos de negocios de las empresas y además ayudan a desarrollar los Sistemas de Información. [9]

1.7.1- Visual Paradigm.

Visual Paradigm for UML es una herramienta que soporta el ciclo de vida completo en el desarrollo de software: análisis y desarrollos orientados a objetos, construcción, prueba y despliegue. Permite dibujar todo tipo de diagrama de clases, código inverso, generación de código a partir de diagramas y generar documentación [10]. Es una herramienta muy potente, con gran interoperabilidad, que soporta varias plataformas e ideal para el modelado de los diferentes diagramas necesarios para el desarrollo de la aplicación, lo que propició que se haya elegido esta herramienta para el desarrollo del software.

1.8-Gestores de Base de Datos.

Un Sistema Gestor de base de datos (SGBD) es un conjunto de programas que permiten crear y mantener una base de datos, asegurando su integridad, confidencialidad y seguridad. Por tanto, debe permitir:

- Definir una base de datos: especificar tipos, estructuras y restricciones de datos.

- Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD.

- Manipular la base de datos: realizar consultas, actualizarla, generar informes. [11]

Algunos de los SGBD más usados son Oracle, SQL Server de Microsoft, MySQL y PostgreSQL.

1.8.1-PostgreSQL.

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente está disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales. Utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto, y el sistema continuará funcionando. [12] Es potente para el trabajo también con lenguajes de programación para aplicaciones estadísticas como R a través de PL/R e ideal para el desarrollo del módulo.

1.8-NetBeans

NetBeans es un IDE (Integrated Development Environment) de código abierto y una plataforma de aplicaciones que permiten a los desarrolladores crear rápidamente aplicaciones web, escritorio y aplicaciones móviles utilizando la plataforma Java, así como JavaFX, PHP, Java Script y Ajax, Ruby y Ruby onRails, Groovy y Grails, y C/C++ [13]. Es un IDE gratuito e ideal para el trabajo con Symfony 2.0 en su versión 7.0.1 y resuelve las necesidades propias del proceso de desarrollo del módulo.

1.9- TM librería de R para minería de texto.

La minería de texto en R utiliza el marco proporcionado por el paquete TM. Donde se presentan los métodos para la importación de datos (data import), exportación de datos (data export), manipulación de documentos (corpus handling), pre-procesamiento (preprocessing), gestión de metadatos (meta data management) y la creación de las matrices de documento, centrándose en sus principales aspectos. Este paquete ofrece la funcionalidad de gestión de documentos de texto. Está diseñado en una forma modular para permitir una fácil integración de nuevos formatos de archivo, los lectores, las transformaciones y operaciones de filtrado. El paquete TM proporciona un fácil acceso a los mecanismos de pre-procesamiento y manipulación, tales como la eliminación de los espacios en blanco, producto, o la conversión entre formatos de archivo. Además de una arquitectura genérica de filtros que está disponible con el fin de filtrar los documentos para ciertos criterios, o realizar búsquedas de texto completo. El paquete es compatible con la exportación de un documento de colecciones a las matrices de documento plazo, que se utilizan con frecuencia en la literatura de la minería de textos. Este permite la integración

directa de avance de los métodos existentes para la clasificación y agrupación [3]. Con el uso de este paquete se garantiza que las aplicaciones con el propósito de realizar minería de textos cumplan con sus funciones.

1.10 Conclusiones parciales

Después de haber estudiado cuidadosamente se tomaron un grupo de decisiones importantes para el desarrollo del módulo:

- ✓ Como metodología de desarrollo se decidió utilizar OpenUp, metodología usada en el centro DATEC ya que esta propone procesos ágiles para el desarrollo del módulo.
- ✓ Como herramienta a utilizar para el modelado de los diferentes diagramas que se usarán para el desarrollo del módulo se seleccionó el Visual Paradigm una herramienta muy potente en este aspecto, que presenta una gran rapidez en la creación de los mismos y una gran integración entre sus componentes.
- ✓ Como Framework de desarrollo se seleccionó a Symfony en su versión 2.0, que es muy potente en lo que respecta al desarrollo de aplicaciones y permite grandemente la reutilización del código implementado.
- ✓ Como lenguaje para el desarrollo de la aplicación se seleccionó PHP en su versión 5.3 ya que es un lenguaje muy dinámico y en la versión antes expuesta permite la comunicación con Symfony en su versión 2.0.
- ✓ Como IDE se utilizará Netbeans en la versión 7.0.1.
- ✓ Como gestor de base de datos para garantizar la persistencia de los datos fue escogido PostgreSQL.

CAPÍTULO 2. ANÁLISIS Y DISEÑO DEL SISTEMA

2.1 Introducción

En el presente capítulo se describe todo lo que respecta al funcionamiento de la aplicación a través de diagramas que recogen el comportamiento de la misma desde sus inicios hasta que se culmina su desarrollo y despliegue. Para la correcta realización se han definido y desarrollado un grupo de artefactos que corresponden al flujo de trabajo de diseño. Se han definido un grupo de requisitos funcionales y no funcionales, se realizan diagramas de clases del diseño donde se muestra la estructura que tiene el sistema, se muestra el diagrama de casos de uso del sistema y se describen cada uno de los casos de uso que intervienen, especificándolos detalladamente para que tanto desarrolladores como clientes comprendan el funcionamiento de la aplicación. Se realizan los diagramas de secuencia para comprender el flujo de información entre cada una de las clases y actores, entre otras cuestiones importantes para el comienzo de la implementación del sistema.

2.2-Análisis del sistema

2.2.1-Requisitos Funcionales

Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir para realizar las funciones a las que fue destinado, en el caso del módulo de minería de texto se tienen los siguientes:

RF 1-Autenticar Usuario.
RF 2 - Adicionar Usuario.
RF 3 - Consultar Usuario.
RF 4 - Actualizar Usuario.
RF 5 - Eliminar Usuario.
RF 6-Registrar Usuario.
RF 7- Analizar Frecuencia de términos.
RF 8- Analizar Asociación y Correlación de Términos.
RF 9- Cambiar Propiedades de Documentos.
RF 10 - Realizar Diccionario.
RF 11-Guardar Análisis.
RF 12-Administrar Marco de Trabajo.

RF 13-Consultar Marco de Trabajo.

RF 14-Adicionar Archivos.

RF 15-Eliminar Archivos.

2.2.2-Requisitos no Funcionales

Los requisitos no funcionales son cualidades y características que el sistema debe tener para poder realizar un correcto funcionamiento.

Usabilidad

Tipo de usuario final

- ✓ Para trabajar con el módulo no necesariamente el usuario tiene que ser experto en la rama de la informática, solo basta con tener conocimientos básicos acerca de la misma, no siendo así con la minería de texto, puesto que tiene que tener conocimientos medios en esta rama para poder trabajar.

Tipo de Aplicación Informática

- ✓ El software es una herramienta web que va a contribuir en gran medida a la clasificación, interpretación y extrapolación de la información contenida dentro de documentos.

Finalidad

- ✓ Pretende realizar minería de texto a un cúmulo grande de documentos, esto contribuirá al aprovechamiento de la información por parte de los clientes finales.

Software

- ✓ PC Cliente: Instalado un explorador de Internet y conexión a la red.
- ✓ PC servidor: Sistema Operativo Linux, con R en la versión 2.13 y el marco de trabajo TM instalados, debe tener el servidor Apache en la versión 2.2 y php en la versión 5.3

Hardware

- ✓ PC Cliente: Al menos 32 mega bytes de RAM, una conexión a la red para la conexión con el módulo.
- ✓ PC Servidor: Luego del análisis de los requerimientos mínimos de hardware para la ejecución de Apache en la versión 2.2 y de R en la versión 2.13 se llegó a la conclusión de que debe contar con: Un protocolo TCP/IP, una memoria RAM de al menos 100 megabytes y al menos 100 MB de espacio libre en disco.

Disponibilidad

- ✓ El software debe estar disponible las 24 horas del día prestando servicio. En caso de fallo de corriente eléctrica u otra acción que afecte la integridad de sus funciones, el mismo no será responsable de este inconveniente.

Seguridad

- ✓ El cliente o administrador que desee usar la aplicación deberá ser autenticado previamente para así tener control sobre los usuarios que acceden a la misma.

2.2.3-Modelo de Dominio

El modelo de dominio es la representación de los conceptos que se van a manejar dentro del problema, dentro se tienen objetos, clases, asociaciones entre ellos y atributos, los que en su interacción explicarán como se maneja el problema y que estructura se tendrá en cuenta para la resolución. En la Figura 1 se muestra dicho modelo en el caso del módulo de minería de texto.

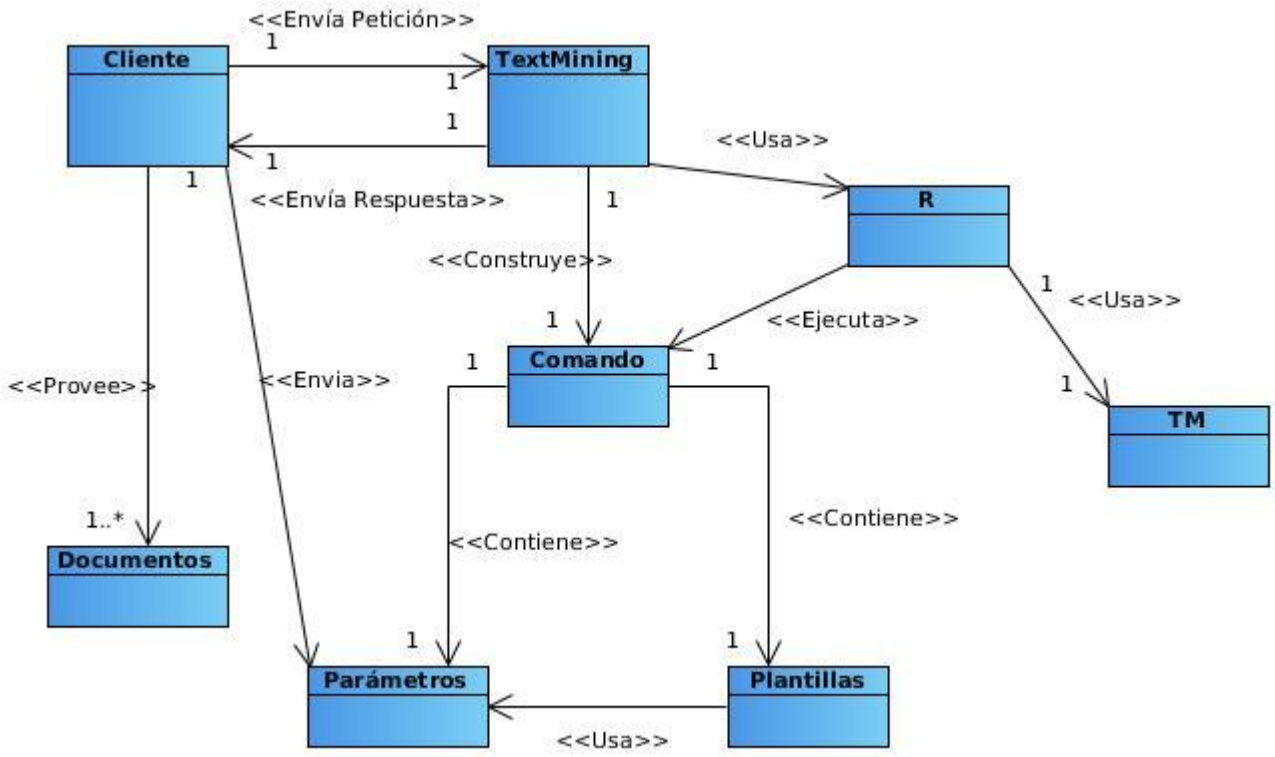


Figura 1. Modelo de Dominio.

Definición de las clases del modelo de dominio.

- ✓ **Cliente**

Persona o aplicación que hace uso del módulo, este realiza la petición, con la acción que desea realizar.

- ✓ **Text_Mining**

Sistema que permitirá el análisis y la minería de texto.

- ✓ **Documentos**

Artefactos ya sean en formatos específicos (txt, pdf, xml) de donde se extraerá la información para su posterior análisis.

- ✓ **Comandos**

Estructuras y métodos que se construirán en el módulo para garantizar el resultado final.

- ✓ **Parámetros**

Grupo de datos que provee el cliente para poder realizar las operaciones con los documentos en cuestión.

- ✓ **Plantilla**

Archivo donde se almacenan las funciones de R que luego se ejecutarán para devolver el resultado del análisis en cuestión.

- ✓ **TM**

Librería que usando el motor de análisis estadísticos de R realiza las operaciones con los textos.

- ✓ **R**

Motor de análisis estadístico utilizado en la aplicación para realizar los distintos tipos de análisis sobre los textos.

2.2.4-Modelo del sistema

Justificación de los actores del sistema

El actor del sistema es el usuario o sistema que interactuará directamente con el módulo, es el que se encarga de introducir los parámetros necesarios e intercambiar información con la misma, en la siguiente tabla se muestra una descripción detallada de cada uno de los actores así como su objetivo dentro del módulo.

Actor	Objetivo
Cliente	Actor humano que necesite la realización de minería de texto a un grupo de documentos predefinidos, es el responsable registrarse a sí

	mismo o registrar cualquier cliente, también realiza la actualización de su marco de trabajo y los análisis una vez autenticado.
Usuario	Actor humano o sistema, responsable de autenticarse a sí mismo.
Administrador	Actor humano responsable de la gestión de todos los usuarios que intervienen en el módulo.

Tabla 1. Justificación de actores

Diagrama de Casos de Uso del sistema

El diagrama de Casos de Uso del sistema es el encargado de la representación del o los actores que van a estar presentes en el sistema e interactuarán con los casos de uso o las funcionalidades que el sistema proveerá. Se muestran además las relaciones entre casos de uso ya sean de extensión, inclusión, asociación o generalización/especialización. Este diagrama es muy importante para que tanto clientes como desarrolladores tengan una idea clara del software a desarrollar.

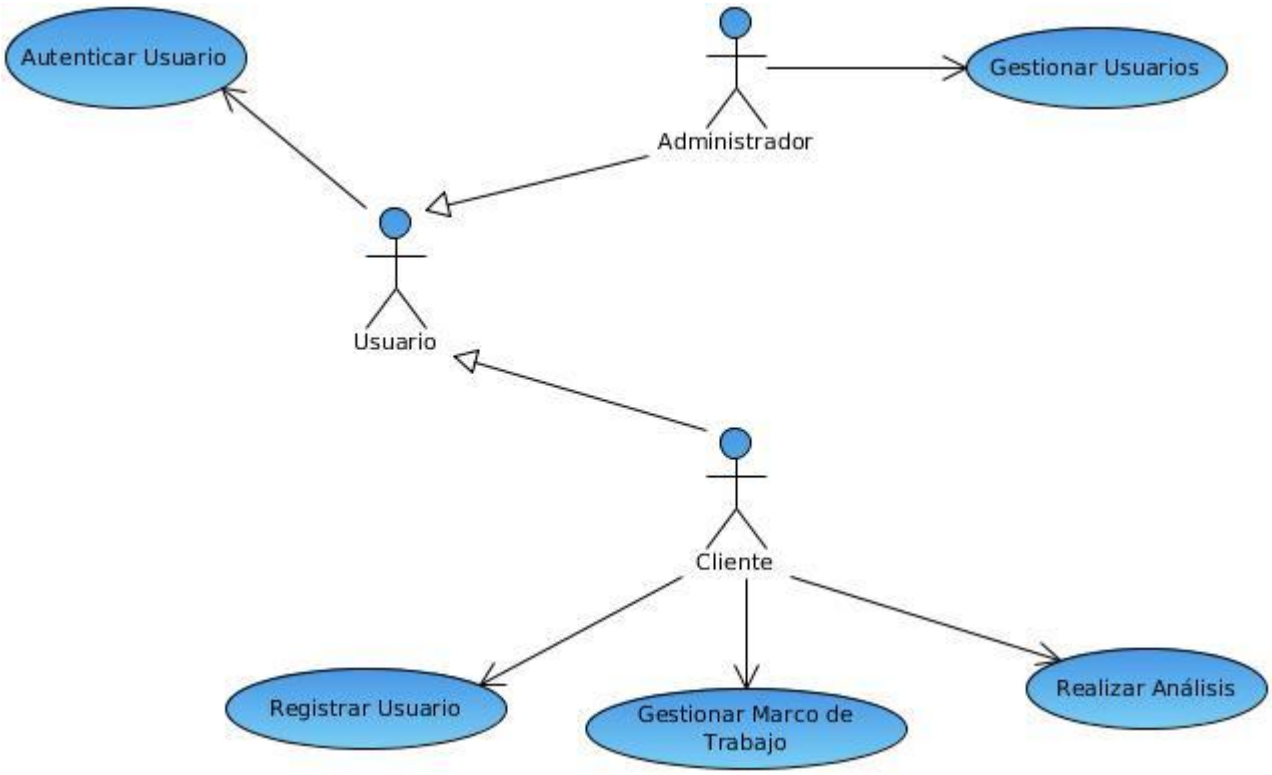


Figura 2.Diagrama de Casos de Uso del Sistema.

Aplicación de patrones de casos de uso

En el módulo se aplican una serie de patrones necesarios para su ejecución y mejor entendimiento como son el caso de los casos de uso ¨Realizar Análisis¨, ¨Gestionar Usuario¨ y ¨Administrar Marco de Trabajo¨. Estos patrones agrupan una serie de funcionalidades que se realizarán internamente a raíz de la llamada del patrón.

Listado de los Casos de Uso del sistema

- ✓ Gestionar Usuario
- ✓ Realizar Análisis
- ✓ Autenticar Usuario
- ✓ Registrar Usuario

- ✓ Administrar Marco de Trabajo

Descripción textual

La descripción textual de cada uno de los casos de uso que intervienen en la aplicación es muy importante para lograr que clientes y desarrolladores tengan una percepción detallada de lo que se quiere implementar.

Objetivo	Autenticar todos los usuarios que interactúen con la aplicación.
Actores	Usuario
Resumen	El caso de uso inicia cuando el usuario consulta la dirección web e introduce su nombre de usuario y contraseña respectiva a lo que el sistema responderá con la interfaz para elegir la carpeta a analizar en caso de ser correctos.
Complejidad	Media
Prioridad	Crítico
Precondiciones	Tiene que existir la conexión con la base de datos y la misma debe contener al menos un usuario con rol de administrador.
Pos condiciones	Se ejecuta una interfaz para que se escoja el directorio, o la pantalla de gestión de usuarios.
Flujo de eventos	
Flujo básico Autenticar Usuario	

	Actor	**Sistema**
1.	Introduce “Nombre de Usuario” y “Contraseña” y presiona el botón Aceptar.	1.1 Verifica usuario y contraseña. 1.2 Muestra una interfaz brindándole al

		usuario un menú con los directorios que se encuentran en su marco de trabajo en caso de contener documentos en él, de lo contrario el usuario autenticado deberá Administrar el marco de trabajo.
2	Selecciona la carpeta donde desea realizar el análisis.	2.1 El sistema envía los datos a la clase controladora. 2.2 Termina el caso de uso.
Flujos alternos		
Datos Incorrectos		
	Actor	**Sistema**
		1.1 a Muestra un mensaje diciendo que el usuario o contraseña son incorrectos.

Tabla 2. CU Autenticar Usuario.

Objetivo	Gestionar los usuarios que intervienen en la aplicación
Actores	Administrador
Resumen	El caso de uso inicia cuando el administrador se autentica, apareciendo una interfaz con los usuarios del sistema que le permitirá Adicionar, Actualizar y Eliminar todos los usuarios presentes en el sistema.

Complejidad	Alta	
Prioridad	Crítico	
Precondiciones	Debe existir conexión con la base de datos.	
Pos condiciones	Se ejecuta una interfaz que permitirá la gestión de los gsuarios.	
Flujo de eventos		
Flujo básico Gestionar Usuario		
	Actor	**Sistema**
1.	Introduce “Nombre de Usuario” y “Contraseña” y presiona el botón Aceptar.	1.1 Verifica los datos. 1.2 Muestra una interfaz brindándole un listado con todos los usuarios que están dentro de la base de datos. Lista también las opciones: • Editar Usuario (Ver sección 1) • Adicionar Usuario (Ver sección 2) • Eliminar Usuario (Ver sección 3)
Flujos alternos		
Datos Incorrectos		
	Actor	**Sistema**
		1.1 a Muestra un mensaje diciendo

		que el usuario y la contraseña son incorrectos.
Sección 1: Adicionar Usuario		
	Actor	**Sistema**
1.	Selecciona el botón añadir	1.1 Muestra una interfaz para que se introduzcan los campos correspondientes para ese usuario.
2.	Introduce los datos	2.1 Verifica datos. 2.2 Introduce el nuevo usuario en la base de datos 2.3 Termina el caso de uso
Flujos alternos		
Datos Incorrectos		
	Actor	**Sistema**
		2.1 a Muestra un mensaje diciendo que los datos son incorrectos.
Sección 2: Actualizar Usuario		
	Actor	**Sistema**
1	Selecciona el usuario que desea actualizar y link Editar en el usuario que desea	1.1 Muestra una interfaz para que se introduzcan los campos que desee modificar.

2	Introduce los nuevos datos.	2.1 Verifica datos. 2.2 Actualiza los nuevos datos del usuario seleccionado. 2.3 Termina el caso de uso
Flujos alternos		
Datos Incorrectos		
	Actor	**Sistema**
		2.1 a Muestra un mensaje diciendo que los datos son incorrectos.
Sección 3: Eliminar Usuario		
	Actor	**Sistema**
1	Selecciona el Usuario y la opción Eliminar	1.1 Elimina el usuario seleccionado de la base de datos 1.2 Termina el caso de uso.

Tabla 3. CU Gestionar usuario

Objetivo	Registrar nuevos usuarios en la base de datos
Actores	Cliente
Resumen	El caso de uso se inicia cuando el cliente selecciona la opción de registrar un nuevo usuario, mostrándose una interfaz para introducir los datos del nuevo usuario.
Complejidad	Media

Prioridad	Crítico	
Precondiciones	Debe existir conexión con la Base de Datos.	
Pos condiciones	Se ejecuta una interfaz donde podrá registrar nuevos usuarios	
Flujo de eventos		
Flujo básico Registrar Usuario		
	Actor	**Sistema**
1.	Introduce “Nombre de Usuario”, “Contraseña”, confirmación de contraseña y presiona el botón Aceptar.	1.1 Valida que no exista un usuario en la base de datos con el mismo nombre y lo inserta como nuevo usuario. 1.2 Termina el caso de uso.
Flujos alternos		
Datos Incorrectos		
	Actor	**Sistema**
		1.1 a Muestra un mensaje diciendo que el usuario ya se encuentra en la base de datos.

Tabla 4. CU Registrar usuario

Objetivo	Realizar los tipos de análisis con los que cuenta el módulo de minería de texto
Actores	Cliente

Resumen	El caso de uso se inicia cuando el cliente selecciona el tipo de análisis que desea realizar sobre los documentos que se encuentran en su marco de trabajo y el sistema muestra los resultados de los mismos.
Complejidad	Alta
Prioridad	Crítico
Precondiciones	El usuario debe estar autenticado.
Pos condiciones	Se ejecuta una interfaz para introducir los parámetros de cada análisis y luego se muestra el resultado.

Flujo de eventos

Flujo básico Realizar Análisis

	Actor	**Sistema**
		1.1 Muestra una interfaz brindándole un listado con todos tipos de análisis con los que cuenta el módulo. Permite realizar las siguientes operaciones: • Frecuencia de Términos (Ver sección 1) • Asociación y Correlación de Palabras (Ver sección 2) • Cambiar Propiedades (Ver sección 3) • Realizar Diccionario (Ver sección 4) • Guardar Análisis (Ver sección

		5)
Sección 1: Realizar análisis "Frecuencia de Términos"		
	Actor	**Sistema**
1.	Selecciona el tipo de análisis "Frecuencia de Términos"	1.1 Muestra una interfaz con los parámetros que el usuario debe introducir para realizar el tipo de análisis.
2.	Introduce los datos	2.1 Verifica datos. 2.2 Muestra una interfaz con el resultado del análisis. 2.3 Termina el caso de uso.
Flujos alternos		
Datos Incorrectos		
	Actor	**Sistema**
		2.1 a Muestra un mensaje diciendo que los datos son incorrectos.
Sección 2: Realizar análisis "Asociación y Correlación de Palabras"		
	Actor	**Sistema**
1	Selecciona el tipo de análisis "Asociación y Correlación de Palabras"	1.1 Muestra una interfaz con los parámetros que el usuario debe introducir para realizar el tipo de

		análisis.
2	Introduce los datos	2.1 Verifica datos. 2.2 Muestra una interfaz con el resultado del análisis. 2.3 Termina el caso de uso.
Flujos alternos		
Datos Incorrectos		
	Actor	**Sistema**
		2.1a Muestra un mensaje diciendo que los datos son incorrectos.
Sección 3: Realizar análisis "Cambiar Propiedades"		
	Actor	**Sistema**
1	Selecciona el tipo de análisis "Cambiar Propiedades"	1.1 Muestra una interfaz con los parámetros que el usuario debe introducir para realizar el tipo de análisis.
2	Introduce los datos	2.1 Verifica datos. 2.2 Muestra una interfaz con el resultado del análisis. 2.3 Termina el caso de uso.
	Flujos alternos	

	Datos Incorrectos	
	Actor	**Sistema**
		2.1 a Muestra un mensaje diciendo que los datos son incorrectos.
Sección 4: Realizar análisis "Realizar Diccionario"		
	Actor	**Sistema**
1	Selecciona el tipo de análisis "Realizar Diccionario"	1.1 Muestra una interfaz para que el usuario seleccione la cantidad de palabras que contendrá el diccionario.
2	Selecciona la cantidad de palabras	2.1 Muestra la interfaz donde el usuario introducirá cada una de las palabras
3	Introduce los datos	2.1 Verifica datos. 2.2 Muestra una interfaz con el resultado del análisis.
	Flujos alternos	
	Datos Incorrectos	
	Actor	**Sistema**
		2.1 a Muestra un mensaje diciendo que los datos son incorrectos

Sección 5: Guardar Análisis		
	Actor	**Sistema**
1	Selecciona la opción de Guardar el análisis.	1.1 Muestra una interfaz de descarga
2	Selecciona la carpeta donde quiere hacer la descarga del resultado y selecciona el botón aceptar	2.1 Realiza la descarga del documento con el resultado del análisis. 2.2 Termina el caso de uso.

Tabla 5. CU Realizar análisis

Objetivo	Administrar el marco de trabajo que tiene cada usuario.
Actores	Cliente
Resumen	El caso de uso se inicia cuando el cliente encontrándose dentro de su marco de trabajo, selecciona la opción administrar marco de trabajo, selecciona una carpeta para comenzar a administrar los archivos y selecciona un documento o grupo de documentos que desea analizar.
Complejidad	Media
Prioridad	Alta
Precondiciones	El usuario debe estar autenticado
Pos condiciones	Se muestra una interfaz con los archivos dentro de la carpeta seleccionada en su marco de trabajo
Flujo de eventos	
Flujo básico Administrar Marco de Trabajo	

	Actor	Sistema
1.	Selecciona la opción para realizar las distintas operaciones sobre su marco de trabajo.	1.1 Muestra una interfaz brindándole un listado con los directorios.
2.	Escoge el directorio para realizar operaciones.	2.1 Muestra interfaz con los archivos del directorio seleccionado. Permite realizar las siguientes operaciones: • Adicionar Archivo (Ver sección1) • Eliminar Archivo (Ver sección 2).
Sección 1: Adicionar Archivo		
	Actor	**Sistema**
1.	Selecciona el botón Examinar	1.1 Muestra una interfaz con los directorios para que el usuario busque el archivo que desea adicionar
2.	Selecciona el archivo y selecciona el botón Adicionar.	2.1 Adiciona el archivo al marco de trabajo. 2.2 Termina el caso de uso
Sección 2: Eliminar Usuario		
	Actor	**Sistema**
1	Selecciona el archivo y la opción Eliminar	1.1 Elimina el archivo de la carpeta seleccionada

Tabla 6. CU Actualizar marco de trabajo

2.3 Diseño del sistema

El diseño del sistema es el encargado de crear los modelos a raíz de los requisitos antes definidos y las descripciones de los casos de uso, con el objetivo de preparar el sistema para su posterior implementación. A través de diseño se modela la aplicación, el mismo debe soportar todos los requerimientos del sistema, para lo que se utilizan los diagramas de secuencia y diagramas de diseño.

2.3.1-Modelo de Diseño

Diagramas de clases del diseño

En los diagramas de clases del diseño se muestran las propiedades y las funcionalidades de cada clase en el lenguaje desarrollado y usando las tecnologías seleccionadas, se representa en cada uno de ellos las responsabilidades de cada clase y las relaciones entre las mismas. Estos diagramas se realizan para cada caso de uso. A partir de la figura 3 se pone de manifiesto diagrama de clases del diseño (DCD) para los casos de uso más importantes dentro del módulo.

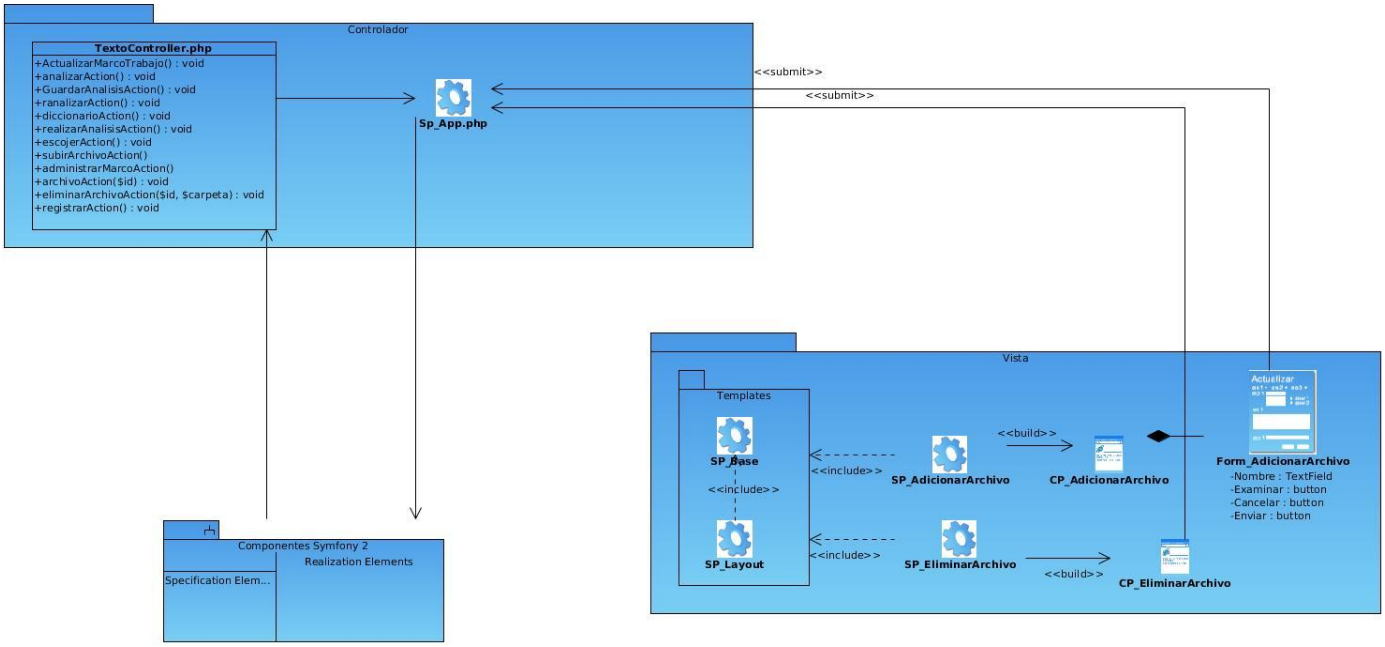


Figura 3.DCD Caso de Uso Administrar Marco de Trabajo

Descripción de clases en el caso de uso Administrar Marco de Trabajo

TextoController.php: Clase que se encarga de la gestión de los archivos que se tendrán dentro del marco de trabajo que tiene cada usuario, permite la subida, eliminación y consulta de cada uno de los archivos contenidos en la misma.

CP_AdicionarArchivo: Clase que hace la llamada al método de actualizar marco dentro de la clase controladora y permite la subida de archivos al marco de trabajo de usuario.

CP_Index: Clase que muestra cada uno de los archivos que se encuentran en las carpetas de cada Usuario.

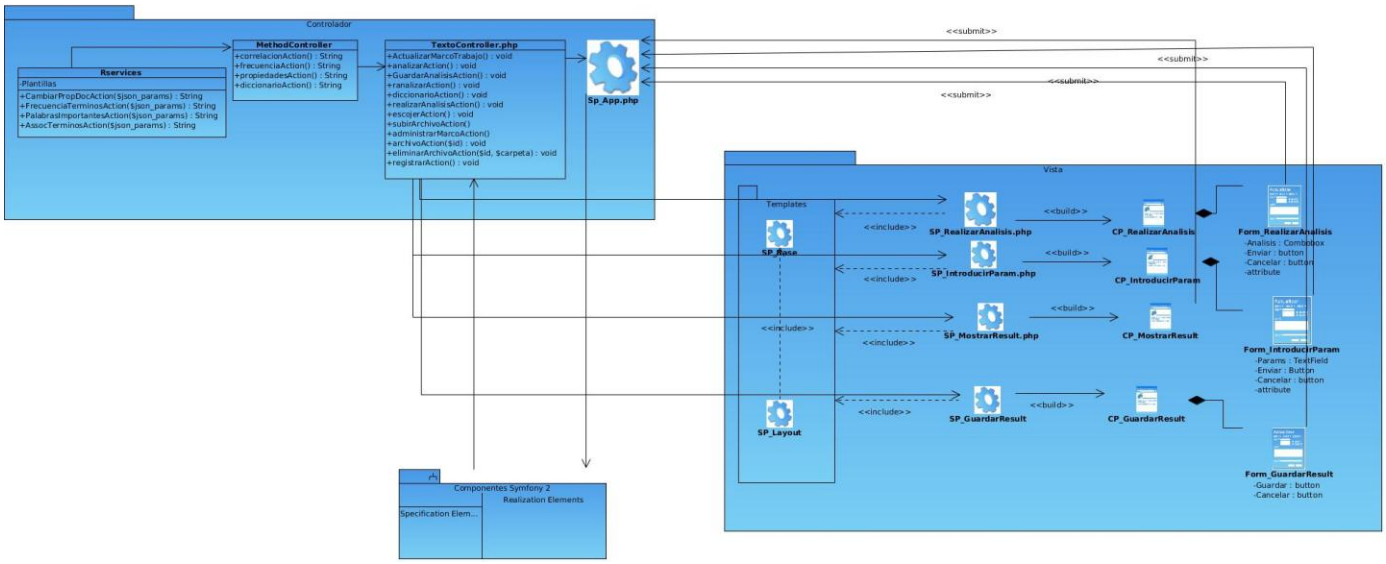


Figura 4. DCD Caso de Uso Gestionar Análisis.

Descripción de clases en caso de uso Gestionar Análisis

MethodController: Clase que permite la gestión de los análisis que provee el módulo, dentro de ella esta la llamada a cada uno de los servicios para la ejecución de cada plantilla.

Rservices.php: Clase encargada de la inyección de dependencias para la ejecución de cada una de las plantillas.

CP_RealizarAnalisis: Clase que permite la realización del tipo de análisis escogido el usuario.

CP_IntroducirParam: Clase que contiene los campos necesarios para la introducción de los parámetros dependiendo del tipo de análisis que se realizará.

CP_MostrarResult: Clase que contiene el resultado del análisis recién hecho.

CP_GuardarResult: Clase que posibilitara al usuario el guardado del resultado en el lugar que desee.

Diagramas de Secuencia

En los diagramas de secuencia se muestra la relación que presentan los objetos que se encuentran dentro del sistema al transcurrir el tiempo, así como los mensajes que se producen entre ellos y la evolución de los mismos durante el desarrollo de todo el escenario que se analiza. Es uno de los diagramas más efectivos para mostrar la evolución e interacción de los objetos de una aplicación. Los diagramas de interacción o secuencia se realizan llevando a cabo el estudio detallado de las descripciones de casos de uso.

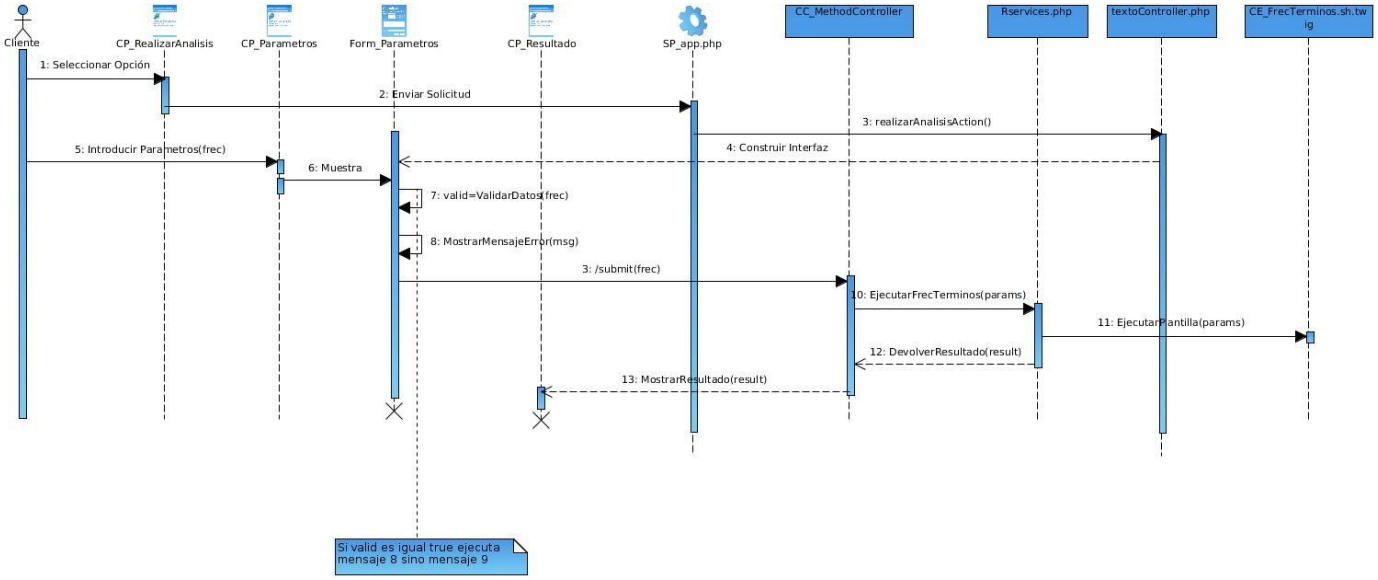


Figura 5 Diagrama de interacción del caso de uso Gestionar Análisis (Escenario Realizar Análisis Frecuencia de Término).

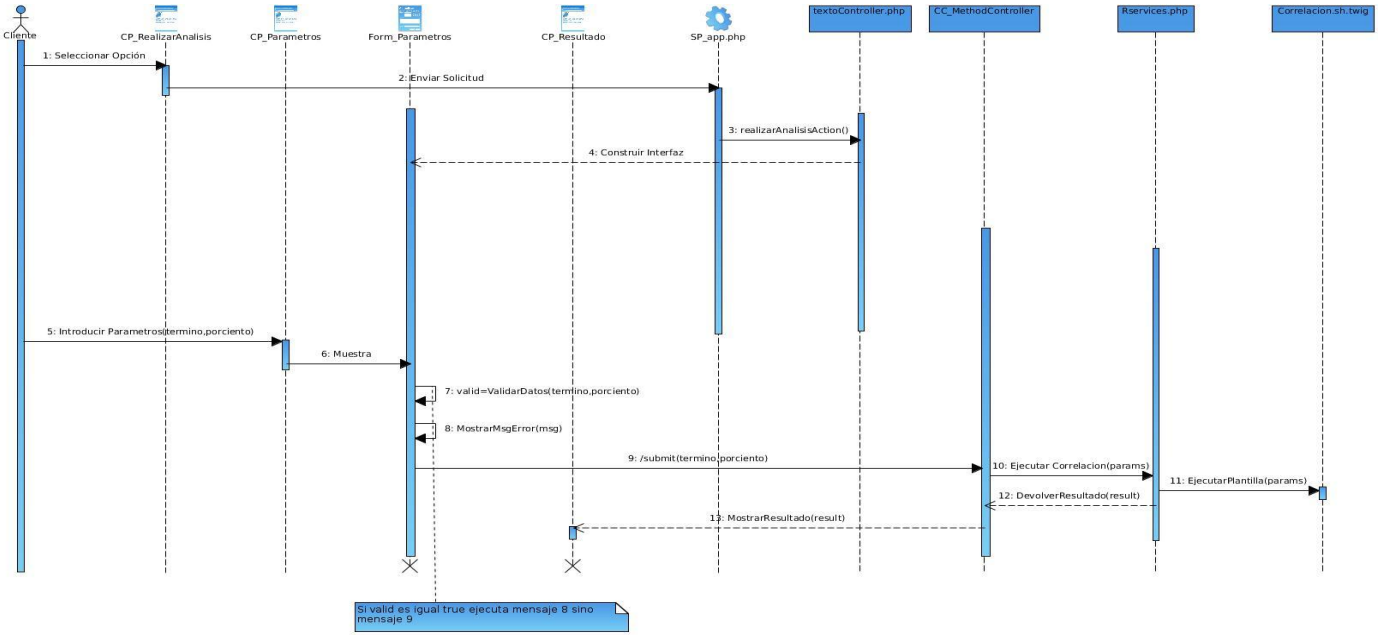


Figura 6 Diagrama de interacción del caso de uso Gestionar Análisis (Escenario Realizar Análisis Correlación de Términos).

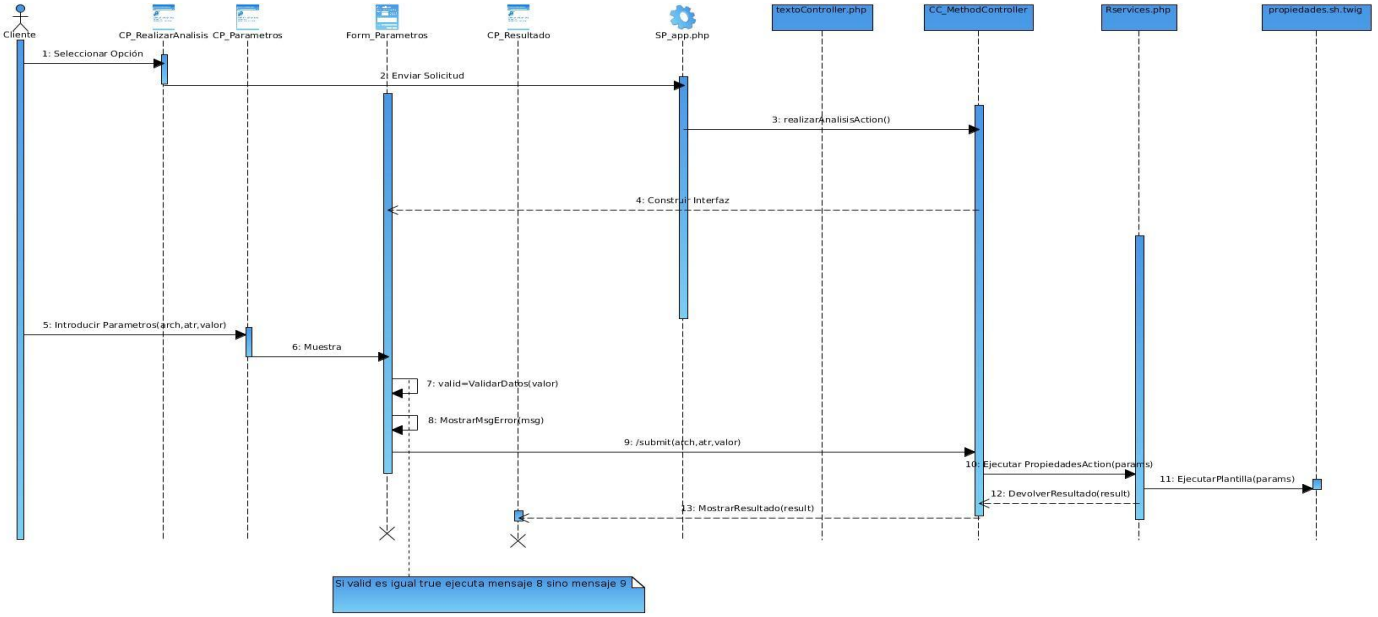


Figura 7 Diagrama de interacción del caso de uso Gestionar Análisis (Escenario Cambiar Propiedades).

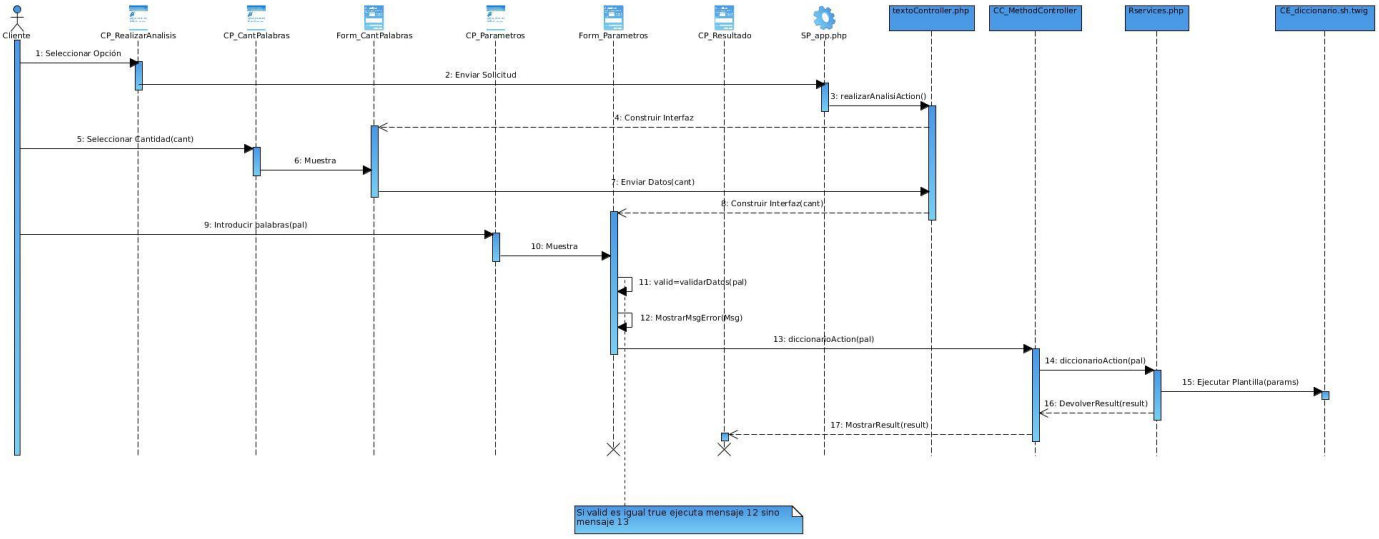


Figura 8 Diagrama de interacción del caso de uso Gestionar Análisis (Escenario Realizar Diccionario).

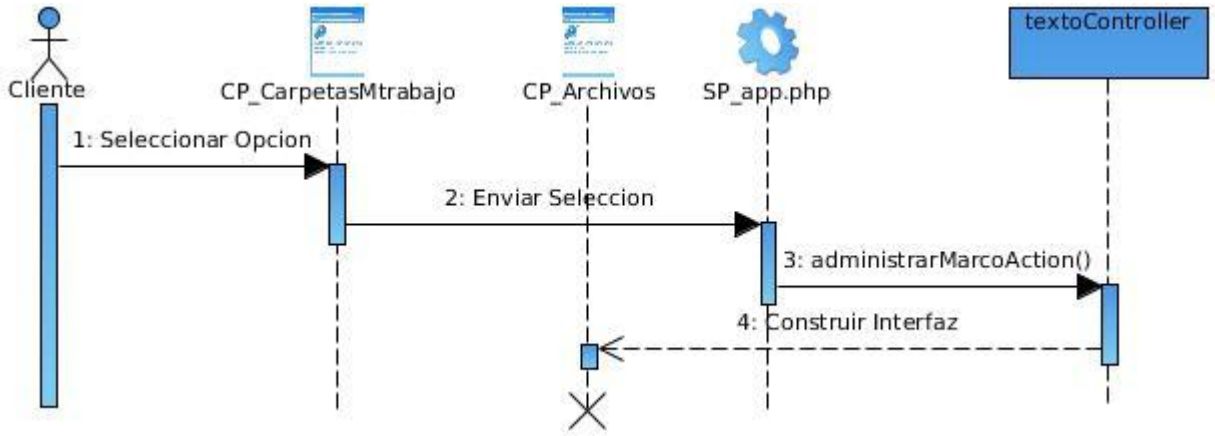


Figura 9 Diagrama de interacción del caso de uso Administrar Marco de Trabajo (Escenario Consultar Archivos).

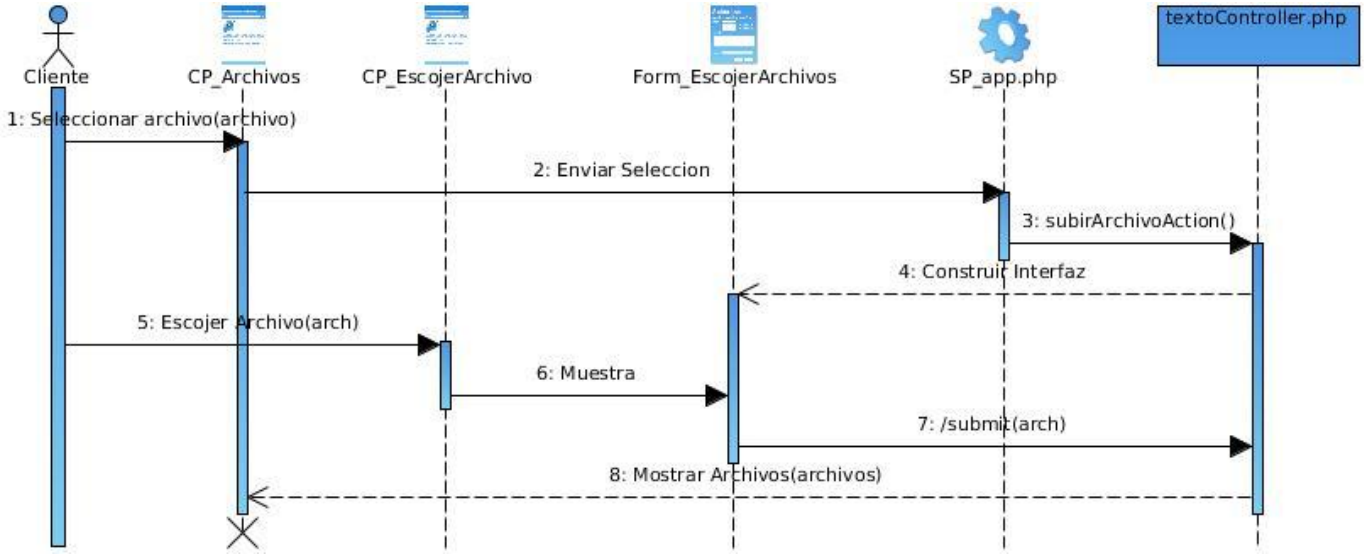


Figura 10. Diagrama de interacción del caso de uso Administrar Marco de Trabajo (Escenario Adicionar Archivo).

2.3.2-Estilo Arquitectónico

Para el desarrollo del sistema se escogió una arquitectura orientada a objetos por lo que se usó el estilo arquitectónico de llamada y retorno, que se responsabiliza del encapsulamiento de la arquitectura en capas, realizando así la separación de la lógica de negocio con la lógica de diseño. Es uno de los estilos más usados en el mundo y una de las causas fundamentales de la popularidad de su uso es que permite el desarrollo en varios niveles.

Dentro de las tecnologías utilizadas se encuentra el Symfony 2, el mismo proporciona un avanzado Modelo Vista Controlador, del inglés, Model-View-Controller (MVC) el cual se utiliza para crear la estructura de la solución final.

MVC es un patrón de diseño de arquitectura de software usado principalmente en aplicaciones que manejan gran cantidad de datos y transacciones complejas donde se requiere una mejor separación de conceptos para que el desarrollo esté estructurado de mejor manera facilitando la programación en diferentes capas de una manera paralela e independiente [14]. Dentro de este patrón se pone de manifiesto el estilo arquitectónico antes expuesto, y cuando se aplica correctamente le permite a los desarrolladores obtener varios niveles de abstracción ya que soporta varias vistas que se separan del modelo y de las clases controladora, lo cual elimina las dependencias directas entre ellos. La figura 11 muestra una vista del mismo.

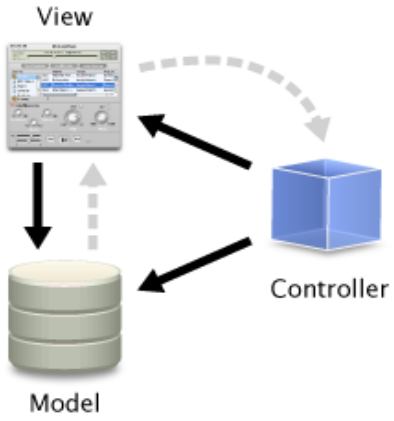


Figura 11. Modelo Vista Controlador

En la figura 12 se puede observar como se evidencia es estilo arquitectónico en el módulo implementado.

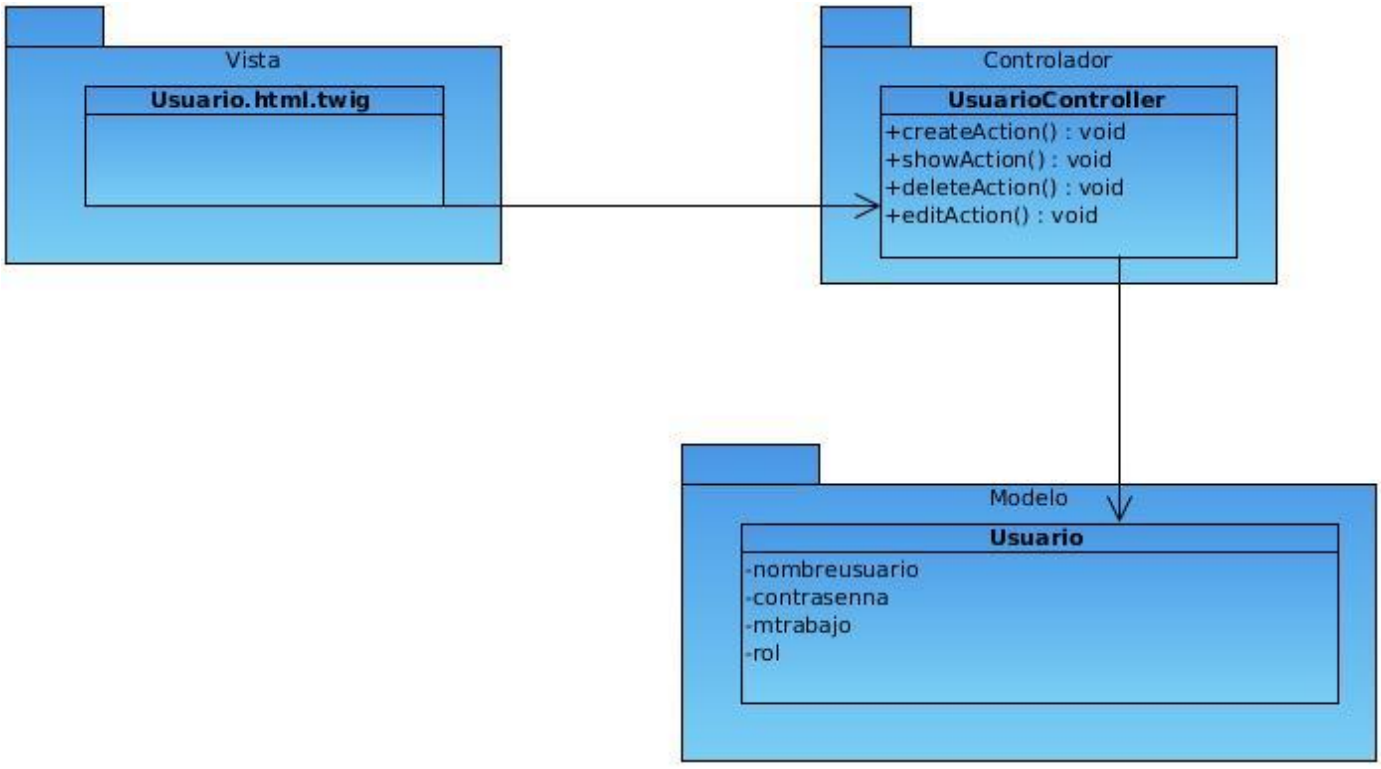


Figura 12 Estilo arquitectónico Modelo Vista Controlador en el módulo de minería de texto.

2.3.3-Patrones de Diseño

Un patrón de diseño nombra, abstrae e identifica los aspectos claves de un diseño estructurado, común, que lo hace útil para la creación de diseños orientados a objetos reutilizables. Los patrones de diseño identifican las clases participantes y las instancias, sus papeles y colaboraciones, y la distribución de responsabilidades [15]. Existen varios patrones que se usaron durante el desarrollo de toda la aplicación y los cuales ayudan a desarrolladores, clientes y personas en general a entender el código expuesto ya que plantean estructuras de programación que lo propician.

Patrones GOF

Los patrones Gang of Four (GoF) es un grupo de patrones definidos por Erich Gamma, Richard Helm, Ralph Johnson y John Vlisides. Dentro de ellos se encuentran un grupo de patrones que especifican y definen un grupo de aspectos claves para el correcto desarrollo de un software.

Patrón Inyección de Dependencias

El patrón de diseño inyección de dependencias es un patrón orientado a objetos, en el que se suministran objetos a una clase en lugar de ser la propia clase quien cree el objeto. El término fue acuñado por primera vez por Martin Fowler. [16]

Este patrón se manifiesta en el sistema en la clase MethodController con la inyección de datos para la creación y utilización de los objetos de la clase Rservices y realizar la llamada del método en cada caso.

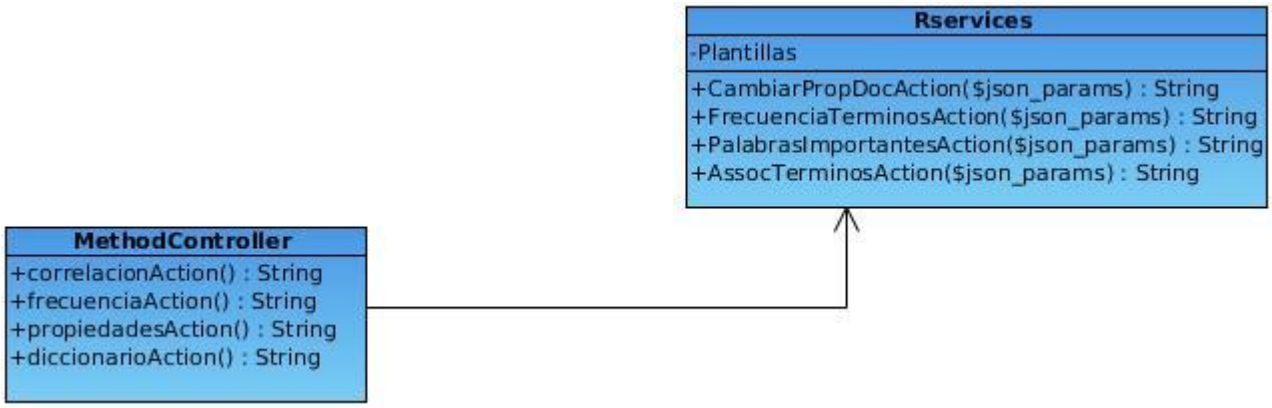


Figura 13. Inyección de dependencias

Patrones GRASP

Los patrones GRASP describen los principios fundamentales de diseño de objetos para la asignación de responsabilidades. Constituyen un apoyo para la enseñanza que ayuda a entender el diseño de objeto esencial y aplica el razonamiento para el diseño de una forma sistemática, racional y explicable. [17]

Patrón Experto

Cuando las responsabilidades a las clases se asignan en forma adecuada, los sistemas tienden a ser más fáciles de entender, mantener y ampliar, lo que permite reutilizar los componentes en futuras aplicaciones. Por lo que este patrón brinda la posibilidad de asignar una responsabilidad al experto en información es decir asignar dicha responsabilidad la clase que cuenta con la información necesaria para cumplirla. [18]

Este patrón se puso en práctica en el módulo de minería de texto con el uso de clases que poseen responsabilidades específicas a cumplir de acuerdo con la información que manejan. Además, estas clases poseen funciones concretas de acuerdo con los datos que gestionan. La imagen representa como se usa en la clase UsuariosController una EntityManager que provee Symfony 2.0 para el trabajo con los datos de las entidades de la base de datos.

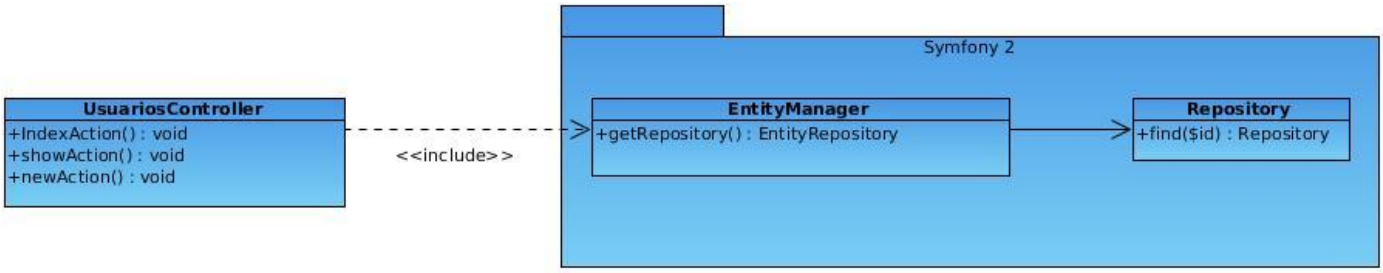


Figura 14. Patrón Experto

Patrón Alta Cohesión

La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no

realicen un trabajo enorme. Como solución el patrón permite asignar una responsabilidad, de modo que la cohesión siga siendo alta. [18]

En la figura 14 se pone de manifiesto el uso de este patrón al distribuir las operaciones de creación de formularios en las clases UsuariosType y RegisterType, con esto, al invocar el método createAction de la clase UsuarioController es capaz de crearse los formularios específicos para cada caso.

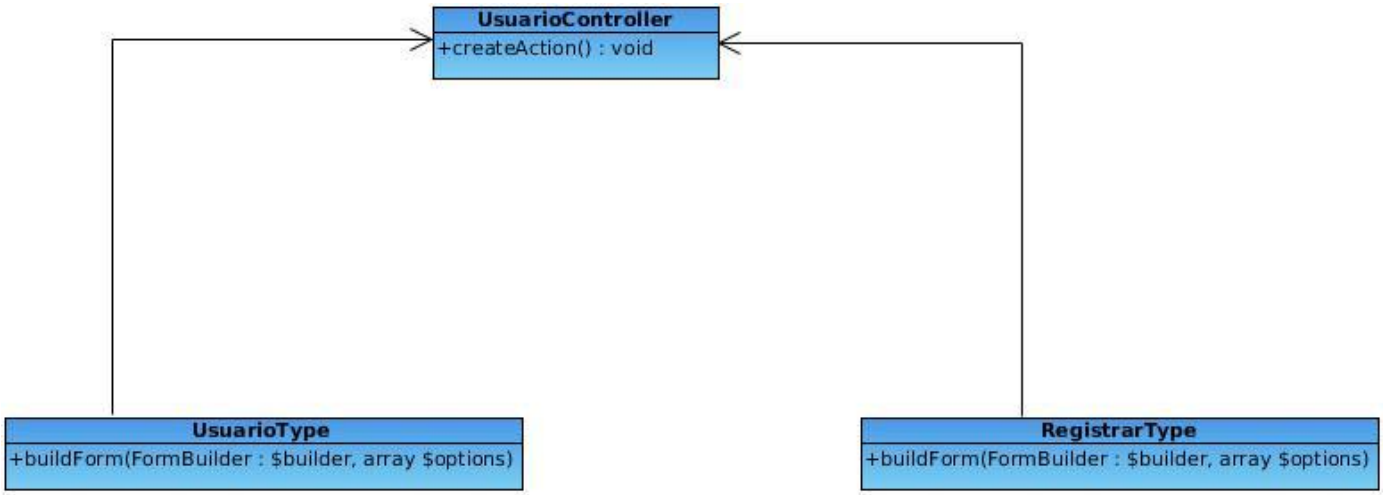


Figura 15. Patrón Alta Cohesión

2.3.4-Vista de Despliegue

En la vista de despliegue se muestra el hardware que se utilizará en la aplicación así como la comunicación que existirá en cada caso, es un diagrama de vital importancia para tener constancia de que tipo y cuanta tecnología se utilizará.

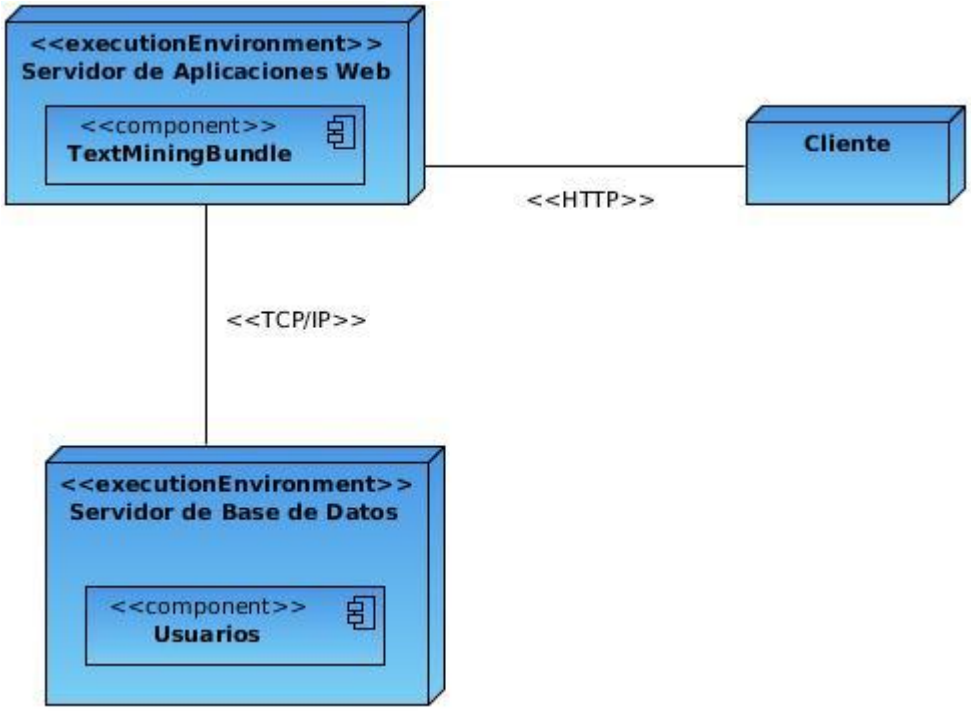


Figura 16. Diagrama de Despliegue del MÓDULO de Minería de texto.

2.4-Conclusiones parciales

- ✓ Se identificaron todos los requisitos funcionales y se definieron un grupo de requisitos no funcionales con los cuales se explicó de una manera más concreta y formal la constitución del sistema y las operaciones que deberá realizar.
- ✓ Se definió el diagrama de casos de uso del sistema correspondiente a la aplicación para proveer a clientes y desarrolladores de un mejor entendimiento de las funcionalidades, también se aplicaron patrones a los mismos para que fueran mejor entendidos.
- ✓ Se realizaron descripciones textuales de los casos de uso en cuestión para hacer un mejor desglose de los mismos.
- ✓ Se mostró el modelo de diseño que presentará el sistema, y se explicaron los diagramas de clases del diseño más importantes que existen en el módulo.

- ✓ Se explicó el patrón arquitectónico usado y algunos de los patrones de diseño tenidos en cuenta para comenzar el desarrollo del módulo.
- ✓ Se modelaron los diagramas de interacción para un grupo de escenarios importantes dentro de la implementación.
- ✓ Se definió la Vista de despliegue de la aplicación y para tener conocimiento del hardware que se usará y el tipo de comunicación entre ellos.

CAPÍTULO 3. IMPLEMENTACIÓN Y PRUEBA

3.1 Introducción

El proceso de diseño provee un grupo de resultados consistentes en artefactos, dentro de los que se encuentran un grupo de diagramas y descripciones con los que se comienza a implementar el sistema. A través de componentes expone lo que será la implementación del mismo, el presente capítulo tiene como objetivo fundamental el total desarrollo de la arquitectura propuesta utilizando el método de caja negra para la realización de las pruebas de aceptación y la verificación de las funcionalidades del sistema.

3.2 Implementación del sistema

A través del modelo de implementación se describen cada uno de los componentes y subsistemas que están físicamente en el sistema, con ellos se tiene una vista más aterrizada a lo que será la estructura física del mismo ya que al finalizar el proceso se puede ver cómo quedará cada uno de los componentes físicos que contendrá la aplicación y su relación con los subsistemas y las clases del modelo de diseño.

3.2.1 Diagrama de Componentes

El diagrama de componentes describe elementos físicos del sistema y las relaciones que existen entre ellos, mostrando así las opciones de realización incluyendo código fuente, binario y ejecutable. Un componente puede ser uno o varios archivos, paquetes de archivos, bibliotecas cargadas dinámicamente, subsistemas que interactúen con la aplicación, en fin, en estos diagramas se encontrarán todos los elementos de software que se encuentran en la aplicación que los desarrolladores construirán posteriormente.

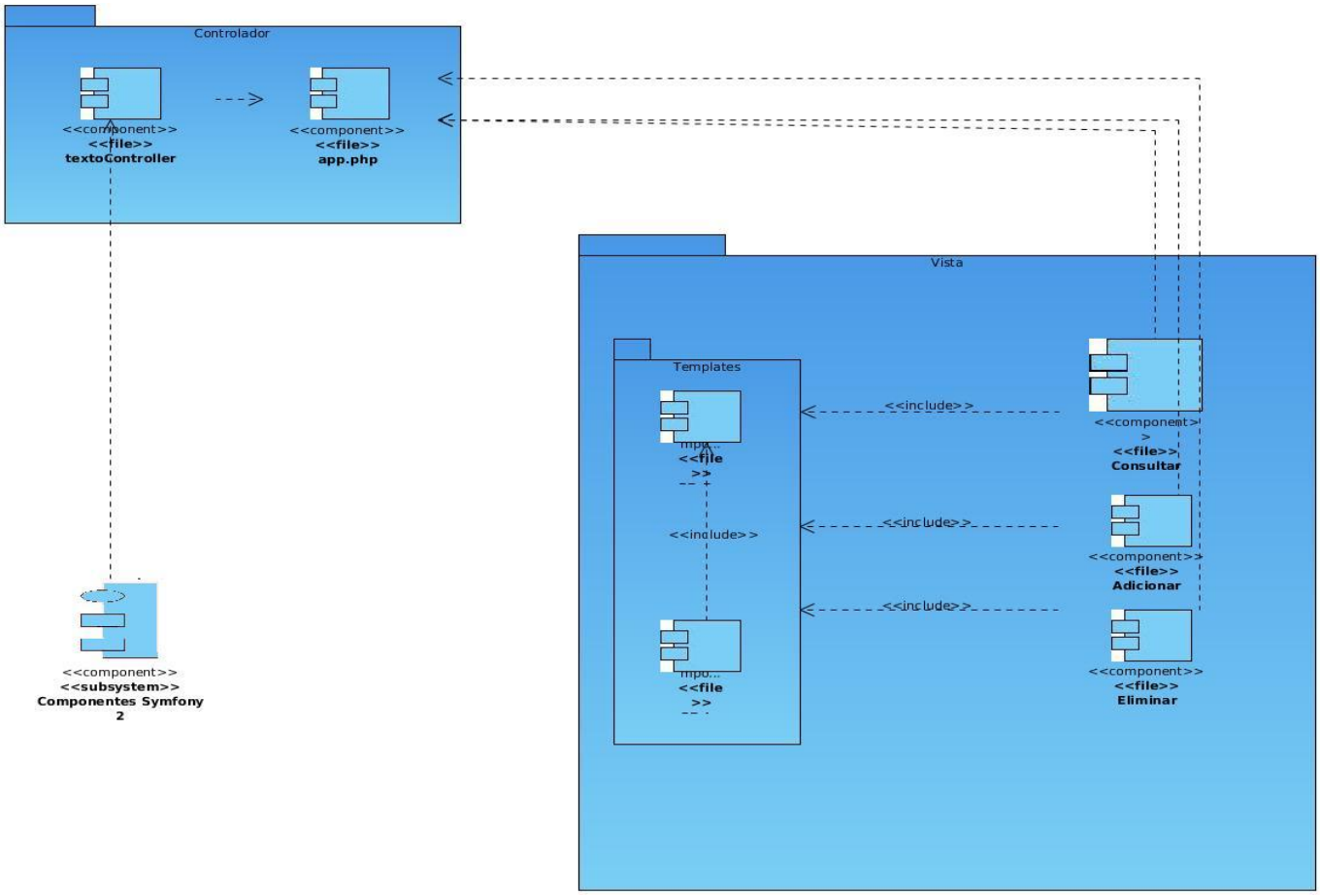


Figura 17. Diagrama de Componentes CU Administrar Marco Trabajo.

El diagrama representa como se realizará la subida de archivos por parte del usuario a la aplicación, dentro de la clase **textoController** se encuentran todas las operaciones necesarias para garantizar dichas acciones.

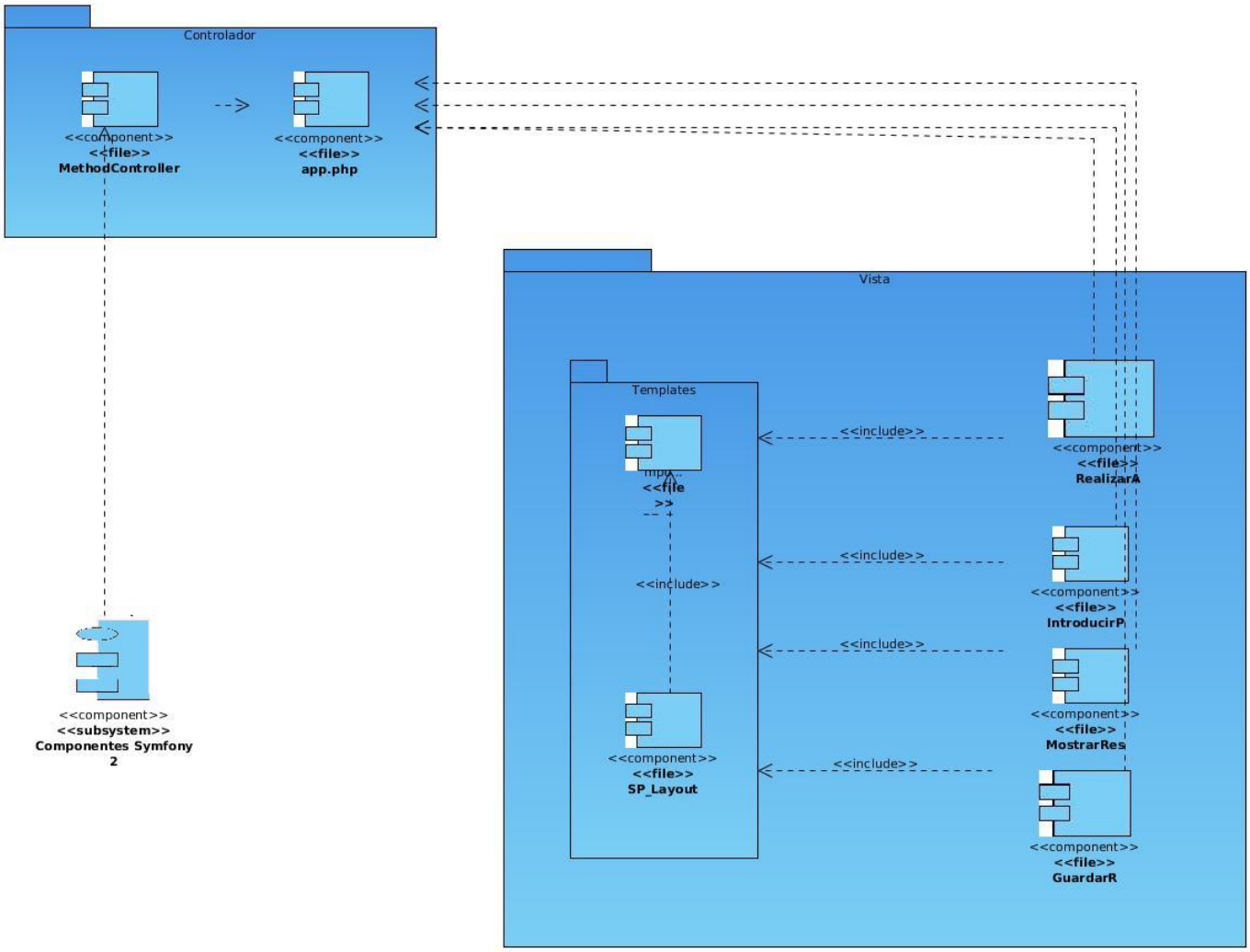


Figura 18. Diagrama de Componentes CU Gestionar Análisis

En este diagrama de componentes se pone de manifiesto como se gestionan los análisis a través de la clase **MethodController** que es la encargada de la ejecución de cada uno de los tipos de análisis de la aplicación y que a su vez usa la clase **textoController** para la ejecución de cada una de las plantillas.

3.3 Pruebas del sistema

El proceso de pruebas es un proceso fundamental a la hora de la comprobación del sistema, es un proceso que permite la verificación del correcto funcionamiento de cada una de las funcionalidades. Con

este proceso se obtiene la calidad que tiene dicho sistema y es vital para el completo desarrollo del mismo, se realiza a través de casos de prueba que se le hacen a cada uno de los requisitos funcionales permitiéndole a la persona observar el comportamiento del caso de uso específico y tomar decisiones en cada caso. Luego de realizarle pruebas al sistema se obtuvieron diversos resultados.

3.3.1 Tipos de prueba de software

La fase de pruebas es una de las más costosas del ciclo de vida software. En sentido estricto, deben realizarse pruebas de todos los artefactos generados durante la construcción de un producto, lo que incluye especificaciones de requisitos, casos de uso, diagramas de diversos tipos y, por supuesto, el código fuente y el resto de productos que forman parte de la aplicación. [19]

Existen diversos tipos de pruebas entre las que se encuentran:

- ✓ Pruebas de requisitos.
- ✓ Pruebas del diseño.
- ✓ Revisiones e inspecciones del código fuente.
- ✓ Pruebas estructurales o de caja blanca.
- ✓ Pruebas funcionales o de caja negra.

Esta última fue la escogida debido a la complejidad del módulo. A partir de la tabla 7 se pueden observar las distintas respuestas que se dieron en cada escenario del caso de uso Gestionar Análisis (caso de uso más importante dentro del módulo de minería de texto).

Escenario	Descripción	Respuesta del sistema
EC 1.1 Realizar el Análisis correctamente	Se realiza correctamente el análisis	Se realiza correctamente el análisis y se muestra el resultado.
EC 1.3 Realizar el	Se desea realizar el	Se muestra un mensaje indicando que los

Análisis con datos no válidos(Ejemplo: Letras)	análisis con datos no válidos. Se muestra un mensaje indicando el error.	datos son incorrectos.
EC 1.2 Realizar el análisis. Dejando el campo en blanco	Se desea realizar el análisis dejando un campo en blanco. Se muestra un mensaje indicando el error.	Se muestra un mensaje indicando que existen campos requeridos en blanco.
EC 1.4 Cancelar. Realizar Análisis Frecuencia de Términos	Se cancela la operación Realizar Análisis Frecuencia de Términos	Se cancela la operación Realizar Análisis Frecuencia de Términos

Tabla 7 Casos de Prueba CU Gestionar Análisis (Escenario Realizar Análisis Frecuencia de Términos)

Escenario	**Descripción**	**Respuesta del sistema**
EC 1.1 Realizar el Análisis correctamente	Se realiza correctamente el análisis	Se realiza correctamente el análisis y se muestra una interfaz con la respuesta.
EC 1.3 Realizar el Análisis con datos no válidos(Ejemplo: Letras en lugar de	Se desea realizar el análisis con datos no válidos. Se muestra un mensaje	Se muestra un mensaje indicando que los datos son incorrectos.

números)	indicando el error.	
EC 1.2 Realizar el análisis. Dejando el campo en blanco	Se desea realizar el análisis dejando un campo en blanco. Se muestra un mensaje indicando el error.	Se muestra un mensaje indicando que existen campos requeridos en blanco.
EC 1.4 Cancelar. Realizar Análisis Correlación de Palabras	Se cancela la operación Realizar Análisis Correlación de Palabras	Se cancela la operación Realizar análisis Correlación de palabras

Tabla 8. Casos de Prueba CU Gestionar Análisis (Escenario Realizar Análisis Correlación de Palabras)

Escenario	**Descripción**	**Respuesta del sistema**
EC 1.1 Realizar el Análisis correctamente	Se realiza correctamente el análisis	Se realiza correctamente el análisis y se muestra una interfaz con la respuesta.
EC 1.3 Realizar el Análisis con datos no válidos(Ejemplo: Letras en lugar de números)	Se desea realizar el análisis con datos no válidos. Se muestra un mensaje indicando el error.	Se muestra un mensaje indicando que los datos son incorrectos.

EC 1.2 Realizar el análisis. Dejando el campo en blanco	Se desea realizar el análisis dejando un campo en blanco. Se muestra un mensaje indicando el error.	Se muestra un mensaje indicando que existen campos requeridos en blanco.
EC 1.4 Cancelar. Realizar Análisis Cambiar Propiedades	Se cancela la operación Realizar Análisis Cambiar Propiedades	Se cancela la operación Realizar Análisis Cambiar Propiedades

Tabla 9. Casos de Prueba CU Gestionar Análisis (Escenario Realizar Análisis Cambiar Propiedades)

3.4 Conclusiones parciales

En el presente capítulo se ha demostrado como será la implementación del sistema y se han definido:

- ✓ La descripción de lo que será la implementación del módulo.
- ✓ Los diagramas de componentes que muestran los distintos archivos que componen el sistema así como su explicación.
- ✓ Los diferentes casos de prueba hechos a los casos de uso en los distintos escenarios que se ponen de manifiesto.

CONCLUSIONES

Con el desarrollo de los 3 capítulos del presente trabajo se demostró todo el cursar del desarrollo de la aplicación informática en cuestión, con el mismo se puso de manifiesto la importancia de la minería de texto y la necesidad de trabajar con ella en las empresas actuales.

- ✓ Se definieron todas las funcionalidades para darle solución a la necesidad de realización de minería de texto dentro del RServer 2.0
- ✓ Se realizó un completo diseño del módulo donde se definieron un total de 15 requisitos funcionales y 7 requisitos no funcionales y se le mostró tanto a clientes como desarrolladores como se comportaría el intercambio de información en cada caso a través de los distintos diagramas.
- ✓ Se realizaron pruebas de caja negra para comprobar los resultados que se obtenían en cada caso viendo como se comportaba el sistema en cada uno de los casos con el caso de uso Gestionar Análisis.

RECOMENDACIONES

Que se continúe con el desarrollo de nuevos análisis de minería de texto integrando nuevas librerías de R, existen librerías que de conjunto con TM hacen gran trabajo con los textos, esto se pudiera seguir profundizando para lograr un trabajo más eficaz con ellos.

REFERENCIAS BIBLIOGRÁFICAS

[1] **CRESPO, JAIME SEPTIÉN.** Jarra sin tapadera. [En línea] 23 de Febrero de 2011. [Citado el: 12 de Noviembre de 2011.] http://jaimeseptien.com/2011/02/jarra-sin-tapadera/.

[2] *Minería de Textos.* **Dürsteler, Juan C. 2001.** 2001.

[3] **Ingo Feinerer, Kurt Hornik, David Meyer .**Journal of Statistical Software**,** Marzo 2008, Volumen 25, Issue 5.[Text Mining Infrastructure in R.pdf]

[4] **Quezada, Carlos Villegas.** *Análisis comparativo de herramientas informáticas para "Minería de texto" y sus posibilidades de aplicación en el análisis de documentos de educación a distancia seleccionados en el web.* [En línea] [Citado el: 25 de Diciembre de 2011.] http://www.uned.es/catedraunesco-ead/Colaboraciones/villegas/apartado4.htm.

[5] **Lancia, Franco.** T-LAB. [En línea] [Citado el: 25 de Diciembre de 2011.] http://www.tlab.it/es/presentation.php.

[6] *R, un lenguaje de programación que seduce.* **Marcos. 2009.** 2009

[7] **Potencier, François Zaninotto y Fabien.** *Symfony en pocas palabras.* [En línea] [Citado el: 21 de Noviembre de 2011.] http://www.librosweb.es/symfony_1_1/cMódulotulo1/symfony_en_pocas_palabras.html.

[8] **Software, Ingenieria de.** *Modelado de Software*. [En línea] http://softwareiimarfrednarvaez.blogspot.com/2012_02_01_archive.html.

[9] **Perissé, Marcelo Claudio.** *Herramientas Case*. [En línea] [Citado el: 20 de Diciembre de 2011.] http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/proyectoinformatico/libro/c5/c5.htm.

[10] **2007.** *Paradigma visual para UML (Plataforma Java)*. [En línea] 5 de Marzo de 2007. [Citado el: 15 de Enero de 2012.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.

[11] **2004.** *Sistema Gestor de base de datos SGBD*. [En línea] 01 de Noviembre de 2004. [Citado el: 10 de Enero de 2012.] http://www.error500.net/garbagecollector/bases_de_datos/sistema_gestor_de_base_de_dato.php.

[12] **2010.** *PostgreSQL*. [En línea] 2 de Octubre de 2010. [Citado el: 10 de Enero de 2012.] http://www.postgresql.org.es/sobre_postgresql.

[13] **2010.** Acerca de NetBeans IDE . [En línea] 26 de Julio de 2010. [Citado el: 11 de Enero de 2012.] http://aprendamos-netbeans.blogspot.com/.

[14] **López, Alejandro Rivera. 2008.** *Arquitectura del Software* . [En línea] 16 de Enero de 2008. [Citado el: 12 de Enero de 2012.] http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/rivera_l_a/cMódulotulo2.pdf.

[15] **JUAN, FRANCISCO JAVIER MARTÍNEZ.** *Guía de construcción de software en java con patrones de diseño.* Oviedo : s.n.

[16] **Fowler, Martin**. [En línea] 23 de Enero de 2004. [Citado el: 23 de Febrero de 2012.] http://martinfowler.com/articles/injection.html.

[17] **Gutierrez, Jorge A. Saavedra.** [En línea] 8 de Mayo de 2007. [Citado el: 20 de Febrero de 2012.] http://jorgesaavedra.wordpress.com/category/patrones-grasp/.

[18] **Erich Gamma, Richard Helm, Ralph Johnson, John Vlisside.** *Design Patterns. Elements of Reusable Object-Oriented Software* s.l.: Addison-Wesley, 1995.

[19] **Usaola, Dr. Macario Polo.** *Mantenimiento Avanzado de Sistemas de Informacion Pruebas del Software.*

.

BIBLIOGRAFÍA

1. **2010.** Acerca de NetBeans IDE . [En línea] 26 de Julio de 2010. [Citado el: 11 de Enero de 2012.] http://aprendamos-netbeans.blogspot.com/.
2. *Jarra sin tapadera.* **Septién, Jaime. 2011.** 2011.
3. **JUAN, FRANCISCO JAVIER MARTÍNEZ.** *Guía de construcción de software en java con patrones de diseño.* Oviedo : s.n.
4. **Lancia, Franco.** T-LAB. [En línea] [Citado el: 25 de Diciembre de 2011.] http://www.tlab.it/es/presentation.php.
5. **López, Alejandro Rivera. 2008.** Arquitectura del Software . [En línea] 16 de Enero de 2008. [Citado el: 12 de Enero de 2012.] http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/rivera_l_a/cMódulotulo2.pdf.
6. *Minería de Textos.* **Dürsteler, Juan C. 2001.** 2001.
7. **2007.** Paradigma visual para UML (Plataforma Java). [En línea] 5 de Marzo de 2007. [Citado el: 15 de Enero de 2012.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.
8. *Patrones de Diseño.* **Hernandez, Jimmy. 2009.** 2009.
9. **Perissé, Marcelo Claudio.** Herramientas Case. [En línea] [Citado el: 20 de Diciembre de 2011.] http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/proyectoinformatico/libro/c5/c5.htm.
10. PHP. [En línea] [Citado el: 2011 de Diciembre de 27.] http://es.scribd.com/doc/51421503/php.

11. **2010.** PostgreSQL. [En línea] 2 de Octubre de 2010. [Citado el: 10 de Enero de 2012.] http://www.postgresql.org.es/sobre_postgresql.

12. **Potencier, François Zaninotto y Fabien.** Symfony en pocas palabras. [En línea] [Citado el: 21 de Noviembre de 2011.] http://www.librosweb.es/symfony_1_1/cMódulotulo1/symfony_en_pocas_palabras.html.

13. **Quezada, Carlos Villegas.** ANÁLISIS COMPARATIVO DE HERRAMIENTAS INFORMÁTICAS PARA "MINERÍA DE TEXTO" Y SUS POSIBILIDADES DE APLICACIÓN EN EL ANÁLISIS DE DOCUMENTOS DE EDUCACIÓN A DISTANCIA SELECCIONADOS EN EL WEB. [En línea] [Citado el: 25 de Diciembre de 2011.] http://www.uned.es/catedraunesco-ead/villegas/apartado4.htm.

14. *R, un lenguaje de programación que seduce.* **Marcos. 2009.** 2009.

15. **2004.** Sistema Gestor de base de datos SGBD. [En línea] 01 de Noviembre de 2004. [Citado el: 10 de Enero de 2012.] http://www.error500.net/garbagecollector/bases_de_datos/sistema_gestor_de_base_de_dato.php.

16. **Ingo Feinerer, Kurt Hornik, David Meyer .**Journal of Statistical Software**,** Marzo 2008, Volumen 25, Issue 5.[Text Mining Infrastructure in R.pdf]

17. **Adeva JJG, Calvo R (2006).** "Mining Text with Pimiento." IEEE Internet Computing, 10(4), http://ieeexplore.ieee.org/

18. **Weiss S, Indurkhya N, Zhang T, Damerau F (2004).** Text Mining: Predictive Methods for Analyzing Unstructured Information.

19. **Blanco, M.a del Pilar Cantero.** *Tecnimap.* 2004.

20. **Rnews.** *An Introduction to Text Mining in R.* 2008.

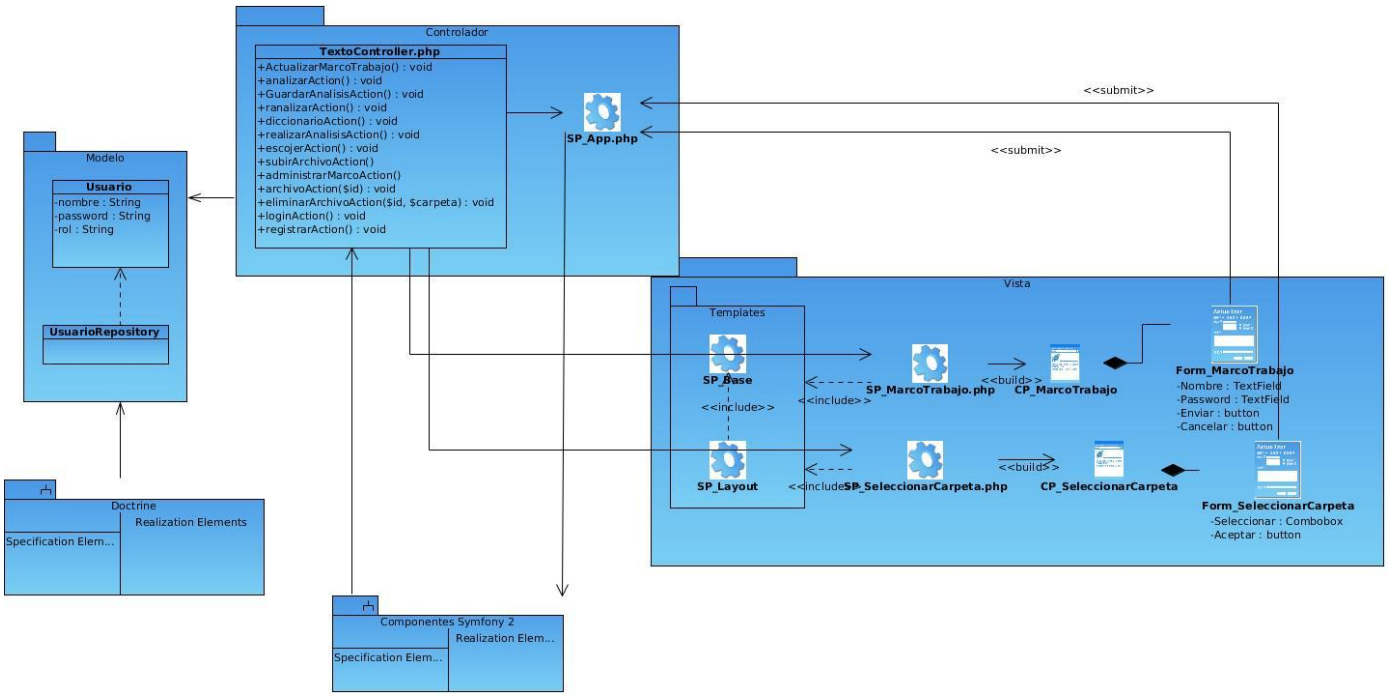
21. **Rnews.** *The profileModel R Package.* 2008.

22. **Version, 1.5.0.1 OpenUP.** *Ayuda de OpenUP Version 1.5.0.1.*

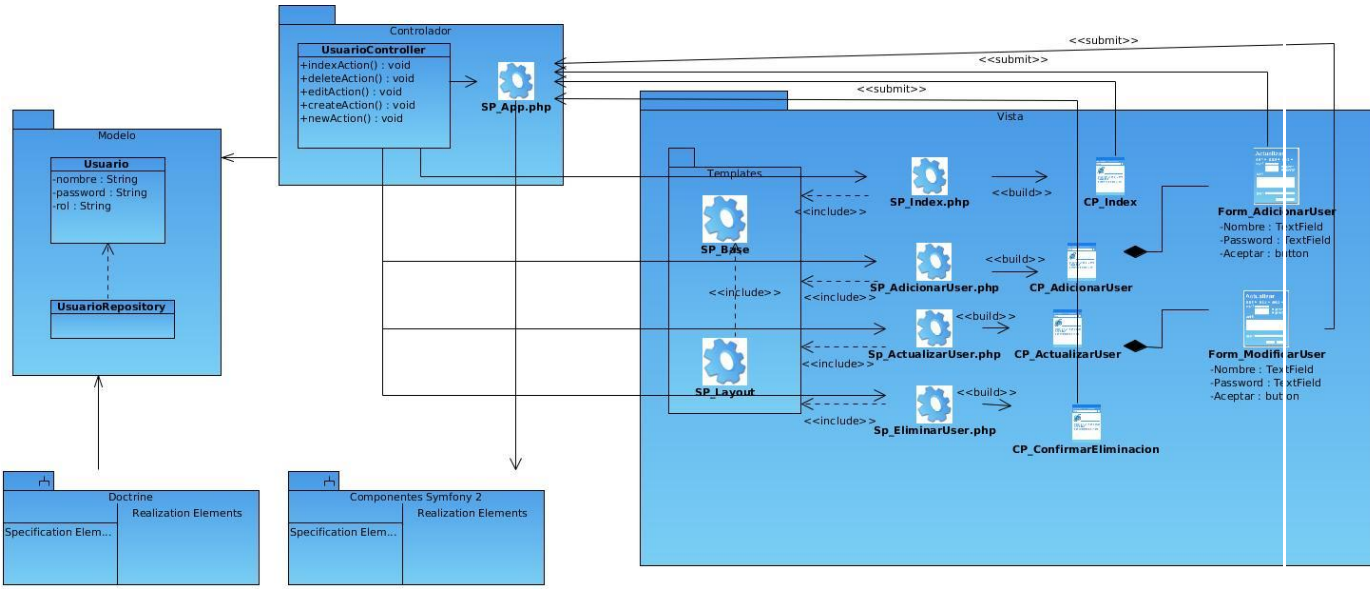
23. **visual-paradigm.** [En línea] http://www.visual-paradigm.com/product/vpsuite/.

24. **netbeans.org**. [En línea] http://netbeans.org/index_es.html.

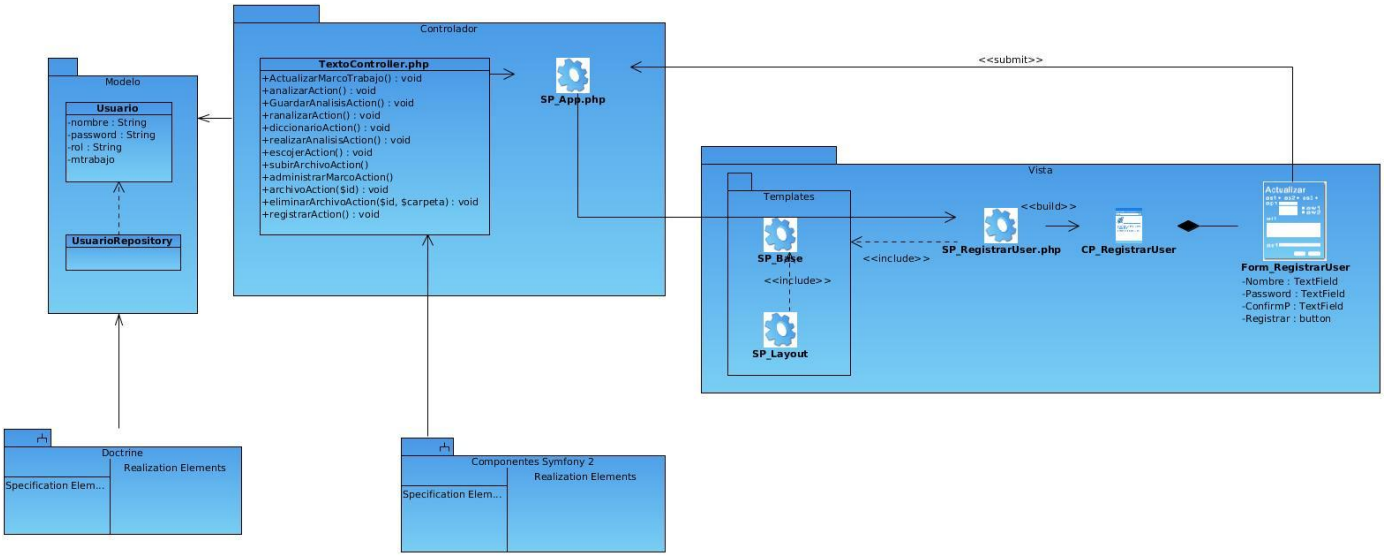
ANEXOS



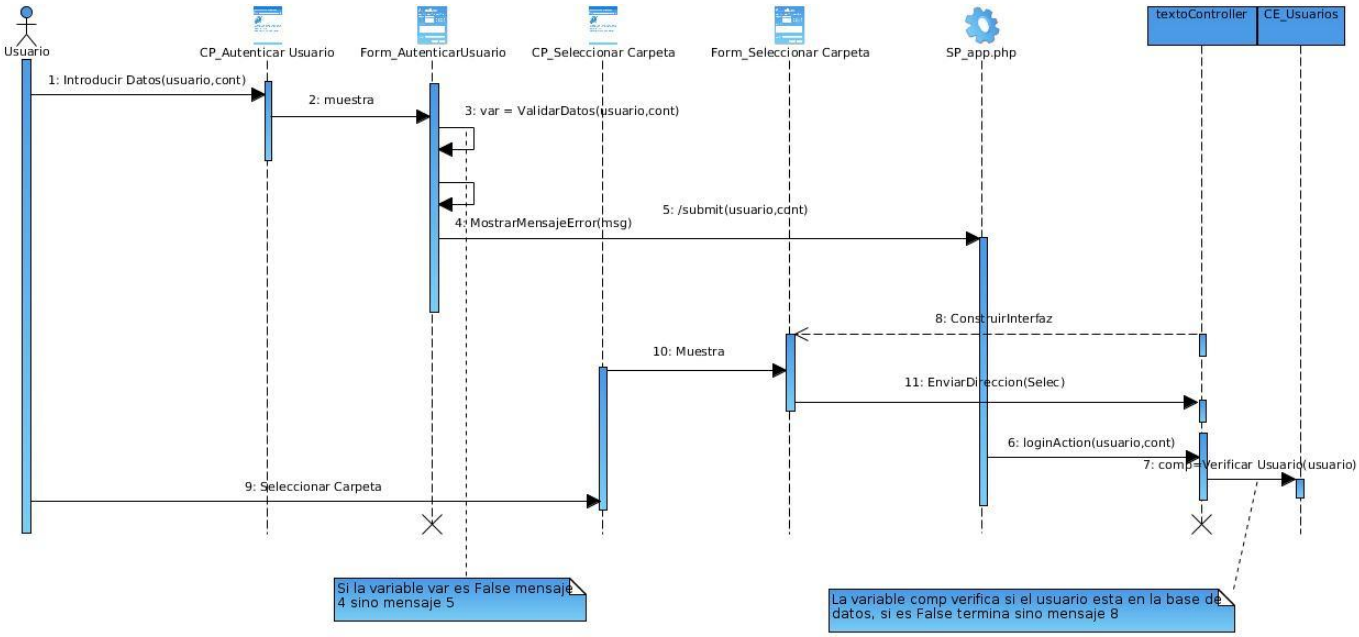
Anexo 1: Diagrama de diseño CU Autenticar Usuario



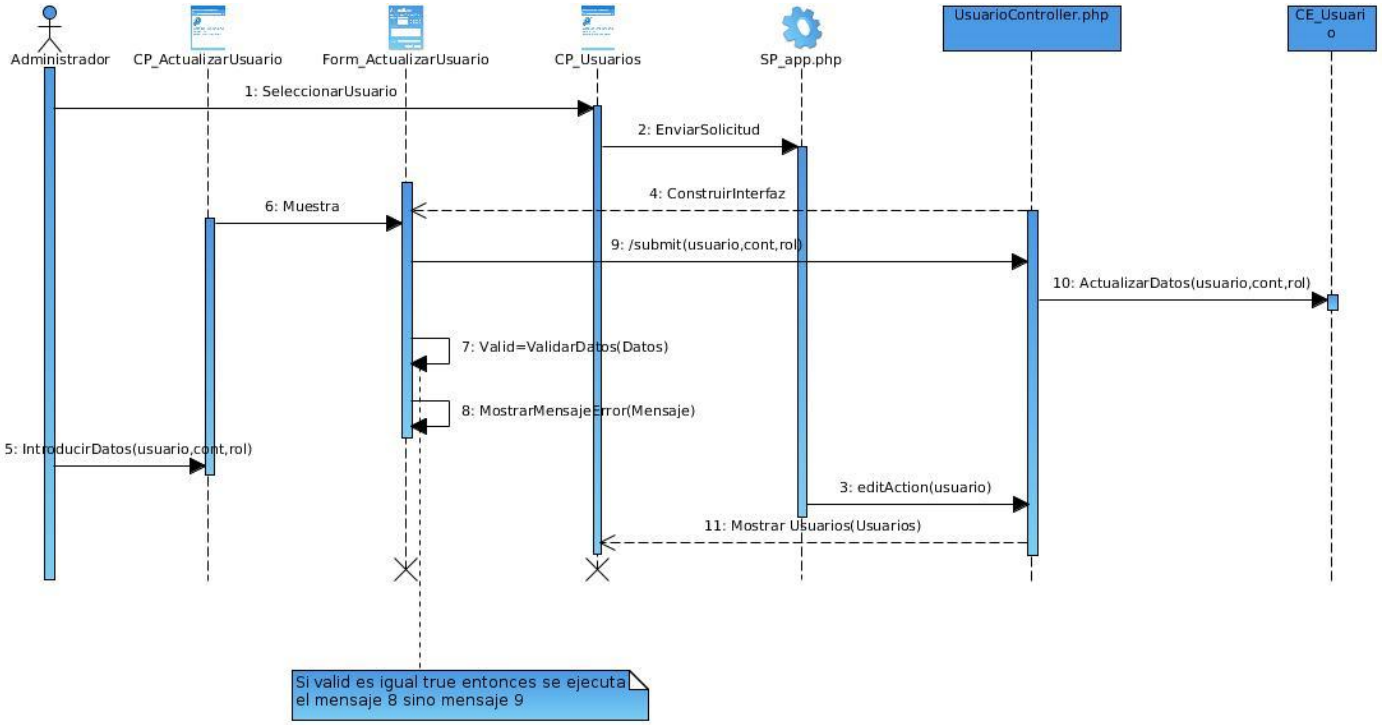
Anexo 2: Diagrama de diseño CU Gestionar Usuario



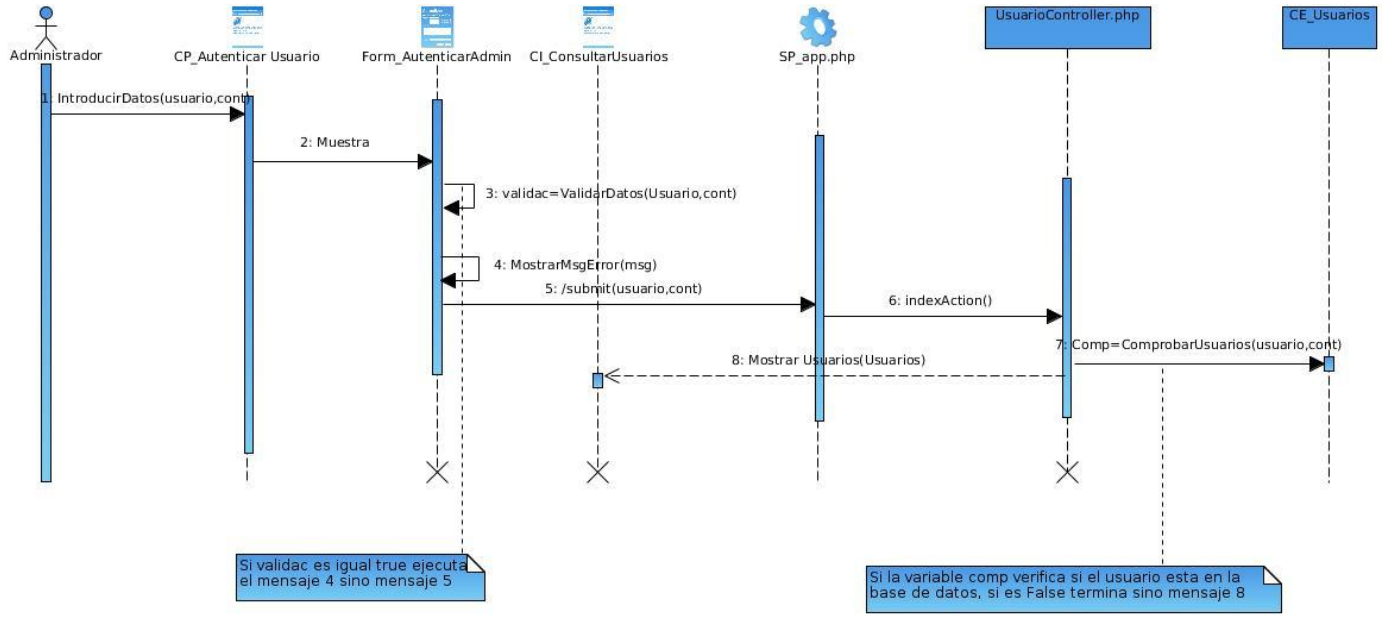
Anexo 3: Diagrama de diseño CU Registrar Usuario



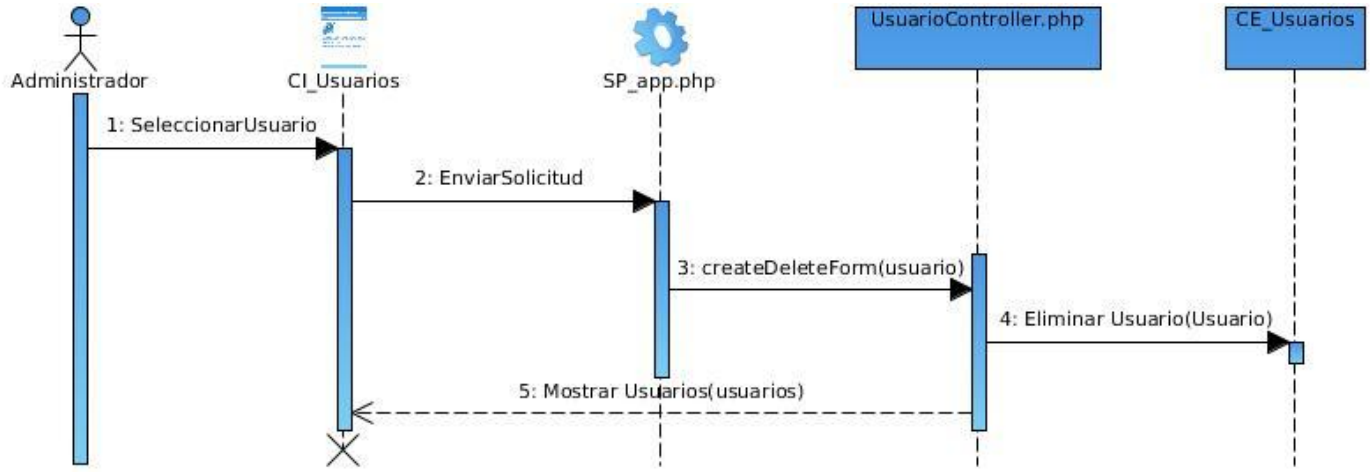
Anexo 4: Diagrama de interacción CU Registrar Usuario Escenario Autenticar Usuario



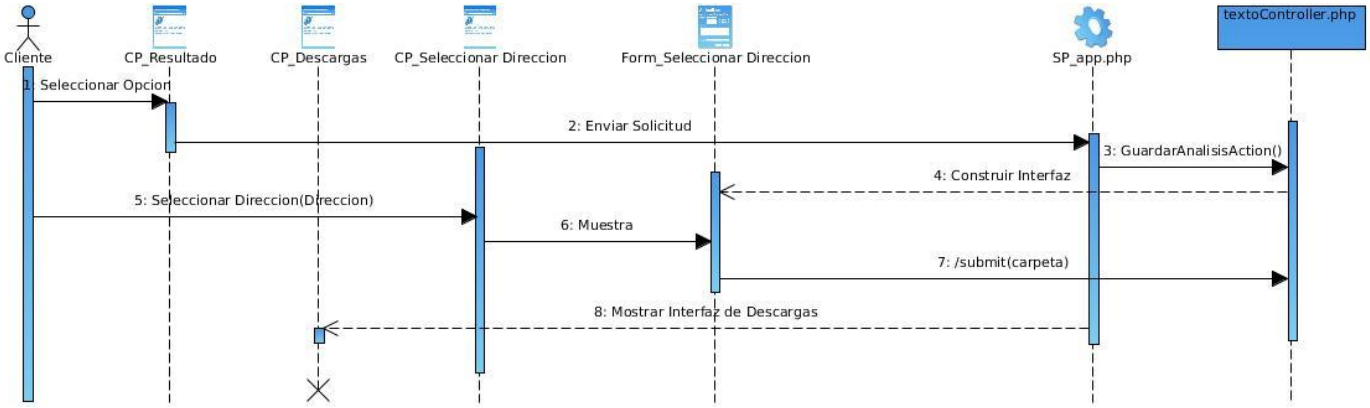
Anexo 5: Diagrama de interacción CU Gestionar Usuario Escenario Actualizar Usuario



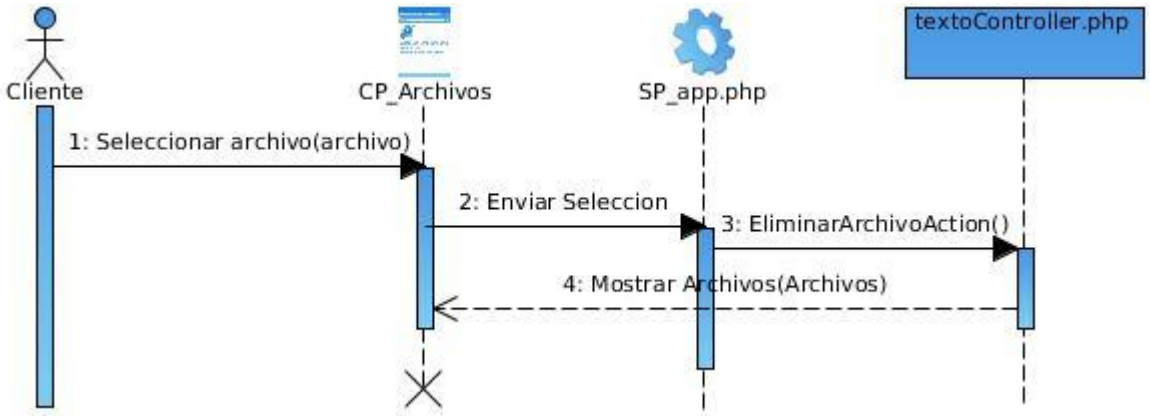
Anexo 6: Diagrama de interacción CU Gestionar Usuario Escenario Consultar Usuario



Anexo 7: Diagrama de interacción CU Gestionar Usuario Escenario Eliminar Usuario



Anexo 8: Diagrama de interacción CU Gestionar Análisis Escenario Guardar Análisis.



Anexo 9: Diagrama de interacción CU Administrar Marco de Trabajo Escenario Eliminar Archivo.

GLOSARIO DE TÉRMINOS

MÓDULO: Aplication Program Interface (Interfaz de Programación de Aplicaciones).

Base de Datos: Es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

BSD: Berkeley Software Distribution (Distribución de software Berkeley)

CASE: Computer Aided Software Engineering (Herramientas de Software Asistidas por Ordenador)

DATEC: Centro de Tecnología de Gestión de Datos.

ICRC: Coeficiente interactivo de confianza en la codificación.

IDE: Integrated Development Environment (Entorno de Desarrollo Integrado)

Licencia BSD: Licencia de software otorgada principalmente para los sistemas BSD (Berkeley Software Distribution).

OpenUP: Open Unified Process

PHP: Lenguaje de programación interpretado muy usado del lado del servidor en la programación web.

R: Lenguaje y entorno de programación para análisis estadístico y gráfico.

SPSS: Statistical Package for the Social Sciences (Paquete estadístico para ciencias Sociales).

SQL: Structured Query Language(Lenguaje Estructurado de Consultas)

TM: Librería que provee el trabajo con textos dentro del lenguaje de programación R.