



Facultad 1

“Subsistema de recopilación automática de noticias de los medios informativos de la Universidad de las Ciencias Informáticas”

**Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas**

Autor: Liana Maletá Terry

Tutores:

Ing. Disnayle Jorge Chacón

Ing. Yulio Alemán Rodríguez

Lic. María del Rosario Gutierrez Posada

La Habana, junio de 2019

“Año 59 de la Revolución”



“ No existe arma más potente que la convicción profunda y la idea clara de lo que debe hacerse. De ese tipo de armas que no requiere de fabulosas sumas de dinero, sino sólo de la capacidad de crear y transmitir ideas justas y valores. ”

Fidel Castro

Declaración de autoría

Declaro que yo Liana Maletá Terry, con carnet de identidad 96100314016 soy el único autor principal del trabajo titulado “Subsistema de recopilación automática de noticias de los medios informativos de la Universidad de las Ciencias Informáticas” y autorizo a la Universidad de las Ciencias Informáticas los derechos patrimoniales de este, con carácter exclusivo.

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

Liana Maletá Terry

Autor

Ing. Disnaye Jorge Chacón

Tutor

Lic. María del Rosario Gutierrez

Tutor

Ing. Yulio Alemán Rodríguez.

Tutor

Agradecimientos

Le agradezco a mis padres por haberme apoyado para estudiar la carrera y enseñarme a ser una mujer independiente, los amo mucho.

A mis hermanas Diana y Lianna por estar juntas en todos los momentos buenos y malos.

Al nuevo integrante de la familia Andrecito por brindarme su sonrisa y ser un niño maravilloso para mi.

Zuiero agradecerle a mis tutores Disnayle, Chavez, Rosario y Yulio por haberme guiado en esta etapa final de mi carrera.

A mis amistades Lisbet, Lisandra, Laura y Celia por haber tenido una infancia tan bella junto a ustedes, las extraño mucho.

A Mayelín que nunca pense que nos íbamos a llevar bien, mi pensamiento era que todo el tiempo estaríamos fajadas en el apartamento y que todo iba a ser un desastre, pero yo hice como dice el dicho ama tus amigos por sus defectos y no por sus virtudes. Eres muy buena amiga y no eres solo una amiga eres una hermana más.

A mi tías Nliana, Xiomara y Concha por haberme ayudado, apoyado y preocuparse por mi.

Agradecerle a toda mi familia por ayudarme en especial a mi abuela y a mi tía Natacha

Zuiero agradecerle a mi novio Felix, por quererme tanto, ser tan cariñoso, por ser mi amigo y compañero, por apoyarme en los momentos tan difíciles, por darme consejos, por quererme y aceptarme con mis defectos, te quiero mucho MyCosito.

Agradecerle a Vilma, Conchita y Nestor por acojerme en su familia y siempre preocuparse por mi escuela.

Por último agradecerle a Mily, a Barbón, a Nereida, a Mayde y a Roquito por ser una familia tan buena y siempre sacarme una sonrisa cuando estaba triste.

A todos muchas gracias.

Dedicatoria

*A mi mamá y mi papá por haberme dado la vida y la posibilidad de seguir adelante,
a ustedes le dedico este sueño hecho realidad.*

RESUMEN

El presente trabajo propone un subsistema de recopilación automática de noticias de los medios informativos de la Universidad de las Ciencias Informáticas (UCI). Se analizaron diferentes fuentes bibliográficas relacionadas con las características y la utilización de herramientas que están enfocadas en la recopilación de información a nivel nacional e internacional. La solución consiste en un subsistema capaz de recopilar noticias de forma automática obtenidas de los sitios de la UCI para el Departamento de Operaciones Web y Análisis de Información (DOWAI), debido a que la recopilación de información que se realiza en el departamento es de forma manual. El proceso de desarrollo estuvo guiado por la metodología de software AUP en su versión UCI, seleccionándose como principales tecnologías: el marco de trabajo Symfony 3.4, el lenguaje de programación PHP 7.2, el sistema gestor de base de datos PostgreSQL 10.4 y como herramienta para el modelado Visual Paradigm 8.0. Las pruebas de software aplicadas al subsistema de recopilación automática de noticias, demostraron que es una solución funcional, segura y con un rendimiento adecuado. Los resultados de la investigación científica para el subsistema desarrollado evidenciaron que posee un alto valor y satisface las necesidades existentes en el departamento DOWAI.

Palabras clave: noticias, sitios, recopilación de información, recopilación automática.

INTRODUCCIÓN	1
CAPÍTULO 1. Fundamentación teórica sobre el proceso de recopilación automático de noticias.	4
1.1 Conceptos Fundamentales	4
1.2 Proceso de extracción de información	5
1.2.1 Extracción de información desde la web	5
1.2.2 Técnicas para la extracción de información desde la web	6
1.3 Estudio de homólogos	6
1.4 Entorno de trabajo	9
1.4.1 Metodología de desarrollo de software	9
1.5 Lenguajes y herramientas a utilizar	10
Conclusiones parciales	15
CAPÍTULO 2. Análisis y diseño del subsistema de recopilación automática de noticias de los medios informativos de la UCI.	16
Introducción.	16
2.1 Propuesta de solución	16
2.2 Especificación de los requisitos de software	16
2.2.1 Requisitos funcionales	16
2.2.2 Requisitos no funcionales	18
2.3 Historia de Usuario	19
2.4 Análisis y diseño	22
2.4.1 Diseño Arquitectónico	22
2.4.2 Patrones de Diseño	24
2.5 Diseño de clases	25
2.5.1 Diagrama de Clases del Diseño	25
2.5.2 Diseño de datos	26
2.6 Diagrama de secuencia	27
2.7 Diagrama de despliegue	28
Conclusiones parciales	29
CAPÍTULO 3. Implementación y prueba del subsistema de recopilación automática de noticias de los medios informativos de la UCI.	30
3.1 Diagrama de componentes	30
3.2 Estándares de codificación	31

3.3 Interfaces.....	32
3.4 Validación del subsistema de recopilación automática de noticias	34
3.4.1 Pruebas funcionales.....	34
3.4.2 Pruebas de rendimiento	37
3.4.3 Pruebas de seguridad	39
3.5 Validación científica	40
Conclusiones parciales.....	42
CONCLUSIONES GENERALES	43
RECOMENDACIONES.....	44
BIBLIOGRAFÍA	45
ANEXOS	49
Anexo 1. Entrevista a especialistas del centro DOWAI para identificar las necesidades del cliente	49
Anexo 2. Entrevista a especialistas del centro SINI para conocer cómo funciona Red c.u.b.a.....	49
Anexo 3. Historias de Usuario	49
Anexo 4. Diseños de casos de pruebas.....	55
Anexo 5. Diagrama de clases.....	60
Anexo 6. Diagramas de secuencia	64
Anexo 7. Encuestas.....	72

ÍNDICE DE FIGURAS

Figura 1. Representación del Modelo Vista Controlador	23
Figura 2. Aplicación del patrón MVC del subsistema (Elaboración propia)	23
Figura 3. Código de la clase Registro.php (Elaboración propia)	24
Figura 4. Patrón GoF Decorador en una vista (Elaboración propia).....	25
Figura 5. Diagrama de clases de diseño de la HU_3 Gestionar sitio (Elaboración propia).....	26
Figura 6. Diagrama Entidad-Relación (Elaboración propia)	27
Figura 7. Diagrama de secuencia del RF_6 (Elaboración propia).....	28
Figura 8. Diagrama de despliegue del subsistema de recopilación automática de noticias (Elaboración propia).....	28
Figura 9. Diagrama de componente (Elaboración Propia).	30
Figura 10. Interfaz para ver las noticias recopiladas (Elaboración propia)	32
Figura 11. Interfaz con el listado de usuarios (Elaboración propia).....	33
Figura 12. Interfaz con el listado de sitios (Elaboración propia)	33
Figura 13. Comportamiento de las no conformidades para cada iteración de las pruebas funcionales (Elaboración propia)	37
Figura 14. Resultados de las pruebas de seguridad (Elaboración propia)	40
Figura 15. Diagrama de clase de la HU_1 Autenticar usuario (Elaboración propia).....	60
Figura 16. Diagrama de clase de la HU_2 Gestionar usuario (Elaboración propia)	61
Figura 17. Diagrama de clase de la HU_4 Gestionar categoría (Elaboración propia)	62
Figura 18. Diagrama de clases de la HU_6 Listar noticias (Elaboración propia)	63
Figura 19. Diagrama de clases de la HU_7 Mostrar noticias (Elaboración propia).....	63
Figura 20. Diagrama de clases de la HU_8 Clasificar noticia (Elaboración propia).....	64
Figura 21. Autenticar usuario (Elaboración propia).....	64
Figura 22. Insertar usuario (Elaboración propia).....	65
Figura 23. Eliminar usuario (Elaboración propia).....	65
Figura 24. Modificar usuario (Elaboración propia)	66
Figura 25. Mostrar usuario (Elaboración propia).....	66
Figura 26. Eliminar sitio (Elaboración propia)	67
Figura 27. Mostrar sitio (Elaboración propia)	67
Figura 28. Modificar sitio (Elaboración propia).....	68
Figura 29. Insertar categoría (Elaboración propia).....	68
Figura 30. Modificar categoría (Elaboración propia)	69
Figura 31. Mostrar categoría (Elaboración propia).....	70
Figura 32. Eliminar categoría (Elaboración propia).....	70
Figura 33. Listar noticia (Elaboración propia)	71

ÍNDICE DE FIGURAS

Figura 34. Mostrar noticia (Elaboración propia)	71
Figura 35. Clasificar noticias (Elaboración propia).....	72

ÍNDICE DE TABLAS

Tabla 1. Resumen del análisis de los sistemas homólogos (Elaboración propia)	8
Tabla 2. Requisitos Funcionales (Elaboración propia)	16
Tabla 3. HU_3 Gestionar sitio (Elaboración propia)	19
Tabla 4. HU_5 Recopilación automática (Elaboración propia)	20
Tabla 5. HU_7 Mostrar noticias (Elaboración propia)	21
Tabla 6. Descripción de las variables del caso de prueba de la HU_3 Gestionar sitio.	34
Tabla 7. Caso de prueba de HU_3 Gestionar sitio (Elaboración propia)	36
Tabla 8. Cantidad de no conformidades por cada iteración de las pruebas funcionales (Elaboración propia).....	37
Tabla 9. Resultado a partir de la prueba de carga y estrés (Elaboración Propia).....	38
Tabla 10. Cuadro lógico de ladov	40
Tabla 11. Resultados de las escalas de satisfacción (Elaboración propia)	41
Tabla 12. HU_1 Autenticar usuario (Elaboración propia)	49
Tabla 13. HU_2 Gestionar usuario (Elaboración propia).....	50
Tabla 14. HU_4 Gestionar categoría (Elaboración propia)	52
Tabla 15. HU_6 Listar noticias (Elaboración propia)	53
Tabla 16. HU_8 Clasificar noticias (Elaboración propia)	54
Tabla 17. Descripción de las variables del caso de prueba de la HU_1 Autenticar usuario (Elaboración propia).....	55
Tabla 18. Caso de prueba de la HU_1 Autenticar usuario (Elaboración propia)	56
Tabla 19. Descripción de las variables del caso de prueba de la HU_2 Gestionar Usuario (Elaboración propia).....	56
Tabla 20. Caso de prueba de la HU_2 Gestionar usuario (Elaboración propia).....	57
Tabla 21. Descripción de las variables del caso de prueba de la HU_4 Gestionar categoría (Elaboración propia).....	57
Tabla 22. Caso de prueba de la HU_4 Gestionar categoría (Elaboración propia).....	57
Tabla 23. Descripción de las variables del caso de prueba de la HU_6 Listar noticias (Elaboración propia).....	58
Tabla 24. Caso de prueba de la HU_6 Listar noticias (Elaboración propia)	58
Tabla 25. Descripción de las variables del caso de prueba de la HU_8 Clasificar noticias (Elaboración propia).....	59
Tabla 26. Caso de prueba de la HU_8 Clasificar noticias (Elaboración propia)	59

INTRODUCCIÓN

El avance de las Tecnologías de la Información y Comunicación (TIC), se ha convertido en un ingrediente esencial para la supervivencia del negocio y mejora la competitividad de las empresas, perfeccionando las actividades en la cadena de valor de una organización (Michael, 2015). Precisamente, estas agrupan un conjunto de sistemas para administrar la información, permitiendo fácil acceso a grandes fuentes de información, facilitando el procesamiento rápido y confiable de todo tipo de datos, así como la digitalización de toda la información.

Las TIC proporcionan penetración en diversos sectores de la sociedad: salud, educación, economía, calidad de imagen y/o sonido lo que da fiabilidad y fidelidad a la información transferida, e instantaneidad, facilita la rapidez de acceso e intercambio de la información. Sin duda alguna, el continuo desarrollo de las TIC ha traído consigo una mejora en la vida del hombre. Es por eso que en la actualidad se crean múltiples aplicaciones y surgen definiciones de sistemas dentro del mundo de la informática, como los sistemas de gestión de noticias (Cabero, 2013).

Diariamente el volumen de información crece a un ritmo acelerado en Internet, donde son publicadas múltiples noticias en diarios, revistas digitales y diversos Sistemas de Información (SI). Un SI es un sistema, automático o manual, que comprende personas, máquinas y/o métodos organizados para agrupar, procesar, transmitir y diseminar datos que representan información para el usuario (Longatt, 2012).

En Cuba existen diferentes organizaciones que se dedican al desarrollo de *software* como parte de la estrategia de informatización de la sociedad cubana. Una de ellas es la Universidad de las Ciencias Informáticas (UCI) cuya misión es formar profesionales, comprometidos con su Patria, calificados en la rama de la Informática, a partir de un modelo pedagógico flexible, que vincula dinámica y coherentemente el estudio con la producción y la investigación, acorde con las necesidades sociales del país y de otros pueblos hermanos (MES, 2014).

La UCI cuenta con 6 facultades, una de ellas es la Facultad 1 que posee tres centros productivos dedicados a la producción de software, uno de estos es el Centro de Ideoinformática (CIDI). En CIDI, se encuentra el Departamento de Operaciones Web y Análisis de Información (DOWAI), que utiliza la herramienta Sistema de Monitoreo de Análisis Estadístico (SIMAE), que está enfocada en el almacenamiento de noticias con el objetivo de monitorear y analizar las publicaciones de los sitios web de la UCI.

En SIMAE, los usuarios almacenan diariamente noticias de forma manual, trayendo consigo que se dificulte el trabajo con la misma, teniendo en cuenta que se maneja mucha información desde los

medios informativos. La aplicación no garantiza insertar varios autores a una misma publicación, utilizando como variante almacenarla a nombre de un solo autor o repetirla por la cantidad de autores que sean, lo que trae como consecuencia la duplicación y la sobrecarga de información en la base de datos. Otras de las deficiencias que presenta SIMAE es que está desarrollada con tecnología obsoleta, contiene alto nivel de dependencias y carece de documentación que respalde su desarrollo, lo que provoca que al ser desplegada no funcione correctamente.

A partir de la situación problemática anterior, se propone como **problema de investigación**: ¿Cómo contribuir a la recopilación automática de noticias de los medios informativos de la universidad?

Para dar solución al problema en cuestión se define como **objeto de estudio**: el proceso de recopilación de noticias que se encuentra enmarcado en el **campo de acción**: la recopilación automática de noticias de los medios informativos de la universidad.

Para dar respuesta al problema a resolver se plantea como **objetivo general**: Desarrollar un subsistema que contribuya al proceso de recopilación automática de noticias de los medios informativos de la universidad.

Preguntas Científicas:

1. ¿Cuáles son los referentes teóricos –metodológicos que sustentan la investigación sobre el proceso de recopilación automática de noticias?
2. ¿Cuál es el estado actual de la recopilación de noticias en los medios informativos de la universidad?
3. ¿Cuáles son los elementos a tener en cuenta para realizar el diseño del subsistema de recopilación automático de noticias?
4. ¿Cómo implementar el subsistema de recopilación automático de noticias?
5. ¿Qué resultados se obtienen una vez aplicada la propuesta del subsistema de recopilación automático de noticias en los medios informativos de la universidad?

Tareas de Investigación:

1. Determinación de los referentes teóricos – metodológicos que sustentan el proceso de recopilación automática de noticias.
2. Determinación del estado actual de la recopilación de noticias en los medios informativos de la universidad.
3. Diseño del subsistema de recopilación de noticias.
4. Implementación del subsistema de recopilación de noticias.

5. Valoración de los resultados de la propuesta del subsistema de recopilación automática de noticias en los medios informativos de la universidad.

Métodos Teóricos:

Histórico-lógico: Se utiliza para analizar y estudiar las características actuales, conceptos y evolución de los elementos relacionados con la recopilación de noticias.

Analítico-sintético: Se emplea realizar un estudio y análisis referente a las herramientas que realizan extracción de información, así como enunciar y describir la técnica que será utilizada para realizar la extracción de información para la aplicación a desarrollar.

Inductivo-Deductivo: Se utiliza para desarrollar razonamientos lógicos que permiten arribar a conclusiones generales a partir de premisas vinculadas al proceso de recopilación.

Métodos Empíricos:

Observación: Se aplicó en la indagación sistemática y el registro cuidadoso de las actividades realizadas por los miembros del departamento, lo que permitió comprender cómo transcurren las funciones de SIMAE.

Entrevista: Se realizaron entrevistas a los especialistas del centro CIDI en el departamento DOWAI que tienen experiencia de trabajado con la herramienta SIMAE con el propósito de conocer como la herramienta realiza el almacenamiento de noticias y tener en cuenta opiniones y sugerencias para la propuesta de solución que se va a desarrollar.

El presente trabajo está estructurado en 3 capítulos, a continuación, se describe el contenido de estos:

Capítulo 1: Fundamentación teórica sobre el proceso de recopilación automático de noticias.

En este capítulo se expone los elementos teóricos referentes al problema de investigación, estudio del estado del arte, así como tecnologías y herramientas para la solución del problema.

Capítulo 2: Análisis y diseño del subsistema de recopilación automática de noticias de los medios informativos de la UCI.

En este capítulo se expone el diseño y análisis para el subsistema a través de patrones de diseños, arquitecturas y diagramas que representarán la construcción de la propuesta de solución.

Capítulo 3: Implementación y prueba del subsistema de recopilación automática de noticias de los medios informativos de la UCI.

En este capítulo describe el proceso de desarrollo del software, implementación y validación de las funcionalidades de la propuesta de solución.

CAPÍTULO 1. Fundamentación teórica sobre el proceso de recopilación automático de noticias.

Introducción

En este capítulo serán presentados los principales conceptos y términos fundamentales que se utilizan como soporte teórico del subsistema que se va a diseñar e implementar. Será realizado el estudio de sistemas homólogos en el área de extracción de información, así como las herramientas, lenguajes de programación y tecnologías que serán empleadas en el desarrollo de la aplicación.

1.1 Conceptos Fundamentales

Para una mejor comprensión de la investigación se exponen un conjunto de conceptos importantes que están estrechamente relacionados con el dominio del problema.

La información

La información "es un conjunto de datos con un significado, o sea, que reduce la incertidumbre o que aumenta el conocimiento de algo. En verdad, la información es un mensaje con significado en un determinado contexto, disponible para uso inmediato y que proporciona orientación a las acciones por el hecho de reducir el margen de incertidumbre con respecto a nuestras decisiones" (Chiavenato, 2010).

Para resolver un determinado problema, se emplean diversas fuentes de información y se construye el conocimiento o la información organizada que permite aumentar el conocimiento de la persona y garantiza la toma de decisiones en la resolución de problemas. Esto equivale a que la información debe ser exacta, oportuna, completa y clara (Chiavenato, 2010).

Gestión de Información

La gestión de la información no es más que el proceso de organizar, evaluar, presentar, comparar los datos en un determinado contexto, controlando su calidad, de manera que esta sea veraz, significativa, exacta y útil y que esta información esté disponible en el momento que se le necesite. Ella se encamina al manejo de la información, documentos, metodologías, informes, publicaciones, soportes y flujos en función de los objetivos estratégicos de una organización (Ledo, 2017).

Las funciones de la gestión de la información abarcarían desde:

- Determinar las necesidades de información en correspondencia a sus funciones y actividades
- Mejora de los canales de comunicación y acceso a la información
- Mejora de los procesos informativos
- Empleo eficiente de los usuarios

Una adecuada gestión de los flujos de información en las organizaciones, constituye la base primaria, para contar con un adecuado Sistema de Gestión de Información (Arévalo, 2010).

Sistema de Gestión de Información

Según lo escrito por (Montejo, 2015), un Sistema de Gestión de Información (SGI) es un conjunto de sistemas y procedimientos que recopilan información de una variedad de fuentes, la compilan y la presentan en un formato legible. Los sistemas actuales de gestión de la información se basan en gran medida en la tecnología para recopilar y presentar datos.

El aumento considerable de la información que alcanzan los SGI, trae consigo que se pierda tiempo y esfuerzo en su búsqueda y análisis, lo que puede provocar demoras y errores en la toma de decisiones. Un SGI confiable, preciso y bien estructurado garantiza el éxito de sus consumidores. Para obtener resultados, es necesario un flujo eficiente e ininterrumpido de información que sustente los procesos institucionales para que esto fluya con libertad, a partir del quehacer de un personal con conocimientos necesarios para el desarrollo de actividades.

Recopilación Automática

La recopilación automática de datos se basa en recopilar datos sobre las actividades del sistema de forma periódica o para informar lo que se ha recopilado en los archivos de actividad diaria de un sistema u organización. Presenta como ventajas decir adiós a la introducción manual de datos, centralizar todas las fuentes de información, permitir el enriquecimiento de la información y la comprobación de la coherencia de los datos (Alvarado, 2017).

1.2 Proceso de extracción de información

En el siguiente epígrafe será abordado contenido acerca de la extracción de información, serán mencionadas varias técnicas de extracción de información y cuál de ellas será utilizada en la propuesta de solución.

1.2.1 Extracción de información desde la web

La extracción de información es el área de la ciencia y la tecnología que trata la identificación, clasificación y estructuración en clases semánticas de información específica encontrada en fuentes no estructuradas, para así permitir su posterior tratamiento automático en tareas de procesamiento de la información (Grisham, 2010).

Dada una colección de documentos, tiene como objetivo, identificar y extraer de los mismos aquellos hechos relevantes para un dominio particular, ignorando la información extraña e irrelevante.

1. La información obtenida se devuelve de forma estructurada
2. Ha de establecerse a priori que constituye un hecho/relación.
3. Sistemas muy especializados de dominio acotado.
4. El tratamiento automático de la información facilita a los usuarios la manipulación, evaluación y

utilización de grandes cantidades de documentos, tarea que mediante técnicas completamente manuales no se podría realizar. Esta situación se encuentra afectada por la alta disponibilidad de documentos en Internet y la existencia de fuentes de información “en línea”, como por ejemplo las agencias de noticias y periódicos digitales (Grisham, 2010).

1.2.2 Técnicas para la extracción de información desde la web

La extracción de información es cada día más notable en las empresas e instituciones que utilizan abundante información y como es consecuente resulta muy tedioso y aburrido estar copiando y pegando desde la base de datos dicha información. Pero la tecnología ha avanzado considerablemente y para estos casos se utilizan varias técnicas de extracción de información como la técnica web scraping y la utilización de spiders. A continuación, estas técnicas serán descritas con el objetivo de conocer en que consiste cada una de ellas:

-*Web scraping* es una técnica de programación para la extracción de información de sitios web. Por lo general, este tipo de programas simulan la exploración humana de la Web, ya sea implementado a bajo nivel del Protocolo de Transferencia de Hipertexto (HTTP, Hyper Text Transfer Protocol), o incluido en ciertos navegadores web, como el Internet Explorer (IE) o el navegador web Mozilla. *Web-scraping* está estrechamente relacionado con la indexación web, indexa todo el contenido de la *Web* mediante un robot y es una técnica universal adoptada por la mayoría de los motores de búsqueda. *Web-scraping* se centra en la transformación de contenido web no estructurado, por lo general en formato HTML, en datos estructurados que pueden ser almacenados y analizados en una base de datos local o de hoja de cálculo. *Web-scraping* también está relacionado con la automatización web, que simula la navegación humana *web* utilizando el *software* de computadora (Micó, 2014).

-*Spiders* o arañas que inspeccionan las páginas web de internet de forma metódica y automatizada son utilizadas para rastrear la red. Lee la estructura de hipertexto y accede a todos los enlaces referidos en el sitio web. Son utilizadas la mayoría de las veces para poder crear una copia de todas las páginas web visitadas para que después puedan ser procesadas por un motor de búsqueda; esto hace que se puedan indexar las páginas, proporcionando un sistema de búsquedas rápido. (Hernández, 2015).

A partir de las técnicas de extracción de información estudiadas, se va a utilizar *Web scraping* con el objetivo de obtener las noticias de los sitios de la UCI a partir de la información que brinda el HTML de los sitios.

1.3 Estudio de homólogos

En el mundo se han desarrollado diferentes aplicaciones que recopilan información de manera automatizada. En la investigación se realiza un análisis de estas soluciones con el fin de obtener una

aproximación de las funcionalidades que debe tener la solución. A continuación, se detallan sus principales características y se hace una valoración acerca de los resultados obtenidos.

Análisis de herramientas informáticas existentes

Import. IO

Import.io es un *scraper online* gratuito que permite recolectar diferentes tipos de datos y organizar la información obtenida en tablas. Posee una interfaz moderna y facilidades de uso tanto para usuarios novatos como para expertos, con un sistema de ayuda que guía todo el proceso de scraping. Contiene una amplia gama de funcionalidades útiles para el consumo y reutilización de la información, siendo una de las herramientas gratuitas más completas y novedosas, ya que fue fundada en 2012 y lanzada al mercado en 2013.

El objetivo principal de esta aplicación es facilitar el proceso de extracción, almacenamiento y reutilización de datos, ofreciendo un sinfín de características impresionantes en este sentido. Sin embargo, no brinda opciones avanzadas para el consumo de información, teniendo como principales limitantes:

- No posibilita los boletines electrónicos personalizados a partir del contenido recolectado.
- Carece de opciones para aplicar filtros avanzados al contenido.
- Además de las limitantes en cuanto a funcionalidades, es necesario señalar que se trata de una aplicación aún en fase de desarrollo, que puede presentar errores o comportamientos inesperados ocasionalmente (Afday, 2017).

Kimono

Kimono es un servicio de web scraping en línea que permite crear APIs a partir de sitios web. Fue lanzado en enero de 2014, y aún se encuentra en desarrollo. Es un proyecto muy prometedor, que permite recuperar la información recolectada en varios tipos de formatos como JSON¹, CSV² y RSS³. Kimono permite recuperar versiones anteriores de los datos, enviar alertas sencillas por correo electrónico, y una gran cantidad de opciones para configurar y gestionar las APIs creadas. Es una herramienta bastante completa, presentando algunas limitaciones, que son señaladas a continuación:

- Carece de opciones para aplicar filtros avanzados de contenidos.

¹ JavaScript Object Notation, es un formato ligero de intercambio de datos, donde leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y generarlo.

² Comma-Separated Values, es un archivo de texto que almacena los datos en forma de columnas y es una forma muy sencilla de representar la información.

³ Es un sencillo formato de datos que es utilizado para redifundir contenidos a suscriptores de un sitio web.

-Dependencia de conectividad a Internet para obtener los datos.

-Su versión profesional tiene un costo de \$ 168 (Afday, 2017).

Buscador de la plataforma del Contenido unificado para la búsqueda avanzada (c.u.b.a)

La plataforma c.u.b.a creada a partir del año 2015 como continuación del motor de búsqueda Orión, es un sistema cuyo propósito fundamental es proveer a la red nacional de una herramienta principal para la búsqueda y análisis de contenidos cubanos. Implementada con varias herramientas libres como: Nutch, en su versión 1.9 para el proceso de rastreo; Solr, en su versión 6.3 para el componente de indexación; Symfony, en su versión 2.8; MongoDB y MySQL, como gestores de base de datos para almacenar estadísticas; Java, versión 8.0 para el manejo de plugins en Nutch y Solr, HTML 5 y CSS 3 para la interfaz web; PHP en su versión 7.0 y Bootstrap en su versión 4.0. Está enfocada en la actualidad en la búsqueda simple, avanzada de contenido y realiza varios servicios como web scraping dirigidos a varios sitios. La plataforma c.u.b.a enlaza el servicio de web scraping a sitios ajenos a la universidad y sitios que se encuentran disponibles en internet como: la enciclopedia EcuRed, el sitio Cubadebate, la cartelera cultural La Papeleta, la plataforma de blogs Reflejos, entre otros.

Sistema de Monitoreo de Análisis Estadístico (SIMAE)

SIMAE es una herramienta que permite almacenar la información referente a las publicaciones de sitios web y blogs internos de la UCI, para realizar un estudio de tendencias sobre los principales temas tratados a través de la red universitaria. En esta herramienta todos los procesos que se realizan son de forma manual como crear noticias, listar noticias, generar reportes a partir de las noticias registradas y clasificar noticias de forma semiautomática.

En la siguiente tabla se realiza una comparación entre herramientas antes mencionadas que están dedicadas a la recopilación de información. Los indicadores que se tuvieron en cuenta para realizar la comparación están basados a partir de las características de las mismas.

Tabla 1. Resumen del análisis de los sistemas homólogos (Elaboración propia)

Indicadores/ Herramientas	Import.lo	Kimono Labs	c.u.b.a	SIMAE
Extracción de información a través de URL	si	si	si	no
Técnica web scraping	si	si	si	no

Formato de los datos (JSON,HTML,RSS)	no	si	no	no
Licencia	pública	privada	pública	pública

Teniendo en cuenta el estudio realizado se concluye que las mismas no satisfacen las necesidades del cliente debido a que Import.IO y Kimono no cuentan con un nivel de madurez suficiente porque aún se encuentran en desarrollo. La herramienta Import.IO devuelve los datos en tabla lo que resulta engorroso realizar el procesamiento de la información; la herramienta Kimono tiene como desventaja que no es gratis y tiene total dependencia de conectividad a Internet, la plataforma c.u.b.a muestra información de sitios que son ajenos de la universidad y que son públicos en internet y SIMAE es una herramienta que recopila noticias, pero de forma manual. Luego de haber realizado el análisis se concluyen que estas herramientas no satisfacen las necesidades del departamento, pero serán utilizadas como guía de referencia para el desarrollo de la propuesta de solución, debido a que Import, Kimono y c.u.b.a realizan la recopilación automática de noticias utilizando la técnica web scraping y será utilizado la clasificación semiautomática de noticias como se realiza en la herramienta SIMAE.

1.4 Entorno de trabajo

1.4.1 Metodología de desarrollo de software

Es un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con altas posibilidades de éxito. Una metodología de desarrollo de software comprende actividades a seguir para idear, implementar y mantener un producto de software desde que surge la necesidad del producto hasta que se cumple el objetivo por el cual fue creado (Flores, 2017).

Proceso unificado ágil (AUP-UCI)

La metodología seleccionada para el desarrollo de la propuesta de solución es AUP en su variación para la UCI, la cual fue elaborada en la Universidad de las Ciencias Informáticas. Esta metodología logró estandarizar el proceso de desarrollo de software en la universidad, dando cumplimiento a las buenas prácticas y logrando un lenguaje común en cuanto a fases, disciplinas, roles y productos de trabajos. El software será desarrollado en el escenario 4 que se aplica a los proyectos que hayan evaluado el negocio a informatizar y como resultado obtengan un negocio muy bien definido. El cliente estará siempre acompañando al equipo de desarrollo para convenir los detalles de los requisitos y así poder implementarlos, probarlos y validarlos; además se recomienda en proyectos no muy extensos, ya que una Historia de Usuario (HU) no debe poseer demasiada información (UCI, 2014).

1.5 Lenguajes y herramientas a utilizar

Lenguaje unificado de modelado (UML 2.0)

El Lenguaje Unificado de Modelado (UML) prescribe un conjunto de notaciones y diagramas estándares para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan; posibilitando así visualizar, especificar y documentar los artefactos o toda información que se obtiene o modifica durante un proceso de desarrollo de software, además de poder utilizarse para modelar distintos tipos de sistemas de *software*, *hardware* y organizaciones del mundo real (UML, 2010).

Principales características (UML, 2010):

- Permitir modelar sistemas haciendo uso de técnicas orientada a objetos.
- Permite especificar todas las decisiones de análisis y diseño, construyendo así modelos precisos, no ambiguos y completos.
- Puede conectarse con lenguajes de programación (Ingeniería de directiva e inversa).

1.5.1 Herramienta para el modelado a utilizar

Visual Paradigm 8.0

Visual Paradigma para UML 8.0 es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una rápida construcción de aplicaciones de calidad, haciéndolas mejores y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta proporciona abundantes tutoriales, demostraciones interactivas y proyectos de UML (UML, 2014).

1.5.2 Tecnología para la implementación del subsistema

A continuación, se describe las tecnologías que serán utilizadas para la implementación de la propuesta de solución.

Lenguaje de programación

Lenguaje de programación es un conjunto de reglas y estándares que es utilizado para escribir programas de computador (software), que puedan ser entendidos por él (Hernández, 2011).

Pre-Procesador de Hipertexto PHP 7.2

PHP es el acrónimo recursivo del inglés Hypertext Pre-processor (Pre-procesador de hipertextos), es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación.

Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor web, justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red y otras tareas para crear la página final que verá el cliente. El cliente solamente recibe una página con el código HTML resultante de la ejecución del PHP. Como la página resultante contiene únicamente código HTML, es compatible con todos los navegadores (Claudio, 2016).

Según el autor antes mencionado, a continuación, se detallan las siguientes características:

- Es un lenguaje multiplataforma.
- Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- El código fuente escrito en PHP es invisible al navegador web y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.

1.5.3 Entorno de desarrollo integrado (IDE):

PhpStorm 8.0

PhpStorm es un Entorno de Desarrollo Integrado (IDE) inteligente para desarrollar aplicaciones en Preprocesador de hipertextos (PHP), proporcionando herramientas esenciales como refactorización, análisis de código y comprobación de errores. Las principales novedades en PhpStorm 8.0 incluyen

- Soporte para PHP 5.3, incluyendo espacios de nombre.
- Depuración con cero configuraciones con todos los navegadores.
- Editores de Lenguaje de Consulta Estructurado (SQL) con resultados editables.
- Soporte para Lenguaje Marcado de Hipertextos (HTML5) (Posada, 2010)

1.5.4 Herramienta de prueba de software

Apache Jmeter

Jmeter es una herramienta de carga para llevar a cabo simulaciones sobre cualquier recurso de *software*. Inicialmente diseñada para pruebas de estrés en aplicaciones web, su arquitectura ha evolucionado no sólo para llevar a cabo pruebas en componentes habilitados en Internet (HTTP), sino además en Bases de Datos, programas en Perl, requisiciones FTP y prácticamente cualquier otro medio. Además, posee la capacidad de realizar desde una solicitud sencilla hasta secuencias de

requisiciones que permiten diagnosticar el comportamiento de una aplicación en condiciones de producción. JMeter es una herramienta de testing cuyas funcionalidades se pueden resumir en tres:

- Diseñar un testplan, esto es, generar un fichero.jmx
- Ejecutar un testplan.
- Ver de distintas formas los resultados de la ejecución de un testplan (Sánchez, 2018).

Acunetix Web Vulnerability Scanner 9.5

Acunetix es una herramienta de seguridad de aplicaciones Web automatizada. Acunetix WVS es capaz de escanear cualquier sitio Web o aplicación Web que es accesible a través del protocolo HTTPS. Sin embargo, no todas las pruebas se puede realizar de forma automática, y por lo tanto Acunetix WVS proporciona herramientas de Penetración manuales para pruebas particulares.

- Acunetix es una herramienta automatizada de pruebas de seguridad de aplicaciones *web*
- Comprueba diferentes vulnerabilidades (por ejemplo inyección de SQL, Cross Site Scripting). Hasta la fecha Acunetix comprueba sobre más de 500 tipos diferentes de vulnerabilidades.
- Acunetix puede escanear cualquier sitio Web que es accesible a través del protocolo HTTPS, básicamente, si el sitio Web se puede ver en un navegador, Acunetix puede escanearlo.
- La herramienta también proporciona herramientas de pruebas de penetración manuales que aumentan y contribuyen a las pruebas automatizadas, así como ayudar con la prueba de vulnerabilidades lógicas, (Armas, 2017).

1.5.5 Herramientas de maquetación web

Lenguaje de Marcas de Hipertexto HTML 5

Lenguaje de Marcado de Hipertextos (HTML, por sus siglas en inglés), es el lenguaje que se utiliza para crear todas las páginas web de Internet. Es un lenguaje reconocido universalmente y que permite publicar información de forma global. Define una estructura básica y un código HTML para la definición de contenido de una página web, como texto, imágenes, entre otros y se basa en la referenciación por hipertextos o enlaces entre páginas (Lapuente, 2013).

Entre las ventajas que ofrece HTML 5 se encuentran:

- Sin conexión y almacenamiento: Permite a las páginas web almacenar datos localmente en el lado del cliente y operar sin conexión de manera más eficiente.
- Multimedia: Otorga un excelente soporte para utilizar contenido multimedia como lo son audio y video nativamente.
- Rendimiento e Integración: Proporciona una mayor optimización de la velocidad y un mejor uso del *hardware* (Cordero, 2017).

Hojas de Estilo en Cascada CCS3

CSS es un lenguaje de hojas de estilos en cascada (en inglés Cascading Style Sheets) creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. Este lenguaje ofrece una nueva gran variedad de opciones para hacer diseños más sofisticados, mejora la accesibilidad de los documentos, reduce la complejidad de su mantenimiento y permite visualizar los mismos documentos en infinidad de diferentes dispositivos. Se utiliza para definir el aspecto de cada elemento: color, tamaño y tipo de letra de texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, entre otros (Cordero, 2017).

Bootstrap 3

Bootstrap es un marco de trabajo de software libre, intuitivo y de gran alcance para el diseño de sitios y aplicaciones web. En la actualidad se ha convertido en uno de los proyectos de código abierto de gran popularidad en el mundo. Desarrollado por Twitter, Bootstrap simplifica el proceso de creación de diseños web combinando CSS y JavaScript. La mayor ventaja es que se puede crear interfaces que se adapten a los distintos navegadores con la propiedad denominada diseño adaptativo. Los diseños creados con Bootstrap son simples, limpios e intuitivos, dándole agilidad a la hora de cargar y adaptarse a otros dispositivos como tabletas y móviles a distintas escalas y resoluciones (Thornton, 2013).

1.5.6 Sistema Gestor de Base de Datos

PostgreSQL 10.4

Herramienta de código abierto para la administración de bases de datos PostgreSQL. Diseñada para responder a las necesidades de los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. La aplicación incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor, un agente para lanzar comandos programados y soporte para la replicación. La conexión al servidor puede hacerse mediante la familia de protocolos de internet (TCP/IP por sus siglas en inglés) y encriptarse mediante el protocolo de Capa de Conexión Segura (SSL por sus siglas en inglés) para mayor seguridad (Martínez, 2015).

1.5.7 Servidor de aplicación web

Nginx 1.14

Nginx es un servidor web de alto rendimiento, capaz de trabajar junto con diversas tecnologías de desarrollo y lenguajes. La asincronía es su característica fundamental, junto con su rapidez, ya que es un servidor web ligero. Sirve para páginas con programación del lado del servidor, con lenguajes como PHP, aunque es muy utilizado también para servir archivos estáticos. La principal ventaja de Nginx como servidor es que consume mucho menos recursos al servir contenido estático y esto convierte a Nginx en una excelente opción para funcionar como proxy inverso o como balanceador de carga. Lo

que hace que Nginx sea diferente a otros servidores web es su arquitectura, que permite responder a millones de peticiones por segundo aprovechando al máximo los núcleos e hilos de ejecución del servidor con una configuración muy simple (Pablo, 2018).

1.5.8 Marco de trabajo

En el desarrollo de un software, un marco de trabajo, o *framework* en inglés, es una estructura de soporte definida en la cual otro proyecto de software puede ser organizada y desarrollada. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros softwares para ayudar a desarrollar y unir los diferentes componentes de un proyecto (Eguiluz, 2015).

Symfony 3.4

Es un marco de trabajo que ayuda a simplificar el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Además, proporciona estructura al código fuente, forzando al desarrollador a crear código legible y fácil de mantener. Por último, facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas. Symfony es un completo marco de trabajo diseñado para optimizar, gracias a sus características, 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. Entre las características destacadas que ofrece a los desarrolladores de productos se encuentran las siguientes:

- Fácil de instalar y configurar en la mayoría de plataformas.
- Independiente del sistema gestor de bases de datos.
- Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
- Sigue la mayoría de las buenas prácticas de Symfony y la utilización de patrones de diseño para la web (Eguiluz, 2015).

Symfony posee facilidades debido a la gran cantidad de documentación y tutoriales que se pueden encontrar en la web, se adapta a cualquier plataforma, permite elegir la base de datos, es ideal para gestionar todas las consultas y peticiones al servidor, así como mostrar las respuestas que la lógica de negocio requiera.

Conclusiones parciales

Al realizar un análisis del marco teórico se logró organizar y guiar el trabajo, profundizando en el estudio de conceptos fundamentales para conocer sobre la recopilación automática y la técnica *web scraping* que será utilizada para extraer información a partir del HTML de un sitio. El estudio realizado de las herramientas existentes permitió determinar que las mismas no satisfacen las necesidades del usuario, pero servirán de guía para el desarrollo de la aplicación. Se desarrolló un entorno de trabajo a partir de la metodología, tecnologías, herramientas y lenguaje de programación para la implementación de la propuesta de solución.

CAPÍTULO 2. Análisis y diseño del subsistema de recopilación automática de noticias de los medios informativos de la UCI.

Introducción.

En este capítulo se realiza una descripción de las características del subsistema, las cuales son el punto de partida para comenzar el proceso de desarrollo. Estas permiten tener una visión objetiva acerca del producto que se desea obtener. Además, se identifican los requisitos funcionales y no funcionales que debe cumplir el subsistema. Se hace alusión al análisis y diseño del subsistema que se va a desarrollar y se realiza el modelado de los artefactos necesarios para la implementación de la propuesta de solución.

2.1 Propuesta de solución

Se propone un subsistema de recopilación automática de noticias, con el objetivo de centralizar las noticias actualizadas que se adquieren a partir de los diferentes sitios de la universidad. Para realizar la recopilación automática será utilizada la técnica web scraping, que permite obtener de las noticias el título, resumen, autor, cuerpo de la noticia, la cantidad de visitas y comentarios, ejecutándose la recopilación de forma automática cada una hora. La recopilación será iniciada cuando se realice la autenticación en el subsistema con un usuario y contraseña de dominio UCI, y a medida que se haga la recopilación será insertada en la base de datos todas las noticias de los sitios de la universidad. El usuario a través de una interfaz web podrá visualizar las noticias recopiladas, realizar la clasificación de las noticias, así como visualizar los sitios de donde provienen estas noticias

2.2 Especificación de los requisitos de software

La especificación de requisitos, es la base que permite verificar si se alcanzaron o no los objetivos establecidos en el proyecto, debido a que son un reflejo detallado de las necesidades del cliente. Facilita un mejor entendimiento de los procesos a desarrollar, permitiendo comprender con profundidad el problema en cuestión y facilitando una mejor identificación de las funcionalidades que serán implementadas (Pressman, 2010).

2.2.1 Requisitos funcionales

Los requisitos funcionales reflejan lo que debe hacer y de qué manera debe reaccionar el software ante las entradas y cómo debe comportarse en situaciones específicas. Estos detallan la función del producto del software, entradas, salidas excepciones y usuarios (Pressman, 2010).

Tabla 2. Requisitos Funcionales (Elaboración propia)

No	Nombre	Prioridad para el cliente	Complejidad
-----------	---------------	----------------------------------	--------------------

RF1	Autenticar usuario	Media	Baja
RF2	Insertar usuario	Media	Baja
RF3	Eliminar usuario	Media	Baja
RF4	Modificar usuario	Media	Media
RF5	Mostrar usuario	Media	Media
RF6	Insertar sitio	Alta	Alta
RF7	Eliminar sitio.	Baja	Baja
RF8	Modificar sitio.	Alta	Alta
RF9	Mostrar sitio	Baja	Baja
RF10	Insertar categoría.	Baja	Baja
RF11	Eliminar categoría.	Baja	Baja
RF12	Modificar categoría.	Baja	Baja
RF13	Mostrar categoría	Baja	Baja
RF14	Extraer código html de las páginas que contienen las noticias.	Alta	Alta

RF15	Extraer contenido de los campos de las publicaciones.	Alta	Alta
RF16	Listar noticias	Baja	Media
RF17	Mostrar noticias	Baja	Baja
RF18	Clasificar noticia	Baja	Baja

2.2.2 Requisitos no funcionales

Los requisitos no funcionales (RnF) son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. A menudo son aplicadas en su totalidad al sistema y normalmente apenas se aplican a características o servicios individuales (Pressman, 2010).

Software

RnF 1. Se requiere la instalación del sistema operativo Linux Mint en su versión 19.

RnF 2. Se requiere la instalación del servidor de base de datos Postgresql en su versión 10.4 en adelante.

RnF 3. Se requiere la instalación del servidor web Nginx 1.14.

RnF 4. Se requiere la instalación de PHP en su versión 7.2.

Hardware

RnF 5. El servidor de base de datos debe poseer como mínimo un disco duro de 50 GB, 4 GB de RAM y un núcleo con velocidad de procesamiento a 2.20GHZ.

RnF 6. El servidor de aplicación requiere como mínimo 10 GB de disco duro, 4 GB de RAM y un núcleo con velocidad de procesamiento a 2.20GHZ.

Disponibilidad

RnF 7. Solo los usuarios autorizados tendrán acceso la recopilación de la información.

Eficiencia

RnF 8. La aplicación *web* debe poseer un tiempo de respuesta por transacción de hasta 5 segundos en dependencia de la colección datos, el número de usuarios realizando peticiones y el servidor donde se encuentre instalado.

RnF 9. La aplicación *web* debe permitir que los usuarios interactúen de manera concurrente.

Licencia

RnF 10. Se requiere el uso de herramientas y *software* libre, las cuales se podrán usar, modificar y distribuir libremente.

2.3 Historia de Usuario

Las Historias de Usuario (HU) son una forma rápida de describir los requisitos de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin tener que requerir de mucho tiempo para administrarlos (Cobo, 2017).

En el desarrollo del trabajo investigativo se crearon 8 historias de usuarios de las que se serán mostradas 3 de ellas.

Tabla 3. HU_3 Gestionar sitio (Elaboración propia)

Historia de usuario	
Número: HU_3	Nombre: Gestionar sitio.
Programador: Liana Maletá Terry	Iteración asignada: 1
Prioridad en negocio: Alta	
<p>Descripción: La interfaz web principal de la aplicación contiene un Listado de sitios para insertar, mostrar, eliminar y modificar los sitios para realizar la recopilación de las noticias, que hacen referencia a los requisitos funcionales <i>RF6</i>, <i>RF7</i>, <i>RF8</i> y <i>RF9</i>.</p> <p>Antes de insertar un sitio el usuario puede poner en el buscador el nombre del sitio para saber si ya existe y si no se encuentra debe presionar la opción Insertar sitio y le será presentado un formulario para llenar los campos de un sitio. Luego se comprueba que los datos cumplan con los requisitos necesarios para que sea válido y es inserta el sitio. Teniendo en cuenta la alta complejidad que tiene este requisito para llenar los campos, se decidió realizar un Manual de usuario para que el usuario conozco los datos que debe poner en los campos.</p> <p>Para mostrar un sitio se escoge la opción Mostrar de acuerdo al sitio que se quieran ver las características y luego se presiona el botón Volver al listado.</p> <p>Para eliminar un sitio se escoge la opción mostrar, luego aparece una interfaz con los datos del sitio, se hace hacer clic en el botón Eliminar y le aparecerá una nueva interfaz para confirmar la eliminación del sitio y presiona el botón Eliminar.</p>	

Para modificar un sitio, se selecciona el sitio a modificar y se hace en el botón Editar y se mostrará un formulario para editar el campo que desee y luego se debe presionar el botón Guardar.

Prototipo de interfaz:



Tabla 4. HU_5 Recopilación automática (Elaboración propia)

Número: HU_5	Nombre: Recopilación automática
Programador: Liana Maletá Terry	Iteración asignada: 1
Prioridad en negocio: Alta	
<p>Descripción: La historia de usuario Recopilación automática se basa en la recopilación automática de noticias de varios sitios de la universidad, la misma describe los requisitos RF_14 y RF_15.</p> <p>Para que se lleve a cabo la recopilación automática de las noticias se creó un servicio que contiene varios métodos uno de ellos es getUrl (\$url) donde se le pasa por parámetros una variable php que contiene la url de un sitio y permite extraer el código html de las páginas que contiene las noticias; luego de obtener el html de un sitio se va a extraer el contenido de los campos de las publicaciones que fue implementado en el método Obtenerdatos, que se encargará de recoger del html de un sitio cada uno de los campos de las noticias y devuelve un arreglo de noticias de cada uno de los sitios y por último se creó el método Insertarbd que es el encargado de insertar en la base de datos del subsistema las noticias que devuelve el método Obtenerdatos. Con la implementación de estos métodos en el servicio se creó un comando con el nombre Comand para que del servicio creado se ejecute el método Insertarbd cada una hora.</p>	

Tabla 5. HU_7 Mostrar noticias (Elaboración propia)

Historia de usuario	
Número: HU_7	Nombre: Mostrar noticias
Programador: Liana Maletá Terry	Iteración asignada: 1
Prioridad en negocio: Baja	
<p>Descripción:</p> <p>El usuario autenticado va a escoger la url y el rango de fecha para obtener un listado de noticias y para poder visualizar completamente una noticia se hace clic sobre el título de la noticia que representa un link y será mostrado otra página con el resto de los datos de la noticia como son el título, redactor, fecha de publicación, resumen, cuerpo de la noticia, cantidad de visitas, vínculo de la noticia cantidad de comentarios y los comentarios.</p>	
Prototipo de interfaz:	



Título	Redactor	Cuerpo	Categoría	Cantidad Visitas	Cantidad Comentarios
Venezuela y Cuba	Joaquin Perez	Cuba y Venezuela...	Política	0	0

2.4 Análisis y diseño

El diseño del sistema permite determinar cómo funcionará el producto final de forma general sin entrar en detalles incorporando consideraciones de la implementación tecnológica. Consiste en el diseño de los componentes del sistema que dan respuesta a las funcionalidades y requisitos descritos en la etapa de especificación. Generalmente se realiza en base de diagramas que permitan describir las interacciones entre las entidades y su secuenciado, impone una estructura del sistema (Jacobson, 2018).

2.4.1 Diseño Arquitectónico

La arquitectura de software es un conjunto de decisiones acerca de la organización de un sistema de software, consiste en la selección de elementos estructurales a partir de los cuales se compone el sistema. La arquitectura de software se interesa no solo por la estructura y el comportamiento, sino también por las restricciones y compromisos de uso, funcionalidad, flexibilidad al cambio, reutilización, comprensión y tecnología, así como por aspectos estéticos (Jacobson, 2016).

Patrón arquitectónico Modelo-Vista-Controlador (MVC)

La propuesta de solución se encuentra desarrollada en Symfony, cuyo funcionamiento interno se establece en la arquitectura Modelo-Vista-Controlador (MVC). Este patrón permite dividir la aplicación en tres capas:

-Vista: Presenta el modelo en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho modelo la información que debe representar como salida.

-Modelo: Es la representación de la información con la cual el sistema opera, por lo tanto, gestiona todos los accesos a dicha información, tanto, consultas como actualizaciones, implementado también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación (lógica de negocio). Envía a la vista aquella parte de la información que en cada momento se le solicita para que sea mostrada. Las peticiones de acceso o manipulación llegan al modelo a través del controlador.

-Controlador: Responde a eventos e invoca peticiones al modelo cuando hace alguna solicitud sobre la información. También puede enviar comandos a su vista asociada si se solicita un cambio en la forma

en que se presenta el modelo por tanto se podría decir que el controlador hace de intermediario entre la vista y el modelo (Sommerville, 2011).

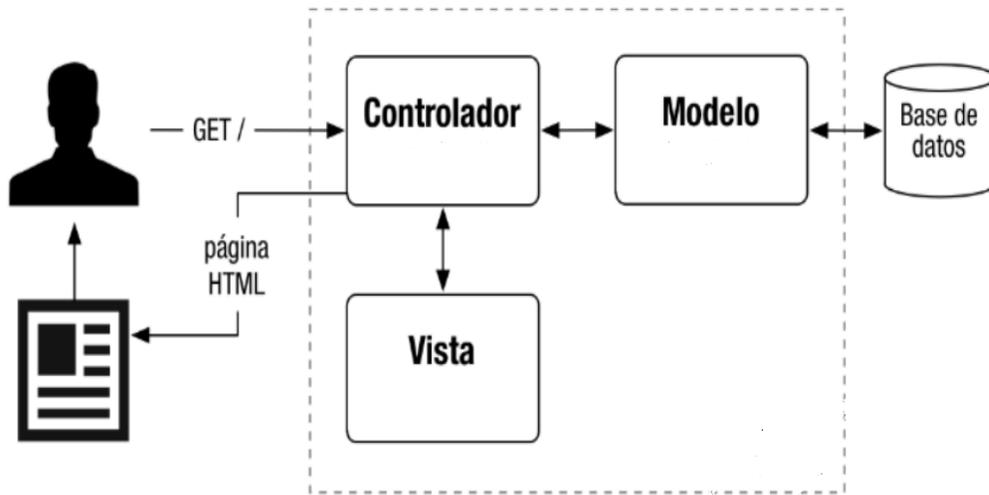


Figura 1. Representación del Modelo Vista Controlador (Eguiluz, 2017)

La figura 2 representa la separación de las clases de la aplicación según el marco de trabajo Symfony 3.4; en el paquete Controlador se encuentran las clases controladoras Sitio, Categoría, Usuario y GestRecopilacion, en el paquete Modelo se encuentra las entidades Registro.php, Publicación.php, Comentario.php, Categoría.php y Usuario.php que se corresponden con las tablas que se encuentran en la base de datos de la aplicación y en el paquete Vista se encuentra index.html.twig, login_content.html.twig, base.html.twig y mostrar_noticia.html.twig.

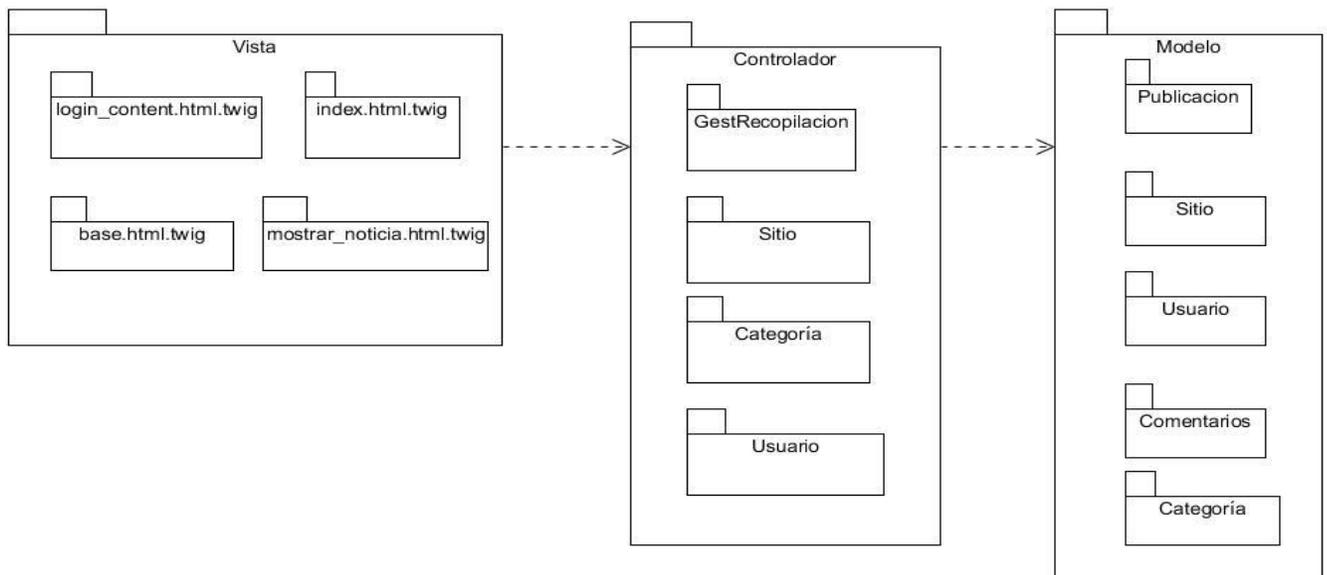


Figura 2. Aplicación del patrón MVC del subsistema (Elaboración propia)

2.4.2 Patrones de Diseño

Los patrones de diseño representan la descripción de un problema particular y recurrente, que aparece en contextos específicos, y presenta un esquema genérico demostrado con éxito para su solución; este último se especifica mediante la descripción de los componentes que la constituyen, sus responsabilidades y desarrollos, así como también la forma como estos colaboran entre sí (Larman, 2003).

Patrones GRASP

En el diseño del subsistema de recopilación automática de noticias se tuvieron en cuenta los patrones GRASP (Patrones Generales de Software para Asignación de Responsabilidades), que describen los principios fundamentales de la asignación de responsabilidades a objetos:

- **Experto:** determina cual es la clase que debe asumir una responsabilidad a partir de la información que posee cada una (Borges, 2017). Este patrón se evidencia en la clase Registro.php, ya que contiene las funcionalidades necesarias para acceder a la información de un sitio.

```
class Registro
[
/**
 * @var int
 *
 * @ORM\Column(name="id"
 * @ORM\Id
 * @ORM\GeneratedValue(s
 */
private $id;

/**
 * @var string
 *
 * @ORM\Column(name="por
 */
private $portTitulo;
```

Figura 3. Código de la clase Registro.php (Elaboración propia)

-**Alta cohesión:** se basa en asignar a las clases responsabilidades que trabajen sobre una misma área de la aplicación y que no tengan mucha complejidad, evitando así que una clase sea la única responsable de muchas tareas en áreas funcionales muy heterogéneas (Borges, 2017).

Symfony permite la organización del trabajo en cuanto a la estructura del proyecto y la asignación de responsabilidades con una alta cohesión. Un ejemplo de ello son las clases RegistroType y CategoriaType, que son las responsables de crear los formularios para las entidades Sitio y Categoría.

-Bajo acoplamiento: se basa asignar responsabilidades de forma tal que cada clase se comuniquen con el menor número de clases, minimizando el nivel de dependencia (Borges, 2017). Esto se evidencia en la clase GestRecopilacionController que se encarga de listar las noticias guardadas en base de datos y mostrar el contenido de las noticias sin depender de otra clase.

Patrones GoF

Decorador: Se pone en práctica en las vistas del subsistema realizado en Symfony. La plantilla base.html.twig contiene el diseño común para todo el sistema que se realiza a través de la herencia entre plantillas. Se representa en la siguiente figura:

```
{% extends 'base.html.twig' %}

{% block body %}

<table class="table">
  <thead>
    <tr>
      <th>Titulo</th>
      <th>Redactor</th>
      <th>Fecha de Publicacion</th>
      <th>Resumen</th>
      <th>cuerpo</th>
      <th>Cantidad de visitas</th>
      <th>Cantidad de comentarios</th>
    </tr>
  </thead>
  <tbody>
```

Figura 4. Patrón GoF Decorador en una vista (Elaboración propia)

2.5 Diseño de clases

2.5.1 Diagrama de Clases del Diseño

Los diagramas de clases del diseño representan las especificaciones de las clases e interfaces de software. Entre la información que representa se encuentran clases, interfaces con sus operaciones y constantes, métodos, navegabilidad y dependencias. Durante el diseño del sistema, el diagrama de clase del diseño se enfoca a los detalles de la implementación y refleja el funcionamiento de la aplicación en términos lógicos (Larman, 2016). En la figura 6 se muestra el diagrama de clases del diseño de la HU Gestionar sitio. En la CP_Listado de sitios el usuario va a visualizar un listado de sitios de los cuales se van a extraer noticias, en la misma hay un link que se dirige hacia la CP_Insertar sitio que contiene un formulario para insertar los datos de un sitio, cuando se realiza submit se dirige hacia

la SP_Registro que se relaciona con la clase controladora CC_Registro que contiene el método Insertarsitio(); la clase parameters.yml se encarga de conectarse con la base de datos para insertar los datos de un nuevo sitio. En la CP_Listado de sitios se encuentran la CP_Editar sitio y la CP_Mostrar sitio, en la que cada una de ella contiene un formulario y cuando realiza submit se ejecuta el método que le corresponde a cada una que contiene la clase controladora CC_Registro.

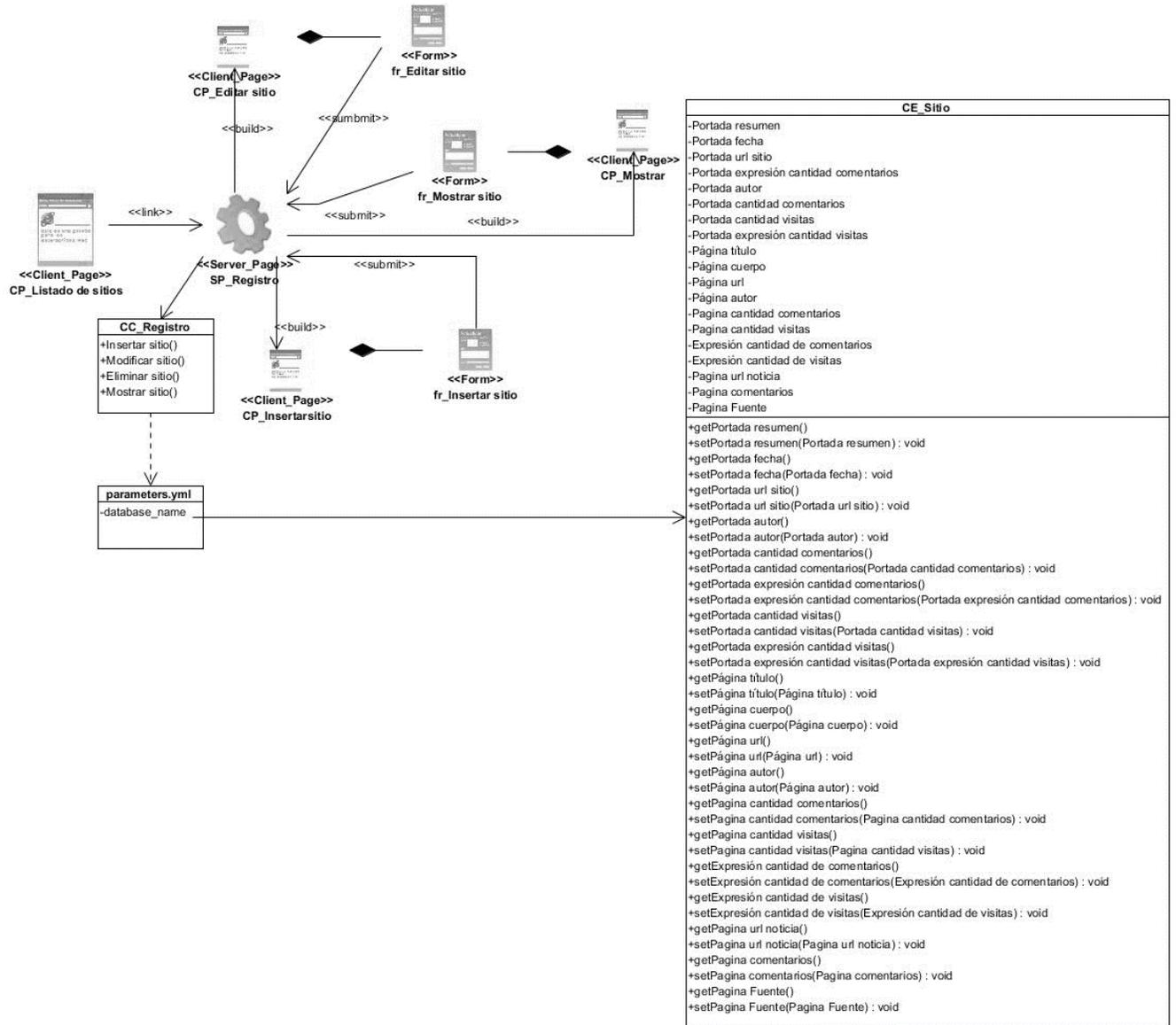


Figura 5. Diagrama de clases de diseño de la HU_3 Gestionar sitio (Elaboración propia)

2.5.2 Diseño de datos

Diagrama Entidad Relación

El modelo de datos es una representación abstracta de los datos de una organización y las relaciones entre ellos. El propósito de un modelo de datos es, por una parte, representar los datos y, por otra parte, ser comprensible. Elaborar un correcto modelo de datos es imprescindible para manejar y

obtener correctamente los datos (Morales, 2017). A continuación, se muestra el modelo de datos que representa físicamente las tablas de la base de datos que contiene el subsistema y las relaciones entre las tablas. Las tablas que se encuentran son Publicación donde se almacenan las noticias cuando se hace la recopilación, la tabla Comentarios que hace referencia a los comentarios de una publicación, la tabla Registro que hace referencia a un sitio, la tabla Categoría que se refiere a la clasificación que el usuario le va asignar a una publicación.

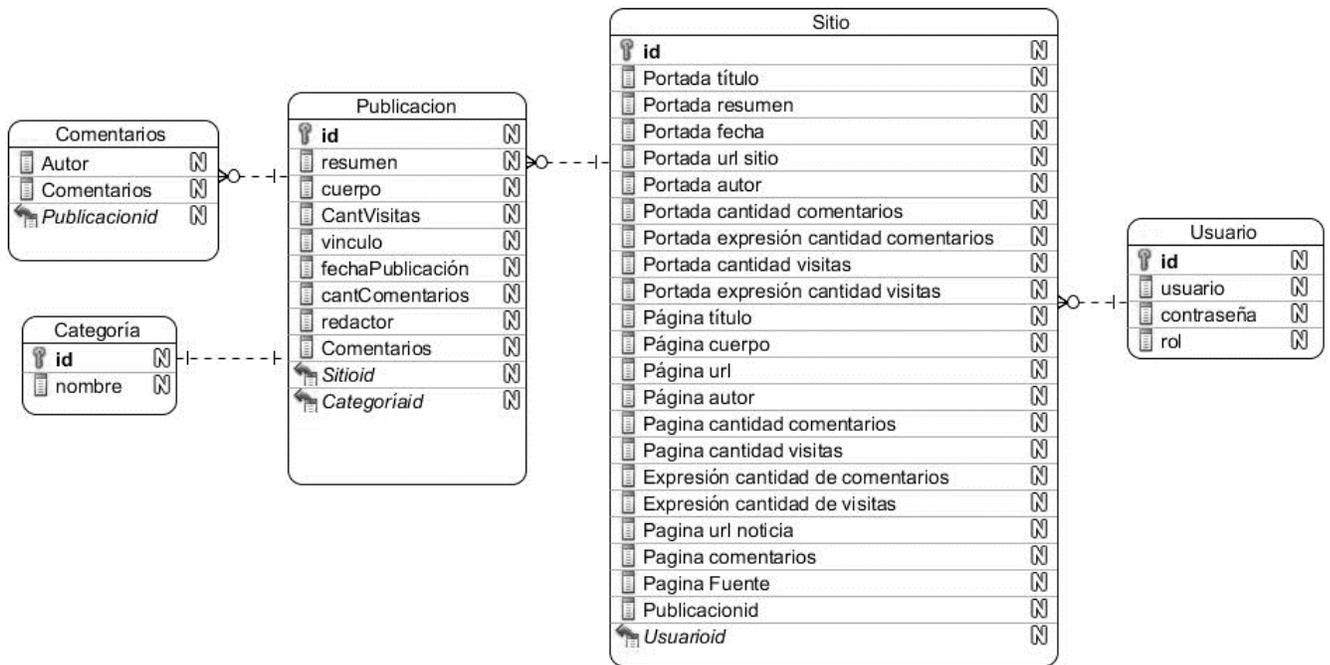


Figura 6. Diagrama Entidad-Relación (Elaboración propia)

2.6 Diagrama de secuencia

Los diagramas de secuencia modelan el flujo de la lógica dentro del sistema de forma visual, permitiendo documentarla y validarla. Pueden usarse tanto en análisis como en diseño, proporcionando una buena base para identificar el comportamiento del sistema. Típicamente se usan para modelar los escenarios de uso del sistema, describiendo de qué formas puede usarse (Aykar, 2015). La figura 8 representa el diagrama de secuencia de la HU Insertar sitio. En la CP_Listado de sitios el usuario va a seleccionar Insertar sitios, la página servidora Registro va a construir a la CP_Insertar sitio que contiene un formulario para que el usuario inserte los datos de un sitio, luego que se realiza submit se verifica si el sitio se encuentra en la base de datos si no esta se inserta y se muestra en la CP_Insertar sitio, “Registro satisfactorio”, si el sitio se encuentra en base de datos, será mostrado un mensaje que diga “El sitio ya existe”.

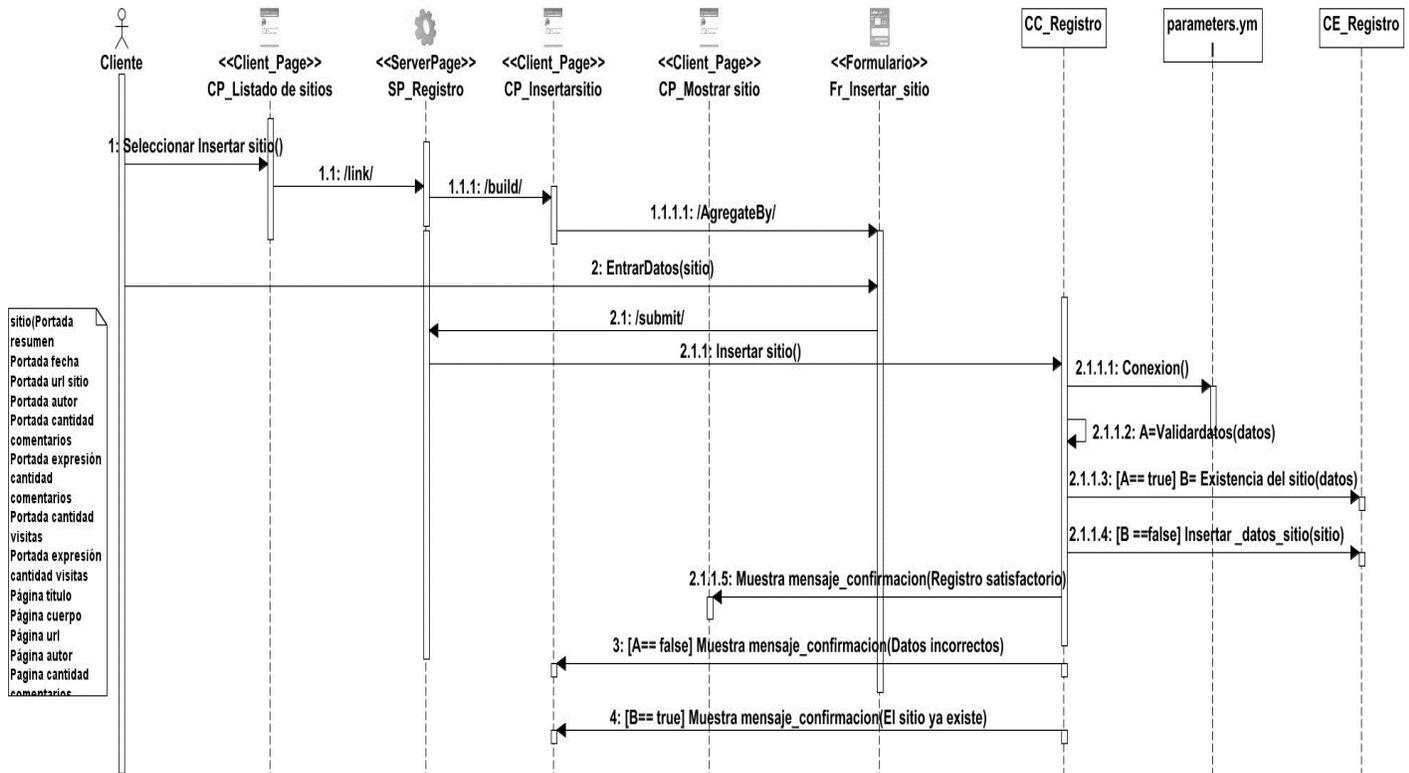


Figura 7. Diagrama de secuencia del RF_6 (Elaboración propia)

2.7 Diagrama de despliegue

El diagrama UML de despliegue representa una vista estática de la configuración en tiempo de ejecución de los nodos que intervienen en el proceso y de los componentes que se ejecutan en esos nodos. Puede mostrar el *hardware* del sistema y los paquetes *software* que hay instalados en él, así como el *middleware* que los conecta (Aykar, 2015).

A continuación, se muestra el diagrama de despliegue del subsistema de recopilación automática de noticias. El diagrama muestra la disposición física de los nodos que componen el sistema, los componentes que se encuentran en cada nodo, así como la relación de protocolo y puerto entre los nodos.



Figura 8. Diagrama de despliegue del subsistema de recopilación automática de noticias (Elaboración propia)

Conclusiones parciales

En este capítulo para llevar a cabo el desarrollo del subsistema de recopilación automática de noticias para la UCI, se tuvieron en cuenta varios aspectos como son la identificación de los requisitos funcionales y no funcionales que permitió definir las características y las condiciones que debe suministrar el subsistema a desarrollar; la definición de la arquitectura y los patrones de diseño a utilizar, permitieron establecer las bases para fomentar la reutilización y las buenas prácticas de programación durante la fase de implementación, así como disminuir el impacto de los cambios futuros en el código fuente; el diagrama de secuencia con el objetivo de conocer el flujo con que se desarrollan las actividades en la aplicación y la elaboración del diagrama de despliegue permitió identificar la disposición física de los artefactos de la propuesta de solución.

CAPÍTULO 3. Implementación y prueba del subsistema de recopilación automática de noticias de los medios informativos de la UCI.

Introducción

En el presente capítulo se describe el proceso de implementación y de validación desarrollado para evaluar la propuesta de solución. Se verifica la calidad del resultado de la implementación, mediante los productos de trabajos generados durante las disciplinas de pruebas, exponiendo los resultados obtenidos. Además, se aplica la técnica de Iadov para validar el objetivo de la investigación a través de la medición del índice de satisfacción de los futuros usuarios del subsistema de recopilación automática de noticias.

3.1. Diagrama de componentes

Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre los elementos de implementación. De esta forma, se modela la vista estática de un sistema, estableciendo la organización y las dependencias entre un conjunto de componentes de *software*, sean código fuente, paquetes, bibliotecas, binarios o ejecutables (Wesley, 2016). A continuación, se muestra el diagrama de componentes de la aplicación.

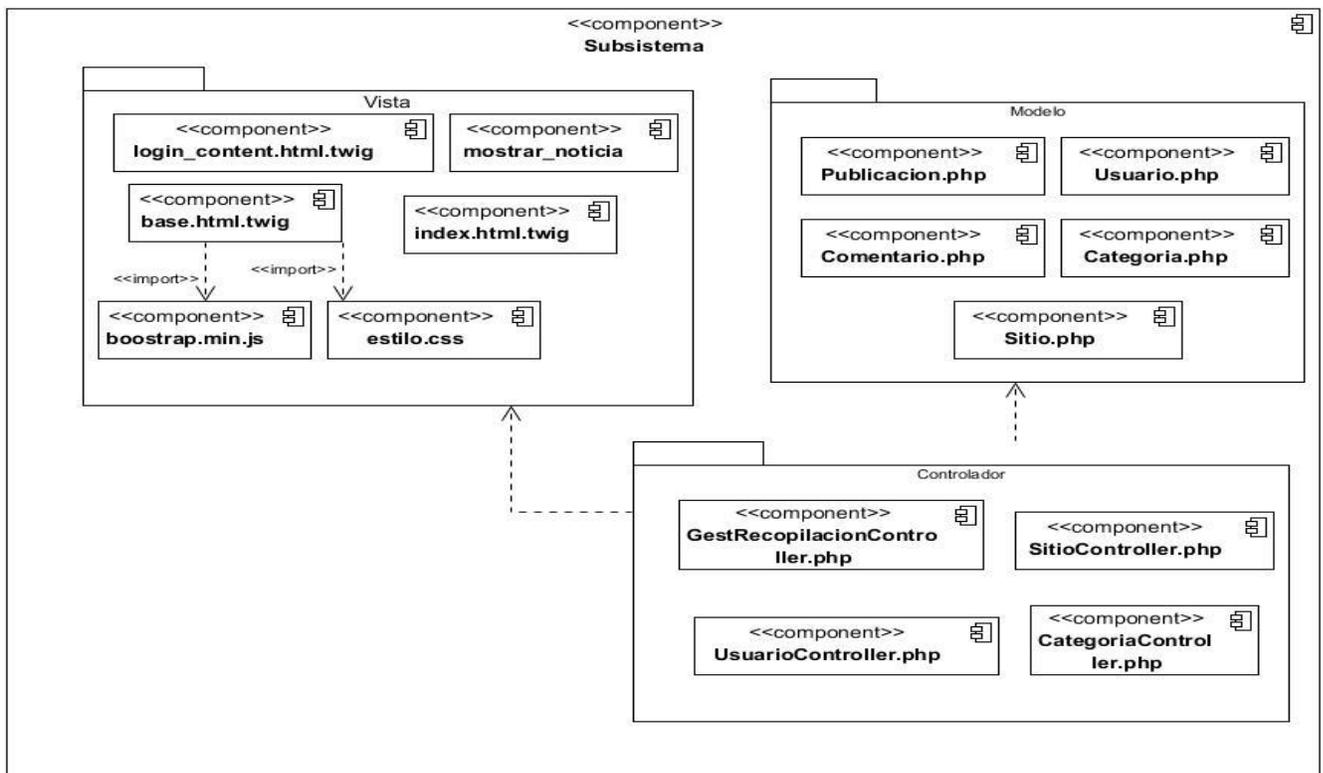


Figura 9. Diagrama de componente (Elaboración Propia).

Los componentes contenidos en el diagrama son:

-En el paquete Vista se encuentran los componentes `login_content.html.twig` que es la interfaz para que el usuario sea autenticado, `base.html.twig` contiene todo el diseño que va a contener la aplicación y por tanto el resto de las vistas van a heredar de ella para mantener el diseño, `index.html.twig` que contiene el listado de noticias que fueron insertadas en la base de datos y `mostrar_noticia.html.twig` contiene todos los datos de una noticia como título, cuerpo de la noticia, autor, vínculo, entre otros.

-En el paquete Modelo se encuentra el componente Sitio que contiene toda la información de un sitio, se encuentra el componente Publicación que contiene todos los datos de una noticia, en el componente Comentario se encuentra todos los comentarios relacionados a una publicación, el componente Categoría se refiere a la clasificación que el usuario le va asignar a una publicación que puede ser de cocina, humor u otra categoría y el componente Usuario representa a los usuarios autorizados para acceder a la recopilación del subsistema.

-En el paquete Controlador se encuentra el componente RegistroController que se encarga de realizar la recopilación automática y el componente GestRegistro que se encarga de listar las noticias y mostrárselas al usuario.

3.2. Estándares de codificación

Los estándares de codificación son reglamentos de la programación que están enfocados a la estructura y apariencia física del programa con el objetivo de facilitar la lectura, comprensión y mantenimiento del código. Un estándar de codificación comprende todos los aspectos de la generación de código. Los programadores deben implementar un estándar de forma prudente que tienda siempre a lo práctico. La legibilidad del código fuente influye directamente en el entendimiento que puedan tener los desarrolladores del equipo de trabajo, pues todo *software* tiene que someterse constantemente a mantenimiento y mejora de sus funcionalidades (Ramírez, 2015).

Los estándares en la codificación fueron los siguientes:

- Para la definición del nombre de las clases, funciones y variables se tiene en cuenta el estilo *Lower Camel Case*. El estilo establece que la separación entre palabras internas de los identificadores deberá realizarse escribiendo la letra inicial en mayúscula, a excepción de la primera palabra. Además, no deberá colocarse ningún carácter especial entre palabras de los identificadores (Takeyas, 2015). El estilo se evidencia en la clase Registro con todos sus atributos ejemplo de ellos `portAutor` y `porTitulo`.
- En la presente investigación las llaves de apertura se colocarán inmediatamente al final de la línea de cabecera del bloque, así como en las estructuras `if`, `for`, `else`, `switch`, `foreach`. Las llaves de cierre se colocarán solitarias en la línea siguiente a la última línea dentro del bloque.
- Los comentarios en el código representan la documentación interna más precisa de un software. Estos garantizan el entendimiento de lo que realmente realiza un determinado bloque de código, evitando

confusiones y agilizando considerablemente las tareas de revisión y mantenimiento. Para la inclusión de comentarios es necesario respetar algunas reglas básicas como: abreviar el contenido de los comentarios e incluirse un espacio simple entre los caracteres “/” y el texto del comentario.

- El nombre de las clases controladoras terminan con la palabra Controller como se evidencia en las clases RegistroController y GestRecopilacionController.

- Las variables que poseen nombres de palabras compuestas se escriben juntas por ejemplo la variable cantidadComentario que se encuentra en la clase Publicación.

3.3 Interfaces

Una vez desarrollado el subsistema de recopilación automática de noticias de los medios informativos de la UCI, serán mostradas las interfaces principales que la misma posee.

La siguiente imagen muestra una interfaz para ver las noticias recopiladas, primero se debe escoger el sitio y un rango de fecha para ver las noticias que pertenecen al sitio seleccionado y será mostrado un listado de noticias.

The screenshot shows a web interface for filtering news. It includes three input fields: 'Sitio' with a dropdown menu set to 'dragones.uci.cu', 'Fecha inicio' with a date picker set to '04 / 09 / 2019', and 'Fecha fin' with a date picker set to '04 / 21 / 2019'. Below these is a 'Mostrar' button. The main part of the interface is a table with the following columns: Título, Redactor, Fecha de Publicación, Resumen, Categoría, Cantidad de visitas, and Cantidad de comentarios.

Título	Redactor	Fecha de Publicación	Resumen	Categoría	Cantidad de visitas	Cantidad de comentarios
La torre Eiffel de Santa Clara	Osiel	2019-04-09 23:31:25	Imágenes con más de 120 años muestran una réplica de la famosa torre Eiffel , uno de los símbolos de París, capital de Francia, construida en medio de la Plaza Mayor, de esta ciudad, para la inauguración del alumbrado eléctrico, en febrero de 1895.	Docente	172	0
ETECSA y Google firman memorando de entendimiento para mejorar calidad de acceso a Internet	Osiel	2019-04-09 23:31:25	Directivos de la Empresa de Telecomunicaciones de Cuba S.A., ETECSA, y la compañía estadounidense Google, firmaron este jueves en La Habana un memorando de entendimiento para comenzar la negociación de un posterior acuerdo de servicio de intercambio de tráfico de Internet.	Curiosidades	118	0
Vale - No vale	Julio Luis	2019-04-09 23:31:25	Cubadebate retoma su sección Vale-No Vale para reconocer, una vez más gracias a los comentarios y correos de los lectores, todo aquello que vale porque mejora nuestras vidas, desarrolla el país, o nos hace mejores ciudadanos, y señalar también todo lo que No Vale y merece ser cambiado. VALE: Que el país destine recursos para […]	Docente	128	0
Convocatoria Evento	Liset	2019-04-09	En el marco de la Jornada del Ingeniero en Ciencias Informáticas (JICI), el departamento	Docente	102	0

Figura 10. Interfaz para ver las noticias recopiladas (Elaboración propia)

La aplicación contiene una interfaz para ver los usuarios que acceden a la aplicación. La misma tendrá dos roles para el usuario: el rol como usuario y el rol de administrador. El usuario con rol de usuario tendrá acceso solamente a la opción Mostrar noticia y si tiene permisos de administrador podrá acceder

al resto de las opciones que contiene el subsistema. Los usuarios se podrán añadir, modificar, eliminar y mostrar en la aplicación.

Mostrar noticias Listado de sitios Listado de usuarios Categorías Salir

Listado de usuarios

Insertar un usuario

Id	Usuario	Email	Acciones
3	user	user@gmail.cu	Mostrar Editar
19	lmaleta	lmaleta@estudiantes.uci.cu	Mostrar Editar
12	liana	liana@gmail.com	Mostrar Editar

Figura 11. Interfaz con el listado de usuarios (Elaboración propia)

La siguiente interfaz muestra la lista de sitios para realizar la recopilación automática de noticias. En caso que se quiera añadir un nuevo sitio se da clic en la opción Insertar un sitio.

Listado de sitios

Insertar un sitio

Mostrar: 10 Entradas Buscar:

Id	Portada url sitio	Habilitado	Portada artículo	Portada título	Portada resumen	Portada autor	Portada cantidad comentarios	Portada expresión cantidad comententarios	Portada cantidad de visitas	Portada expresión cantidad visitas	Página título	Página cuerpo
1	https://dragones.uci.cu	1	.post-listing > .loop-post.standard	.col-md-offset-0 h2 a	.col-sm-8	.author-info a	.col-sm-7 ul.toolbar-links li a	/[[0-9+]]/	.col-sm-7 ul.toolbar-links li span.hit-count	/[[0-9+]]/	h1	.post-content
2	https://facultad2.uci.cu/category/noticias/	1	#left-area article	h2.entry-title a		p.post-meta span					.entry-title a	.entry-content
3	https://zorros.uci.cu/enterate	1	.view-portada-enterate .view-content > .views-row	h2.field-content a	.views-field-body						.page-header	.field-name-body .field-item.ever

Copyright © Company SRAN 2019. Liana

Figura 12. Interfaz con el listado de sitios (Elaboración propia)

3.4 Validación del subsistema de recopilación automática de noticias

Las pruebas de *software* son un elemento crítico para garantizar la calidad de la aplicación desarrollada. Representan una revisión final de las especificaciones, del diseño y de la codificación de dicha aplicación. Tienen como objetivo general validar un programa, independientemente de las características y el entorno donde se desarrolla, además permite documentar los errores en el sistema y posibilita probar la correcta implementación de los requisitos que el *software* debe cumplir.

3.4.1 Pruebas funcionales

Las pruebas funcionales, también denominadas pruebas de comportamiento se centran en los requisitos funcionales del *software*. Estas permiten al desarrollador obtener un conjunto de datos de entrada que evalúan todos los requisitos funcionales del sistema. Por esto se denominan pruebas funcionales, donde se suministran datos de entrada y se observa la salida, sin necesidad de conocer el funcionamiento interno del *software*.

Dentro de esta clasificación se encuentra la técnica partición equivalente el cual es una técnica de prueba de caja negra que divide el campo de entrada de un programa en clases de datos de los que se pueden derivar casos de prueba. La partición equivalente busca obtener casos de prueba ideales que descubran de forma inmediata clases de errores, reduciendo el número total de casos de prueba a desarrollar. Una clase de equivalencia representa un conjunto de estados válidos o no válidos para condiciones de entrada. Una condición de entrada es un valor numérico específico, un rango de valores, un conjunto de valores relacionados o una condición lógica (Pressman, 2010).

A continuación, se muestra uno de los casos de pruebas para el requisito funcional Insertar sitio cuya prioridad es alta para el usuario, descrito en el capítulo anterior. La siguiente tabla muestra la descripción de las variables del caso de prueba.

Tabla 6. Descripción de las variables del caso de prueba de la HU_3 Gestionar sitio.

Variable	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Portada url del sitio	Campo de texto	No	No permite caracteres extraños
2	Habilitado	Campo seleccionable	No	Permite marcar para ver si el sitio está habilitado.
3	Portada artículo	Campo seleccionable	No	Permite marcar para ver si el sitio está habilitado.

4	Portada título	Campo de texto	No	No permite caracteres extraños.
5	Portada resumen	Campo de texto	No	No permite caracteres extraños.
6	Portada autor	Campo de texto	No	No permite caracteres extraños.
7	Portada cantidad de comentarios	Campo de texto	No	No permite caracteres extraños.
8	Portada expresión cantidad de comentarios	Campo de texto	No	No permite caracteres extraños.
9	Portada cantidad de visitas	Campo de texto	No	No permite caracteres extraños.
10	Portada expresión cantidad de visitas	Campo de texto	No	No permite caracteres extraños.
11	Página url	Campo de texto	No	No permite caracteres extraños.
12	Página autor	Campo de texto	No	No permite caracteres extraños.
13	Página título	Campo de texto	No	No permite caracteres extraños.
14	Página cuerpo	Campo de texto	No	No permite caracteres extraños.
15	Borrar detalles	Campo de texto	No	No permite caracteres extraños.
16	Página cantidad de comentarios	Campo de texto	No	No permite caracteres extraños.
17	Expresión cantidad de comentarios	Campo de texto	No	No permite caracteres extraños.

18	Página cantidad de visitas	Campo de texto	No	No permite caracteres extraños.
19	Expresión cantidad de visitas	Campo de texto	No	No permite caracteres extraños.
20	Página url noticia	Campo de texto	No	No permite caracteres extraños.
21	Página comentarios	Campo de texto	No	No permite caracteres extraños.
22	Página Fuente	Campo de texto	No	No permite caracteres extraños.

Tabla 7. Caso de prueba de HU_3 Gestionar sitio (Elaboración propia)

Entrada	Resultados	Condiciones
El usuario introduce los campos de un sitio correctamente.	Se muestra un mensaje que el sitio fue insertado correctamente.	Establecer conexión con la red para acceder al HTML de los sitios y extraer las noticias de cada uno de ellos.
El usuario repite la url de un sitio en el campo Portada url.	El sistema le muestra un mensaje informando al usuario que el sitio ya existe.	Establecer conexión con la red para acceder al HTML de los sitios y extraer las noticias de cada uno de ellos.
El usuario deja los campos vacíos Portada url y Portada artículo.	El sistema le muestra un mensaje informando al usuario autenticado que rellene los campos vacíos.	Establecer conexión con la red para acceder al HTML de los sitios y extraer las noticias de cada uno de ellos.

Resultados de las pruebas funcionales

Con el objetivo de verificar el correcto funcionamiento de las funcionalidades se realizaron 3 iteraciones para el sistema. En la siguiente tabla se muestran los resultados obtenidos por cada iteración de prueba al subsistema de recopilación automático de noticias, las cuales se detectaron en la primera iteración 10 no conformidades: 4 cuando se insertaron los sitios, 4 de validación y 2 errores en las interfaces con

respecto al diseño que deben cumplir. En la segunda iteración se detectaron 6 no conformidades, 4 de la iteración anterior, 2 nuevas no conformidades en cuanto a faltas de ortografía, siendo todas resueltas en su totalidad. En la tercera iteración no se encontró ninguna no conformidad por lo que la aplicación se encuentra funcionando correctamente.

Tabla 8. Cantidad de no conformidades por cada iteración de las pruebas funcionales (Elaboración propia)

No Conformidades	1ra Iteración	2da Iteración	3ra Iteración
Detectadas	10	6	0
Resueltas	6	6	0
Pendientes	4	0	0

En la siguiente figura se puede apreciar el comportamiento de las no conformidades de las pruebas funcionales ejecutadas

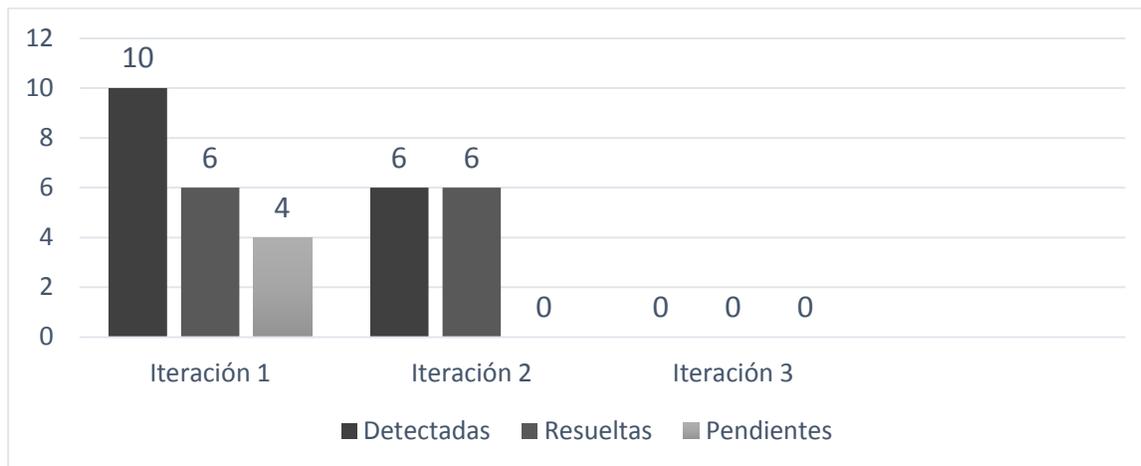


Figura 13. Comportamiento de las no conformidades para cada iteración de las pruebas funcionales (Elaboración propia)

3.4.2 Pruebas de rendimiento

Las pruebas de rendimiento se usan para descubrir problemas de rendimiento que pueden ser resultado de: falta de recursos en el lado servidor, red con ancho de banda inadecuada, capacidades de base de datos inadecuadas, capacidades de sistema operativo deficientes o débiles, funcionalidad de *webapp* pobremente diseñada y otros conflictos de *hardware* o *software* que pueden conducir a rendimiento cliente-servidor degradado (Pressman, 2010). Las pruebas de rendimiento se dividen en dos tipos de pruebas de carga y estrés.

La prueba de **carga** consiste en evaluar cómo actúa el sistema con una carga variable de usuarios, pero dentro de los niveles esperados de la aplicación. Esta prueba da una idea al propietario de la aplicación como actuara su sistema bajo una carga “normal” cuando este se encuentre en producción (Falcón, 2016).

La prueba de **estrés** evalúa el sistema sometiéndolo a una carga creciente hasta que este falle. Esta prueba permitirá identificar cuellos de botella “bottleneck” y conocer que carga es la máxima admitida por la aplicación hasta que esta empieza a tener problemas de rendimiento (Falcón, 2016).

Para la realización de las pruebas de carga y estrés se utilizó la herramienta Apache Jmeter. A continuación, se describen las variables que miden el resultado para estas pruebas:

Usuario: cantidad de usuarios conectados en el subsistema.

Muestra: Cantidad de peticiones realizadas para cada URL.

Media: Tiempo promedio en milisegundos en el que se obtienen los resultados.

Mediana: Tiempo en milisegundos en el que se obtuvo el resultado que ocupa la posición central.

Min: Tiempo mínimo que demora un hilo en acceder a una página.

Max: Tiempo máximo que demora un hilo en acceder a una página.

Línea 90 %: Máximo tiempo utilizado por el 90 % de la muestra, al resto de la misma le llevo más tiempo.

% Error: Por ciento de error de las páginas que no se llegaron a cargar de manera satisfactoria.

Tabla 9. Resultado a partir de la prueba de carga y estrés (Elaboración Propia)

Usuario	Muestra	Media	Mediana	Min	Max	Línea 90 %	% Error
10	1020	1358	1368	1	1278	1745	0
30	3298	4578	2014	4	1489	3127	0
50	5700	9567	2267	3	2223	5146	0

Las pruebas realizadas muestran que el subsistema de recopilación automática es capaz de responder:

- Para un total de 1020 peticiones iniciadas por 10 usuarios conectados, el subsistema respondió en un tiempo máximo de 1278 milisegundos (1.2 segundos aproximadamente) con 0 % de error.
- Para un total de 3298 peticiones iniciadas por 30 usuarios conectados, el subsistema respondió en un tiempo máximo de 1489 milisegundos (1.4 segundos aproximadamente).
- Para un total de 5700 peticiones iniciadas por 50 usuarios conectados, el subsistema respondió en un tiempo máximo de 2223 milisegundos (2.2 segundos aproximadamente).

Luego de haber realizado las pruebas, se obtuvo un tiempo máximo de 4.8 segundos para 10018 peticiones, con un 0 por ciento (%) de errores lo que resulta que el tiempo de respuesta es favorable y

los usuarios puedan acceder de manera concurrente a la aplicación dándole respuesta a los requisitos no funcionales 8 y 9.

3.4.3 Pruebas de seguridad

La seguridad informática comprende la puesta en práctica de un conjunto de medidas preventivas y reactivas en los sistemas informáticos y tecnológicos, que posibilitan la protección de la información, persiguiendo como objetivo principal la integridad, confidencialidad y disponibilidad de la misma. El objetivo de esta prueba es evaluar el funcionamiento de los controles de seguridad del sistema para asegurar la integridad y confidencialidad de los datos, así como probar las vulnerabilidades del subsistema frente a accesos o manipulaciones no autorizadas.

Para la ejecución de esta prueba se utilizó la herramienta Acunetix la cual se encarga de señalar las vulnerabilidades del subsistema de recopilación automático de noticias. Como resultado final de las pruebas de seguridad se obtuvo en una primera iteración 10 no conformidades, 7 de nivel medio y 3 de nivel bajo.

Las no conformidades de nivel medio se basan en inyecciones SQL, estas se basan en una consulta SQL que es una petición sobre una base de datos, los hackers utilizan los cuadros de introducción de datos para enviar sus propias peticiones con el objetivo de obtener información confidencial, este problema fue resuelto adoptando técnicas de validación de entrada adecuada, verificando los datos que el usuario introduce y asignando los usuarios que se conectan con la base de datos mínimos privilegios sobre las mismas. Otra no conformidad que se identificaron fueron vulnerabilidades Cross Site Request Forgery (CSRF, por sus siglas en inglés) que son ataques de usuarios malintencionados con el objetivo de que los usuarios de la web les envíen datos sin saberlo, este problema fue resuelto configurando el navegador para que no recuerde usuarios y contraseñas, así como recordar cerrar la sesión cuando ya no se vaya a hacer uso de la aplicación.

Las no conformidades de nivel medio estuvieron enfocadas a las vulnerabilidades Cross Site Scripting (XSS, por sus siglas en inglés) donde el hacker ataca a otros usuarios que se conectan a dicha aplicación web, y los ataca en busca de varias cosas: secuestrarles la sesión, robarles la contraseña o cualquier otra información que considere de interés, para evitar estas vulnerabilidades la implantación y la configuración de un cortafuego de aplicación web.

En una segunda iteración no se encontraron nuevas no conformidades, obteniéndose un subsistema seguro y en condiciones de ser utilizado por el usuario. Los resultados antes descritos, se muestran en la siguiente gráfica:



Figura 14. Resultados de las pruebas de seguridad (Elaboración propia)

3.5 Validación científica

Para la validación científica se utiliza la técnica de ladov. “Cuando se realiza una propuesta, es recomendable retroalimentarse con la opinión de los usuarios potenciales. Esta información es útil para conocer las debilidades de la propuesta y profundizar en sus fortalezas. En ese sentido, la técnica de ladov es un instrumento que ayuda a conocer el grado de satisfacción de los potenciales usuarios” (Zamora, 2016).

Para aplicar esta técnica se realizó una encuesta a 10 especialistas del centro CIDI. La encuesta está estructurada en tres preguntas cerradas y dos abiertas las cuales se pueden ver en el **Anexo 7**. La siguiente tabla muestra la relación entre las preguntas realizadas:

Tabla 10. Cuadro lógico de ladov (Elaboración propia)

3-Luego de haber visto las noticias de los sitios recopiladas por el subsistema de recopilación automático de noticias a través de la interfaz web desarrollada, indique en qué medida le gusta la propuesta de solución.	1- ¿Considera que es necesario la recopilación automática de noticias?								
	Si			No sé			No		
	2-¿Estima provechosa la herramienta desarrollada para el análisis de los principales temas tratados acerca de las publicaciones en la web.								
	Si	No sé	No	Si	No sé	No	Si	No sé	No
Me gusta mucho	1	2	6	2	2	6	6	6	6
Me gusta más de lo que me disgusta	2	2	3	2	3	3	6	3	6
Me da lo mismo	3	3	3	3	3	3	3	3	3

Me disgusta más de lo que me gusta	6	3	6	3	4	4	3	3	4
No me gusta nada	6	6	6	6	4	4	6	6	5
No sé decir	2	3	6	3	3	3	6	6	4

Esta técnica también permite obtener el índice de satisfacción grupal (ISG), para lo cual se trabaja con los diferentes niveles de satisfacción que se expresan en la escala numérica que oscila entre +1 y - 1 de la siguiente forma: Máxima satisfacción (+ 1), Satisfecho (0.5), No definido (0), Insatisfecho (- 0.5) y Máxima insatisfacción (- 1).

A partir de la cantidad de respuestas es posible calcular el Índice de Satisfacción Grupal (ISG) siguiendo la siguiente fórmula:

$$ISG = \frac{A (+ 1) + B (0.5) + C (0) + D (- 0.5) + E (-1)}{N}$$

En esta fórmula A, B, C, D y E representan la cantidad de sujetos que interactúan con la aplicación y N representa el total de sujetos.

La escala de valores del índice grupal que se toma al aplicar la técnica se encuentran entre:

- 1 y -0.5 Insatisfacción.
- -0.49 y 0.49 Contradicción.
- 0.5 y 1 Satisfacción.

Los resultados obtenidos fueron los siguientes:

Tabla 11. Resultados de las escalas de satisfacción (Elaboración propia)

Total de usuarios(N)	10	Escala
Máxima satisfacción	8	A
Satisfecho	1	B
No definido	1	C
Insatisfecho	0	D
Máxima insatisfacción	0	E

Cálculo del ISG a partir de la fórmula anterior:

$$\text{ISG} = \frac{8(+1) + 1(0.5) + 1(0) + 0(-0.5) + 0(-1)}{10} = 0.85$$

A partir de análisis realizado se puede concluir que el uso de la Técnica de ladov confirmó su factibilidad de uso, expresado cuantitativamente en el alto Índice de Satisfacción Grupal (ISG= 0.85) lo que refleja aceptación de la propuesta por parte de los encuestados. Las respuestas con respecto a las preguntas abiertas brindadas por los encuestados reafirman los beneficios que traerá la utilización del sistema propuesto. Por lo tanto, se puede afirmar que se cumplió el objetivo de la investigación.

Conclusiones parciales

Con las pruebas realizadas a la aplicación se logró detectar y documentar a tiempo los errores que existían para corregirlos. Fue validado que la implementación satisface todos los requisitos identificados para desarrollar la aplicación. Al finalizar con la fase de prueba se obtuvo como resultado una aplicación que funciona correctamente, cumpliéndose la totalidad de los objetivos planteados para el subsistema de recopilación automática de noticias de los medios informativos de la UCI y se evidenció la alta satisfacción del cliente a través de los resultados arrojados al aplicar la técnica de ladov.

CONCLUSIONES GENERALES

Una vez realizada la presente investigación, se concluye que:

- A partir del estudio realizado de los conceptos asociados a los referentes teóricos de la investigación se logró una mayor comprensión de la propuesta de solución y se determinó la necesidad de crear un subsistema de recopilación automática de noticias.
- La metodología AUP en su variación UCI, el uso de tecnologías y herramientas seleccionadas, permitieron analizar y describir las funcionalidades que se debían de ejecutar, concretando así, en concordancia con las especificaciones del cliente y las características que debería de tener el subsistema a desarrollar.
- Con el desarrollo de la presente investigación se obtuvo un subsistema de recopilación automática de noticias de las publicaciones que se realizan en los sitios web de la UCI.
- Las pruebas de *software* realizadas, permitieron detectar no conformidades, las cuales fueron resueltas, garantizando la calidad del producto al comprobar su correcto funcionamiento.
- La validación científica, a través de la técnica de ladov evidenció que la propuesta de solución satisface las necesidades del cliente.

RECOMENDACIONES

Una vez concluida la investigación y el desarrollo de la propuesta de solución, en el presente trabajo se recomienda al departamento DOWAI implementar nuevas funcionalidades que permitan al subsistema generar reportes estadísticos a partir de las noticias recopiladas y llevar a cabo el desarrollo de un algoritmo que permita la clasificación automática de las noticias.

BIBLIOGRAFÍA

- Afday, Kenny. 2017.** Papeles de inteligencia. [En línea] 2017.
<https://papelesdeinteligencia.com/herramientas-de-web-scraping> [citado 5-11-2018][En línea].
- Alvarado, Tovar Noe. 2017.** Recopilación de datos de forma automática . 2017.
- Apache, Servidor Web. 2016.** [En línea] <http://recursostic.educacion.es>[citado 28 de noviembre]. 2016.
- Arévalo, Julio Alonso. 2010.** Gestión de la información, gestión de contenidos y conocimientos. 2010.
- Armas, Esteban Alejandro. 2017.** Herramientas de Análisis Dinámico de Vulnerabilidades en Aplicaciones Web. 2017.
- Aykar, David Perez. 2015.** Ingeniería en entornos de software libre. 2015.
- . 2015. Ingeniería en entornos de software libre. [aut. libro] David Perez. 2015.
- Borges, Juan Manuel. 2017.** Patrones de diseño. Ampliación de Programación Orientada a Objetos. 2017.
- Cabero, José. 2013.** Las aportaciones de las nuevas tecnologías a las instituciones. 2013.
- Cedeño, Luis. 2017.** ApacheLucene. [En línea] 2017. [Citado el: 15 de 10 de 2018.]
<http://lucene.apache.org/solr/index.html>.
- Chiavenato, Idalberto. 2010.** Introducción a la Teoría General de la Administración. Chiavenato, Idalberto : Séptima Edición, 2010.
- . 2010. Introducción a la Teoría General de la Administración. s.l. : Séptima Edición, 2010.
- Claudio, Balerio. 2016.** Programación en php básico . s.l. : ENI Ediciones, 2016.
- Cobo, Angel. 2017.** dspace.esPOCH.edu.ec. [En línea] 2017.
<http://dspace.esPOCH.edu.ec/bitstream/EspecificaciónRequerimientosSoftware.pdf>.
- Cordero, J. 2017.** La guía definitiva de diseño web: HTML, XHTML, CSS y herramientas de diseño. Santander y alrededores, España: Universidad de Europea del Atlántico. 2017.
- Eguiluz, Javier. 2015.** Desarrollo web ágil con Symfony. [En línea] 2014. <http://www.colegiobrains.com/wp-content/uploads/2015/11/desarrollo-agil-symfony-3.pdf>. 2015.
- Falcón, Diego Alberto. 2016.** Comparativa de herramientas para aplicar pruebas de carga y estrés para aplicaciones de desarrollo web. 2016.
- Flores, Eder. 2017.** Metodología de Desarrollo de Software. 2017.
- Gil, Manuel Torres. 2016.** indalog.ual.es. [En línea] 2016. [Citado el: 22 de 3 de 2019.]
<http://indalog.ual.es/mtorres/LP/index.php?opcion=inicio>. [En línea].
- Grisham, R. 2010.** Information Extraction: Techniques and Challenges. 2010.

- Hernández, Alexis Tadeo. 2015.** Metodologías para análisis político utilizando Web Scraping.pdf. 2015.
- Hernández, Luis. 2011.** Fundamentos de la Programación. 2011.
- Huergo, Jorge. 2015.** Los procesos de gestión. 2015.
- Jacobson, Ivar. 2016.** El proceso unificado de desarrollo de software. 2016.
- Jacobson, Ivar, Boocch, Grady and Rumbaugh, James. 2018.** [aut. libro] Ivar Jacobson. El proceso unificado de desarrollo de software. Madrid, España : Pearsons Education S.A, 2018.
- Jmeter, Apache. 2017.** [En línea] 2017. [Citado el: 3 de 25 de 2019.] <http://jmeter.apache.org/i..> [En línea].
- Lapuente, María Jesús Lamarca. 2013.** Hipertexto: El nuevo concepto de documento en la cultura de la imagen. Madrid : Tesis doctoral. Universidad Complutense de Madrid. 2013.
- Larman, Craig. 2003.** UML y Patrones. s.l. : 2da Edición, 2003.
- . **2015.** UML y Patrones, 2da Edición. 2015.
- . **2016.** UML y Patrones, 2da Edición. 2016.
- Ledo, Vidal. 2017.** Información y conocimiento para la dirección. 2017.
- Longatt, Francisco M. 2012.** Introducción a los sistemas de información:Fundamentos. 2012.
- . **2012.** Introducción a los sistemas de información:Fundamentos. 2012.
- Martínez, J M, Silva Delgado, Camilo Andrés. 2010.** Levantamiento y análisis de Requerimientos de Software con base en procesos de negocio. 2010.
- Martinez, JM,Silva Delgado, Camilo Andrés. 2014.** Levantamiento y análisis de Requerimientos de Software con base de proceso de negocios. 2014.
- Martínez, Rafael. 2015.** Portal en español sobre PostgreSQL. [En línea] 1 de Septiembre de 2015 <http://postgresql.org.es>[consultado 28 de octubre de 2018]. 2015.
- Merino, Marcos. 2014.** www.ticbeat.com. [En línea] 2014. <http://www.ticbeat.com/tecnologias/que-es-una-api-para-que-sirve>[citado 24-10-2018].
- MES, Ministerio de Educación Superior. 2014.** Modelo_del_Profesional. 2014.
- Mesa, Dr. Rafael Yanes. 2017.** La noticia y la entrevista.Una aproximación a su concepto y estructura. 2017.
- Michael, P. 2015.** Strategy and the Internet. Harward Business Review. 2015.
- Micó, Javier Ibáñez. 2014.** Extracción de datos de web usando técnicas de screen-scraping,Proyecto Final de Carrera. 2014.

- Montejo, Yarelys Chávez. 2015.** Gestión documental, Gestión de información y Gestión del conocimiento: nociones e interrelaciones. La Habana : s.n., 2015.
- Morales, Dra. María G. 2017.** Modelos de base de datos. Base de datos Relacionales. 2017.
- Ordóñez, Jonathan. 2018.** <https://www.idento.es/blog/desarrollo-web/que-es-una-api-rest> [citado 24-10-2018]. 2018.
- Pablo, . 2018.** kinsta.com. [Online] 2018. kinsta.com/es/base-de-conocimiento/que-es-nginx.
- Posada, Mario. 2010.** Phpstorm. [En línea] 28 de octubre de 2010. <http://www.phpstorm-themes.com/>.
- Potencier, Fabien. 2017.** Buenas Prácticas para aplicaciones SYMFONY. 2017.
- Pressman, Roger. 2010.** Ingeniería de Software. Un enfoque práctico. 2010.
- . 2010. Software Engineering. A Practtiones's Approach. Seventh. New York : McGraw-Hill, 2010. 978-0-07-337397-7.
- Quezada, Pablo Alejandro. 2017.** Uso de APIS para la extracción de información en canales sociales en contexto educativos. 2017.
- Ramírez, Adrián. 2015.** Reglas de calidad para la codificación estandarizada. 2015.
- Rodríguez, Enrique. 2017.** ApacheLucene. [En línea] 2017. [Citado el: 15 de 10 de 2018.] <http://lucene.apache.org/nutch/index.html>.
- Sánchez, Fernando. 2018.** Marco de desarrollo de la junta de Andalucía. 2018.
- Silega, Martínez Nemury. 2014.** Método para la transformación automatizada de modelos de procesos de negocio a modelos de componentes para sistemas de gestión empresarial. Universidad de las Ciencias Informáticas. 2014.
- Sommerville, . 2011.** Ingeniería de Software . s.l. : 9na edición, 2011.
- Takeyas, B Lopéz. 2015.** Nomenclatura sugerida para identificar los componentes de un proyecto . 2015.
- Thornton, J. 2013.** Bootstrap3 Manual Oficial, 2013 [Citado el: 28 de noviembre de 2018.] Disponible en: <http://conocimientoabierto.es>. 2013.
- UCI. 2014.** Metodología de desarrollo para la actividad productiva de la UCI. La Habana : UCI, 2014.
- UML. 2010.** <http://www.uml.org/>. [En línea]. [citado el 28 de 10 de 2018] : s.n., 2010.
- UML, Visual Paradigm for. 2014.** Visual Paradigm for UML - Software design tools for agile software development [En línea]. s.l. : <http://www.visualparadigm.com/product/vpuml/> [citado 28 de 10 de 2018], 2014.
- Wesley, Adison. 2016.** El lenguaje Unificado de Modelado. Manual de referencia. 2016.

Zamora, Raquel. 2016. PROGRAMA DE FORMACIÓN DE HABILIDADES PARA LA GESTIÓN DE CONTENIDO EN LOS PROFESORES DE LA UNIVERSIDAD DE CIENFUEGOS. 2016.

ANEXOS

Anexo 1. Entrevista a especialistas del centro DOWAI para identificar las necesidades del cliente

- 1- ¿Cómo se realiza el registro de las noticias en la herramienta SIMAE?
- 2- ¿Cómo le gustaría que fuera la recopilación de noticias?
- 3- ¿Qué datos de las noticias le gustaría que se mostrara en la propuesta de solución?

Anexo 2. Entrevista a especialistas del centro SINI para conocer cómo funciona Red c.u.b.a

- 1- ¿Cuál es el mecanismo que utiliza Red c.u.b.a para capturar la información diseminada en la red?
- 2- ¿Red Cuba almacena todo el contenido de las páginas encontradas en la red o solo elementos identificativos?
- 4- ¿Existe un mecanismo implementado en Red c.u.b.a que permita que una aplicación externa pueda consultar la información almacenada en red cuba?
- 5- ¿Qué elementos de una página web se pueden obtener mediante Red cuba?
- 6- ¿Utiliza c.u.b.a la técnica de web scraping?

Anexo 3. Historias de Usuario

Tabla 12. HU_1 Autenticar usuario (Elaboración propia)

Historia de usuario	
Número: HU_1	Nombre: Autenticar usuario.
Programador: Liana Maletá Terry	Iteración asignada: 1
Prioridad en negocio: Media	
Descripción: Antes de acceder a la aplicación es necesario que el usuario se encuentre autenticado, por lo que será mostrado una interfaz web para colocar el usuario y la contraseña. Según el rol que tenga el usuario autenticado el mismo tendrá acceso a varias funcionalidades de la aplicación.	
Prototipo de interfaz:	



Tabla 13. HU_2 Gestionar usuario (Elaboración propia)

Historia de usuario	
Número: HU_2	Nombre: Gestionar usuario.
Programador: Liana Maletá Terry	Iteración asignada: 1
Prioridad en negocio: Media	
<p>Descripción: La aplicación contiene las opciones de insertar, mostrar, eliminar y modificar para un usuario, estas funcionalidades se refieren a los requisitos funcionales RF2, RF3, RF4 y RF5.</p> <p>En la aplicación se debe escoger la opción Listar usuarios, luego de haber escogido esta opción se mostrará un listado de usuarios que pueden acceder a la aplicación. Si se quiere insertar un usuario se escoge la opción Insertar usuario y será mostrado un formulario para insertar los datos del nuevo usuario.</p> <p>Para modificar un usuario, se selecciona el usuario a modificar y se hace clic en el botón Editar y se mostrará un formulario para editar el campo que desee y luego se debe presionar el botón Guardar.</p>	

Para eliminar un usuario se escoge la opción mostrar, luego aparece una interfaz con los datos del usuario, se hace hacer clic en el botón Eliminar y le aparecerá una nueva interfaz para confirmar la eliminación del usuario y presiona el botón Eliminar.

Prototipo de interfaz:

The image shows two side-by-side window prototypes for user management. The left window is titled "Insertar usuario" and contains four input fields labeled "Usuario", "Rol", "Email", and "Contraseña". Below the fields are two buttons: "Insertar" and "Volver a la lista". The right window is titled "Editar usuarios" and contains five input fields with pre-filled values: "Usuario" (Pepe), "Rol" (Administrador), "Contraseña" (123), and "Email" (Pepe@uci.cu). Below the fields are two buttons: "Aceptar" and "Volver a la lista".

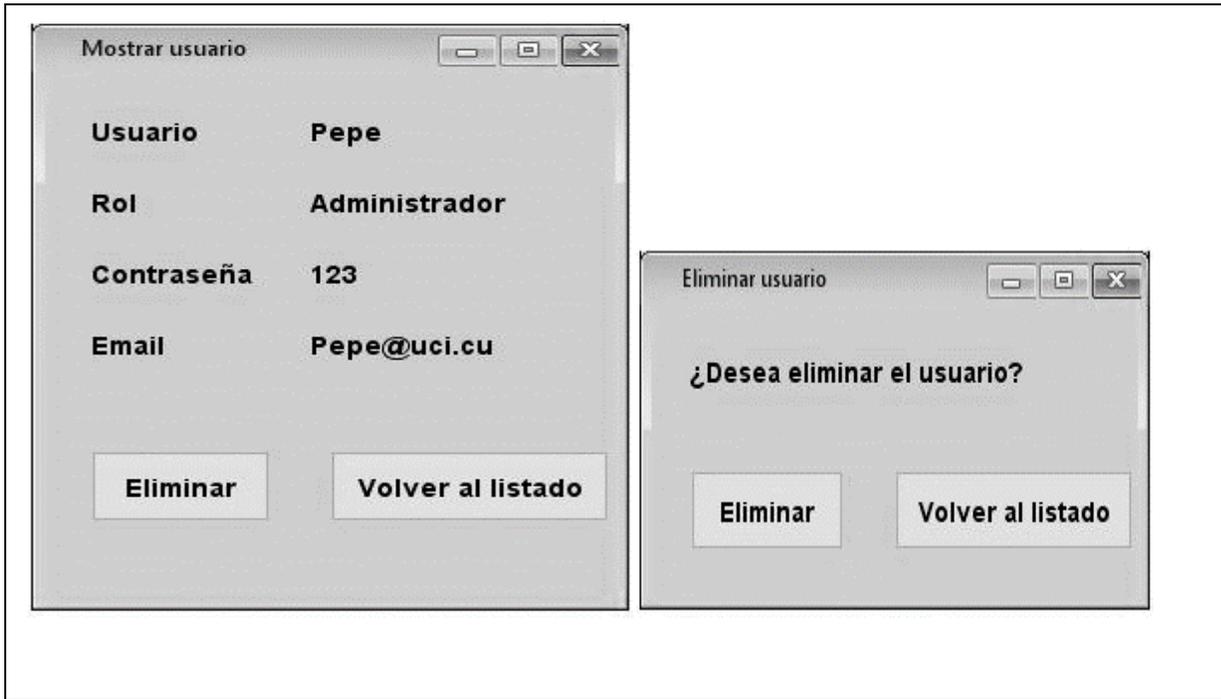


Tabla 14. HU_4 Gestionar categoría (Elaboración propia)

Historia de usuario	
Número: HU_4	Nombre: Gestionar categoría
Programador: Liana Maletá Terry	Iteración asignada: 1
Prioridad en negocio: Baja	
<p>Descripción: La aplicación contiene las opciones de insertar, eliminar y modificar una categoría, estas funcionalidades se refieren a los requisitos funcionales RF8, RF9 y RF10.</p> <p>En la aplicación se debe escoger la opción Categorías, luego de haber escogido esta opción se mostrará un listado de categorías que pueden acceder a la aplicación. Si se quiere insertar una categoría se escoge la opción Insertar categoría y será mostrado un formulario para insertar los datos.</p>	

Para modificar una categoría, se selecciona la categoría a modificar y se hace clic en el botón Editar y se mostrará un formulario para editar el campo que desee y luego se debe presionar el botón Guardar.

Para eliminar una categoría se escoge la opción mostrar, luego aparece una interfaz con los datos de la categoría, se hace hacer clic en el botón Eliminar y le aparecerá una nueva interfaz para confirmar la eliminación de la categoría y presiona el botón Eliminar.

Prototipo de interfaz:



Tabla 15. HU_6 Listar noticias (Elaboración propia)

Historia de usuario	
Número: HU_6	Nombre: Listar noticias.
Programador: Liana Maletá Terry	Iteración asignada: 1

Prioridad en negocio: Baja
Descripción: En la aplicación el usuario va a escoger la opción mostrar noticias, luego será mostrada un interfaz para que el usuario escoja el sitio y el rango de fecha para ver las noticias y será mostrado un listado con todas las noticias que han sido recopilada del sitio que fue escogido.
Prototipo de interfaz:

Tabla 16. HU_8 Clasificar noticias (Elaboración propia)

Historia de usuario	
Número: HU_8	Nombre: Clasificar noticias.
Programador: Liana Maletá Terry	Iteración asignada: 1
Prioridad en negocio: Baja	

Descripción: Cuando el usuario se le muestra el listado de noticias después de haberlo elegido el sitio y el rango de fecha, este debe escoger una noticia, escoger la opción Categoría y se va a mostrar un formulario para asignar una categoría.

Prototipo de interfaz:

Título	Redactor	Cuerpo	Categoría	Cantidad Visitas	Cantidad Comentarios
Venezuela y Cuba	Joaquin Perez	Cuba y Venezuela cada dia ...		12	0

Elegir categoría

Nombre

Anexo 4. Diseños de casos de pruebas

Tabla 17. Descripción de las variables del caso de prueba de la HU_1 Autenticar usuario (Elaboración propia)

Variable	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Usuario	Campo de texto	No	No permite caracteres extraños
2	Contraseña	Campo de texto	No	No permite caracteres extraños

3	Recordar contraseña	Campo seleccionable	No	Permite que la contraseña del usuario se quede en base de dato
---	---------------------	---------------------	----	--

Tabla 18. Caso de prueba de la HU_1 Autenticar usuario (Elaboración propia)

Entrada	Resultados	Condiciones
El usuario introduce el usuario y la contraseña correctamente.	Se tiene acceso a la aplicación.	Establecer conexión con la red para acceder a la aplicación
El usuario introduce el usuario y la contraseña incorrectamente.	La aplicación le muestra un mensaje informando al usuario que el sitio ya existe.	Establecer conexión con la red para acceder a la aplicación
El usuario introduce el usuario y la contraseña correctamente y marca el campo recordar contraseña.	El sistema le muestra un mensaje informando al usuario autenticado que rellene los campos vacíos.	Establecer conexión con la red para acceder al HTML de los sitios y extraer las noticias de cada uno de ellos.

Tabla 19. Descripción de las variables del caso de prueba de la HU_2 Gestionar Usuario (Elaboración propia)

Variable	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Usuario	Campo de texto	No	No permite caracteres extraños
2	Contraseña	Campo de texto	No	No permite caracteres extraños

3	Rol	Campo de texto	No	No permite caracteres extraños
4	Email	Campo de texto	No	No permite caracteres extraños

Tabla 20. Caso de prueba de la HU_2 Gestionar usuario (Elaboración propia)

Entrada	Resultados	Condiciones
El usuario todos los datos correctamente.	Se muestra un mensaje que el usuario fue insertado correctamente.	Establecer conexión con la red
El usuario selecciona la opción mostrar usuario	La aplicación le muestra los datos del usuario.	Establecer conexión con la red
El usuario deja los campos vacíos.	La aplicación le muestra un mensaje informando al usuario que rellene los campos vacíos.	Establecer conexión con la red

Tabla 21. Descripción de las variables del caso de prueba de la HU_4 Gestionar categoría (Elaboración propia)

Variable	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Nombre	Campo de texto	No	Nombrar una categoría para las noticias

Tabla 22. Caso de prueba de la HU_4 Gestionar categoría (Elaboración propia)

Entrada	Resultados	Condiciones
---------	------------	-------------

El usuario coloca una nueva categoría.	Se muestra un mensaje un que la categoría fue insertada correctamente.	No acepta caracteres extraños.
El usuario deja el campo vacío.	Se muestra un mensaje un para que llene el campo.	No acepta caracteres extraños.

Tabla 23. Descripción de las variables del caso de prueba de la HU_6 Listar noticias (Elaboración propia)

Variable	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Sitio	Campo de seleccionar	No	Permite seleccionar un sitio
2	Fecha inicio	Campo de seleccionar	No	Permite seleccionar una fecha
3	Fecha fin	Campo de seleccionar	No	Permite seleccionar una fecha

Tabla 24. Caso de prueba de la HU_6 Listar noticias (Elaboración propia)

Entrada	Resultados	Condiciones
El usuario selecciona correctamente los datos.	Se muestra un listado de noticias de acuerdo al sitio y al rango de fecha.	El usuario debe estar autenticado para que la aplicación recopile y se le pueda mostrar las noticias que son guardadas en la base de dato.
El usuario no selecciona la fecha en los campos fecha inicio y fecha fin.	No será mostrada el listado de noticias	El usuario debe estar autenticado para que la aplicación recopile y se le pueda mostrar las noticias que son guardadas en la base de dato.

El usuario deja los campos vacíos.	No será mostrada el listado de noticias	El usuario debe estar autenticado para que la aplicación recopile y se le pueda mostrar las noticias que son guardadas en la base de dato.
------------------------------------	---	--

Tabla 25. Descripción de las variables del caso de prueba de la HU_8 Clasificar noticias (Elaboración propia)

Variable	Nombre del campo	Clasificación	Valor Nulo	Descripción
1	Categoría	Campo de seleccionar	No	Permite seleccionar una categoría para una noticia

Tabla 26. Caso de prueba de la HU_8 Clasificar noticias (Elaboración propia)

Entrada	Resultados	Condiciones
El usuario selecciona correctamente la categoría.	La noticia tendrá una categoría	El usuario debe seleccionar una noticia del listado de noticias.
El usuario no selecciona la categoría.	La noticia tendrá una categoría	El usuario debe seleccionar una noticia del listado de noticias.

Anexo 5. Diagrama de clases

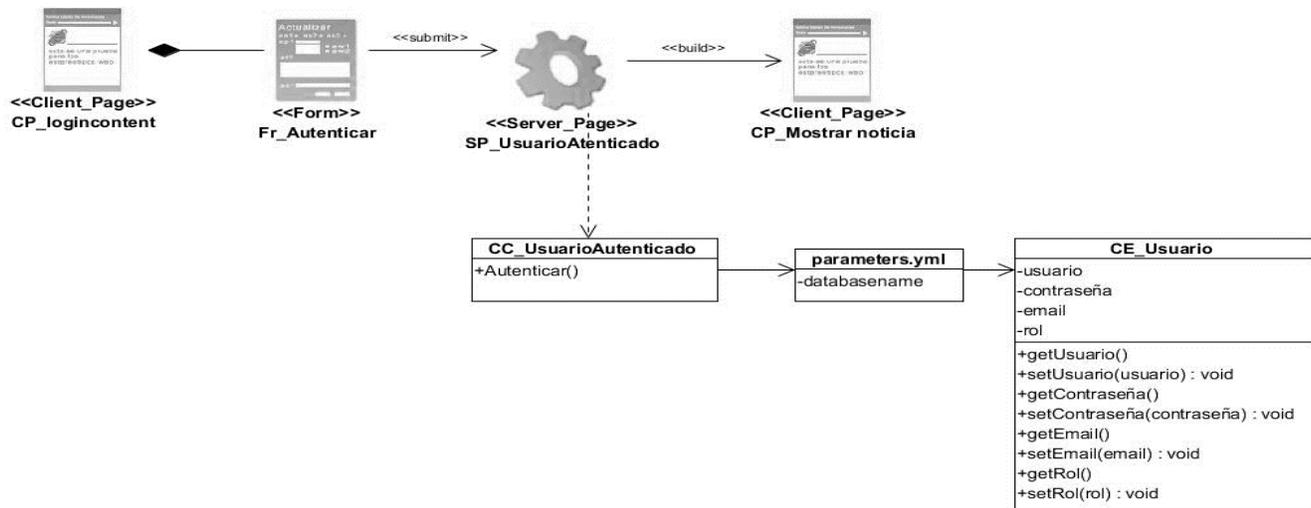


Figura 15. Diagrama de clase de la HU_1 Autenticar usuario (Elaboración propia)

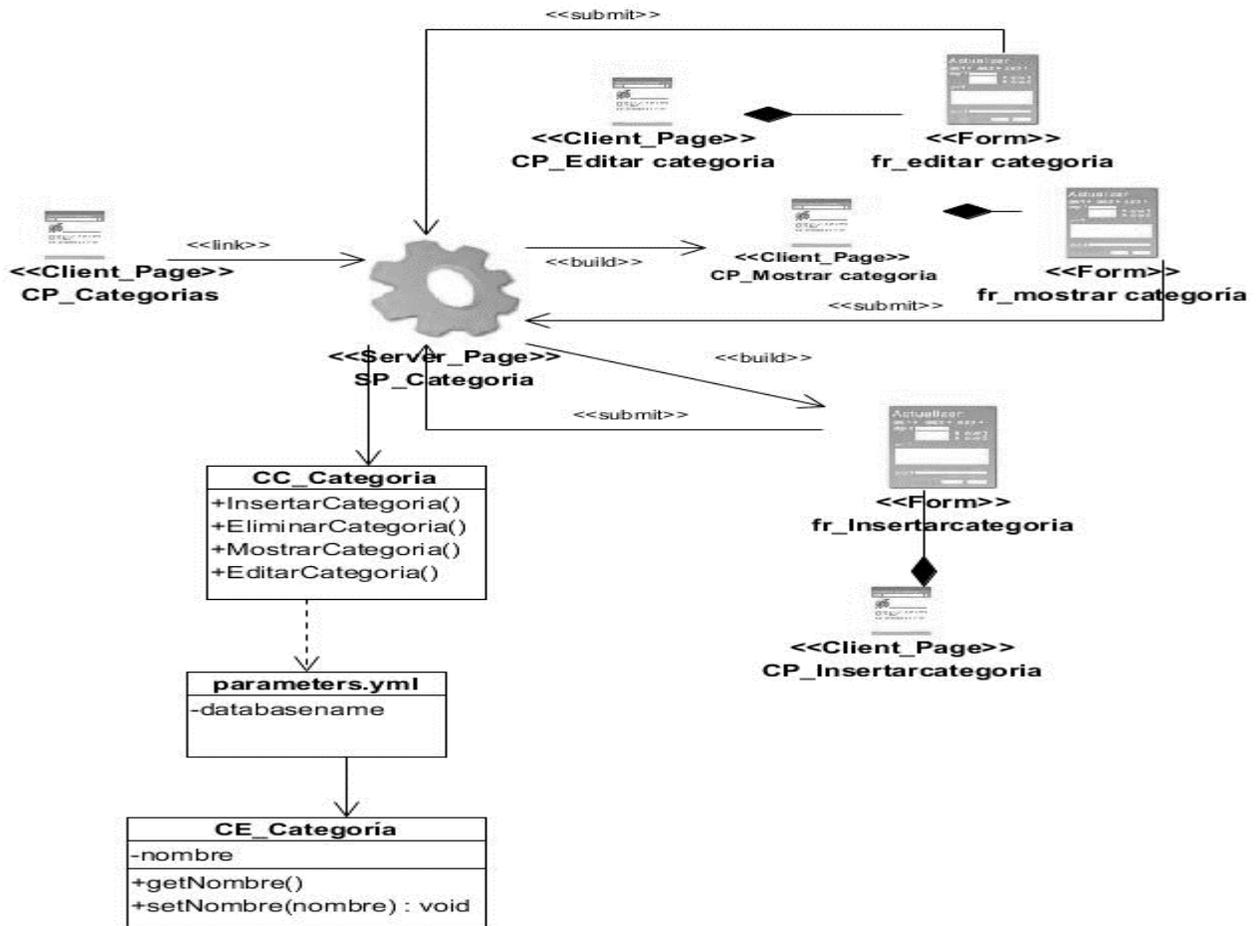


Figura 17. Diagrama de clase de la HU_4 Gestionar categoría (Elaboración propia)

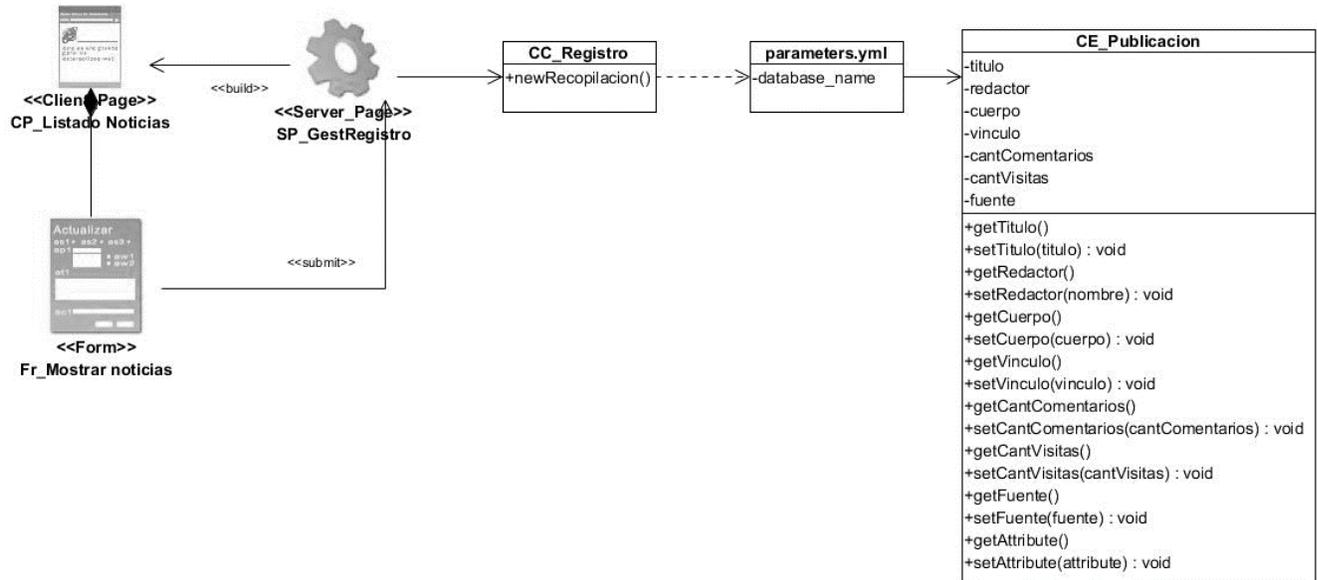


Figura 18. Diagrama de clases de la HU_6 Listar noticias (Elaboración propia)

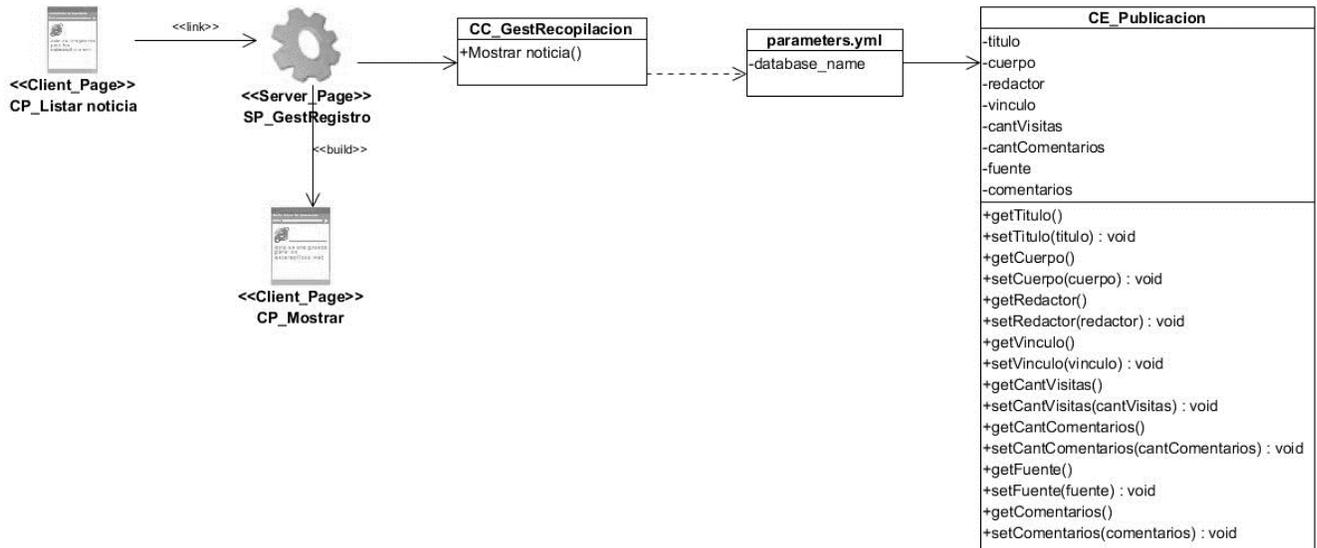


Figura 19. Diagrama de clases de la HU_7 Mostrar noticias (Elaboración propia)

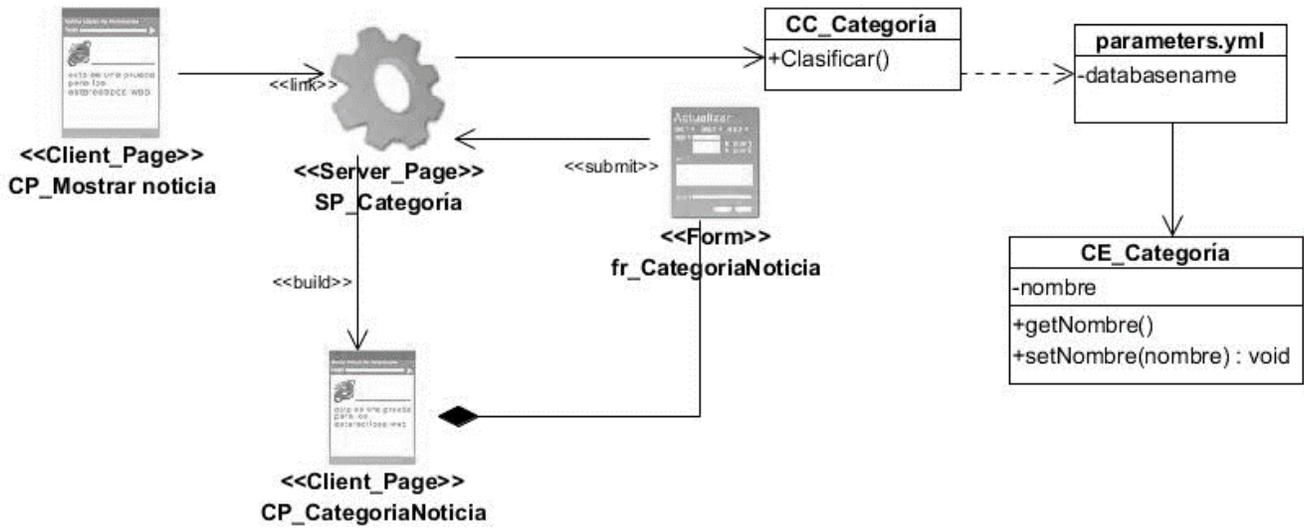


Figura 20. Diagrama de clases de la HU_8 Clasificar noticia (Elaboración propia)

Anexo 6. Diagramas de secuencia

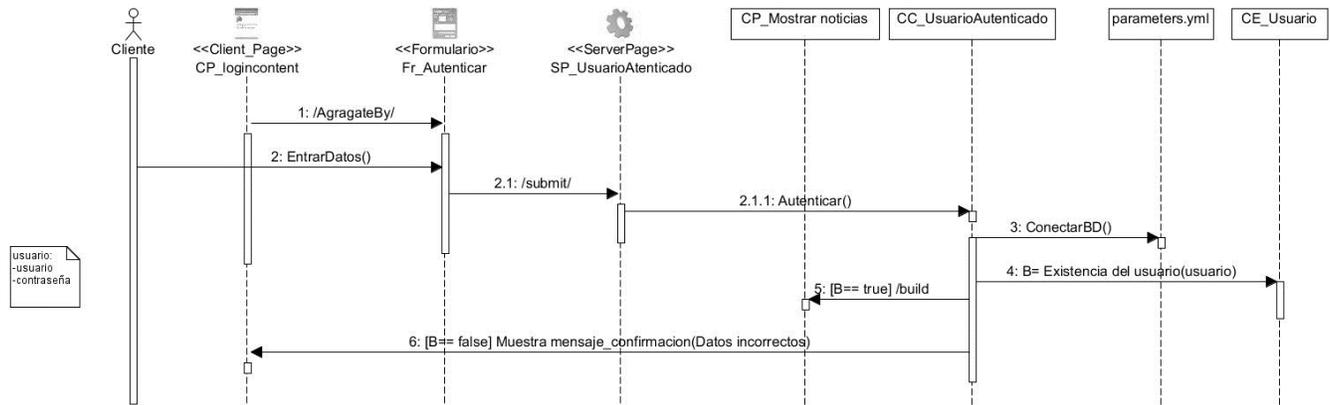


Figura 21. Autenticar usuario (Elaboración propia)

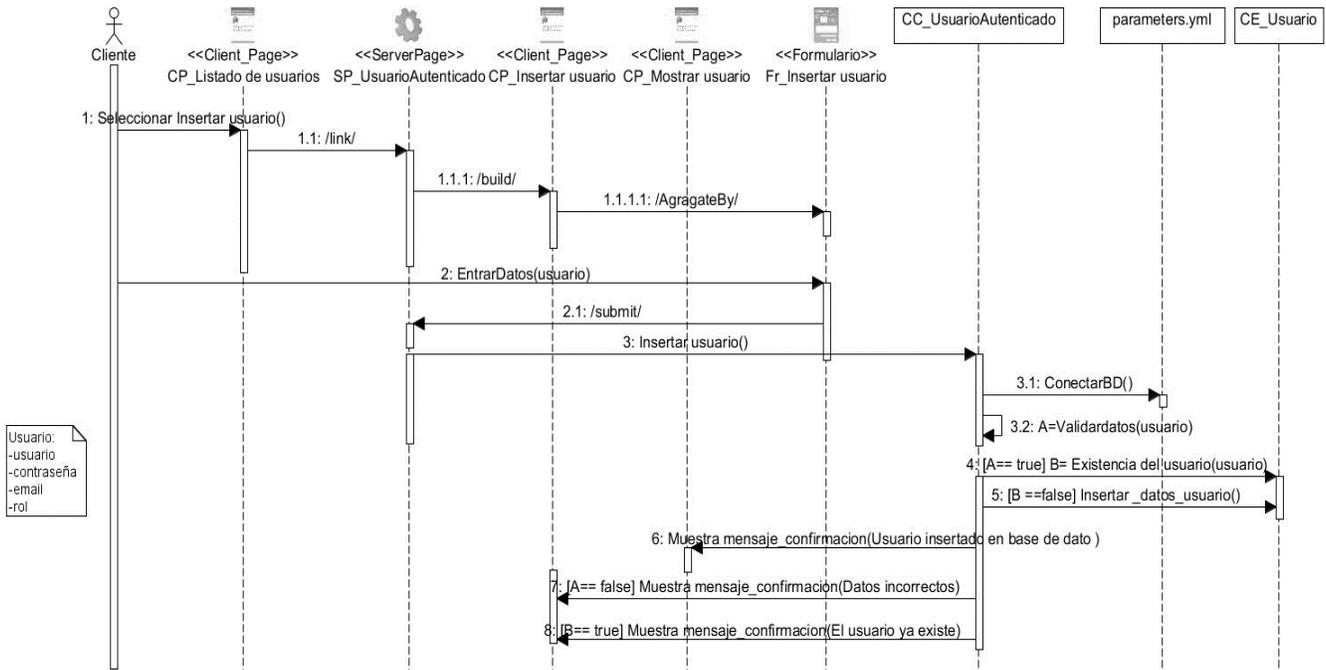


Figura 22. Insertar usuario (Elaboración propia)

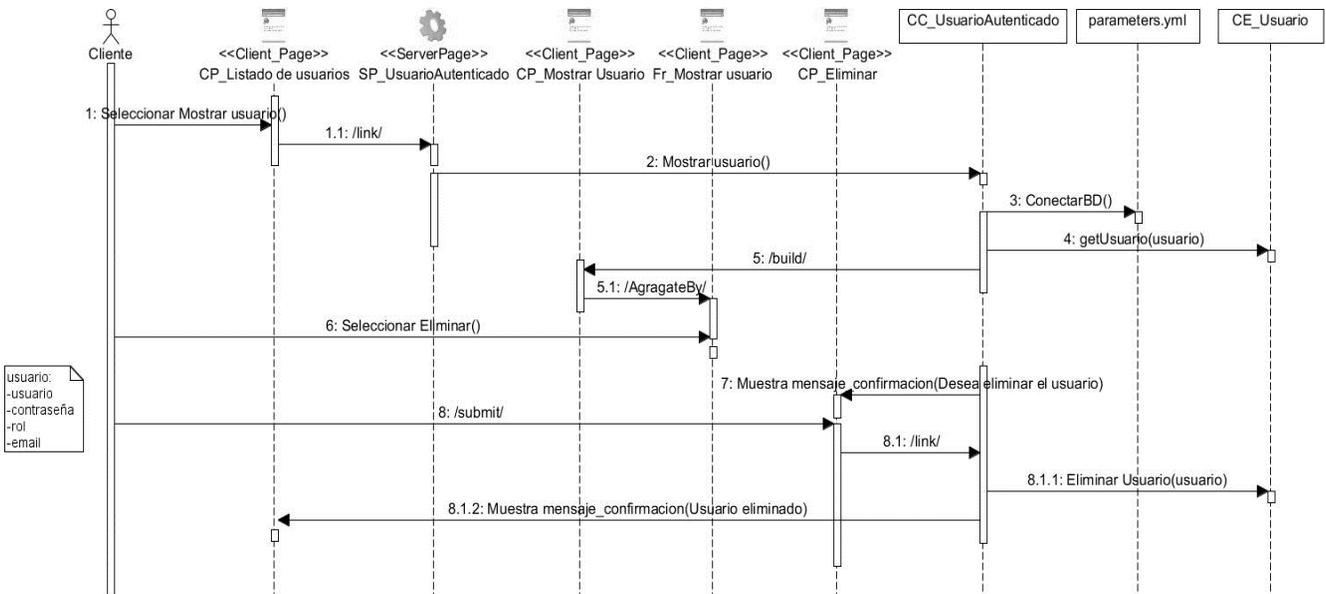


Figura 23. Eliminar usuario (Elaboración propia)

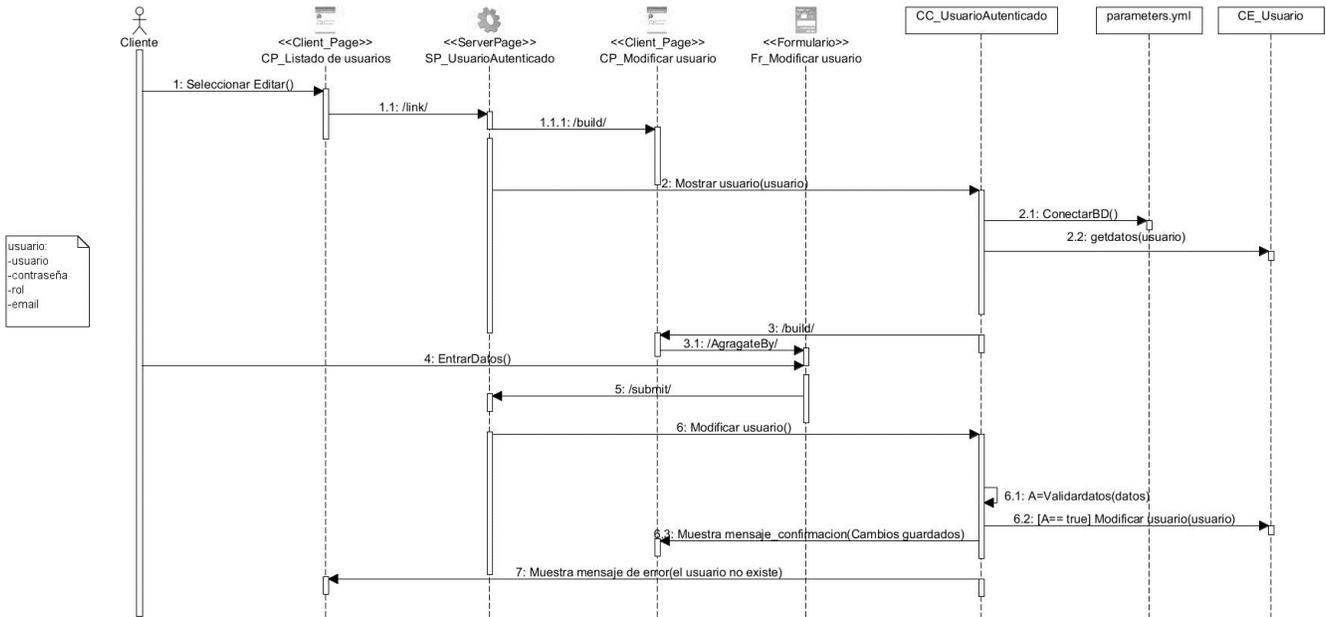


Figura 24. Modificar usuario (Elaboración propia)

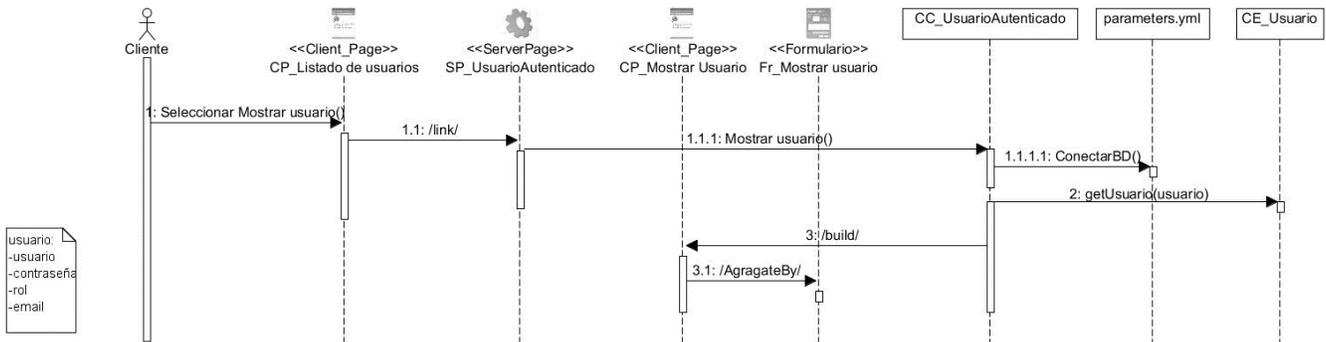


Figura 25. Mostrar usuario (Elaboración propia)

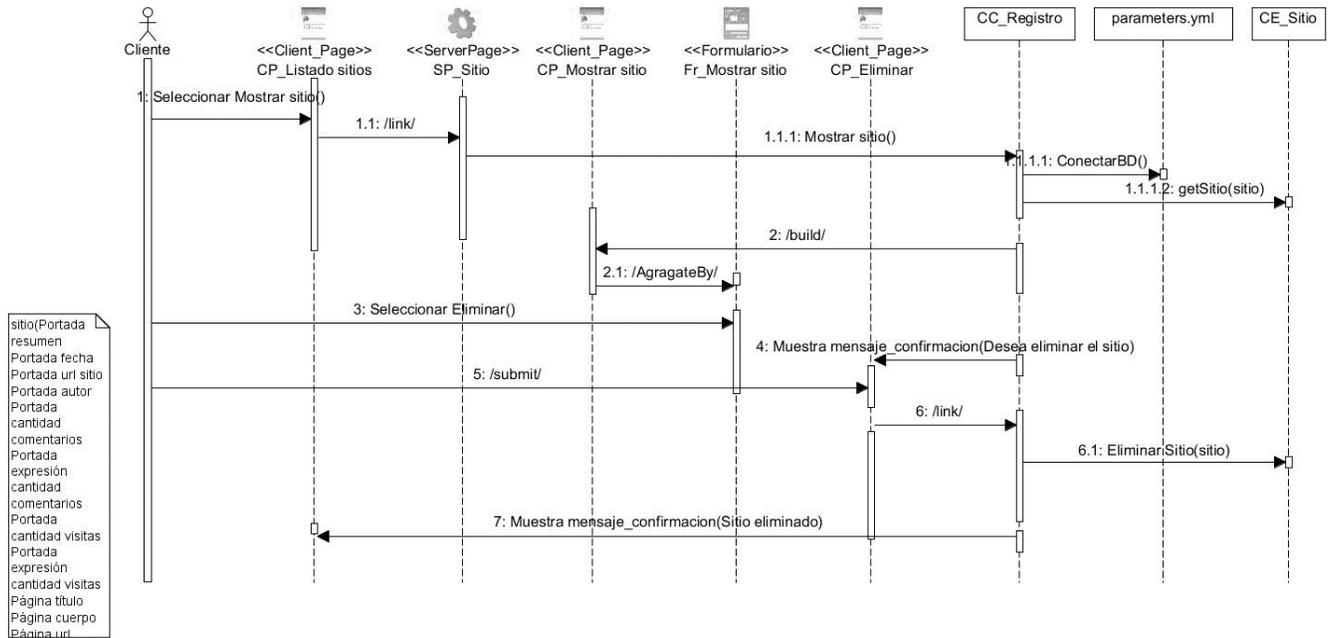


Figura 26. Eliminar sitio (Elaboración propia)

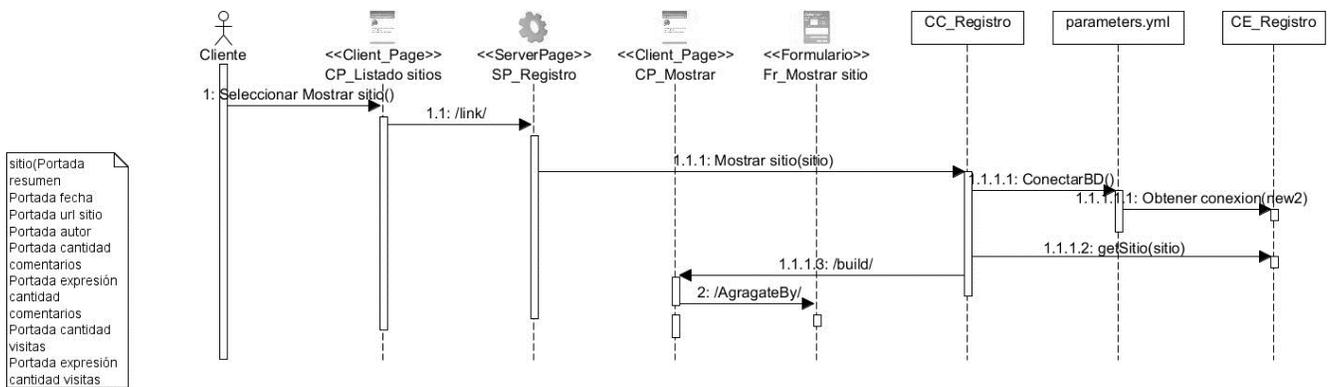


Figura 27. Mostrar sitio (Elaboración propia)

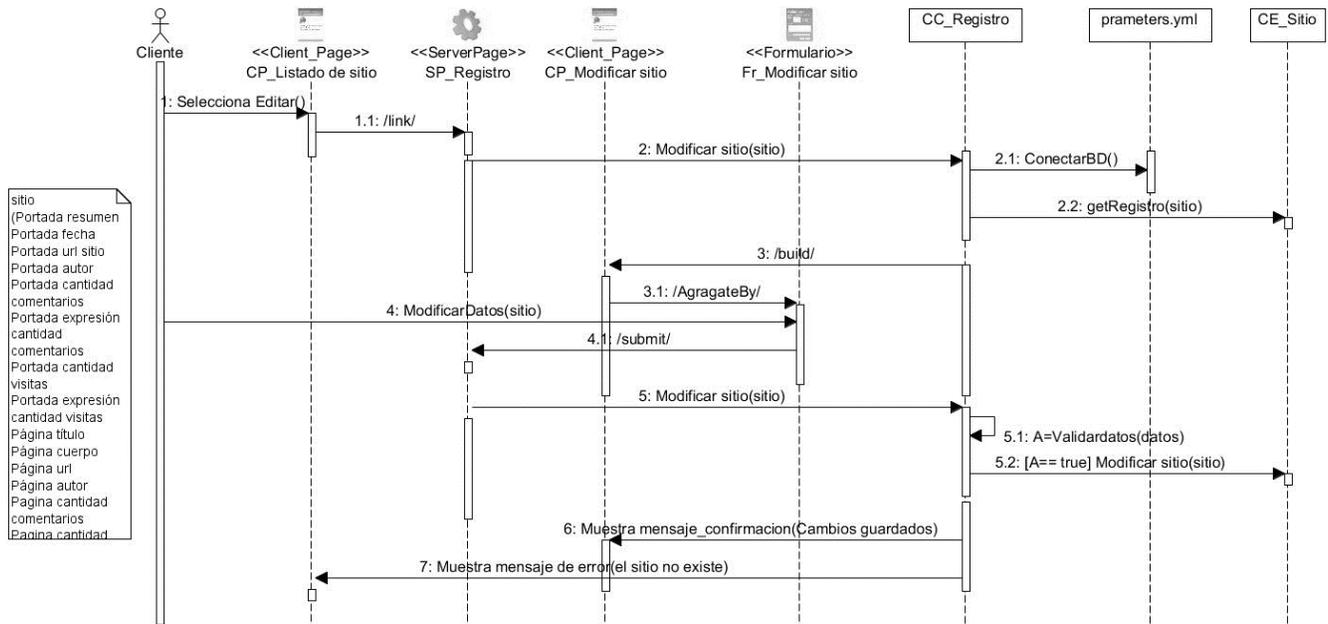


Figura 28. Modificar sitio (Elaboración propia)

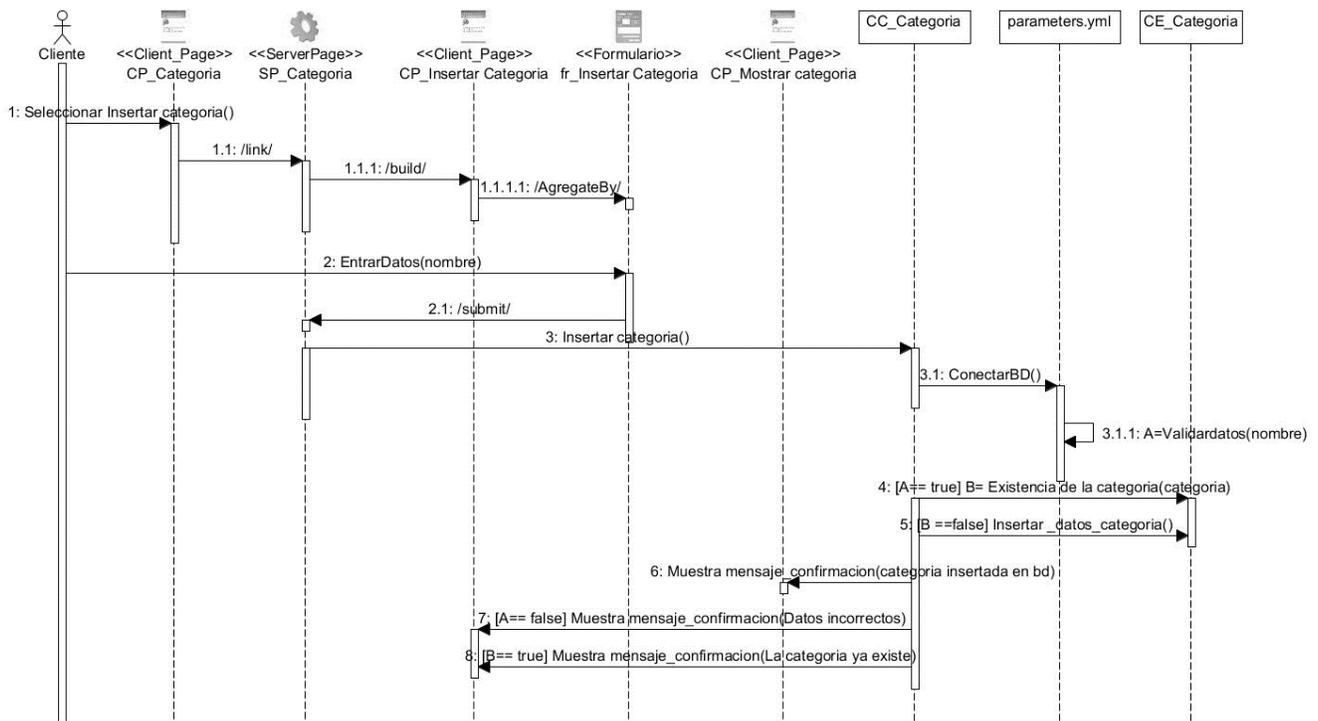


Figura 29. Insertar categoría (Elaboración propia)

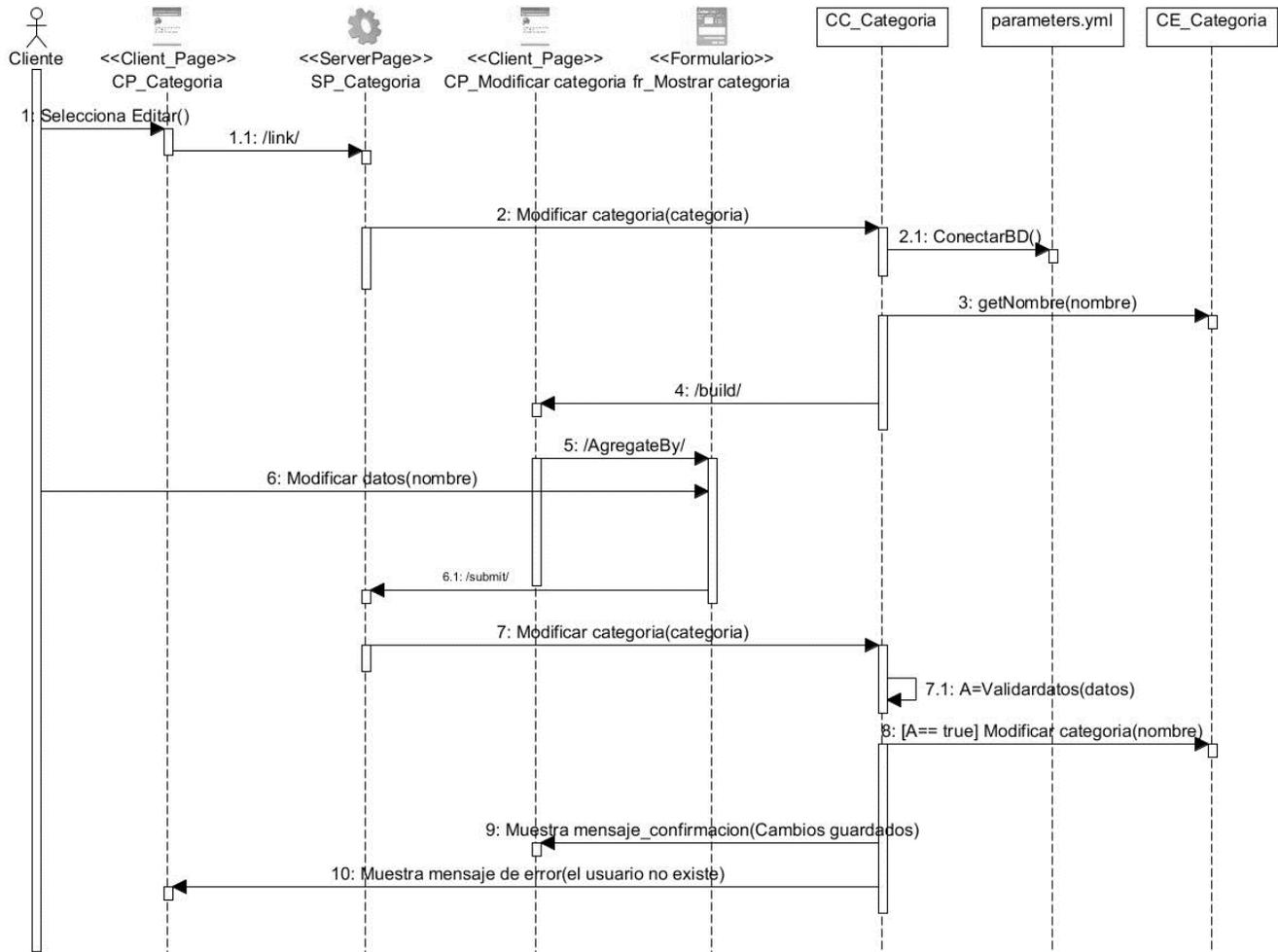


Figura 30. Modificar categoría (Elaboración propia)

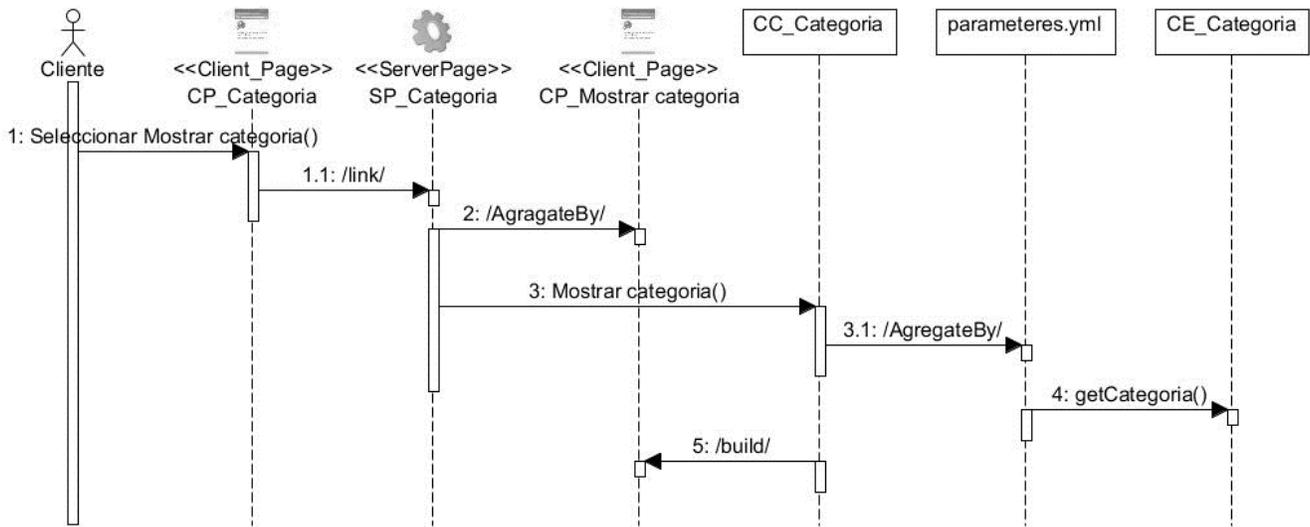


Figura 31. Mostrar categoría (Elaboración propia)

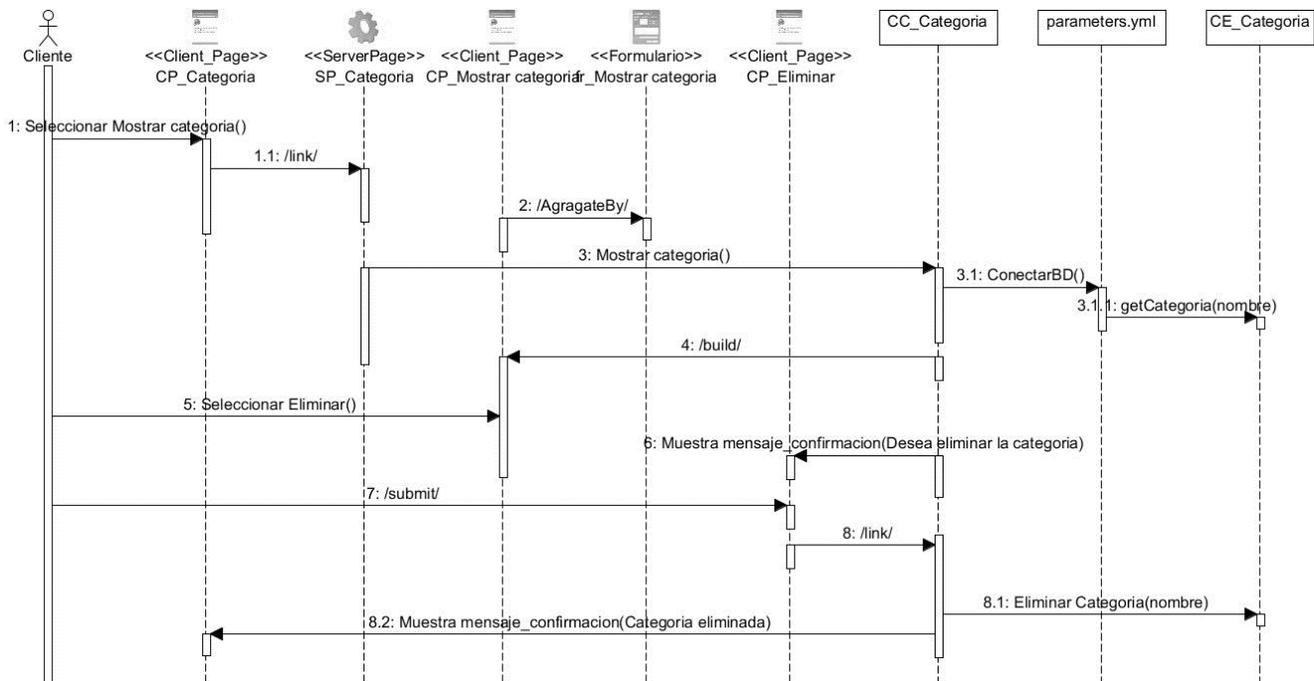


Figura 32. Eliminar categoría (Elaboración propia)

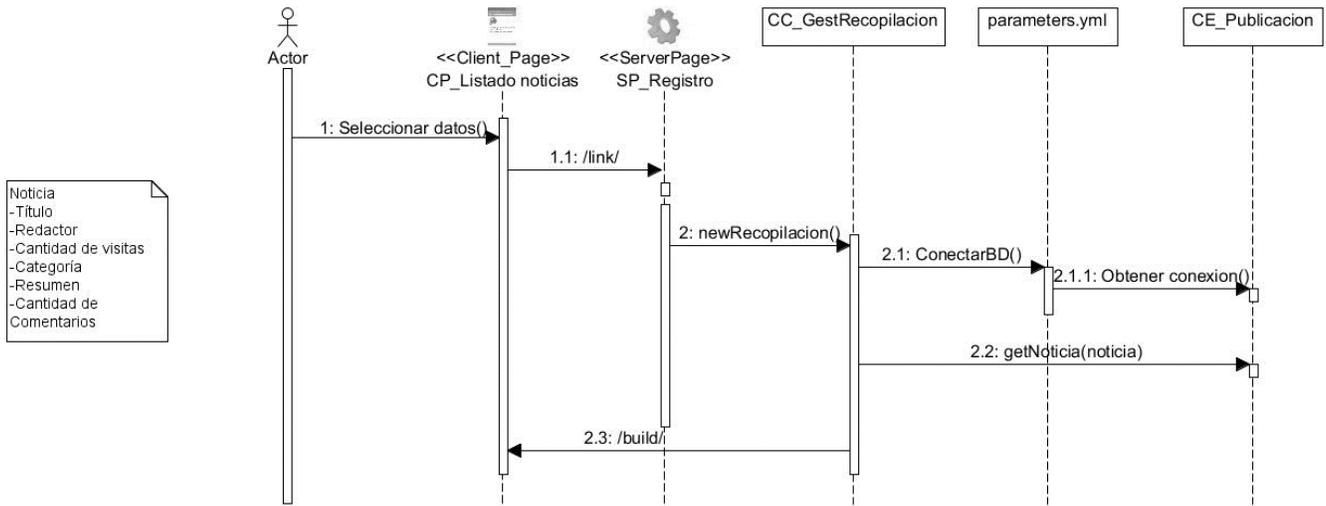


Figura 33. Listar noticia (Elaboración propia)

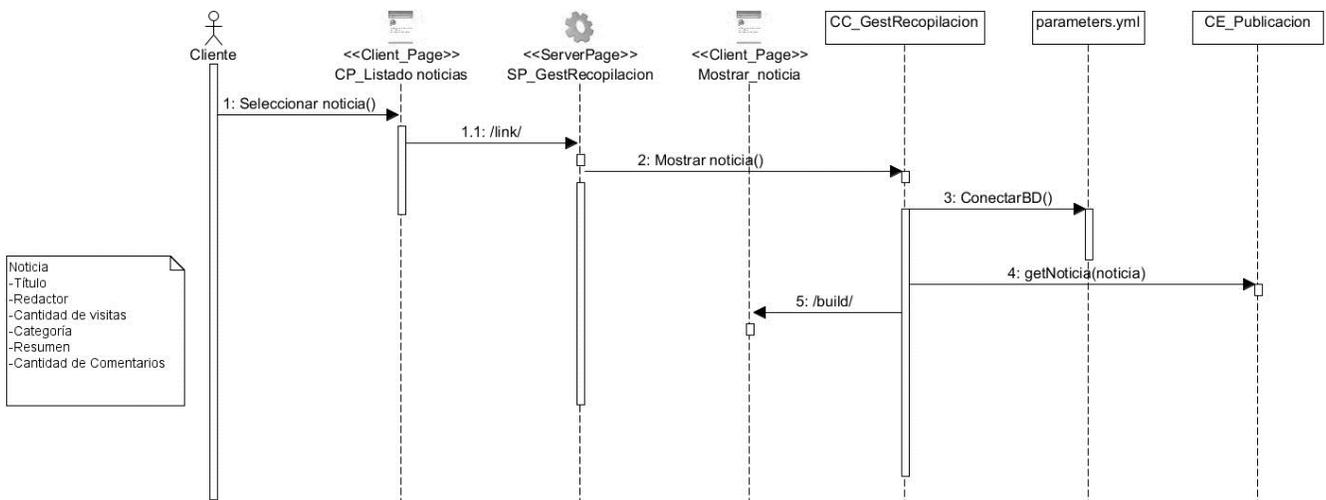


Figura 34. Mostrar noticia (Elaboración propia)

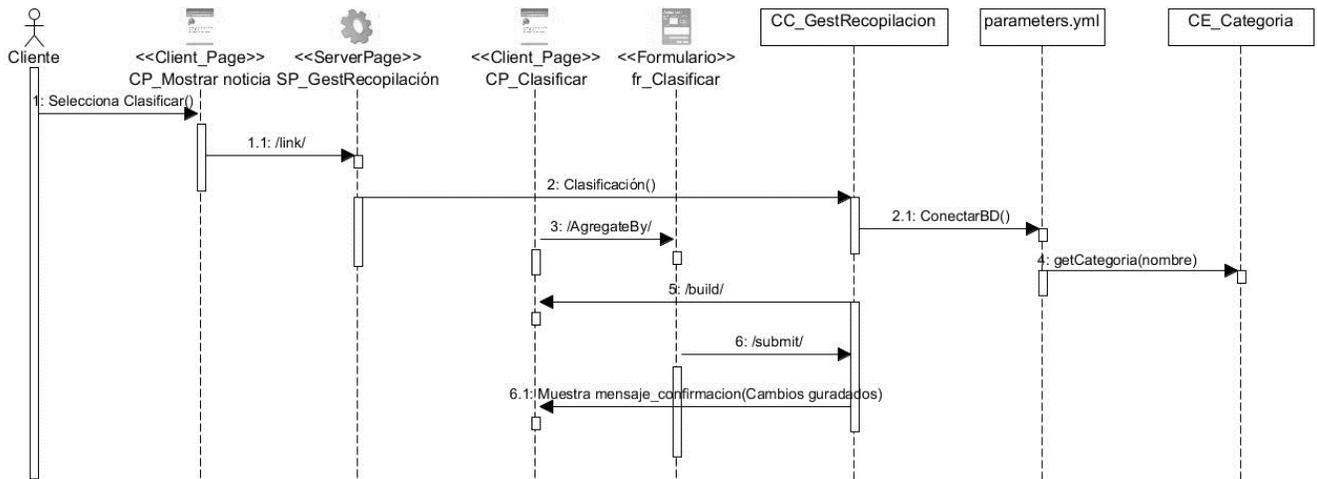


Figura 35. Clasificar noticias (Elaboración propia)

Anexo 7. Encuestas

Encuesta a especialistas del proyecto CIDI en los departamentos DOWAI y SINI para la validación de la propuesta de solución.

Especialistas, los invito a responder el siguiente cuestionario con el fin de conseguir su colaboración en la presente investigación. Solicito que exprese en sus respuestas criterios verídicos que guíen al autor del trabajo. Marque en cada pregunta con una X en una sola opción. Por el tiempo brindado, muchas gracias.

1. ¿Considera que es necesario la recopilación automática de noticias?

Sí ___ No ___ No sé ___

2. ¿Estima provechosa la herramienta desarrollada para el análisis de los principales temas tratados acerca de las publicaciones en la web?

Sí ___ No ___ No sé ___

3. Luego de haber visto las noticias de los sitios recopiladas por el subsistema de recopilación automático de noticias a través de la interfaz web desarrollada, indique en que medida le gusta la propuesta de solución

___ Me gusta mucho

___ Me disgusta más de lo que me gusta

___ Me gusta más de lo que me disgusta

___ No me gusta nada

___ Me da lo mismo

___ No sé decir

- 4 ¿Qué importancia usted le otorga a la creación del subsistema de recopilación automática de noticias de los medios informativos de la UCI?
- 5 ¿Considera que subsistema de recopilación automática de noticias de los medios informativos de la UCI se puede extender a otros centros de desarrollo de la universidad?