



Facultad 2

Sistema web para la gestión de la información en temas sociopolíticos.

Trabajo de diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autor: José David Avila Batista

Tutor: MSc. Darling Darias Pérez

MSc. Eddy McDonald Torres

La Habana, noviembre de 2023

Año 65 de la Revolución

DECLARACIÓN DE AUTORÍA

El autor del trabajo de diploma con título “*Sistema web para la gestión de la información en temas sociopolíticos*”, concede a la Universidad de las Ciencias Informáticas (UCI) los derechos patrimoniales de la investigación, con carácter exclusivo. De forma similar se declara como único autor de su contenido. Para que así conste firma la presente a los <día> días del mes de octubre del año 2023.

Jose David Avila Batista

Firma del Autor

Darling Darías Pérez

Firma del Tutor

DATOS DEL LOS CONTACTOS

Tutora:

Msc. Darling Darías Pérez: Graduada de Ingeniera en Ciencias Informáticas. Profesora Principal de 5to año en la carrera de ICI de la Facultad 2. Posee 15 años de experiencia laboral impartiendo clases de las asignaturas de Ingeniería de Software, Gestión de Software, Base de datos, Proyecto de Investigación y Desarrollo e Introducción a las Ciencias Informáticas. De ellos, 6 años como directiva en un Centro y luego en la facultad. (subdirectora del Centro de Informática Médica: CECIM y Vicedecana de la Facultad 7) Además de tener experiencia en el rol de Analista y probadora de la calidad en dicho centro productivo. Realizó y defendió la maestría externa de Informática en Salud, en el Centro de Cibernética Aplicada a las Matemáticas. Ha participado en eventos científicos y grupos investigativos.

Correo electrónico: ddarias@uci.cu

MSc. Eddy McDonald Torres: Graduado de Master en Estudios Políticos y Sociales por la Universidad de La Habana en 2018. Profesor auxiliar de ciencias sociales. Ha participado en múltiples eventos nacionales e internacionales, así como ha publicado en revistas científicas y especializadas sobre temas pedagógicos, de ciencias sociales, y, específicamente de ciencias políticas. Sus investigaciones se vinculan con el impacto sociopolítico de las tecnologías de la información y las comunicaciones (Tic), las Relaciones Políticas Internacionales, la sociedad civil y la ciencia, tecnología y la política, entre otros.

Ha sido conferencista y profesor de enseñanza posgraduada en la Universidad de las Ciencias Informáticas (UCI), en el Instituto Internacional de Periodismo José Martí, en la Universidad de Holguín y para organizaciones juveniles y estudiantiles sobre asuntos vinculados a contenidos socioeconómicos y políticos relacionados con las Tic.

Ha cursado posgrados en temas sociopolíticos, de ciencia y tecnología, así como de idioma extranjero, en diferentes instituciones educativas e investigativas cubanas y del Centro de Investigaciones Científicas Sociales y Tecnológicas del Perú (CICSTEP).

Correo electrónico: torres@uci.cu

AGRADECIMIENTOS

Agradecer a mi familia por el apoyo siempre por no rendirse conmigo ni dejar q me rendiera yo tampoco, a mi madre faro y guía en mi vida y la que me ha hecho ser como soy esto es para ti me toca ahora devolverte lo que has hecho por mi aunque para eso me lleve 10 vidas más, a mi papá que también siempre ha estado conmigo y pendiente de todo, a mis abuelos que doy gracias que puedan estar aun conmigo y verme conseguir metas en mi vida, a mi abuelo Alcides que desde el cielo me ha cuidado y me ha guiado y solo espero que te sientas orgulloso de tu nieto, a mi hermano que muy a su manera pero sé que se ha preocupado por mí y que también está feliz que es un referente para mí, mi tío, mi tía y mi primo gracias por estar ahí en todo el camino por apoyarme. Gracias a mis hermanos de vida que desde la distancia siempre han estado conmigo me decían que el traje se iba a echar a perder y q no lo iba a usar y mira al final parece q si lo estreno, a los amigos que me llevo de la universidad a los que están y los que se han ido por el camino a mis compañeros de 1er año nos mantuvimos los pocos pero lo logramos, a la familia del 94 gracias por esa bienvenida por ayudarme a instalarme después de esa llegada desastrosa, a la familia de la pelota gracias por las risas y por la amistad ha sido un lujo para mi ser parte de este equipo el año que viene nos llevamos ese oro, a ese grupo con el que llevo viviendo ya 3 años al final se les coje cariño gracias por todos los momentos desde luego sin ustedes esto no habría sido igual y una bonita familia que me llevo de aquí,

Gracias a todos ha sido un placer compartir todos estos años con ustedes,

Se les quiere,

JOSE DAVJD

RESUMEN

La Dirección de Historia y Marxismo Leninismo de la Universidad de las Ciencias Informáticas necesita gestionar la información en temas sociopolíticos, proceso que actualmente se realiza de forma manual lo cual hace que sea complejo dada la considerable cantidad de información. El presente estudio de investigación se centra en la creación de un sistema web concebido con tecnologías de código abierto para simplificar dicho proceso. Este sistema en línea fue dirigido por la metodología Extreme Programming (XP). En cuanto a las tecnologías del lado del usuario, se emplearon: HTML5, JavaScript, CSS3, respaldadas por el marco de diseño Bootstrap. El lenguaje del lado del servidor utilizado fue JavaScript mediante su framework NodeJS, se empleó el Visual Studio Code como entorno de desarrollo, PgAdmin 4 y PostgreSQL para la administración de bases de datos. Se incorporaron funcionalidades en la gestión de temas políticos, seminarios, cursos, entre otros. Asimismo, se introdujeron funcionalidades para cargar documentos e informes en el sistema. Se llevaron a cabo pruebas de software a la propuesta de solución, las cuales proporcionan niveles de confirmación de las funcionalidades y la satisfacción por parte del usuario.

PALABRAS CLAVE: Funcionalidades, pruebas de software, sistema web, gestión de información.

ABSTRACT

The Department of History and Marxism-Leninism of the University of Computer Sciences wants to manage information on sociopolitical issues, a process that is currently carried out manually, which makes it complex given the considerable amount of information. This research study focuses on the creation of a web system designed with open source technologies to simplify this process. This online system was driven by the Extreme Programming (XP) methodology. Regarding user-side technologies, the following were used: HTML5, JavaScript, CSS3, supported by the Bootstrap design framework. The server-side language used was JavaScript through its NodeJS framework, Visual Studio Code was used as the development environment, PgAdmin 4 and PostgreSQL for database administration. Functionalities were incorporated in the management of political issues, seminars, courses, among others. Likewise, functionalities were introduced to upload documents and reports to the system. Software tests were carried out on the proposed solution, which provide levels of confirmation of the functionalities and user satisfaction.

KEYWORDS: Functionalities, software testing, web system, information management.

TABLA DE CONTENIDOS

INTRODUCCIÓN	2
CAPÍTULO I: FUNDAMENTOS TEÓRICO-METODOLÓGICOS SOBRE EL OBJETO DE ESTUDIO	7
I.1 Conceptos asociados al dominio del problema	7
I.2 Sistemas similares existentes en el mundo	8
I.3 Metodologías, lenguajes, herramientas y tecnologías informáticas para el desarrollo del sistema.	9
Conclusiones del capítulo	14
CAPÍTULO II: DISEÑO DE LA SOLUCIÓN PROPUESTA AL PROBLEMA CIENTÍFICO	17
II.1 Exploración	17
II.2 Planificación	20
II.2.1 Estimación del esfuerzo	20
II.2.2 Plan de iteraciones	21
II.3 Diseño	23
II.3.2 Patrón de diseño	25
II.4 Modelo de datos	33
Conclusiones del capítulo	34
CAPÍTULO III: VALIDACIÓN DE LA SOLUCIÓN PROPUESTA	35
III.2 Tareas de ingeniería	35
III.3 Pruebas de software	36
III.3.1 Estrategia de prueba	37
III.3.4 Validación del Sistema	40
Conclusiones del capítulo	41
CONCLUSIONES FINALES	42
RECOMENDACIONES	43
REFERENCIAS BIBLIOGRÁFICAS	44
ANEXOS	¡ERROR! MARCADOR NO DEFINIDO.

ÍNDICE DE TABLAS

Tabla 1 HU 1 Autenticar Usuario	19
Tabla 2 Estimación de esfuerzo.....	20
Tabla 3 Plan de iteraciones / cantidad de semanas	22
Tabla 4 Plan de entrega.....	23
Tabla 5 Tarjeta CRC 1 Comité Científico.....	30
Tabla 6 Tarjeta CRC 2 Consejo Editorial	30
Tabla 7 Tarjeta CRC 3 Tema Político	31
Tabla 8 Tarjeta CRC 5 Conferencia.....	31
Tabla 9 Tarjeta CRC 6 Seminario	31
Tabla 10 Tarjeta CRC 7 Curso	32
Tabla 11 Tarjeta CRC 10 link Vínculos de Interés	32
Tabla 12 Tarea de Ingeniería 1 Autenticar Usuario	36
Tabla 13 Caso de Prueba de Aceptación 1	40
Tabla 14 HU 2 Gestionar Usuario	49
Tabla 15 HU 3 Gestionar Comité Científico	49
Tabla 16 HU 4 Gestionar Consejo Editorial	49
Tabla 17 HU 5 Gestionar Temas Políticos.....	50
Tabla 18 HU 6 Gestionar Conferencia.....	50
Tabla 19 HU 7 Gestionar Seminarios	51
Tabla 20 HU 8 Gestionar Cursos.....	51
Tabla 21 HU 9 Insertar link a vínculos de interés	51
Tabla 22 HU 10 Insertar contenido inherente al portal	52
Tabla 23 HU 11 Buscar Usuarios	52
Tabla 24 HU 12 Buscar Tema Político.....	53
Tabla 25 HU 13 Buscar Conferencia	53
Tabla 26 HU 14 Buscar Seminario	54
Tabla 27 HU 15 Buscar Curso	54
Tabla 28 Tarea de Ingeniería 2 Gestionar Usuario	56

Tabla 29 Tarea de Ingeniería 3 Gestionar Comité Científico	56
Tabla 30 Tarea de Ingeniería 4 Gestionar Consejo Editorial	57
Tabla 31 Tarea de Ingeniería 6 Gestionar Tema Político	57
Tabla 32 Tarea de Ingeniería 8 Gestionar Conferencias	58
Tabla 33 Tarea de Ingeniería 9 Gestionar Seminario	59
Tabla 34 Tarea de Ingeniería 10 Gestionar Curso.....	59
Tabla 35 Tarea de Ingeniería 13 Insertar link a vínculos de interés	60
Tabla 36 Tarea de Ingeniería 14 Insertar contenido inherente al portal	60
Tabla 37 Tarea de Ingeniería 15 Buscar Usuario	60
Tabla 38 Tarea de Ingeniería 17 Buscar Tema político	61
Tabla 39 Tarea de Ingeniería 18 Buscar Conferencia	61
Tabla 40 Tarea de Ingeniería 19 Buscar Seminario	61
Tabla 41 Tarea de Ingeniería 20 Buscar curso.....	62
Tabla 42 Caso de Prueba de Aceptación 2	63
Tabla 43 Caso de Prueba de Aceptación 3	63
Tabla 44 Caso de Prueba de Aceptación 4	64
Tabla 45 Caso de Prueba de Aceptación 7	65
Tabla 46 Caso de Prueba de Aceptación 8	65
Tabla 47 Caso de Prueba de Aceptación 9	66
Tabla 48 Caso de Prueba de Aceptación 10	67
Tabla 49 Caso de Prueba de aceptación 11.....	67
Tabla 50 Caso de Prueba de Aceptación 12	68
Tabla 51 Caso de Prueba de Aceptación 13	68

ÍNDICE DE FIGURAS

Figura 1: Modelo Físico de datos diseñados para el sistema	33
--	----

Opinión del(os) tutor(es)

OPINIÓN DEL(OS) TUTOR(ES)

AVAL DEL CLIENTE

INTRODUCCIÓN

En la época actual, considerada como era de la información y la explosión de las tecnologías, en que la humanidad ha alcanzado un desarrollo imprevisible y cada día son mayores las diferencias sociales, políticas y económicas, se habla constantemente sobre la sociedad de la información, donde el factor esencial de progreso es el conocimiento. Esta nueva sociedad, con organizaciones basadas en el aprendizaje, cuyo capital máspreciado es el ser humano, se sustenta en un desarrollo tecnológico sin precedentes, en el que las tecnologías de la informática y las comunicaciones constituyen un medio esencial para procesar y transmitir información y conocimiento, factor fundamental para la creación de riquezas.

En la era de la información, uno de los principales problemas es su exceso, es necesario invertir mucho tiempo en ella debido, entre otras causas, a la liberación de los mecanismos regulatorios existentes en materia de publicaciones, sobre todo como resultado del surgimiento y desarrollo de Internet; por esta razón, es frecuente encontrar un número significativo de publicaciones redundantes y de baja calidad mezcladas con otras importantes y sólidas, difíciles de hallar entre la información ruidosa que la "envuelve". Son múltiples las definiciones de un mismo concepto, por ejemplo, existe una multitud de definiciones, que en su conjunto son bastante dispares y, en muchos casos, ambigua. Esto se debe, por una parte, al enfoque con el que se define el concepto (cultural, organizativo o tecnológico), y por otra, a que existen diferentes aproximaciones, en dependencia de su aplicación concreta a casos reales. (Quiroga, 2002)

Actualmente la gestión de la información constituye un elemento clave para su análisis en diversos campos, incluyendo temas sociopolíticos. En este contexto, el desarrollo de herramientas informáticas se ha vuelto cada vez más importante y relevante, debido a la variedad, amplitud y elevado número que circulan a diario, muy difícil de abarcar por los especialistas en otras condiciones.

El almacenamiento de grandes cantidades de datos empleando la informática, tanto en sistemas de bases de datos como en la nube, ha permitido a las organizaciones recopilar y

conservar información de manera más segura y accesible, eliminando la necesidad de grandes archivos físicos y facilitando su búsqueda y recuperación.

La informática ha proporcionado herramientas y software especializados para la gestión de la información, lo que ha llevado a avances significativos en campos como la medicina, la investigación científica, la educación, el comercio, la gestión de procesos de producción en la industria, la administración de recursos humanos y financieros, gestión de ventas, entre otros. En resumen, la informática es una disciplina fundamental en la sociedad actual que se aplica en una gran variedad de campos y que tiene un impacto significativo en la forma en que se vive y se trabaja. (InCuatro Informatica para empresas, 2019)

Cuba es un país, que a pesar de las limitaciones económicas que presenta debido a las consecuencias del bloqueo, no ha quedado exento de la utilización de las Tecnologías de la Información y las Comunicaciones (TIC). Desde hace un tiempo el gobierno cubano se ha enfocado en la informatización de los distintos sectores de la sociedad, respaldado mayormente por las investigaciones, aportes y proyecciones hechas por entidades o grupos de desarrollo de software. Se utilizan sistemas de información y bases de datos en diversos ámbitos y se promueve la formación en informática para impulsar la innovación y el desarrollo tecnológico en el país. (Ruiz, Y. Álvarez, A. & González, G. (2019)).

Hoy día, las Tecnologías de la Información y la Comunicación (TIC) constituyen un componente importante en el ámbito educativo y el impacto que producen en las instituciones de educación superior es cada vez más evidente, ya que dada su versatilidad brindan posibilidades de renovación en los contenidos, a fin de que éstos puedan ser reutilizados en los procesos de enseñanza-aprendizaje, apoyados en los entornos de aprendizaje (Rodríguez y López, 2018).

Sin embargo, el tratamiento de datos e información mediante la informática también plantea desafíos en términos de seguridad y privacidad. Es importante tener en cuenta que en Cuba existe una regulación sobre el uso y acceso a la información. Se debe cumplir con las leyes y regulaciones vigentes para garantizar la seguridad y privacidad de los datos, así como para

evitar el uso inapropiado de la información. (Ley 149/2022 “De protección de datos personales” y Resolución 58/2022 del Ministerio de Comunicaciones “Reglamento para la Seguridad y Protección de los Datos Personales en Soporte Electrónico”)

La Dirección de Historia y Marxismo Leninismo de la Universidad de las Ciencias Informáticas tiene entre otras, la necesidad de un sitio para organizar, gestionar bibliografía y documentos acorde a los temas que utilizan para impartir sus clases o para su preparación, pero el personal que allí labora no cuenta con una gran preparación en temas de informática, lo que les dificulta el uso de herramientas o sistemas complejos para la gestión de la información y aunque tienen los documentos en sus computadoras los trabajan de forma manual, lo que trae como consecuencia una mayor lentitud en el proceso y el riesgo de que factores físicos o virus informáticos puedan dañarlos o destruirlos.

Se necesita mantener los documentos, que puedan ser leídos y consultados, en un mismo lugar para asegurar la organización de este proceso. El presente trabajo responde a la **interrogante** acerca de ¿Cómo facilitar el proceso de gestión de la información asociada a temas de las ciencias sociales de la Dirección de Historia y Marxismo Leninismo en la Universidad de las Ciencias Informáticas?, teniendo como **objetivo** desarrollar un sistema web para la gestión de la información en temas sociopolíticos de la Dirección de Historia y Marxismo Leninismo de la UCI. El **objeto de estudio** es la gestión de la información en temas sociopolíticos, teniendo como **campo de acción** la información de temas sociopolíticos en la Dirección de Historia y Marxismo Leninismo de la UCI.

Para esto se llevan a cabo las siguientes tareas científicas específicas:

1. Sistematización de los principales fundamentos teóricos-metodológicos para el desarrollo de un sistema informático que facilite la gestión de la información en temas sociopolíticos en la Dirección de Historia y Marxismo Leninismo de la Universidad de las Ciencias Informáticas
2. Diagnóstico del estado actual del proceso de gestión de la información en temas sociopolíticos en la Dirección de Historia y Marxismo Leninismo

3. Desarrollo del sistema informático para la gestión de la información en temas sociopolíticos en la Dirección de Historia y Marxismo Leninismo de la Universidad de las Ciencias Informáticas
4. Validación de cuánto favoreció el desarrollo del sistema informático para la gestión de la información en temas sociopolíticos en la Dirección de Historia y Marxismo Leninismo de la Universidad de las Ciencias Informáticas

Métodos científicos de investigación utilizados

Teóricos:

- **Histórico lógico:** para determinar las formas en que se ha llevado a cabo la gestión de la información, descubrir las ventajas y desventajas de los procesos mediante los cuales se realiza, basado en los datos encontrados, de manera que permita establecer los parámetros a tener en cuenta para el desarrollo del sistema web que se necesita implementar.
- **Analítico-sintético:** para realizar el análisis integral de la gestión de la información en temas sociopolíticos en la Dirección de Historia y Marxismo Leninismo en la Universidad de las Ciencias Informáticas, la evaluación de la actividad en general y realizar propuestas concretas para la mejora del proceso mediante el diseño del sistema web a desarrollar.
- **Modelación:** Para el diseño de la aplicación informática que, como aporte, se ofrece en esta investigación.

Empíricos:

- **Observación:** para obtener el conocimiento real del objeto de investigación, conocer cómo se desarrolla el procesamiento de la información en la Dirección de Historia y Marxismo Leninismo y las insuficiencias que presenta este proceso.
- **Entrevista:** A especialistas de la Dirección de Historia y Marxismo Leninismo para conocer en qué estado actual se encuentra el procesamiento de información en temas sociopolíticos y de qué manera se gestionan.

El presente documento se ha estructurado de la siguiente manera:

Capítulo 1: “Fundamentos teórico-metodológicos sobre el objeto de estudio”

En este capítulo se define el marco teórico conceptual y se realiza un estudio del estado del arte de los sistemas similares para el tratamiento de datos. Además, se analizan las herramientas, lenguajes y metodologías a usar en el desarrollo del proceso de investigación

Capítulo 2: Diseño de la solución propuesta al problema científico

En este capítulo se brinda una propuesta de solución basándose en el proceso de gestión de la información. Se hace una descripción minuciosa del proceso y se da paso a modelar el mismo.

Capítulo 3: Validación de la solución propuesta

Este capítulo corresponde a las etapas previstas por la metodología de software ágil, definidas por: Exploración, Planificación y Diseño. Además, aquí se denotan los requerimientos del sistema mediante HU, la planificación temporal del proyecto y la estructura final.

CAPÍTULO I: FUNDAMENTOS TEÓRICO-METODOLÓGICOS SOBRE EL OBJETO DE ESTUDIO

Este capítulo está dedicado a analizar los sistemas similares que existen y son utilizados en el tratamiento de datos a fin de crear un software basado en buenas prácticas y tener conocimiento de los principales problemas que afectan dichos sistemas. Además, se comenta sobre las metodologías, tecnologías, lenguajes de programación y herramientas que se recomienda para el desarrollo del sistema, teniendo en cuenta las ventajas que se obtienen.

I.1 Conceptos asociados al dominio del problema

Ciencias Sociales

Las Ciencias Sociales son aquellas que estudian los aspectos sociales y culturales del comportamiento humano. Se interesan tanto por las sociedades en su conjunto como por las relaciones entre los individuos y los grupos que forman parte de esas sociedades.

Entre las disciplinas que integran las ciencias se encuentran la antropología, la sociología, la psicología, la economía, la geografía, las ciencias políticas y la historia. En comparación con muchas de las ciencias naturales, son, en su mayoría, ciencias relativamente jóvenes.

Según el enfoque con que aborden su objeto de estudio, las ciencias sociales pueden tender a las ciencias naturales y utilizar métodos semejantes a ellas o, por el contrario, orientarse hacia las humanidades y utilizar métodos interpretativos, como la crítica social o la interpretación simbólica. (Etecé, 2022)

Gestión de la información

Es la denominación convencional de un conjunto de procesos por los cuales se controla el ciclo de vida de la información, desde su obtención (por creación o captura), hasta su disposición final. Tales procesos también comprenden la extracción, combinación, depuración y distribución de la información a los interesados. El objetivo de la gestión de la

información es garantizar la integridad, disponibilidad y confidencialidad de la información. Se puede identificar como la disciplina que se encargaría de todo lo relacionado con la obtención de la información adecuada, en la forma correcta, para la persona indicada, al coste adecuado, en el momento oportuno, en el lugar apropiado y articulando todas estas operaciones para el desarrollo de una acción correcta. (Reyes, 2017)

Sistema de gestión de información

Un sistema de gestión de la información, es un conjunto de herramientas para registrar, almacenar y procesar datos de diversas fuentes y obtener información que impulse el crecimiento del negocio. Las empresas dependen de los sistemas de gestión de la información para realizar análisis en profundidad de sus operaciones y para tomar decisiones altamente informadas. La información es recopilada en todos los niveles de operación y desde la gerencia se agregan y difunden los datos necesarios para llevar a cabo las operaciones diarias de todos los que trabajan en la empresa. Muchas empresas se concentran en el uso de una plataforma de gestión documental con los objetivos comerciales para lograr una ventaja competitiva sobre otras empresas y disminuir el procesamiento manual de documentos. (Argontech, 2022)

I.2 Estudio de sistemas homólogos

- **Alfresco:** Alfresco es un sistema de gestión de contenidos de código abierto que permite a las organizaciones capturar, almacenar, buscar y colaborar en documentos de muchos tipos distintos. Se habla de gestión de contenidos porque no sólo ofrece a las empresas herramientas de gestión documental, también se ocupa de la gestión de contenido web, records management, trabajo colaborativo, flujos de trabajo, gestión de procesos empresariales, etc. un producto dirigido a grandes y medianas empresas, que puedan aprovechar todas las ventajas que ofrece. No obstante, presenta algunos inconvenientes para lograr un rendimiento aceptable es necesario tener un servidor de amplia capacidad, es de pago, aunque cuenta con una versión gratuita, aunque esta no presenta soporte técnico o garantía alguna.

- **XABAL EXCRIBA:** es un sistema de gestión documental que propicia la generación de documentos, su revisión, administración, distribución, custodia y disposición. Cumple con las normas internacionales para esta función y mantiene la seguridad de los documentos en formato electrónico. Incluye, además, módulos para la gestión de: acuerdos, reportes, notificaciones, registros de entrada y salida de documentos, y del cuadro de clasificación de la entidad. Presenta algunas funcionalidades como gestión de documentos, gestión de carpetas, control de accesos y permisos entre otras.
- **DOCUMENTUM:** Este software es una suite de productos de gestión documental que permite la comunicación entre diferentes repositorios. Entre sus funciones permite: escanear, digitalizar, indexar y visionar los documentos en cualquier momento hasta su almacenamiento o su eliminación. Permite la gestión de documentos, la búsqueda, clasificación de contenido entre otras. Es una herramienta bastante compleja necesita un alto grado de conocimiento del programa para para su total comprensión, es de pago.

En los sistemas estudiados se encontraron diferentes funcionalidades que pueden ser de utilidad en el desarrollo de un nuevo sistema de acuerdo a las necesidades planteadas en la presente investigación, como son: gestionar documentos, gestionar usuarios, no obstante presentan una serie de dificultades por las cuales no es posible utilizar dichos sistemas ya sea porque son de pago, porque son demasiado complicados para personas que no poseen un gran conocimiento de informática o sobre estos sistemas. Por lo antes mencionado se decide desarrollar un sistema que cumpla con las necesidades del cliente y el dominio del problema.

I.3 Metodologías, lenguajes, herramientas y tecnologías informáticas para el desarrollo del sistema.

Las características del equipo de desarrollo son un factor determinante en la elección de las tecnologías a utilizar. En este caso, se cuenta con un equipo de desarrollo conformado por una persona con poca experiencia en el trabajo y desarrollo de software, ya que está en pleno proceso de aprendizaje y desarrollo. Dado que es pequeño el equipo y cuenta con

poca experiencia, es fundamental utilizar tecnologías que nos permitan agilizar el proceso de desarrollo y maximizar la productividad. En este sentido, las metodologías ágiles como XP se adaptan perfectamente a dichas necesidades. Esta tecnología permitirá aprovechar al máximo las habilidades, conocimientos, así como, trabajar de manera más eficiente y efectiva. Además, las metodologías ágiles promueven la entrega continua de software funcional, lo cual facilitaría obtener retroalimentación temprana del cliente y realizar ajustes rápidos en base a sus necesidades y expectativas. Agregar que es muy probable que surjan cambios y ajustes en los requisitos del proyecto y utilizando este tipo de metodologías podría responderse de manera ágil y eficiente a estos cambios, asegurando que el software entregado cumpla con las expectativas del cliente.

Metodología de desarrollo

Programación Extrema

La programación extrema es una metodología ágil de gestión de proyectos que se centra en la velocidad y la simplicidad con ciclos de desarrollo cortos y con menos documentación. Al igual que otras metodologías ágiles, la programación extrema es un método de desarrollo de software dividido en sprints de trabajo. Los marcos ágiles siguen un proceso iterativo, en el que se completa y revisa el marco al final de cada sprint, refinándolo para adaptarlo a los requisitos cambiantes y alcanzar la eficiencia máxima. Al igual que otros métodos ágiles, el diseño de la programación extrema permite a los desarrolladores responder a las solicitudes de los clientes, adaptarse y realizar cambios en tiempo real. Sin embargo, la programación extrema es mucho más disciplinada; realiza revisiones de código frecuentes y pruebas unitarias para realizar cambios rápidamente. Se basa en 5 valores como son simplicidad, comunicación, comentarios, valentía y respeto. Cuenta con 5 reglas las cuales son planificación, gestión, diseño, codificación y prueba. (Raeburn, 2022)

El equipo que consta de una sola persona utilizará la metodología XP para trabajar por varias razones. La metodología XP promueve la interacción presencial y la retroalimentación continua, lo que facilita la comunicación fluida y la resolución rápida de problemas. Dicha

metodología mantiene una comunicación armoniosa entre el cliente y el equipo y así poder llegar a un producto informático que cumpla las peticiones y garantice el éxito del desarrollo del software.

Herramientas y Tecnologías

Gestor de Bases de Datos

PostgreSQL:

PostgreSQL es un sistema de gestión de bases de datos relacional de código abierto que ofrece una amplia gama de características y funcionalidades avanzadas. Su importancia radica en su capacidad para manejar grandes cantidades de datos de manera eficiente y confiable, lo que lo convierte en una opción preferida para aplicaciones empresariales y de misión crítica. Una de las características destacadas de PostgreSQL es su soporte para múltiples tipos de datos, incluyendo tipos geométricos y de texto completo, lo que lo hace adecuado para aplicaciones que requieren un alto nivel de flexibilidad en el manejo de información. Se puede agregar que PostgreSQL cuenta con una arquitectura robusta que garantiza la integridad y la consistencia de los datos, así como la capacidad de recuperación en caso de fallos. También ofrece un alto nivel de escalabilidad, lo que le permite manejar grandes volúmenes de información y adaptarse a las necesidades crecientes de una organización (PostgreSQL., s.f.).

Node JS:

Node.js es un entorno de tiempo de ejecución de JavaScript que se utiliza para crear aplicaciones escalables del lado del servidor y de red a través de servidores privados virtuales. Node.js puede ser combinado con una bases de datos relacionales como MySQL, PostgreSQL, fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como por ejemplo, servidores web. (Díaz, 2023)

Visual Studio Code:

Visual Studio Code es un editor de código fuente desarrollado por Microsoft. Ofrece funciones avanzadas como autocompletado inteligente, inspección de código, resaltado de errores en tiempo real y soluciones rápidas. Además, cuenta con herramientas de refactorización automática, navegación avanzada y depuración integrada. Visual Studio Code también incluye una terminal incorporada, integración con sistemas de control de versiones (VCS) y herramientas de base de datos. Además, permite el desarrollo remoto con intérpretes remotos y una terminal SSH integrada. (Flores, 2022)

Visual Paradigm

Visual Paradigm ha sido concebida para soportar el ciclo de vida completo del proceso de desarrollo del software a través de la representación de todo tipo de diagramas. Constituye una herramienta privada disponible en varias ediciones, cada una destinada a satisfacer diferentes necesidades: Enterprise, Professional, Community, Standard, Modeler y Personal. Existe una alternativa libre y gratuita de este software, la versión Visual Paradigm UML 6.4 Community Edition (Community Edition, ya que existe la Enterprise, Professional, etc). Fue diseñado para una amplia gama de usuarios interesados en la construcción de sistemas de software de forma fiable a través de la utilización de un enfoque Orientado a Objetos. (Pressman, Roger S. Ingeniería de Software, un enfoque práctico. 2002)

PgAdmin 4

PgAdmin 4 es una herramienta de administración gráfica para PostgreSQL. Diseñada para funcionar en diversas plataformas, PgAdmin 4 satisface las necesidades de usuarios con diferentes niveles de experiencia, desde la escritura de consultas SQL simples hasta la administración de bases de datos complejas. Su interfaz gráfica es compatible con todas las características de PostgreSQL, lo que facilita la gestión de la base de datos. Además, la aplicación incluye un editor de sintaxis SQL, un editor de código del lado del servidor, un agente para programación de tareas y soporte para el motor de replicación Slony-I (Niemeyer et al., 2020).

Lenguaje de programación:

Lenguaje marcado de hipertexto (HTML5):

Cuando se habla de HTML5 lo primero que debe saberse es que es la última versión de la tecnología HTML, cuyas siglas corresponden a “HyperText Markup Language”, que tiene el siguiente significado:

- **HyperText**, cuyo significado es hipertexto, que no es más que un texto que enlaza con otros contenidos, que pueden ser otro texto u otro archivo. Esto es la base del funcionamiento de la web tal y como la conocemos, que no es más que páginas y recursos interconectados.
- **Markup**, que significa marca o etiqueta, ya que todas las páginas web están construidas en base a etiquetas, desde las primeras versiones hasta las últimas etiquetas de HTML5. Un ejemplo de una etiqueta HTML es la que identifica a un párrafo, que se compone de la etiqueta, el contenido de la etiqueta y el cierre del párrafo: `<p>HOLA</p>`.
- **Lenguaje**, cuyo significado es lenguaje, porque HTML es un lenguaje, es decir, tiene sus normas, tiene su estructura y una serie de convenciones que nos sirven para definir tanto la estructura como el contenido de una web.

Se puede definir HTML5 como un estándar que sirve para definir la estructura y el contenido de una página Web. (Jimenez, 2019)

Hoja de estilo en cascada (CSS):

CSS son las siglas en inglés para «hojas de estilo en cascada» (Cascading Style Sheets). Básicamente, es un lenguaje que maneja el diseño y presentación de las páginas web, es decir, cómo lucen cuando un usuario las visita. Funciona junto con el lenguaje HTML que se encarga del contenido básico de los sitios.

Se les denomina hojas de estilo «en cascada» porque puedes tener varias y una de ellas con las propiedades heredadas (o «en cascada») de otras. (Santos, 2023)

JavaScript:

JavaScript es un lenguaje de programación de alto nivel, interpretado y orientado a objeto, utilizado principalmente para el desarrollo web, pensado para agregar potencial de interacción y dinamismo a las páginas web.

Es compatible con todos los navegadores modernos y se ejecuta del lado del cliente, lo que significa que se ejecuta en el navegador web del usuario final. Además de su uso en el desarrollo web, JavaScript también se utiliza en aplicaciones de servidor (con tecnologías como Node.js) y en el desarrollo de aplicaciones móviles y de escritorio.

JavaScript proporciona una amplia gama de funcionalidades, como manipulación del DOM (Document Object Model) para interactuar con elementos de una página web, manejo de eventos, comunicación con servidores a través de AJAX, creación de animaciones, validación de formularios y mucho más. Es un lenguaje flexible y dinámico que permite a los desarrolladores crear experiencias interactivas y ricas en contenido. (Coppola, 2023)

Conclusiones del capítulo

- El análisis de los principales conceptos asociados al dominio del problema, facilitó a la comprensión del presente trabajo.
- Los homólogos estudiados (nacional e internacional) no son completamente factibles, dado que no satisfacen las demandas del cliente
- Se realizó la selección de los lenguajes y herramientas informáticas, además de la metodología de software, lo que permitirá lograr el objetivo de este trabajo acorde a las necesidades.

CAPÍTULO II: DISEÑO DE LA SOLUCIÓN PROPUESTA AL PROBLEMA CIENTÍFICO

Este capítulo estará dedicado a la identificación de las principales funcionalidades de la solución propuesta. Además, en los epígrafes del presente capítulo, se expondrán los artefactos generados por la propuesta dada como metodología de desarrollo de software a emplear, siendo XP la seleccionada, en sus fases de Exploración, Planificación y Diseño.

II.1 Exploración

En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo. La fase de exploración toma de pocas semanas a pocos meses, dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología. (Raeburn, 2022)

II.1.2 Historias de Usuario

Las HU son la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas en las cuales el cliente o propietario describe brevemente las características que el sistema debe poseer. El tratamiento de las HU es muy dinámico y flexible. Cada HU es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en poco tiempo. En estas deben estar presente elementos tales como el rol específico del usuario a realizar la funcionalidad, el valor o beneficio de la misma para el cliente y los criterios de aceptación o condiciones de satisfacción de la propia HU.

Las historias de usuario deben ser escritas en conjunto con el propietario del software y los programadores del equipo de desarrollo. En caso de ser escritas por el equipo de desarrollo y no por el cliente directamente, deben ser realizadas en lenguaje natural para un mejor entendimiento del cliente. Además, se le debe proporcionar una entrega del producto con el

cúmulo de resultados de la historia de usuario.

Las historias de usuario deben ser confeccionadas de forma tal que para ser completadas el equipo de desarrollo dedicará de una a cinco semanas. A pesar de esto, se decide acortarlo a un tiempo de hasta 2 semanas, siendo este un proyecto de corta duración.

Con el fin de seleccionar que historia de usuario tiene mayor importancia, según las necesidades del cliente y poder realizar una planificación adecuada de la implementación, cada HU es clasificada por el propietario del producto según la prioridad que implique esta para el negocio en:

- **Alta:** HU que son fundamentales en el desarrollo del sistema.
- **Media:** HU que son funcionalidades necesarias, pero no imprescindibles.
- **Baja:** HU que constituyen funcionalidades que son de ayuda para el control de elementos asociados al equipo de desarrollo, a la estructura y no tienen nada que ver con el sistema en desarrollo.

También se analiza la dificultad y la posible existencia de errores durante el proceso de implementación de cada HU, por ello el equipo de desarrollo realiza la clasificación una a una según el riesgo encontrado en su desarrollo, tales como:

- **Alta:** cuando en la implementación de la historia de usuario se considera la posible existencia de errores que lleven a la deficiencia del código.
- **Media:** es la situación en la cual pueden suceder errores en la implementación de la HU que puedan retrasar la entrega de la versión.
- **Baja:** cuando pueden aparecer errores que serán manejados con relativa facilidad sin que traigan daños para el desarrollo del proyecto.

Las HU representadas a través de tablas divididas por las siguientes secciones:

- **Número:** esta sección representa el número, incremental en el tiempo, de la HU que se describe.
- **Nombre de Historia de Usuario:** reconoce la HU que se describe entre el cliente y los desarrolladores.
- **Usuario:** rol de usuario que realiza la funcionalidad.
- **Prioridad de negocios:** se le otorga a la HU una prioridad (Alta, Media, Baja) de acuerdo a las necesidades en cuanto a desarrollo.
- **Riesgo en Desarrollo:** se le otorga a la HU una medida de (Alto, Medio, Bajo) la ocurrencia de errores en el proceso de desarrollo.
- **Iteración asignada:** número de la iteración donde se desea desarrollar la HU.
- **Puntos estimados:** es el tiempo estimado en semanas ideales (40 horas) que se demorará el desarrollo de la HU.
- **Descripción:** descripción breve de la HU.
- **Observaciones:** advertencia o alerta del sistema.(López 2018)

En vista del análisis realizado al sitio web quedaron totalmente claros todos los aspectos relacionados al mismo por lo que no será necesario realizar actualizaciones ni adicionar nuevas HU en torno a próximas iteraciones, a continuación, se exponen las historias de usuarios:

Tabla 1 HU 1 Autenticar Usuario

Número: 1	Nombre: Autenticar Usuario
Prioridad: Alta	Iteración asignada: 1
Riesgo en desarrollo: Medio	Tiempo estimado: 0.5
Usuario: Administrador	

Programador: Jose David Avila Batista

Descripción: En la historia de usuario "Autenticar", el sistema verifica la validez de las credenciales ingresadas y permite el acceso solo si son correctas. Si las credenciales son incorrectas, el sistema muestra un mensaje de error adecuado: "Las credenciales son incorrectas". Una vez autenticado, el usuario tiene acceso a las funcionalidades y características correspondientes a su rol o perfil de usuario.

Observaciones: Ninguna

Ver en anexos las Historias de Usuarios restantes. (Anexo 3).

II.2 Planificación

En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días. (Raeburn, 2022)

II.2.1 Estimación del esfuerzo

Con el propósito de saber cuánto le tomará al equipo de desarrollo elaborar las historias de usuario en su totalidad y comprender cuántos puntos debe desarrollar, se realiza una estimación de esfuerzos por cada HU. Se establecieron 4 iteraciones.

Tabla 2 Estimación de esfuerzo

Iteración	Historias de usuarios		Puntos estimados(semanas)	
1	1	Autenticar Usuario	0.5	
	2	Gestionar Usuario	0.5	
	3	Gestionar Comité Científico	1	
	4	Gestionar Consejo Editorial	1	
2	5	Gestionar Tema Político	0.5	
	6	Gestionar Conferencias	0.5	
	7	Gestionar Seminario	1	
	8	Gestionar Cursos	1	
3	9	Insertar link a vínculos de interés	0.5	

	10	Insertar contenido inherente al portal	0.5	
	11	Buscar Usuarios	0.2	
	12	Buscar Tema político	0.2	
4	13	Buscar Conferencia	0.2	
	14	Buscar Seminario	0.2	
	15	Buscar Curso	0.2	
Total			8	

II.2.2 Plan de iteraciones

Luego de ser identificadas y descritas las HU y estimar el esfuerzo dedicado a la realización de cada una de estas, en la metodología seleccionada (XP) se da paso a generar el plan de iteraciones. Este plan muestra cuáles son las HU que serán implementadas en cada iteración para perfeccionar el desempeño del equipo de desarrollo. Al finalizar las iteraciones el sistema quedará listo para pasar a manos de producción.

Posterior a esto se da paso a planificar la etapa de implementación. En esta etapa las HU se desarrollarán y se aprobarán en un ciclo iterativo. Para un mayor crecimiento y desempeño en la organización del equipo de desarrollo se establece un total de 4 iteraciones.

Iteración 1

En esta iteración se implementarán las HU con un nivel de prioridad Alta, además, el cliente tendrá la posibilidad de comprobar algunas de las funcionalidades descritas por sí mismo, mediante la primera versión que le proveerá el equipo de desarrollo. Dentro de estas funcionalidades podrá encontrar: Autenticar Usuario, Gestionar Usuario, Gestionar Comité Científico, Gestionar Consejo Editorial, descritas en las historias de usuario 1, 2,3, 4.

Iteración 2

En esta iteración se implementarán las HU con un nivel de prioridad Alta. Se rectificarán los errores que haya sido encontrado en la anterior iteración. En esta iteración se implementarán las HU 5,6,7,8. Las cuales corresponden a Gestionar Tema Político, Gestionar Conferencias, Gestionar Seminario, Gestionar Cursos,

Iteración 3

En esta iteración se realizarán las HU con un nivel de prioridad Alta y Media, las cuales estarán relacionadas con la etapa intermedia del desarrollo del sistema, con funcionalidades igual de importantes y se rectificarán los errores que haya sido encontrados en la anterior iteración. En esta iteración se implementarán las HU 9,10,11,12. Las cuales corresponden a, Insertar link a vínculos de interés, Insertar contenido inherente al portal, Buscar Usuarios, Buscar Tema político.

Iteración 4

En esta iteración se implementarán las HU con un nivel de prioridad Media y Baja, completando el proceso de desarrollo del sistema. En esta iteración se implementarán las HU 13,14,15, que corresponden a Buscar Conferencia, Buscar Seminario, Buscar Curso. Completando la primera versión del sistema para la gestión de la información en temas sociopolíticos para la Dirección de Historia y Marxismo Leninismo de la UCI. Una vez finalizada el cliente podrá realizar una revisión total de las funcionalidades implementadas y a su vez probar el sistema.

Tabla 3 Plan de iteraciones / cantidad de semanas

Iteración	Historias de usuarios		Cantidad de semanas
1	1	Autenticar Usuario	3
	2	Gestionar Usuario	
	3	Gestionar Comité Científico	
	4	Gestionar Consejo Editorial	
2	5	Gestionar Tema Político	3
	6	Gestionar Conferencias	
	7	Gestionar Seminario	
	8	Gestionar Cursos	
3	9	Insertar link a vínculos de interés	1.4
	10	Insertar contenido inherente al portal	

	11	Buscar Usuarios	
	12	Buscar Tema político	
4	13	Buscar Conferencia	0.6
	14	Buscar Seminario	
	15	Buscar curso	
Total			8

II.2.3 Plan de entrega

Con el fin de mantener al cliente centrado en mejorar el producto, y a su vez hacer que sea el propio cliente quien realice las pruebas pertinentes y para ello se realizan varias entregas programadas. A continuación, se presenta una tabla que refleja lo antes mencionado, se especifica la fecha en la cual se realiza el encuentro entre el equipo de desarrollo y el cliente o propietarios del software, para así mostrarles las versiones del producto.

Tabla 4 Plan de entrega.

Iteración	Fecha de entrega
Iteración 1	15 de septiembre de 2023
Iteración 2	03 de octubre de 2023
Iteración 3	19 de octubre de 2023
Iteración 4	31 de octubre de 2023

II.3 Diseño

En XP solo se diseñan aquellas historias de usuario que el cliente ha seleccionado para la iteración actual por dos motivos: por un lado, se considera que no es posible tener un diseño completo del sistema y sin errores desde el principio. El segundo motivo es que, dada la naturaleza cambiante del proyecto, el hacer un diseño muy extenso en las fases iniciales del proyecto para luego modificarlo, se considera un desperdicio de tiempo. Es importante resaltar que esta tarea es permanente durante la vida del proyecto partiendo de un diseño inicial que va siendo corregido y mejorado en el transcurso del proyecto. XP construye un

proceso de diseño evolutivo que se basa en refactorizar un sistema simple en cada iteración. Todo el diseño se centra en la iteración actual y no se hace nada anticipadamente para necesidades futuras. El resultado es un proceso de diseño disciplinado, que combina la disciplina con adaptabilidad de una manera que la hace una de las más desarrolladas entre todas las metodologías ágiles. (Raeburn, 2022)

II.3.1 Patrón arquitectónico

Los patrones arquitectónicos son la representación de las buenas prácticas y estructuras de diseño probadas, de modo que puedan reutilizarse. Proporcionan subsistemas predefinidos junto con directrices para su interconexión y disposición, expresando de esta forma una estructura general para el sistema. Esto permite generar componentes y subsistemas de manera guiada. El objetivo es reutilizar las experiencias y conocimientos arquitectónicos que han dado buenos resultados en el pasado (Lopez, 2022)

Para el desarrollo de este proyecto se desea utilizar como patrón arquitectónico el siguiente a explicar:

Modelo Vista Controlador

El patrón de arquitectura MVC (Modelo-Vista-Controlador) divide los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para lograr esto, MVC propone la construcción de tres componentes distintos: el modelo, la vista y el controlador. El modelo se encarga de representar la información, mientras que la vista se ocupa de la interacción con el usuario. El controlador actúa como el intermediario entre el modelo y la vista. Este enfoque de diseño se basa en la reutilización de código y la separación de conceptos, lo cual busca simplificar el desarrollo y el mantenimiento de aplicaciones (Patrones de diseño en la arquitectura de software: MVC, 2018).

Modelo: Es la representación de los datos con los cuales el sistema interactúa, por lo tanto, administra todos los accesos a esa información, tanto las consultas como las actualizaciones, implementando además los permisos de acceso que se hayan descrito en las especificaciones.

Vista Presenta el 'modelo' (información y lógica de negocio) en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho 'modelo' la información que debe representar como salida. Esto implica que el modelo debe ser capaz de proporcionar los datos necesarios para mostrar al usuario de manera clara y comprensible, ya sea a través de elementos visuales, tablas, gráficos u otros medios interactivos. Además, el modelo también debe ser capaz de procesar y manejar las acciones del usuario, como entradas de datos o selecciones, y aplicar la lógica de negocio correspondiente para generar los resultados esperados.

Controlador: Responde a eventos e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información. (Sarmiento, 2019)

Ventajas de usar el MVC

1. Se puede tener diferentes vistas para un mismo modelo.
2. Facilita crear nuevas visitas sin que se modifique el modelo subyacente.
3. Contiene un mecanismo de configuración más tratable a componentes complejos que el basado en eventos (El modelo puede representarse como un estado de interacción con una forma estructurada).

La conectividad del Modelo y sus vistas es dinámica, ya que se produce en tiempo de ejecución, no de implementación (Sarmiento, 2019)

II.3.2 Patrón de diseño

Un patrón de diseño se define como una regla de tres partes que establece una relación entre un contexto específico, un problema y una solución. En el diseño de software, el contexto proporciona información sobre el entorno en el que se encuentra el problema y cuál sería la solución adecuada en ese entorno. Un conjunto de requisitos, que incluye limitaciones y restricciones, actúa como un sistema de fuerzas que influye en la forma en que se interpreta el problema en este contexto y en cómo se puede aplicar de manera efectiva la

solución. Esta perspectiva ayuda a comprender la importancia de considerar el contexto y los requerimientos al aplicar los patrones de diseño en el desarrollo de software. (Canelo, 2020)

Los patrones de diseño son soluciones probadas y documentadas para problemas recurrentes en el diseño de software. Proporcionan un enfoque estructurado y reutilizable para abordar desafíos comunes en el desarrollo de sistemas. Los patrones de diseño permiten a los desarrolladores aprovechar las mejores prácticas y la experiencia acumulada en el diseño de software, lo que resulta en sistemas más eficientes y mantenibles. Estos patrones se basan en la idea de capturar soluciones exitosas a problemas comunes y documentarlas en un formato comprensible. Al utilizar patrones de diseño, los desarrolladores pueden evitar repetir errores y aplicar soluciones probadas en situaciones similares. Esto promueve la reutilización del diseño y acelera el proceso de desarrollo. Además, los patrones de diseño fomentan el modularidad y la flexibilidad en el diseño de software. Al dividir el sistema en componentes interconectados y asignar responsabilidades claras a cada uno, los patrones de diseño facilitan la comprensión y el mantenimiento del código, así como la adaptación a cambios futuros. (Marini, 2017)

El sistema que se desea desarrollar cuenta con una arquitectura basada en 3 capas, usando a su vez patrones arquitectónicos y de diseño, para ello se desean utilizar los patrones GRASP y GOF.

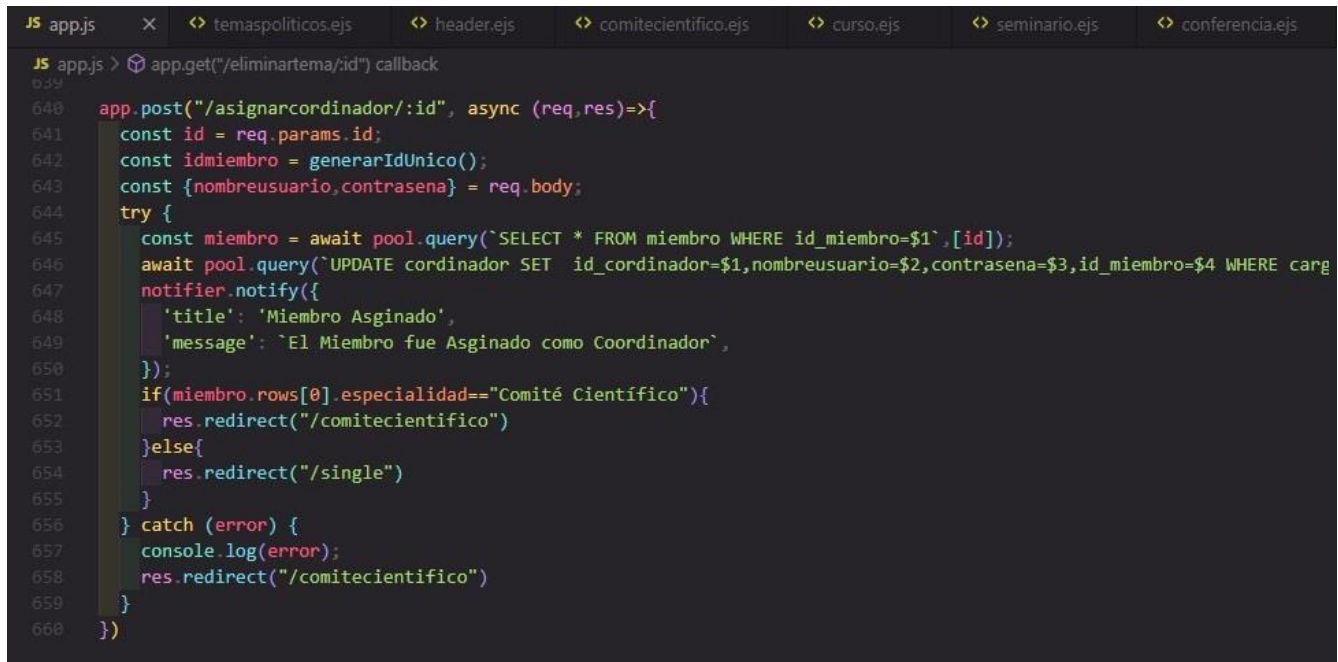
Patrones GRASP

Los patrones GRASP son patrones generales de software para asignación de responsabilidades y tienen una serie de buenas prácticas de aplicación recomendable en el diseño de software. Es importante destacar que estos patrones están estrechamente relacionados entre sí, pues en la práctica, el nivel de acoplamiento no se puede considerar de manera aislada a otros principios como el Experto o Alta Cohesión.

De los patrones *GRASP* a utilizar se tiene:

Experto: es un patrón que se usa más que cualquier otro al asignar responsabilidades; es un principio básico que suele utilizarse en el diseño orientado a objetos. Da origen a diseños

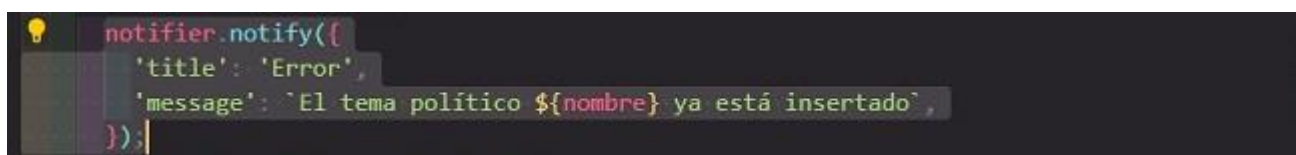
donde el objeto de software realiza las operaciones que normalmente se aplican a la cosa real que representa, por lo que ofrece una analogía con el mundo real.



```
JS app.js x <> temaspoliticos.ejs <> header.ejs <> comitecientifico.ejs <> curso.ejs <> seminario.ejs <> conferencia.ejs
JS app.js > app.get("/eliminartema/:id") callback
639
640 app.post("/asignarcordinador/:id", async (req,res)=>{
641   const id = req.params.id;
642   const idmiembro = generarIdUnico();
643   const {nombreusuario,contrasena} = req.body;
644   try {
645     const miembro = await pool.query(`SELECT * FROM miembro WHERE id_miembro=$1`,`[id]`);
646     await pool.query(`UPDATE cordinador SET id_cordinador=$1,nombreusuario=$2,contrasena=$3,id_miembro=$4 WHERE carg
647     notifier.notify({
648       'title': 'Miembro Asginado',
649       'message': `El Miembro fue Asginado como Coordinador`,
650     });
651     if(miembro.rows[0].especialidad=="Comité Científico"){
652       res.redirect("/comitecientifico")
653     }else{
654       res.redirect("/single")
655     }
656   } catch (error) {
657     console.log(error);
658     res.redirect("/comitecientifico")
659   }
660 })
```

Ilustración 1 Ejemplo de cómo se utiliza el patrón grasp experto

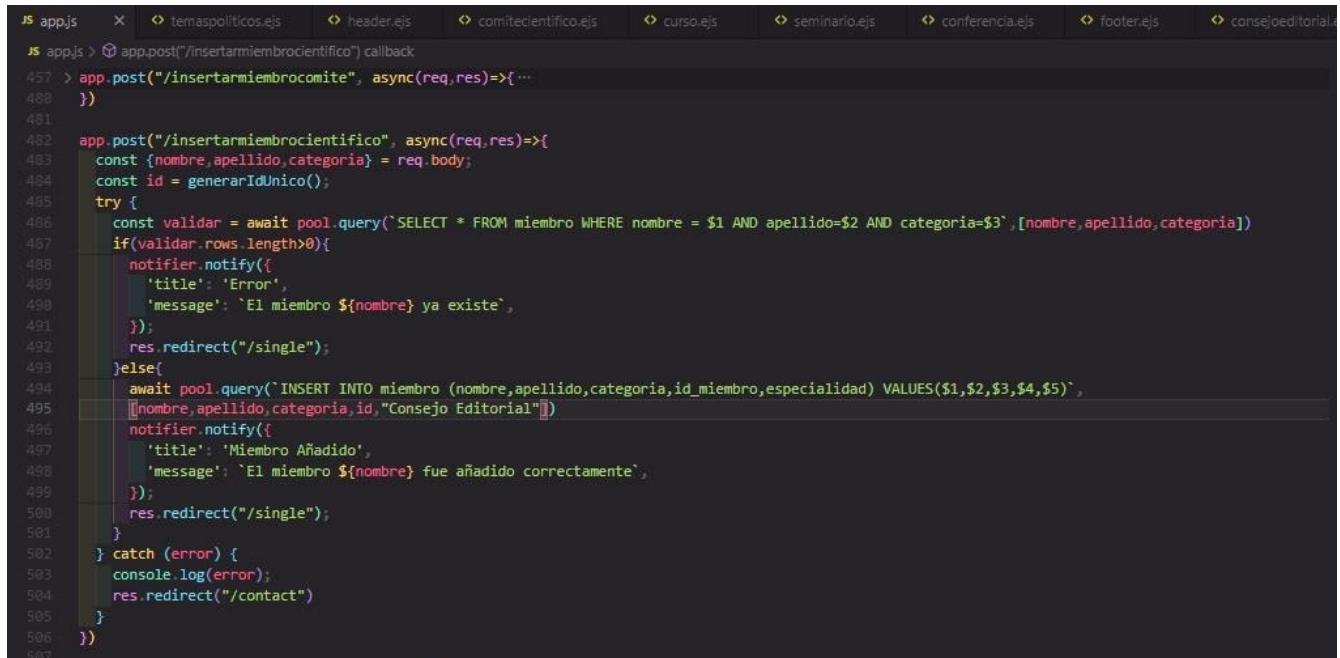
Controlador: La mayor parte de los Sistemas reciben eventos de entrada externa, los cuales generalmente incluyen una interfaz gráfica para el usuario operado por una persona. Otros medios de entrada son los mensajes externos o las señales procedentes de sensores como sucede en los sistemas de control de procesos. En todos los casos, si se recurre a un diseño orientado a objetos, hay que elegir los controladores que manejen esos eventos de entrada. Este patrón ofrece una guía para tomar decisiones apropiadas que generalmente se aceptan. La misma clase controlador debería utilizarse con todos los eventos sistémicos de un caso de uso, de modo que se pueda conservar la información referente al estado del caso.



```
💡
notifier.notify({
  'title': 'Error',
  'message': `El tema político ${nombre} ya está insertado`,
});
```

Ilustración 2 Ejemplo de cómo se utiliza el patrón grasp controlador

Creador: El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos. El propósito fundamental de este patrón es encontrar un creador que se debe conectar con el objeto producido en cualquier evento. (Marini, 2017)



```
JS app.js x temaspoliticos.ejs header.ejs comitecientifico.ejs curso.ejs seminario.ejs conferencia.ejs footer.ejs consejoeditorial.ejs
JS app.js > app.post("/insertarmiembrocientifico") callback
457 > app.post("/insertarmiembrocomite", async(req,res)=>{...
458 })
459
460
461
462 app.post("/insertarmiembrocientifico", async(req,res)=>{
463   const {nombre,apellido,categoria} = req.body;
464   const id = generarIdUnico();
465   try {
466     const validar = await pool.query(`SELECT * FROM miembro WHERE nombre = $1 AND apellido=$2 AND categoria=$3`,[nombre,apellido,categoria])
467     if(validar.rows.length>0){
468       notifier.notify({
469         'title': 'Error',
470         'message': `El miembro ${nombre} ya existe`,
471       });
472       res.redirect("/single");
473     }else{
474       await pool.query(`INSERT INTO miembro (nombre,apellido,categoria,id_miembro,especialidad) VALUES($1,$2,$3,$4,$5)`,
475         [nombre,apellido,categoria,id,"Consejo Editorial"])
476       notifier.notify({
477         'title': 'Miembro Añadido',
478         'message': `El miembro ${nombre} fue añadido correctamente`,
479       });
480       res.redirect("/single");
481     }
482   } catch (error) {
483     console.log(error);
484     res.redirect("/contact")
485   }
486 })
487
```

Ilustración 3 Ejemplo de cómo se utiliza el patrón grasp creador

Patrones GoF

Los patrones GoF fueron introducido popularmente a la industria del software con el libro Design Patterns: Elements of reusable object oriented programming escrito por Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides en 1995. Estos introducen estructuras de clases e interfaces que solucionan un problema general, pensado para que las aplicaciones robustas sean escalables y flexibles, además rápidas de desarrollar y diseñar.

El empleado en este trabajo de diploma es:

Solitario: está diseñado para restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto. Su intención consiste en garantizar que una clase sólo tenga una instancia y proporcionar un punto de acceso global a ella. No se encarga de la creación de objetos en sí, sino que se enfoca en la restricción en la creación de un objeto. (Marini, 2017)

```

JS app.js
641 app.post("/asignarcordinador/:id", async (req,res)=>{
642   const id = req.params.id;
643   const idmiembro = generarIdUnico();
644   const {nombreusuario,contrasena} = req.body;
645   try {
646     const miembro = await pool.query(`SELECT * FROM miembro WHERE id_miembro=$1`,`[id]`);
647     await pool.query(`UPDATE cordinador SET id_cordinador=$1,nombreusuario=$2,contrasena=$3,id_miembro=$4
648     WHERE cargo=$5`,`[idmiembro,nombreusuario,contrasena,miembro.rows[0].id_miembro,miembro.rows[0].especialidad]`);
649     notifier.notify({
650       'title': 'Miembro Asginado',
651       'message': `El Miembro fue Asginado como Coordinador`,
652     });

```

Ilustración 4 Ejemplo de cómo se utiliza el patrón GoF solitario

II.3.3 Tarjetas CRC

La metodología XP usa la técnica de las tarjetas CRC como una extensión informal de UML como representación del sistema.

Una tarjeta CRC representa una clase en la programación orientada a objetos. Son utilizadas por la metodología XP para diseñar el sistema uniendo todas las funcionalidades. Además, ayudan a definir actividades mientras se está diseñando el sistema.

La elaboración de las tarjetas CRC es una técnica que está orientada a establecer un planteamiento inicial de los objetos que participan en el sistema directamente. El producto generado por el uso de esta técnica, es un conjunto de tarjetas asociadas a las clases identificadas para el sistema, en las que se especifican las responsabilidades y las colaboraciones con otras clases.

Las tarjetas CRC constan de las siguientes partes:

- Nombre de la clase.
- Responsabilidades: son sentencias de alto nivel acerca de las acciones que realiza un objeto como también del conocimiento que mantiene y provee.
- Colaboradores: son aquellas clases con las cuales se van a trabajar.

Crear tarjetas CRC tiene como ventajas:

- Permiten contribuir al diseño de todo el equipo del proyecto.
- Permiten reducir el modo de pensar procedural y apreciar la tecnología de objetos.
- Son simples, fáciles de comprender y flexibles de realizarlas.
- Se visualiza en forma resumida todo el funcionamiento de una clase.

A continuación, se presentan las tarjetas CRC elaboradas en función del sistema a desarrollar:

Tabla 5 Tarjeta CRC 1 Comité Científico

Tarjeta CRC	
Clase: Comité Científico	
Responsabilidad	Colaboración
Comité_CientíficoList	Comité_Científico_list Comité_Científico
AddComité_Científico	Comité_Científico_add Comité_CientíficoForm
Comité_CientíficoUpdate	Comité_Científico_update Comité_Científico
Comité_CientíficoDelete	Comité_Científico_delete Comité_Científico

Tabla 6 Tarjeta CRC 2 Consejo Editorial

Tarjeta CRC	
Clase: Consejo Editorial	
Responsabilidad	Colaboración
Consejo_EditorialList	Consejo_Editorial_list Consejo_Editorial
AddConsejo_Editorial	Consejo_Editorial_add

	Consejo_EditorialForm
Consejo_EditorialUpdate	Consejo_Editorial_update Consejo_Editorial
Consejo_EditorialDelete	Consejo_Editorial_delete Consejo_Editorial

Tabla 7 Tarjeta CRC 3 Tema Político

Tarjeta CRC	
Clase: Tema Político	
Responsabilidad	Colaboración
Temas_PolíticosList	Temas_Políticos_list Temas_Políticos
AddTemas_Políticos	Temas_Políticos_add Temas_PolíticosForm
Temas_PolíticosDelete	Temas_Políticos_delete Temas_Políticos

Tabla 8 Tarjeta CRC 5 Conferencia

Tarjeta CRC	
Clase: Conferencia	
Responsabilidad	Colaboración
ConferenciaList	Conferencia_list Conferencia
AddConferencia	Conferencia_add ConferenciaForm
ConferenciaUpdate	Conferencia_update Conferencia
ConferenciaDelete	Conferencia_delete Conferencia

Tabla 9 Tarjeta CRC 6 Seminario

Tarjeta CRC	
Clase: Seminario	
Responsabilidad	Colaboración
SeminarioList	Seminario_list Seminario
AddSeminario	Seminario_add SeminarioForm
SeminarioUpdate	Seminario_update Seminario
SeminarioDelete	Seminario_delete Seminario

Tabla 10 Tarjeta CRC 7 Curso

Tarjeta CRC	
Clase: Curso	
Responsabilidad	Colaboración
CursoList	Curso_list Curso
AddCurso	Curso_add CursoForm
CursoUpdate	Curso_update Curso
CursoDelete	Curso_delete Curso

Tabla 11 Tarjeta CRC 10 link Vínculos de Interés

Tarjeta CRC	
Clase: link_vinculos_interes	
Responsabilidad	Colaboración
link_vinculos_interesList	link_vinculos_interes_list link_vinculos_interes

Addlink_vinculos_interes	link_vinculos_interes_add link_vinculos_interesForm
--------------------------	--

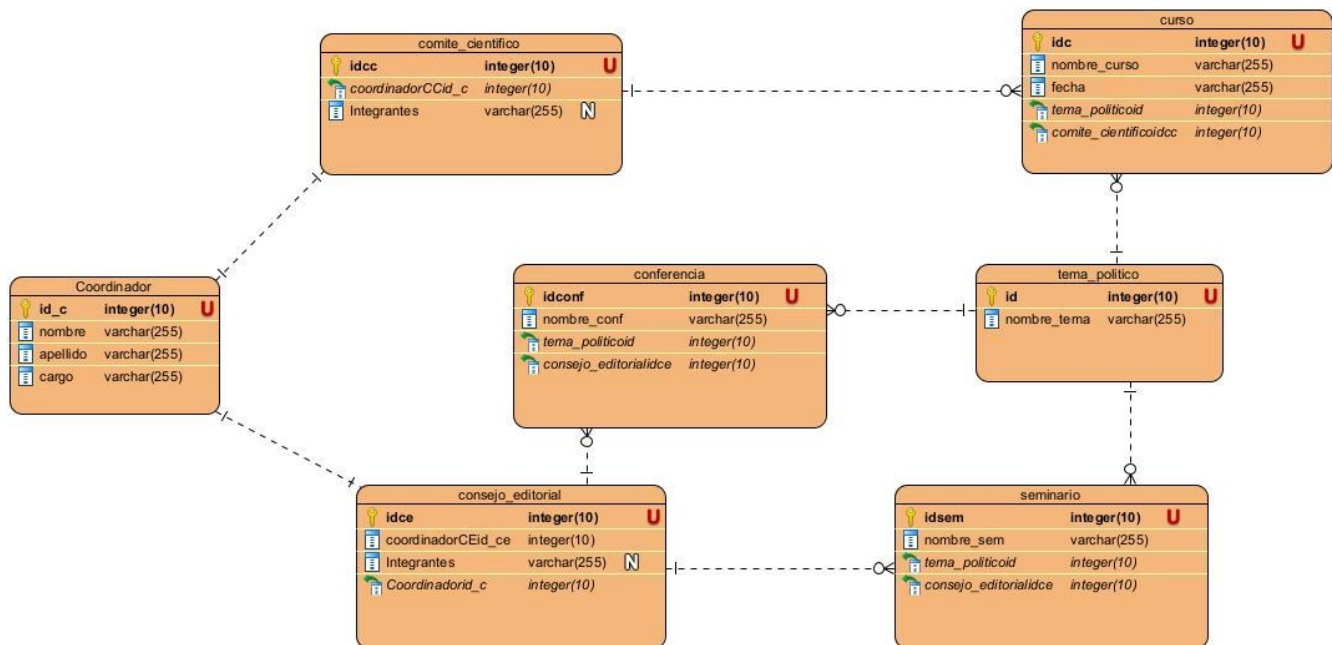
II.4 Modelo de datos

Luego de ser diseñadas las tarjetas CRC, y definidas las clases entidades, se diseña el modelo de datos que usará el sistema para almacenar la información.

El modelo de datos permite describir las principales clases que componen la capa modelo del sistema, el tipo de dato de los atributos que contendrá la base de datos y la forma en que se relacionan las entidades. Permite describir los elementos de la realidad que intervienen en un problema dado y la forma en que estos se relacionan.

A continuación, se muestra el diagrama entidad-relación que representa el modelo físico de datos diseñados para el sistema.

Ilustración 5: Modelo Físico de datos diseñados para el sistema



Conclusiones del capítulo

En este capítulo se describió las fases por la cuales transita la metodología XP.

- En la fase de Exploración, se obtuvo la generación de las HU acorde a las funcionalidades descritas por el cliente, lo cual permite un mejor entendimiento del proceso de desarrollo.
- En la fase de Planificación, se realizó la estimación de esfuerzos y el plan de entrega en base a las iteraciones definidas, lo cual permite establecer un margen de tiempo para la liberación de las versiones y mediante esto, el cliente pueda constatar el avance del proyecto que desea obtener de parte del equipo de desarrollo.
- En la fase de Diseño, se da paso a elegir un patrón arquitectónico, Se decidió emplear el patrón MVC, el cual permite garantizar una solución que tiene como premisa la reutilización del código y la solución a problemas similares dentro del desarrollo del software. Además, se conformó el artefacto principal de esta fase, que son las tarjetas CRC, lo cual posibilita definir las clases y las relaciones entre ellas a través de sus colaboradores y sus propias responsabilidades permitiendo un orden en el diseño.

Puede concluirse, por tanto, que la metodología XP es una alternativa eficaz para el desarrollo del presente proyecto, ya que se enfoca en la satisfacción del cliente, la comunicación constante entre el equipo de desarrollo y el cliente, y la entrega de software funcional en iteraciones cortas. Las fases de Exploración, Planificación y Diseño son fundamentales para el éxito del proyecto, ya que permiten una comprensión clara de las necesidades del cliente, una planificación adecuada de los esfuerzos y entregas, y un diseño ordenado y reutilizable.

CAPÍTULO III: VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

Tras culminada la etapa de diseño centrando las bases estructurales del sistema se da paso a este capítulo que persigue el objetivo de comenzar la fase de implementación y pruebas. En esta etapa se dividen las HU en tareas de ingeniería (TI) para realizar de una forma rápida la comprensión e implementación, se realizan las pruebas unitarias. También, se realiza la implementación de las TI y posterior a esto se ejecutan las pruebas de aceptación. Además, se tratarán aspectos en la codificación y se muestran las iteraciones realizadas para la implementación de las HU.

III.1 Implementación

La colaboración constante del cliente como parte integral del equipo de trabajo resulta fundamental en todas las etapas del enfoque de XP. Durante la fase de codificación e implementación de una HU, la presencia del cliente adquiere incluso mayor relevancia, debido a que son ellos quienes proponen y definen las HU, negocian los plazos para su implementación, brindan detalles precisos sobre su funcionalidad y verifican que la historia implementada cumpla con los requisitos específicos durante las pruebas. El proceso de codificación se rige por estándares establecidos, y los programadores deben seguirlos de manera rigurosa para garantizar que el código del sistema se presente de forma lógica. Esta coherencia en el código facilita su comprensión y mantenimiento.

III.2 Tareas de ingeniería

Las tareas de ingeniería (TI) son el artefacto que se encarga en la fase de implementación y pruebas de almacenar toda la información detallada de cada una de las HU. Una HU puede contener más de una TI dependiendo del nivel de complejidad de dicha HU. Cuando el equipo de desarrollo conforma las TI deja bien claro en sus campos a que HU pertenece y el programador responsable de implementarla, estableciendo a su vez fecha de inicio y de fin en correspondencia con la información que aporta la HU en cuestión. El tiempo estimado indica el tiempo real que utilizará el programador en realizar la funcionalidad, se especifica además el tipo de tarea a realizar (desarrollo, corrección o mejora) y el nombre de la misma, así como la descripción de dicha tarea; en este campo como su nombre lo indica se detalla todo lo que se va a realizar y en lo que se van a apoyar para darle solución a la funcionalidad

correspondiente.

Tabla 12 Tarea de Ingeniería 1 Autenticar Usuario

Número de la tarea: 1	Nombre de la tarea: Autenticar usuario	
Identificador de la HU: 1		Tipo de tarea: Desarrollo
Fecha de inicio: 01/09/2023		Fecha de fin: 04/09/2023
Tiempo estimado: 0.3	Programador: Jose David Avila Batista	
Descripción:		

Ver las tareas de ingeniería restantes en los anexos (Anexo 4)

III.3 Pruebas de software

Hoy en día el software es una parte importante en las actividades realizadas a través de las empresas y negocios que lo solicitan. Se debe tener en cuenta que los sistemas o aplicaciones son creadas, desarrolladas e implementadas por seres humanos y por ende tienden a equivocarse en cualquiera de las etapas del desarrollo del software. Al generarse esa equivocación se genera un error en el proceso trayendo consigo pérdida de tiempo, mala digitalización, errores en la codificación y que el resultado no sea el esperados, como también suceden errores por factores externos.

Para evitar que sucedan estos acontecimientos que entorpecen el trabajo y demoran la entrega del se hace necesario el uso de las pruebas de software pues estas ayudan a disminuir el riesgo en las aplicaciones y sistemas desarrollados y a su vez poder identificar los errores en una fase temprana antes de la ejecución. (Profesional, 2023)

Se define como **prueba de software** a la verificación dinámica del comportamiento esperado, usando un conjunto finito de casos de pruebas, seleccionados de manera adecuada.

III.3.1 Estrategia de prueba

En la elaboración de la estrategia de pruebas, se consideró la propuesta de la metodología XP de llevar a cabo un amplio conjunto de pruebas con el fin de identificar la mayor cantidad de problemas en las funcionalidades del sistema. Dicha estrategia comprende la ejecución de pruebas unitarias y pruebas de aceptación. Las pruebas unitarias se enfocan en verificar el comportamiento adecuado de las unidades individuales de código, mientras que las pruebas de aceptación están diseñadas para validar si el sistema cumple con los requisitos establecidos por los usuarios finales. La inclusión de ambas categorías de pruebas garantiza una cobertura integral y exhaustiva del software desarrollado, permitiendo detectar y corregir posibles fallos en las funcionalidades implementadas.

III.3.2 Pruebas Unitarias

Las pruebas unitarias representan una etapa fundamental e imprescindible dentro de la metodología XP. Estas pruebas son aplicadas y ejecutadas de manera constante en respuesta a cada modificación realizada en el sistema. Son diseñadas por los programadores con el propósito de verificar la correcta funcionalidad de todo el código desarrollado, con miras a su posterior publicación. Es importante destacar que tanto el sistema como el conjunto de pruebas deben ser almacenados conjuntamente con el código, permitiendo que otros desarrolladores puedan utilizarlos en caso de requerir correcciones, modificaciones o recodificaciones en partes del mismo.

Las pruebas unitarias fueron realizadas al concluir la implementación de cada una de las funcionalidades de las HU. Las mismas se aplicaron en dos iteraciones, mediante el uso del módulo '*Mocha*' que ofrece el lenguaje JavaScript, reduciendo así el tiempo y el esfuerzo en esta fase.

Luego de realizada la primera iteración se encontró con el siguiente resultado:

- En la HU_5 se encontraron dos no conformidades, tales como errores de código de respuestas y redireccionamientos.
- En la HU_8 se encontraron seis no conformidades tales como parámetros incorrectos y retorno de datos incorrectos.

- En la HU_10 se encontraron 4 no conformidades tales como parámetros incorrectos y retorno de datos incorrectos.

Después de ser aplicada la segunda iteración de las pruebas unitarias, no se detectaron no conformidades. **La Ilustración 5** muestra los resultados de las pruebas unitarias en la siguiente gráfica:

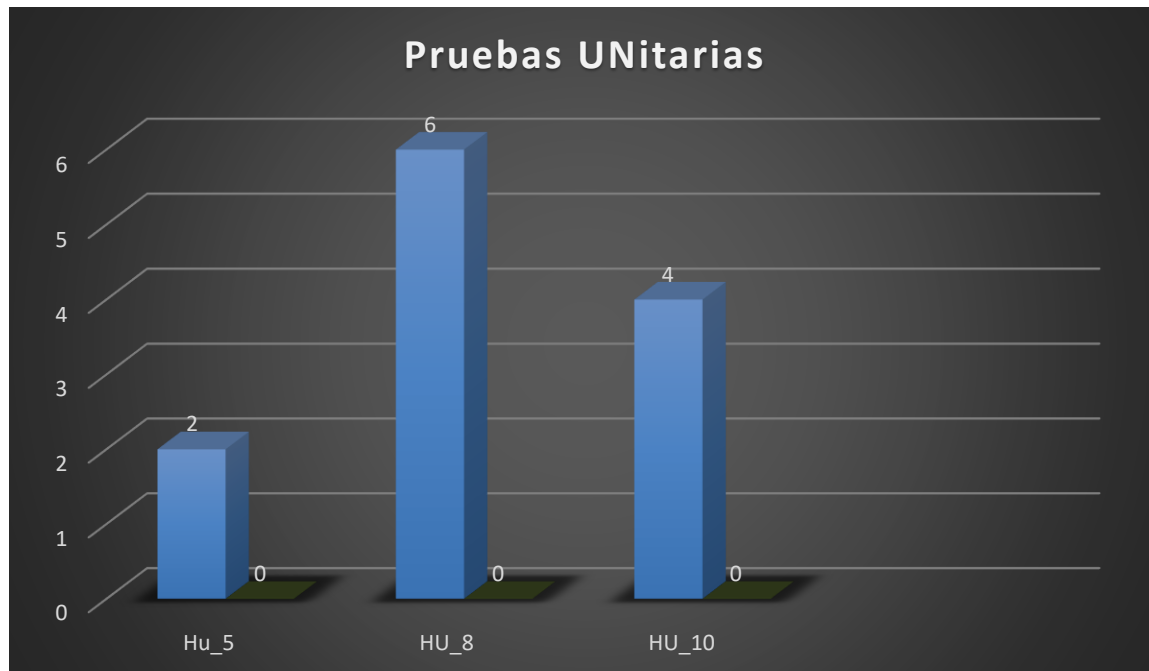


Ilustración 5 Pruebas Unitarias

A continuación se muestra una parte del código para la realización de las pruebas unitarias:

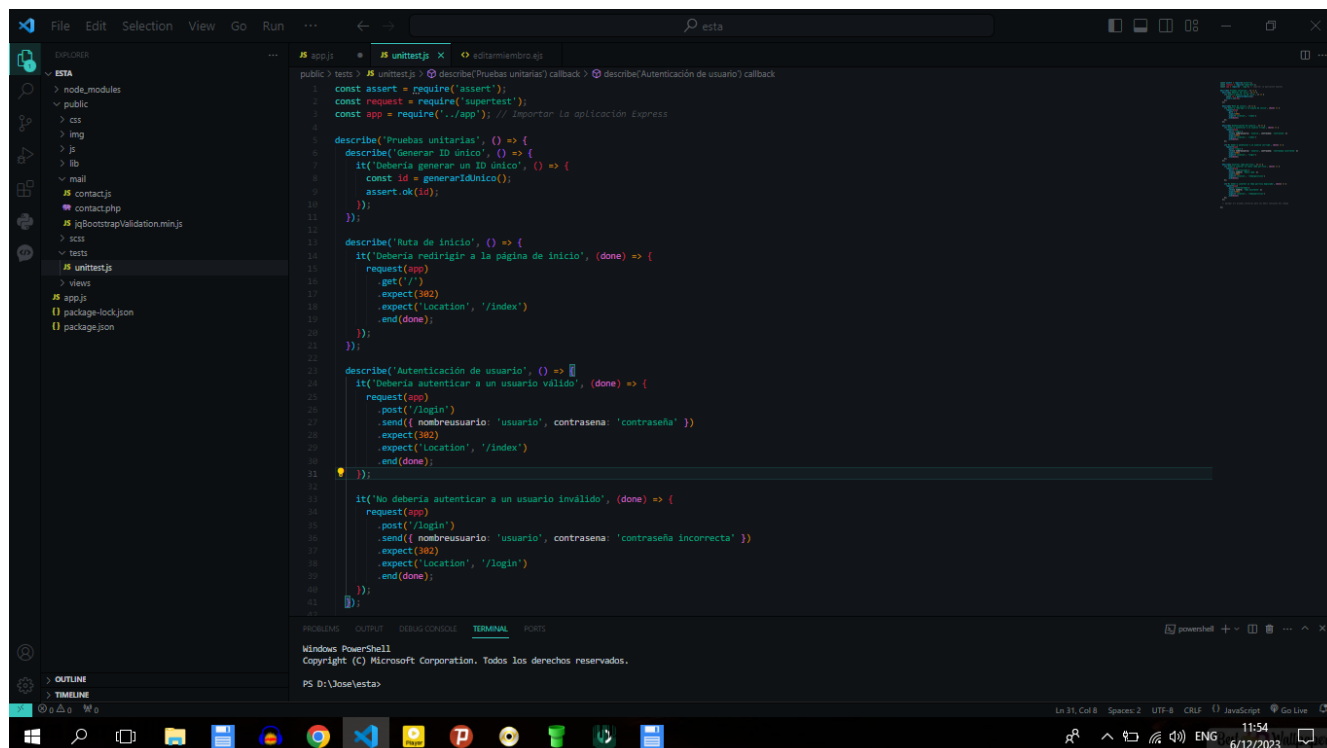


Ilustración 6 Ejemplo del código de las pruebas unitarias

III.3.3 Pruebas de Aceptación

creadas en base a las HU en cada ciclo de iteración del desarrollo. Es responsabilidad del cliente especificar uno o varios escenarios que permitan comprobar la correcta implementación de una HU. Estas pruebas de aceptación se consideran como "pruebas de caja negra", donde el cliente asume la responsabilidad de verificar la precisión de los resultados obtenidos. En caso de que se produzcan fallas en varias pruebas, el cliente debe indicar el orden de prioridad para su resolución. Es importante destacar que una HU no puede considerarse completa hasta que supere exitosamente todas las pruebas de aceptación. Dado que la responsabilidad es compartida por todo el equipo, es recomendable que los resultados de las pruebas de aceptación se publiquen, de manera que todos los miembros estén informados al respecto.

Para llevar a cabo las pruebas de aceptación de usuarios, se utiliza una herramienta manual basada en casos de prueba diseñados por el programador, con el objetivo de guiar al cliente en la realización de pruebas efectivas que validen la funcionalidad de manera adecuada. Es importante destacar que una HU puede contar con tantas pruebas de aceptación como sea

necesario para garantizar su correcto funcionamiento. El objetivo específico de estas pruebas es asegurar que los requisitos han sido cumplidos y que el sistema es aceptable en su conjunto.

Tabla 13 Caso de Prueba de Aceptación 1

Código: HU3_P1	Historia de Usuario: 3
Nombre: Insertar Comité Científico	
Descripción: Prueba para la funcionalidad de insertar un miembro del Comité Científico en el sistema.	
Condiciones de ejecución: ➤ El usuario debe estar previamente registrado como administrador. O coordinador del Comité Científico	
Pasos de ejecución: ➤ El usuario selecciona de la barra de navegación la opción '<i>Comité Científico</i>' ➤ El usuario presiona el botón '<i>Añadir Miembro</i>' ➤ El usuario llena todos los campos requeridos con información válida ➤ El usuario presiona el botón '<i>Guardar</i>'	
Resultados esperados: Se insertará un miembro del Comité Científico en la base de datos y se mostrará un mensaje que indica que se ha insertado satisfactoriamente.	
Evaluación de la prueba: Satisfactoria	

Ver los casos de prueba de aceptación restantes en los Anexo (Anexo 5).

III.3.4 Validación del Sistema

Las deficiencias mencionadas en la introducción de este estudio tienen un impacto directo en la gestión de la información de temas sociopolíticos. En este sentido, se presenta la solución informática propuesta como una forma de abordar y resolver estos problemas:

Tabla 53. Validación del sistema

Indicadores	Antes	Después
-------------	-------	---------

Disponibilidad	No existía	24h
Integridad	La información no era exacta y tenía probabilidades altas de alteración o pérdida.	La información es válida y segura.

Conclusiones del capítulo

El capítulo III se centró en la validación de la solución propuesta, a través de la implementación y las pruebas. Durante la implementación, se establecieron directrices para el código y se crearon tareas de ingeniería para cada Historia de Usuario. La colaboración del cliente fue fundamental en este proceso. Las pruebas unitarias y las pruebas de aceptación se llevaron a cabo para verificar el funcionamiento correcto del sistema. Las pruebas unitarias demostraron que el código funcionaba adecuadamente. Las pruebas de aceptación, basadas en casos de prueba diseñados por los programadores, validaron la implementación exitosa de las Historias de Usuario. La solución informática propuesta cumple con todos los requisitos para mejorar la gestión de la información en temas sociopolíticos.

CONCLUSIONES FINALES

Luego de exponer el problema que se desea resolver, los diferentes conceptos asociados a la investigación, los detalles de los elementos que conforman la propuesta de solución, implementación y pruebas correspondientes, se puede expresar que los objetivos han sido cumplidos con el mayor rigor.

1. Se logró establecer los fundamentos y referentes teórico-metodológicos para el desarrollo de un sistema web para la gestión de la información en temas sociopolíticos.
2. La selección de XP como metodología de desarrollo de software permitió guiar y documentar el ciclo de vida, permitiendo que se cumpliera con los estándares y estilos definidos por el cliente.
3. La utilización de las tecnologías y herramientas definidas permitió que la solución informática desarrollada gestione el proceso de gestión de la información en temas sociopolíticos contribuyendo a la disponibilidad e integridad de los datos.
4. Las pruebas realizadas permitieron verificar que el sistema para la gestión de la información en temas sociopolíticos posee un correcto funcionamiento.

RECOMENDACIONES

Se propone para futuras versiones tener en cuenta técnicas de inteligencia artificial para aplicar con la información que brinda el sistema web posibilitando comparación de información, palabras más empleadas y asociadas al tema sociopolítico, una sección para la revista de la Dirección de Historia y Marxismo Leninismo. Una sección para noticias referentes a Cuba.

REFERENCIAS BIBLIOGRÁFICAS

- *¿Qué Es Un Sistema Web?* (22 de septiembre de 2022). Obtenido de Crea System: <https://www.creasystem.net/posts/que-es-un-sistema-web>
- Canelo, M. M. (24 de junio de 2020). *Profile*. Obtenido de ¿Qué son los patrones de diseño de software?: <https://profile.es/blog/patrones-de-diseno-de-software/>
- Coppola, M. (10 de julio de 2023). *HubSpot*. Obtenido de Qué es JavaScript, para qué sirve y cómo funciona: <https://blog.hubspot.es/website/que-es-javascript>
- CreaSystem. (22 de Septiembre de 2022). *CreaSystem*. Obtenido de ¿Qué Es Un Sistema Web?: <https://www.creasystem.net/posts/que-es-un-sistema-web>
- Estévez, J. R. (2020). La ingeniería de requisitos en el desarrollo de aplicaciones informáticas. 12.
- Etecé, E. (24 de junio de 2022). *Conceptos*. Obtenido de Ciencias Sociales: <https://concepto.de/ciencias-sociales/>
- Flores, F. (22 de Julio de 2022). *Open Webinars*. Obtenido de Qué es Visual Studio Code y qué ventajas ofrece.
- Gomez, C. (8 de Agosto de 2020). *Diario de QA*. Obtenido de Niveles de prueba: <https://www.diariodeqa.com/post/niveles-de-prueba>
- Huambachano, J. F. (25 de septiembre de 2017). *¿Qué es Scrum?* Obtenido de Scrum.org.
- *InCuatro Informatica para empresas*. (30 de Julio de 2019). Obtenido de ¿Para qué sirve la informática?: <https://incuatro.com/para-que-sirve-la-informatica/>
- Jimenez, J. D. (20 de enero de 2019). *OpenWebinars*. Obtenido de Que es HTML5 : <https://openwebinars.net/blog/que-es-html5/>
- Lomelí, L. (2 de Diciembre de 2022). *Innevo Consulting*. Obtenido de Metodologías de Desarrollo de Software: ¿Cuál Elegir?: <https://blog.innevo.com/metodologias-desarrollo-software>
- Londoño, P. (3 de Abril de 2023). *HubSpot*. Obtenido de Qué es Python, para qué sirve y cómo se usa (+ recursos para aprender): <https://blog.hubspot.es/website/que-es-python>

- Lopez, J. (2022). *Dreams*. Obtenido de Patrones de arquitectura de software: <https://www.dreams.es/transformacion-digital/desarrolladores-paginas-web/patrones-de-arquitectura-de-software>
- Marini, F. (13 de noviembre de 2017). *Prezi*. Obtenido de GoF y Grasp: https://prezi.com/mymp91_tnj3g/gof-y-grasp/#:~:text=Experto%20en%20Informaci%C3%B3n-,Los%20GRASP%20no%20implementan%20las%20soluciones%2C%20m%C3%A1s%20bien%20nos%20llevar,forma%20en%20que%20debemos%20usarlos.
- Paez, N. (26 de mayo de 2020). *Blog de Nico Paez*. Obtenido de Definición de la estrategia de pruebas: <https://blog.nicopaez.com/2020/05/26/estrategia-de-pruebas/>
- Pressman. (s.f.). Técnicas de pruebas de software. En Pressman, *Ingeniería de software* (págs. 281-302).
- Profesional, S. d. (27 de Marzo de 2023). *Linkedin*. Obtenido de Pruebas de software: <https://www.linkedin.com/pulse/pruebas-de-software-servicios-de-informatica-profesion/?originalSubdomain=es>
- Pursell, S. (20 de Enero de 2023). *HubSpot*. Obtenido de Metodología Agile: qué es y cómo aplicarla a tu proyecto: <https://blog.hubspot.es/marketing/metodologia-agile>
- Quiroga, L. A. (2002). Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones. En L. A. Quiroga, *Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones* (págs. vol.10, n.5, pp.7-8). ISSN 1024-9435. Obtenido de Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones.
- Raeburn, A. (28 de Noviembre de 2022). *Asana*. Obtenido de a programación extrema (XP) produce resultados, pero ¿es la metodología adecuada para ti?
- Reyes, L. A. (16 de Abril de 2017). *Infomed Instituciones*. Obtenido de ¿Que es la Gestion de la informacion?: [https://instituciones.sld.cu/toximed/2017/04/16/que-es-gestion-de-la-informacion/#:~:text=Gesti%C3%B3n%20de%20la%20informaci%C3%B3n%20\(GI,\(su%20archivo%20o%20eliminaci%C3%B3n\)](https://instituciones.sld.cu/toximed/2017/04/16/que-es-gestion-de-la-informacion/#:~:text=Gesti%C3%B3n%20de%20la%20informaci%C3%B3n%20(GI,(su%20archivo%20o%20eliminaci%C3%B3n)).

- Rodríguez, C. (8 de noviembre de 2023). *Apliint*. Obtenido de Tipos de pruebas del software: <https://apliint.com/2021/07/27/tipos-de-pruebas-del-software/>
- Santander/Becas. (21 de Diciembre de 2020). *Santander/Becas*. Obtenido de Metodologías de desarrollo de software: ¿qué son?
- Santos, D. (25 de julio de 2023). *HubSpot*. Obtenido de Introducción al CSS: qué es, para qué sirve y otras 10 preguntas frecuentes: <https://blog.hubspot.es/website/que-es-css>
- Sarmiento, R. (20 de junio de 2019). *Prezi*. Obtenido de Modelo Vista Template: <https://prezi.com/nd9ydrb01kqv/modelo-vista-template/>
- System, C. (22 de setiembre de 2022). *¿Qué Es Un Sistema Web?* (Crea System) Obtenido de Crea System: <https://www.creasytem.net/posts/que-es-un-sistema-web#:~:text=El%20sistema%20web%20o%20tambien,pr%C3%A1ctica%20en%20el%20navegador%20web.>
- Universidades, S. (9 de Abril de 2021). *Python: qué es y por qué deberías aprender a utilizarlo*.

ANEXOS.

Anexo 1. Guía de observación para analizar cómo se gestiona el procesamiento de la información en la Dirección de Historia y Marxismo Leninismo

Observador: José David Ávila Batista.

Lugar: Dirección de Historia y Marxismo Leninismo de la Universidad de Ciencias Informáticas.

Objetivos: Conocer cómo se realiza el procesamiento de la información sobre temas sociopolíticos en la Dirección de Historia y Marxismo Leninismo y las insuficiencias que presenta este proceso.

Elementos a observar:

1. Características de las informaciones que se reciben y procesan.
2. Periodicidad con que se reciben las informaciones.
3. Formas en que se clasifican y organizan las informaciones recibidas.
4. De qué manera se emplean las mismas.
5. Tiempo que se emplea en la recopilación, organización y procesamiento de la información que se recibe.
6. Insuficiencias que presenta el proceso de recopilación, organización, tratamiento.

Anexo 2: Entrevista realizada a _____

Estimado compañero:

La presente entrevista tiene como objetivo conocer las características de la información sobre temas sociopolíticos que se procesa en la Dirección de Historia y Marxismo Leninismo, de qué manera se gestiona y el estado actual en que se encuentra el procesamiento de la misma. Su colaboración es de vital importancia para la presente investigación ya que contribuye con el desarrollo de una aplicación informática que mejore la gestión de la información en esa Dirección.

Cuestionario:

1. ¿Qué medios y sitios web usted emplea para su información?
2. ¿Cómo considera usted deba informarse para el desempeño de su actividad profesional?
3. ¿Consideraría pertinente la creación de un grupo de trabajo que monitoree y analice información sobre temas sociopolíticos? ¿Por qué?
4. ¿Conoce herramientas informáticas para el análisis de la información y/o de grandes volúmenes de datos?
5. ¿Qué actividades académicas a su juicio, serían pertinentes realizar en función de producir contenido para la superación de los profesores e investigadores de la Dirección de Historia y Marxismo Leninismo (DHML) en la Universidad de las Ciencias Informáticas?
6. ¿Cómo vincularía sus clases con el perfil profesional de los estudiantes de la Universidad apoyado en un proyecto de Observatorio para temas sociopolíticos?
7. ¿Desde su experiencia, qué aspectos teórico metodológicos deberían estar presentes en un Observatorio para temas sociopolíticos?

Anexo 3: Historias de usuarios.*Tabla 14 HU 2 Gestionar Usuario*

Número: 2	Nombre: Gestionar Usuario	
Prioridad: Alta		Iteración asignada: 1
Riesgo en desarrollo: Medio		Tiempo estimado: 0.3
Usuario: Administrador		
Programador: Jose David Avila Batista		
Descripción: El sistema permite al administrador gestionar los datos de cada uno de los usuarios, en correspondencia con su nombre de usuario y contraseña.		
Observaciones: Ninguna		

Tabla 15 HU 3 Gestiona Comité Científico

Número: 3	Nombre: Gestionar Comité Científico	
Prioridad: Alta		Iteración asignada: 1
Riesgo en desarrollo: Medio		Tiempo estimado: 1
Usuario: Administrador, Coordinador Comité científico		
Programador: Jose David Avila Batista		
Descripción: El sistema permite al administrador y al Coordinador Comité científico gestionar los datos de cada uno de los miembros del Comité Científico que constan de los siguientes atributos: id, nombres, apellidos, cargo.		
Observaciones: Ninguna		

Tabla 16 HU 4 Gestionar Consejo Editorial

Número: 4	Nombre: Gestionar Consejo Editorial	
Prioridad: Alta		Iteración asignada: 1

Riesgo en desarrollo: Medio	Tiempo estimado: 1
Usuario: Administrador, Coordinador Consejo editorial	
Programador: Jose David Avila Batista	
Descripción: El sistema permite al administrador y al Coordinador Consejo editorial gestionar los datos de cada uno de los integrantes del Consejo Editorial, que constan de los siguientes atributos: id, nombres, apellidos, cargo.	
Observaciones: Ninguna	

Tabla 17 HU 5 Gestionar Temas Políticos

Número: 5	Nombre: Gestionar Tema Político
Prioridad: Alta	Iteración asignada: 2
Riesgo en desarrollo: Medio	Tiempo estimado: 0.5
Usuario: Administrador	
Programador: Jose David Avila Batista	
Descripción: El sistema permite al administrador gestionar los datos de cada uno de los Temas Políticos que tiene como atributos: id, nombre_tema, categoría.	
Observaciones: Ninguna	

Tabla 18 HU 6 Gestionar Conferencia

Número: 6	Nombre: Gestionar Conferencias
Prioridad: Alta	Iteración asignada: 3
Riesgo en desarrollo: Medio	Tiempo estimado: 0.5
Usuario: Administrador y Profesores	
Programador: Jose David Avila Batista	
Descripción: El sistema permite al administrador planificar y gestionar los datos de las Conferencias. Cuentan con los siguientes atributos: id, nombre_conf, tema	
Observaciones: Ninguna	

Tabla 19 HU 7 Gestionar Seminarios

Número: 7	Nombre: Gestionar Seminario	
Prioridad: Alta		Iteración asignada: 3
Riesgo en desarrollo: Medio		Tiempo estimado: 1
Usuario: Administrador		
Programador: Jose David Avila Batista		
Descripción: El sistema permite al administrador planificar y gestionar los datos de los Seminarios. Cuenta con los siguientes atributos: id, nombre_sem, tema		
Observaciones: Ninguna		

Tabla 20 HU 8 Gestionar Cursos

Número: 8	Nombre: Gestionar Cursos	
Prioridad: Alta		Iteración asignada: 3
Riesgo en desarrollo: Medio		Tiempo estimado: 1
Usuario: Administrador		
Programador: Jose David Avila Batista		
Descripción: El sistema permite al administrador planificar y gestionar los datos de los Cursos. Cuenta con los siguientes atributos: id, nombre_curso, tema, fecha.		
Observaciones: Ninguna		

Tabla 21 HU 9 Insertar link a vínculos de interés

Número: 9	Nombre: Insertar link a vínculos de interés	
Prioridad: Alta		Iteración asignada: 3
Riesgo en desarrollo: Medio		Tiempo estimado: 0.2

Usuario: Administrador
Programador: Jose David Avila Batista
Descripción: El sistema permite subir las url vinculadas a sitios de interés para los usuarios
Observaciones: Ninguna

Tabla 22 HU 10 Insertar contenido inherente al portal

Número: 10		Nombre: Insertar contenido inherente al portal	
Prioridad: Alta		Iteración asignada: 3	
Riesgo en desarrollo: Medio		Tiempo estimado: 0.2	
Usuario: Administrador			
Programador: Jose David Avila Batista			
Descripción: El sistema permite subir contenido con la información referente al portal			
Observaciones: Ninguna			

Tabla 23 HU 11 Buscar Usuarios

Número: 11		Nombre: Buscar Usuario	
Prioridad: Alta		Iteración asignada: 4	
Riesgo en desarrollo: Medio		Tiempo estimado: 0.2	
Usuario: Administrador			

Programador: Jose David Avila Batista
Descripción: El sistema permite buscar un usuario entre los ya existentes en el mismo.
Observaciones: Ninguna

Tabla 24 HU 12 Buscar Tema Político

Número: 12		Nombre: Buscar Tema político	
Prioridad: Alta		Iteración asignada: 4	
Riesgo en desarrollo: Medio		Tiempo estimado: 0.2	
Usuario: Administrador			
Programador: Jose David Avila Batista			
Descripción: El sistema permite buscar un Tema político entre los ya existentes en el mismo.			
Observaciones: Ninguna			

Tabla 25 HU 13 Buscar Conferencia

Número: 13		Nombre: Buscar conferencia	
Prioridad: Alta		Iteración asignada: 4	
Riesgo en desarrollo: Medio		Tiempo estimado: 0.2	
Usuario: Administrador			

Programador: Jose David Avila Batista
Descripción: El sistema permite buscar una conferencia entre los ya existentes en el mismo.
Observaciones: Ninguna

Tabla 26 HU 14 Buscar Seminario

Número: 14	Nombre: Buscar Seminario
Prioridad: Alta	Iteración asignada: 4
Riesgo en desarrollo: Medio	Tiempo estimado: 0.2
Usuario: Administrador	
Programador: Jose David Avila Batista	
Descripción: El sistema permite buscar un seminario entre los ya existentes en el mismo.	
Observaciones: Ninguna	

Tabla 27 HU 15 Buscar Curso

Número: 15	Nombre: Buscar curso
Prioridad: Alta	Iteración asignada: 4
Riesgo en desarrollo: Medio	Tiempo estimado: 0.2
Usuario: Administrador	

Programador: Jose David Avila Batista
Descripción: El sistema permite buscar un curso entre los ya existentes en el mismo.
Observaciones: Ninguna

Anexo 4: Tareas de Ingeniería.*Tabla 28 Tarea de Ingeniería 2 Gestionar Usuario*

Número de la tarea: 2	Nombre de la tarea: Gestionar Usuario
Identificador de la HU: 2	Tipo de tarea: Desarrollo
Fecha de inicio: 05/09/2023	Fecha de fin: 07/09/2023
Tiempo estimado: 0.5	Programador: Jose David Avila Batista
Descripción:	

Tabla 29 Tarea de Ingeniería 3 Gestionar Comité Científico

Número de la tarea: 3	Nombre de la tarea: Gestionar Comité Científico
Identificador de la HU: 3	Tipo de tarea: Desarrollo
Fecha de inicio: 08/09/2023	Fecha de fin: 11/09/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: El administrador selecciona de la barra de navegación la opción 'Comité Científico', luego selecciona la opción 'Gestionar Comité Científico' el sistema muestra una vista en donde aparecerá el botón 'Añadir Miembro' el cual será presionado para poder acceder al formulario donde el administrador podrá llenar los campos con información válida de cada estudiante a insertar. En conjunto al botón , aparecerá una tabla de los miembros del Comité Científico en donde se apreciarán dos botones en cada fila, 'Eliminar' que sería para eliminar el miembro referente a la fila correspondiente y el botón 'Editar' el cual mostrará un formulario auto completado con los datos del miembro y podrán ser modificados sus datos y guardados al presionar el botón 'Guardar', si no se desea modificar ni agregar información a la hora de agregar un miembro se puede pulsar el botón 'Cancelar' para abortar cualquier acción.	

Tabla 30 Tarea de Ingeniería 4 Gestionar Consejo Editorial

Número de la tarea: 4	Nombre de la tarea: Gestionar Consejo Editorial
Identificador de la HU: 4	Tipo de tarea: Desarrollo
Fecha de inicio: 12/09/2023	Fecha de fin: 15/09/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: El administrador selecciona de la barra de navegación la opción 'Consejo Editorial', luego selecciona la opción 'Gestionar Consejo Editorial' el sistema muestra una vista en donde aparecerá el botón 'Añadir Miembro' el cual será presionado para poder acceder al formulario donde el administrador podrá llenar los campos con información válida de cada profesor a insertar. En conjunto al botón , aparecerá una tabla de miembros del Consejo Editorial en donde se apreciarán dos botones en cada fila, 'Eliminar' que sería para eliminar el miembro referente a la fila correspondiente y el botón 'Editar' el cual mostrará un formulario auto completado con los datos del miembro y podrán ser modificados sus datos y guardados al presionar el botón 'Guardar', si no se desea modificar ni agregar información a la hora de agregar un miembro se puede pulsar el botón 'Cancelar' para abortar cualquier acción.	

Tabla 31 Tarea de Ingeniería 6 Gestionar Tema Político

Número de la tarea: 5	Nombre de la tarea: Gestionar Tema Político
Identificador de la HU: 5	Tipo de tarea: Desarrollo
Fecha de inicio: 16/09/2023	Fecha de fin: 21/09/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: El administrador selecciona de la barra de navegación la opción 'Tema Político', el sistema muestra una vista en donde aparecerá el botón 'Añadir Tema Político' el cual será presionado para poder acceder al formulario donde el administrador podrá llenar los campos con información válida de cada Tema	

Político a insertar. En conjunto al botón, aparecerá una tabla de los Tema Político en donde se apreciará un botón en cada fila, 'Eliminar' que sería para eliminar el Tema Político referente a la fila correspondiente, si no se desea agregar información a la hora de agregar un Tema Político se puede pulsar el botón 'Cancelar' para abortar cualquier acción.

Tabla 32 Tarea de Ingeniería 8 Gestionar Conferencias

Número de la tarea: 6	Nombre de la tarea: Gestionar Conferencias
Identificador de la HU: 6	Tipo de tarea: Desarrollo
Fecha de inicio: 22/09/2023	Fecha de fin: 25/09/203
Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: El administrador selecciona de la barra de navegación la opción 'Conferencias', el sistema muestra una vista en donde aparecerá el botón 'Añadir Conferencias' el cual será presionado para poder acceder al formulario donde el administrador podrá llenar los campos con información válida de cada Conferencias a insertar. En conjunto al botón, aparecerá una tabla de las Conferencias en donde se apreciará un botón en cada fila, 'Eliminar' que sería para eliminar el Conferencias referente a la fila correspondiente, si no se desea agregar información a la hora de agregar una Conferencias se puede pulsar el botón 'Cancelar' para abortar cualquier acción.	

Tabla 33 Tarea de Ingeniería 9 Gestionar Seminario

Número de la tarea: 7	Nombre de la tarea: Gestionar Seminario
Identificador de la HU: 7	Tipo de tarea: Desarrollo
Fecha de inicio: 26/09/2023	Fecha de fin: 29/09/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: El administrador selecciona de la barra de navegación la opción 'Seminario', el sistema muestra una vista en donde aparecerá el botón 'Añadir Seminario' el cual será presionado para poder acceder al formulario donde el administrador podrá llenar los campos con información válida de cada Seminario a insertar. En conjunto al botón, aparecerá una tabla de los Seminario en donde se apreciará un boton en cada fila, 'Eliminar' que sería para eliminar el Seminario referente a la fila correspondiente, si no se desea agregar información a la hora de agregar un Seminario se puede pulsar el botón 'Cancelar' para abortar cualquier acción.	

Tabla 34 Tarea de Ingeniería 10 Gestionar Curso

Número de la tarea: 8	Nombre de la tarea: Gestionar Cursos
Identificador de la HU: 8	Tipo de tarea: Desarrollo
Fecha de inicio: 30/09/2023	Fecha de fin: 03/10/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: El administrador selecciona de la barra de navegación la opción 'Cursos', el sistema muestra una vista en donde aparecerá el botón 'Añadir Cursos' el cual será presionado para poder acceder al formulario donde el administrador podrá llenar los campos con información válida de cada Cursos a insertar. En conjunto al botón, aparecerá una tabla de los Cursos en donde se apreciará un boton en cada fila, 'Eliminar' que sería para eliminar el Curso referente a la fila correspondiente, si no se desea agregar información a la hora de agregar un	

Cursos se puede pulsar el botón 'Cancelar' para abortar cualquier acción.

Tabla 35 Tarea de Ingeniería 13 Insertar link a vínculos de interés

Número de la tarea: 9	Nombre de la tarea: Insertar link a vínculos de interés
Identificador de la HU: 9	Tipo de tarea: Desarrollo
Fecha de inicio: 04/10/2023	Fecha de fin: 07/10/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista

Tabla 36 Tarea de Ingeniería 14 Insertar contenido inherente al portal

Número de la tarea: 10	Nombre de la tarea: Insertar contenido inherente al portal
Identificador de la HU: 10	Tipo de tarea: Desarrollo
Fecha de inicio: 08/10/2023	Fecha de fin: 11/10/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista

Tabla 37 Tarea de Ingeniería 15 Buscar Usuario

Número de la tarea: 11	Nombre de la tarea: Buscar Usuarios
Identificador de la HU: 11	Tipo de tarea: Desarrollo
Fecha de inicio: 12/10/2023	Fecha de fin: 15/10/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: En la esquina superior derecha de la barra de navegación aparece un sistema de búsqueda que consta de un campo de entrada donde se puede introducir datos en específico y validos de la búsqueda en cuestión y un botón que al pulsarlo devolverá todas las coincidencias entre los datos introducidos y la información existente en la base de datos.	

Tabla 38 Tarea de Ingeniería 17 Buscar Tema político

Número de la tarea: 12	Nombre de la tarea: Buscar Tema político
Identificador de la HU: 12	Tipo de tarea: Desarrollo
Fecha de inicio: 16/10/2023	Fecha de fin: 19/10/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: En la esquina superior derecha de la barra de navegación aparece un sistema de búsqueda que consta de un campo de entrada donde se puede introducir datos en específico y validos de la búsqueda en cuestión y un botón que al pulsarlo devolverá todas las coincidencias entre los datos introducidos y la información existente en la base de datos.	

Tabla 39 Tarea de Ingeniería 18 Buscar Conferencia

Número de la tarea: 13	Nombre de la tarea: Buscar Conferencia
Identificador de la HU: 13	Tipo de tarea: Desarrollo
Fecha de inicio: 20/10/2023	Fecha de fin: 23/11/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: En la esquina superior derecha de la barra de navegación aparece un sistema de búsqueda que consta de un campo de entrada donde se puede introducir datos en específico y validos de la búsqueda en cuestión y un botón que al pulsarlo devolverá todas las coincidencias entre los datos introducidos y la información existente en la base de datos.	

Tabla 40 Tarea de Ingeniería 19 Buscar Seminario

Número de la tarea: 14	Nombre de la tarea: Buscar Seminario
Identificador de la HU: 14	Tipo de tarea: Desarrollo
Fecha de inicio: 24/10/2023	Fecha de fin: 27/10/2023

Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: En la esquina superior derecha de la barra de navegación aparece un sistema de búsqueda que consta de un campo de entrada donde se puede introducir datos en específico y validos de la búsqueda en cuestión y un botón que al pulsarlo devolverá todas las coincidencias entre los datos introducidos y la información existente en la base de datos.	

Tabla 41 Tarea de Ingeniería 20 Buscar curso

Número de la tarea: 15	Nombre de la tarea: Buscar curso
Identificador de la HU: 15	Tipo de tarea: Desarrollo
Fecha de inicio: 28/10/2023	Fecha de fin: 31/10/2023
Tiempo estimado: 1	Programador: Jose David Avila Batista
Descripción: En la esquina superior derecha de la barra de navegación aparece un sistema de búsqueda que consta de un campo de entrada donde se puede introducir datos en específico y validos de la búsqueda en cuestión y un botón que al pulsarlo devolverá todas las coincidencias entre los datos introducidos y la información existente en la base de datos.	

Anexo 5

Tabla 42 Caso de Prueba de Aceptación 2

Código: HU3_P2	Historia de Usuario: 3
Nombre: Modificar Comité Científico	
Descripción: Prueba para la funcionalidad de modificar un miembro del Comité Científico en el sistema.	
Condiciones de ejecución: ➤ El usuario debe estar previamente registrado como administrador. O coordinador del Comité Científico ➤ El miembro del Comité Científico debe existir en la base de datos.	
Pasos de ejecución: ➤ El usuario selecciona de la barra de navegación la opción '<i>Comité Científico</i>' ➤ El usuario selecciona del submenú la opción '<i>Gestionar Comité Científico</i>' ➤ El usuario presiona el botón '<i>Editar</i>' ➤ El usuario llena todos los campos requeridos con información válida ➤ El usuario presiona el botón '<i>Guardar</i>'	
Resultados esperados: Se modificará el miembro del Comité Científico en la base de datos y se mostrará un mensaje que indica que se ha modificado satisfactoriamente.	
Evaluación de la prueba: Satisfactoria	

Tabla 43 Caso de Prueba de Aceptación 3

Código: HU4_P1	Historia de Usuario: 4
Nombre: Insertar Consejo Editorial	
Descripción: Prueba para la funcionalidad de insertar un miembro del Consejo Editorial en el sistema.	

Condiciones de ejecución:
➤ El usuario debe estar previamente registrado como administrador. O coordinador del Consejo Editorial
Pasos de ejecución:
➤ El usuario selecciona de la barra de navegación la opción ' <i>Consejo Editorial</i> '
➤ El usuario presiona el botón ' <i>Añadir Miembro</i> '
➤ El usuario llena todos los campos requeridos con información válida
➤ El usuario presiona el botón ' <i>Guardar</i> '
Resultados esperados: Se insertará el miembro del Consejo Editorial en la base de datos y se mostrará un mensaje que indica que se ha insertado satisfactoriamente.
Evaluación de la prueba: Satisfactoria

Tabla 44 Caso de Prueba de Aceptación 4

Código: HU4_P2	Historia de Usuario: 4
Nombre: Modificar Consejo Editorial	
Descripción: Prueba para la funcionalidad de modificar un miembro del Consejo editorial en el sistema.	
Condiciones de ejecución:	
➤ El usuario debe estar previamente registrado como administrador. O coordinador del Consejo Editorial	
➤ El miembro del Consejo Editorial debe existir en la base de datos.	
Pasos de ejecución:	
➤ El usuario selecciona de la barra de navegación la opción ' <i>Consejo Editorial</i> '	
➤ El usuario selecciona del submenú la opción ' <i>Gestionar Consejo Editorial</i> '	
➤ El usuario presiona el botón ' <i>Editar</i> '	
➤ El usuario llena todos los campos requeridos con información válida	

➤ El usuario presiona el botón ' <i>Guardar</i> '
Resultados esperados: Se modificará el miembro del Consejo Editorial en la base de datos y se mostrará un mensaje que indica que se ha modificado satisfactoriamente.
Evaluación de la prueba: Satisfactoria

Tabla 45 Caso de Prueba de Aceptación 7

Código: HU7_P1	Historia de Usuario: 7
Nombre: Insertar Tema Político	
Descripción: Prueba para la funcionalidad de insertar un Tema Político en el sistema.	
Condiciones de ejecución:	
➤ El usuario debe estar previamente registrado como administrador.	
Pasos de ejecución:	
➤ El usuario selecciona de la barra de navegación la opción 'Tema Político'	
➤ El usuario presiona el botón ' <i>Añadir</i> Tema Político'	
➤ El usuario llena todos los campos requeridos con información válida	
➤ El usuario presiona el botón ' <i>Guardar</i> '	
Resultados esperados: Se insertará un Tema Político en la base de datos y se mostrará un mensaje que indica que se ha insertado satisfactoriamente.	
Evaluación de la prueba: Satisfactoria	

Tabla 46 Caso de Prueba de Aceptación 8

Código: HU7_P1	Historia de Usuario: 7
Nombre: Insertar Conferencia	
Descripción: Prueba para la funcionalidad de insertar una Conferencia en el sistema.	
Condiciones de ejecución:	

➤ El usuario debe estar previamente registrado como administrador.
Pasos de ejecución: ➤ El usuario selecciona de la barra de navegación la opción '<i>Conferencia</i>' ➤ El usuario presiona el botón '<i>Añadir Conferencia</i>' ➤ El usuario llena todos los campos requeridos con información válida ➤ El usuario presiona el botón '<i>Guardar</i>'
Resultados esperados: Se insertará una Conferencia en la base de datos y se mostrará un mensaje que indica que se ha insertado satisfactoriamente.
Evaluación de la prueba: Satisfactoria

Tabla 47 Caso de Prueba de Aceptación 9

Código: HU6_P2	Historia de Usuario: 6
Nombre: Modificar Conferencia	
Descripción: Prueba para la funcionalidad de modificar una Conferencia en el sistema.	
Condiciones de ejecución: ➤ El usuario debe estar previamente registrado como administrador. ➤ La Conferencia debe existir en la base de datos.	
Pasos de ejecución: ➤ El usuario selecciona de la barra de navegación la opción '<i>Conferencia</i>' ➤ El usuario presiona el botón '<i>Editar</i>' ➤ El usuario llena todos los campos requeridos con información válida ➤ El usuario presiona el botón '<i>Guardar</i>'	
Resultados esperados: Se modificará la información referente a la Conferencia en la base de datos y se mostrará un mensaje que indica que se ha modificado satisfactoriamente.	
Evaluación de la prueba: Satisfactoria	

Tabla 48 Caso de Prueba de Aceptación 10

Código: HU7_P1	Historia de Usuario: 7
Nombre: Insertar Seminario	
Descripción: Prueba para la funcionalidad de insertar un Seminario en el sistema.	
Condiciones de ejecución: ➤ El usuario debe estar previamente registrado como administrador.	
Pasos de ejecución: ➤ El usuario selecciona de la barra de navegación la opción 'Seminario' ➤ El usuario presiona el botón 'Añadir Seminario' ➤ El usuario llena todos los campos requeridos con información válida ➤ El usuario presiona el botón 'Guardar'	
Resultados esperados: Se insertará un Seminario en la base de datos y se mostrará un mensaje que indica que se ha insertado satisfactoriamente.	
Evaluación de la prueba: Satisfactoria	

Tabla 49 Caso de Prueba de aceptación 11

Código: HU6_P2	Historia de Usuario: 6
Nombre: Modificar Seminario	
Descripción: Prueba para la funcionalidad de modificar un Seminario en el sistema.	
Condiciones de ejecución: ➤ El usuario debe estar previamente registrado como administrador. ➤ El Seminario debe existir en la base de datos.	
Pasos de ejecución: ➤ El usuario selecciona de la barra de navegación la opción 'Seminario' ➤ El usuario presiona el botón 'Editar' ➤ El usuario llena todos los campos requeridos con información válida	

➤ El usuario presiona el botón ' <i>Guardar</i> '
Resultados esperados: Se modificará la información referente al Seminario en la base de datos y se mostrará un mensaje que indica que se ha modificado satisfactoriamente.
Evaluación de la prueba: Satisfactoria

Tabla 50 Caso de Prueba de Aceptación 12

Código: HU7_P1	Historia de Usuario: 7
Nombre: Insertar Curso	
Descripción: Prueba para la funcionalidad de insertar un Curso en el sistema.	
Condiciones de ejecución:	
➤ El usuario debe estar previamente registrado como administrador.	
Pasos de ejecución:	
➤ El usuario selecciona de la barra de navegación la opción 'Curso'	
➤ El usuario presiona el botón ' <i>Añadir</i> Curso'	
➤ El usuario llena todos los campos requeridos con información válida	
➤ El usuario presiona el botón ' <i>Guardar</i> '	
Resultados esperados: Se insertará un Curso en la base de datos y se mostrará un mensaje que indica que se ha insertado satisfactoriamente.	
Evaluación de la prueba: Satisfactoria	

Tabla 51 Caso de Prueba de Aceptación 13

Código: HU6_P2	Historia de Usuario: 6
Nombre: Modificar Curso	
Descripción: Prueba para la funcionalidad de modificar un Curso en el sistema.	
Condiciones de ejecución:	
➤ El usuario debe estar previamente registrado como administrador.	
➤ El Curso debe existir en la base de datos.	

Pasos de ejecución:

- El usuario selecciona de la barra de navegación la opción '*Curso*'
- El usuario presiona el botón '*Editar*'
- El usuario llena todos los campos requeridos con información válida
- El usuario presiona el botón '*Guardar*'

Resultados esperados: Se modificará la información referente al Curso en la base de datos y se mostrará un mensaje que indica que se ha modificado satisfactoriamente.

Evaluación de la prueba: Satisfactoria