



Universidad de las Ciencias Informáticas

Facultad 4

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

Título:

Desarrollo del Portal Web de la Facultad 4 de la
Universidad de las Ciencias Informáticas.

Autor: Roylan Mijares Aguirre

Tutor: Ing. Yasirys Terry González

Ing. Lester Collado Rolo

Ing. Alejandro Mendaro Morell

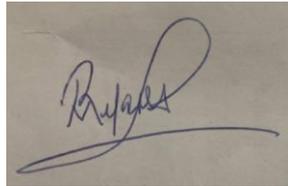
La Habana, diciembre de 2022

Año 64 de la Revolución

Declaración de Autoría

Declaro ser autor del presente trabajo de diploma y autorizo a la Universidad de las Ciencias Informáticas, para que hagan el uso que estimen pertinente con el mismo.

Para que así conste firmo la presente a los 14 días del mes de diciembre del año 2022.



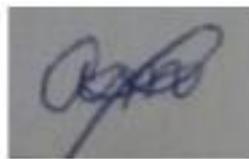
Firma del Autor

Roylan Mijares Aguirre



Firma del Tutor

Ing. Yasirys Terry González



Firma del Tutor

Ing. Alejandro Mendaro Morell

Firma del Tutor

Ing. Lester Collado Rolo



**"DUDA DEL QUE QUIERAS,
PERO NUNCA DE TI MISMO"**

@DICHOSDEMALUMA

Dedicatoria

*A mi madre por ser mi ejemplo a
seguir*

Agradecimientos

Dicen que el mejor legado que se nos pueden dejar los padres son los estudios y yo particularmente me siento muy agradecido con mi familia por haberme guiado y permitirme trazar mi camino con los pies puestos en la tierra. El amor y la dedicación recibida, sencillamente fue especial y se refleja en la vida de un hijo. Gracias a mi madre por ser en todo momento mi motor y mi mayor inspiración para convertirme en una mejor persona cada día. A mi padre Luis que donde quiera que este esto va dedicado a él. A mi abuela Ofelia por darme todo el amor que un nieto pueda soñar. A mis otros padres Meriño y Gordito por estar ahí siempre que los necesité. A mis hermanos Luis Dayan, Lavy, Baby, Yeison y Daylan por el apoyo, ellos son la otra mitad de mi vida. A mi tía María, mis primas María y Claudia, mi tío Osmani y en general a toda mi familia que ellos son mi mayor motivación.

Quiero dedicar esta tesis también a una persona muy especial que llegó en el momento indicado para reorganizar mi vida y darme la tranquilidad que necesitaba, gracias Mi Fritura, Mi Pastelito Camila por apoyarme en este último tiempo de mi vida. Gracias por todo lo lindo que hemos vivido y por todo lo que nos queda por vivir.

Gracias a todos los Orishas, mis padrinos y madrinas por cada día bendecir mi vida y permitirme celebrar este logro al lado de las personas que amo. En estas líneas quiero agradecer a todas las personas que de una manera u otra fueron parte de esta investigación y hablo de Lester el mejor tutor que pude haber tenido, gracias por el apoyo incondicional y el cariño sincero, a Yasirys por la paciencia y por todos los momentos recorridos en los últimos años, sepan que ustedes son parte fundamental para el cumplimiento de este sueño.

Quisiera agradecerles a todas las personas que he conocido en estos últimos años y hoy los considero amigos dígame Alejandro (Mi Compadre), Jordan (Mi negro, mi hermano y mi sangre), Dayron (Hermano de Batallas y sueños), Claudio (El Hermano menor que la vida me dio), Ale (El Yumita ya), Carlos (Antes Puro ahora Compadre),

a Javy Nigga a Willy el Charlie Flow, a mi gemelo de pensar Rubén, Roloo que a cada rato quería dejar la escuela. A mis hermanitas Maivi y Yennifer por ser incondicional siempre, a Jennifer por todo lo vivido (Historia larga), a mi Chamaca Rossy (Muchas anécdotas entre tú y yo), a Mi Amiga Jenny (Quien llegó para quedarse con una parte de mi cora), A mi Nany Chupieska (No se me va a olvidar nuestro 1er encuentro), a mis perdu Natha, Charo y Dayanis (Muchas pero muchas historias vividas), a mi tata Lianet (más que eso es mi hermana).

A mi fanática Nailita mi Kikis Favorita, a mi Karol Yaili la Bichota, a mi Bby Toxic Damaris, a mis piezas Naomi y Ceilan, a mi Modelo Lily, mi champagne Yaile, mi ecológica Diana Laura, a Perdu Lia todas fritas con Papi Roy jajaja, Mi hermanito Amiel, EL Tanque Manu Manuu, mi pura Didi, mi pura Jessica, mi pura y comadre Melissa, mi pura Leslie, mis cuñis Nelsa, Elizabeht, Mercedes, Yari, Camila, Adalys, Rache...

Gracias a esas personas que conocí en 1er año algunas ya graduadas, otras no y las otras ustedes saben dónde, gracias Migue Pututi, Jose el mangote (Maestros en el arte de la seducción), el super Javy, mi puro Migue, Ida, Ana Karla, Javier el presi, J Balbu, Cristian Rubio, Jenny mi cantante,

Gracias a ese CA. que tantas cosas buenas me trajo, gracias mi bro el Pichy, El Paciente, Fifi (que clase combo éramos en el padrino) a mis, Rirris Legna y Diansy, a mi Pelirroja Laura, mis hermanitos Angel Yony, Randi Nota, mi chamaco Liam, mis panas Pedro, Pichy y Rene, mi otra ma Rache, lo jefazos Kilmer y Enier, mi flaca Laura, mi secuestro Rosme, mi Flaca Emy.

No quiero que en realidad se me quede nadie siempre digo que soy afortunado de tener a mi lado personas que me aprecian de corazón, gracias por estar presente Flaco Rodney, Flaky G, Yaine Rizitos, La pequeña toxica llena de odio, mi hermanito Tito, Mendaro, El Dayan, Hector, Elio, La Rata, El Pingüino, La rata vieja, Thais, Carlita, Gaby, Mi 3er Chama Daniel, Daniela la mala influencia, Betty los ojitos, Julito, El Prieto Clarke, Freddy, Ryota, Félix, Frank, Yari, El Jorgito Peña, Rosa Elena, Vivi, El Yisus...

Gracias a mi Facultad 4 que tanto amo y lucho a muerte por ella, a todos los profes que tuve a los buenos y a los que me hicieron este camino más difícil también gracias.

Se que se me quedan pero que sepan q a todos los que significaron algo en este transcurso los llevo en mi corazón.

Atte. Papi Roy

Resumen

La presente investigación se refiere a un problema de actualidad y gran importancia para los centros educativos como es la gestión de sus recursos informativos. La misma se encarga de desarrollar el portal web de la Facultad 4, el cual tiene como objetivo diseñar un nuevo portal web con funcionalidades que favorecerán la divulgación y acceso a los recursos de información que gestiona la facultad. Para guiar el proceso que se desprende de la realización de este trabajo de diploma se utilizó como metodología de desarrollo de software AUP-UCI, como lenguaje de modelado UML y Visual Paradigm como herramienta CASE, lo que permitió un correcto entendimiento entre el cliente y el equipo de desarrollo. Para la construcción de la solución se utilizó WordPress como Sistema de Gestión de Contenidos (CMS) así como otras herramientas y tecnologías de código abierto. El principal aporte de esta investigación es el desarrollo de una herramienta para mantener una continua comunicación entre los miembros de la facultad.

Palabras claves: portal web, wordpress

Abstract:

The present investigation refers to a topical problem and great importance for educational centers such as the management of their information resources. It is in charge of developing the Faculty 4 web portal, which aims to design a new web portal with functionalities that will favor the dissemination and access to the information resources managed by the Faculty. To guide the process that emerges from the completion of this diploma work, AUP-UCI software development methodology was used, as UML modeling language and Visual Paradigm as a CASE tool, which allowed a correct understanding between the client and the user. development team. For the construction of the solution, WordPress was used as a Content Management System (CMS) as well as other open source tools and technologies. The main contribution of this research is the development of a tool to maintain continuous communication between faculty members.

Keywords: web portal, wordpress

Índice

Declaración de Autoría.....	i
Dedicatoria.....	iii
Agradecimientos	iv
Resumen.....	vii
Abstract:.....	viii
Índice	ix
Índice de Figuras.....	xi
Índice de Tablas.....	xii
Introducción.....	1
Métodos teóricos:.....	3
Métodos empíricos:	3
Capítulo 1: Fundamentación teórica	5
1.1 Conceptos fundamentales	5
1.1.1 Evolución de la Web	5
Web 1.0.....	6
Web 2.0.....	6
Web 3.0.....	6
Web 4.0.....	7
1.2 Sistemas homólogos	7
1.2.1 Portales de organizaciones internacionales	8
1.2.2. Portales de Organizaciones Nacionales.....	8
1.2.3 Resultado del estudio de sistemas homólogos	9
1.3 Tecnologías y Herramientas.....	9
1.3.1 Metodología del desarrollo del software.....	9
1.3.2 Sistema de gestión de contenidos	11
1.3.3 Lenguaje para el modelado.....	12
1.3.4 Tecnologías y lenguajes del lado del cliente	12
1.3.5 Lenguajes de programación.....	14
1.3.5.1 Lenguajes del lado del servidor.....	14
1.3.6 Servidores de aplicación	14
1.3.7 Gestor de Base de Datos.....	15
1.3.8 Herramientas	15
1.4 Conclusiones parciales.....	16
Capítulo 2: Descripción del Portal Web para la Facultad 4 de la Universidad de las Ciencias Informáticas.....	17
2.1 Propuesta de Solución	17
2.1.1 Estructura del Portal	17

2.2 Especificaciones de requisitos de software	18
2.2.1 Requisitos Funcionales	18
2.2.2 Requisitos No Funcionales	23
2.2.3 Descripción de requisitos de software (Historia de usuarios)	24
2.3 Análisis y Diseño	26
2.3.1 Diseño Arquitectónico	26
2.4 Patrones de Diseño	27
2.4 Conclusiones Parciales	28
Capítulo 3. Implementación y evaluación de la solución propuesta.....	29
3.1 Modelo de Despliegue	29
3.1. Estándares de Codificación	30
3.2. Aplicación de la estrategia de validación del portal web de la Facultad 4	34
3.2.1. Diseño de los escenarios de prueba.....	36
3.2.2. Resultado de las pruebas realizadas	37
Pruebas de funcionalidad:	37
Pruebas de Rendimiento	38
3.3 Conclusiones Parciales	42
Conclusiones Generales.....	43
Recomendaciones	44
Bibliografía.....	45
Anexos	48

Índice de Figuras

Figura 1 Diagrama de Despliegue. (Fuente: Elaboración Propia)	29
Figura 2 Estilo en Bloques	30
Figura 3 Uso de Espacio.....	31
Figura 4 Condición Yoga	32
Figura 5 Selectores.....	33
Figura 6 Orden de prioridad	33
Figura 7 Entorno de pruebas	36
Figura 8 Cantidad de defectos por iteración. (Fuente: Elaboración propia)	37
Figura 9 Resumen de las pruebas de rendimiento (Fuente: Elaboración propia)	40
Figura 10 Defecto por tipo de prueba. Fuente: Elaboración propia)	41
Figura 11 Defecto por tipo de prueba. Fuente: Elaboración propia)	42

Índice de Tablas

Tabla 1 Parámetros e indicadores para el estudio de sistemas homólogos (Fuente: Elaboración propia).....	7
Tabla 2 Arquitectura de información de la propuesta de solución Fuente: (Elaboración Propia).....	18
Tabla 3 Requisitos Funcionales Fuente: (Elaboración Propia).....	19
Tabla 4 Requisitos no funcionales Fuente: (Elaboración Propia).....	23
Tabla 5 Autenticar Usuario Fuente: (Elaboración Propia).....	25
Tabla 6 Especificación de pruebas.....	35
Tabla 7 Diseño del Caso de Prueba Autenticar Usuario. (Fuente: Elaboración propia).....	36
Tabla 8 Resumen de los resultados de las pruebas de rendimiento. (Elaboración Propia).....	39
Tabla 9 Cerrar Sesión Fuente: (Elaboración Propia).....	48
Tabla 10 Asignar rol a un usuario Fuente: (Elaboración Propia).....	48
Tabla 11 Añadir Noticia Fuente: (Elaboración Propia).....	48
Tabla 12 Editar Noticia Fuente: (Elaboración Propia).....	49
Tabla 13 Eliminar Noticia Fuente: (Elaboración Propia).....	49
Tabla 14 Mostrar Noticia Fuente: (Elaboración Propia).....	50
Tabla 15 Añadir Encuesta Fuente: (Elaboración Propia).....	50
Tabla 16 Editar Encuesta Fuente: (Elaboración Propia).....	51
Tabla 17 Eliminar Encuesta Fuente: (Elaboración Propia).....	51
Tabla 18 Mostrar Encuesta Fuente: (Elaboración Propia).....	51
Tabla 19 Realizar Búsqueda Fuente: (Elaboración Propia).....	52
Tabla 20 Añadir comentario Fuente: (Elaboración Propia).....	52
Tabla 21 Editar Comentario Fuente: (Elaboración Propia).....	53
Tabla 22 Eliminar comentario Fuente: (Elaboración Propia).....	53
Tabla 23 Mostrar comentario Fuente: (Elaboración Propia).....	53
Tabla 24 Añadir aviso Fuente: (Elaboración Propia).....	54
Tabla 25 Editar aviso Fuente: (Elaboración Propia).....	54
Tabla 26 Eliminar aviso Fuente: (Elaboración Propia).....	55
Tabla 27 Mostrar aviso Fuente: (Elaboración Propia).....	55
Tabla 28 Añadir evento Fuente: (Elaboración Propia).....	56
Tabla 29 Editar evento Fuente: (Elaboración Propia).....	56
Tabla 30 Eliminar evento Fuente: (Elaboración Propia).....	57
Tabla 31 Mostrar evento Fuente: (Elaboración Propia).....	57
Tabla 32 Añadir imagen de portada Fuente: (Elaboración Propia).....	58

Tabla 33 Mostrar enlace a redes sociales Fuente: (Elaboración Propia)	58
Tabla 34 Añadir colección de imágenes Fuente: (Elaboración Propia)	58
Tabla 35 Editar colección de imágenes Fuente: (Elaboración Propia)	59
Tabla 36 Eliminar colección de imágenes Fuente: (Elaboración Propia)	59
Tabla 37 Mostrar colección de imágenes Fuente: (Elaboración Propia).....	60
Tabla 38 Contactar al administrador Fuente: (Elaboración Propia)	60
Tabla 39 Editar página básica Fuente: (Elaboración Propia).....	61
Tabla 40 Mostrar página básica Fuente: (Elaboración Propia).....	61
Tabla 41 Filtrar contenido Fuente: (Elaboración Propia)	61
Tabla 42 Mostrar enlaces de interés Fuente: (Elaboración Propia).....	62
Tabla 43 Asignar permisos de publicación por sección Fuente: (Elaboración Propia).	62
Tabla 44 Mostrar información de los directivos de la facultad Fuente: (Elaboración Propia).....	63
Tabla 45 Mostrar Información de los dirigentes de la FEU Fuente: (Elaboración Propia)	63
Tabla 46 Mostrar información de los Centros de Desarrollo de la Facultad Fuente: (Elaboración Propia)	64
Tabla 47 Mostrar información de la facultad Fuente: (Elaboración Propia)	64

Introducción

El uso de las Tecnologías de la Información y las Comunicaciones (TIC) en la cotidianidad, ha significado a escala mundial un salto vertiginoso en el desarrollo científico técnico. Desde su llegada a los escenarios cubanos se han convertido en un elemento indispensable para establecer las líneas de desarrollo de la sociedad. (1)

Strickland y Thompson (2004) definen las tecnologías de información y comunicación, como aquellos dispositivos, herramientas, equipos y componentes electrónicos; capaces de manipular información que soportan el desarrollo y crecimiento económico de cualquier organización. Cabe destacar que en ambientes tan complejos como los que deben enfrentar hoy en día las organizaciones, sólo aquellas que utilicen todos los medios al alcance, y aprendan a aprovechar las oportunidades del mercado visualizando siempre las amenazas, podrán lograr el objetivo de ser exitosas.

Dado que las TIC tienen un alcance y evolución amplio, llegando a impactar en sectores importantes de la sociedad actual, como lo son el campo económico, social y educativo, existen ventajas específicas que se podrían aprovechar con su utilización. Se destacan como favorecen la comunicación a larga distancia pues cada vez son menos las barreras que frenan la interacción entre unos y otros, pues han hecho posible el intercambio de mensajes a distancia y de forma instantánea. (1)

Como parte de las acciones acometidas por el estado cubano para impulsar la utilización de las TIC, a partir de la Batalla de Ideas se emprendieron nuevos programas destinados a elevar el nivel cultural de la población y su calidad de vida. El Comandante en Jefe Fidel Castro Ruz propuso convertir el territorio que ocupaba la Base Lourdes, en la Universidad de Ciencias Informáticas (UCI). Fue fundada en el año 2002 y constaba en su estructura de varias facultades con el objetivo de contribuir a la informatización del país y desarrollar la industria del software para apoyar el progreso económico del mismo.

La implementación de las nuevas tecnologías de la información en los ambientes de aprendizaje ha alcanzado tal importancia, que cada día es mayor el número de establecimientos educativos que poseen un portal web y ponen a disposición de los usuarios diferentes recursos. Estos sirven de plataforma para la divulgación de información institucional, la presentación del equipo docente, el conocimiento de las actividades y otros servicios.

La universidad en general y en particular las facultades que la componen, deben contar con un espacio y dominio propio en la red, diseñado y creado especialmente para servir a sus fines educativos y comunicacionales, por lo que se desarrollan y utilizan portales

web. Un portal web visualmente atractivo, personalizado y funcional, potencia la relación entre las instituciones y sus estudiantes; pero también con cualquier otra persona: trabajadores o profesores que buscan mayor información sobre dicha institución, que desean orientación sobre los contenidos, cursos especializados, promociones, noticias, libros, entre otros.

El uso de la red en la Universidad de las Ciencias Informáticas (UCI) constituye un medio ideal para difundir en la comunidad universitaria una realidad institucional en la que se exhiban los avances en los quehaceres investigativos, científicos y tecnológicos y otros detalles sobre la vida sociopolítica y cultural del centro. (2)

En la Facultad 4 de la Universidad de las Ciencias Informáticas se desarrollan los procesos claves de formación, investigación y extensión, instituidos por el Ministerio de Educación Superior y además se realizan actividades relacionadas con la producción de software. Mantener informados a más de 600 estudiantes entre el curso regular diurno y el curso por encuentros, además de 150 trabajadores que se involucran en las actividades cotidianas de esta facultad implica una gran complejidad. Anteriormente se encontraba en uso un portal web conocido como el portal de los caimanes. El portal poseía deficiencias en diferentes elementos que van desde el aspecto visual que, no se corresponde con las expectativas y necesidades de representatividad e identidad de la facultad, hasta problemas funcionales.

Los principales medios para la divulgación de información son el correo electrónico y el mural. Los espacios que permiten mayores intercambios son las reuniones de trabajo, mítines y matutinos. Estos espacios están planificados para momentos específicos, por lo que no garantizan la inmediatez en el flujo de información. Dadas las características de los espacios informativos más utilizados en la Facultad 4, también es difícil que se propague hacia otras áreas de la Universidad la información sobre las actividades que se realizan los resultados que se obtienen.

Teniendo en cuenta la situación problemática planteada, se define como **problema de investigación**: ¿Cómo mejorar la difusión de la información asociada a las áreas y procesos de la Facultad 4 en la Universidad de las Ciencias Informáticas?

El **objeto de estudio** está centrado en la difusión de la información a través de portales web y el **campo de acción** queda enmarcado en el proceso de difusión de la información en portales web institucionales.

Basado en la idea anteriormente expuesta se define como **objetivo general** desarrollar un portal web que permita mejorar la difusión de la información asociada a las áreas y procesos de la Facultad 4 en la Universidad de las Ciencias Informáticas.

Con el fin de lograr el anterior objetivo se proponen las siguientes **Tareas investigativas**:

1. Analizar los fundamentos teóricos y tendencias en el desarrollo de portales web.
2. Investigación de soluciones existentes.
3. Definir la metodología, tecnologías y herramientas a emplear en la implementación del portal web de la Facultad 4 en la Universidad de las Ciencias Informáticas
4. Especificación de los requisitos funcionales y no funcionales de la solución.
5. Implementar las funcionalidades del portal web de la Facultad 4 en la Universidad de las Ciencias Informáticas.
6. Validar la solución desarrollada a partir de una estrategia de pruebas.

Para lograr un correcto cumplimiento del objetivo de la investigación se utilizaron:

Métodos teóricos:

Analítico-Sintético: Utilizado para el análisis de la información y la bibliografía referente a los servicios web, conceptos fundamentales y la informatización de procesos dentro de una organización, tomando datos de diferentes autores para la síntesis de los elementos más importantes o relevantes del tema.

Histórico-Lógico: Utilizado con el objetivo de comprender los antecedentes y las tendencias actuales de los portales web, y su uso para la comunicación, así como la informatización de procesos dentro de las organizaciones y su creciente importancia a medida que pasa el tiempo y surgen nuevas tecnologías.

Modelado: El empleo de este método posibilita la elaboración de diagramas para el desarrollo de los componentes visuales teniendo presentes los artefactos que propone la metodología de desarrollo de software seleccionada.

Métodos empíricos:

La observación: Posibilita obtener conocimiento acerca de los portales existentes en la actualidad, con la finalidad de detectar carencias y potencialidades en el entorno observado que pueden ser corregidas y aprovechadas respectivamente en la propuesta de solución para satisfacer las necesidades del usuario final.

Análisis documental: Se emplea en la revisión de la bibliografía asociada al objeto de estudio. Se revisa el proceso de gestión de información, recursos informativos, fuentes de información, así como materiales relacionados con el estudio de parámetros e indicadores definidos para evaluar portales web, se incluyen las tecnologías, herramientas, estándares, patrones y metodologías para el desarrollo de la propuesta.

La presente investigación queda estructurada de la siguiente manera: introducción, tres capítulos, conclusiones generales, recomendaciones y referencias bibliográficas empleadas durante el desarrollo de la investigación. A continuación, se muestra la descripción de los capítulos:

Capítulo 1: Fundamentación teórica sobre el desarrollo web: consiste en llevar a cabo la fundamentación teórica del tema a investigar a partir de un estudio del estado del arte a nivel nacional e internacional teniendo en cuenta las tendencias actuales y sistemas homólogos. Además, contendrá la fundamentación del uso de la metodología, tecnologías y herramientas escogidas para el desarrollo de la propuesta de investigación.

Capítulo 2: Diseño y propuesta de solución al problema: en este capítulo se explica cómo se desarrolla el flujo actual de los procesos, y se realiza la propuesta de solución para resolver el problema planteado. Por otra parte, se especifican los requisitos funcionales y no funcionales, y los elementos fundamentales del diseño y de la arquitectura que se deben tener en cuenta para el desarrollo del sistema, ayudado por la realización del modelado de diagramas.

Capítulo 3: Implementación y pruebas de la propuesta de solución: En este capítulo se muestran las clases e implementaciones más relevantes que constituyen un aporte a la investigación, y se establecen los estándares de codificación que se tuvieron en cuenta para el desarrollo del portal web. Se describen las pruebas realizadas para la validación del portal web y se exponen los resultados obtenidos.

Capítulo 1: Fundamentación teórica

En este capítulo se realiza un estudio de las principales tendencias de la web en el mundo, así como de las tecnologías, herramientas y la metodología a utilizar para el desarrollo del portal web. Se abordarán temas relacionados con la evolución y desarrollo de los portales web. Esto dará un punto de partida para la implementación y el desarrollo de la propuesta solución.

1.1 Conceptos fundamentales

Sitio Web:

Un sitio web es el conjunto de archivos electrónicos y páginas web referentes a un tema en particular, que incluye una página inicial de bienvenida, generalmente denominada *home page*, con un nombre de dominio y dirección en Internet específicos. Un sitio web es un gran espacio documental organizado que la mayoría de las veces está típicamente dedicado a algún tema particular o propósito específico. (3)

Difusión:

La difusión se define como un proceso por el cual una información es comunicada a través de ciertos canales y en el tiempo, entre los miembros de un sistema social. (4)

Información:

Se denomina información al conjunto de datos, ya procesados y ordenados para su comprensión, que aportan nuevos conocimientos a un individuo o sistema sobre un asunto, materia, fenómeno o ente determinado. (4)

Difusión de la información:

Se puede definir, por tanto, la difusión de información como el proceso por el cual se transmite al usuario la información que necesita o en darle la posibilidad de obtenerla. Se trata de una operación documental de salida. (4)

1.1.1 Evolución de la Web

WEB (*World Wide Web*, o *www*), es un conjunto de documentos (*webs*) interconectados por enlaces de hipertexto, disponibles en Internet que se pueden comunicar a través de la tecnología digital. Se entiende por "hipertexto" la mezcla de textos, gráficos y archivos de todo tipo, en un mismo documento. (5)

Con el paso del tiempo las tecnologías de la información y la comunicación han ido mejorando, proporcionando mejores conexiones a internet, proporcionando nuevas formas de acceso a la red (como con tecnologías móviles), apareciendo nuevos dispositivos (como *tablets* y *smartphones*) y un sinnúmero de cambios que afectan directamente a las necesidades de personas, organizaciones y empresas. La web ha

ido adaptándose a estos cambios para satisfacer estas necesidades, sobre todo de comunicación y acceso a la información. Para ver esta evolución de la web a lo largo del tiempo se pueden diferenciar cuatro etapas distintas que se conocen por la palabra “web”, seguidas de los números 1.0, 2.0, 3.0 y 4.0, para cada una de estas cuatro generaciones de la web. (6)

Web 1.0

El comienzo de la web data de los años 60, donde el usuario es un mero consumidor del contenido que es subido a servidores por parte de expertos informáticos. En esta etapa los navegadores eran solo de texto (ELISA) y a pesar de que eran muy simples, disfrutaban de una gran rapidez. La web 1.0 mejora con la aparición del lenguaje HTML, que proporciona contenido con mejor estructuras y más atractivos para leer. Aun así, el usuario aún no podía interactuar y la web era como una especie de libro donde buscar y leer información. (6)

Web 2.0

A partir del año 2001, y como consecuencia de la crisis de las empresas denominadas .com, surge un cambio importante que da comienzo a la web 2.0. A partir de tres principios básicos, el usuario empezaría a interactuar con las webs. Estos tres principios eran:

- La web como plataforma.
- La inteligencia colectiva.
- La arquitectura de participación.

La web 2.0 traería consigo la aparición de grupos de usuarios, las redes sociales, los blogs y las wikis entre otras, todas ellas fomentando la colaboración entre usuarios. El usuario ahora no solo accede a la información, sino que la crea. La implementación de la web 2.0, será utilizada para el desarrollo del portal de la FEU en la UCI, teniendo en cuenta su facilidad de uso, sus nuevas tecnologías y estándares. (6)

Web 3.0

La web 3.0 surge en 2006 para relacionar las webs de forma semántica, lo que permite que la información pueda ser encontrada de forma más rápida y eficiente debido a su estructuración. La web 3.0 está relacionada con una visión en tres dimensiones de la web, abriendo nuevas formas de comunicación y colaboración utilizando espacios tridimensionales. Ya no solo se accede a la web desde el navegador, sino que se hace uso de otros dispositivos y tecnología inteligente. En la web 3.0 el contenido y el

conocimiento se relacionan de manera más eficiente. A pesar de que la web 1.0 y 2.0 están bien definidas por hechos que marcan su desarrollo, la web 3.0 es aún fruto de debate para considerar su correcta definición. (6)

Web 4.0

Tras la web 1.0 ,2.0 y 3.0 llega la etapa actual conocida como web 4.0. En esta etapa la inteligencia artificial aparece como principal tecnología haciendo que los sitios sean inteligentes y capaces de interactuar y responder a las necesidades de los usuarios. Por eso se asocia la web 4.0 a una web predictiva.

En la web 4.0, la voz aparece como vehículo de intercomunicación (utilizado por ejemplo para realizar búsquedas). Es posible dar una orden de voz y que la misma se cumpla de manera efectiva y eficiente. “*Pide un taxi*” o “*llama a un contacto*” son órdenes que pueden darse desde dispositivos móviles u ordenadores y obtener un resultado óptimo. (6)

Algunas de las características de la web 4.0 son:

- Comprensión del lenguaje cotidiano o natural.
- Comunicación entre dispositivos (m2m, máquina a máquina).
- Uso de información relacionada (GPS, sensores de temperatura, etc.).
- Nuevas formas de interacción con el usuario.

1.2 Sistemas homólogos

Se realizó una investigación relacionada con portales web de organizaciones nacionales e internacionales con el fin de determinar características comunes que puedan ser agregadas a la solución propuesta. Para el estudio de los diferentes portales se seleccionan un conjunto de parámetros con sus indicadores correspondientes.

Tabla 1 Parámetros e indicadores para el estudio de sistemas homólogos (Fuente: Elaboración propia)

Parámetros	Indicadores
Autoría	Posibilidad de contactar con el autor o alguna institución
Informativos	Noticias
Socialización	Seguimiento en redes

1.2.1 Portales de organizaciones internacionales

Portal Web de la Universidad Nacional Autónoma de México (<https://www.unam.mx/inicio/cee.mx>)

En la página principal, en la parte superior se encuentra el logotipo de la organización, así como un buscador interno. En la misma se encuentra un menú con listas desplegables evidenciando las diferentes secciones de la misma. Se caracteriza por un diseño en grupos donde se muestra todo lo relacionado con información diaria de la organización. En el pie de página se encuentran enlaces con carácter social, además de contactos.

Portal Web Principal de la Universidad Javeriana (<https://www.javeriana.edu.co/inicio>)

En la página principal, en la parte superior se encuentra el logo, un video y 4 enlaces destinados a la información necesaria para conocimiento de quien conoce el portal por primera vez. Cuenta con un menú que proporciona distintas secciones de interés en torno a lo que se realiza en la organización. Se refleja un diseño bloques exponiendo las noticias relevantes. Al final de la página muestra enlaces rápidos dirigidos a directorios, archivos, informaciones legales y a las distintas redes sociales.

Portal Web de La Universidad Autónoma de Chile (<https://www.uaautonoma.cl/>)

En la página principal, se muestra de inicio el logo de la universidad junto con unos 7 elementos en forma de lista desplegable con la información perteneciente a todo lo relacionado con la organización. La restante información destinada a lo más reciente se encuentra organizada en bloques. En el pie de página se encuentra los distintos enlaces a redes sociales y a procesos de la misma.

1.2.2. Portales de Organizaciones Nacionales

Portal Web Universidad Central “Marta Abreu” de las Villas(<https://www.uclv.edu.cu/>)

En la página principal, en la parte superior se encuentra el logo, enlaces a las redes sociales y un buscador interno. Posee un menú con una lista desplegable con todo lo relacionado a las distintas áreas de la misma. La información del quehacer cotidiano se encuentra organizada por bloques, ya en el pie de página se encuentran enlaces de interés y acceso a distintos documentos oficiales.

Portal Web de la Universidad de las Ciencias Informáticas (<https://www.uci.cu/>)

En la página principal se evidencia el logotipo de la universidad junto con enlaces a las diferentes redes sociales además de un buscador interno y un menú relacionando los aspectos más atractivos a vistas de un usuario que nunca ha visitado la misma. Contiene la información perteneciente al período universitario en bloques y en al final posee información de distintos proyectos realizados por la institución.

1.2.3 Resultado del estudio de sistemas homólogos

Los portales web estudiados brindaron información referente a la autoría, información y socialización. Cada uno cuenta con funcionalidades que cubren sus propias necesidades, en algunos casos son muy similares a las que se necesitan establecer ya que están encaminadas a la capacidad de interactuar, consultar y leer artículos, tanto en el ámbito de alcance del centro consultado como en otros encaminados por todo el mundo. Luego de analizar lo arrojado anteriormente se toma como punto de partida para el inicio de nuestra propuesta.

1.3 Tecnologías y Herramientas

En la realización de un proyecto es sumamente importante la etapa donde se define la tecnología y las herramientas a utilizar ya que en el pasado estas herramientas eran básicamente editores de textos que permitían generar una página, pero ahora estas han evolucionado incorporando nuevas funcionalidades enfocadas a la creación que a continuación se enuncia:

1.3.1 Metodología del desarrollo del software

Las metodologías de desarrollo de software son un conjunto de técnicas y métodos organizativos que se aplican para diseñar soluciones de software informático. El objetivo de las distintas metodologías es el de intentar organizar los equipos de trabajo para que estos desarrollen las funciones de un programa de la mejor manera posible. El trabajo con una metodología de desarrollo de software permite reducir el nivel de dificultad, organizar las tareas, agilizar el proceso y mejorar el resultado final de las aplicaciones a desarrollar. (7)

Dentro de las más conocidas metodologías de desarrollo de software se encuentran RUP, MSF, XP y AUP. Otra metodología es AUP-UCI, la cual es una variante a la metodología AUP en unión con el modelo CMMI-DEV v 1.3.

La programación extrema (XP) es una metodología de desarrollo de software que forma parte de lo que se conoce colectivamente como metodologías ágiles. XP se basa en valores, principios y prácticas, y su objetivo es permitir que equipos pequeños y medianos produzcan software de alta calidad y se adapten a los requisitos cambiantes y en evolución. (8)

Lo que diferencia a XP de las demás metodologías ágiles es que hace hincapié en los aspectos técnicos del desarrollo de software. La programación extrema es precisa sobre cómo trabajan los ingenieros, ya que seguir las prácticas de ingeniería permite a los equipos entregar código de alta calidad a un ritmo sostenible. (8)

La programación extrema consiste, en pocas palabras, en las buenas prácticas llevadas al extremo. Como programación en parejas es buena, hagámosla siempre. Ya que las pruebas tempranas son buenas, probemos antes de escribir el código de producción. (8)

El Proceso Unificado Ágil o Agile Unified Process (AUP) en inglés es una versión simplificada del Proceso Unificado Racional (RUP). Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP (9). La UCI le ha realizado modificaciones con el fin de adaptarlo al ciclo de vida definido para la actividad productiva de dicha institución. De las 4 fases que encierra la metodología AUP se simplificaron en las siguientes:

Inicio: en esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo, costo y decidir si se ejecuta o no el proyecto.

Ejecución: en esta fase se recogen las actividades que desarrolla AUP de elaboración, construcción y transición. Se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura.

Cierre: En esta fase se analizan tanto los resultados del proyecto como su ejecución y se realizan las actividades formales de cierre del proyecto.

En la variación de la metodología AUP-UCI, existen tres formas de encapsular los requerimientos Casos de Uso del Sistema (CUS), Historias de usuario(HU) y Descripción de requerimientos por proceso (DRP), agrupados en cuatro escenarios:

- **Escenario No 1:** Proyectos que modelen el negocio con CUN solo pueden modelar el sistema con CUS
- **Escenario No 2:** Proyectos que modelen el negocio con MC solo pueden modelar el sistema con CUS.
- **Escenario No 3:** Proyectos que modelen el negocio con DPN solo pueden modelar el sistema con DRP.

- **Escenario No 4:** Proyectos que no modelen negocio solo pueden modelar el sistema con HU

Teniendo en cuenta las características antes mencionadas, la necesidad de una metodología que responda con facilidad a los cambios continuos y siguiendo las políticas de desarrollo de software de la UCI, se tuvo en cuenta en el desarrollo del sistema propuesto, lo planteado en la metodología AUP-UCI, sobre el encapsulamiento de los requerimientos a través del escenario cuatro, que aplica a los proyectos que hayan evaluado el negocio a informatizar y como resultado obtengan un negocio bien definido. El cliente estará siempre acompañando al equipo de desarrollo para convenir los detalles de los requisitos y así poder implementarlos, probarlos y validarlos. Se recomienda en proyectos no muy extensos, ya que una HU no debe poseer demasiada información. Se recomienda este escenario para proyectos donde el objetivo primario es la gestión y presentación de información.

1.3.2 Sistema de gestión de contenidos

Un sistema de contenido, a menudo abreviado como CMS, es un software que ayuda a los usuarios a crear, administrar y modificar contenido en un sitio web sin la necesidad de conocimientos técnicos especializados. En un lenguaje más simple, un sistema de gestión de contenido es una herramienta que le ayuda a construir un sitio web sin necesidad de escribir todo el código desde cero (o incluso saber cómo codificar). (10) Entre los CMS más conocidos se encuentran Drupal, Typo3, Contao, WordPress entre otros.

Drupal es un sistema de gestión de contenidos multipropósito, modular, libre y con una amplia capacidad de personalización. Te permite publicar archivos, imágenes, artículos, al igual que crear y administrar todo tipo de contenidos como votaciones, encuestas, foros, entre otros. A pesar de no ser tan popular como WordPress, Drupal es uno de los CMS más completos para grandes portales corporativos, además se trata de una plataforma flexible y fácilmente integrable con otras soluciones de negocio. (11)

Por otro lado, también es de las plataformas más flexibles, siendo ideal para usuarios con conocimientos avanzados. Te ayudará a crear proyectos complejos, generar una cantidad de tráfico elevada, procesar datos y desarrollar funcionalidades específicas.

WordPress (6.0) es un sistema de gestión de contenidos (CMS) que permite crear y mantener un blog u otro tipo de web. Con casi 10 años de existencia y más de un millar de temas (plantillas) disponibles en su web oficial, no es solo un sistema sencillo e intuitivo para crear un blog personal, sino que permite realizar toda clase de web más

complejas. Es un sistema ideal para un sitio web que se actualice periódicamente. Si se escribe contenido con cierta frecuencia, cuando alguien accede al sitio web, puede encontrar todos esos contenidos ordenados cronológicamente (primero los más recientes y por último los más antiguos). En muchas ocasiones se asocia WordPress con una herramienta que solo sirve para hacer blogs y no es correcto porque se pueden crear webs empresariales, tiendas online, periódico digital, central de reserva entre otras. (12)

Por las razones antes citadas se decide tomar WordPress como *framework* de desarrollo ya que el mismo es de fácil y rápida creación de contenido web. Es importante resaltar que se ha convertido en una de las plataformas más populares utilizadas para el desarrollo de aplicaciones web, sistemas de gestión de contenidos (CMS), aplicaciones de escritorio y servicios.

1.3.3 Lenguaje para el modelado

El Lenguaje Unificado de Modelado (UML) fue creado para forjar un lenguaje de modelado visual común y semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento. UML tiene aplicaciones más allá del desarrollo de software, p. ej., en el flujo de procesos en la fabricación. Es comparable a los planos usados en otros campos y consiste en diferentes tipos de diagramas. En general, los diagramas UML describen los límites, la estructura y el comportamiento del sistema y los objetos que contiene. UML no es un lenguaje de programación, pero existen herramientas que se pueden usar para generar código en diversos lenguajes usando los diagramas UML. UML guarda una relación directa con el análisis y el diseño orientados a objetos. (13)

1.3.4 Tecnologías y lenguajes del lado del cliente

HTML 5(*HyperText Markup Language*, versión 5)

HTML5 es la quinta revisión del lenguaje HTML y permite definir los nuevos estándares de desarrollo web, modificando el código existente para solucionar problemas y actualizándolo a las nuevas necesidades de hoy en día. Permite crear la estructura de una página web, por así decirlo, su esqueleto, donde vamos a incorporar posteriormente, imágenes, textos, vídeos y todo tipo de material multimedia de forma que se pueda visualizar correctamente. (14)

Algunas de las nuevas características más destacables de HTML5 son:

- ✓ Los nuevos elementos multimedia como <audio> y <video>.

- ✓ El cambio, redefinición o estandarización de algunos elementos, como <a>, <cite> o <menu>
- ✓ Nuevas etiquetas semánticas para estructurar los documentos HTML, destinadas a remplazar la necesidad de tener una etiqueta <div> que identifique cada bloque de la página.
- ✓ La integración de gráficos vectoriales escalables (SVG) en sustitución de los genéricos <object>, y un nuevo elemento <canvas> que permite dibujar en él.
- ✓ Almacenamiento local en el lado del cliente.
- ✓ MathML para fórmulas matemáticas.

Derivado de lo anterior, la compatibilidad con otros lenguajes y su facilidad de uso lo hace ideal para el desarrollo del portal web. Al incorporar etiquetas (canvas 2D y 3D, audio, video) para mostrar los contenidos multimedia, otras etiquetas para manejar grandes conjuntos de datos: *Datagrid*, *Details*, *Menu* y *Command*, permiten generar tablas dinámicas que pueden filtrar. Al incluir mejoras en los formularios y nuevos tipos de datos. (14)

CCS 3

CSS es un lenguaje de diseño gráfico que permite definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Es muy usado para establecer el diseño visual de los documentos web e interfaces de usuario escritas en HTML. (15) CSS3 es una tecnología que ha tenido una evolución en el tiempo, que actualmente se encuentra en su versión 3, como su propio nombre indica.

Sus siglas corresponden a “*Cascading Style Sheets*”, que tiene el siguiente significado:

- ✓ **Cascading**, que significa que los estilos que se aplican a los elementos de una página web se propagan a los elementos que contiene, se propagan en cascada.
- ✓ **Style**, porque mediante CSS lo que se hace es aplicar estilos visuales a los distintos elementos de nuestra página web.
- ✓ **Sheets**, que significa hojas, porque los estilos de una página web se añaden en ficheros aparte, en ficheros con la extensión .css de manera general.

En la propia definición de CSS se evidencia que está muy ligado desde su nacimiento a HTML. Desde que nació, el objetivo de CSS fue poner un poco de orden a la hora de aplicar los estilos a las páginas web. (15)

JavaScript (JS) 6

JavaScript es un lenguaje de programación que los desarrolladores utilizan para hacer páginas web interactivas. Desde actualizar fuentes de redes sociales a mostrar animaciones y mapas interactivos, las funciones de JavaScript pueden mejorar la experiencia del usuario de un sitio web. Como lenguaje de scripting del lado del servidor, se trata de una de las principales tecnologías de la *World Wide Web*. Por ejemplo, al navegar por Internet, en cualquier momento en el que vea un carrusel de imágenes, un menú desplegable “*click-to-show*” (clic para mostrar), o cambien de manera dinámica los elementos de color en una página web, estará viendo los efectos de JavaScript. (16)

1.3.5 Lenguajes de programación

Un lenguaje de programación es un lenguaje de computadora que los programadores utilizan para comunicarse y para desarrollar programas de software, aplicaciones, páginas webs, scripts u otros conjuntos de instrucciones para que sean ejecutadas por los ordenadores. (17)

1.3.5.1 Lenguajes del lado del servidor

PHP (7.4) es el acrónimo de *Hypertext Preprocessor* es un lenguaje de código abierto interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor para desarrollar aplicaciones y crear sitios web que conquista cada día más seguidores. Fácil de usar y en constante perfeccionamiento es una opción segura para aquellos que desean trabajar en proyectos calificados y sin complicaciones.

Entre los factores que hicieron que PHP se volviera tan popular, se destaca el hecho de que es de código abierto. Esto significa que cualquiera puede hacer cambios en su estructura. En la práctica, esto representa dos cosas importantes: (18)

- Es de código abierto, no hay restricciones de uso vinculadas a los derechos. El usuario puede usar PHP para programar en cualquier proyecto y comercializarlo sin problemas.
- Está en constante perfeccionamiento, gracias a una comunidad de desarrolladores proactiva y comprometida.

El código PHP se ejecuta en el servidor que, al leer los comandos, puede activar todos los elementos funcionales y la interfaz visual del sitio web. Quizás, la aplicación principal del lenguaje PHP, cuando hablemos de la web, es estructurar sitios web en WordPress.

1.3.6 Servidores de aplicación

Apache (2.4) es un poderoso servidor web, cuyo nombre proviene de la frase inglesa a *patchy server* y es completamente libre, ya que es un software *Open Source* y con

licencia GPL. Una de las ventajas más grandes de Apache, es que es un servidor web multiplataforma, es decir, puede trabajar con diferentes sistemas operativos y mantener su excelente rendimiento. Apache es usado primariamente para enviar páginas web estáticas y dinámicas en la *World Wide Web*. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor web.

Entre las principales características de Apache, se encuentran las siguientes:

- ✓ Soporte de seguridad SSL y TLS.
- ✓ Puede realizar autenticación de datos utilizando SGDB.
- ✓ Puede dar soporte a diferentes lenguajes, como Perl, PHP, Python y tcl.

Por todo lo antes planteado, se escoge como servidor a Apache en su versión 2.4 debido a su extremada rapidez para servir páginas estáticas. Por ser modular, multi-plataforma, extensible y de código abierto, por su capacidad de configuración, su potencia y por ser este servidor web recomendado para el desarrollo de portales y módulos. (19)

1.3.7 Gestor de Base de Datos

MySQL 8.0 es un SGBD relacional, multihilo, multiplataforma y multiusuario. Es una idea originaria de la empresa Open *Source* MySQL AB fundada en 1995, que pasó a manos de *Sun Microsystems* en 2008 cuando adquirió la empresa, luego en 2010 *Sun Microsystems* fue adquirida por la empresa *Oracle Corporation*, lo que justifica el desarrollo de MySQL como software libre en un esquema de licenciamiento dual. Por un lado, se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. (20)

1.3.8 Herramientas

Visual Paradigm 5.0 es una herramienta multiplataforma de modelado que hace uso del UML. Actualmente existen una gran variedad de herramientas CASE (*Computer Aided Software Engineering* o Ingeniería de Software Asistida por Computadora) para el proceso de desarrollo de software. Las herramientas CASE, están tomando cada vez más relevancia en la planeación y ejecución de proyectos que involucren sistemas de información, pues suelen inducir a sus usuarios a la correcta utilización de metodologías que le ayudan a llegar con facilidad a los productos de software construidos. Todas las herramientas CASE prestan soporte a un lenguaje de modelado para acompañar la metodología y es lógico suponer, que un alto porcentaje de ellas soportan el Lenguaje de Modelado Unificado (UML). Esto se debe a la amplia aceptación de este lenguaje,

su valor conceptual y visual, así como su facilidad para ser extendido para representar elementos particulares a determinados tipos de aplicaciones. (21)

En la UCI se ha estandarizado el uso del *Visual Paradigm for UML* en su distribución libre como herramienta CASE para el modelado de los procesos de desarrollo de software que en ella se llevan a cabo, dado por la gran cantidad de ventajas que posee, las cuales están en concordancia con los intereses y políticas establecidas en la institución. Entre sus principales características se encuentran que es multiplataforma, posee interoperabilidad, facilita la colaboración en equipo y brinda apoyo al ciclo de vida completo del desarrollo de software (22)

1.4 Conclusiones parciales

La definición del marco teórico brindó todos los elementos que sustentan la solución del problema. El estudio de los sistemas homólogos permitió identificar los principales requisitos que debe tener el portal web de la Facultad 4, para su correcto funcionamiento y aceptación por parte de los usuarios.

La decisión de establecer la metodología AUP-UCI como guía para el desarrollo y la base tecnológica permitió el análisis de herramientas y tecnologías existentes para el desarrollo de la solución propuesta; seleccionándose para el desarrollo WordPress como CMS y PHP como lenguaje de programación. Como sistema gestor de bases de datos se escogió MySQL por ser un sistema libre y seguro para la gestión de la base de datos del portal web. Como herramienta para el modelado de diagramas se seleccionó Visual Paradigm porque es una herramienta para el diseño de los múltiples artefactos necesarios para representar la información en las metodologías de desarrollo y ofrece diversas facilidades cuando se realizan los diagramas UML.

Capítulo 2: Descripción del Portal Web para la Facultad 4 de la Universidad de las Ciencias Informáticas.

En el presente capítulo se describe la propuesta de solución para lograr mejorar la difusión de la información asociada a las áreas de la Facultad 4 de la UCI. El objetivo del capítulo es presentar los componentes que conforman la solución propuesta y explicar su funcionamiento y relación. Se definen las historias de usuarios y los artefactos necesarios que servirán de base para la fase de implementación. Se presentan, además, los requisitos funcionales y no funcionales.

2.1 Propuesta de Solución

Utilizando la información recopilada en el capítulo anterior, se propone el desarrollo de un portal web que brinde a los usuarios la posibilidad de tener acceso al mismo de forma rápida y sencilla, en cualquier momento, desde cualquier lugar dentro de la UCI y a través de cualquier dispositivo diseñado con este fin. El desarrollo del portal web permitirá perfeccionar la difusión de la información asociada a las áreas y procesos de la Facultad 4 de la Universidad de las Ciencias Informáticas, de forma tal que los usuarios puedan consultar dicha información.

En la solución existirá un espacio para mostrar las novedades de cada centro de producción perteneciente a la facultad y las noticias importantes enmarcadas en el mundo del conocimiento y la información. También presentará las características de un portal especializado proporcionando a los miembros de la Facultad 4 de la UCI, de información interna o externa relacionada con la facultad y la comunidad. Además, potenciará la participación y el intercambio entre los estudiantes, trabajadores y profesores a través del correo.

Dispondrá de un buscador de contenidos que permite buscar por letras, palabras o frases, obteniendo los resultados de forma inmediata. El portal propiciará la participación del usuario para generar contenidos que pueden ser publicados y compartidos con toda la comunidad, y al mismo tiempo comentar siempre estando autenticado.

2.1.1 Estructura del Portal

El portal web de la Facultad 4 de la UCI estará conformado por un conjunto de páginas dinámicas que se irán incluyendo a medida que se vaya generando contenido en el portal. El acceso a las diferentes vistas del portal lo realizarán los usuarios, a los que se les asignarán roles que previamente tendrán permisos asignados. Cada página quedará construida por una cabecera que se dividirá en el menú principal y el logo de la facultad. Poseerá una región para mostrar el contenido principal de la página que se está visitando y un *footer* que contendrá los principales enlaces a servicios, portales y blogs

de la UCI. Además contará con una página inicial (*home*) que se divide en las regiones antes mencionadas e incluye las siguientes: una región que mostrará el *banner* promocional, seguido de los bloques de columnas relacionadas a las esferas de la FEU, a continuación se mostrará un bloque con las últimas noticias de la Facultad 4, luego las noticias de ámbito nacional e internacional, seguidamente de un espacio para los diferentes centros de la facultad y por último una región para las galerías de imágenes, contactos, enlaces de interés y las nuestras diferentes redes sociales.

Tabla 2 Arquitectura de información de la propuesta de solución Fuente: (Elaboración Propia)

Logo	Menú	Buscar	Acceder
Foto Portada			
Cultura	Deporte	Docencia	Residencia
Noticias de la Facultad			
Otras Noticias			
Centro de Desarrollo			
Galería de Imágenes			
Contáctenos		Enlaces de Interés	
Enlace a Redes Sociales			
Pie de Página			

2.2 Especificaciones de requisitos de software

La tarea del análisis de requisitos es un proceso de descubrimiento, refinamiento, modelado y especificación. Se refina en detalle el ámbito del software, y se crean modelos de los requisitos de datos, flujo de información y control, y del comportamiento operativo. Se analizan soluciones alternativas y se asignan a diferentes elementos del software. El análisis de requisitos permite al desarrollador o desarrolladores especificar la función y el rendimiento del software, indica la interfaz del software con otros elementos del sistema y establece las restricciones que debe cumplir el software. (23)

2.2.1 Requisitos Funcionales

Los requisitos funcionales para un sistema refieren lo que el sistema debe hacer. Tales requerimientos dependen del tipo de software que se esté desarrollando, de los usuarios esperados del software y del enfoque general que adopta la organización cuando se escriben los requerimientos. Al expresarse como requerimientos del usuario, los

requerimientos funcionales se describen por lo general de forma abstracta que entiendan los usuarios del sistema. Sin embargo, requerimientos funcionales más específicos del sistema detallan las funciones del sistema, sus entradas y salidas, sus excepciones... (23)

Tabla 3 Requisitos Funcionales Fuente: (Elaboración Propia)

Requisitos Funcionales			
No	Requisito	Descripción	Prioridad
1	Autenticar Usuario	El sistema debe permitir a los usuarios autenticarse.	Alta
2	Cerrar Sesión	El sistema debe permitir cerrar la sesión del usuario.	Media
3	Asignar Rol a Usuario	El Sistema permite que los usuarios con rol administrador editen al usuario.	Media
4	Añadir Noticia	El Sistema permite que los usuarios con rol de editor creen noticias.	Alta
5	Editar Noticia	El Sistema permite que los usuarios con rol de editor editen noticias.	Media
6	Eliminar Noticia	El Sistema permite que los usuarios con rol de editor eliminen noticias.	Baja
7	Mostrar Noticia	El Sistema permite que todos los usuarios observen las noticias.	Alta
8	Añadir Encuesta	El Sistema permite que los usuarios con rol de editor añadan encuestas.	Alta
9	Editar Encuesta	El Sistema permite que los usuarios con rol de editor editen encuestas.	Media
10	Eliminar Encuesta	El Sistema permite que los usuarios con rol de editor creen noticias.	Baja

11	Mostrar Encuesta	El Sistema permite que los usuarios observen las encuestas.	Alta
12	Realizar Búsqueda	El Sistema debe permitir a los usuarios realizar búsquedas.	Alta
13	Añadir Comentario	El Sistema permite que los usuarios añadan comentarios.	Alta
14	Moderar Comentario	El sistema permite que los usuarios con rol de administrador den permisos para mostrar el comentario.	Media
15	Eliminar Comentario	El sistema permite que los usuarios con rol de administrador puedan eliminar los comentarios.	Baja
16	Mostrar Comentario	El sistema permite que los usuarios puedan ver los comentarios.	Alta
17	Añadir Aviso	El sistema permite que los usuarios con rol de administrador añadan avisos.	Alta
18	Editar Aviso	El sistema permite que los usuarios con rol de administrador editen los avisos.	Media
19	Eliminar Aviso	El sistema permite que los usuarios con rol de administrador eliminen los avisos.	Baja
20	Mostrar Aviso	El sistema permite que se muestren los avisos en el portal web.	Alta
21	Añadir Evento	El sistema permite que los usuarios con rol de administrador añadan eventos.	Alta

22	Editar Evento	El sistema permite que los usuarios con rol de administrador editen los eventos.	Media
23	Eliminar Evento	El sistema permite que los usuarios con rol de administrador eliminen los eventos.	Baja
24	Mostrar Evento	El sistema permite que se muestre el evento en el portal web.	Alta
25	Añadir Imagen de Portada	El sistema permite que los usuarios con rol de administrador añadan la imagen portada.	Alta
26	Mostrar Enlace de Redes Sociales	El sistema permite que los usuarios accedan a los perfiles oficiales de la organización en las redes sociales.	Alta
27	Añadir Colección de Imágenes	El sistema permite que los usuarios con rol de administrador añadan colecciones de imágenes.	Alta
28	Editar Colección de Imágenes	El sistema permite que los usuarios con rol de administrador editen colecciones de imágenes.	Media
29	Eliminar Colección de Imágenes	El sistema permite que los usuarios con rol de administrador eliminen colecciones de imágenes.	Baja
30	Mostrar Colección de Imágenes	El sistema permite que los usuarios vean las colecciones de imágenes.	Alta
31	Contactar con el Administrador	El sistema permite que los usuarios puedan comunicarse	Alta

		fácilmente con el administrador del sitio a través del correo.	
32	Editar Página Básica	El sistema permite que los usuarios con rol de administrador editen las páginas básicas.	Media
33	Mostrar Página Básica	El sistema permite mostrar al cliente las páginas básicas.	Alta
34	Filtrar Contenido	El sistema permite a los usuarios filtrar el contenido.	Media
35	Mostrar Sección de Interés	El Sistema permite mostrar a los clientes la sección de interés.	Alta
36	Asignar permisos de publicación en secciones por usuarios	El sistema debe permitir mediante el rol de administrador asignar permisos de publicación a usuarios específicos para la elaboración de la información del portal web.	Alta
37	Mostrar Información de los directivos de la facultad	El Sistema permite mostrar a los usuarios la información de los directivos de la facultad.	Alta
38	Mostrar Información de los dirigentes de la FEU	El Sistema permite mostrar a los usuarios la información de los dirigentes de la FEU.	Alta
39	Mostrar Información de los Centros de Desarrollo de la Facultad	El Sistema permite mostrar a los usuarios la información de los de los centros de desarrollo de la Facultad.	Alta
40	Mostrar Información de la Facultad	El Sistema permite mostrar a los usuarios la información de la Historia de la Facultad.	Alta

2.2.2 Requisitos No Funcionales

Los requisitos no funcionales no se relacionan directamente con los servicios específicos que el sistema entrega a sus usuarios. Pueden relacionarse con propiedades emergentes del sistema, como fiabilidad, tiempo de respuesta y uso de almacenamiento. De forma alternativa, pueden definir restricciones sobre la implementación del sistema, como las capacidades de los dispositivos I/O las representaciones de datos usados en las interfaces con otros sistemas. (24)

Tabla 4 Requisitos no funcionales Fuente: (Elaboración Propia)

Requisitos No Funcionales		
Clasificación	No.	Descripción
Usabilidad	RnF 1.	La interfaz de usuario del portal debe contar con un diseño "Responsive" con el objetivo de garantizar la adecuada visualización de este en múltiples computadoras personales, tabletas y teléfonos inteligentes
	RnF 2.	El sistema debe presentar una interfaz agradable e intuitiva para el usuario.
	RnF 3.	El sistema debe tener visibilidad en los principales navegadores web como Chrome v.45, Firefox v.38, Safari v.9, Opera v.30.
Seguridad	RnF 4.	En caso de que el sistema presente alguna falla, los errores deben mostrar la menor cantidad de detalles posible, de forma tal, que se evite dar información que comprometa la seguridad e integridad del sistema. Sólo se mostrarán detalles ampliados del error a usuarios con privilegios de administración.
	RnF 5.	La contraseña del usuario debe tener un mínimo de 8 caracteres, combinación de letras, números y caracteres especiales.
	RnF 6.	Se asignarán los permisos de acceso, escritura, lectura en dependencia del rol que desempeñe cada usuario del sistema
	RnF 7.	Para garantizar la fiabilidad del portal, se debe contar con un sistema de salvacopias externas del código, del sistema de archivos y de la base de datos para asegurar la información que maneja en el mismo.

	RnF 8.	Para garantizar la fiabilidad del portal, se debe contar con un sistema de salvadas externas del código, del sistema de archivos y de la base de datos para asegurar la información que maneja en el mismo.
	RnF 9.	El acceso al sistema de administración del portal web se realizará mediante un proceso de autenticación.
Eficiencia	RnF 10.	Se recomienda tener instalado un sistema operativo Windows 7 o superior, o cualquier sistema operativo de la familia GNU/Linux (KDE Plasma).
	RnF 11.	El tiempo de demora de una petición al servidor debe ser menor de ocho (8) segundos aproximadamente.
	RnF 12.	El sistema debe permitir trabajar de manera concurrente con un mínimo de 30 usuarios.
Software	RnF 13.	Servidor web Apache en su versión 2.4.
	RnF 14.	Servidor de base de datos MySQL en su versión 5.7.24 o superior.
	RnF 15.	Lenguaje de programación PHP 7 como soporte al CMS WordPress.
Hardware	RnF 16.	El servidor de base de datos debe poseer una capacidad mínima de 20 GB.
	RnF 17.	El servidor de aplicaciones web debe poseer una capacidad mínima de 80 GB
	RnF 18.	Los servidores web y de base de datos deben poseer como mínimo 1 GB de memoria RAM.

2.2.3 Descripción de requisitos de software (Historia de usuarios)

La Historia de Usuarios es la técnica que propone el escenario 4 de la metodología AUP-UCI para encapsular los requisitos funcionales del sistema, misma que se utiliza para la especificación de requisitos del Portal Web de la Facultad 4. Con el empleo de esta técnica se describió qué se espera como salida de la implementación, y cómo se ve beneficiado el usuario final. Se expresa en lenguaje natural y sencillo, considerando que los usuarios finales tienden a desconocer el lenguaje técnico. A continuación, se muestra la historia de usuario correspondientes al requisito funcional Autenticar usuario.

Tabla 5 Autenticar Usuario Fuente: (Elaboración Propia)

Historia de Usuario	
No. 1	Nombre del Requisito: Autenticar Usuario
Programador Responsable: Admin	Iteración Asignada: 1
Prioridad en Negocio: Alta	Tiempo Estimado: 1.5 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 1.5 Hr
Descripción: Permite que el usuario se autentique en el sistema.	
Observaciones:	
Acciones para lograr el objetivo (Precondiciones y Datos):	
<ul style="list-style-type: none"> • Ingresar los datos de Usuario y Contraseña. 	
Comportamientos Válidos y No Válidos (Flujo Central y Alternos):	
<ul style="list-style-type: none"> • Usuario: No admite ni caracteres extraños, ni numéricos, ni especiales. • Contraseña: Campo que admite un mínimo de 8 caracteres, combinados entre mayúscula, minúscula y caracteres especiales 	
Flujo de la Acción a Realizar:	
<ul style="list-style-type: none"> • El sistema debe permitir la autenticación • Cuando el usuario toca el botón aceptar ubicado en la parte superior derecha de la aplicación, se muestra una interfaz donde debe ingresar sus datos (usuario y contraseña), luego selecciona la opción acceder y si son correctos inicia sesión en la aplicación. • Si los datos son incorrectos se señala el campo en cuestión dando la posibilidad de realizar esa acción nuevamente. • Si selecciona la acción cancelar regresa a la vista previa. 	
Prototipo Interfaz:	

En el proceso de desarrollo del sistema de gestión se definió para mantener la organización, el nivel de prioridad para el negocio de cada uno de los requisitos obtenidos en la entrevista con el cliente. Los niveles se dividen en:

- **Alta:** Funcionalidades que son sumamente primordiales en la aplicación, que por su función no deberían faltar en el editor, es decir, la aplicación depende de estas para su correcto funcionamiento.
- **Media:** Funcionalidades que no son muy primordiales, pero otras dependen de estas para poder ejecutarse.

- **Baja:** Funcionalidades que su existencia en la aplicación no es significativa, es decir, la aplicación puede funcionar correctamente sin estas.

2.3 Análisis y Diseño

2.3.1 Diseño Arquitectónico

La arquitectura del software se trata de abstracciones, de descomposición y composición, de estilos y estética. También tiene relación con el diseño y la implementación de la estructura de alto nivel del software. Los diseñadores construyen la arquitectura usando varios elementos arquitectónicos elegidos apropiadamente. Estos elementos satisfacen la mayor parte de los requisitos de funcionalidad y performance del sistema, así como también otros requisitos no funcionales. (25)

En el presente trabajo se utilizará como modelo arquitectónico el Modelo-Vista-Controlador el cual se menciona a continuación:

MVC (Modelo-Vista-Controlador) es un patrón en el diseño de software comúnmente utilizado para implementar interfaces de usuario, datos y lógica de control. Enfatiza una separación entre la lógica de negocios y su visualización. Esta "separación de preocupaciones" proporciona una mejor división del trabajo y una mejora de mantenimiento. Algunos otros patrones de diseño se basan en MVC, como MVVM (Modelo-Vista-modelo de vista), MVP (Modelo-Vista-Presentador) y MVW (Modelo-Vista-Whatever). (26)

Las tres partes del patrón de diseño de software MVC se pueden describir de la siguiente manera:

1. Modelo: Maneja datos y lógica de negocios.
2. Vista: Se encarga del diseño y presentación.
3. Controlador: Enruta comandos a los modelos y vistas.

Modelo

El modelo (a veces denominado "objeto de modelo") contiene todo el contenido y la lógica de procesamiento específicos de la aplicación, incluso todos los objetos de contenido, acceso a fuentes de datos o información externos y todas las funciones de procesamiento que son específicas de la aplicación. El cual se pone de manifiesto en el sistema en lo que sería la parte de la base de datos.

Vista

La vista contiene todas las funciones específicas de la interfaz y permite la presentación de contenido y lógica de procesamiento, incluidos todos los objetos de contenido, el acceso a fuentes de datos o información del exterior y todas las funciones de procesamiento que requiere el usuario final. Lo que sería en la parte visual e interfaz del proyecto.

Controlador

El controlador administra el acceso al modelo y la vista, y coordina el flujo de datos entre ellos. En una *webapp* la vista es actualizada por el controlador con datos del modelo, basándose en las entradas que da el usuario. Las clases controladoras en el trabajo son las encargadas de llevar esta lógica.

2.4 Patrones de Diseño

Los patrones de diseño o *design patterns*, son una solución general, reutilizable y aplicable a diferentes problemas de diseño de software. Se trata de plantillas que identifican problemas en el sistema y proporcionan soluciones apropiadas a problemas generales a los que se han enfrentado los desarrolladores durante un largo periodo de tiempo, a través de prueba y error. (26)

Patrones Generales de Software para la Asignación de Responsabilidades (GRASP)

Los patrones generales de software para asignar responsabilidades del inglés (General Responsibility Assignment Software Patterns) describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones (27). Entre los patrones GRASP a utilizar se encuentran:

Controlador: es un patrón que sirve como intermediario entre una interfaz y el algoritmo que la implementa, de tal forma que recibe los datos del usuario y los envía a las distintas clases según el método llamado. Este patrón sugiere que la lógica de negocio debe estar separada de la capa de presentación, para aumentar la reutilización de código y a la vez tener un mayor control. En este caso se evidencia la utilización de este patrón en la clase `MainController` en la cual se encuentran los métodos que se encargan de la lógica del negocio.

Alta cohesión: la cohesión es una medida de cuán relacionada y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme. Este se evidencia en la clase `MainController`.

Bajo acoplamiento: este patrón es un principio que asigna la responsabilidad de controlar el flujo de eventos del sistema, a clases específicas. Esto facilita la centralización de actividades. El controlador no realiza estas actividades, las delega en otras clases con las que mantiene un modelo de alta cohesión. Esto se evidencia en la clase MainController.

Creador: guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. Este patrón se usa con el objetivo de que la clase que maneja la información de los nuevos objetos a crear sea la responsable de crearlos. Este se evidencia en la clase MainController.

2.5 Conclusiones Parciales

El análisis de la propuesta de solución que se pretende concretar permitió, con el levantamiento de los requisitos del sistema, determinar las funcionalidades básicas a desarrollar durante el proceso; definiendo 40 HU. El análisis y diseño permitió seleccionar el patrón arquitectónico Modelo Vista Controlador, el cual permite el fácil manejo de errores. Al emplear el lenguaje UML para modelar los artefactos propuesto por la metodología de desarrollo seleccionada durante el capítulo 1 se alcanzó una mejor comprensión sobre el cómo debe comportarse el portal web.

Capítulo 3. Implementación y evaluación de la solución propuesta.

En este capítulo se abordarán las diferentes pruebas realizadas al software para determinar así el correcto funcionamiento y la calidad de la propuesta que se ha desarrollado. También se describirá el estándar de codificación utilizado y se detallan los diferentes resultados arrojados por cada prueba realizada y la respuesta a dichas no conformidades.

3.1 Modelo de Despliegue

El diagrama de despliegue se utiliza para modelar la disposición física de los componentes que integran al sistema. Este diagrama describe el despliegue físico de información generada por el programa de software en los componentes de hardware (28). A continuación, se presenta el diagrama de despliegue de la solución.

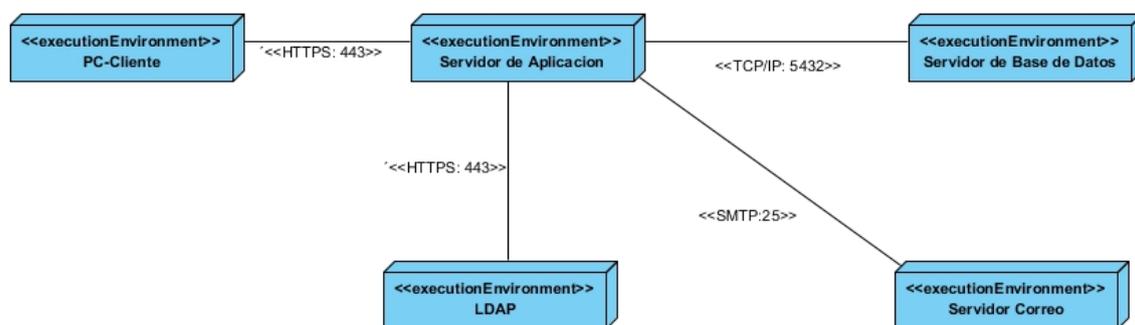


Figura 1 Diagrama de Despliegue. (Fuente: Elaboración Propia)

LDAP: Servidores que brinda la posibilidad de comprobar las credenciales de los usuarios que se autentican en el sistema.

Servidor Correo: Este servidor es el encargado de la gestión de notificaciones a través del correo electrónico.

PC_Cliente: Es la estación de trabajo cliente que presenta un navegador web para conectarse a la aplicación hospedada en el servidor de aplicaciones utilizando el protocolo de comunicación HTTPS.

Servidor de Aplicación: Es la estación de trabajo que hospeda el código fuente de la aplicación, y que les brinda a los usuarios las interfaces de la misma para realizar los procesos definidos por cada uno de los roles del sistema. Esta estación se comunica con el servidor de base de datos donde se almacenan los datos de la aplicación realizando la comunicación mediante el protocolo TCP/IP. Además, se comunica con el

servidor LDAP que le brinda la posibilidad de comprobar las credenciales de los usuarios que se autentican en el sistema.

Servidor de Base de Datos: Este servidor es el encargado del almacenamiento de los datos del sistema, y se comunica con el servidor de aplicaciones de dicho sistema.

3.2 Estándares de Codificación

Los estándares de codificación constituyen un principio esencial en el desarrollo de software. Garantizan que el código obtenido sea fácil de leer, entendido y modificado independientemente de quién haya sido el desarrollador del producto. Son una guía para el equipo de desarrollo, permiten asegurar que el código presente calidad y no contenga errores. (29) En este momento WordPress tiene cuatro guías, una para cada tecnología web que se utiliza: PHP, HTML, JavaScript y CSS. Forman parte de un conjunto más amplio de conocimientos, el Manual Core. A continuación, se destaca lo más importante, en lo que los usuarios fallan al momento de programan para Wordpress. (29)

PHP

PHP es el lenguaje principal de WordPress, y es un lenguaje bastante conocido, lo cual es propicio para ser usado en diferentes proyectos web.

Estilos En Bloques

Llaves de partida inicio deben colocarse al final de las líneas. Si existe una nueva condicional, esta debe añadirse justo después del cierre de la anterior condicional, justo de la siguiente manera:

```
[php]if ( condition ) {  
    // Do Something  
} elseif ( condition ) {  
    // Do Something  
} else {  
    // Do Something  
} [/php]
```

Figura 2 Estilo en Bloques

Uso De Espacio

Reducir el tamaño de los archivos de código es algo que beneficia cuestiones de velocidad, aunque esto es útil en ocasiones, hay otros escenarios en los que el uso de espacio es bastante útil para la lectura del código:

```
[php]

function my_function( $complete_array = null, $key_1 = 4, $key_2 = 'bar' ) {
    if ( null == $complete_array ) {
        $final_array = $complete_array;
    } else {
        $key_1 = (integer) $key_1;
        $final_array[0] = 'this';
        $final_array[ $key_1 ] = 'is';
        $final_array[ $key_2 ] = 'an';
        $final_array['last'] = 'example';
    }
    return $final_array;
}

[/php]
```

Figura 3 Uso de Espacio

Convenciones De Nomenclatura

Éste puede ser difícil acostumbrarse, especialmente si vienes de otros lenguajes de programación. En pocas palabras:

- Los nombres de variables deben estar en minúsculas, palabras separadas por guiones bajos.
- Los nombres de clase deben usar palabras en mayúsculas separadas por guiones bajos. Los acrónimos deben estar en mayúsculas.
- Constantes deben estar en mayúsculas, separadas por guiones.
- Los nombres de archivo deben estar en minúsculas, separados con guiones.

Condiciones Yoda

Escribir condiciones al revés te podría ayudar a prevenir errores de análisis. Puede ser un poco raro, pero es mejor para el código.

```
[php]if ( 'alex' === $name ) {  
echo 'Podría escribir un artículo';  
}  
[/php]
```

Figura 4 Condición Yoga

HTML

HTML no tiene que bastantes reglas asociadas a él, puede llegar a mucho para hacer las cosas más modulares. Sólo hay cinco reglas que usted necesita saber la hora de escribir HTML:

- El código debe validar contra el validador del W3C.
- Las etiquetas HTML de cierre automático deben tener exactamente un espacio antes de la barra diagonal (es una especificación del W3C, no un motivo favorito de WordPress)
- Los atributos y etiquetas deben estar en minúsculas. La única excepción es cuando los valores de atributos están destinados para el consumo humano, en cuyo caso deben ser escritos de forma natural.
- Todos los atributos deben tener un valor y deben ser citados (<input disabled> No es la escritura correcta)
- La sangría debe lograrse mediante pestañas y debe seguir la estructura lógica.

CSS

CSS es otro lenguaje de etiquetado bastante flexible para escribir, por lo que hay un montón de trabajo por hacer aquí. Aun así, las buenas prácticas son bastante fáciles al momento de codificar.

Selectores

Los selectores deben ser tan cualificados como sea necesario, ser humanamente legible, estar en minúsculas con palabras separadas con guiones y selectores de atributos deben utilizar comillas dobles. He aquí un ejemplo justo de lo que debes hacer:

```
[css]input[type="text"],
input[type="password"],
.name-field {
background: #f1f1f1;
} [/css]
```

Figura 5 Selectores

Orden de propiedad

Las reglas reconocen la necesidad de un poco de espacio personal aquí, ya que no prescriben un orden específico de reglas CSS. Lo que no dicen es que debes seguir una estructura semántica que tenga sentido. Las propiedades del grupo por sus relaciones o agruparlos alfabéticamente, simplemente no las escriben de forma aleatoria.

El agregar las propiedades se basa en la lógica en la que se desee visualizar el contenido:

- Visualización
- Posicionamiento
- Modelo de caja
- Los colores y la tipografía
- Otros

```
[css].profile-modal {
display: block;
position: absolute;
left: 100px;
top: 90px;
background: #ff9900;
color: #fff;
} [/css]
```

Figura 6 Orden de prioridad

Javascript

Javascript se ha convertido en un lenguaje bastante usado en la web, en WordPress no ha sido la excepción para ser usado. Para usar Javascript en WordPress, debes basarte

en la guía de jQuery, en la cual se refieren a la longitud de una línea de programación, la cual no debe ser extensa.

Punto Y Coma

error de programación

Es algo que sucede y es bastante común, tal vez sea la regla más sencilla es la que se pasa por alto con mayor frecuencia. Nunca, jamás, debes omitir un punto y coma, por la sencilla razón que tu código no podría funcionar.

Asegúrate de que tu código tenga un punto y coma al final de cada sentencia válida, recuerda que los corchetes no llevan el punto y coma al final.

Sangría

Las tabulaciones siempre se deben utilizar sangría. También debes añadir espacio en el contenido de un cierre, incluso si el contenido de un archivo está contenido en uno.

Líneas De Ruptura

Limitar hasta donde llega determinada instrucción, sirve para conocer en qué punto termina y comienza un nuevo bloque de código. Con esto también se ahorra el dolor de cabeza que podría dar el punto y coma.

Ciclos Con jQuery

De acuerdo con las normas de iteración con jQuery (jQuery.each ()) sólo debe usarse en objetos jQuery. Puedes usar los típicos ciclos de for/in para iterar otros tipos de datos.

Finalmente

La consistencia es la regla más importante. Es mejor desarrollar bastante de cerca con los estándares, y no tener que cambiar a mitad del desarrollo del proyecto una gran cantidad de líneas de código. (30)

3.3 Aplicación de la estrategia de validación del portal web de la Facultad 4

El único instrumento adecuado para determinar el status de la calidad de un producto de software es el proceso de pruebas. En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema de software en su totalidad, con el objetivo de medir el grado en que cumple con los requerimientos. (31)

La Junta Internacional de Calificación de Pruebas de Software (ISTQB por sus siglas en inglés) define las pruebas de software, como el proceso que consiste en todas las actividades del ciclo de vida, tanto estáticas como dinámicas, relacionadas con la planificación, preparación y evaluación de productos de software y productos

relacionados con el trabajo para determinar que cumplen los requisitos especificados, demostrar que son aptos para el propósito y para detectar defectos. (32)

Las pruebas se realizaron a nivel de sistema, ejecutando evaluaciones dinámicas. Teniendo en cuenta la tipología de requisitos pactada con el cliente, atendiendo los elementos planteados en la plantilla Plan de Pruebas del LPS de la UCI (33) y a partir de las características de calidad identificadas en la norma ISO/IEC 25010, *Software Product Quality Requirements and Evaluation (SquaRE)*, se describen a continuación los tipos de pruebas realizados:

Tabla 6 Especificación de pruebas

Características calidad	Tipos de pruebas del sistema
Funcionalidad	Consisten en la revisión de los requisitos aceptados por el cliente a partir de las funcionalidades presentes en la aplicación
Rendimiento	Se diseña para poner a prueba el rendimiento del software en tiempo de ejecución, dentro del contexto de un sistema integrado.
Usabilidad	Prueba enfocada a factores humanos, estéticos, consistencia en la interfaz de usuario
Seguridad	Asegurar que los datos o el sistema solamente son accedidos por los actores definidos según niveles de acceso
Portabilidad	Enfocada a asegurar que funciona en diferentes configuraciones de hardware y software

Para verificar el cumplimiento de los RnF descritos fue diseñada una lista de chequeo, identificándose indicadores a chequear para la evaluación de cada requisito. En el caso de los requisitos de eficiencia las pruebas se realizarán en el entorno real de la aplicación por lo que no se mostrarán resultados de estas como parte de la investigación.

Montar el entorno de pruebas apropiado garantiza la correcta ejecución de las mismas. En la figura 7 puede observarse los elementos que conforman el entorno de pruebas montado para llevar a cabo las pruebas a la herramienta propuesta.



Figura 7 Entorno de pruebas

3.2.1. Diseño de los escenarios de prueba

Los casos de prueba son la forma de verificar las diversas funcionalidades existentes en un producto de software descritas en el formato de las historias de usuarios. Estos se realizan con el objetivo de conseguir un margen de confianza aceptable, de que serán encontrados todos los defectos existentes. Para el diseño de los mismos se aplicaron las técnicas de *caja negra*, *valores límites* y *partición equivalente*. Para el diseño de casos de prueba se utilizaron como entradas las historias de usuario.

En cada CP se describen las variables y se especifican los límites a considerar para los valores de prueba. En la figura 8 puede observarse el DCP Autenticar usuario.

Tabla 7 Diseño del Caso de Prueba Autenticar Usuario. (Fuente: Elaboración propia)

Escenario	Descripción	Variable 1	Variable 2	Respuesta del sistema	Flujo central
RF1					
Condiciones de ejecución					
El usuario debe estar registrado en el sistema.					
SC <Autenticar usuario>					
EC 1.1 Autenticar usuario con datos válidos	El usuario se autentica colocando los datos válidos	usuario rmijares	contraseña Royl@n4*	El sistema permite la entrada del usuario	Entrar a la url del sistema
EC 1.2 Autenticar usuario con datos inválidos	El usuario se autentica colocando los datos inválidos	usuario rmijares	contraseña hola	El sistema no permite la entrada del usuario. Muestra un mensaje "Usuario o contraseña no válidos"	Entrar a la url del sistema
EC 1.3 Autenticar usuario con información incompleta	El usuario se autentica sin colocar toda la información	usuario	contraseña Royl@n4*	El sistema no permite la entrada del usuario. Muestra un mensaje "Existen campos vacíos"	Entrar a la url del sistema

[Las celdas de la tabla contienen V, I, o N/A. V indica válido, I indica inválido, y N/A que no es necesario proporcionar un valor del dato en este caso, ya que es irrelevante.]

Para verificar el cumplimiento de los RnF descritos fue diseñada una lista de chequeo, identificándose indicadores a chequear para la evaluación de cada requisito.

3.2.2. Resultado de las pruebas realizadas

Teniendo en cuenta las condiciones de la aplicación, se realizan diversas pruebas de las cuales se exponen sus resultados:

Pruebas de funcionalidad:

Se denominan pruebas funcionales a las pruebas de software que tienen como objetivo fundamental asegurar el trabajo apropiado de los requisitos funcionales, incluyendo la navegación, entrada de datos, procesamiento y obtención de resultados. (32)

Las metas de esta prueba son:

- Verificar el procesamiento, recuperación e implementación adecuada de las reglas del negocio.
- Verificar la apropiada aceptación de datos

Para la realización de estas pruebas se utilizó el método de caja negra haciendo uso de la técnica partición de equivalencia que se encarga de examinar los valores válidos e ***inválidos de las entradas existentes en el sistema.***

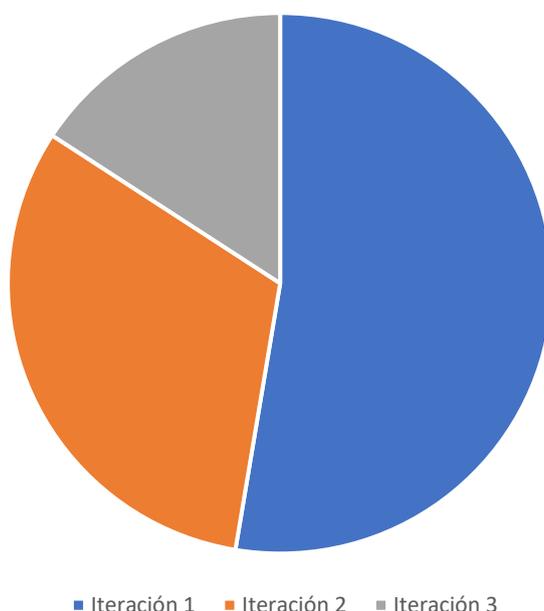


Figura 8 Cantidad de defectos por iteración. (Fuente: Elaboración propia)

Se aplicaron los casos de pruebas correspondientes a las funcionalidades del sistema durante tres iteraciones de pruebas, se detectaron en cada una de estas 10, 6 y 3 no

conformidades respectivamente, llegando al total de 19 no conformidades, cabe añadir que de la primera iteración quedaron dos no conformidades sin resolver las cuales se adicionaron a la segunda iteración y en esta iteración quedó una, la cual se adicionó a la tercera.

Las pruebas realizadas revelaban errores en la documentación, codificación y en el diseño de las interfaces de usuario que en la mayoría de los casos no incidían significativamente en las respuestas esperadas. Todas estas dificultades fueron solucionadas en el menor tiempo posible, siempre antes de pasar a la iteración siguiente. Una vez llegada a la solución final, se le demostró al cliente el cumplimiento de todos los requisitos solicitados.

Pruebas de Rendimiento

La prueba de rendimiento se diseña para poner a prueba el rendimiento del software en tiempo de ejecución, dentro del contexto de un sistema integrado. Esta prueba ocurre a lo largo de todos los pasos del proceso de prueba. Incluso en el nivel de unidad, puede accederse al rendimiento de un módulo individual conforme se realizan las pruebas. Sin embargo, no es sino hasta que todos los elementos del sistema están plenamente integrados cuando puede determinarse el verdadero rendimiento de un sistema. (28)

Los tipos de pruebas de rendimiento que se le realizaron al portal web fueron las pruebas de carga y estrés y se utilizó como herramienta el Apache JMeter en su versión 2.10, diseñado para pruebas de carga de comportamientos funcionales y la medición del rendimiento. El ambiente de prueba estuvo conformado por:

Hardware de prueba (PC servidor):

- ✓ Sistema Operativo: Linux Mint 18
- ✓ Microprocesador: Intel(R) Core(TM) i5-2430M CPU @2.40GHz 2.40GHz
- ✓ Memoria RAM: 4.00 GB
- ✓ Disco Duro: 1024 GB

Software instalado en la PC:

- ✓ Tipo de Servidor web: Apache 2.4
- ✓ Plataforma: SO Linux
- ✓ Servidor de Base de Datos: MySQL 8.0

Luego de definido el hardware se configuran los parámetros del Apache JMeter logrando un ambiente de simulación con un total de 50 y 100 usuarios conectados concurrentemente en una primera y segunda prueba respectivamente, se realizan peticiones a diferentes páginas del portal web para la Facultad 4.

Análisis de los resultados de las pruebas de rendimiento

Para un mejor entendimiento de las pruebas de rendimiento, se explica cada parámetro que la compone a continuación:

- ✓ **Usuarios:** total de usuarios.
- ✓ **# Muestras:** el número de peticiones.
- ✓ **Media:** El tiempo medio transcurrido en milisegundos para un conjunto de resultados.
- ✓ **Mín:** El mínimo tiempo transcurrido en milisegundos para las muestras de la URL dada.
- ✓ **Máx:** El máximo tiempo transcurrido en un milisegundo para las muestras de la URL dada.
- ✓ **% Error:** Porcentaje de las peticiones con errores.
- ✓ **Rendimiento:** Rendimiento medido en base a peticiones por segundo/minuto/hora.
- ✓ **Kb/s Recibidos:** Rendimiento medido en Kbytes por segundos.

Tabla 8 Resumen de los resultados de las pruebas de rendimiento. (Elaboración Propia)

Usuarios	Muestras	Media	Min	Max	%Error	Rendimiento (Peticiones/Segundo)	Kb/s Recibidos
50	200	5039	1006	15528	0%	6,8/sec	22.4
100	400	3020	1186	5258	0%	8,5/sec	66.1

De los resultados obtenidos en las pruebas se determinó que el portal cumple con los requisitos no funcionales (software, hardware y eficiencia) definidos en el levantamiento realizado con el cliente. Para ello se realizó una primera prueba donde para un total de 50 usuarios conectados al portal de forma concurrente, el mismo respondió 200

peticiones al servidor en un promedio de 5039 segundos, se obtuvo un tiempo mínimo de 1,006 segundos al cargar la página y 15,258 segundos como tiempo máximo, para un porcentaje de error de 0,00 %.

Para reducir el tiempo de respuesta del servidor, se realizaron diferentes acciones. Se especificó el tiempo mínimo de permanencia en caché y la caducidad de las páginas en caché. Se optimizaron los ficheros CSS y JavaScript, quitando espacios y nuevas líneas; almacenando la información en un solo fichero. De esta forma se reduce el tamaño al mínimo posible, mejorando así la velocidad de carga de la web. También se activó la compresión de CSS y JavaScript en la sección «Optimización de ancho de banda» pues permite comprimir los recursos de la página en un solo archivo. Se definió un estilo específico a cada tipo de imagen pues el peso de las mismas influye en la velocidad de carga del portal web.

En una segunda prueba se obtuvo un menor tiempo de respuesta del servidor para 100 usuarios conectados concurrentemente en un intervalo de 8,5 de peticiones por segundos, se obtuvo un tiempo mínimo de 1,18segundos en cargar la página para un tiempo máximo de 5,28 segundos, con un porcentaje de error de 0,00 % para un total de 400 peticiones al servidor. Atendiendo a la cantidad de peticiones por cada segundo que se enviaron, las prestaciones del hardware donde se realizaron las pruebas se considera que constituye un resultado satisfactorio.

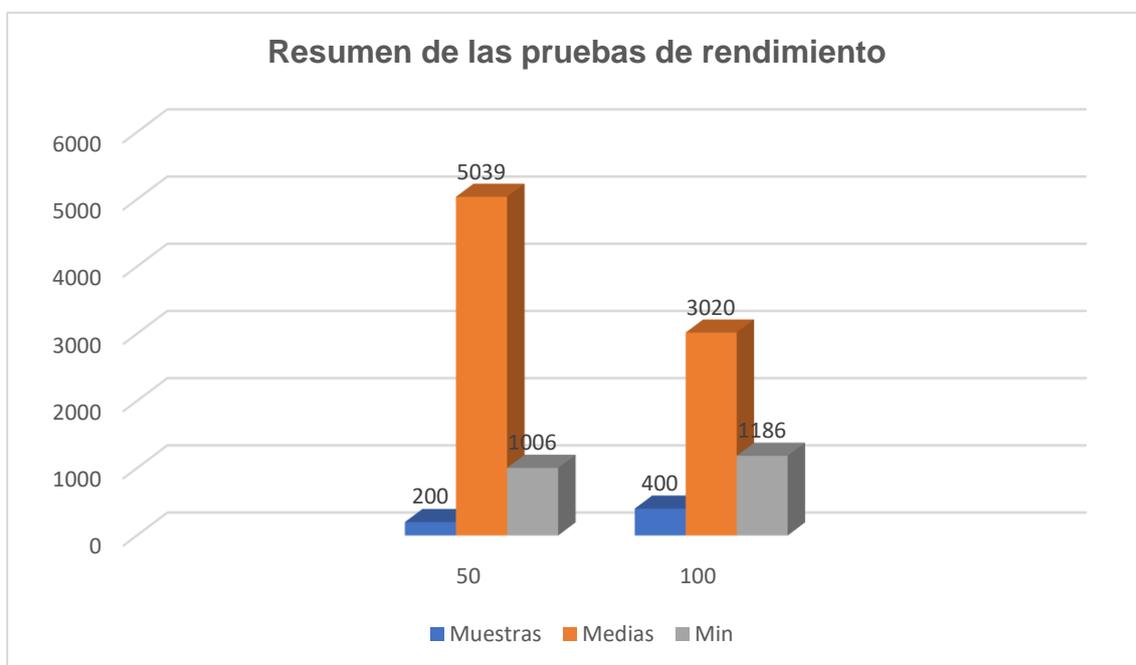


Figura 9 Resumen de las pruebas de rendimiento (Fuente: Elaboración propia)

Con el propósito de mostrar una visión objetiva de las pruebas realizadas, se analizan los resultados obtenidos teniendo en cuenta los indicadores definidos en el informe de

análisis de tendencias del expediente de proyecto de la UCI en su versión 5.0: estado del plan de pruebas, defectos por iteraciones, defectos por tipo de prueba, defectos por tipo de error.

- Estado del plan de pruebas: En este caso se evalúa el cumplimiento de las actividades planificadas en la estrategia de pruebas (tipos de pruebas realizadas, actividades del cronograma de pruebas).
- Para ellos el 100 por ciento de los tipos de pruebas planificados fueron ejecutados.
- Defectos por iteraciones: Se analiza la cantidad de defectos detectados en cada iteración de pruebas. En la figura 9 pueden observarse los datos, en la cual se aprecia que en la primera iteración se detectó mayor cantidad de defectos.
- Defectos por tipo de prueba: Se analiza la cantidad de defectos detectados en cada tipo de prueba realizada. En la figura 11 se muestran los resultados.

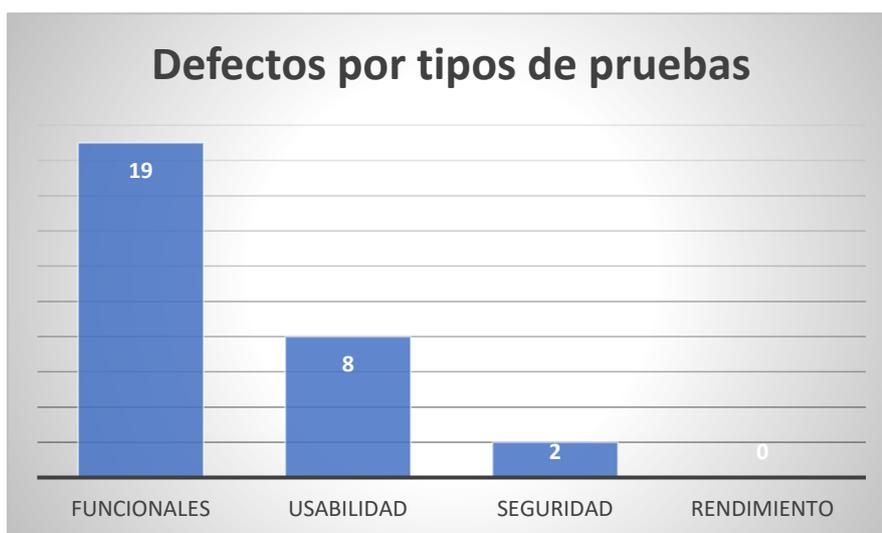


Figura 10 Defecto por tipo de prueba. Fuente: Elaboración propia)

El tipo de pruebas donde más defectos fueron identificados fueron las pruebas funcionales. En el caso de las pruebas no funcionales se detectaron más defectos en las pruebas de usabilidad.

- Defectos por tipo de error según su clasificación: se presentan los tipos de defectos detectados según las clasificaciones. En la figura 12 se muestran los resultados relacionados con el indicador.

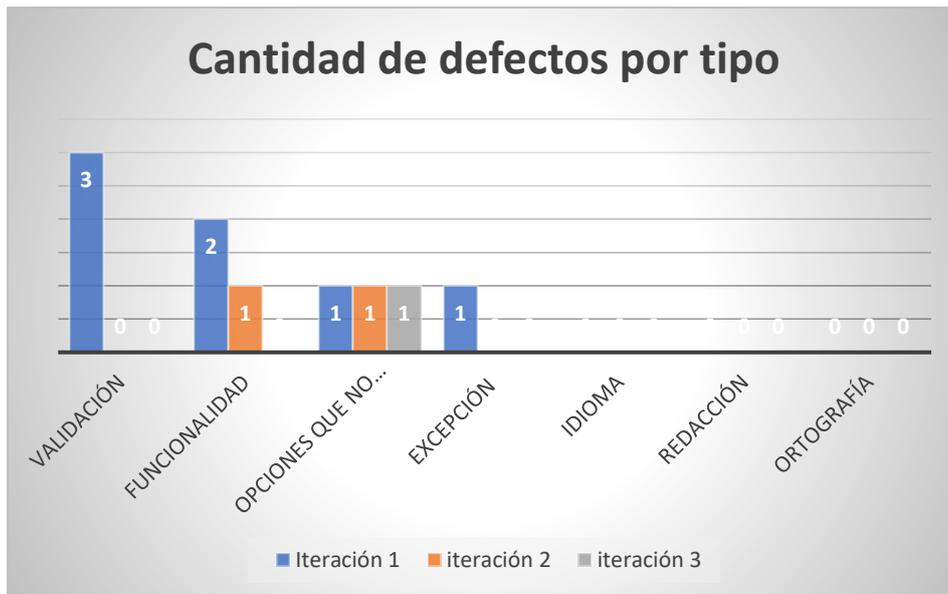


Figura 11 Defecto por tipo de prueba. Fuente: Elaboración propia)

Puede observarse que la mayor cantidad de defectos detectados son de validación. Este resultado tuvo utilidad durante la ejecución de la segunda iteración pues se hizo mayor énfasis en la verificación de este tipo de error.

3.3 Conclusiones Parciales

Al Concluir este capítulo se termina la realización de la aplicación, obteniendo los siguientes resultados:

- El diagrama de despliegue mostró los nodos procesadores la distribución de los procesos y de los componentes.
- La realización de las pruebas al sistema desarrollado, permitió detectar y corregir las no conformidades presentes en el mismo por tanto los resultados obtenidos son satisfactorios.

Conclusiones Generales

Luego de culminada la presente investigación se puede arribar a las siguientes conclusiones:

- El estudio de los referentes teóricos logró un mejor entendimiento de los conceptos asociados a la evolución de la web, difusión de la información y portales web.
- El análisis de las diferentes herramientas y tendencias para la realización de portales web permitió determinar la no existencia de un portal web que responda a las necesidades requeridas por el cliente.
- La implementación del sistema a través de las herramientas y lenguajes seleccionados permitió obtener un portal web capaz mejorar la difusión de la información asociada a las áreas y procesos de la Facultad 4 en la Universidad de las Ciencias Informáticas.
- El diseño de la propuesta de solución permitió generar los artefactos más significativos de acuerdo con la metodología de desarrollo de software AUP-UCI tomándose como referencia los requisitos detectados.
- Las definiciones de las necesidades del cliente a través de las historias de usuario, propició el funcionamiento adecuado del portal de la Facultad 4.
- Las técnicas de validación aplicadas a la propuesta de solución permitieron la detección y corrección de las no conformidades detectadas y evidenciaron que el sistema constituye una solución funcional.

Recomendaciones

Una vez concluida la investigación y el desarrollo de la propuesta de solución se recomienda:

- Continuar desarrollando funcionalidades que sean requeridas en el futuro para asegurar la disponibilidad y usabilidad, así como la satisfacción de los usuarios potenciales.

Bibliografía

1. Carbó. Cuba y el impacto de las TIC en la informatización de la sociedad. 2017.
2. Chirino, Alejandro Alvarez. Portal web de la Federación Estudiantil Universitaria en la Universidad de Ciencias Informáticas. La Habana : s.n., 2019.
3. Docs, MDN Web. ¿Cuál es la diferencia entre la página web, el sitio web, el servidor web y el motor de búsqueda? [En línea] 25 de abril de 2022. https://developer.mozilla.org/es/docs/Learn/Common_questions/Pages_sites_servers_and_search_engines.
4. Blasco, Lourdes Castillo. Direcciones de internet de interés para Biblioteconomía y Documentación. [En línea] 2005. <https://www.uv.es/macas/T6.pdf>.
5. Martínez-López, Francisco. Online brand communities. Online brand communities. 2016.
6. Felipe. Evolución de la Web a lo largo del tiempo: web 1.0, 2.0, 3.0 y 4.0. [En línea] 18 de junio de 2020. <https://www.hostingplus.pe/blog/evolucion-de-la-web-a-lo-largo-del-tiempo-web-1-0-2-0-3-0-y-4-0/>.
7. Santander. Santander Becas. [En línea] 20 de 12 de 2022. <https://www.becas-santander.com/es/blog/metodologias-desarrollo-software.html>.
8. INC, Digite. Digite INC. [En línea] 2022. <https://www.digite.com/es/agile/programacion-extrema-xp/>.
9. Sanchez, Tamara Rodriguez. Metodología de Desarrollo para la Actividad Productiva UCI.
10. Knowledge Base. ¿Qué Es un Sistema de Gestión de Contenidos (CMS)? 2020.
11. Gimenez, Monica. Hiberus Blog. [En línea] 2021. <https://www.hiberus.com/crecemos-contigo/que-es-drupal/>.
12. RODRÍGUEZ, L.M. Wordpress. Obtenido de freidercreativo. 2021.
13. RODAS, A., RÍOS, J. I. y SOLARTE, G. R. LUCIDCHART. [En línea] 2020. <https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml>.
14. Mi Formación Gratis. [En línea] 3 de 3 de 2022. <https://www.miformaciongratis.com/blog-post/html5-que-es-y-para-que-sirve/>.
15. OpenWebinars.net. [En línea] <https://openwebinars.net/blog/que-es-css3/>.

16. Roy. Amazon.com. [En línea] 2022. <https://aws.amazon.com/es/what-is/javascript/>.
17. Silva, Avila y Elena, Alicia. SILVA, A., et al. Utilidad del Lenguaje Unificado de Modelado (UML) en el desarrollo de software profesional dentro del sector empresarial y educativo. [En línea] 2018.
18. Souza, Ivan de. RockContent Blog. [En línea] 2020. <https://rockcontent.com/es/blog/php/>.
19. NIÑO, Jesus. Servidores de aplicaciones web (Aplicaciones web). s.l. : Editex, 2011.
20. Arias, Miguel Angel. Aprende Programación Web con PHP y MySQL: 2ª Edición. [En línea] 2017.
21. Mendoza, Dayana Peña. Extensión de la herramienta Visual Paradigm for UML para la evaluación . [En línea] 2016.
22. Oliva, Adrian Trujillo, Morales, Yanet Rosales y Clarck, Michel Eduardo Marrero. Extensión de la herramienta Visual Paradigm para la generación de las clases de acceso a datos con Doctrine 2.0. [En línea] 2015.
23. PRESSMAN, Roger S. Ingeniería del software Un enfoque práctico, Séptima Edición. [En línea] University of Connecticut, 2010.
24. Vera, Raúl Antonio Aguila. Ingeniería de Software en México: Educación, Industria e Investigación. [En línea] 2019. <http://amexcomp.mx/files/Aguilar-LibroISW-ISBN>.
25. KRUCHTEN, Philippe. Planos Arquitectónicos: El Modelo de 4+ 1 Vistas de la Arquitectura del Software. 1995.
26. Canelo, Miriam Martínez. Profile. [En línea] 24 de Junio de 2020. <https://profile.es/blog/patrones-de-diseno-de-software/>.
27. Ortega, Gilberto Andres Vargas. Lineamientos para el diseño de aplicaciones web soportados en patrones GRASP. Ciencia e Ingeniería. 2021, Vol. Vol 8.
28. Pressman, Roger S. Ingeniería del Software. 2010.
29. Cruz, Nicky Alejandro Cortina de la y Zabaleta, Eduard Elias Marquez. Generar código de programación en la etapa de implementación de un software. [En línea] Inka Mall SAC, 2021.
30. HDCO. Estándares De Programación Para WordPress. [En línea] 2015. <https://blog.hostdime.com.co/estandares-de-programacion-para-wordpress-guia/>.

31. Sommerville, Ian. Ingeniería del Software. 2005.
32. Sánchez, Milton Eduardo Escobar. Modelo formal de pruebas funcionales de software para alcanzar en nivel de madurez integrado 2. [En línea] 2015. <http://dialnet.unirioja.es>.
33. Holgado, Alicia Garcia. Modelo del Dominio. Modelo del Dominio. [En línea] 2022.

Anexos

Tabla 9 Cerrar Sesión Fuente: (Elaboración Propia)

Historia de Usuario	
No. 2	Nombre del Requisito: Cerrar Sesión
Programador Responsable: Admin	Iteración Asignada: 1
Prioridad en Negocio: Media	Tiempo Estimado: 5 Seg
Riesgo en Desarrollo: Media	Tiempo Real: 10 Seg
Descripción: El cliente tendrá la opción de cerrar su sesión cuando así lo desee. Una vez que ingrese a esta opción se volverá a la página de autenticación.	
Observaciones: N/A	
Prototipo Interfaz:	

Tabla 10 Asignar rol a un usuario Fuente: (Elaboración Propia)

Historia de Usuario	
No. 3	Nombre del Requisito: Asignar Rol a Usuario
Programador responsable: Admin	Iteración Asignada: 1
Prioridad en Negocio: Media	Tiempo Estimado: 1 Hr
Riesgo en Desarrollo: Media	Tiempo Real: 1 Hr
Descripción: El Admin podrá asignar al usuario los roles de suscriptor, autor, editor	
Observaciones: Todos los campos deben llenarse, en caso de lo contrario el sistema emite un error.	
Prototipo Interfaz:	

Tabla 11 Añadir Noticia Fuente: (Elaboración Propia)

Historia de Usuario	
No. 4	Nombre del Requisito: Añadir Noticia
Programador Responsable: Admin	Iteración Asignada: 1
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr

Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite añadir información referente a una noticia	
<p>Observaciones: En el caso añadir noticia el sistema permite que los usuarios con roles de administrador y colaborador añadan las noticias.</p> <ul style="list-style-type: none"> • Título (obligatorio, campo de texto) • Fuente de publicación (opcional, campo de texto) • Fecha de publicación (obligatorio, calendario) • Descripción (obligatorio, campo de texto múltiples) • Categoría (obligatorio, selección) • Autor (obligatorio, campo de texto) • Imagen (obligatorio, campo de imagen) • Tipo de noticia (obligatorio, selección) 	
Prototipo Interfaz:	

Tabla 12 Editar Noticia Fuente: (Elaboración Propia)

No. 5	Nombre del Requisito: Editar Noticia	
Programador Responsable: Admin	Iteración Asignada: 2da	
Prioridad en Negocio: Media	Tiempo Estimado: 2hr	
Riesgo en Desarrollo: Media	Tiempo Real: 2hr	
Descripción: El sistema permite modificar información referente a una noticia		
<p>Observaciones: En el caso editar noticia el sistema permite que los usuarios con roles de administrador y colaborador editen las noticias.</p> <ul style="list-style-type: none"> • Título (obligatorio, campo de texto) • Fuente de publicación (opcional, campo de texto) • Fecha de publicación (obligatorio, calendario) • Descripción (obligatorio, campo de texto múltiples) • Categoría (obligatorio, selección) • Autor (obligatorio, campo de texto) • Imagen (obligatorio, campo de imagen) • Tipo de noticia (obligatorio, selección) 		
Prototipo Interfaz:		

Tabla 13 Eliminar Noticia Fuente: (Elaboración Propia)

Historia de Usuario	
No. 6	Nombre del Requisito: Eliminar Noticia
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Baja	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Baja	Tiempo Real: 2 Hr
Descripción: El sistema permite eliminar información referente a una noticia	
Observaciones: En el caso eliminar noticia el sistema permite que los usuarios con roles de administrador y colaborador eliminen las noticias.	
Prototipo Interfaz:	

Tabla 14 Mostrar Noticia Fuente: (Elaboración Propia)

Historia de Usuario	
No. 7	Nombre del Requisito: Mostrar Noticia
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrara todos los usuarios información referente a una noticia.	
Observaciones: En el caso mostrar noticia el sistema permite que todos los usuarios observen las noticias añadidas por un administrador.	
Prototipo Interfaz:	

Tabla 15 Añadir Encuesta Fuente: (Elaboración Propia)

Historia de Usuario	
No. 8	Nombre del Requisito: Añadir Encuesta
Programador Responsable: Admin	Iteración Asignada: 1
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr

Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite añadir una encuesta	
Observaciones: En el caso añadir encuestas el sistema permite que los usuarios con roles de administrador, editor y colaborador añadan las encuestas.	
Prototipo Interfaz:	

Tabla 16 Editar Encuesta Fuente: (Elaboración Propia)

Historia de Usuario	
No. 9	Nombre del Requisito: Editar Encuesta
Programador Responsable: Admin	Iteración Asignada: 2
Prioridad en Negocio: Media	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Media	Tiempo Real: 2 Hr
Descripción: El sistema permite editar una encuesta	
Observaciones: En el caso editar encuestas el sistema permite que los usuarios con roles de administrador, editor y colaborador editen las encuestas.	
Prototipo Interfaz:	

Tabla 17 Eliminar Encuesta Fuente: (Elaboración Propia)

Historia de Usuario	
No. 10	Nombre del Requisito: Eliminar Encuesta
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Baja	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Baja	Tiempo Real: 2 Hr
Descripción: El sistema permite eliminar una encuesta	
Observaciones: En el caso eliminar noticia el sistema permite que los usuarios con roles de administrador y colaborador eliminen las encuestas.	
Prototipo Interfaz:	

Tabla 18 Mostrar Encuesta Fuente: (Elaboración Propia)

Historia de Usuario	
No. 11	Nombre del Requisito: Mostrar Encuesta
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrara todos los usuarios una encuesta.	
Observaciones: En el caso mostrar encuesta el sistema permite que todos los usuarios observen las encuesta añadidas por un administrador.	
Prototipo Interfaz:	

Tabla 19 Realizar Búsqueda Fuente: (Elaboración Propia)

Historia de Usuario	
No. 12	Nombre del Requisito: Realizar búsqueda
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite todos los usuarios realicen una búsqueda.	
Observaciones: N/A	
Prototipo Interfaz:	

Tabla 20 Añadir comentario Fuente: (Elaboración Propia)

Historia de Usuario	
No. 13	Nombre del Requisito: Añadir Comentario
Programador Responsable: Admin	Iteración Asignada: 1
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite añadir un comentario	

Observaciones: En el caso añadir comentarios el sistema permite que los usuarios después de estar autenticados emitan un comentario.
Prototipo Interfaz:

Tabla 21 Editar Comentario Fuente: (Elaboración Propia)

Historia de Usuario	
No. 14	Nombre del Requisito: Editar comentario
Programador Responsable: Admin	Iteración Asignada: 2
Prioridad en Negocio: Media	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Media	Tiempo Real: 2 Hr
Descripción: El sistema permite editar un comentario	
Observaciones: En el caso editar comentario el sistema permite que los usuarios con roles de administrador, editor y colaborador editen los comentarios.	
Prototipo Interfaz:	

Tabla 22 Eliminar comentario Fuente: (Elaboración Propia)

Historia de Usuario	
No. 15	Nombre del Requisito: Eliminar comentario
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Baja	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Baja	Tiempo Real: 2 Hr
Descripción: El sistema permite eliminar un comentario.	
Observaciones: En el caso eliminar comentario el sistema permite que los usuarios con roles de administrador y colaborador eliminen los comentarios.	
Prototipo Interfaz:	

Tabla 23 Mostrar comentario Fuente: (Elaboración Propia)

Historia de Usuario	
No. 16	Nombre del Requisito: Mostrar comentario
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrara todos los usuarios un comentario.	
Observaciones: En el caso mostrar comentarios el sistema permite que todos los usuarios observen los comentarios.	
Prototipo Interfaz:	

Tabla 24 Añadir aviso Fuente: (Elaboración Propia)

Historia de Usuario	
No. 17	Nombre del Requisito: Añadir Aviso
Programador Responsable: Admin	Iteración Asignada: 1
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite añadir un aviso.	
Observaciones: N/A	
Prototipo Interfaz:	

Tabla 25 Editar aviso Fuente: (Elaboración Propia)

Historia de Usuario	
No. 18	Nombre del Requisito: Editar aviso
Programador Responsable: Admin	Iteración Asignada: 2
Prioridad en Negocio: Media	Tiempo Estimado: 2 Hr

Riesgo en Desarrollo: Media	Tiempo Real: 2 Hr
Descripción: El sistema permite editar un aviso	
Observaciones: En el caso editar comentario el sistema permite que los usuarios con roles de administrador, editor y colaborador editen los avisos.	
Prototipo Interfaz:	

Tabla 26 Eliminar aviso Fuente: (Elaboración Propia)

Historia de Usuario	
No. 19	Nombre del Requisito: Eliminar aviso
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Baja	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Baja	Tiempo Real: 2 Hr
Descripción: El sistema permite eliminar un aviso.	
Observaciones: En el caso eliminar aviso el sistema permite que los usuarios con roles de administrador y colaborador eliminen los avisos.	
Prototipo Interfaz:	

Tabla 27 Mostrar aviso Fuente: (Elaboración Propia)

Historia de Usuario	
No. 20	Nombre del Requisito: Mostrar aviso
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrara todos los usuarios un aviso.	
Observaciones: En el caso mostrar avisos el sistema permite que todos los usuarios observen los avisos.	
Prototipo Interfaz:	

Tabla 28 Añadir evento Fuente: (Elaboración Propia)

Historia de Usuario	
No. 21	Nombre del Requisito: Añadir evento
Programador Responsable: Admin	Iteración Asignada: 1
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite añadir información referente a un evento.	
<p>Observaciones: En el caso añadir evento el sistema permite que los usuarios con roles de administrador y colaborador añadan los eventos.</p> <ul style="list-style-type: none"> • Título (obligatorio, campo de texto) • Fuente de publicación (opcional, campo de texto) • Fecha de publicación (obligatorio, calendario) • Descripción (obligatorio, campo de texto múltiples) • Categoría (obligatorio, selección) • Autor (obligatorio, campo de texto) • Imagen (obligatorio, campo de imagen) • Tipo de evento (obligatorio, selección) 	
Prototipo Interfaz:	

Tabla 29 Editar evento Fuente: (Elaboración Propia)

Historia de Usuario	
No. 22	Nombre del Requisito: Editar evento
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Media	Tiempo Estimado: 2hr
Riesgo en Desarrollo: Media	Tiempo Real: 2hr
Descripción: El sistema permite modificar información referente a un evento.	
<p>Observaciones: En el caso editar evento el sistema permite que los usuarios con roles de administrador y colaborador editen los eventos.</p> <ul style="list-style-type: none"> • Título (obligatorio, campo de texto) • Fuente de publicación (opcional, campo de texto) 	

<ul style="list-style-type: none"> • Fecha de publicación (obligatorio, calendario) • Descripción (obligatorio, campo de texto múltiples) • Categoría (obligatorio, selección) • Autor (obligatorio, campo de texto) • Imagen (obligatorio, campo de imagen) • Tipo de evento (obligatorio, selección)
Prototipo Interfaz:

Tabla 30 Eliminar evento Fuente: (Elaboración Propia)

Historia de Usuario	
No. 23	Nombre del Requisito: Eliminar evento
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Baja	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Baja	Tiempo Real: 2 Hr
Descripción: El sistema permite eliminar información referente a un evento.	
Observaciones: En el caso eliminar evento el sistema permite que los usuarios con roles de administrador y colaborador eliminen los eventos.	
Prototipo Interfaz:	

Tabla 31 Mostrar evento Fuente: (Elaboración Propia)

Historia de Usuario	
No. 24	Nombre del Requisito: Mostrar evento
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrara todos los usuarios información referente a un evento.	
Observaciones: En el caso mostrar evento el sistema permite que todos los usuarios observen los eventos añadidos por un administrador.	

Prototipo Interfaz:

Tabla 32 Añadir imagen de portada Fuente: (Elaboración Propia)

Historia de Usuario	
No. 25	Nombre del Requisito: Añadir imagen de portada
Programador Responsable: Admin	Iteración Asignada: 1
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite al administrados añadir una imagen de portada.	
Observaciones: N/A	
Prototipo Interfaz:	

Tabla 33 Mostrar enlace a redes sociales Fuente: (Elaboración Propia)

Historia de Usuario	
No. 26	Nombre del Requisito: Mostrar enlace a redes sociales
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrara todos los usuarios enlace a las redes sociales.	
Observaciones: En el caso mostrar enlace a redes sociales el sistema permite que todos los usuarios observen los enlaces a redes sociales añadidos por un administrador.	
Prototipo Interfaz:	

Tabla 34 Añadir colección de imágenes Fuente: (Elaboración Propia)

Historia de Usuario	
No. 27	Nombre del Requisito: Añadir colección de imágenes
Programador Responsable: Admin	Iteración Asignada: 1
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite añadir una colección de imágenes.	
Observaciones: N/A	
Prototipo Interfaz:	

Tabla 35 Editar colección de imágenes Fuente: (Elaboración Propia)

Historia de Usuario	
No. 28	Nombre del Requisito: Editar colección de imágenes.
Programador Responsable: Admin	Iteración Asignada: 2
Prioridad en Negocio: Media	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Media	Tiempo Real: 2 Hr
Descripción: El sistema permite editar una colección de imágenes.	
Observaciones: En el caso editar una colección de imágenes el sistema permite que los usuarios con roles de administrador, editor y colaborador editen la colección de imágenes.	
Prototipo Interfaz:	

Tabla 36 Eliminar colección de imágenes Fuente: (Elaboración Propia)

Historia de Usuario	
No. 29	Nombre del Requisito: Eliminar colección de imágenes
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Baja	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Baja	Tiempo Real: 2 Hr
Descripción: El sistema permite eliminar una colección de imágenes.	

Observaciones: En el caso eliminar colección de imágenes el sistema permite que los usuarios con roles de administrador y colaborador eliminen la colección de imágenes.
Prototipo Interfaz:

Tabla 37 Mostrar colección de imágenes Fuente: (Elaboración Propia)

Historia de Usuario	
No. 30	Nombre del Requisito: Mostrar colección de imágenes
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrar a todos los usuarios una colección de imágenes.	
Observaciones: En el caso mostrar colección de imágenes el sistema permite que todos los usuarios observen la colección de imágenes.	
Prototipo Interfaz:	

Tabla 38 Contactar al administrador Fuente: (Elaboración Propia)

Historia de Usuario	
No. 31	Nombre del Requisito: Contactar al administrador
Programador Responsable: Admin	Iteración Asignada: 2da
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite que todos los usuarios contacten al administrador.	
Observaciones: En el caso de contactar al administrador sería por medio del correo electrónico.	
Prototipo Interfaz:	

Tabla 39 Editar página básica Fuente: (Elaboración Propia)

Historia de Usuario	
No. 32	Nombre del Requisito: Editar página básica
Programador Responsable: Admin	Iteración Asignada: 2
Prioridad en Negocio: Media	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Media	Tiempo Real: 2 Hr
Descripción: El sistema permite editar una página básica	
Observaciones: En el caso editar página básica el sistema permite que los usuarios con roles de administrador, editor y colaborador editen las páginas básicas.	
Prototipo Interfaz:	

Tabla 40 Mostrar página básica Fuente: (Elaboración Propia)

Historia de Usuario	
No. 33	Nombre del Requisito: Mostrar página básica
Programador Responsable: Admin	Iteración Asignada: 2
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrar una página básica	
Observaciones: En el caso mostrar página básica el sistema permite que los usuarios observen una página básica.	
Prototipo Interfaz:	

Tabla 41 Filtrar contenido Fuente: (Elaboración Propia)

Historia de Usuario

No. 34	Nombre del Requisito: Filtrar Contenido	
Programador Responsable: Admin		Iteración Asignada: 2
Prioridad en Negocio: Media		Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Media		Tiempo Real: 2 Hr
Descripción: El sistema permite filtrar contenido.		
Observaciones: En el caso filtrar contenido el sistema permiten que todos los usuarios filtren contenidos.		
Prototipo Interfaz:		

Tabla 42 Mostrar enlaces de interés Fuente: (Elaboración Propia)

Historia de Usuario		
No. 35	Nombre del Requisito: Mostrar enlaces de interés	
Programador Responsable: Admin		Iteración Asignada: 2
Prioridad en Negocio: Alta		Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta		Tiempo Real: 2 Hr
Descripción: El sistema permite mostrar enlaces de interés.		
Observaciones: En el caso mostrar enlaces de interés el sistema permite que los usuarios observen enlaces de interés.		
Prototipo Interfaz:		

Tabla 43 Asignar permisos de publicación por sección Fuente: (Elaboración Propia)

Historia de Usuario		
No. 36	Nombre del Requisito: Asignar permisos de publicación en secciones por usuarios	
Programador responsable: Admin		Iteración Asignada: 1
Prioridad en Negocio: Alta		Tiempo Estimado: 1 Hr
Riesgo en Desarrollo: Alta		Tiempo Real: 1 Hr
Descripción: El administrador podrá asignar al usuario los permisos de publicación en secciones por usuarios		

Observaciones: Todos los campos deben llenarse, en caso de lo contrario el sistema emite un error.

Prototipo Interfaz:

Tabla 44 Mostrar información de los directivos de la facultad Fuente: (Elaboración Propia)

Historia de Usuario	
No. 37	Nombre del Requisito: Mostrar información de los directivos de la facultad
Programador Responsable: Admin	Iteración Asignada: 2
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrar información de los directivos de la facultad	
Observaciones: En el caso mostrar enlaces de interés el sistema permite que los usuarios observen la información de los directivos de la facultad.	
Prototipo Interfaz:	

Tabla 45 Mostrar Información de los dirigentes de la FEU Fuente: (Elaboración Propia)

Historia de Usuario	
No. 38	Nombre del Requisito: Mostrar información de los dirigentes de la FEU
Programador Responsable: Admin	Iteración Asignada: 2
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrar información de los dirigentes de la FEU	
Observaciones: En el caso mostrar enlaces de interés el sistema permite que los usuarios observen la información de los dirigentes de la FEU.	
Prototipo Interfaz:	

Tabla 46 Mostrar información de los Centros de Desarrollo de la Facultad Fuente: (Elaboración Propia)

Historia de Usuario	
No. 39	Nombre del Requisito: Mostrar información de los Centros de Desarrollo de la Facultad
Programador Responsable: Admin	Iteración Asignada: 2
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrar información de los Centros de Desarrollo de la Facultad	
Observaciones: En el caso mostrar enlaces de interés el sistema permite que los usuarios observen la información de los Centros de Desarrollo de la Facultad.	
Prototipo Interfaz:	

Tabla 47 Mostrar información de la facultad Fuente: (Elaboración Propia)

Historia de Usuario	
No. 39	Nombre del Requisito: Mostrar información de la Facultad
Programador Responsable: Admin	Iteración Asignada: 2
Prioridad en Negocio: Alta	Tiempo Estimado: 2 Hr
Riesgo en Desarrollo: Alta	Tiempo Real: 2 Hr
Descripción: El sistema permite mostrar información de la Facultad	
Observaciones: En el caso mostrar enlaces de interés el sistema permite que los usuarios observen la información de la Facultad	
Prototipo Interfaz:	