



FACULTAD 1

Portal Web de la Oficina Provincial de la ONAT de Villa Clara

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

Autor: Ciara Ruiz Gutiérrez

Tutores:

Ing. Neimary Pedraza Martínez

Ing. Wendy Muñoz Rodríguez

Ing. Yojahny Chávez Marrero

La Habana, diciembre 2021

“Año 63 de la Revolución”



*El futuro de nuestra Patria tiene que ser
necesariamente un futuro de hombres de ciencia*

DECLARACIÓN DE AUTORÍA

El autor del trabajo de diploma con título **Portal web de la Oficina Provincial de la ONAT de Villa Clara** concede a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la investigación, con carácter exclusivo. De forma similar se declara como único autor de su contenido. Para que así conste firman la presente a ____ días del mes de diciembre del año 2021.

Firma del autor

Ciara Ruiz Gutiérrez

Firma del tutor

Ing. Neimary Pedraza Martínez

Firma del tutor

Ing. Wendy Muñoz Rodríguez

Firma del tutor

Ing. Yojahny Chávez Marrero

DEDICATORIA

A mis padres por ser el regalo más lindo que me ha dado la vida, mi razón de ser, por convertirme en la mujer que soy hoy, por su apoyo incondicional y ayudarme a lograr cada uno de mis sueños.

AGRADECIMIENTOS

A mis padres por ser mi ejemplo, mi guía, por darme siempre los mejores consejos, por enseñarme que las mejores cosas en la vida se logran con empeño, dedicación y esfuerzo propio.

A mis hermanos por demostrarme siempre su amor incondicional, porque tenerlos es lo mejor que me ha pasado en la vida.

A mis tíos Robertico y Yadira por estar presente siempre en mi vida, por apoyarme en cada momento y seguir mi carrera tan de cerca.

A mis abuelos Ofelia, Roberto y Eva por quererme tanto y estar tan pendiente de mí.

A mi chucha por ser mi mejor amiga.

A todos mis amigos en especial a las mimis Claudia, Carolyne, Susan y Yudisley, A mi equipo favorito Victor, Rasiel, Edel, Tania y María por estar presente en los buenos y malos momentos durante toda la carrera, por ser mi familia en la UCI.

A mis tutores Wendy por estar pendiente de mí durante todo el proceso de la tesis a pesar de la distancia, a Neimary por recibirme tan bien desde el principio en la ONAT, a Yojahny por ser mi amigo, por estar siempre presente, por ayudarme en todo momento, porque cada regaño me hizo ser cada vez mejor.

A todas las personas que de una forma u otra formaron parte de esta maravillosa experiencia.

RESUMEN

El conocimiento referente a temas tributarios y legislativos constituye una necesidad tanto en Cuba como en todos los países del mundo. El presente trabajo se encarga de desarrollar un portal web para la Oficina Provincial de la ONAT de Villa Clara, que contribuya a la gestión de información acerca de los trámites que se realizan en la entidad, para facilitar de esa manera la atención a los contribuyentes. Para guiar el proceso institución y portales homólogos existentes en la actualidad para encontrar similitudes entre las características planteadas por la entidad cliente. Además, se identifican las tecnologías y lenguajes a utilizar en el que debe presentar el portal web al terminar su implementación. Para la construcción de la solución se utilizó Drupal como Sistema de Gestión de Contenidos (*CMS*) así como otras herramientas y tecnologías de código abierto. Para verificar el correcto funcionamiento del sistema de realizan las pruebas funcionales, de seguridad, usabilidad y carga y estrés.

Palabras Claves: ONAT, portal web, drupal.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA ACERCA DEL DESARROLLO DEL PORTAL WEB DE LA OFICINA PROVINCIAL DE LA ONAT DE VILLA CLARA	4
1.1 Introducción del capítulo	4
1.2 Categorías conceptuales.....	4
1.3 Entorno actual de los portales web institucionales.....	9
1.4 Tecnologías y Herramientas a utilizar.....	12
1.5 Conclusiones Parciales	20
CAPÍTULO 2: DESCRIPCIÓN DEL PORTAL WEB DE LA OFICINA PROVINCIAL DE LA ONAT DE VILLA CLARA.....	21
2.1 Introducción del capítulo	21
2.2 Propuesta de Solución	21
2.3 Ingeniería de requisitos	22
2.4 Análisis y Diseño.....	34
2.5 Conclusiones Parciales	41
CAPÍTULO 3. IMPLEMENTACIÓN Y VALIDACIÓN DEL PORTAL WEB	42
3.1 Introducción del capítulo	42
3.2 Diagrama de Componentes.....	42
3.3 Estándares de Codificación.....	43
3.4 Pruebas de Software.....	45
3.5 Validación del Portal web	51
3.6 Conclusiones Parciales	53
CONCLUSIONES GENERALES	54
RECOMENDACIONES.....	55
ANEXOS	59

ÍNDICE DE TABLAS

Tabla 1. Comparación de Sistemas	11
Tabla 2. Requisitos Funcionales	22
Tabla 3. Historia de Usuario Registrar Usuario	31
Tabla 4. Historia de Usuario Añadir Noticia	32
Tabla 5. Historia de Usuario Añadir Documento.....	33
Tabla 6. Caso de Prueba basado en el requisito Añadir Documento.....	46
Tabla 7. Caso de Prueba de Aceptación Añadir Documento.....	47
Tabla 8. Cuadro Lógico de Iadov para usuarios potenciales	51
Tabla 9. Satisfacción de usuarios potenciales.....	52
Tabla 10. Escala numérica para el ISG	52
Tabla 11. Anexo: Historia de Usuario Añadir Pregunta Frecuente.....	62
Tabla 12. Anexo: Historia de Usuario Añadir Trámite.....	63
Tabla 13. Encuesta de Satisfacción	64
Tabla 14. Anexo: Lista de Chequeo de Usabilidad.....	65

ÍNDICE DE ILUSTRACIONES

Ilustración 1. Página Principal. Parte 1	29
Ilustración 2. Página Principal. Parte 2.....	30
Ilustración 3. Página Principal. Parte 3.....	31
Ilustración 4. Arquitectura 5-capas	35
Ilustración 5. Diagrama de clases	36
Ilustración 6. Diagrama de Secuencia.....	37
Ilustración 7. Factory Method	39
Ilustración 8. Dependency injection.....	39
Ilustración 9. Diagrama de Despliegue.....	40
Ilustración 10. Diagrama de Componentes	42
Ilustración 11. Estándares: Operadores	43
Ilustración 12. Estándares: Etiquetas de apertura y cierre de PHP	43
Ilustración 13. Estándares: Uso de comillas.....	44
Ilustración 14. Estándares: Estructuras de control.....	44
Ilustración 15. Estándares: Arreglos.....	45
Ilustración 16. Estándares: Funciones.....	45
Ilustración 17. Resultados Pruebas Funcionales.....	48
Ilustración 18. Resultados de las pruebas de seguridad	49
Ilustración 19. Resultados de las pruebas de carga para 200 usuarios concurrentes.....	50
Ilustración 20. Anexo: Página Quiénes Somos.....	60
Ilustración 21. Anexo: Vista de noticias ampliada.....	61

INTRODUCCIÓN

La constante evolución de las Tecnologías de la Información y las Comunicaciones (en lo adelante TIC), junto a la aparición de nuevas y más complejas formas de su utilización, hacen de estas un elemento estratégico para el crecimiento, maduración y transformación de las empresas e instituciones. El uso de las TIC potencia el desarrollo económico y social de cada país, evidenciado por la fuerte influencia que tiene en los distintos ámbitos. Es por ello que a día de hoy se le ofrece especial atención a su inclusión en todos los sectores de la sociedad, como estrategia para alcanzar un correcto manejo de la información y de los recursos disponibles.

Fomentar y desarrollar la informatización de la sociedad cubana, y acercar a los ciudadanos a las (TIC), constituye uno de los objetivos estratégicos de nuestro país, según lo establecido en la Conceptualización del Modelo Económico y Social Cubano y en las Bases del Plan Nacional de Desarrollo Económico y Social hasta 2030. El Estado cubano promueve el desarrollo y utilización de las TIC, con el objetivo de que constituyan una fuerza política, científica y económica, que contribuya y propicie la integración y conducción de los procesos asociados a la informatización de la sociedad (Ministerio de Justicia, 2018).

De forma paulatina, y en cumplimiento del cronograma propuesto por el país, se ha llevado a cabo la informatización de los procesos en los Órganos de Administración Central del Estado, en el Banco Central de Cuba y en empresas e instituciones de la salud, la educación y el sector artístico, científico y económico, desencadenando una mayor productividad, mejor comunicación entre empresas y usuarios, menores costos y mejores beneficios y dotando a los más diversos sectores poderosas herramientas para el acceso a la sociedad conocimiento. La Administración Tributaria es una de las entidades cubanas beneficiadas por el proceso de informatización, adoptando procedimientos y tecnologías más avanzados, para facilitar el acceso y uso de la información y el conocimiento a sus contribuyentes.

Según Ministerio de Justicia (2013) la Oficina Nacional de Administración Tributaria (ONAT), es una entidad presupuestada adscrita al Ministerio de Finanzas y Precios, tiene como misión principal brindar un servicio de calidad a la sociedad para el cumplimiento de las responsabilidades tributarias, conforme a la ley. Es la encargada de velar por la aplicación de la legislación relativa a impuestos y otros ingresos no tributarios; desarrollar la organización para su recaudación en todo el país, organizar y dirigir la auditoría fiscal, adoptando las medidas requeridas para contrarrestar la evasión fiscal.

En Villa Clara esta organización cuenta con una oficina provincial que constituye el segundo nivel con funciones de dirección, mando y control sobre las trece oficinas municipales, distribuidas de acuerdo con la división Política Administrativa cubana. Una de las funciones principales de esta organización es

favorecer las condiciones para el cumplimiento voluntario de las obligaciones fiscales. La ONAT provincial ofrece información a los contribuyentes mediante la divulgación de la información para la realización de trámites y servicios que se les ofrece. La localización de contenidos sobre temas tributarios y legislativos, y la presentación oportuna de información a los contribuyentes sobre su situación fiscal son algunos de los elementos que conforman la atención a los clientes en la ONAT de Villa Clara. Esta asistencia se ofrece por diferentes vías, a través de correos electrónicos o correo tradicional, teléfono, la radio y televisión, y a través de materiales impresos si la persona se presenta en la entidad.

Aún y cuando existen varios mecanismos y espacios creados para informar oportunamente sobre los cambios que sufre la legislación, no son vías suficientes para mantener informados a todo el universo de clientes sobre sus obligaciones y derechos, pues están en constante actualización y varían a menudo. Entre otros elementos se observa que el flujo de información no corresponde o satisface las necesidades de información de los contribuyentes, los que se han visto afectados en ocasiones porque la información no le llega de forma oportuna, o no llega.

Teniendo en cuenta la problemática anteriormente descrita se identifica como **problema de investigación**: ¿Cómo contribuir a la gestión de la información de los procesos de la Oficina Nacional de Administración Tributaria en Villa Clara?

Como **objetivo general** de la presente investigación se plantea: Desarrollar un portal web que permita la gestión de la información de los procesos de la Oficina Nacional de Administración Tributaria de Villa Clara.

Por tanto, **el objeto de estudio** se centra en la gestión de información, delimitando el **campo de acción** al proceso de la gestión de información en la Oficina Nacional de Administración Tributaria.

Para darle cumplimiento al **objetivo general** de la investigación se trazaron los siguientes **objetivos específicos**:

1. Realizar de los referentes metodológicos de la investigación a partir de la problemática identificada.
2. Identificar y describir las principales tecnologías y herramientas a utilizar para la creación del portal web.
3. Desarrollo del Portal web para la Oficina Provincial de la ONAT de Villa Clara.
4. Validar el correcto funcionamiento del sistema mediante la aplicación de pruebas.

Se define como **hipótesis**: el desarrollo del portal web contribuirá a los procesos de gestión de la información en la oficina provincial de la ONAT en Villa Clara.

Se utilizaron los siguientes métodos científicos para el desarrollo de la investigación:

Métodos Teóricos

- **Analítico – Sintético:** Se utiliza al realizar la descomposición del problema en varias partes para profundizar en el estudio de cada una de ellas por separado y luego integrarlas para llegar a conclusiones precisas acerca de la investigación.
- **Modelación:** Se emplea al realizar una representación de las características del sistema mediante diagramas en el proceso de desarrollo del portal web para la oficina provincial de la ONAT de Villa Clara.

Métodos Empíricos

- **Observación:** Permite obtener la información necesaria acerca del desarrollo de portales web, así como el estudio de portales homólogos para identificar elementos que tienen en común estos sitios, lo cual posibilita arribar a conclusiones concretas.
- **Entrevista:** Se realiza a la gestora del negocio, con el objetivo de obtener las funcionalidades que debe presentar el portal web.

Estructuración por capítulos

El presente documento se encuentra estructurado en tres capítulos:

CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA DEL PORTAL WEB DE LA OFICINA PROVINCIAL DE LA ONAT DE VILLA CLARA: En este capítulo se exponen las principales definiciones asociadas a la entidad y al objeto de estudio. Se realiza el estudio del estado del arte y se definen las tecnologías y herramientas a utilizar.

CAPÍTULO 2 DESCRIPCIÓN DEL PORTAL WEB DE LA OFICINA PROVINCIAL DE LA ONAT DE VILLA CLARA: Se describe en el capítulo la propuesta de solución, se definen los requisitos funcionales y no funcionales que debe cumplir el portal. Se diseña la arquitectura del sistema y se caracterizan los patrones de diseño utilizados.

CAPÍTULO 3 IMPLEMENTACIÓN Y VALIDACIÓN DEL PORTAL WEB: Se ejecutan las pruebas funcionales aplicando la técnica de caja negra, pruebas de seguridad, carga y estrés y usabilidad para verificar el correcto funcionamiento del sistema.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA ACERCA DEL DESARROLLO DEL PORTAL WEB DE LA OFICINA PROVINCIAL DE LA ONAT DE VILLA CLARA

1.1 Introducción del capítulo

En el presente capítulo se definen las categorías conceptuales asociadas a la entidad y al objeto de estudio de la investigación. Se realiza un estudio de sistemas homólogos como fuente de información, con el objetivo de obtener los requerimientos de la propuesta de solución. Se definen las herramientas y tecnologías necesarias para el desarrollo del portal web de la ONAT.

1.2 Categorías conceptuales

Teniendo en cuenta la importancia que se le confiere a un correcto entendimiento de los elementos asociados a la entidad y al objeto de estudio, y que serán tratados durante todo el proceso de investigación y desarrollo de la propuesta de solución, al autor decide presentar la conceptualización de los mismos. A continuación, se muestra la definición de ellos agrupados en dos categorías: Categorías conceptuales asociadas a la entidad y Categorías conceptuales al objeto de estudio.

1.2.1 Categorías conceptuales asociadas a la entidad

Los contribuyentes asisten mayormente a las oficinas municipales, para realizar trámites propios del Registro de Contribuyentes. Los trámites fiscales están regulados en la Ley No.113 del Sistema Tributario y en el Decreto No. 308 Reglamento de las Normas Generales y de los Procedimientos Tributarios, publicados en la Gaceta Oficial Extraordinaria No. 53 de 2012. Teniendo en cuenta que, para la exitosa ejecución de estos, es imprescindible conocer como cumplimentarlos; a continuación, se ejemplifican y describen los trámites que se realizan en la entidad.

❖ Inscripción

La inscripción en el Registro de Contribuyentes (RC) debe producirse en los 15 días naturales siguientes a: la emisión de la aprobación del ejercicio de la actividad por el órgano rector, si se trata de una persona natural (trabajador por cuenta propia, artista, comunicador, diseñador u otros) y la fecha de inscripción en el registro constitutivo correspondiente si es una persona jurídica (empresa, cooperativa no agropecuaria u otra entidad).

Para inscribirse la persona natural debe presentar:

- El documento de identidad.
- La autorización, licencia, permiso y afiliación emitidos por el órgano rector que aprueba la actividad.
- El Poder Notarial o Tutoría Legal que acreditan al representante, si el obligado a inscribirse no puede acudir.
- La Declaración de trabajadores contratados y copropietarios.
- El documento para la afiliación al Registro de Seguridad Social.

La persona jurídica debe presentar:

- La certificación de la inscripción en el registro constitutivo correspondiente.
- La Copia de la Escritura Pública protocolizada u otro documento acreditativo de su constitución como entidad.
- El documento oficial o poder notarial que acredite a la persona que acude a efectuar la inscripción como representante legal de la entidad.
- La declaración de dependencias, cuya acreditación en el RC debe hacerla la entidad que actúa en calidad de contribuyente principal en los 45 días naturales siguientes a su inscripción en la ONAT.

❖ **Baja**

Para causar baja en el Registro de Contribuyentes de la ONAT se deben presentar:

- El documento emitido por el órgano rector que autoriza la baja, en el caso de las personas naturales.
- El RC-05 Identificación Fiscal Única (carnet del contribuyente)
- El documento que acredita la cancelación de la inscripción de la entidad en el registro correspondiente, en el caso de las personas jurídicas.

Si no presenta el documento acreditativo de la baja en la ONAT, el contribuyente seguirá activo en la base de datos de la oficina, acumulando deudas. Por eso es muy importante que después de recibir el documento de baja se dirija a la ONAT para que esta se haga efectiva.

❖ **Modificación o Actualización**

Si ocurre algún cambio que implique una modificación del Registro del Contribuyente (traslado, cambio de actividad, régimen, dirección, cuota, de local, representante legal de las entidades, etc.), el contribuyente dígame persona natural o jurídica, debe informarlo, en un término de treinta (30) días naturales. Para ello debe presentar el RC-05 Identificación Fiscal Única y los mismos documentos solicitados para la inscripción.

❖ **Exoneración por suspensión temporal**

La suspensión temporal del ejercicio de la actividad económica es otorgada por el órgano rector a solicitud del contribuyente. Mientras que la exoneración de las obligaciones fiscales la concede la ONAT, una vez que el órgano rector ha aprobado dicha suspensión temporal. Para solicitar la exoneración por suspensión temporal el contribuyente debe presentar ante la oficina tributaria:

- El documento del órgano rector que dispone la suspensión temporal del ejercicio de la actividad.
- El modelo RC-05 Identificación Fiscal Única, o carnet del contribuyente.

Las suspensiones temporales también pueden producirse por decisión de los gobiernos territoriales cuando existen circunstancias que obligan a ello.

❖ **Certificaciones Fiscales**

Constituye un documento emitido por ONAT que da fe de la situación fiscal actualizada del contribuyente persona natural (ciudadano) y jurídica (empresas, unidades presupuestadas, cooperativas no agropecuarias u otras entidades). El contribuyente puede solicitar certificaciones fiscales de:

- Inscripción en el Registro de Contribuyentes.
- Residencia Fiscal en Cuba (según convenios para evitar la doble imposición).
- No adeudos fiscales.
- Contribuciones efectuadas al Régimen de Seguridad Social.
- Cumplimiento de Obligaciones Tributarias.

El contribuyente puede solicitar estas certificaciones por escrito en la ONAT del municipio donde está inscrito y la institución se la ofrece en correspondencia con lo que haya pedido que se le certifique. El escrito de solicitud puede ser presentado por el contribuyente o su representante legal ante notario, quien debe mostrar las escrituras notariales y documentos oficiales que lo acreditan como tal.

Si la certificación se necesita para realizar trámites migratorios, el contribuyente debe informarlo en el momento de solicitarla. En estos casos debe ser firmada por la Directora Jurídica de la ONAT, antes de ser legalizada en el Ministerio de Relaciones Exteriores. Este proceso puede demorar hasta 15 días, pues el documento debe viajar del municipio a la provincia, de esta a la ONAT Central en La Habana y luego retornar al punto de origen: la oficina municipal.

❖ **Aplazamiento de la deuda**

Los contribuyentes tienen derecho a solicitar convenios de aplazamiento de la deuda tributaria, cuando no cuentan con el dinero suficiente para liquidarla. Para ello se debe presentar un escrito donde consten: nombres y apellidos (si es persona natural), denominación o razón social (si es persona jurídica); NIT y dirección del domicilio legal. En caso de ser formulada por el representante legal del contribuyente incluye sus nombres y apellidos, número de identidad permanente o de pasaporte y el domicilio legal.

Este documento contendrá también, la deuda tributaria que pretende aplazarse, importe, concepto y fecha en que finaliza el término voluntario de pago del tributo, las causas que motivan el aplazamiento, el tipo de aplazamiento que se pide, y si fuera con fraccionamiento, los plazos que se necesitan. Además, deben incluirse en él los bienes muebles e inmuebles del patrimonio del deudor o del responsable, cuyo importe baste para el cumplimiento de la deuda tributaria; el lugar, la fecha de la solicitud y la firma del solicitante, con cuño gomígrafo, en el caso de las personas jurídicas.

El aplazamiento es una potestad de la ONAT, la cual puede aplazar toda, o parte de la deuda tributaria. Su otorgamiento exige el pago de un interés que incrementa la deuda. Puede ser con o sin fraccionamiento, o combinar ambas modalidades. Si se solicita antes del vencimiento del término de pago, no se exigen ni recargo por mora ni sanciones fiscales. En caso contrario, comprenderá el importe del principal (el tributo o aporte a liquidar), el recargo por mora, la multa por la infracción cometida, más el interés.

1.2.2 Categorías conceptuales asociadas al objeto de estudio

Para lograr una mayor comprensión de la presente investigación se recogen a continuación un conjunto de conceptos asociados al objeto de estudio.

Aplicaciones web

Son aplicaciones de software, a las cuales se puede acceder a través de un servidor web, haciendo uso de un navegador. Para su codificación o desarrollo son empleados lenguajes que permitan su entendimiento y soporte por parte de los navegadores web.

En la actualidad las aplicaciones web son ampliamente utilizadas por los usuarios, a partir de las grandes ventajas que les ofrece desde el punto de vista funcional y a nivel de sistema. A continuación, se exponen algunas de sus características:

- Son aplicaciones multiplataforma independientes del sistema operativo.
- Parte del procesamiento se realiza en el servidor por lo que los clientes web no necesitan ser muy potentes.
- Comunicación mediante HTTP sobre TCP/IP.
- Acceso a bases de datos.
- Arquitectura por capas.
- Distintos tipos de usuarios (Universidad de Sevilla, 2012).

Portales web

Son aplicaciones, compuestas por un conjunto de documentos electrónicos o páginas web, las cuáles se relacionan entre sí, a través de referencias (vínculos) entre ellas. Según Moreno (2018) los portales web son páginas estructuradas, sobre la base de una organización de contenidos definidos con anterioridad, y que responden a una temática determinada. Como parte de la organización de las páginas web que conforman en portal, se define una página principal o home page, que contiene los contenidos principales o los enlaces hacia los mismo a través de bloques de menú. Para acceder a los portales web se emplea una dirección de internet o nombre de dominio que hace referencia al lugar donde se encuentra hosteado o anidado.

En correspondencia con el contenido que se ofrece o publica en ellos y el público objetivo al que va dirigido, los portales web se clasifican en dos tipos. Según García (2021), las clasificaciones que se mencionan responden a Portales Generales y Portales Especializados. A continuación, se conceptualizan estas clasificaciones.

- ❖ Portales Generales: Como su nombre lo muestra, está orientado a todo tipo de público. Ofrece contenidos de carácter muy amplio, cuya pretensión es cubrir las temáticas más demandadas. Entre las características más notables de este tipo de portales está que ofrecen servicios de valor añadido como comunidades virtuales, espacio web gratuito informaciones de diversos tipos, personalización de la información, chat, e-mail gratuito, mensajes a teléfonos móviles, software gratuito, grupos de discusión, comercio electrónico y buscador.
- ❖ Portales Especializados: Se pueden emplear en un área geográfica determinada, un tema concreto o bien estar destinados a cubrir las necesidades de las personas relacionadas con una corporación concreta (empleados, directivos, clientes, proveedores). Esta cobertura la ofrecen los denominados portales verticales y los portales corporativos.
 - Portales Corporativos: Por lo general se muestran en forma de una intranet que provee de información de la empresa a los empleados, así como de acceso a una selección de Web públicos y Webs de mercado vertical (proveedores, vendedores, etc.) Incluye un motor de búsqueda para documentos internos, así como la posibilidad de personalizar el portal para diferentes grupos de usuarios y particulares.
 - Portales Verticales: Brindan información y servicios a una industria en particular. Han de contener más datos sobre su mercado que cualquier otro sitio, lo que en sí mismo ya es un valor añadido considerable. Permiten cubrir las carencias que muestran muchos de los portales generalistas, que pretenden llegar a todo tipo de usuarios pero que, en realidad, no son capaces de cubrir de una forma profunda ninguna de estas áreas temáticas.

Luego del estudio de las definiciones abordadas con anterioridad y teniendo en cuenta las características del sistema a desarrollar y el público al cual se dirige la propuesta de solución, la autora de la presente investigación considera que la categoría que más se ajusta es Portal especializado, y dentro de este, corporativo.

Arquitectura de Información de las aplicaciones web

Los portales web al igual que otros tipos de aplicaciones, definen estructuras de contenidos para mostrar la información que brindarán a los usuarios. Estas estructuras forman parte de la Arquitectura de Información del sistema o aplicación, y su función principal el diseño de espacios de información,

que permiten una mejor organización y entendimiento del sistema. De ahí la importancia de su correcto diseño y conformación.

Rodríguez, et al (2017) plantea que: La Arquitectura de Información (En lo adelante, AI) es la disciplina encargada de distribuir los elementos formales y de contenido que integran un sitio web. Utiliza actividades enfocadas al usuario para satisfacer sus necesidades. Abarca la organización de la información y el resultado de dicha actividad. En los portales web, como resultado de la actividad, comprende los sistemas de organización y estructuración de los contenidos, los sistemas de rotulado o etiquetado de dichos contenidos, y los sistemas de recuperación de información y navegación que provea el sitio web que provee al sitio web

Usabilidad de las aplicaciones web

Estrechamente ligada al diseño de la AI, se encuentra la usabilidad web. Y es que al diseñar la primera debe pensarse siempre en si responde a un sistema usable, es decir, si el producto contiene las características necesarias que le permita ser usado por usuarios específicos para conseguir objetivos determinados con eficacia, eficiencia y satisfacción en un contexto de uso dado (Rocca & Solari, 2018). Todos los sistemas exigen cierto grado de usabilidad para poder ser operados, los contenidos deben estar organizados, debe ser intuitivo y contener la información necesaria para ofrecer a los usuarios lo que buscan. Es importante que se visualice el ícono que identifica la entidad y que sea de fácil navegabilidad.

La atención a este requisito es fundamental en términos de coste y tiempo, ya que se estima que el 80% de los errores detectados en los sistemas de información provienen de la interfaz de usuario, de los cuales, alrededor de un 64% son causados por problemas de usabilidad. Esto supone que la desatención a la usabilidad como requisito no funcional pueda llegar a ser el origen de la mayoría de desarrollos correctivos que necesite un sistema de información a lo largo de su ciclo de vida (Yáñez, 2019).

Para el desarrollo del portal web de la ONAT, la autora considera necesaria la incorporación de esta característica a la aplicación, y para ello debe definir correctamente los elementos que permitirán concebir el portal web como una aplicación usable. Debe prestarse especial atención al diseño y uso de la iconografía, los componentes de la navegación, la estructura de los contenidos, los nombres de las secciones, entre otros. De ello depende en gran medida que el resultado de la interacción con el portal sea satisfactorio.

1.3 Entorno actual de los portales web institucionales

Teniendo en cuenta las necesidades detectadas en la organización se decide realizar un estudio de sistemas homólogos con el objetivo de identificar características en los sistemas que permitan dar

respuesta a las necesidades del cliente. Se analizan además las tecnologías empleadas para el desarrollo de los sistemas estudiados. A continuación, se muestran las soluciones estudiadas:

1.3.1 Portales Tributarios Institucionales

Servicios de Impuestos Internos en Estados Unidos

En la página principal del portal, se estructuran los contenidos mediante sesiones, a las que se puede acceder a través de enlaces a los bloques de menú, localizados en la región superior de la portada. El sistema permite acceder a distintas informaciones, sobre negocios y trabajadores por cuenta propia, obtener un pago determinado o registro tributario, elegir métodos de pago y gestionar las cuentas que el usuario haya creado en el sistema. Existen sesiones que son solo informativas como reembolsos, créditos y deducciones y formularios e instrucciones las cuales son sólo informativas, aunque esta última permite realizar descargas. Presenta adaptabilidad a las distintas resoluciones de pantalla.

Dirección General Impositiva de Uruguay

El portal en la página principal muestra un contenido muy variado, organizados en bloques, cuenta con un menú principal, localizado en la región superior de la portada, que permite acceder a las páginas interiores a través de enlaces, uno de ellos llamado Institucional, el cual posibilita conocer información acerca de la entidad. El sistema permite a los usuarios autenticarse para acceder a los trámites y servicios que se brindan en línea, solicitud de inicio de trámites, envío de declaraciones, pagos web, entre otros. Se puede realizar descargas de archivos en formato pdf. Presenta adaptabilidad a las distintas resoluciones de pantalla.

Ministerio de Hacienda y Crédito Público de Colombia

El portal muestra en la portada el contenido organizado en bloques, cuenta con un menú principal que se encuentra en la región superior, con distintas sesiones que poseen enlaces a las páginas interiores como catálogo de trámites, que permite al usuario registrar su hoja de vida para solicitud de empleo. En la sección servicios y carpeta privada el cliente puede obtener informaciones específicas y realizar solicitudes en línea, para realizar estas acciones en la página debe existir una autenticación previa del usuario, por lo que el sistema permite registrarse y autenticarse, se pueden descargar archivos en formato pdf. El portal es adaptativo a las distintas resoluciones de pantalla.

1.3.2 Portales Tributarios Nacionales

Portal Tributario

El portal tributario permite acceder a las sesiones informativas, a todos los usuarios. En la región superior de la portada se muestra el menú principal permitiendo el acceso a los contenidos a través de enlaces como nuestra institución que posibilita conocer mediante una multimedia cual es la misión y visión de la institución, así como la estructura que presenta y sus directivos. Permite descargar en

formato pdf las principales leyes relacionadas con la administración tributaria, también posee una sección propia para descargas la cual permite obtener todos los modelos y formularios establecidos para declarar, pagar tributos y registrar los ingresos y gastos de la actividad económica. El sistema ofrece servicios en línea como consultas de pagos, acceso al vector fiscal, modificación de la cuota y suspensiones, a los cuales el usuario puede acceder siempre que esté autenticado. El portal es adaptable a las diferentes resoluciones de pantalla.

El estudio y análisis de los sistemas homólogos expuestos anteriormente, permite la construcción de la siguiente tabla comparativa. La misma permitirá a la autora, llegar a conclusiones sobre características y elementos del desarrollo a incorporar a la propuesta de solución.

Tabla 1. Comparación de Sistemas
Fuente: Elaboración Propia

Portales	Tecnologías	Funcionalidades
Servicios de Impuestos Internos en Estados Unidos	CMS: Drupal Framework: Bootstrap Lenguaje: PHP Servidor Web: Apache, Amazon EC2 Gestor de Base de Datos: Percona Librerías: JQuery UI, Boomerang, JQuery	- Cambiar Idioma - Buscar - Registrar usuario - Autenticar usuario - Obtener pagos - Efectuar Pagos - Realizar Solicitudes en Línea - Descargar Leyes en formato pdf
Dirección General Impositiva de Uruguay	Lenguaje: Java Servidor Web: Apache Librerías: JQuery	- Buscar - Registrar Usuario - Autenticar Usuario - Realizar Solicitudes en Línea - Descargar Leyes en formato pdf
Ministerio de Hacienda y Crédito Público de Colombia	Lenguaje: Java Servidor Web: Apache Librerías: JQuery	- Buscar - Registrar Usuario - Autenticar Usuario - Descargar Leyes en formato pdf - Realizar Solicitudes en Línea - Validar documentos

Portal Tributario	Framework: Bootstrap, Laravel Lenguaje: Java, Python, PHP Servidor Web: Apache Gestor de Base de Datos: Adobe Experience Manager Librerías: JQuery, Lodash, Axios	- Buscar - Registrar Usuario - Autenticar Usuario - Descargar Leyes en formato pdf - Realizar Consultas - Realizar Solicitudes en Línea
--------------------------	--	--

Resultados

El análisis de los sistemas estudiados permitió definir características y requisitos, que corresponden con la respuesta a las necesidades del cliente, los que se enumeran a continuación:

1. Adaptabilidad a las distintas resoluciones de pantalla
2. Funcionalidades
 - Buscar
 - Registrar Usuario
 - Autenticar Usuario
 - Realizar Solicitudes en Línea (Informaciones fiscales)
 - Descargar Leyes en formato pdf

Se incorporan nuevas características, resultado del análisis de otras fuentes de información, como documentos y requisitos planteados por la entidad cliente.

1.4 Tecnologías y Herramientas a utilizar

Para la correcta implementación de un portal web, es de gran importancia definir con anterioridad las tecnologías, metodología de desarrollo, sistema de gestión de contenidos, lenguajes de programación, servidor web, gestor de base de datos, lenguaje a utilizar en el modelado del sistema, así como las herramientas para realizar dicho modelado y las pruebas al sistema.

1.4.1 Metodología de desarrollo

Una metodología de desarrollo es un proceso para producir un software de forma organizada, empleando una colección de técnicas y convenciones de notación predefinidas. Desde una perspectiva de Ingeniería de Software, describe cómo se organiza un proyecto, establece el orden en el que la mayoría de las actividades tienen que realizarse y los enlaces entre ellas e indica cómo tienen que realizarse algunas tareas proporcionando las herramientas concretas e intelectuales (Dr. F. J. García et al., 2020).

Para guiar el proceso de desarrollo de la propuesta de solución se decide el empleo de la metodología AUP en la versión UCI, pues teniendo en cuenta las características del negocio y el equipo de

desarrollo, es la más apropiada. Además, se tiene en cuenta que es la metodología estandarizada en la Universidad de las Ciencias Informáticas (UCI) para soportar el ciclo completo de desarrollo de los proyectos.

Proceso Ágil Unificado en la versión UCI

Según Rodríguez (2015) AUP en la versión UCI es una modificación que realiza la Universidad a la metodología Proceso Ágil Unificado (AUP), con el fin de tener una metodología que se adapte al ciclo de vida definido por la actividad productiva en los centros de desarrollo de la universidad. AUP propone 4 fases para el desarrollo de los proyectos: Inicio, Elaboración, Construcción y Transición. En la nueva versión se mantiene la fase de Inicio, aunque se modifica el objetivo de ella, se unifican las restantes fases en una sola y se le llama Ejecución y se agrega una nueva fase nombrada Cierre. A continuación, se muestran las actividades que se realizan en cada fase:

- ❖ Inicio:
 - Planeación del proyecto.
 - Realización del estudio inicial de la entidad cliente que permite obtener información fundamental acerca del alcance del proyecto.
 - Realizar estimaciones de tiempo, esfuerzo y costo y decidir si se ejecuta o no el proyecto.
- ❖ Ejecución:
 - Ejecución de las actividades requeridas para desarrollar el software.
 - Modelado del negocio, mediante la obtención de los requisitos.
 - Elaboración de la arquitectura y el diseño.
 - Implementación y liberación del producto.
 - Capacitación de los usuarios finales sobre la utilización de la aplicación.

Nota: Durante esta fase el software es transferido al ambiente de los usuarios finales o entregado al cliente junto con la documentación.
- ❖ Cierre:
 - Análisis de los resultados del proyecto y ejecución.
 - Ejecución de las actividades formales de cierre del proyecto.

La metodología AUP en su variante UCI propone tres variantes a utilizar en los proyectos, CUN (Casos de uso del negocio), DPN (Descripción de proceso de negocio) y MC (Modelo conceptual) y en correspondencia formas para encapsular los requisitos, CUS (Casos de uso del sistema), HU (Historias de usuario), DRP (Descripción de requisitos por proceso). A continuación, se muestra la descripción de cada escenario donde se describe para cada variante de proyecto la forma de encapsular los requisitos:

- 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.

En este caso se ha seleccionado el escenario 4, ya que la entidad cliente estará presente en todo el proceso de desarrollo del sistema, facilitando así que se puedan definir con claridad los requisitos para la correcta implementación y validación del portal web. Este escenario es recomendado para proyectos que no sean extensos.

1.4.2 Sistemas de Gestión de Contenidos

Los Sistemas de Gestión de Contenidos, en lo adelante (CMS) se definen como herramientas informáticas estructuradas sobre la base de módulos que permiten la creación, almacenamiento, actualización, recuperación y visualización de los contenidos, a distintos niveles de gestión y acceso. Ofrecen la ventaja de mantener separado el contenido de su apariencia o presentación final. El sistema que gestiona la base de datos en los CMS está diseñado para realizar otras operaciones; como son la posibilidad de integrar tesauros, taxonomías y ontologías para mejorar la gestión, la importación y exportación de base de datos ya existentes, y la incorporación a la misma de documentos en distintos formatos (Sorzano, 2019).

Drupal 9

El uso de un CMS constituye una herramienta fundamental en el desarrollo del sistema ya que brindan la posibilidad de cambiar el diseño y la personalización del portal web de manera instantánea gracias a las plantillas o temas. Posibilitan la creación rápida del contenido mediante las capacidades y tareas que brinda el software. En la propuesta de solución se decide usar Drupal como Sistema de Gestión de Contenidos, pues constituye un requisito (de tipo restricción), definido por la entidad cliente ya que cuenta con gran capacidad para almacenar y gestionar grandes volúmenes de contenido y soportar webs con alto tráfico. Está diseñado para crear fácilmente contenido estructurado y versátil y conectar potentes herramientas de integración, es utilizado para crear muchos de los sitios web y aplicaciones que se encuentran en la actualidad disponibles en internet.

Drupal es un software de gestión de contenido que posee características estándares, como creación de contenido sencilla, rendimiento confiable y excelente seguridad, pero lo que lo distingue realmente es su flexibilidad y escalabilidad, permite crear todo tipo de portales webs personalizados y adaptados,

facilitando que la adición o reducción de módulos, funcionalidades, componentes, etc., no afecte el ciclo de desarrollo del proyecto, garantizando la máxima efectividad y el mejor funcionamiento. Este CMS facilita la creación portales multilingües, permite escoger entre varios idiomas para llevar a cabo el desarrollo (*Drupal, 2021*).

Drupal 9 es totalmente *responsive*, lo que permite la correcta visualización del portal en las diferentes resoluciones de pantalla. Sus herramientas le ayudan a crear el contenido estructurado y versátil que se necesita en las webs dinámicas, los módulos amplían sus funcionalidades y los temas le permiten personalizar la presentación de su contenido. Está basado en la plantilla "Twig", lo que le aporta seguridad contra los ataques informáticos. Dispone de un amplio equipo de trabajo centrado en las políticas de seguridad, en la constante revisión del código y en la búsqueda de soluciones de vulnerabilidades. Es uno de los CMS más seguros y potentes del mundo (*Drupal, 2021*).

Twig

Es un motor de plantillas moderno y rápido, compila plantillas en código PHP simple y optimizado. Es seguro, contiene un modo de zona de pruebas para evaluar el código de plantilla que no es de confianza, esto permite que Twig se utilice como lenguaje, para aplicaciones donde los usuarios pueden modificar el diseño de la plantilla. Permite al desarrollador definir sus propias etiquetas y filtros personalizados (*Twig, 2021*).

1.4.3 Lenguaje para el modelado del sistema

La construcción de sistemas y aplicaciones informáticas requiere de una preparación previa, que tribute al éxito en su proyecto de desarrollo. Desde etapas iniciales del ciclo de vida del proyecto deben elaborarse modelos, diagramas y representaciones arquitectónicas que describan los elementos componentes del sistema y las relaciones entre ellos, facilitando así, el adecuado entendimiento del negocio y el desarrollo de la propuesta. A continuación, se describe el lenguaje y la herramienta de modelado seleccionada por la autora para el modelado de la aplicación.

Según Aristega & Lárraga (2018) el lenguaje de Modelado Unificado (UML) es una técnica de modelado de objetos y como tal supone una abstracción de un sistema para llegar a realizarlo en términos concretos. Es la construcción de un modelo a partir de una especificación, una abstracción que se elabora para comprender un proyecto antes de su implementación. Los diagramas UML utilizados para el diseño del sistema suelen ser: diagramas de clases, objetos, colaboración, secuencia, de estado, actividad, componentes y de ejecución, además se pueden generar híbridos de estos, extenderlos y organizarlos.

Debido a la elección de Aup en la versión UCI como metodología de desarrollo, en combinación con el escenario seleccionado se generan varios artefactos en todo el proceso de desarrollo del sistema,

permitiendo así visualizar, especificar, construir y documentar las partes del software. Ejemplo de ello es el levantamiento de requisitos funcionales y no funcionales, las historias de usuario, el modelo de despliegue, los diagramas de componentes, entre otros.

1.4.4 Herramientas CASE

Las Herramientas CASE pueden definirse como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un Software: Investigación Preliminar, Análisis, Diseño, Implementación e Instalación (Pua, 2016). A través de UML, estas herramientas pueden automatizar los procesos y facilitar las tareas de coordinación de los eventos que necesitan ser mejorados en el ciclo de desarrollo de software.

Visual Paradigm

Visual Paradigm es una herramienta CASE que ofrece un entorno para la creación de diagramas para UML. Su diseño está centrado en casos de usos y enfocado al negocio, lo que permite obtener un software de mayor calidad. Usa un lenguaje estándar, común a todo el equipo de desarrollo lo que facilita la comunicación. Cuenta con disponibilidad de múltiples versiones, para cada necesidad, permitiendo integrarse en los principales entornos de desarrollo (*Visual Paradigm, 2021*).

1.4.5 Tecnologías y Lenguajes de programación

Los lenguajes de programación constituyen herramientas que permiten construir aplicaciones. Consiste en un conjunto de órdenes o comandos que describen el proceso deseado. Cada lenguaje tiene sus instrucciones y enunciados verbales propios, que se combinan para formar los programas de cómputo (*Lenguajes de Programación, 2021*).

HTML5

El Lenguaje de Marcas de Hipertexto (HTML) permite definir el contenido de las páginas web. Se compone en base a etiquetas, también llamadas marcas o *tags*, mediante las cuales se pueden expresar las partes de un documento, cabecera, cuerpo, encabezados, párrafos. (*Lenguaje HTML, 2021*). El empleo de HTML permite estructurar y desplegar el portal web y sus contenidos, a través de una estructura lógica y fácil de interpretar. Es multiplataforma, puede ser visualizado por cualquier navegador sin distinción del sistema operativo. HTML5 abarca lenguajes como CSS y también JavaScript.

CSS3

Es un lenguaje utilizado en el diseño web que define la presentación de una página web o documento HTML. Sus siglas significan Hojas de estilo en cascada, permite aplicar estilos a uno o varios documentos o páginas webs de forma masiva. La idea de CSS es utilizar el concepto de separación de

presentación y contenido, intentando que los documentos HTML incluyan sólo información y datos (*Lenguaje CSS*, 2021). Este lenguaje es de gran importancia en la implementación de la propuesta de solución, permite incorporar diferentes estilos (colores, formas e imágenes) al sitio siguiendo las normas establecidas en estos tipos de portales, se encarga del diseño visual de las interfaces escritas en HTML permitiendo que el sistema sea agradable a la vista del usuario.

JavaScript

Es un lenguaje de programación ligero, conocido como lenguaje de *scripting* (secuencias de comandos) para páginas web. Está basado en prototipos, es multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa, por lo que utiliza clases y objetos como estructuras, reutilizables durante todo el desarrollo, lo que permite organizarse de forma simple. Es un lenguaje de alto nivel ya que su sintaxis es fácilmente comprensible por su similitud al lenguaje de las personas (*Lenguaje JavaScript*, 2021). JavaScript se utiliza principalmente en el desarrollo web del lado del cliente, va a permitir realizar validaciones en funcionalidades como registrar y autenticar usuario y en los diferentes formularios llenados por dichos usuarios al realizar una determinada solicitud en el sistema.

JQuery

Es una biblioteca de JavaScript rápida, pequeña y rica en funciones. Es flexible y rápido para el desarrollo web. Permite que el recorrido y la manipulación de documentos HTML, el manejo de eventos, la animación y Ajax sean mucho más simples con una API fácil de usar que funciona en una multitud de navegadores (*JQuery*, 2021). JQuery permite realizar funcionalidades en la interfaz del proyecto que se tornan más difíciles si solo se utiliza JavaScript, mejorando así la interacción con el usuario. Es una de las librerías que tienen en común los sistemas estudiados.

PHP 7.4

Preprocesador de hipertexto (PHP), es un lenguaje de programación de código abierto, su uso general es especialmente adecuado para el desarrollo web y puede integrarse en HTML. El objetivo principal del lenguaje es permitir a los desarrolladores web escribir rápidamente páginas web generadas dinámicamente. Es rápido, flexible y pragmático, impulsa todo, desde su *blog* hasta los sitios web más populares del mundo (*Lenguaje PHP*, 2021). Drupal está escrito sobre PHP, este es el lenguaje más popular en el desarrollo web del lado del servidor, así lo demuestra el estudio realizado sobre algunos portales web relacionados con la Administración Tributaria.

Marco de Trabajo Bootstrap 5

Según (Pavón, 2013) Bootstrap es constituido por un conjunto de herramientas de código abierto para el diseño de aplicaciones web. Ofrece la posibilidad de crear un sitio web totalmente *responsive*

mediante el uso de librerías CSS. En las librerías, se encuentra un gran número de elementos ya desarrollados y listos para ser utilizados como pueden ser botones, menús, cuadros e incluso un amplio listado de tipografías. Bootstrap se caracteriza por tratarse de una excelente herramienta para crear interfaces de usuarios totalmente adaptables a cualquier tipo de dispositivo y pantalla, independientemente de su tamaño. Con su uso se pueden obtener sistemas sencillos y ligeros, puede bastar con un fichero CSS y uno JavaScript. Los portales son compatibles con todos los navegadores habituales y presentan adaptabilidad a las diferentes resoluciones de pantalla.

Herramienta de Maquetación ForeUI 3.20

Es herramienta que permite la creación de prototipos de interfaz de usuario fácil de usar, diseñada para crear maquetas, estructuras alámbricas, prototipos para cualquier aplicación o sitio web. Permite diseñar el comportamiento de los prototipos definiendo diagramas de flujo intuitivos para manejar eventos específicos. Se puede exportar a documentos PDF o simulación HTML5. Es muy útil para compartir ideas, revisar conceptos de diseño, recopilar comentarios y realizar pruebas de usabilidad (*ForeUI*, 2021).

1.4.6 Servidores de aplicaciones web

Los servidores web son aplicaciones encargadas de recibir las peticiones referidas a páginas o elementos de la web a través del protocolo http o https y de devolver el resultado de la petición, que suele ser un recurso alojado en el servidor. El navegador realiza una petición al servidor web del recurso que desea el usuario, para finalmente recibirlo, si fue válida la petición y traducirlo si es necesario a la forma legible por el usuario (J. Sánchez, 2012).

Apache 2.4

Se decide Apache como servidor web, por ser un requisito (de tipo restricción) surgido por una solicitud de la entidad cliente, debido a la experiencia en el trabajo con este servidor ya que es utilizado para el trabajo con las bases de datos locales en la ONAT, es gratuito, cuenta con gran documentación sobre su uso y se actualiza frecuentemente. Es flexible, funciona en la mayoría de los sistemas operativos.

Puede manejar varios sitios y permite el trabajo con contenido dinámico y estático, además de ser recomendado para el trabajo con Drupal, la mayor parte del desarrollo y la implementación de este CMS se realiza en Apache. Es estable, lo cual es una consecuencia de su probada robustez que impide caídas o cambios inesperados en el servidor. Posee infinidad de paquetes y módulos que permiten trabajar con gran cantidad de lenguajes de programación web, así como intérpretes de SQL y otras funciones (Márquez et al., 2002). Para facilitar la instalación de Apache, que puede resultar muy compleja, se pueden utilizar soluciones de software como XAMPP.

XAMPP 3.3

Es un software que permite la distribución de Apache, es completamente gratuita, fácil de instalar, las configuraciones son mínimas, lo que permite ahorrar tiempo. Consiste principalmente en el sistema de gestión de bases de datos MySQL y los intérpretes para lenguajes de script PHP y Perl. Para manejar los sistemas de gestión de base de datos, XAMPP instala la herramienta phpMyAdmin (XAMPP, 2021).

1.4.7 Gestor de Base de Datos

MySQL 10.4

Se decide utilizar MySQL como gestor de bases de datos, pues es el recomendado para el trabajo con Drupal, ya que este CMS generalmente funciona con una configuración predeterminada de MySQL (Drupal, 2021). El uso de otros gestores puede ocasionar errores en el desarrollo del sistema.

MySQL es un sistema de gestión de bases de datos relacionales de código abierto, el cual es un software o servicio utilizado para crear y administrar bases de datos basadas en un modelo relacional. El principal objetivo de MySQL es alcanzar la velocidad y robustez adecuada para dar respuesta a las peticiones del cliente, soporta gran cantidad de tipos de datos para las columnas. Cada base de datos cuenta con 3 archivos: estructura, datos e índice. Presenta un flexible sistema de contraseñas y gestión de usuarios, con un muy buen nivel de seguridad en los datos, aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo (Gustavo, 2020).

PhpMyAdmin es una herramienta de software gratuita escrita en PHP, ofrece una interfaz gráfica intuitiva que facilita la administración del servidor MySQL a través de la web. Admite una amplia gama de operaciones: permite crear, editar y eliminar bases de datos, tablas, vistas, columnas e índices. Muestra múltiples conjuntos de resultados a través de consultas o procedimientos almacenados. Se pueden importar datos y estructuras MySQL desde hojas de cálculo. PhpMyAdmin agregar, editar y eliminar cuentas de usuario y privilegios de MySQL, y crear consultas complejas conectando automáticamente las tablas requeridas (PhpMyAdmin, 2021).

1.4.8 Herramientas de Pruebas

El desarrollo de pruebas de software no solo permite la obtención de un producto acorde a las necesidades del cliente, sino, un producto seguro, eficiente y eficaz. Es por ello que deba trazarse una estrategia de prueba adecuada, acorde a la metodología de desarrollo empleada para la construcción del producto. Las pruebas de software permiten detectar errores de diseño, implementación, funcionalidad, usabilidad y seguridad. La detección temprana de errores permitirá que estos sean arrastrados de una etapa a otra, que se cumpla el cronograma establecido para el desarrollo, el despliegue y la entrega de la solución al usuario final, y la disminución de costos del proyecto (recursos humanos, técnicos y financieros y tiempo). Para el desarrollo de pruebas al portal web de la ONAT se

emplearás dos Herramientas informáticas, la primera de ellas Acunetix Web Vulnerability Scanner 9.5 permitirá comprobar el grado de robustez en cuestión de seguridad del sistema y la segunda JMeter, la capacidad de respuesta que provee el sistema. A continuación, se detallan elementos de ambas herramientas.

Acunetix Web Vulnerability Scanner 9.5

Es una herramienta automatizada que no solo constituye un escáner de vulnerabilidades web, sino que es una solución completa de pruebas de seguridad de aplicaciones web que se puede utilizar de forma independiente y como parte de entornos complejos. Acunetix escanea cualquier sitio o aplicación web que sea accesible a través de un navegador web y utilice el protocolo HTTP / HTTPS. Ofrece una solución sólida y única para analizar estas aplicaciones web personalizadas y listas para usar (Invicti, 2021). Al escoger Acunetix como una de las medidas de seguridad, se puede aumentar significativamente la postura de ciberseguridad y eliminar muchos riesgos de seguridad a un bajo costo de recursos.

JMeter 2.10

Es una herramienta que permite realizar pruebas de carga y estrés para llevar a cabo simulaciones sobre cualquier recurso de Software. Inicialmente diseñada para pruebas de estrés en aplicaciones web, hoy en día, su arquitectura ha evolucionado no sólo para llevar a cabo pruebas en componentes habilitados en Internet (HTTP), sino además en Bases de datos, programas en Perl, requisiciones FTP y prácticamente cualquier otro medio. Posee la capacidad de realizar desde una solicitud sencilla hasta secuencias de requisiciones que permiten diagnosticar el comportamiento de una aplicación en condiciones de producción (Díaz et al., 2008).

1.5 Conclusiones Parciales

- Definir categorías principales asociadas a la entidad y al objeto de estudio posibilitó comprender términos que son necesarios en la investigación.
- El estudio de Sistemas Homólogos tanto en el contexto internacional como nacional permitió encontrar características y funcionalidades necesarias a incorporar en el portal web.
- Se definieron las principales tecnologías y herramientas a utilizar en el desarrollo del portal.

CAPÍTULO 2: DESCRIPCIÓN DEL PORTAL WEB DE LA OFICINA PROVINCIAL DE LA ONAT DE VILLA CLARA

2.1 Introducción del capítulo

En el presente capítulo se describe la estructura que presenta el portal web, se realiza la especificación de los requisitos funcionales y no funcionales, a partir de la cual se elaboran los artefactos que documentan la ingeniería de la propuesta de solución. Además, se especifica el estilo arquitectónico para el portal web de la Oficina Provincial de la ONAT de Villa Clara, así como la topología definida para su despliegue en la institución.

2.2 Propuesta de Solución

El portal web de la Oficina Provincial de la ONAT de Villa Clara presentará en la portada contenidos variados, de interés para los contribuyentes. El sitio permitirá acceder a páginas interiores mediante enlaces de menú, el cual se encuentra en la región superior con varias secciones como “Quiénes Somos” y “Contribuyentes” donde se debe mostrar información general acerca de la institución y las personas naturales y jurídicas respectivamente. Existirá una sección llamada “Documentos” los cuales pueden ser leyes o formularios que el usuario puede visualizar y descargar. El sistema permitirá el acceso a “Directorio”, donde se encuentra la información de contacto de las oficinas municipales de Villa Clara. La página principal debe contener los bloques: documentos destacados, trámites, novedades y preguntas frecuentes y en el pie de página tendrá enlaces a sitios de interés y medios informativos pertenecientes a la provincia.

2.2.1 Usuarios y Premisos

Para contribuir a la seguridad de la información del sistema, Drupal permite asignar roles y permisos a los diferentes usuarios que visitan el portal web, otorgando a ellos ciertos privilegios y controlando las acciones que pueden realizar y las funcionalidades a las que pueden acceder. A continuación, se muestran los roles con los respectivos permisos existentes en el sitio.

- **Webmaster:** Usuario con permisos para gestionar noticias, leyes, formularios, preguntas frecuentes, enlaces a sitios de interés, videos. Puede modificar todos los contenidos existentes y crear artículos.
- **Usuario Anónimo:** Usuario que navega por el portal sin estar autenticado, solo tiene permiso de lectura en el portal.
- **Cliente:** Usuario que puede autenticarse en el portal y tiene permisos para publicar comentarios.

- **Administrador:** Responsable de la administración del portal, puede añadir usuarios, gestionar y asignar roles a un usuario y asignar permisos a un rol. Puede realizar todas las tareas que ejerce el Webmaster.

2.3 Ingeniería de requisitos

La ingeniería de requisitos cumple un papel primordial en el proceso de producción de software, ya que enfoca un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, el comportamiento del sistema. Es la fase de un proyecto software donde se definen las propiedades y la estructura del mismo; y que a la vez comprende el desarrollo y gestión de requisitos (Toro & Gálvez, 2016). Pueden clasificarse en dos categorías: funcionales y no funcionales.

Para la elicitación de requisitos se emplearon como fuentes de información: el estudio de sistemas homólogos y las necesidades de los implicados, en este caso la ONAT de Villa Clara como entidad cliente. Se utilizaron como técnicas para su obtención, la observación de campo, y una entrevista a la jefa del departamento de informática de la ONAT (Ver Anexo 1). Estas técnicas proporcionaron una manera directa de ver a los trabajadores en su ambiente y el modo en que realizan las tareas y ejecutan los procesos.

2.3.1 Requisitos Funcionales

Según Sommerville (2011) constituyen enunciados acerca de funciones que el sistema debe proveer al finalizar el proceso de desarrollo. Una función es descrita como un conjunto de entradas, comportamientos y salidas. Definen cómo debería reaccionar el sistema a entradas particulares y de cómo debería comportarse en situaciones específicas. Pueden clasificarse de la siguiente manera:

- Requisitos del usuario: son declaraciones, en lenguaje natural y en diagramas, de los servicios que se espera que el sistema proporcione y de las restricciones bajo las cuales debe funcionar.
- Requisitos del sistema: establecen con detalle las funciones, servicios y restricciones operativas del sistema. El documento de requisitos del sistema (algunas veces denominado especificación funcional) debe ser preciso. Debe definir exactamente qué es lo que se va a implementar.

A continuación, se muestra una tabla con los distintos requisitos funcionales.

Tabla 2. Requisitos Funcionales
Fuente: Elaboración Propia

Requisitos Funcionales				
No	Nombre	Prioridad	Complejidad	Descripción
RF1.	Registrar Cuenta de Usuario	Alta	Alta	El sistema debe permitir al usuario crearse una cuenta (Datos de registro:

				Nombre, Apellidos, CI, Provincia, Municipio, Usuario y Contraseña, Imagen: opcional (".jpg", ".png").
RF2.	Eliminar Cuenta de Usuario	Media	Media	El sistema debe permitir al usuario eliminar su cuenta.
RF3.	Notificar activación de la cuenta	Alta	Alta	El sistema debe permitir notificar al usuario sobre la activación de su cuenta.
RF4.	Autenticar Usuario	Alta	Media	El sistema debe permitir al usuario autenticarse (Datos de autenticación: Usuario y Contraseña).
RF5.	Emitir reporte de los usuarios registrados	Media	Media	El sistema debe permitir al usuario con rol de administrador, visualizar en forma de listado, los usuarios registrados (Datos a mostrar: Nombre, Apellidos, CI, Provincia, Municipio, Usuario, Imagen).
RF6.	Modificar usuario	Media	Alta	El sistema debe permitir modificar los datos del perfil del usuario (Datos del usuario: Nombre, Apellidos, CI, Provincia, Municipio, Usuario, Contraseña, Imagen (".jpg", ".png")).
RF7.	Mostrar información de usuario	Media	Media	El sistema debe permitir visualizar la información del usuario registrado (Datos del usuario: Nombre, Apellidos, CI, Provincia, Municipio, Usuario, Imagen).
RF8.	Crear Rol	Alta	Alta	El sistema debe permitir al usuario con permisos de administración, crear un rol (Datos: Nombre del rol, Descripción)
RF9.	Eliminar rol	Baja	Baja	El sistema debe permitir al usuario con permisos de administración, eliminar un rol.
RF10.	Visualizar información del rol	Media	Media	El sistema debe permitir al usuario con permisos de administración, visualizar la información referente a los roles de los

				usuario (Datos: Nombre del rol, Descripción).
RF11.	Actualizar rol	Media	Media	El sistema debe permitir al usuario con permisos de administración, modificar la información de un rol (Datos: Nombre del rol, Descripción).
RF12.	Asignar permiso a un rol	Alta	Alta	El sistema debe permitir al usuario con permisos de administración, asignar permisos a un rol.
RF13	Asignar rol a usuario	Alta	Alta	El sistema debe permitir al usuario con permiso de administración, asignar un rol a un usuario.
RF14.	Listar roles	Media	Alta	El sistema debe permitir al usuario con permiso de administración, listar los roles definidos para el sistema.
RF15.	Realizar búsqueda	Media	Alta	El sistema debe permitir al usuario realizar una búsqueda en todo el sitio.
RF16.	Añadir imágenes al banner	Media	Media	El sistema debe permitir al usuario con permisos de Webmaster añadir las imágenes necesarias. Deben añadirse como mínimo 2 imágenes (".jpg", ".png").
RF17.	Modificar imágenes del banner	Media	Media	El sistema debe permitir al usuario con permisos de Webmaster modificar las imágenes añadidas.
RF18.	Mostrar imágenes en el banner publicitario	Alta	Media	El sistema debe mostrar en la portada un banner con un carrusel de imágenes.
RF19.	Añadir Noticia	Alta	Alta	El sistema debe permitir al usuario con permisos de Webmaster añadir noticias (Datos: Título, Fecha, Resumen, Descripción, Fuente, Autor e Imagen (".jpg", ".png")).
RF20.	Editar Noticia	Media	Media	El sistema debe permitir al usuario con permisos de Webmaster editar las noticias añadidas.

RF21.	Mostrar Noticia	Alta	Media	El sistema debe mostrar las noticias añadidas (Datos: Título, Fecha, Resumen, Descripción, Fuente, Autor e Imagen (“.jpg”, “.png”)).
RF22.	Eliminar Noticia	Baja	Baja	El sistema debe permitir al usuario con permisos de Webmaster eliminar noticias.
RF23.	Crear comentario	Media	Alta	El sistema debe permitir al usuario con permisos de cliente, crear comentarios.
RF24.	Modificar Comentario	Baja	Media	El sistema debe permitir al usuario con permisos de cliente, modificar los comentarios añadidos.
RF25.	Mostrar Comentario	Media	Alta	El sistema debe mostrar los comentarios creados.
RF26.	Eliminar Comentario	Baja	Baja	El sistema debe permitir al usuario con permisos de administración, eliminar comentarios.
RF27.	Añadir videos como multimedia	Media	Alta	El sistema debe permitir al usuario con permisos de Webmaster, añadir videos de interés (Formato: “.mp4”).
RF28.	Modificar video	Media	Media	El sistema debe permitir al usuario con permisos de Webmaster, modificar videos añadidos.
RF29.	Mostrar video	Media	Media	El sistema debe mostrar los videos añadidos.
RF30.	Eliminar video	Baja	Baja	El sistema debe permitir al usuario con permisos de Webmaster, eliminar videos.
RF31.	Añadir enlace a sitios.	Media	Alta	El sistema debe permitir al usuario con permisos de Webmaster, añadir enlaces a sitios de interés y medios informativos de la provincia (Datos: Nombre, Tipo, url).

RF32.	Mostrar listado de enlaces a sitios	Media	Alta	El sistema debe mostrar dos listas, una con enlaces a sitios de interés y otra con enlaces a sitios de medios informativos.
RF33.	Modificar enlace a sitios	Media	Media	El sistema debe permitir al usuario con permisos de Webmaster, modificar enlaces a sitios de interés y medios informativos de la provincia.
RF34.	Eliminar enlace a sitios.	Media	Baja	El sistema debe permitir al usuario con permisos de Webmaster, eliminar enlaces.
RF35.	Añadir Organigrama	Media	Alta	El sistema debe permitir al usuario con permisos de Webmaster, añadir organigrama de la organización (Dato: Imagen (“.jpg”, “.png”).
RF36.	Modificar Organigrama	Baja	Media	El sistema debe permitir al usuario con permisos de Webmaster, modificar organigrama de la organización.
RF37.	Mostrar Organigrama	Media	Baja	El sistema debe mostrar el organigrama de la organización.
RF38.	Añadir Documento	Alta	Alta	El sistema debe permitir al usuario con permisos de Webmaster, añadir documentos (Datos: Título, Descripción, Tipo, Destacado y Archivo (“.pdf”, “.xls”).
RF39.	Modificar Documento	Media	Baja	El sistema debe permitir al usuario con permisos de Webmaster, modificar documentos.
RF40.	Mostrar Listado de Documentos	Alta	Media	El sistema debe mostrar un listado con los documentos añadidos.
RF41.	Descargar documento	Media	Alta	El sistema debe permitir a todos los usuarios descargar los documentos.
RF42.	Eliminar documento	Baja	Baja	El sistema debe permitir al usuario con permisos de Webmaster, eliminar documentos.

RF43.	Añadir Trámite	Alta	Alta	El sistema debe permitir al usuario con permisos de Webmaster, añadir trámites (Datos: Título, Contenido, Imagen (“.jpg”, “.png”).
RF44.	Modificar Trámite	Media	Media	El sistema debe permitir al usuario con permisos de Webmaster, modificar trámites.
RF45.	Mostrar listado de trámites	Alta	Alta	El sistema debe mostrar un listado con todos los trámites añadidos.
RF46.	Eliminar Trámite	Media	Baja	El sistema debe permitir al usuario con permisos de Webmaster, eliminar trámites.
RF47.	Mostrar contactos de la entidad	Media	Alta	El sistema debe mostrar los contactos de la entidad (Datos de contacto: Mapa, Nombre, Correo, Dirección Postal, Teléfono, Redes Sociales).
RF48.	Mostrar datos para Suscripción	Media	Media	El sistema debe mostrar el campo correo y botones para suscribirse o anular la suscripción.
RF49.	Realizar Suscripción	Media	Alta	El sistema debe permitir a los usuarios realizar suscripción (Dato: Correo).
RF50.	Anular Suscripción	Baja	Media	El sistema debe permitir a los usuarios anular suscripción (Dato: Correo).
RF51.	Enviar noticias a usuarios suscritos	Media	Alta	El sistema debe enviar un correo con las noticias diarias a los usuarios suscritos.
RF52.	Añadir preguntas frecuentes	Alta	Alta	El sistema debe permitir al usuario con permisos de Webmaster, añadir preguntas frecuentes (Datos: Pregunta, Respuesta).
RF53.	Modificar preguntas frecuentes	Media	Baja	El sistema debe permitir al usuario con permisos de Webmaster, modificar preguntas frecuentes añadidas.
RF54.	Mostrar preguntas frecuentes	Alta	Alta	El sistema debe mostrar las preguntas frecuentes añadidas.

RF55.	Eliminar preguntas frecuentes	Media	Baja	El sistema debe permitir al usuario con permisos de Webmaster, eliminar preguntas frecuentes.
RF56.	Añadir Sede	Media	Alta	El sistema debe permitir al usuario con permisos de Webmaster, añadir la sede de cada municipio (Datos: Municipio, Teléfono, Dirección).
RF57.	Modificar Sede	Media	Baja	El sistema debe permitir al usuario con permisos de Webmaster, modificar las sede añadidas (Datos: Municipio, Teléfono, Dirección).
RF58.	Eliminar Sede	Media	Baja	El sistema debe permitir al usuario con permisos de Webmaster, eliminar las sedes.

2.3.2 Requisitos no funcionales

Los requisitos no funcionales, pueden relacionarse con propiedades emergentes del sistema, como fiabilidad, tiempo de respuesta y uso de almacenamiento y no con los servicios específicos que el sistema entrega a sus usuarios (Sommerville, 2011). En ocasiones definen restricciones o condiciones que impone el cliente al software que necesita, pueden ser sobre la implementación del sistema, tiempo de entrega del programa, el lenguaje o la cantidad de usuarios. A continuación, se muestran los requisitos no funcionales que presenta el portal web.

❖ Apariencia

RnF 1 Se debe emplear los colores rojo y blanco e interfaces sencillas.

❖ Software

RnF 2: El sistema debe visualizarse en los principales navegadores, a partir de las versiones: Chrome v.45, Firefox v.38, Safari v.9, Opera v.30.

RnF 3: El sistema debe ejecutarse en los sistemas operativos Windows y Linux.

RnF 4: Lenguaje de programación PHP 7.4.19.

RnF 5: Sistema de Gestión de Contenidos Drupal 9.

RnF 6: Servidor de base de datos MySQL en su versión 10.4.

RnF 7: Servidor web Apache versión 2.4.47.

❖ Hardware

RnF 8: El servidor de base de datos debe poseer una capacidad mínima de 20 GB.

RnF 9: El servidor de aplicaciones web debe poseer una capacidad mínima de 80 GB.

RnF 10: Los servidores web y de base de datos deben poseer como mínimo 1 GB de memoria RAM.

RnF 11: 2 GB de memoria RAM como mínimo.

RnF 12: Un procesador de 2 GHZ como mínimo.

❖ **Legales**

RnF 13: Uso de la licencia GNU/GPL para Drupal.

RnF 14: Uso de la licencia Apache License.

RnF 15: Uso de la licencia PHP License.

2.3.3 Arquitectura de Información

A partir de los elementos descritos con anterioridad y que son el resultado del levantamiento de requisitos, se propone la siguiente arquitectura de información para la propuesta de solución.

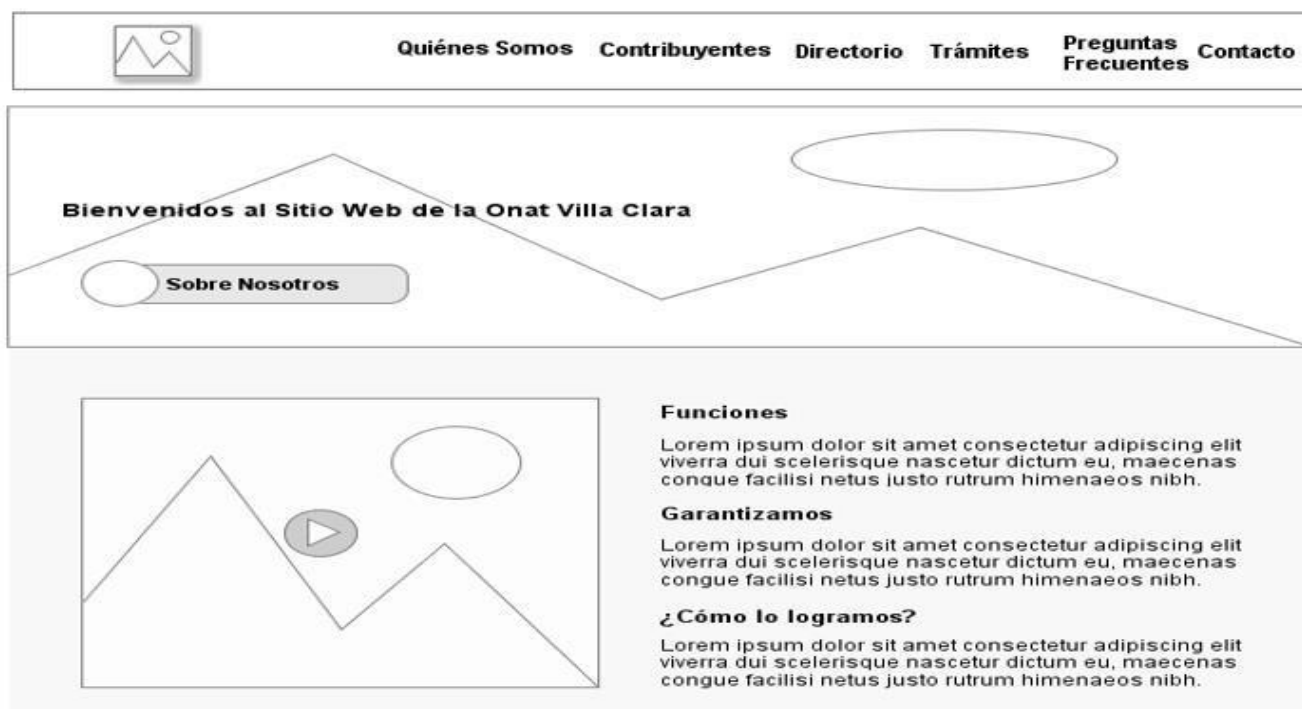


Ilustración 1. Página Principal. Parte 1
Fuente: Elaboración Propia

Documentos Destacados



Leyes

Ley No.113 del Sistema Tributario, el Decreto No.308

Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum

 [Descargar](#)

Ley No.118 de Inversión Extranjera

Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum

 [Descargar](#)



Formularios

Modelo de Ingresos al presupuesto

Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum

 [Descargar](#)

DECLARACIÓN JURADA IMPUESTO SOBRE TRANSMISIÓN DE BIENES Y HERENCIAS

Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum

 [Descargar](#)

[Ver Todos](#)

Trámites

INSCRIPCIÓN

Lorem ipsum dolor sit amet consectetur adipiscing

[Leer más...](#)

APLAZAMIENTOS

Lorem ipsum dolor sit amet consectetur adipiscing

[Leer más...](#)

RECLAMACIONES

Lorem ipsum dolor sit amet consectetur adipiscing

[Leer más...](#)

Novedades

Lorem ipsum dolor sit amet consectetur

Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum

[Leer más...](#)

Lorem ipsum dolor sit amet consectetur

Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum

[Leer más...](#)

Preguntas Frecuentes

¿Lorem ipsum dolor sit amet consectetur adipiscing elit viverra?

Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum eu, maecenas congue facilisi netus justo rutrum himenaeos nibh mi purus praesent.


¿Lorem ipsum dolor sit amet consectetur adipiscing elit viverra?


Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum eu, maecenas congue facilisi netus justo rutrum himenaeos nibh mi purus praesent.


Ilustración 2. Página Principal. Parte 2

Fuente: Elaboración Propia

Contáctenos


Nuestra dirección


Correo


Teléfono

Asunto

Mensaje

Dirección	Sitios de Interés	Medios Informativos	Subscríbete
	Portal Tributario	Vanguardia	<input style="width: 100%; border: 1px solid #ccc; border-radius: 15px;" type="button" value="Subscribirse"/>
Teléfono	Cubadebate	CMHW	
Correo	Gaceta Oficial	Telecubanacán	
	UCLV		

Ilustración 3. Página Principal. Parte 3

Fuente: Elaboración Propia

Historias de Usuarios

Las historias de usuarios constituyen artefactos escritos en lenguaje natural. Son utilizadas para la especificación de los requisitos funcionales en el escenario 4 de la metodología de desarrollo AUP versión UCI. A continuación, se muestran las historias de usuarios.

Tabla 3. Historia de Usuario Registrar Usuario

Fuente: Elaboración Propia

Historia de Usuario	
Nombre de la Historia: Registrar Usuario	
Número: 1	Usuario: Cliente
Prioridad en negocio: Alta	Riesgo en desarrollo: Media


Tiempo Estimado: 24h	Tiempo Real: 12h	Iteración Asignada: 1
Programador Responsable: Ciara Ruiz Gutiérrez		
Descripción: El sistema permite a los usuarios registrarse.		
<p>Observaciones: Para que un usuario cliente pueda registrarse el en portal web debe completar el formulario que incluye los siguientes campos:</p> <ul style="list-style-type: none"> ▪ Nombre y Apellidos (obligatorio, campo de texto) ▪ Carnet de identidad (obligatorio, campo de texto) ▪ Provincia (obligatorio, campo de selección) ▪ Municipio (obligatorio, campo de selección) ▪ Usuario (obligatorio, campo de texto) ▪ Contraseña (obligatorio, campo de texto) ▪ Confirmar Contraseña (obligatorio, campo de texto) ▪ Imagen (opcional, campo de texto) <p>El sistema verifica si el usuario existe, si ha llenado los campos obligatorios, si los datos son correctos y si las validaciones son correctas, el usuario se crea satisfactoriamente.</p>		
Prototipo de Interfaz gráfica		
		


Tabla 4. Historia de Usuario Añadir Noticia
Fuente: Elaboración Propia

Historia de Usuario		
Nombre de la Historia: Añadir Noticia		
Número: 2	Usuario: Webmaster	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media	
Tiempo Estimado: 24h	Tiempo Real: 12h	Iteración Asignada: 1

Programador Responsable: Ciara Ruiz Gutiérrez
Descripción: El sistema permite añadir noticias.
<p>Observaciones: El usuario con permisos de Webmaster puede añadir noticias llenando los siguientes campos:</p> <ul style="list-style-type: none"> ▪ Título (obligatorio, campo de texto) ▪ Fecha (obligatorio, campo de texto) ▪ Resumen (obligatorio, campo de texto) ▪ Descripción (opcional, campo de texto) ▪ Imagen (obligatorio, campo de texto) ▪ Fuente (opcional, campo de texto) ▪ Autor (opcional, campo de texto) Todos los días deben añadirse noticias.
<p>Prototipo de Interfaz gráfica</p> <p>El prototipo muestra un formulario con los siguientes campos y elementos:</p> <ul style="list-style-type: none"> Título*: Campo de texto obligatorio. Fecha*: Campo de texto obligatorio. Resumen*: Campo de texto obligatorio. Descripción: Campo de texto opcional. Imagen*: Campo de texto obligatorio con un botón "Examinar..." adjunto. Fuente: Campo de texto opcional. Autor: Campo de texto opcional. Añadir: Botón para guardar la noticia.

Tabla 5. Historia de Usuario Añadir Documento
Fuente: Elaboración Propia

Historia de Usuario		
Nombre de la Historia: Añadir Documento		
Número: 3	Usuario: Webmaster	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media	
Tiempo Estimado: 24h	Tiempo Real: 12h	Iteración Asignada: 1

Programador Responsable: Ciara Ruiz Gutiérrez
Descripción: El sistema permite que se puedan añadir documentos de interés para los contribuyentes.
<p>Observaciones: El usuario con permisos de Webmaster puede añadir documentos de interés para los contribuyentes, que pueden ser leyes o formularios. Debe llenar los siguientes campos:</p> <ul style="list-style-type: none"> ▪ Título (obligatorio, campo de texto) ▪ Descripción (opcional, campo de texto) ▪ Tipo (obligatorio, campo de selección) ▪ Destacado (opcional, campo de marcación) ▪ Archivo (obligatorio, campo de texto)
<p>Prototipo de Interfaz gráfica</p>  <p>El prototipo muestra un formulario con el título 'Añadir Documentos'. Incluye los siguientes campos y elementos:</p> <ul style="list-style-type: none"> Título*: Campo de texto obligatorio. Descripción: Campo de texto opcional. Tipo*: Campo de selección obligatorio. <input type="checkbox"/> Destacado: Opción de marcación. Archivo*: Campo de texto obligatorio con un botón 'Examinar...' para seleccionar archivos. Añadir: Botón para guardar el documento.

2.4 Análisis y Diseño

2.4.1 Arquitectura de Software

La arquitectura 5-capas es una técnica que se aplica al desarrollo web, permite separar en varias capas, los elementos que contiene el sistema, de manera muy fácil por la misma naturaleza y evolución de esta tecnología. La utilización de esta arquitectura en Drupal permite la división de los contenidos en 5 elementos fundamentales: datos, módulos, bloques y menús, permisos de usuarios y plantillas (Drupal, 2021). Cada capa presenta una responsabilidad específica.

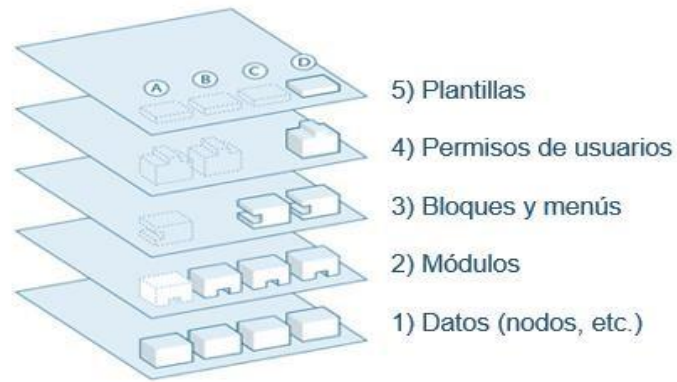


Ilustración 4. Arquitectura 5-capas

Fuente: (*Drupal*, 2021)

- ✓ **Datos:** Los nodos constituyen fragmentos de contenidos en un sitio. Puede ser una página de información, un blog o un comunicado de prensa. Las imágenes, videos o contenedores para recoger las páginas con acceso restringido a determinados usuarios, también pueden ser nodos. Al realizar la instalación de Drupal, existen dos tipos de nodos: artículo y página básica. En el portal se han añadido otros como, Documento, Noticia, Directorio, Preguntas Frecuentes, entre otros.
- ✓ **Módulos:** Conjunto de archivos PHP, JavaScript y/o CSS que amplían las características del sitio y agregan determinadas funcionalidades. Cada módulo que se instala aumenta el tiempo necesario para generar páginas en el sitio, por lo que se deben desinstalar los que no son necesarios. Para el desarrollo del portal web se han instalado los siguientes módulos: *Views*, *Simplenews*, *Security kit*, *Honeypot*, entre otros.
- ✓ **Bloques y Menús:** Los bloques constituyen contenedores que permiten ubicar los contenidos en las regiones definidas en el tema web y los menús son una lista horizontal o vertical que contiene enlaces a páginas interiores o bloques. El portal de la ONAT contiene un Menú Principal y bloques como banner publicitario, documentos destacados, novedades, entre otros.
- ✓ **Permisos de Usuarios:** Drupal dispone de un sistema de permisos completo que permite cambiar de una manera muy sencilla las operaciones que realizan los usuarios. Permite asignar permisos y roles a los usuarios creados en el sistema. Los roles que se proponen para el portal son: Administrador, Webmaster, Usuario Anónimo y Cliente.
- ✓ **Plantillas:** Conjunto de archivos que definen la apariencia visual del sitio. Determinan el marcado HTML y el estilo CSS que envuelve el contenido. Drupal permite utilizar plantillas gratuitas, pero además permite la implementación del motor de plantillas Twig. Para el desarrollo de la propuesta de solución se implementa el tema Onat.

2.4.2 Modelado del diseño

Diagrama de clases del diseño con estereotipos web

Los diagramas de clases utilizan elementos de sistemas para representar estados gráficamente y al desglosar las estructuras hasta la instancia más pequeña, mostrando las clases, sus atributos, operaciones (o métodos), y las relaciones entre los objetos. Son idóneos para visualizar arquitecturas de software detalladas de las que pueden derivarse pasos concretos de programación (Riesco, 2020). Los diagramas de clases con estereotipos web presentan las mismas peculiaridades de un diagrama de clases común, con la excepción de que son utilizados en el desarrollo web. A continuación, se muestra el diagrama de clases para el escenario Gestionar Documento.

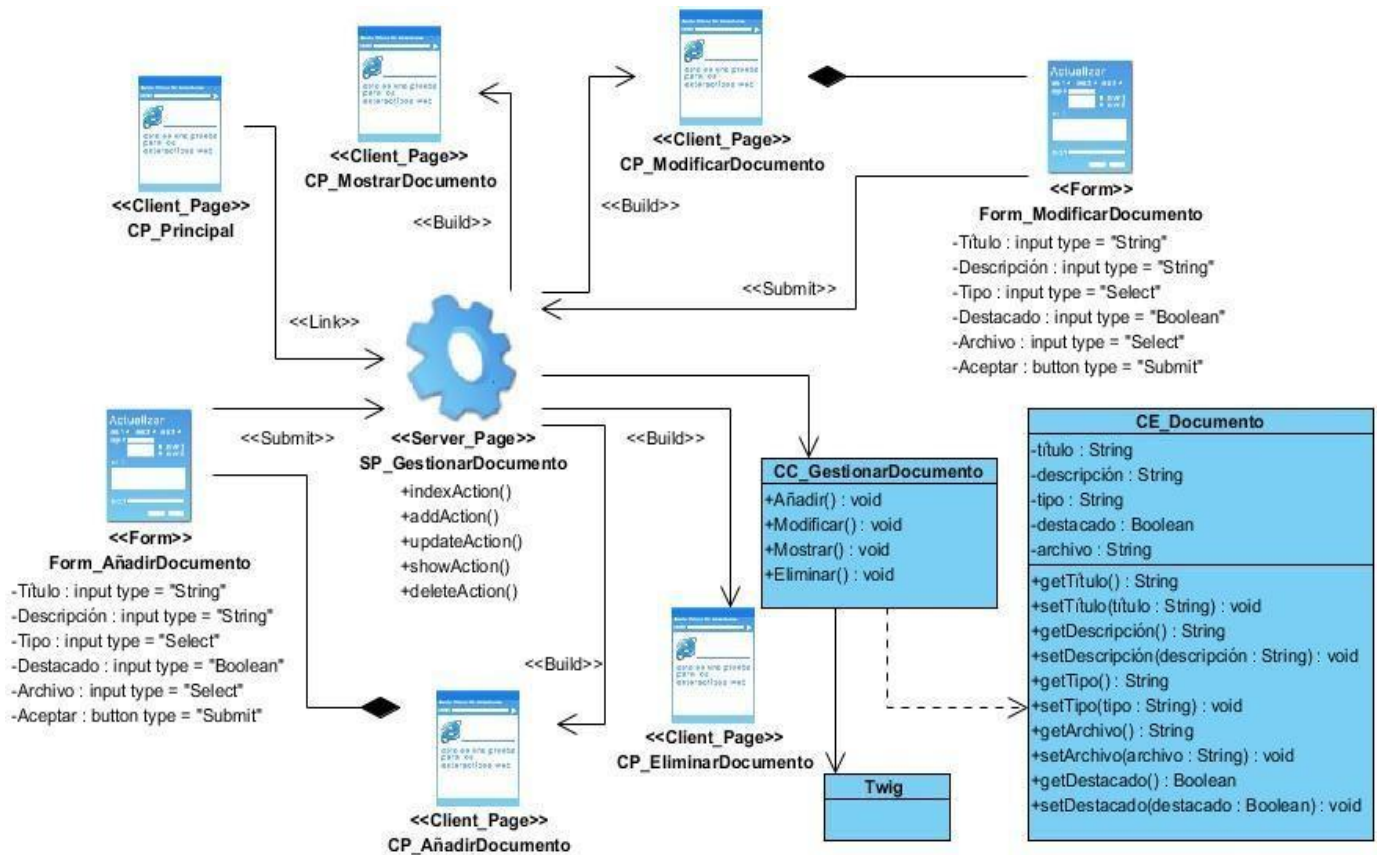


Ilustración 5. Diagrama de clases
Fuente: Elaboración Propia

Diagrama de Secuencia

Los diagramas de secuencia muestran la interacción de un conjunto de objetos de una aplicación a través del tiempo, en el cual se indican los módulos o clases que forman parte del programa y las llamadas que hacen cada uno de ellos para realizar una tarea determinada, por esta razón permite observar la perspectiva cronológica de las interacciones (Cevallos, 2015). A continuación, se muestra el diagrama de secuencia para el requisito Añadir Documento.

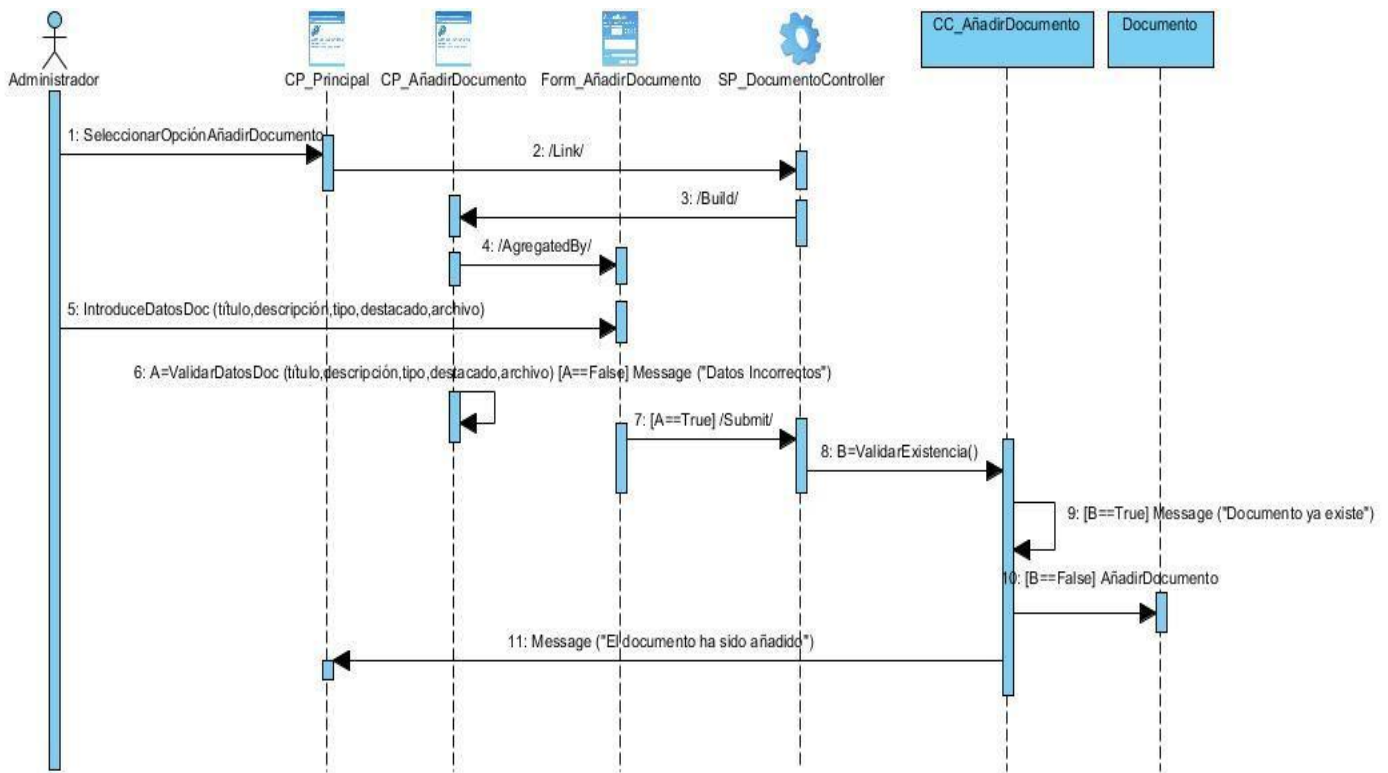


Ilustración 6. Diagrama de Secuencia
Fuente: Elaboración Propia

Patrones de Diseño

Un patrón de diseño es una descripción o plantilla que permite resolver un problema determinado en el desarrollo de un software. El uso de estos patrones permite acelerar el proceso de desarrollo al proporcionar paradigmas probados y comprobados en el diseño de software. Estandarizar las soluciones a problemas comunes permite mejorar la comunicación entre desarrolladores, definiendo un lenguaje común donde se emplean nombres bien conocidos y entendidos a la hora de describir un problema de diseño y cómo abordar su solución (Quiroga, 2020). Drupal 9 utiliza para el desarrollo web los patrones Grasp y *Gang Of Four* (GOF) cuyo significado es pandilla de cuatro.

Los patrones Grasp constituyen patrones generales de software para la asignación de responsabilidades. Son una serie de buenas prácticas de aplicación recomendable en el diseño de software. A continuación, se muestran los patrones utilizados.

- **Experto en información:** Constituye el principio básico de asignación de responsabilidades. Indica que la responsabilidad de la creación de un objeto o la implementación de un método, debe recaer sobre la clase que conoce toda la información necesaria para crearlo. De este modo se obtiene un diseño con mayor cohesión y así la información se mantiene encapsulada (disminución del acoplamiento). A continuación, se muestra un ejemplo de cómo se evidencia

este patrón en Drupal en la clase `NodeTypeForm`, que contiene los elementos necesarios para crear un nodo.

- **Creador:** Ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases. La nueva instancia deberá ser creada por la clase que: tiene la información necesaria para realizar la creación del objeto, o usa directamente las instancias creadas del objeto, o almacena o maneja varias instancias de la clase.
- **Controlador:** Este patrón funciona como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado. Sugiere que la lógica de negocios debe estar separada de la capa de presentación, esto para aumentar la reutilización de código y a la vez tener un mayor control. Se recomienda dividir los eventos del sistema en el mayor número de controladores para poder aumentar la cohesión y disminuir el acoplamiento.

Los patrones GOF proponen soluciones a problemas concretos, no son teorías genéricas. Favorecen la reutilización de código. Ayudan a construir software basado en la reutilización, a construir clases reutilizables. Los propios patrones se reutilizan cada vez que se aplican. Se clasifican en patrones de creación, estructurales y de comportamiento (Guerrero et al., 2013). A continuación, se muestra los patrones utiliza Drupal dentro de las clasificaciones anteriormente mencionadas, junto con una imagen que describe el proceso a través de clases y dependencias entre ellas, además de mostrar un ejemplo de cómo sería este proceso en el código.

Patrones de creación:

- **Singleton:** Consiste en crear una instancia única de la clase y proporcionar un punto único de acceso al objeto. En Drupal los módulos y temas se piensan como objetos para llevar a cabo la gestión de dichos elementos, pues este CMS solo crea una instancia de un nodo, cuando este nodo no existe (Nikitin, 2021). En el portal web se evidencia este patrón cuando el Webmaster crea los tipos de contenidos noticia, documento y pregunta frecuente, el sistema verifica que no existan en la base de datos y posteriormente los crea.
- **Factory method:** Define una interfaz común para la creación de objetos. El tipo de objeto se puede cambiar según la configuración. Este método se utiliza en Drupal para hacer que el objeto funcione con la caché. Una de sus ventajas es que puede devolver objetos creados anteriormente por lo que ayuda a ahorrar recursos del sistema. Además, se pueden emplear diferentes cachés en diferentes subsistemas (Nikitin, 2021). Ejemplo de su uso en el portal es la acción de limpiar caché que puede realizar el administrador. A continuación, se muestra una imagen que indica la obtención de un objeto de tipo caché en la variable `$render_cache`, configurado en el archivo `settings.php`.


```

    /**
     * Creates a new ValidPathConstraintValidator instance.
     *
     * @param \Drupal\Core\Path\PathValidatorInterface $path_validator
     * The path validator.
     */
    public function __construct(PathValidatorInterface $path_validator) {
        $this->pathValidator = $path_validator;
    }

    /**
     * {@inheritdoc}
     */

```

Ilustración 7. Factory Method
Fuente: Elaboración Propia

- **Dependency injection:** Define objetos-servicios en Drupal que son responsables de funciones específicas. Permite distribuir la responsabilidad entre clases. Es una interfaz que se implementa, por lo que en el futuro se puede pasar al objeto otra implementación de dependencia sin ajustar el objeto en sí (Nikitin, 2021). Ejemplo de esto se muestra en la clase *PathValidator* al validar todas las rutas del sistema, implementando la interface *ContainerInjectionInterface*. A continuación, se muestra una imagen donde se utiliza el constructor de la clase para pasar los servicios al objeto *path_validator*.

```

// Clear render cache.
/** @var \Drupal\Core\Cache\MemoryBackend $render_cache */
$render_cache = $this->container->get('cache_factory')->get('render');
$render_cache->deleteAll();

```

Ilustración 8. Dependency injection
Fuente: Elaboración Propia

Patrones estructurales

- **Proxy:** Se aplica cuando en lugar de objetos reales es necesario usar objetos proxy, los cuales permiten la ejecución de código antes o después de las llamadas a los métodos de los objetos originales. También se puede administrar el acceso al objeto original usando este patrón. Este patrón se encuentra en Drupal en la clase *\Drupal\Core\Session\AccountProxy* del servicio "current_user". *AccountProxy* es un proxy de la clase *\Drupal\Core\Session\UserSession* e implementa todos sus métodos. De esta manera, se puede trabajar con objetos de la clase *AccountProxy* como con objetos de la clase *UserSession*. Este enfoque permite establecer y cambiar el usuario actual sin reiniciar el servicio "current_user" (Nikitin, 2021). Además, este

servicio es utilizado por todos los servicios como un único punto de entrada a la sesión del usuario actual.

Patrones de Comportamiento

- **Mediator:** Se utiliza para la implementación de la interacción entre los componentes de la aplicación. Es la base de la característica principal de Drupal: el sistema de ganchos. Este patrón proporciona una única interfaz que se utiliza para la comunicación entre diferentes partes del sistema. Los componentes- receptores no saben quién envió la solicitud, y los componentes remitentes no saben quién procesará su solicitud (Nikitin, 2021). Este patrón es utilizado en el portal en la clase ModuleHandler que permite una vez pasado el parámetro *\$implementation*, llamar al gancho correspondiente, ejemplo al guardar un nodo.
- **Visitor:** Se utiliza para separar las operaciones en el objeto en sí. Este patrón se puede encontrar en el análisis de la plantilla en Twig. Construye un árbol de elementos de la plantilla que son procesados por los objetos-visitantes. Si se necesita procesar el elemento, basta con agregar un visitante sin cambiar la clase del elemento. Si los elementos deben procesarse de manera diferente, el visitante implementa algunas variantes por sí mismo (Nikitin, 2021). Es muy adecuado cuando los elementos procesados rara vez se cambian. Si se actualiza la implementación del elemento, se puede requerir un ajuste de los visitantes.

Modelo de despliegue

Los diagramas de despliegue constituyen complementos de los diagramas de componentes que, unidos, proveen la vista de implementación del sistema. Describen la topología del sistema, la estructura de los elementos de hardware y el software que ejecuta cada uno de ellos. Permiten representar la distribución, entrega e instalación de las partes que constituye el sistema físico. Representan a los nodos y sus relaciones, los cuales forman la topología de hardware sobre el que se ejecuta el sistema y son conectados por asociaciones de comunicación tales como enlaces de red, conexiones TCP/IP (Ramírez, 2020). A continuación, se muestra el diagrama de despliegue.

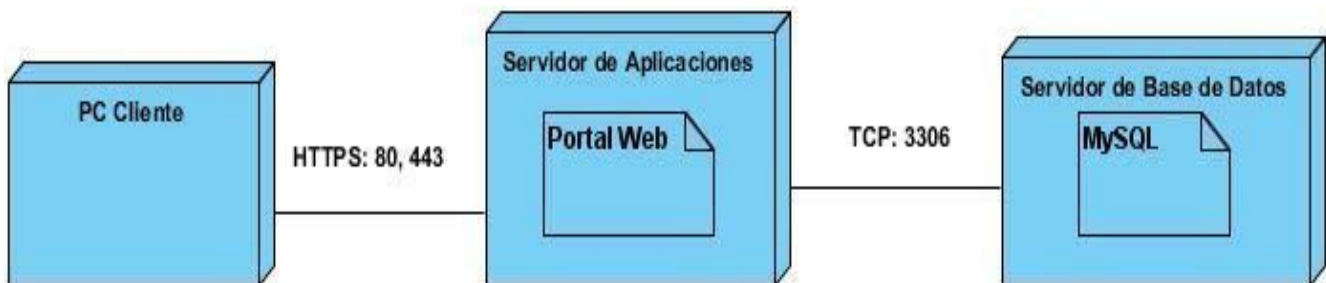


Ilustración 9. Diagrama de Despliegue
Fuente: Elaboración Propia

2.5 Conclusiones Parciales

- Definir la arquitectura de información y los requisitos funcionales y no funcionales permitió identificar los principales elementos y funcionalidades que debe tener el portal web.
- Los artefactos generados permitieron describir las funcionalidades más importantes del sistema para una mejor implementación.

CAPÍTULO 3. IMPLEMENTACIÓN Y VALIDACIÓN DEL PORTAL WEB

3.1 Introducción del capítulo

En el presente capítulo se definen los componentes y estándares de codificación utilizados en la implementación de la propuesta de solución. Se realizan las pruebas al portal web con el objetivo de identificar los principales errores existentes, permitiendo que al solucionarlos el sistema funcione correctamente.

3.2 Diagrama de Componentes

Describen los componentes de un software y sus dependencias entre sí, representando la estructura del código. Los componentes son la implementación en la arquitectura física de los conceptos y la funcionalidad descrita en la arquitectura lógica (clases, objetos, sus relaciones y colaboraciones). Los componentes son típicamente los archivos de implementación en el ambiente de desarrollo (navegador). A continuación, se muestra el diagrama de componentes.

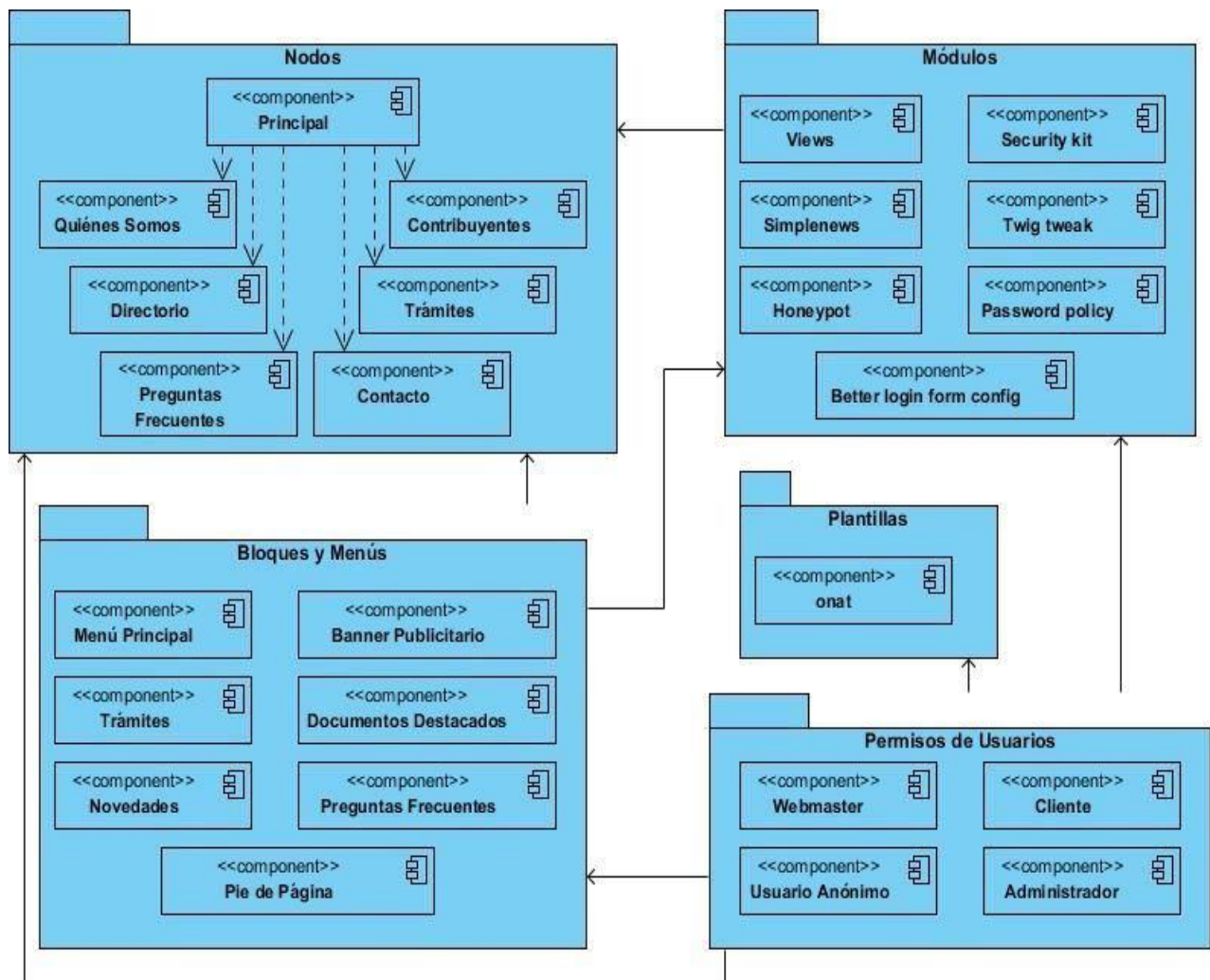


Ilustración 10. Diagrama de Componentes

Fuente: Elaboración Propia

3.3 Estándares de Codificación

- **Indentación:** Consiste en insertar espacios en blanco o tabuladores en determinadas líneas de código para facilitar su comprensión. En programación se utiliza Indentación para anidar elementos. A continuación, se muestra un ejemplo de Indentación.
- **Operadores:** Los operadores binarios (operadores que vienen entre dos valores), tales como +, -, =, !=, ==, >, etc. deben tener un espacio antes y después del operador, para mejorar la legibilidad. Los operadores unarios (operadores que operan en un solo valor), tales como ++, - no deben tener un espacio entre el operador y la variable o número en el que están operando (Drupal, 2021).

```
$url = NULL;
$image_link_setting = $this->getSetting('image_link');
// Check if the formatter involves a link.
if ($image_link_setting == 'content') {
    $entity = $items->getEntity();
    if (!$entity->isNew()) {
        $url = $entity->toUrl()->toString();
    }
}
elseif ($image_link_setting == 'file') {
    $link_file = TRUE;
}
```

Ilustración 11. Estándares: Operadores

Fuente: Elaboración Propia

- **Etiquetas de apertura y cierre de PHP:** Al emplear el lenguaje PHP, se deben utilizar las etiquetas <?php y ?>, y en ningún caso la versión corta <? y ?>. En general se omite la etiqueta de cierre de PHP (?>) al final de los archivos. module y .inc. Esta convención evita que se puedan quedar olvidados espacios no deseados al final del archivo (después de la etiqueta de cierre ?>), que serían identificados como salida HTML y podrían provocar un error muy típico, "Cannot modify header information-headers already sent by ...". (Drupal, 2021).

```
<?php

/**
 * @file
 * Add an URL formatter for image field
 */
use Drupal\Core\Link;
use Drupal\image\Entity\ImageStyle;
```

Ilustración 12. Estándares: Etiquetas de apertura y cierre de PHP

Fuente: Elaboración Propia

- **Uso de comillas:** Se pueden usar tanto las comillas simples ('cadena') como las comillas dobles ("cadena") para delimitar las cadenas de caracteres. Las comillas dobles son necesarias si se desean incluir variables dentro de las cadenas de texto. Por ejemplo, "<h1>\${title}</h1>".

También se recomienda el uso de comillas dobles cuando el texto puede incluir alguna comilla simple (Drupal, 2021).

```
$url_type = isset($variables['url_type']) ? $variables['url_type'] : 0;
if ($url_type == 1){
    $output = str_replace($GLOBALS['base_url'], '', $output);
}
elseif ($url_type == 2){
    $output = str_replace($GLOBALS['base_url'] . '/', '', $output);
}
```

Ilustración 13. Estándares: Uso de comillas

Fuente: Elaboración Propia

- **Uso de punto y coma (;) en código PHP:** Aunque PHP permite escribir líneas de código individuales sin el terminador de línea (;), como por ejemplo <?PHP print \$title?>. En Drupal es siempre obligatorio: <?PHP print \$title; ?>
- **Estructuras de control:** Con respecto a las estructuras de control, hay que tener en cuenta las siguientes normas:
 - Debe haber un espacio entre el comando que define la estructura (if, while, for, etc) y el paréntesis de apertura. Esto es así para no confundir las estructuras de control con la nomenclatura de las funciones.
 - La llave de apertura ({) se situará en la misma línea que la definición de la estructura, separada por un espacio.
 - Se recomienda usar siempre las llaves { } aun en los casos en que no sea obligatorio su uso (una sola "línea" de código dentro de la estructura de control).
 - Las estructuras else y elseif se escribirán en la línea siguiente al cierre de la sentencia anterior (Drupal, 2021).

```
switch ($this->getSetting('url_type')) {
    case 2:
        $summary[] = $this->t('Use relative path');
        break;

    case 1:
        $summary[] = $this->t('Use absolute path');
        break;

    case 0:
        $summary[] = $this->t('Use full URL');
        break;
}
```

Ilustración 14. Estándares: Estructuras de control

Fuente: Elaboración Propia

- **Arreglos:** Los valores dentro de un array (o matriz) se deben separar por un espacio (después de la coma que los separa). El operador => debe separarse por un espacio a ambos lados. Cuando la línea de declaración del array supera los 80 caracteres, cada elemento se debe

escribir en una única línea, identándolo una vez (2 espacios). En este último caso, la coma de separación del último elemento también se escribirá, aunque no existan más elementos. De esta forma se evitan errores al añadir nuevos elementos al vector (*Drupal, 2021*).

```
$link_types = [  
  'content' => $this->t('Content'),  
  'file' => $this->t('File'),  
];  
$element['image_link'] = [  
  '#title' => $this->t('Link image to'),  
  '#type' => 'select',  
  '#default_value' => $this->getSetting('image_link'),  
  '#empty_option' => $this->t('Nothing'),  
  '#options' => $link_types,  
];
```

Ilustración 15. Estándares: Arreglos

Fuente: Elaboración Propia

- **Funciones:** Los nombres de las funciones deben estar escritos en minúsculas y las palabras separadas por guion bajo. Además, se debe incluir siempre como prefijo el nombre del módulo o tema, para evitar así duplicidad de funciones. En su declaración, después del nombre de la función, el paréntesis de inicio de los argumentos debe ir sin espacio. Cada argumento debe ir separado por un espacio, después de la coma del argumento anterior (*Drupal, 2021*).

```
/**  
 * Implements hook_help().  
 */  
function organigrams_help($route_name, RouteMatchInterface $route_match) {
```

Ilustración 16. Estándares: Funciones

Fuente: Elaboración Propia

3.4 Pruebas de Software

Las pruebas del software son el proceso de ejecución de un programa bajo condiciones controladas, evaluando y registrando los resultados con la intención de descubrir errores en el sistema. El objetivo fundamental es diseñar pruebas que sistemáticamente saquen a la luz diferentes clases de errores haciéndolo con la menor cantidad de tiempo y esfuerzos (*Pressman, 2002*).

➤ Pruebas funcionales

Según Sánchez (2015) se basan en las funcionalidades de un sistema, las cuales son descritas en la especificación de requisitos. La funcionalidad representa la capacidad del producto de software para proporcionar funciones que satisfacen las necesidades declaradas e implícitas, cuando el producto se utiliza en condiciones específicas. Se orientan en el comportamiento externo de un producto o aplicativo software, en las pruebas de caja negra.

Método de caja negra

“Las pruebas de caja negra son las que se le aplica a la interfaz del software. Estas pruebas examinan los aspectos funcionales del sistema que tienen poca relación con la estructura de la lógica interna del software. Se prueban las respuestas lógicas del software y la colaboración entre componentes” (Pressman, 2010). Estas pruebas se concentran en los requisitos funcionales del software. Le permiten al ingeniero de software derivar conjuntos de condiciones de entrada que ejercitarán por completo todos los requisitos funcionales de un programa.

Tabla 6. Caso de Prueba basado en el requisito Añadir Documento
Fuente: Elaboración Propia

Escenario	Título	Descripción	Archivo	Tipo	Respuesta del Sistema	Flujo Central
EC 1	V	V	V	V	El sistema almacena los datos insertados y muestra un mensaje de confirmación.	1.-El usuario accede al bloque superior Contenido/+ Agregar Contenido o
Añadir documento de la manera correcta	Título del documento	Insertar la descripción del documento	Subir un archivo que cumpla con las restricciones	Seleccionar el tipo de documento		
EC 2	NA	NA	I	NA	Solo se permiten archivos con las siguientes extensiones: .xlsx, .docx	Agregar Contenido y el sistema muestra todos los tipos de contenidos que pueden ser añadidos. 2.-El usuario selecciona el tipo de contenido Documento.
Añadir Documento de la manera incorrecta	Título del documento	Insertar la descripción del documento	I	Subir un archivo que no cumpla con las restricciones de formato		
			I	Subir un archivo que no cumpla con las restricciones de tamaño		
			I			

			Subir un archivo con formato falso			3.-El sistema muestra un formulario
EC 3	I	I	I	NA	El sistema no almacena la información y muestra los siguientes mensajes de error: "El campo Título es obligatorio." "El campo Archivo es obligatorio." "El campo Tipo es obligatorio."	para que el usuario introduzca la información.
Añadir documento con campos vacíos	Vacío	Vacío	Vacío	Seleccionar el tipo de documento		4.- El usuario introduce la información y presiona el botón: "Guardar".

Tabla 7. Caso de Prueba de Aceptación Añadir Documento
Fuente: Elaboración Propia

Caso de prueba Añadir Documento	
Código de caso de prueba: 1	Nombre de Historia de Usuario: Añadir Documento
Nombre de la persona que realiza la prueba: Ciara Ruiz Gutiérrez	
Descripción de la prueba: Prueba a la funcionalidad Añadir Documento .	
Entrada / Pasos de la ejecución: La entrada consta de la introducción de los datos en los siguientes campos: <ul style="list-style-type: none"> ▪ Título: Resolución 222 del BCC ▪ Descripción: Establece las normas para y el uso y operatoria bancaria para operaciones de compra y venta de bienes y prestación de servicios en moneda libremente convertible. ▪ Archivo: Resolución 222 del BCC.docx 	

- Tipo: Ley

Resultado esperado: Se añade el documento.

Evaluación de la prueba: Satisfactoria.

Entre las no conformidades detectadas en el proceso de pruebas funcionales se encuentran:

- Validación
- Opciones que no funcionan
- Ortografía
- Errores de estructuración de los contenidos mostrados en las vistas
- Errores de estructuración de los contenidos mostrados en las vistas.

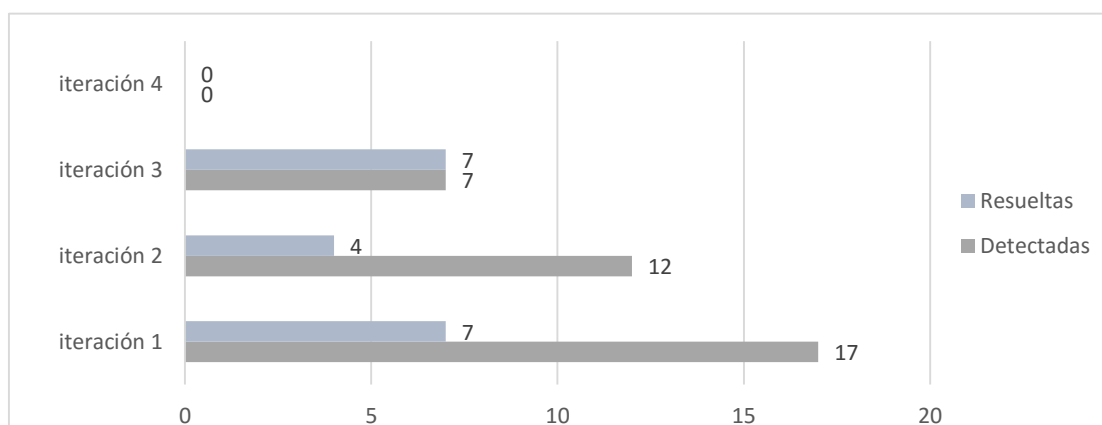


Ilustración 17. Resultados Pruebas Funcionales
Fuente: Elaboración Propia

➤ Pruebas de seguridad

Se pueden definir como el conjunto de actividades que se llevan a cabo para encontrar fallas y vulnerabilidades en aplicaciones web, buscando disminuir el impacto de ataques a ellas y pérdida de información importante. Permiten asegurar la confidencialidad, disponibilidad e integridad de los datos y funciones que maneja el software, teniendo en cuenta el impacto que pueden tener estas fallas según el contexto empresarial (Díaz, 2019).

Para realizar las pruebas de seguridad se emplea la herramienta Acunetix Web Vulnerability Scanner 11.0, donde se establecen 4 criterios para definir los errores (Alto, Medio, Bajo e Informativo). Al realizar la primera iteración se detectaron 117 no conformidades, divididas en una de nivel alto, 8 de nivel medio, 89 de nivel bajo y 19 informativas. Luego de instalar y configurar los módulos de seguridad *Security kit*, *Honeypot*, *Password Policy* se corrigieron 54 no conformidades. Al realizar la

segunda iteración se corrigieron 36 y en la tercera 27. Al ejecutar una cuarta iteración no fue detectada ninguna vulnerabilidad. A continuación, se muestra un gráfico expone los resultados anteriormente descritos.

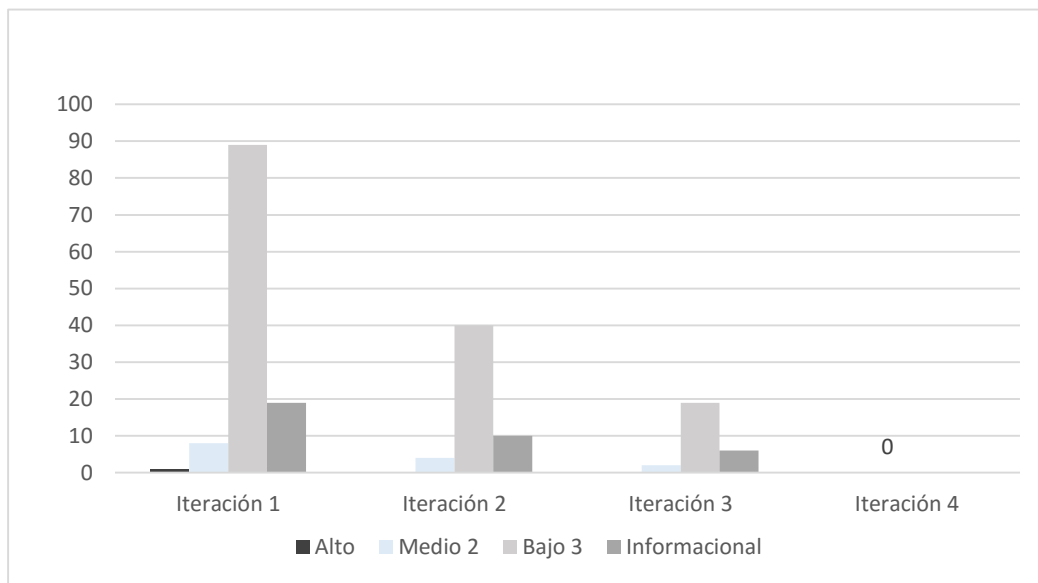


Ilustración 18. Resultados de las pruebas de seguridad
Fuente: Elaboración Propia

➤ Pruebas de carga y estrés

Pruebas de carga: Consisten en la medición del comportamiento del sistema para aumentar la carga del mismo, ya sea mediante el número de peticiones que se realizan a una web al mismo tiempo o el número de usuarios que trabajan simultáneamente (Sánchez, 2015).

Pruebas de estrés: Se realizan con el objetivo de revelar los errores de la aplicación bajo condiciones de carga máxima; mediante la ejecución de un número de usuarios muy superior al esperado o bien por la substracción de recursos. Tiene como finalidad determinar la robustez de una aplicación cuando la carga es extrema, así como el límite real de la aplicación en cuanto a número de usuarios concurrentes, el número de transacciones por segundo, entre otros. En este tipo de pruebas los tiempos de respuesta de la aplicación no son importantes y tienden a ser ignorados (Delgado, 2015).

Para realizar las pruebas de carga y estrés al Portal Web de la Oficina Provincial de la ONAT de Villa Clara se propone utilizar la herramienta JMeter en su versión 2.10, el cual es un software de código abierto, diseñado para pruebas de carga de comportamiento funcionales y la medición del rendimiento. Es necesario definir las condiciones del escenario teniendo en cuenta el software y hardware que posee la PC servidor donde se encuentra ubicado el sitio, así como las variables que se tienen en cuenta para realizar las pruebas.

Hardware

- **Tipo de procesador:** Inter(R) Core(TM) i3-5020U CPU @2.20GHz 2.19GHz
- **Memoria:** 4 GB de RAM
- **Disco Duro:** 1024 GB

Software

- **Sistema Operativo:** Windows 10 de 64bits
- **Servidor Web:** Apache
- **Servidor de base de datos:** MySQL

A continuación, se muestran los resultados de las pruebas aplicadas para 50, 100 y 200 usuarios concurrentes.

Etiqueta	#Muestras	Media	Mediana	90% Line	Min	Máx	%Error	Rendimiento	kb/sec	Sent kb/sec
Página Principal	200	3	4	5	2	3	0,00%	4.0/sec	62.9	0.91
Quiénes Somos	200	2	3	4	1	1	0,00%	4.0/sec	38.3	0.50
Contribuyentes	200	1	1	3	2	2	0,00%	4.0/sec	34.4	0.53
Directorio	200	2	2	5	2	1	0,00%	4.0/sec	30.3	0.51
Trámites	200	2	2	5	1	1	0,00%	4.0/sec	26.6	0.50
Total	1000	2	2.4	4	1	3	0,00%	20.1/sec	166.1	2.95

Ilustración 19. Resultados de las pruebas de carga para 200 usuarios concurrentes.

Fuente: Elaboración Propia

➤ Pruebas de usabilidad

Las pruebas de usabilidad son un servicio de aseguramiento de calidad que consiste en probar un producto y detectar no conformidades sobre su facilidad de uso. El objetivo principal de estas pruebas es identificar los problemas de usabilidad, para mejorar el producto y la satisfacción de sus clientes, proporciona retroalimentación para mejorar el diseño, valora en qué medida se cumplen los objetivos de los usuarios y la organización y exponen cuantitativamente el grado de usabilidad que presenta el producto probado (Valle et al., 2017). La realización de estas pruebas se efectuará mediante la lista de chequeo.

En correspondencia con lo anteriormente planteado y con el objetivo de determinar el grado de usabilidad que posee la propuesta de solución, se define para su evaluación una lista de chequeo compuesta por 47 elementos de comprobación que responden a 8 características que debe poseer el sistema (Ver anexo 5). En la primera iteración el resultado de la prueba fue regular, debido a que son

encontradas 8 no conformidades. En la segunda iteración se solucionan todas las no conformidades, resultando satisfactoria la prueba. A continuación, se muestra una gráfica con los resultados obtenidos.

3.5 Validación del Portal web

Técnica de ladov

Constituye una vía indirecta para el estudio de la satisfacción, ya que los criterios que se utilizan se fundamentan en las relaciones que se establecen entre tres preguntas cerradas que se intercalan dentro de un cuestionario. Estas tres preguntas se relacionan a través de lo que se denomina el Cuadro Lógico de ladov e indica la posición de cada sujeto en una escala (Fabre, 2014). Esta técnica es empleada en la investigación con el objetivo de evaluar el grado de satisfacción de los contribuyentes con el Portal web desarrollado para la Oficina Provincial de la ONAT de Villa Clara.

Tabla 8. Cuadro Lógico de ladov para usuarios potenciales
Fuente: Elaboración Propia

		1- ¿Utilizaría usted un portal web dirigido a la Oficina Provincial de la ONAT de Villa Clara para gestionar y generar información?								
		No			No sé			Sí		
8- ¿Le satisface el portal web desarrollado para contribuir a la gestión de la información en la ONAT?	4- ¿Considera usted que la Oficina Provincial de la ONAT de Villa Clara precise de un portal web que le permita una mejor gestión de la información?									
	Sí	No	No sé	Sí	No	No sé	Sí	No	No sé	
Me satisface mucho	4	3	2	2	4	2	5	4	2	
No me satisface mucho	2	1	3	2	3	3	3	4	3	
Me da lo mismo	2	2	1	3	4	2	3	2	3	
Me insatisface más de lo que me satisface	1	3	4	4	3	4	2	1	1	
No me satisface nada	1	2	1	5	4	2	1	3	2	
No sé decir	2	3	2	3	1	3	3	4	3	

La escala de satisfacción es la siguiente:

1. Clara satisfacción
2. Más satisfecho que insatisfecho
3. No definida
4. Más insatisfecho que satisfecho
5. Clara insatisfacción
6. Contradictoria

Para medir el grado de satisfacción se tomó una muestra de 25 usuarios, que mantuvieron acceso continuo al portal web. El resultado obtenido se muestra a continuación:

Tabla 9. Satisfacción de usuarios potenciales
Fuente: Elaboración Propia

Resultado	Cantidad	%
Máximo de satisfacción	19	76,00%
Más satisfecho que insatisfecho	4	16,00%
No definida	2	8,00%
Más insatisfecho que satisfecho	0	0%
Clara insatisfacción	0	0%
Contradictoria	0	0%

Con el objetivo de obtener el índice de satisfacción grupal (ISG) se trabaja con los diferentes niveles de satisfacción que se expresan en la escala numérica que oscila entre +1 y -1 de la siguiente forma:

Tabla 10. Escala numérica para el ISG
Fuente: Elaboración Propia

Escala	Nivel de satisfacción
1	Máximo de satisfacción
0,5	Más satisfecho que insatisfecho
0	No definido y contradictorio
-0,5	Más insatisfecho que satisfecho
-1	Máxima insatisfacción

Luego es posible calcular el ISG a partir de la siguiente ecuación:

$$ISG = \frac{A(+1) + B(+0,5) + C(0) + D(-0,5) + E(-1)}{N} \quad (1)$$

Donde:

- A, B, C, D, E, representan el número de sujetos con su índice individual.
- N representa el número total de sujetos del grupo.

El ISG obtenido para los usuarios potenciales fue el siguiente:

$$ISG = \frac{19(+1) + 4(+0,5) + 2(0) + 0(-0,5) + 0(-1)}{25} = 0.84 \quad (2)$$

Como se aprecia, el índice de satisfacción grupal es de 0,88 lo que significa una clara satisfacción con la propuesta y reconocimiento de su utilidad en el mejoramiento de lo que plantea el marco teórico de esta investigación.

En la siguiente figura se representan los porcentajes de satisfacción obtenidos luego de calculado el ISG, no encontrándose los tres últimos colores por estar en 0%.

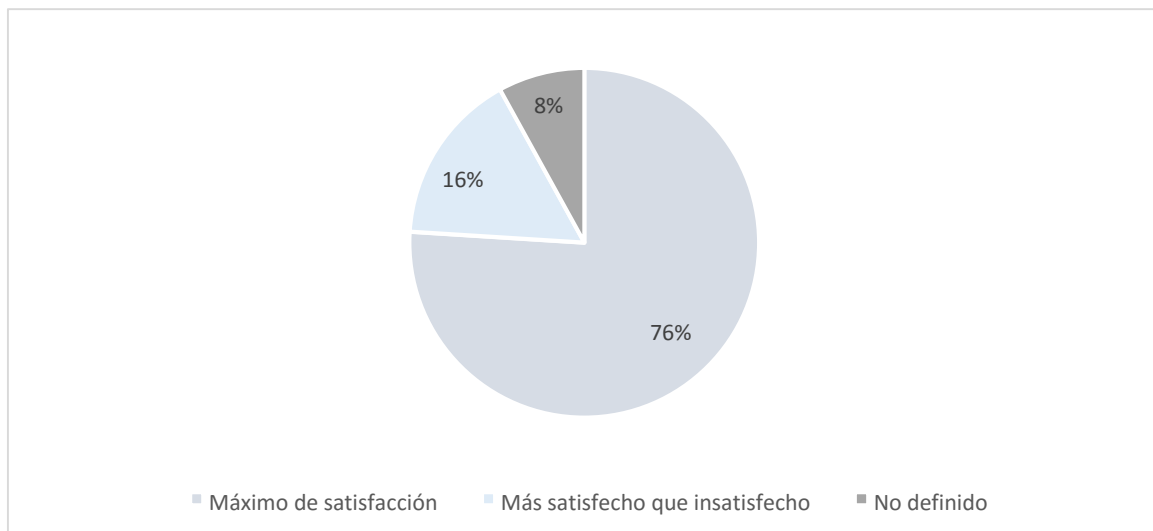


Ilustración 20. Satisfacción de usuarios potenciales con el sitio web
Fuente: Elaboración Propia

3.6 Conclusiones Parciales

- Definir los estándares de codificación utilizados en el desarrollo facilitó la comprensión del código fuente, permitiendo desarrollar el portal en un menor tiempo.
- Las pruebas realizadas al software permitieron determinar y corregir errores y fallos en el funcionamiento de la aplicación, lo que permite asegurar que la propuesta de solución es estable y segura.

CONCLUSIONES GENERALES

Al culminar el desarrollo de este trabajo se obtuvo una propuesta de solución que da cumplimiento al objetivo general planteado:

- El estudio de referentes metodológicos de la investigación a partir de la problemática identificada permitió describir categorías conceptuales de importancia para la investigación.
- La selección de las tecnologías y herramientas necesarias para el desarrollo permitió la implementación de la propuesta de solución.
- El desarrollo del portal web contribuyó a mejorar los procesos de gestión de la información en la oficina provincial de la ONAT en Villa Clara.
- Las pruebas de software realizadas al sistema permitieron identificar y corregir las no conformidades mejorando así la calidad del portal web desarrollado.

RECOMENDACIONES

Al concluir la investigación y el desarrollo de la propuesta de solución se recomienda:

- Desplegar el Portal en la entidad
- Añadir nuevas funcionalidades o características que sean necesarias en el futuro

Referencias Bibliográficas

- Aristega, K. J., & Lárraga, C. F. (2018). Análisis de la importancia del modelado UML en el diseño de software.
- Arizaca Ramírez, E. (2020). Diagrama de Despliegue.
- Cevallos, K. (2015). UML: Diagrama de Secuencia.
- Delgado, L. (2015). Portal Web de la Empresa Exportadora e Importadora de la Construcción—IMECO”.
- Diaz. (2019). Pruebas de seguridad en aplicaciones web como imperativo en la calidad de desarrollo del software.
- Diaz, J., Tzancoff, C., Rodríguez, A., & Soria, V. (2008). Usando JMeter para pruebas de rendimiento.
- Drupal. (2021). <https://www.drupal.org>
- Fabre, F. (2014). Validation by means of users' of the system of indicators approach to foresee, to design and to measure the impact in the projects of investigation of the agricultural sector.
- ForeUI. (2021). <http://www.foreui.com>
- García, Dr. F. J., Moreno, Dra. M. N., García, Dra. A., & Vázquez, A. (2020). UML. Unified Modeling Language.
- García, J. C. (2021). PORTAL: DEFINICIÓN, EVOLUCIÓN Y CLASIFICACIÓN.
- Guerrero, C. A., Suárez, J. M., & Gutiérrez, L. E. (2013). Patrones de Diseño GOF (The Gang of Four) en el contexto de Procesos de Desarrollo de Aplicaciones Orientadas a la Web.
- Gustavo, B. (2020). ¿Qué es MySQL? Explicación detallada para principiantes.
- Invicti. (2021). Acunetix. <https://www.acunetix.com>
- JQuery. (2021). <https://jquery.com>
- Lenguaje CSS. (2021). <https://lenguajecss.com>
- Lenguaje HTML. (2021). <https://lenguajehtml.com>
- Lenguaje JavaScript. (2021). <https://www.javascript.com>
- Lenguaje PHP. (2021). <https://www.php.net>

Lenguajes de Programación. (2021).

Márquez, J., Sampedro, L., & Vargas, F. (2002). Instalación y configuración de Apache, un servidor Web gratis.

Ministerio de Justicia. (2013). Ley No. 113 Del Sistema Tributario. Gaceta Oficial de la República de Cuba.

Ministerio de Justicia. (2018). Decreto-Ley No.370 Sobre la Informatización de la Sociedad en Cuba. Gaceta Oficial de la República de Cuba.

Moreno, E. M. (2018). El portal académico, objetivo y sus funciones.
<https://www.eumed.net/rev/atlante/index.html>

Nikitin, E. (2021). Design patterns in Drupal 9.

Pavón, J. (2013). Aplicaciones Web.

PhpMyAdmin. (2021). <https://www.phpmyadmin.net/>

Pressman, R. (2002). Ingeniería de Software.

Pressman, R. (2010). Ingeniería de Software. Un enfoque Práctico.

Pua, F. (2016). GUÍA DE ESTUDIO MODULAR HERRAMIENTAS CASE.

Quiroga, D. (2020). PATRONES DE DISEÑO DE SOFTWARE.

Riesco, D. (2020). Diagrama de Clases y Diagrama de Objetos.

Rocca, J. I., & Solari, M. (2018). Análisis de Procesos de Software Orientados a la Usabilidad.

Rodríguez, L., González, D. L., & Pérez, Y. (2017). De la arquitectura de información a la experiencia de usuario: Su interrelación en el desarrollo de software de la Universidad de las Ciencias Informáticas. 7. <http://revistaebci.ucr.ac.cr>

Rodríguez, T. (2015). Metodología de desarrollo para la Actividad productiva de la UCI.

Sánchez, J. (2012). Servidores de aplicaciones web.

Sánchez, Jo. M. (2015). Pruebas de Software. Fundamentos y Técnicas.

Sommerville, I. (2011). SOFTWARE ENGINEERING (Ninth Edition).

Sorzano, M. (2019). LA INFLUENCIA DE LOS CMS EN LA CREACIÓN DE LAS PÁGINAS WEB. 3.
<http://www.fitecvirtual.org/ojs-3.0.1/index.php/clic/article/view/323>

- Toro, A., & Gálvez, J. G. (2016). Especificación de requisitos de software: Una mirada desde la revisión teórica de antecedentes.
- Twig. (2021). <https://twig.symfony.com/>
- Universidad de Sevilla. (2012). Introducción a las Aplicaciones Web.
- Valle, Y., Vázquez, A., & Perlacia, V. (2017). Procedimiento para pruebas de usabilidad en las aplicaciones informáticas basado en la ISO/IEC 25000.
- Visual Paradigm. (2021). <https://www.visual-paradigm.com/>
- XAMPP. (2021). <https://www.apachefriends.org>
- Yáñez, R. (2019). Evaluación Heurística de la usabilidad en contextos específicos.

ANEXOS

Anexo 1. Entrevista Técnica para Obtención de Requisitos: Entrevista

Entrevistado: Ing. Neimary Pedraza Martínez

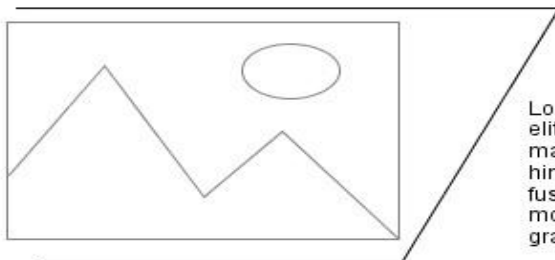
Preguntas a Realizar:

1. ¿Para qué público va dirigido el portal web?
2. ¿Cuál es el propósito general del sitio?
3. ¿Cómo, cuándo y desde dónde usará el usuario el sitio web? (contexto del usuario, características del ordenador, resolución, etc.).
4. ¿Qué colores deben predominar? ¿Existen reglas específicas para la ONAT?
5. ¿Qué va a poder hacer el usuario?
6. ¿Cómo será el acceso al portal?
7. ¿Es necesario autenticarse para navegar por el sitio?
8. ¿Qué tareas son las más utilizadas por los usuarios?
9. ¿Qué trámites se realizan en la entidad?
10. ¿Qué servicios va a brindar el portal?
11. ¿Qué tipo de información se va a mostrar en el sitio?
12. El material que va a ser publicado, ¿está creado o hay que producirlo? (logo, imágenes de las noticias, videos, documentos)
13. ¿Qué tipo de documentos tendrá?
14. ¿Qué herramientas debe tener el sitio web? (blog, RSS, foro...)
15. ¿Qué tecnologías propone para el desarrollo? (lenguajes, cms, servidor web y de base de datos)
16. ¿Qué características presenta la computadora que se utilizará como servidor web?
17. ¿Quién será el contacto del proyecto? (Se solicita nombre, email y teléfono)

Anexo 2 Arquitecturas de Información



Quiénes Somos



Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum eu, maecenas congue facilisi netus justo rutrum himenaeos nibh mi purus praesent. Tempus vitae fusce id erat nulla vel cras malesuada aliquam mollis quisque, sed consequat praesent curabitur gravida fermentum elementum cursus himenaeos

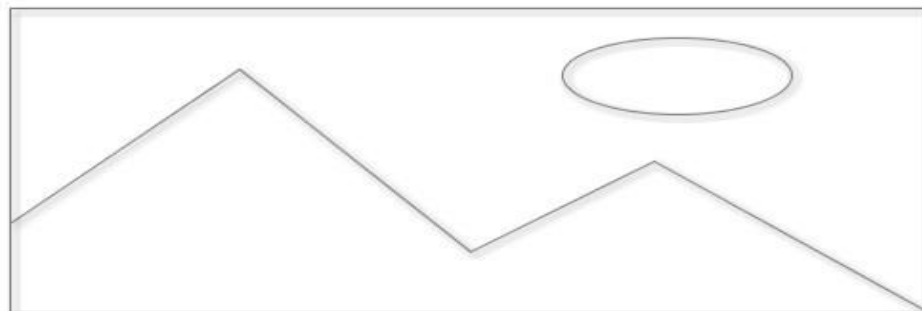
Misión

Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum.

Visión

Lorem ipsum dolor sit amet consectetur adipiscing elit viverra dui scelerisque nascetur dictum.

Organigrama



Dirección

Teléfono

Correo

Sitios de Interés

Portal Tributario

Cubadebate

Gaceta Oficial

UCLV

Medios Informativos

Vanguardia

CMHW

Telecubanacán

Subscríbete

Subscribirse

Ilustración 20. Anexo: Página Quiénes Somos

Fuente: Elaboración Propia



LOREM IPSUM



Lorem ipsum dolor sit amet

17/5/2021

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute inure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. pariatur. Excepteur sint occaecat cupidatat non proident. pariatur.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute inure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. pariatur. Excepteur sint occaecat aliqua ex ea commodo consequat.

[Leer más ..](#)



Lorem ipsum dolor sit

17/5/2021

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

[Leer más ..](#)



Lorem ipsum dolor sit

17/5/2021

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

[Leer más ..](#)



Lorem ipsum dolor sit

17/5/2021

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

[Leer más ..](#)



Lorem ipsum dolor sit

17/5/2021

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

[Leer más ..](#)



Lorem ipsum dolor sit

17/5/2021

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

[Leer más ..](#)



Lorem ipsum dolor sit

17/5/2021

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

[Leer más ..](#)



Lorem ipsum dolor sit

17/5/2021

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

[Leer más ..](#)



Lorem ipsum dolor sit

17/5/2021

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

[Leer más ..](#)



Lorem ipsum dolor sit

17/5/2021

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

[Leer más ..](#)

Dirección

Teléfono

Correo

Sitios de Interés

Portal Tributario

Cubadebate

Gaceta Oficial

UCLV

Medios Informativos

Vanguardia

CMHW

Telecubanacán

Subscríbete

[Subscribirse](#)

Ilustración 21. Anexo: Vista de noticias ampliada

Fuente: Elaboración Propia

Anexo 3. Historias de Usuarios

Tabla 11. Anexo: Historia de Usuario Añadir Pregunta Frecuente
Fuente: Elaboración Propia

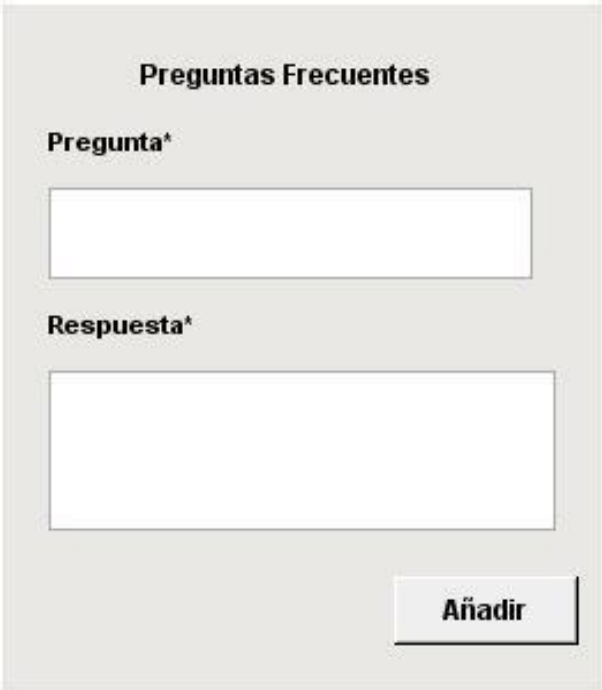

Historia de Usuario		
Nombre de la Historia: Añadir Pregunta Frecuente		
Número: 4	Usuario: Webmaster	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja	
Tiempo Estimado: 24h	Tiempo Real: 12h	Iteración Asignada: 1
Programador Responsable: Ciara Ruiz Gutiérrez		
Descripción: El sistema permite que se añadan preguntas que usualmente realizan los usuarios.		
Observaciones: Para añadir la pregunta deben llenarse los campos: <ul style="list-style-type: none">▪ Pregunta (obligatorio, campo de texto)▪ Respuesta (obligatorio, campo de texto)		
Prototipo de Interfaz gráfica		
 <p>El prototipo de interfaz gráfica muestra un formulario con el título "Preguntas Frecuentes". Incluye dos campos de texto obligatorios: "Pregunta*" y "Respuesta*", y un botón "Añadir" situado en la parte inferior derecha del formulario.</p>		

Tabla 12. Anexo: Historia de Usuario Añadir Trámite
Fuente: Elaboración Propia

Historia de Usuario		
Nombre de la Historia: Añadir Trámite		
Número: 5	Usuario: Webmaster	
Prioridad en negocio: Alta	Riesgo en desarrollo: Media	
Tiempo Estimado: 24h	Tiempo Real: 12h	Iteración Asignada: 1
Programador Responsable: Ciara Ruiz Gutiérrez		
Descripción: El sistema permite que se añadan trámites que se realizan en la entidad.		
Observaciones: Para añadir los trámites deben llenarse los campos: <ul style="list-style-type: none"> ▪ Título (obligatorio, campo de texto) ▪ Contenido (obligatorio, campo de texto) ▪ Imagen (opcional, campo de texto) 		
Prototipo de Interfaz gráfica 		

Anexo 4 Encuesta de Satisfacción

Tabla 13. Encuesta de Satisfacción
Fuente: Elaboración Propia

1- ¿Utilizaría usted un portal web dirigido a la Oficina Provincial de la ONAT de Villa Clara? ___ Sí ___ No ___ No sé
2- ¿Considera que es útil la informatización de los procesos que se realizan en la ONAT? ___ Sí ___ No ___ No sé
3- ¿Qué opina usted sobre la gestión de la información en la ONAT? _____ _____
4- ¿Considera usted que la Oficina Provincial de la ONAT de Villa Clara precise de un portal web que le permita una mejor gestión de la información? ___ Sí ___ No ___ No sé
5- ¿Considera usted que los trabajadores tienen una correcta preparación en el uso de las tecnologías? ___ Sí ___ No ___ No sé
6- ¿Qué aspectos considera que deban añadirse al portal para mejor la gestión de la información? _____ _____
7- ¿Qué importancia le concede a la Gestión de la Información en la Oficina Provincial de la ONAT de Villa Clara? Marque con una X ___ Muy importante ___ Relevante ___ Normal ___ No es importante

___ No sé
8- ¿Le satisface el portal web desarrollado para contribuir a la gestión de la información en la ONAT de Villa Clara?
___ Me satisface mucho
___ No me satisface mucho
___ Me da lo mismo
___ Me insatisface más de lo que me satisface
___ No me satisface nada
___ No sé decir

Anexo 5 Lista de Chequeo de Usabilidad

Tabla 14. Anexo: Lista de Chequeo de Usabilidad.
Fuente: Elaboración Propia

Criterio Usabilidad				
No.	Característica	Pregunta	Aprobado (S/N)	Observación
1	Visibilidad del sistema	¿La página refleja la identidad de la empresa logos, compañía...)?		
		¿Cada pantalla empieza con un título que describe su contenido?		
		¿Cuándo se selecciona un icono se diferencia de los no seleccionados?		
		¿Los iconos que aparecen se identifican claramente con lo que representan?		
		¿Los enlaces del menú se resaltan cuando se seleccionan?		
		¿El sitio le indica al usuario en que parte de la estructura del sitio web se encuentra, es decir si muestra 'migas de pan'?		
		¿Cuándo una tarea involucra documentos fuente, la interfaz es compatible con las características del documento fuente?		
		¿No se muestran errores ortográficos?		

		¿Las imágenes se muestran con buena resolución?		
		¿No hay ninguna imagen con información relevante?		
2	Lenguaje común entre sistema y usuario	¿El lenguaje es simple, con un tono adecuado?		
		¿La información que se presenta en la aplicación es fácil de entender y memorizar?		
		¿La información es de rápida lectura, y con una disposición asequible?		
		¿Evita el lenguaje técnico: términos informáticos o propios de Internet?		
		¿Utiliza un texto específico y descriptivo en los vínculos?		
		¿Los acrónimos y abreviaturas son definidos al ser usados por primera vez?		
3	Libertad y control por parte del usuario	¿Existe una manera lógica de acceder a páginas relacionadas o a otras secciones?		
		¿Tras una acción relevante hay una opción de vuelta atrás?		
		¿Si una acción tiene consecuencias, el sistema proporciona información y pide confirmación antes de continuar?		
		¿En las páginas internas hay un acceso a la página de inicio en una zona visible y reconocible?		
		¿El Sitio cuenta con un mapa o buscador que facilite el acceso directo a los contenidos?		
		¿Los usuarios pueden ordenar y filtrar los resultados?		
		¿Al dar click en el botón "Atrás" siempre lleva al usuario de vuelta a la página de dónde vino?		
		¿El sitio evita que los usuarios se registren de manera innecesaria?		
		¿Es posible aumentar y disminuir el tamaño de letra?		
4	Consistencia y estándares	¿Las imágenes tienen tamaños adecuados que no dificultan el acceso a las páginas?		

		¿Existe una clara distinción entre campos “requeridos” y “opcionales” en los formularios?		
		¿Se mantiene una navegación consistente y coherente en todas las pantallas?		
		¿Se usa la misma fuente para todos los navegadores?		
		¿Puede utilizarse en cualquier navegador?		
		¿Se mantiene una tipografía coherente en todo el sitio web?		
		¿Especifica el tamaño de los archivos PDF?		
		¿Tiene un tiempo de respuesta rápida?		
		¿Se han creado atajos de teclado?		
5	Estética y diseño minimalista	¿Cumple el sitio con el principio de usabilidad de realizar las operaciones con un máximo de tres click?		
		¿Existe suficiente contraste entre el color del fondo y el del texto?		
		¿Los tipos y tamaños de letra son legibles y distinguibles?		
		¿Poseen las páginas animaciones innecesarias?		
		¿El uso de los colores es moderado?		
		¿Utiliza un interlineado adecuado para una buena lectura?		
		¿Los párrafos son cortos?		
		¿Resalta en negrita los conceptos principales de los textos densos?		
		¿El sitio evita el uso excesivo del texto en mayúsculas?		
		¿El texto dentro del sitio está justificado?		
		¿Se utiliza el texto subrayado en el cuerpo del texto para resaltar a menos que sea un hipervínculo?		
6	Prevención de errores	¿Existe suficiente espacio entre los elementos de acción (links, botones, etc.)		

		para prevenir que el usuario haga click en el elemento incorrecto?		
		¿Los botones de acción, (tales como "Enviar") siempre son invocados por el usuario y no automáticamente invocados por el sistema cuando el último campo de un formulario ha sido lleno?		
		¿Se evita el contenido importante del sitio en ventanas emergentes?		
		¿El buscador permite errores tipográficos y ortográficos (tildes)?		
7	Ayuda y documentación	¿Existe un vínculo a los datos de contacto en un lugar bien visible en todas las páginas web del sitio?		
		¿Funcionan correctamente los formularios de contacto?		
		¿El sitio soporta a los usuarios novatos y expertos brindando diferentes niveles de explicación? (ej. en páginas de ayuda y mensajes de error)		
		¿Existe una página con las indicaciones sobre la protección de datos en el sitio web?		
		¿Incluye un vínculo a los datos legales en todas las páginas?		
8	Flexibilidad y eficiencia de uso	¿Las partes o secciones más importantes de los sitios son accesibles desde la página de inicio?		
		¿Se implementen validaciones antes de que el usuario envíe información?		
		¿Las páginas no requieren volver a escribir la información solicitada en páginas anteriores?		