



Facultad 1

Diseño de Sistema para Gestionar la Información de los Expedientes de Cuadros en la Oficina Nacional de Administración Tributaria.

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

Autora:

Angela Maria Bauta Rodríguez

Tutor(es):

MSc. Miguel Jaeger Rodriguez Lazo

Ing. José Manuel Pérez Morales

La Habana, 2021

“Año 63 de la Revolución”

Declaración de autoría:

Angela Maria Bauta Rodríguez la autora del trabajo de diploma con título “**Sistema de Diseño para Gestionar la Información de los Expedientes de Cuadros en la Oficina Nacional de Administración Tributaria**”, 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 firma la presente a los 7 días del mes de diciembre del año 2021.



Angela Maria Bauta Rodríguez

Firma del Autor

Ing. José Manuel Pérez Morales

Firma del Tutor



MSc. Miguel Jaeger Rodriguez Lazo

Firma del Tutor

Datos de Contacto

***Autora:* Angela Maria Bauta Rodríguez**, ambauta@estudiantes.uci.cu.

***Tutor:* MSc. Miguel Jaeger Rodríguez Lazo** (jaeger@uci.cu), graduado de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas (UCI) en junio del 2007. Desde su graduación está trabajando en la UCI, desempeñándose como profesor del Departamento de Programación en las asignaturas de Sistemas Operativos y Seguridad Informática, Programación IV y Programación V. En el 2013 defendió exitosamente su tesis de maestría. Se ha desempeñado como líder de varios proyectos productivos. Ha impartido cursos sobre *Web Services* y el CMS Drupal a entidades nacionales. Ha publicado artículos en revistas científicas como *Acimed* y participado en varios eventos nacionales e internacionales.

***Tutor:* Ing. José Manuel Pérez Morales**, Graduado de Ing. En Ciencias Informáticas en el 2017. Se desempeña como especialista C en ciencias informática en administración de redes en la Oficina Nacional de Gestión Tributaria (ONAT).

Agradecimientos

A mis padres que siempre me han apoyado sin importar las decisiones que he tomado. Gracias por todo lo que han hecho por mí, por estar ahí cuando lo necesitaba, por darme todo lo que tengo y enseñarme a ser una mejor persona.

A mi abuelo que nunca dejó de confiar en mí, siempre me alentó y me apoyó, aun siendo nuestras épocas son tan diferentes. A mis tíos que todos y cada uno de ellos estuvo presente en mi formación como ser humano y como persona.

A mis tutores por toda la paciencia y el apoyo, gracias por los consejos y el apoyo. A mis amigos, gracias por todas las risas y los momentos bellos que he pasado en esta universidad, por tantos recuerdos inolvidables.

Resumen

La gestión de la información les permite a las entidades maximizar el valor y los beneficios derivados de su uso. Con el avance de las nuevas tecnologías, la gestión de la información se ha desplazado al ámbito virtual, esto trae consigo el incremento del uso de sistemas de gestión, los sistemas permiten optimizar los procesos de gestión y mejorar su calidad. La gestión de información referente al expediente de cuadros en la Oficina Nacional de Administración Tributaria, el cual es un documento primario de control de la información individual, no sustituye ni forma parte del expediente laboral, social y personal. Es un proceso complejo llevado a cabo a través de planillas y documentos elaborados manualmente, lo cual potencializa la ocurrencia de errores humanos y la falta de garantía en la integridad, confiabilidad y seguridad de los datos, ralentizando el procesamiento de la información. El trabajo tiene como objetivo desarrollar un sistema para la informatización del proceso de gestión del expediente de cuadro en la ONAT, el cual garantice la confiabilidad de la información y se agilice su procesamiento.

El sistema será desarrollado utilizando como metodología de desarrollo de software AUP-UCI, como lenguaje de modelado se utilizará UML y Visual Paradigm como herramienta CASE, lo que permite el entendimiento entre el cliente y el equipo de desarrollo. Para la construcción de la solución se utilizará el Sistema de Administración e Contenidos WordPress. La solución a desarrollar contribuirá a agilizar el proceso de la información del expediente de cuadro garantizando la confiabilidad de la información.

Palabras clave: expediente, gestión de cuadro, información, tecnología.

Abstract

Information management allows entities to maximize the value and benefits derived from its use. With the advancement of new technologies, information management has moved to the virtual field, this brings with it an increase in the use of management systems, the systems allow to optimize management processes and improve their quality. The management of information regarding the file of cadres in the National Tax Administration Office, which is a primary document for the control of individual information, does not replace or form part of the labor, social and personal file. It is a complex process carried out through forms and documents prepared manually, which potentiates the occurrence of human errors and the lack of guarantee in the integrity, reliability and security of the data, slowing down the processing of the information. The objective of the work is to develop a system for the computerization of the cadre management process in ONAT, which guarantees the reliability of the information and streamlines its processing.

The system was developed using AUP-UCI as a software development methodology, as a UML modeling language and Visual Paradigm as a CASE tool, which allows understanding between the client and the development team. For the construction of the solution, the WordPress Administration and Content system was used. The developed solution contributes to streamline the process of the information of the chart file, guaranteeing the reliability of the information.

Keywords: *file, table management, information, technology.*

Índice

1	Capítulo 1: Fundamentos de la información, metodologías y tecnologías empleadas para darle solución a la problemática.....	18
1.1	Principales conceptos asociados al dominio del problema.....	18
1.1.1	Información:.....	18
1.1.2	Gestión:.....	18
1.1.3	Gestión de la Información:.....	19
1.1.4	Cuadro.....	19
1.1.5	Expediente del cuadro:	19
1.2	Sistemas de gestión de expedientes.....	20
1.2.1	<i>Asistencia Jurídica Gratuita</i>	20
1.2.2	<i>AvilaDocPro</i>	21
1.2.3	<i>Workflow</i>	21
1.3	Metodología de desarrollo del Software	21
1.3.1	AUP:	22
1.4	Herramientas, Tecnologías y Lenguajes.....	23
1.4.1	Lenguaje de modelado.....	23
1.4.2	Herramientas de modelado.....	24
1.4.3	Lenguaje de programación	24
1.5	Marco de trabajo.....	26
1.5.1	WordPress.....	26
1.6	Sistema de gestor de Base de Datos (SGBD).....	28
1.6.1	MySQL:.....	28
1.7	Librería:.....	29
1.7.1	JQuery:.....	29
1.8	Servidor web.....	29

1.8.1	Apache	29
1.9	Conclusiones parciales.....	30
2	Capítulo 2: Descripción de la solución propuesta	31
2.1	Descripción general de la propuesta de solución.....	31
2.2	Roles relacionados con el sistema.....	31
2.3	Modelo Conceptual.....	31
2.4	Requisitos de software.....	32
2.4.1	Requisitos Funcionales.....	32
2.4.2	Requisitos no Funcionales	34
2.4.3	Historias de Usuario.....	35
2.4.4	Plan de iteraciones	37
2.5	Patrones de Diseño	39
2.5.1	Patrones GRA SP.....	39
2.5.2	Patrones GoF	40
2.5.3	Modelo de Datos.....	41
2.6	Fases de implementación	41
2.6.1	Estándares de codificación	42
2.6.2	Modelo de despliegue	42
2.7	Conclusiones Parciales.....	43
3	Capítulo 3: Estrategia de validación para la propuesta de solución.....	44
3.1	Pruebas de software para la evaluación de la propuesta de solución	44
3.2	Estrategia de pruebas	44
3.2.1	Pruebas funcionales.....	44
3.2.2	Pruebas de usabilidad.....	45
3.2.3	Pruebas de Seguridad.....	47
3.2.4	Pruebas de Rendimiento.....	47
3.3	Propuesta de validación para la aplicación a desarrollar	48

3.4 Conclusiones Parciales.....	49
Conclusiones Generales	51
Recomendaciones	52
Referencias Bibliográficas	53

Tabla 1 Roles relacionados con el sistema ¡Error! Marcador no definido.8

Tabla 2 Requisitos funcionales ¡Error! Marcador no definido.30

Tabla 3 Requisitos no funcionales ¡Error! Marcador no definido.1

Tabla 4 Historia de usuario: Adicionar usuario ¡Error! Marcador no definido.3

Tabla 5 Historia de usuario: Modificar cuadro ¡Error! Marcador no definido.3

Tabla 6 Historia de usuario: Modificar reserva ¡Error! Marcador no definido.3

Tabla 7 Plan de iteraciones..... ¡Error! Marcador no definido.4

Tabla 8 Cuadro logico de IADOC para medir la satisfaccion ¡Error! Marcador no definido.

Índice de Figuras

Figura.1 Modelo Conceptual**¡Error! Marcador no definido.**29
Figura.2 Diagrama de Datos**¡Error! Marcador no definido.**
Figura.3 Diagrama de Despliegue..... 39

OPINION DE LOS TUTORES

AVAL DEL CLIENTE

INTRODUCCION

La información en la actualidad se considera un activo invaluable de gran importancia ya que a partir de ella se toman decisiones, se solucionan problemas y se adoptan diversas acciones en correspondencia de su contenido. Es por esto que la mayoría de las entidades a nivel mundial consideran que lograr una buena gestión de la información es fundamental para llevar a cabo todos los procesos o actividades que las entidades requieran.

La gestión de la información no es más que un conjunto de procesos por los cuales se controlan el ciclo de vida de la información, desde su obtención, hasta su disposición final. Sus objetivos principales son maximizar el valor y los beneficios derivados del uso de la información, minimizar el coste de adquisición, procesamiento y uso, además de determinar las responsabilidades para el uso efectivo, eficiente y económico de la misma. Mediante la gestión de la información se puede asegurar el suministro continuo de información fiable. Con el avance en las tecnologías a nivel mundial y las ventajas que brinda los sistemas informáticos, la información ha avanzado hacia la informatización de la gestión de los procesos (Arango, Alvarez et al. 2020).

A menudo las entidades que manejan grandes volúmenes de información utilizan los Sistemas de Gestión de la Información (SGI), con el fin de optimizar los procesos de gestión y mejorar la calidad de la misma.

Los Sistemas de Gestión de Información son un conjunto de sistemas y procedimientos que recopilan información de una variedad de fuentes, la compilan y la presentan en un formato legible de forma tal que permita la toma de decisiones. Mediante los Sistemas de Gestión de la Información se puede registrar, procesar, almacenar, recuperar y visualizar la información de manera fácil y eficiente. Estos sistemas gestionan diversos tipos de información, desde documentación hasta recursos humanos de determinada empresa u organización (Alonso-Arévalo 2007).

En toda empresa u organización se consideran como activos de gran valor a los recursos humanos que las conforman. Las organizaciones financieras son un ejemplo de esto ya que los recursos humanos en este tipo de organizaciones son la base de todas las actividades y procesos que se ejecutan en la misma.

La Organización Nacional de Administración Tributaria (ONAT) desarrolla varias actividades que contribuyen a la preparación de cuadros y reservas para el cumplimiento de sus tareas

en la entidad. La planificación, organización y control de los recursos humanos y materiales son procesos vitales en la garantía del éxito de estos eventos.

La estructura de esta organización parte del principio general de una sola Administración Tributaria en el país organizada en 3 niveles: Primer Nivel (Oficinas centrales), Segundo Nivel (Oficinas Provinciales), Tercer Nivel (Oficinas Municipales), donde cada nivel cumple diferentes funciones (Hernan 1999).

Actualmente la organización posee un órgano de cuadros que pertenece al segundo nivel, donde es responsables de la política de cuadros, evalúa el desempeño de sus funciones, se encarga de su superación profesional y del plan de estimulación de estos. Dentro de estos órganos de cuadro se encuentran los especialistas que se encargan del proceso de gestión de cuadros el que es de vital importancia para la organización, pues es una vía de gestionar los datos de los mismos.

La gestión de información referente al expediente de cuadros en la Oficina Nacional de Administración Tributaria, se lleva a cabo a través de documentos y planillas elaboradas manualmente que se archivan de forma digital en formato Word o Excel, lo cual trae consigo un incremento en la ocurrencia de errores humanos, falta de garantía en la integridad de los datos y no agiliza el procesamiento de los mismos, la información se almacena en una computadora donde cualquiera puede acceder a estos datos, además no se puede asegurar la confidencialidad y seguridad de esta, en la cual no se puede restringir el acceso a dicha información. En dicho expediente se almacenan datos bibliográficos del cuadro, autobiografía, copias certificadas de la resolución de nombramiento de cuadros y otros movimientos, de medidas disciplinarias aplicadas o de su rehabilitación y el certificado de evolución como cuadro.

El proceso de actualización y búsqueda de cualquier información resulta ineficiente e insegura, pues cualquiera la puede modificar. Es por esto que con el fin de agilizar el procesamiento de la información se requiere informatizar la gestión de la información generada por estas actividades.

De la situación antes expresada se plantea el siguiente **Problema a resolver**: ¿Cómo agilizar el proceso de gestión de cuadros de la Oficina Nacional de Administración Tributaria (ONAT)?

Identificándose como el **Objeto de estudio**: Los Sistemas de gestión de información de expedientes y como **Campo de acción**: sistemas para la gestión de información de expedientes de los cuadros.

Se propone como **Objetivo general**: Diseñar un sistema informático para agilizar el proceso de gestión de expedientes de cuadros de la Oficina Nacional de Administración Tributaria (ONAT).

Para darle cumplimiento al objetivo planteado se definieron los siguientes **Objetivos específicos**:

1. Caracterizar el marco teórico conceptual de la investigación.
2. Definir las tecnologías, las herramientas y la metodología para la implementación del sistema de gestión y administración de contenidos de forma centralizada.
3. Diseñar las funcionalidades del sistema de gestión y administración de contenidos de forma centralizada.
4. Diseñar la implementación de las funcionalidades del sistema de gestión y administración de contenido de forma centralizada.
5. Diseñar una estrategia de validación para el sistema a desarrollar mediante las técnicas adecuadas.

Durante el desarrollo de la presente investigación se hizo uso de los siguientes métodos científicos:

Métodos Teóricos:

Analítico-sintético: Permite extraer los elementos más importantes de la documentación estudiada, realiza el análisis de la tecnología, metodologías y herramientas a utilizar en el desarrollo del sistema, de manera que se pueda definir las que mejores se adapten al sistema a desarrollar.

Modelación: Permite modelar los artefactos necesarios en cada fase de la metodología seleccionada de forma tal que permita realizar una representación abstracta de la información.

Inductivo-deductivo: Se utilizo para el razonamiento de la información consultada y llegar así a la obtención de un grupo de conocimientos particulares y generales.

Métodos empíricos:

Entrevista: Se utilizó en el intercambio con el cliente para adquirir información procedente a la gestión de cuadros y para definir los requisitos que deben ser cumplidos referentes a la información.

Análisis del documental: Se realizaron consultas a la información referente a la gestión de información de expedientes de cuadros y a diversas tesis que abordan la gestión de información.

El presente documento se compone de tres capítulos, los cuales se desglosan de la siguiente manera:

Capítulo 1: Fundamentos de la información, metodologías y tecnologías empleadas para darle solución a la problemática. Este capítulo trata sobre los principales conceptos asociados a la investigación. Se realizó un estudio acerca de las principales herramientas o sistemas existentes para la gestión de información de expedientes de cuadros. Se seleccionaron las metodologías empleadas para el desarrollo del sistema, los lenguajes, herramientas de desarrollo y patrones a utilizar. Se describirán las pruebas empleadas para llevar a cabo la validación y verificación del sistema.

Capítulo 2: Descripción de la solución propuesta. Se describe la propuesta de solución a la problemática planteada. Se recolectaron los requisitos funcionales y no funcionales del sistema. Se realizó el modelo de los artefactos correspondientes a las fases de planificación, diseño y desarrollo de la solución. Se describen los patrones de diseño y los patrones arquitectónicos, así como los estándares que se utilizarán.

Capítulo 3: Estrategia de validación de la propuesta de solución. Se plantea la estrategia de pruebas a seguir para evaluar el cumplimiento de los objetivos planteados. Se aplicarán las disciplinas de pruebas internas y de aceptación que define la metodología que guía el desarrollo de la solución propuesta, con el fin de que se pueda verificar y revelar la calidad del producto.

1 Capítulo 1: Fundamentos de la información, metodologías y tecnologías empleadas para darle solución a la problemática.

En el presente capítulo son abordados los principales conceptos que intervienen en el proceso de investigación sobre la gestión de información de expedientes de cuadros. Se hace una revisión bibliográfica sobre los modelos, métodos, metodologías, lenguajes y herramientas de desarrollo de aplicaciones web.

1.1 Principales conceptos asociados al dominio del problema

Con el fin de lograr una mayor comprensión a los términos información, gestión, gestión de la información, cuadro, denominación de expediente de cuadro, herramientas, lenguajes, metodologías utilizadas para la propuesta de solución al problema de la investigación y se realiza un estudio del estado del arte donde se establece una descripción de los sistemas homólogos.

1.1.1 Información:

Se denomina información al grupo de datos ya supervisados y ordenados, que sirven para construir un mensaje basado en un cierto fenómeno o ente. La información permite resolver problemas y tomar decisiones, ya que su aprovechamiento racional es la base del conocimiento. Definen además que estos datos ordenados en un contexto específico sirven para disminuir la incertidumbre y aumentar el conocimiento sobre un tema específico (Arango, Alvarez et al. 2020).

Analizados los conceptos abordados anteriormente, del presente trabajo se considera que la información no es más que el conjunto de datos, de los cuales se puede extraer conocimientos específicos sobre un determinado objeto o asunto. Estos datos sirven de guía para la toma de decisiones, permitiendo al que los estudia obtener nuevos conocimientos sobre determinada materia. Según la autora la información se considera útil cuando esta es precisa, veraz, confiable y vigente, de forma tal que permita obtener conocimientos acertados a los individuos que la requieran (!!! INVALID CITATION !!!).

1.1.2 Gestión:

Es la acción de gestionar y administrar una actividad profesional destinado a establecer los objetivos y medios para su realización, a precisar la organización de sistemas, con el fin de elaborar la estrategia del desarrollo y a ejecutar la gestión del personal. Asimismo, en la

gestión es muy importante la acción, porque es la expresión de interés capaz de influir en una situación dada (Vilcarromero Ruiz 2017).

Por tanto, se puede definir a manera de resumen que la gestión no es más que el conjunto de procesos o actividades desarrolladas con el fin de dar cumplimiento a un objetivo en específico. Comprende los procesos de planificar, organizar, dirigir, controlar y dotar al personal. Por lo tanto, podemos decir que gestionar no es más que la realización de una secuencia organizada de tareas, previamente definidas en un plan o cronograma de trabajo, (Villaquirán and Ospina Nieto 2016).

1.1.3 Gestión de la Información:

La gestión de la información se define como el proceso de organizar, evaluar, presentar, comparar los datos en un determinado contexto, controlando su calidad, de manera que esta sea veraz, oportuna, significativa, exacta y útil y que esta información esté disponible en el momento que se le necesite.

Por lo tanto, se expresa que la gestión de la información se puede definir como un conjunto de procesos destinados a la recopilación o creación de la información, su procesamiento y su distribución. Esta tiene como objetivo garantizar la integridad, disponibilidad y confidencialidad de la información. Una buena gestión de la información debe asegurar el procesamiento y uso de la información de manera correcta y eficiente, asegurando el flujo continuo de la información (Zambrano, Campuzano et al. 2020).

1.1.4 Cuadro

Cuadro profesional, o simplemente cuadro, es un conjunto de mandos de una empresa, administración pública o ejército. Como cuadro intermedio es la categoría profesional intermedia: superior a la del trabajador especializado e inferior al ejecutivo (Enguita, 2007)

1.1.5 Expediente del cuadro:

Se denomina archivo de cuadro a los conjuntos orgánicos de documentos, o la reunión de varios de ellos, al servicio de su utilización para la investigación, la información, la gestión administrativa, reunidos en el ejercicio de sus actividades por el ministerio de finanzas y sus organismos autónomos.

En la Oficina Nacional de Administración Tributaria (ONAT) el conjunto de archivos de los cuadros es generado a partir de documentos de consulta permanente para tomar cualquier

decisión relativa al cuadro y se elabora un único ejemplar llamado expediente de cuadro, el cual está compuesto por:

- ✓ Documentos de control del expediente.
- ✓ Datos bibliográficos del cuadro:
 - Identificación personal, social, familiar.
 - Trayectoria laboral.
 - Preparación cultural y profesional.
 - Formación académica, política, económica y militar.
 - Datos sobre salud, atención y estimulación.
- ✓ Autobiografía.
- ✓ Copias certificadas de la resolución de nombramiento de cuadro y otros movimientos.
- ✓ Medidas disciplinarias aplicadas.
- ✓ Rehabilitación.
- ✓ Certificados de evaluaciones como cuadros.

1.2 Sistemas de gestión de expedientes

La Gestión de expedientes de distinto tipo (de personal, de contratación, expedientes laborales, jurídicos, licencias urbanísticas, de gestión de suelo, expedientes de proyectos de urbanización etc.) no son adecuadamente gestionados hoy por hoy por el archivo físico tradicional (Peig Gómez, 2015).

Se plantea a continuación un estudio de algunos sistemas de gestión de expedientes existentes y de información.

1.2.1 Asistencia Jurídica Gratuita

A partir del 01 de diciembre del 2010 se encuentra instalado y en producción la primera versión del Sistema de Gestión de Expedientes para la Sección de Acuerdos de la Secretaría Administrativa de Superintendencia y Judicial en Informática del Superior Tribunal de Justicia. El objetivo principal del Sistema es permitir la gestión de los expedientes que se tramitan en la Secretaría Administrativa de “Sección Acuerdos”, facilitar el seguimiento y control de los mismos, asistir a los usuarios en la realización de sus funciones en cada etapa del proceso y proporcionar información actualizada y confiable tanto a niveles operativos como directivos de forma oportuna para poder realizar sus labores de manera más eficiente (Misiones, 2021).

1.2.2 AvilaDocPro

Es un sistema de gestión de información destinado a la gestión, tramitación y resguardo de información archivística. Esta aplicación facilita la búsqueda o recuperación de forma rápida y sencilla. Este sistema incorpora el fichero de la documentación en un expediente como punto de partida, simulando dos flujos de procesos: el flujo de la documentación en una entidad y el de incidencias, e integrándolos sobre la base del uso de normas internacionales que rigen el trabajo con los archivos de información (Informáticas, 2020).

1.2.3 Workflow

Se define Workflow como un sistema informático para la automatización de los procesos de negocio, el flujo de trabajo a seguir para la consecución de una tarea o trabajo predeterminado. El sistema Workflow automatiza la secuencia de acciones, actividades y tareas en la ejecución del proceso, aporta herramientas necesarias para su gestión y efectúa un seguimiento de cada una de sus etapas. Es una pieza clave para las organizaciones (Tórrez Campos, 2017).

1.2.4 SARCIO

Es un sistema informático para el control automatizado de los documentos ordinarios, clasificados y limitados en cualquier entidad, teniendo en cuenta lo establecido en el decreto ley No.199, "Sobre la seguridad y protección de la información oficial". Esta aplicación permite el cumplimiento del modelaje establecido para el registro, control y tramitación de los documentos oficiales; el control de los destinatarios, el número de ejemplares, así como, la procedencia de los documentos, la gestión de archivos, conversaciones, desclasificación y destrucción de los documentos oficiales (Informáticas, 2020).

La ingente cantidad de expedientes que actualmente se acumulan, deben gestionarse mediante nuevas herramientas electrónicas capaces de actuar ágilmente para lograr la solución y ejecución de los trámites requeridos por la empresa, teniendo en cuenta las necesidades requeridas en la explotación de expedientes (Mesu Abonia).

1.3 Metodología de desarrollo del Software

Con el fin de obtener un producto de calidad actualmente es necesario llevar a cabo un conjunto de procedimientos, técnicas y herramientas que permiten ajustar el proyecto a las exigencias del usuario. Es por ello que la selección de la metodología a aplicar posee tanta importancia.

1.3.1 AUP:

Proceso Unificado Ágil (AUP, por sus siglas en ingles) de *Scott Amblar* es una versión simplificada del proceso Unificado de desarrollo (RUP, por sus siglas en ingles). Se decidió trabajar con esta metodología debido a que esta describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocios usando técnica ágiles y conceptos que se mantienen validados en RUP. Al no existir una metodología de software universal, ya que toda metodología debe ser adoptada a las características de cada proyecto (equipo de desarrollo, recursos, etc.) exigiéndose así que el proceso sea configurable, se utilizó como metodología de desarrollo la variación del Proceso Unificados Ágil en unión con el Modelo CMMI1-DEV v1.3 definida para la actividad productiva de la UCI.

1.3.1.1 Descripción de las fases de AUP:

De las cuatro fases que propone AUP (Inicio, Elaboración, Construcción, Transición) se decide para el ciclo de vida de los proyectos de la UCI mantener la fase de Inicio, se unifican las restantes tres fases de AUP en una sola, a la que se llamará Ejecución y se agrega la fase de Cierre.

- ✓ **Inicio:** se lleva a cabo las actividades relacionadas con la planeación del proyecto. Se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcalde de proyecto, realizar estimaciones de tiempo, esfuerzo, costo y decidir si se ejecuta o no el proyecto.
- ✓ **Ejecución:** se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, se obtienen los requisitos, se elabora la arquitectura, el diseño, se implementa y se libera el proyecto.
- ✓ **Cierre:** se analizan tanto los resultados del proyecto como su ejecución y se realizaron las actividades formales de cierre del proyecto.

1.3.1.2 Las disciplinas definidas en la variación AUP para la UCI son:

- ✓ **Modelado de negocio:** destinada a comprender los procesos de negocio de una organización. Para modelar el negocio se proponen las siguientes variantes: Casos de Uso del Negocio (CUN), Descripción de Proceso de Negocio (DPN) y Modelo Conceptual (MC).
- ✓ **Requisitos:** comprende la administración y gestión de los requisitos funcionales y no funcionales del producto, agrupados en cuatro escenarios condicionados por el

Modelado de negocio. Existen tres formas de encapsular los requisitos [Casos de Uso del Sistema (CUS), Historias de usuario (HU) y Descripción de requisitos por proceso (DRP)].

- ✓ **Análisis y diseño:** se modela el sistema y su forma (incluida su arquitectura) para que soporte los requisitos, incluyendo los requisitos no funcionales.
- ✓ **Implementación:** a partir de los resultados del Análisis y Diseño se construye el sistema.
- ✓ **Pruebas internas:** se verifica el resultado de la implementación probando cada construcción, incluyendo tanto las construcciones internas como intermedias, así como las versiones finales a ser liberadas.
- ✓ **Pruebas de liberación:** pruebas diseñadas y ejecutadas por una entidad certificadora externa, de la calidad a todos los entregables de los proyectos antes de ser entregados al cliente para su aceptación.
- ✓ **Pruebas de Aceptación:** es la prueba final antes del despliegue del sistema. Su objetivo es verificar que el software está listo y que puede ser usado por usuarios finales para ejecutar aquellas funciones y tareas para las cuales el software fue construido. Todas las disciplinas antes definidas se desarrollan en la Fase de Ejecución, de ahí que en la misma se realicen iteraciones y se obtengan resultados incrementales.

1.4 Herramientas, Tecnologías y Lenguajes

Para el desarrollo de este sistema se seleccionará como marco de trabajo WordPress, con Bootstrap como framework para garantizar un mejor desarrollo web. Se seleccionará como gestor de Base de Datos, MySQL. Como lenguajes se utilizará PHP 7 y HTML 5 que se integran con el marco de trabajo seleccionado, permitiendo la ejecución de comandos y auto completamiento de código.

1.4.1 Lenguaje de modelado

Para realizar el modelo de Datos y el diagrama de componentes se utilizará UML (por sus siglas en inglés, *Unified Modeling Language*) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (*Object Management Group*). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación,

esquemas de bases de datos y compuestos reciclados (Analuisa Enríquez and Sangucho Bunshi 2010).

1.4.2 Herramientas de modelado

Para crear los diagramas se seleccionará el *Visual Paradigm* for UML, herramienta de Ingeniería de Software asistida por computadora que soporta el modelado mediante UML. Proporciona existencia para sus usuarios durante todo el ciclo de vida del desarrollo del software. Permite generar código de forma automática, con el cual minimiza el tiempo de desarrollo y evita errores en la codificación. Genera además múltiples informes como respuesta a la información introducida (Hernández, Peña et al.).

1.4.3 Lenguaje de programación

Para realizar el sistema se utilizará el lenguaje de programación PHP 7 para el desarrollo de los controladores API REST y JavaScript para el desarrollo de las interfaces visuales y la lógica de las mismas a través de la tecnología HTML 5, y para el diseño se utilizarán CSS.

1.4.3.1 PHP 7

El lenguaje PHP o *HyperText Pre-processor*, es un lenguaje de programación de código abierto que puede ser incorporado en HTML 5. PHP 7 fue diseñado para el desarrollo de sitios web dinámicos, los cuales corren en el lado del servidor, permitiéndole una mayor velocidad en tiempo de ejecución. PHP es un lenguaje multiplataforma que separa el diseño de la lógica a la hora de desarrollar. Permite aplicar técnicas de programación orientada a objetos. No requiere definición de tipos de variables, aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución (Heurtel 2016).

Se utilizara PHP en su versión 7.2.9 ya que es un lenguaje simple, a cualquier usuario con conocimientos básicos en programación estructurada le sería fácil desarrollar en él ya que su gran parecido con otros lenguajes facilita al usuario su aprendizaje. PHP 7 brinda la posibilidad de expandir las aplicaciones desarrolladas agregándole nuevos módulos y extensiones. Permite el manejo de excepciones y se integra con la mayoría de los motores de bases de datos, principalmente con MySQL y PostgreSQL (Garzón and Mantilla 2006).

1.4.3.2 HTML 5:

(*HyperText Markup Language*) es la quinta revisión del lenguaje de marcado estándar que se emplea para la web. Es uno de los lenguajes de marcado más usados en todo el mundo y la razón es bastante obvia: gracias a HTML podemos crear la estructura de una página

web. Texto, imágenes y material multimedia pueden mostrarse correctamente gracias a HTML 5. Posee compatibilidad con los navegadores más modernos y populares como Chrome, Firefox, Safari y Opera (PEREZ MAMANI 2020).

Se decidió trabajar con HTML en la propuesta de solución debido a que es gratuito no se necesita ningún tipo de programa especial para trabajarlo, garantiza un código ordenado debido a la adición de las nuevas etiquetas que ayudan a nombrar partes de la estructura básica de toda la página web.

1.4.3.3 JavaScript:

JavaScript es un lenguaje ligero e interpretado que puede ser incrustado en documentos HTML. Este también se utiliza para desarrollar sitios web dinámicos, pero brinda una gran cantidad de funcionalidades adicionales como son el desarrollo de Interfaces de programación de Aplicaciones del Navegador (APIs), los marcos de trabajo y librerías de terceros (PEREZ 2020), permite además incorporar funcionalidades de otros proveedores de contenido, mostrar actualizaciones de contenido al instante, interactuar con mapas y el diseño de animaciones gráficas. Es un lenguaje orientado a objetos multiparadigma, basado en prototipos.

Se utilizará *JavaScript* ya que es un lenguaje dinámico, orientado a objeto y eventos, es un lenguaje multiplataforma, se puede ejecutar en Windows, Linux o Mac, en cualquier navegador sin la necesidad de la utilización de *plugins*. Presenta una curva de aprendizaje fácil. Facilita el trabajo con el texto, arrays, fechas y expresiones regulares.

1.4.3.4 Bootstrap:

Bootstrap es un *framework* CCS de código abierto creado por desarrolladores de Twitter que permite dar forma a sitios web a través de librerías CCS. Incluye tipografías, botones, cuadros, menús y otros elementos que pueden ser utilizados en cualquier sitio web. Permite la creación de sitios web *responsive*, es decir, el sitio web se adapta automáticamente al tamaño del dispositivo en el que se visualiza sin perder su estructura. Brinda agilidad a la hora de cargar el contenido, ya que sus diseños son simples, limpios e intuitivos (Luna, Millahual et al. 2018).

Bootstrap utiliza las librerías de JQuery para la validación de datos y para la visualización de ciertos elementos como son las tablas. Bootstrap puede ser personalizable de manera sencilla, no presenta grandes dificultades a la hora de adaptarlo a las necesidades de

nuestros proyectos. También son personalizables los plugins de JQuery que Bootstrap utiliza.

1.4.3.5 CSS

Es una tecnología que ha tenido una evolución en el tiempo, que actualmente, como su propio nombre indica. Sus siglas corresponden a “*Cascading Style Sheets*”, que tiene el siguiente significado:

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

CSS ha ido evolucionando desde hace 25 años a la versión actual, que es CSS, pero su función es la misma, sirve para dar estilos visuales a las páginas web. Es muy usado para establecer el diseño visual de los documentos web e interfaces de usuario escritas en HTML. En la propia definición de CSS vemos que está muy ligado desde su nacimiento a HTML. Desde que nació, el objetivo de CSS fue poner un poco de orden a la hora de aplicar los estilos a las páginas web (Schulz 2008).

Se trabajará con CSS en la propuesta de solución ya que es un lenguaje de diseño gráfico que permite definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado.

1.5 Marco de trabajo

Un marco de trabajo, es el esquema o estructura que se establece y que se aprovecha para desarrollar y organizar un software determinado. Esta definición, podría resumirse como el entorno pensado para hacer más sencilla la programación de cualquier aplicación o herramienta (Antonio,2018). Para el desarrollo de la solución se propone del marco de trabajo WordPress.

1.5.1 WordPress

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

Capítulo 1: Fundamentos de la Información, metodologías y tecnologías empleadas para darle solución a la propuesta.

clase de web más complejas. WordPress es un sistema ideal para un sitio web que se actualice periódicamente (Sanchez 2012).

WordPress se divide en tres partes:

- ✓ **Core:** WordPress en sí, que es absolutamente gratuito y descargable.
- ✓ **Temas:** que sirven para cambiar la apariencia de la web. Hay un enorme repositorio gratuito, pero también hay recursos de pago fuera del repositorio.
- ✓ **Plugins:** utilidades que pueden convertir tu web en casi cualquier cosa. Igualmente, que los temas, hay un repositorio gratuito y miles de empresas que venden sus funcionalidades.

De todas las plataformas para gestionar contenido que existen, WordPress es una de las más usadas dado a que sus principales ventajas se destacan en:

- ✓ Es un software de código abierto.
- ✓ Es fácil de usar y su instalación es sencilla.
- ✓ Posee una gran variedad de diseños atractivos.
- ✓ Se adapta a la pantalla de dispositivos móviles.
- ✓ Permite obtener estadísticas en tiempo real a través de Plugins.

WordPress tiene una estructura de archivos y carpetas bastante simple. La estructura que presenta actualmente WordPress en realidad no ha variado mucho desde las versiones iniciales, evidentemente, la estructura va cambiando y aumenta la cantidad de archivos y carpeta a medida que vas instalando plugins y temas (Sanchez 2012).

A demás se caracteriza por:

- Aplicación de Libre Distribución (GPL). Se puede utilizar y modificar de forma gratuita.
- Gestión de usuarios. Permite crear usuarios y asignarles diferentes niveles de permisos en relación con la creación y publicación de contenidos en el blog.
- Facilidad de instalación y actualización.
- Suscripción RSS. Soporta el sistema de noticias RSS de tal forma que cualquier lector de noticias RSS puede suscribirse a los artículos, comentarios, etc. del blog.

- Personalización del diseño. Se pueden utilizar plantillas y temas editables que permiten personalizar el aspecto gráfico del blog.

1.6 Sistema de gestor de Base de Datos (SGBD)

Un Sistema Gestor de Bases de Datos no es más que un conjunto de programas que administran y gestionan los datos que están contenidos en una base de datos. ES un sistema de software invisible para el usuario final, compuesto por un lenguaje de definición de datos, un lenguaje de manipulación y de consulta, que puede trabajar a distintos niveles.

Un Sistema Gestor de Bases de Datos posee como objetivos principales:

- ✓ La Independencia de los datos: Los programas de aplicación deben verse afectados lo menos posible por cambios efectuados en datos que no usan.
- ✓ La Integridad de los datos: La información almacenada en las bases de datos debe cumplir ciertos requisitos de calidad, para ello hace falta, en el momento de introducirse los valores de los datos, que éstos se almacenen debidamente, y que posteriormente no se deterioren.
- ✓ La Seguridad de los datos: A la información almacenada en las bases de datos sólo pueden acceder las personas autorizadas y de la forma autorizada (Elmasri, Navathe et al. 2002).

1.6.1 MySQL:

Es un manejador de bases de datos principalmente utilizado en las aplicaciones web en conjunto con PHP 7. En su versión 5.0 anexa características importantes con las que se mejora la programación, debido a que se construyen aplicaciones más seguras, y mejora las reglas del negocio se utilizó en la propuesta de solución (Cuadrado and Reinoso 2016).

En el desarrollo de la solución se seleccionará MySQL por su velocidad a la hora de realizar las operaciones, lo que le hace uno de los gestores que ofrecen mayor rendimiento.

1.6.1.1 PhpMyAdmin:

Es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web. Por las facilidades que brinda esta herramienta para la gestión de Bases de Datos se decide utilizarla. Permite crear y eliminar Base de Datos, crear, eliminar y alterar tablas, borrar editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en el campo, administrar privilegios, y exportar datos en varios formatos (Luján-Mora 2018).

1.7 Librería:

Se emplea el término librería para referirse a una biblioteca, por la similitud con el original inglés *library*. Ambos términos, biblioteca y librería, son correctos según las definiciones de la RAE, aunque en términos formales, la acepción correcta es biblioteca, porque es la traducción correcta de *library*. Una librería o biblioteca es un Kit de herramientas software pequeño y autónomo que ofrece una funcionalidad muy específica al usuario. Normalmente se usa junto con otras librerías y herramientas para hacer una aplicación completa, ya que por lo general las bibliotecas no son ejecutables, pero sí pueden ser usadas por ejecutables que las necesiten para poder funcionar.

1.7.1 JQuery:

JQuery es una librería JavaScript especializada en la realización de cambios dinámicos con rapidez a los documentos de los sitios web (Dziamentsyeu 2016). Facilita el trabajo con el DOM ya que el programador no tiene que saber necesariamente todo sobre el para trabajarlo. JQuery permite agregar interactividad a los sitios web, presenta una gran variedad de *plugins* para dar solución a distintos problemas dentro del maquetado del sitio web.

1.8 Servidor web

Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente y generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente (Zambrano Moreira 2015)).

1.8.1 Apache

Aunque llamamos a Apache es un servidor web, no es un servidor físico, sino un software que se ejecuta en un servidor. Su trabajo es establecer una conexión entre un servidor y los navegadores de los visitantes del sitio web (Firefox, Google Chrome, Safari, etc.) mientras envían archivos entre ellos (estructura cliente-servidor). Apache es un software multiplataforma, por lo cual funciona tanto en servidores Unix como en Windows (VERA CEDEÑO, 2019). Es altamente personalizable, ya que tiene una estructura basada en módulos. Los módulos les permiten a los administradores del servidor activar y desactivar funcionalidades adicionales (Martínez Freire, 2020).

1.9 Conclusiones parciales

Al finalizar el análisis realizado en el presente capítulo se arribó a las siguientes condiciones parciales:

- ✓ Los sistemas de gestión de información existentes en su mayoría, son software a la medida y privativos.
- ✓ La metodología de desarrollo de software más propicia para llevar a cabo el sistema es AUP, dada la forma continua de interacción con el cliente, el poco tiempo de duración del proyecto y la experiencia con que cuenta el desarrollador.
- ✓ La revisión de las principales tecnologías para el desarrollo de sistemas web permitirá seleccionar WordPress, los cuales garantizan un mejor trabajo en el desarrollo de los portales web y del sistema.

2 Capítulo 2: Descripción de la solución propuesta

En este capítulo se hace referencia a las principales características de del sistema a desarrollar, sus requisitos funcionales y no funcionales, así como la realización del modelo de dominio y los actores que intervienen en el mismo. Además, se realizarán los artefactos propuestos en las fases de planificación, diseño y desarrollo de la metodología, lo cual facilita la compensación del sistema a la hora de implementar su solución.

2.1 Descripción general de la propuesta de solución

Con el fin de cumplir con la problemática y el objetivo definido se determinó que el sistema a desarrollar realizara la gestión de información del personal y la documentación asociada al Expediente de cuadros.

2.2 Roles relacionados con el sistema

Se describen los ppprincipales roles que interactúan con el sistema a desarrollar. Los cuales se determinación a partir de las restricciones que es necesario otorgar a cada uno de los usuarios.

Tabla.1 Roles de relación con el sistema.

Actores	Descripción
Administrador	Tiene permiso para administrar todas las funcionalidades del sistema.
Especialista	Puede gestionar todas las funcionalidades relacionadas con los cuadros y las reservas, pero no puede eliminar ningún dato almacenado previamente, ya sea creado por ellos mismos.
Usuario	El usuario común por defecto solo puede visualizar la información que maneja el sistema.

2.3 Modelo Conceptual

Como parte del análisis de los sistemas y para lograr un mayor entendimiento de estos, se realiza un modelo conceptual. El cuál es la composición de conceptos que se utilizan para ayudar a las personas a conocer, comprender o simular un tema que representa el modelo, incluye las entidades importantes y las relaciones entre ellas (Marciszack, Moreno et al. 2019).

A continuación, se describen los principales conceptos tratados en el modelo del dominio.

Personal: Representa a todo el personal que gestiona el sistema.

Trabajador: Representa a los trabajadores registrados en el sistema.

Cargo: Representa el cargo que ocupa el personal.

Cuadro: Representa al personal especialista de un departamento.

Reserva: Representa al personal que sustituye al cuadro en su ausencia.

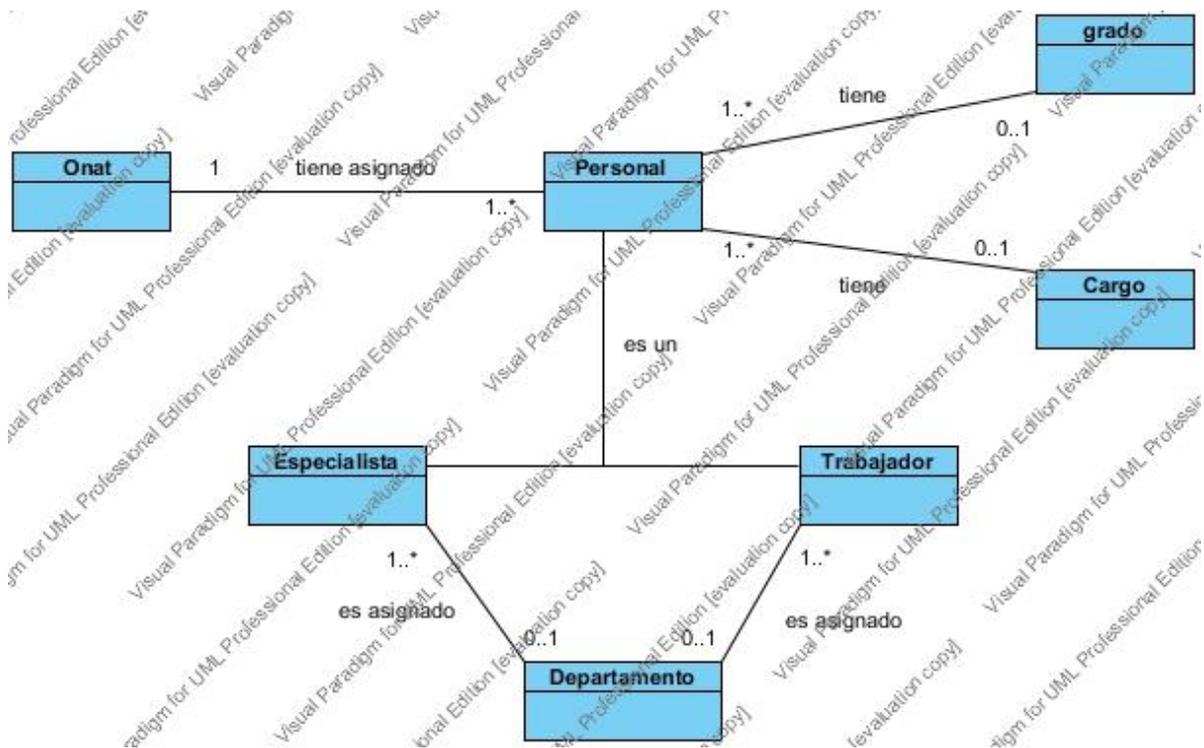


Figura.1 Modelo Conceptual.

2.4 Requisitos de software

Son una descripción completa del comportamiento del sistema que se va desarrollando. Incluye un conjunto de casos de uso que describen todas las interacciones que tendrán los usuarios con el software. Estos reflejan las necesidades del cliente.

2.4.1 Requisitos Funcionales

Los requerimientos funcionales para un sistema definen lo que el sistema debe hacer, una función del sistema de software o sus componentes. Estos describen cualquier actividad

Capítulo 2: Descripción de la solución propuesta.

que el sistema deba realizar, en otras palabras, el comportamiento o función particular de un sistema o software cuando se cumplen ciertas condiciones (Andrade and David 2017).

Tabla.2 Requisitos Funcionales

Código	Requisitos	Prioridad
RF.1	Autenticar usuario	Alta
RF.2	Modificar usuario	Alta
RF.3	Eliminar usuario	Baja
RF.4	Listar usuario	Baja
RF.5	Buscar usuario	Media
RF.6	Cargar archivo con la información de los cuadros	Alta
RF.7	Adicionar cuadro	Alta
RF.8	Modificar cuadro	Alta
RF.9	Eliminar cuadro	Baja
RF.10	Listar cuadro	Baja
RF.11	Buscar cuadro	Media
RF.12	Cargar archivo con la información de las reservas	Alta
RF.13	Adicionar reserva	Alta
RF.14	Modificar reserva	Alta
RF.15	Eliminar reserva	Baja
RF.16	Listar reserva	Baja
RF.17	Buscar reserva	Media
RF.18	Adicionar cargo	Alta

Capítulo 2: Descripción de la solución propuesta.

RF.19	Modificar cargo	Alta
RF.20	Eliminar cargo	Baja
RF.21	Listar cargo	Baja
RF.22	Buscar cargo	Media
RF.23	Cargar archivo con la información de los trabajadores	Alta
RF.24	Adicionar trabajador	Alta
RF.25	Modificar trabajador	Alta
RF.26	Eliminar trabajador	Baja
RF.27	Listar Trabajador	Baja
RF.28	Buscar trabajador	Media
RF.29	Generar reporte por cuadro para Excel	Alta
RF.30	Generar reporte por cuadro para PDF	Alta
RF.31	Generar reporte por reserva para Excel	Alta
RF.32	Generar reporte por reserva para PDF	Alta

2.4.2 Requisitos no Funcionales

Los requisitos no funcionales son los que se refieren directamente a las funciones específicas suministradas por el sistema (características de usuario), sino a las propiedades del sistema: rendimiento, seguridad, disponibilidad. Alternativamente, definen restricciones del sistema tales como la capacidad de los dispositivos de entrada/ salida la representación de los datos utilizados en la interfaz del sistema (Pérez Garay and Macareño Aguilar 2019).

Tabla.3 Requisitos no Funcionales.

Código	Descripción
Usabilidad	
RNF-1	El sistema debe poder ser de fácil uso por cualquier persona con conocimientos básicos en informática.

RNF-2	La interfaz debe brindar mensajes con información de las acciones que se realizan y del éxito o no de estas.
Confiabilidad	
RNF-4	La información en el sistema está protegida de forma que solo los usuarios autorizados a esta pueden visualizarla.
RNF-5	Los usuarios no autenticados no podrán acceder a la información dentro del sistema.
Portabilidad	
RNF-7	La herramienta debe ser un portal web.
Seguridad	
RNF-8	Solo el administrador puede gestionar la información y los permisos de los usuarios.
RNF-9	La contraseña se debe almacenar de manera encriptada en la base de datos.
Implementación	
RNF-10	El sistema debe ser desarrollado utilizando el lenguaje PHP y el marco de trabajo WordPress.
Hardware	
RNF-11	El dispositivo donde se monta el servidor debe tener un mínimo de 2GB RAM, un mínimo de 40GB libre en disco duro y un Procesador de 2.0GHz o superior.
RNF-13	El dispositivo del cliente debe contar con un mínimo recomendado de 512 MB de RAM, además de un procesador Dual Core.
Software	
RNF-14	El ordenador que funcione como servidor debe tener instalado una versión que maneje PHP 7.
RNF-15	El ordenador donde funcione el servidor de base de datos, debe tener instalado el gestor de bases de datos MySQL en su versión 5.0 o superior.
RNF-16	El dispositivo del usuario debe tener instalado un navegador web que soporte todos los estándares que maneja HTML y otras librerías como jQuery y Bootstrap.

2.4.3 Historias de Usuario

Las historias de usuarios son descripciones cortas y simples de una característica contada desde la perspectiva de la persona que desea la nueva capacidad, generalmente el usuario

o cliente del sistema. Es una representación de un requisito escrito en una o dos frases utilizando el lenguaje común del usuario (Darrulat Thoms, Kamil Chalupowicz et al. 2017).

Tabla.4 Historia de Usuario: Adicionar Usuario

Historia de Usuario	
Numero: HU-1	Nombre de historia de usuario: Adicionar usuario
Usuario: Angela Bauta	Iteración asignada: 1
Prioridad en Negocio: Alta	Puntos estimados: 2
	Puntos Reales: 2
Descripción: Permite al administrador agregar un nuevo usuario al sistema.	
Observaciones: Esta funcionalidad solo puede ser realizada por el administrador del sistema.	

Tabla.5 Historia de Usuario: Modificar Cuadro

Historia de Usuario	
Numero: HU-8	Nombre de historia de usuario: Modificar cuadro
Usuario: Angela Bauta	Iteración asignada: 3
Prioridad en Negocio: Alta	Puntos estimados: 3
	Puntos Reales: 2
Descripción: Permite al usuario modificar los datos asociados a un cuadro.	
Observaciones: Esta funcionalidad es realizada solamente por el especialista y el administrador del sistema.	

Tabla.6 Historia de Usuario: Modificar Reserva.

Historia de Usuario	
Numero: HU-14	Nombre de historia de usuario: Modificar reserva
Usuario: Angela Bauta	Iteración asignada: 3
Prioridad en Negocio: Alta	Puntos estimados: 3

Puntos Reales: 2
Descripción: Permite al usuario modificar los datos asociados a una reserva.
Observaciones: Esta funcionalidad es realizada solamente por el especialista y el administrador del sistema.

2.4.4 Plan de iteraciones

Un plan de iteraciones describe la planificación detallada de un periodo de tiempo corto dentro del proyecto, una iteración. En este se incluyen las actividades previamente identificadas, los riesgos que estas involucran, además de los artefactos que se crean. Por otra parte, el plan de iteraciones contiene las herramientas para evaluar sistemáticamente la calidad del producto y del proceso.

Tabla.7 Plan de Iteraciones.

Historia de usuario	Iteraciones	Prioridad	Duración de HU (días)	Duración total (semanas)
Adicionar usuario	1	Alta	2	1
Modificar usuario		Media	1	
Eliminar usuario		Baja	1	
Listar usuario		Baja	1	
Buscar usuario		Baja	1	
Cargar archivo con la información de los cuadros	2	Alta	2	1
Modificar cuadro		Alta	2	
Eliminar cuadro		Baja	1	

Capítulo 2: Descripción de la solución propuesta.

Listar cuadro		Baja	2	
Buscar cuadro		Media	1	
Cargar archivo con la información de las reservas		Alta	2	
Modificar reserva		Alto	2	
Eliminar reserva		Baja	1	
Listar reserva		Baja	2	
Buscar reserva		Media	1	
Adicionar cargo		Alta	1	
Modificar cargo		Media	1	
Eliminar cargo		Baja	1	
Listar cargo		Baja	1	
Buscar cargo		Baja	1	
Cargar archivo con la información del trabajador	3	Alta	3	2
Adicionar trabajador		Alta	3	
Modificar trabajador		Alta	3	
Eliminar trabajador		Baja	1	

Listar trabajador		Baja	1	
Buscar trabajador		Media	2	
Generar reporte por cuadro para Excel	4	Alta	2	
Generar reporte por cuadro para PDF		Alta	1	
Generar reporte por reserva para Excel		Alta	2	
Generar reporte por reserva para PDF		Alta	1	

El plan de iteraciones está conformado por 4 iteraciones, las cuáles se organizan y estructuran el trabajo, de tal forma que el desarrollo del sistema se realice en el menor tiempo posible.

2.5 Patrones de Diseño

Los patrones de diseño son unas técnicas para resolver problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. Para que una solución sea considerada un patrón debe poseer ciertas características. Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reutilizable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias (Jiménez 2009).

2.5.1 Patrones GRA SP

Los patrones GRASP es un acrónimo que significa *General Responsibility Assignment Software Patterns* (patrones generales de software para asignar responsabilidades)

describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones (Turruelles 2016).

- **Experto:** consiste en asignar una responsabilidad al experto en información, es decir a la clase que posee la información necesaria para cumplir con la responsabilidad. En la aplicación este patrón se pone en práctica en la clase *Administrador* que es la encargada de proporcionar información necesaria procedentes de los servicios web.
- **Creador:** consiste en la asignación de responsabilidades relacionadas con la creación de objetos, permite identificar quién debe ser el responsable de la creación de nuevos objetos de una clase. Es donde se asigna la responsabilidad de que una clase B cree un objeto de la clase A.
- **Controlador:** permite que a través de un archivo se procesen todas las peticiones de manipulación por analizar a través de un objeto de controlador único.
- **Alta cohesión:** es un patrón evaluativo que determina cuán relacionadas y adecuadas están las responsabilidades de una clase, de manera que no se realice un trabajo excesivo, y permita un fácil mantenimiento y un aumento de la reutilización.
- **Bajo acoplamiento:** es un patrón evaluativo que permite la creación de clases más independientes que reducen el impacto de los cambios, y también más reutilizables.

2.5.2 Patrones GoF

Los patrones GOF (*Gang of Four*) también conocidos como “Banda de los cuatro”, son patrones que constituyen soluciones técnicas que se basan en la Programación Orientada a Objetos que benefician la reutilización de código (Fuentes Ortega and Guevara Morocho 2011).

- **Instancia única (*Singleton*):** es un patrón diseñado para restringir la creación de objetos pertenecientes a una clase u objeto. Su intención consiste en garantizar que una clase sólo tenga una instancia y proporcionar un punto de acceso global a ella. Este patrón se refleja en las clases controladoras.
- **Mediador (*Mediator*):** define un objeto que coordine la comunicación entre objetos de distintas clases. Se refleja en las librerías que funcionan como mediadoras entre las clases controladoras y los modelos de acceso a datos.

- **Adaptador (Adapter):** convierte la interfaz de una clase en otra distinta que el cliente espera. Un objeto *Adapter* proporciona la funcionalidad prometida por un interfaz sin tener que conocer que clase es utilizada para implementar esa interfaz. Permite trabajo juntas a dos clases con interfaces incompatibles.

2.5.3 Modelo de Datos

El modelo de datos describe los conceptos asociados a la base de datos del sistema. Representa a los objetos o entidades que maneja el sistema, los atributos que describen a cada uno de estos objetos y la relaciones que entre los objetos. A continuación, se muestra el modelo de datos del sistema a desarrollar (Muñoz and Aguilar 2009).

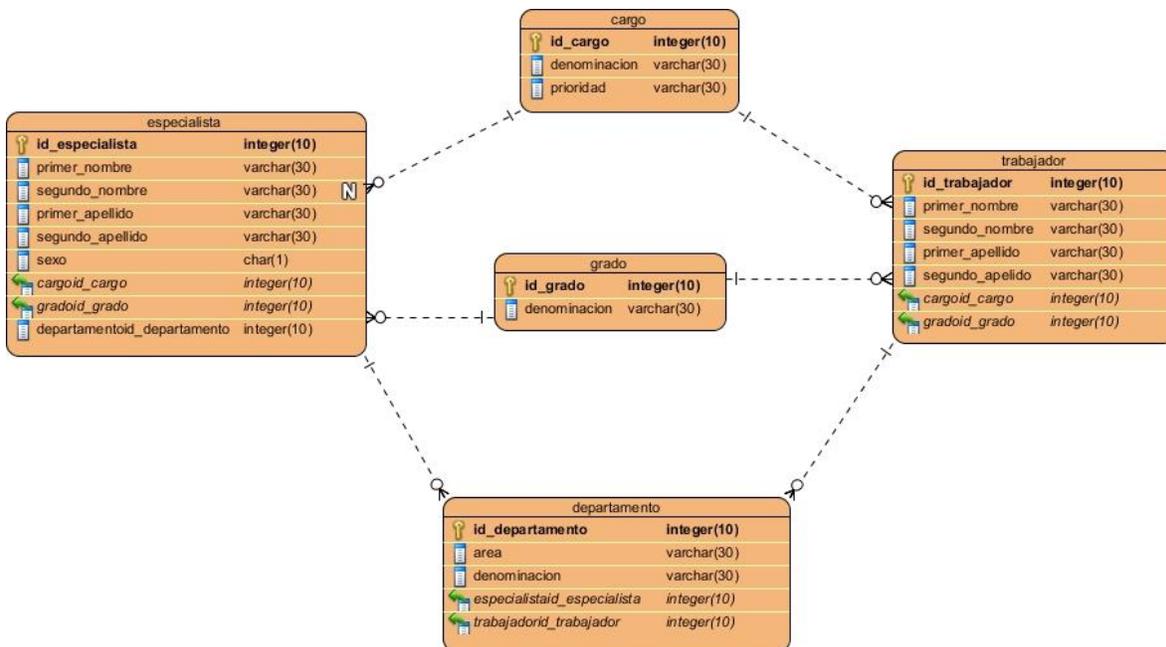


Figura.2 Diagrama de datos.

2.6 Fases de implementación

A partir de los resultados del análisis y el diseño se realiza la implementación del sistema, generándose el modelo de implementación como artefacto de esta disciplina. Este modelo está conformado por el modelo de despliegue.

2.6.1 Estándares de codificación

En la propuesta de solución se utilizan los estándares de codificación los cuales son muy importantes para cualquier proyecto de desarrollo pues establecen la estructura del lenguaje, el orden y normas de estilos. Contribuyen a asegurar que el código contenga menos errores, pueda ser mantenido fácilmente y posea una alta calidad. Todos los programadores tienen su forma personalizada de programar y puede que no se parezca a la de ningún otro, pero de ellos si depende que otros programadores entiendan con facilidad, comprendan el código y lo reutilicen. Es de ahí de donde parte la necesidad de la utilización de los estilos de programación, conocidos también como estándares o convenciones de código que definen un grupo de reglas para escribir código fuente en varios lenguajes de programación (Peis and Ruíz Rodríguez 2004).

Estandartes de codificación a emplear en el desarrollo de la propuesta de solución:

- ¿Cada fichero comenzará con <?php

```
<?php
```

- Los nombres de variables, funciones o parámetros deben seguir la codificación Camel Case.

```
$controller = new ExampleController();  
$controller->exampleFunction(true);
```

- La declaración de las variables siempre será precedida por un \$.

```
$arrayAnimals = ["cat", "dog", "bird", "shark"];  
$userRoot = "Super Admin";  
$ageNow = 24;
```

- La declaración de funciones siempre estará acompañada de una documentación

```
**  
 * @param bool $dump  
 * @return array  
 */  
public function exampleFunction (bool $dump): array  
{  
    (. . .)  
}
```

2.6.2 Modelo de despliegue

El modelo de despliegue se realiza como parte de la implementación para describir la distribución física del sistema. Establece la correspondencia entre la arquitectura lógica, los procesos y nodos. Cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de hardware similar. Los nodos poseen relaciones que representan medios de comunicación entre ellos (Meneses, Fuertes et al. 2011).

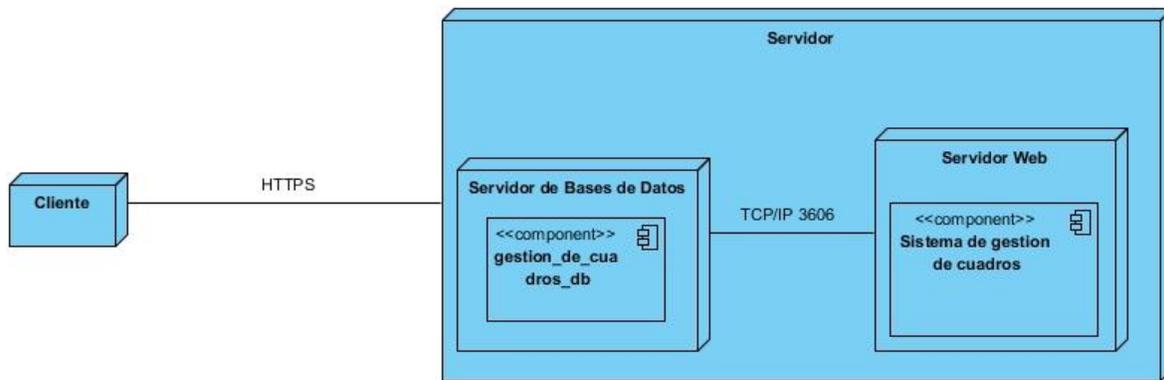


Figura.3 Diagrama de despliegue

2.7 Conclusiones Parciales

Una vez culminado el presente capítulo se arribó a las siguientes conclusiones:

- ✓ Los requerimientos funcionales obtenidos y sus historias de usuario, además de los requisitos no funcionales posibilitaron trazar una propuesta de solución.
- ✓ El modelo conceptual contribuye al entendimiento del sistema de gestión de expedientes de los cuadros al exponer la relación entre cada uno de los conceptos que intervienen en el sistema.
- ✓ La aplicación de los patrones en el desarrollo del sistema y el uso de los estándares de codificación facilitan la reutilización del código y mejora el entendimiento de las clases.

3 Capítulo 3: Estrategia de validación para la propuesta de solución.

En este capítulo se evidencian las estrategias de validación para las pruebas a seguir que se le realizarán al software con el fin de garantizar la calidad del sistema.

3.1 Pruebas de software para la evaluación de la propuesta de solución

Las pruebas funcionales del sistema permitirán verificar que el sistema a desarrollar satisface los requisitos funcionales. Existen una amplia cantidad de trabajos y capítulos de libros que proponen cómo obtener objetivos de prueba a partir de requisitos funcionales expresados como casos de uso (Gutiérrez, 2007).

En los últimos años se ha propuesto una serie de principios que permiten establecer unas pautas comunes para que las empresas de desarrollo de software las conozcan y adapten a sus procesos de pruebas. Las pruebas son unas herramientas que permiten identificar la presencia de defectos; sin embargo, no garantizan que no haya defectos ocultos en el software, y el hecho de que no se identifiquen defectos no es una evidencia de que el software esté totalmente correcto (Mera Paz, 2016).

Las pruebas de software permiten la ejecución de un sistema con el objetivo principal de encontrar errores presentes en el mismo; así se conseguirá reducir y mejorar la calidad del sistema, haciéndolo más robusto frente a fallos.

3.2 Estrategia de pruebas

Una estrategia de pruebas, no es una serie de instrucciones que detallan el paso a paso de cómo se debe hacer las pruebas. En la estrategia se definen los objetivos a alcanzar, a menudo con recursos limitados y en condiciones de incertidumbre, donde se construye un plan de alto nivel para lograr los objetivos específicos de la prueba. La estrategia que se seleccione dependerá principalmente del proceso de desarrollo que se utilice al interior de una organización TI y de las capacidades existentes del equipo de prueba (Blank, 2000).

3.2.1 Pruebas funcionales

Es una prueba de tipo caja negra basada en la ejecución, revisión y retroalimentación de las funcionalidades previamente diseñadas para el software. Es importante mencionar que se debe orientar en el comportamiento externo de un producto o aplicativo de software, en las pruebas de caja negra.

La técnica de **Caja Negra**: Se puede utilizar para lograr objetivos de cobertura de entrada y salida, con entradas humanas, vía interfaces a un sistema, o parámetros de interfaz de las pruebas de integración (Velasco Tanguila, 2019).

Se emplearán las pruebas de caja negra con el objetivo de encontrar errores en las siguientes categorías:

- ✓ Funciones incorrectas o faltantes.
- ✓ Errores de interfaz.
- ✓ Errores en las estructuras de datos o en el acceso a bases de datos externas.
- ✓ Errores de comportamiento o rendimiento.

Para el desarrollo del método de caja negra se utilizan las técnicas (Arocha Caro, 2017):

- **Partición de equivalencia**: divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.
- **Análisis de valor de frontera**: prueba la habilidad del programa para manejar datos que se encuentran en los límites o fronteras aceptables.

Para poner en práctica este método de pruebas, según las técnicas que se describieron anteriormente, se deberán tomara en cuenta los Diseños de Casos de Pruebas (DCP). El diseño de casos de prueba para la **partición de equivalencia** se basa en una evaluación de las clases de equivalencia para una condición de entrada. Una clase de equivalencia representa un conjunto de estados válidos o inválidos para condiciones de entrada. Por su parte, el **análisis de valor de frontera** es una técnica de diseño de casos de prueba que complementan la partición de equivalencia. En lugar de seleccionar algún elemento de una clase de equivalencia, éste conduce a la selección de casos de prueba en los “bordes” de la clase. En lugar de enfocarse exclusivamente en las condiciones de entrada, también deriva casos de prueba a partir del dominio de salida (Suárez Feijóo, 1993).

Para que se lleven a cabo las pruebas de caja negra se confeccionarán CP para los requisitos de mayor impacto en la funcionalidad del sitio web. Esto garantizará un mayor nivel de fiabilidad en el producto final.

3.2.2 Pruebas de usabilidad

Las pruebas de usabilidad son procedimientos con los que se puede probar la usabilidad de una web. Con la utilización de métodos empíricos, una prueba de usabilidad ofrece posibilidades para optimizar la experiencia del usuario. El resultado después de realizar esta prueba puede llevar a cambios en el diseño o en la redacción de la web.

Capítulo 3: Estrategia de validación para la propuesta de solución.

La implementación de las pruebas tiene la ventaja de que pueden implementarse online. Esto beneficia al cliente en términos de coste, especialmente en comparación con los métodos más tradicionales, ya que las pruebas se pueden realizar en gran medida en sus propias webs.

Las pruebas de usabilidad se pueden realizar para muchas áreas y métodos diferentes. Como posibles objetivos de las pruebas se tendrían:

- Comprobación de la usabilidad diaria del software en la fase de desarrollo.
- Análisis de escenarios definidos en el marco de una compra.
- Análisis de una web para explicar fenómenos como un alta tasa de rebote o un bajo tiempo de retención.

Se utilizan los siguientes métodos:

- *Eye Tracking*
- Pruebas *think aloud*
- Prueba de daltonismo
- Evaluación heurística

Para poner en práctica las pruebas de usabilidad en la propuesta de solución se empleará la Evaluación Heurística: Consiste en valorar si los elementos de la interfaz del usuario de un sitio web o aplicación se corresponden con los principios de usabilidad establecidos a partir de una lista de principios o heurísticas. Esta técnica conseguirá detectar un gran número de problemas de usabilidad en sitios web con un menor coste que las técnicas que involucran usuarios. Para su evaluación general en sitios web existen criterios muy establecidos y probados que proporcionan la seguridad de que se está evaluando todo lo que es necesario evaluar (Araujo Falcón, 2014):

- Tipografía
- Navegabilidad
- Diseño, Estructura y Elementos de la Página
- Imágenes
- Contenido escrito para la aplicación

Se utilizará la lista de chequeo establecida por el Departamento de Calidad de la UCI para aplicaciones web para la aplicación de esta técnica, en ella se describen cuáles son los indicadores a evaluarse por cada una de las heurísticas descritas anteriormente.

3.2.3 Pruebas de Seguridad

Son un conjunto de pruebas objetivas con el fin de detectar las vulnerabilidades de un sistema, teniendo muy claro que ningún sistema es 100% seguro o inviolable. Existen diferentes tipos de Pruebas de Seguridad, las más comunes y aceptadas son las Pruebas de Penetración de Caja Negra, las Pruebas de Penetración de Caja Blanca.

- **Prueba de Caja Negra:** En este tipo de auditoría el equipo de consultores no recibe ningún tipo de información sobre los sistemas informáticos y activos pertenecientes a la infraestructura de TI de la organización. En este caso, el equipo de consultores sólo recibe el nombre de la institución, por lo que se trabaja con la información que se puede recolectar a través de medios públicos. Este tipo de pruebas simula el ataque de un cracker, por lo que permite medir el alcance e impacto que tendría un evento real.
- **Prueba de Caja Blanca:** Este enfoque de auditoría se utiliza cuando el cliente necesita realizar un análisis de seguridad a profundidad en los sistemas informáticos. En este tipo de auditoría, el equipo consultor recibe información con mayor detalle sobre los activos y servicios de la infraestructura tecnológica, tal como: versiones de los servicios que se ejecutan, listas de los sistemas operativos instalados en los servidores, código fuente de aplicaciones, entre otros.

Con la implementación de estas pruebas de seguridad se logrará una perspectiva detallada de las vulnerabilidades que se encuentren o se detecten en los sistemas, lo que ayuda al momento de aplicar medidas correctivas para elevar la confiabilidad de la aplicación.

3.2.4 Pruebas de Rendimiento

Esta prueba es llevada a cabo por un grupo de pruebas independientes para asegurar que el rendimiento del sistema está dentro de los parámetros definidos. Se pueden utilizar herramientas para simular clientes y cargas pesadas, y se realizan las mediciones precisas de desempeño. El objetivo de las pruebas de rendimiento es determinar el rendimiento del sistema bajo una carga de trabajo definida utilizando diferentes tipos de pruebas de rendimiento tales como pruebas de carga y de estrés.

- **Pruebas de rendimiento baja intensidad:** Durante la fase de evaluación, cuando ejecuta pruebas funcionales a pequeña escala (pruebas de integración de unidad o

Capítulo 3: Estrategia de validación para la propuesta de solución.

componente), es útil supervisar los tiempos de respuesta de transacciones individuales.

- **Pruebas de rendimiento de alta intensidad:** Las pruebas de rendimiento de alta intensidad se pueden considerar como pruebas de rendimiento instrumentadas. Esta fase de las pruebas se realiza en escenarios completos, tales como pruebas de estrés y pruebas de estabilidad realizadas en procesos de negocio globales.
- **Visualizar restricciones del rendimiento:** Si el entorno de prueba consta de interacciones complejas entre componentes basadas en mensajes, las pruebas de rendimiento comprobarán que estos mensajes fluyen por el sistema a la velocidad necesaria. Para obtener una mejor comprensión de este concepto, es importante visualizar cómo los componentes procesan estos mensajes (Sayago, 2019).

Durante las pruebas de rendimiento se empleará como herramienta auxiliar para el proceso de pruebas de carga y estrés la aplicación Blazemeter para facilitar el procedimiento de validación de la aplicación. El resultado de estas pruebas determinase la capacidad real de la aplicación para enfrentar en entorno de explotación.

3.3 Propuesta de validación para la aplicación a desarrollar

Como complemento del proceso de validación se propone medir la satisfacción de los clientes que se beneficiarán con el producto una vez se despliegue la aplicación. Para ello se propone utilizar la técnica de IADOV. Esta técnica consiste en la realización de una encuesta constituida por 5 preguntas, 3 cerradas y 2 abiertas, las cuales se realizarán a través de un cuadro lógico, el cual permitirá calcular el índice de satisfacción grupal.

A continuación, se muestra el cuadro lógico a obtener a partir de la realización de una encuesta a los trabajadores de la Oficina Nacional de Administración Tributaria (ONAT).

Tabla.7 Cuadro lógico de IADOV para medir la satisfacción

¿Le satisface el sistema desarrollado para la Gestión de la Información de los Expedientes de cuadros en la Oficina Nacional de Administración Tributaria?	¿Considera complejo o difícil el uso del sistema?								
	No			No sé			Sí		
	Si pudiera elegir entre usar o no el software a desarrollar para la gestión de la información de cuadros en la Oficina Nacional de Administración Tributaria ¿lo usaría?								
	Sí	No sé	No	Sí	No sé	No	Sí	No sé	No

Me gusta mucho									
No me gusta tanto									
Me da lo mismo									
Me disgusta más de lo que me gusta									
No me gusta nada									
No sé qué decir									

A partir de las respuestas obtenidas en la encuesta se calculará el índice de satisfacción grupal (ISG), mediante la siguiente formula:

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

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 valor que se obtendrá al realizar el cálculo del índice de satisfacción grupal debe estar regido por las siguientes reglas para poder ver los resultados de la encuesta:

- El índice grupal arroja valores entre +1 y -1.
- Los valores que se encuentran comprendidos entre - 1 y - 0,5 indican insatisfacción.
- Los valores comprendidos entre - 0,49 y + 0,49 evidencian contradicción
- Los valores que caen entre 0,5 y 1 indican que existe satisfacción.

3.4 Conclusiones Parciales

- ✓ La definición de la estrategia de pruebas será fundamental para validar la calidad de la aplicación.
- ✓ La ejecución de varios tipos de pruebas a la propuesta de solución posibilita validar la calidad desde varias artistas.

Capítulo 3: Estrategia de validación para la propuesta de solución.

- ✓ La técnica de IADOV permite corroborar la correcta funcionalidad de la herramienta a través de encuestas realizadas a personas que interactúan con el sitio web, garantizando así que la misma cuente con todos los requisitos contemplados y que el cliente quede satisfecho con el producto elaborado.

Conclusiones Generales

Finalizada la investigación se puede concluir que se cumplieron todos los objetivos propuestos, resaltando que:

- ✓ Las fases de la metodología de software AUP-UCI permitieron la obtención de los artefactos que guiaron el proceso de desarrollo de la herramienta.
- ✓ Con la metodología seleccionada se describe de forma sencilla y simple de entender el modo de desarrollar aplicaciones web.
- ✓ La definición de la estrategia de pruebas es de vital importancia en la validación de la calidad de la aplicación.
- ✓ El sistema informático a obtener contribuirá a la agilización del procesamiento de la información del expediente de cuadro de la ONAT.

Recomendaciones

Como resultado del proceso de investigación se recomienda:

- Realizar la implementación de la aplicación para la gestión de la información de los expedientes de cuadros en la Oficina Nacional de Administración Tributaria.
- Ejecutar las pruebas de software para detectar y corregir deficiencias presentes en las funcionalidades implementadas, validando así la propuesta de solución por parte del cliente a partir de las pruebas de aceptación y su nivel de satisfacción.

Referencias Bibliográficas

Desoft Aplicaciones Informáticas, "AvilaDocPro." from <https://www.desoft.cu/es/node/173>, 2020.

Desoft Aplicaciones Informáticas "SARCIO." from <https://www.desoft.cu/es/productos/156>, 2020.

Alonso-Arévalo, J. (2007). "Gestión de la Información, gestión de contenidos y conocimiento."

Analuisa Enríquez, M. d. R. and L. P. Sangucho Bunshi (2010). "Diseño e implementación de un sitio web para la promoción y administración del hotel San Luis de la ciudad de Latacunga aplicando la metodología UML (Unified Modeling Language), en el periodo 2010."

Andrade, M. and W. David (2017). Sistema informático para la gestión de la evaluación de desempeño de los docentes en la Universidad Técnica de Cotopaxi, Ecuador: La Mana: Universidad Técnica de Cotopaxi; Facultad de Ciencias de

Arango, V., et al. (2020). "Diseño del sistema de información conablele para la empresa programación industria y control SAS en Santiago de Cali."

Cuadrado, Ó. N. and A. J. J. T. y. d. Reinoso (2016). "Sistema de gestión de contenidos y tienda online sincronizada con ERP." **14**.

Darriulat Thoms, N. M., et al. (2017). "NextOne."

Dziamentsyeu, D. (2016). Desarrollo de una aplicación móvil, y rediseño del entorno web, de una empresa del sector de la nutrición vegetal, Universitat Politècnica de València.

Elmasri, R., et al. (2002). Fundamentos de sistemas de bases de datos, Addison-Wesley.

Fuertes Ortega, V. M. and J. C. Guevara Morocho (2011). Análisis del Patrón Modelo Vista Controlador Implementado en Lenguajes de Software Libre para el Desarrollo de Aplicaciones Web. Caso Práctico: Liceo de Talentos Stephen Hawking.

Garzón, G. and J. C. Mantilla (2006). "Diseño e implementación de una interfaz abierta sobre el web para la ejecución de sentencias SQL sobre motores de bases de datos relacionales."

Hernan, I. C. (1999). Manual de Bienvenida.

Hernández, L. R. B., et al. "Extensión de la herramienta Visual Paradigm for UML para la evaluación y corrección de Diagramas de Casos de Uso Plugin of Visual Paradigm for UML tool for evaluation and correction of Use Case Diagram."

Heurtel, O. (2016). PHP 7: Desarrollar un sitio web dinámico e interactivo, Ediciones ENI.

Jiménez, J. M. R. J. R. d. E. a. D. (2009). "Patrones pedagógicos en educación virtual."

Luján-Mora, S. J. P. H. I. (2018). "Programación Hipermedia I. Práctica 8: PHP 3 (MySQL y acceso a una base de datos)(curso 2016-2017)."

Luna, F., et al. (2018). PROGRAMACION WEB Full Stack 12-Sitios multiplataforma con Bootstrap: Desarrollo frontend y backend-Curso visual y práctico, RedUsers.

Marciszack, M. M., et al. (2019). "Patrones en la construcción del Modelo Conceptual para sistemas de información."

Meneses, F., et al. (2011). Modelo Distribuido para la Gestión de Entornos Virtuales de Red. Publicado en las memorias del VI Congreso de Ciencia y Tecnología ESPE.

Muñoz, A. and J. J. A. e. S. e. I. Aguilar (2009). "Ontología para bases de datos orientadas a objetos y multimedia." **6**(2): 167-184.

Peis, E. and A. A. J. H. n. Ruíz Rodríguez (2004). "EAD (Encoded Archival Description): Desarrollo, estructura, uso y aplicaciones." **2**.

Pérez Garay, C. A. and J. N. Macareño Aguilar (2019). Aplicación web para el control y gestión de la Jornada Universitaria de Desarrollo Científico (JUDC) de la Universidad Nacional Autónoma de Nicaragua, Managua (UNAN-Managua), en el año 2019, Universidad Nacional Autónoma de Nicaragua, Managua.

PEREZ, I. T. (2020). MEMORIA DE ESTADÍAS SISTEMA GESTOR DE RECURSOS MATERIALES, UNIVERSIDAD POLITÉCNICA DE TLAXCALA.

PEREZ MAMANI, G. (2020). "SISTEMA DE INFORMACIÓN DE EVALUACIÓN DOCENTE PARA LA CARRERA COMERCIO INTERNACIONAL."

Sanchez, L. J. O. d. h. s. w. c. a.-b.-e.-d.-l.-i.-a. (2012). "WordPress."

Schulz, R. G. (2008). Diseño web con CSS, Marcombo.

Turrueles, Y. G. (2016). Departamento de Informática, Universidad de Matanzas.

- Vilcarrromero Ruiz, R. (2017). "Gestión de la Producción."
- Villaquirán, V. A. and Y. Ospina Nieto (2016). "Sistemas de gestión de calidad, factor importante para el desarrollo laboral del colaborador de Unisalud Palmira."
- Zambrano, C. Y., et al. (2020). "La gestión de la información y el conocimiento a partir de estrategias formativas innovadoras." **4**(1): 109-118.
- Zambrano Moreira, J. R. (2015). "Configuración de servicios de red con el desarrollo de un sistema web para la administración de estudiantes y cursos."
- TÓRREZ CAMPOS, Ezequiel Josué; CHAVARRÍA VÍLCHEZ, Gustavo Rafael; DURÁN SANABRIA, Yamil José. Implementación de un WorkFlow para automatizar el proceso de la Orden de Compras Menores en el área de Adquisiciones del Consejo Nacional de Universidades en el primer semestre del 2017. 2017. Tesis Doctoral. Universidad Nacional Autónoma de Nicaragua, Managua
- MESU ABONIA, Francly Lorena, et al. Propuesta de un modelo para la Gestión Documental basado en la Norma ISO30300.
- ANTONIO, Rendón Rendón Pedro. Optimización en el desarrollo de aplicaciones web o de escritorio mediante el uso de Framework.
- GUTIÉRREZ, J. J.; ESCALONA, M. J., et al. Implementación de pruebas del sistema. Un caso práctico. 2007, vol. 1, nº 4, p. 1.
- MERA PAZ, J. A. J. I. S. Análisis del proceso de pruebas de calidad de software. 2016, vol. 12, nº 20,
- BLANK, S. y DORF, B. J. B. G. El manual del emprendedor. 2000, nº
- VELASCO TANGUILA, G. E. *Análisis comparativo de herramientas para la automatización de pruebas de Software*. Quito: UCE, 2019.
- AROCHA CARO, A. Aplicación android para el Complejo Escultórico Comandante "Ernesto Guevara". Universidad Central "Marta Abreu" de Las Villas, 2017.
- SUÁREZ FEIJÓO, R. Planificación automática y supervisión de operaciones de montaje mediante robots. Universitat Politècnica de Catalunya, 1993. ISBN 846926334X.
- ARAUJO FALCÓN, M. N. Análisis de usabilidad a la interfaz de carga de archivos de la plataforma Paideia PUCP. 2014, nº

SAYAGO, J. P.; FLORES, E. L., et al. Análisis Comparativo entre los Estándares Orientados a Servicios Web SOAP, REST y GRAPHQL. 2019, vol. 9, nº 2, p. 10-22.