

## DISEÑO PRELIMINAR DE UN SOFTWARE DE ADMINISTRACIÓN Y GESTIÓN DE ACTIVOS PARA EL DEPÓSITO DE CARGO DE OPERACIONES DE PAZ

**Fecha de recepción:** 28 de diciembre de 2023

**Fecha de aceptación:** 28 de febrero de 2024

CAP. Fernando Otárola González<sup>1</sup>

**Resumen:** *el propósito del presente artículo es abordar el desarrollo de un demostrador tecnológico destinado a mejorar la gestión de activos de la Fuerza de Paz Conjunta "Cruz del Sur", los cuales se encuentran bajo la custodia del Depósito de Cargo de Operaciones de Paz (DCO) en dependencias del Regimiento Logístico del Ejército N° 2 "Arsenales de Guerra".*

*La investigación empleó diversas herramientas metodológicas que permitieron diseñar un software y desarrollar un demostrador tecnológico. Este último tiene la capacidad de automatizar las labores de administración y gestión de los activos de la Fuerza de Paz Conjunta "Cruz del Sur". Su propósito radica en optimizar el control y desarrollo de las tareas vinculadas a la gestión de dichos activos, lo que se traduce en una mejora significativa de la eficiencia en la ejecución de las tareas y procesos administrativos que desarrolla el DCO.*

**Palabras clave:** *software, gestión de activos, fuerza de paz, demostrador tecnológico, diseño preliminar.*

## PRELIMINARY DESIGN OF AN ASSET MANAGEMENT AND ADMINISTRATION SOFTWARE FOR THE CARGO DEPOT OF PEACE OPERATIONS

**Abstract:** *the purpose of this article is to address the development of a technological demonstrator aimed at improving the management of assets within the Combined Joint Peace Force "Cruz del Sur", which are under the custody of the Depot of Peace Operations (DCO) at the Army Logistic Regiment N° 2 "Arsenales de Guerra".*

*The research employed various methodological tools that allowed for the design of software and the development of a technological demonstrator. The latter has the ability to automate the administrative and management tasks of the assets of the "Cruz del Sur" Joint Peace Force. Its purpose lies in optimizing the control and development of tasks associated with the management of these assets, resulting in a significant improvement in the efficiency of executing administrative tasks and processes carried out by the DCO.*

**Key Words:** *software, asset management, peace force, technological demonstrator, preliminary design.*

---

<sup>1</sup> Oficial del Ejército de Chile, Ingeniero Politécnico Militar mención Comunicaciones e Informática de la Academia Politécnica Militar del Ejército de Chile. Actualmente desempeña labores como Oficial de Mando y Control de la Brigada de Operaciones Especiales del Ejército de Chile. Email: fernando.otarola@ejercito.cl

## **1. INTRODUCCIÓN**

La razón de ser del Ejército de Chile es contribuir de manera fundamental a preservar la paz y su misión principal es garantizar la soberanía nacional, mantener la integridad territorial, junto con proteger a la población, instituciones y recursos vitales del país, frente a cualquier amenaza o agresión externa, y también constituir una importante herramienta para la política exterior de Chile (Ministerio de Defensa, 2017, p. 264). Es en este contexto donde podemos notar que uno de los principales medios con los que cuenta nuestro Ejército es la Fuerza de Paz Conjunta Combinada “Cruz del Sur”.

Para cumplir esta misión, las tareas de administración y gestión son fundamentales, considerando la cantidad y diversidad de material que es custodiado. Este material está compuesto por distintos rubros, que son: armamento, vehículos, munición, vestuario y equipo, contenedores, carpas de habitabilidad, mobiliario, plantas de purificación de agua, óptica y oprónica, telecomunicaciones, equipo técnico eléctrico, equipo técnico de fontanería, entre otros.

Administrar, gestionar y mantener el ciclo de vida de los activos mencionados, requiere contar con información actualizada, confiable, verídica y oportuna, además de un sistema informático que la gestione, sistema con el cual esta unidad no cuenta a la fecha.

El propósito del presente artículo es dar a conocer el diseño preliminar de un software que facilita la administración y gestión de los activos de la Fuerza de Paz Conjunta Combinada “Cruz del Sur”, solución desarrollada a través de la utilización de ingeniería de software y metodología de la investigación, lo que finalizó con el desarrollo de una herramienta informática que permite la digitalización y optimización de las tareas administrativas llevadas a cabo por el Depósito de Cargo de Operaciones de Paz, de manera trazable y eficiente.<sup>1</sup>

## **2. DESARROLLO**

### **2.1 Problema de Investigación**

Para visualizar el problema que afecta a esta organización, se utilizó la metodología general de identificación, preparación y evaluación de proyectos de inversión pública (Ortegón, *et al.* 2005), iniciando con el análisis de los involucrados, organizando luego reuniones para determinar los efectos y las causas de la problemática mediante la herramienta de lluvia de ideas y de esta manera definir el problema de investigación.

---

<sup>1</sup> Se considera el cargo disponible al año 2023.

Además, la aplicación de la metodología propuesta por CEPAL permitió identificar alternativas de solución al problema detectado. Estas alternativas fueron evaluadas por un grupo de expertos con el fin de mantener la objetividad en el proceso de investigación. Grupo que, utilizando los criterios establecidos por el investigador para este propósito y aplicando el proceso jerárquico analítico (AHP), logró definir la solución óptima para resolver el problema investigado en dos iteraciones. Solución que quedó establecida como el diseño de un software para la administración y gestión de los activos de la FPCC “Cruz del Sur”.

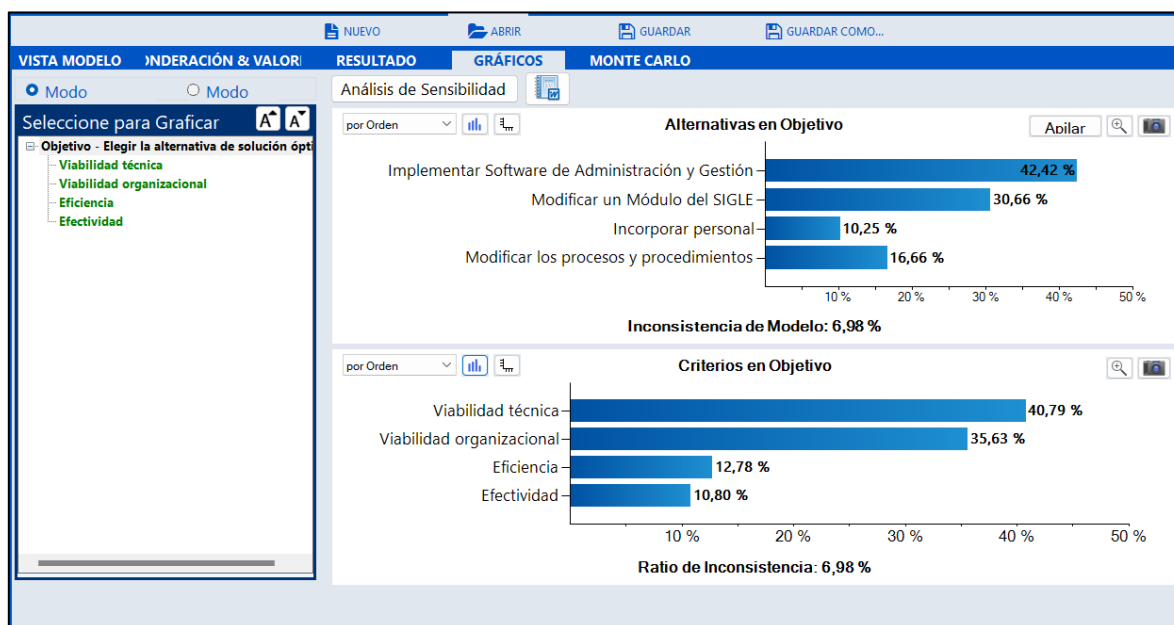


Figura N° 1: Selección de la alternativa de solución óptima, primera iteración. Gráficos finales de la selección de la alternativa óptima.

Fuente: Elaboración propia, en el software Total Decision Edicion (2023).

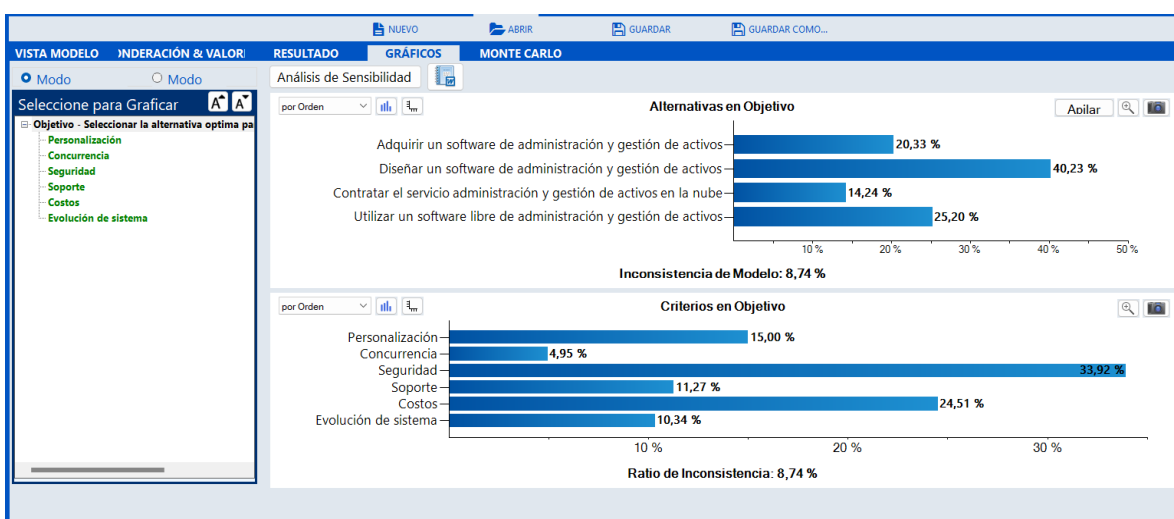


Figura N° 2: Selección de la alternativa de solución óptima, segunda iteración. Gráficos finales de la selección de la alternativa óptima.

Fuente: Elaboración propia, en el software Total Decision Edicion (2023).

Bajo el mismo contexto, a continuación, se definió como objetivo general diseñar preliminarmente un software de administración y gestión de activos para el Depósito de Cargo de Operaciones de Paz, además se definieron las preguntas que ayudarán a delimitar el alcance de la investigación y a enfocar los esfuerzos en la búsqueda de respuestas.

Por último, la delimitación de los alcances permitió establecer los márgenes del trabajo, todo ello con el objetivo de enmarcar y ordenar la investigación y el desarrollo de la solución.

## 2.2 Marco teórico y metodológico de la investigación

### 2.2.1. Marco teórico

Una vez determinado el problema y seleccionada la alternativa de solución, es necesario adquirir el conjunto de conocimientos teóricos acerca de los activos de la FPCC "Cruz del Sur" y sobre la ingeniería de software, incluyendo sus actividades estructurales y los términos específicos relacionados con las tecnologías de la información, de los cuales algunos de los más importantes son:

- **Software:** *“Instrucciones (programas de cómputo) que cuando se ejecutan proporcionan las características, función y desempeño buscados”* (Pressman, 2010, p. 3).
- **Programas de cómputo:** es un conjunto de instrucciones escritas en algún lenguaje de programación. Para el autor Ángel Gutiérrez es importante que *“muchos tipos de programas o subprogramas (subrutinas) pueden integrar lo que se conoce como librerías o bibliotecas de programas, que son de gran ayuda al momento de la programación”* (Gutiérrez, 2018, p. 153).
- **Lenguajes de bajo nivel:** *“Son lenguajes totalmente dependientes de la máquina. Esto significa que el programa que se realiza con este tipo de lenguajes no se puede migrar o utilizar en otras máquinas”* (Gutiérrez, 2018, p. 241).
- **Lenguaje máquina:** *“La información que hace que el hardware de la computadora realice una determinada actividad se llama instrucción. En términos simples, una instrucción es un conjunto de unos y ceros. Las instrucciones así formadas equivalen a acciones elementales de la máquina”* (Gutiérrez, 2018, p. 241).
- **Ingeniería de software:** se define como *“una disciplina de la ingeniería que se interesa por todos los aspectos de la producción de software”* (Sommerville, 2011, p. 6).

- **Base de datos:** es un conjunto organizado de datos relacionados entre sí, estructurados y almacenados de manera sistemática en un sistema informático. Estos datos representan distinto tipo de información, como nombres, números, textos, imágenes, etc. La base de datos permite almacenar, recuperar, modificar y eliminar los datos de manera eficiente y segura, permitiendo el acceso a la información de forma estructurada y coherente. Para el autor A. Gutiérrez una base de datos es *“un conjunto de datos que están organizados para un uso en específico. Los programas que permiten gestionar estos datos son conocidos SGBD”* (Gutiérrez, 2018, p. 274).
  
- **Sistema de Información y Gestión Logística del Ejército (SIGLE):** SIGLE es una herramienta informática implementada por el Ejército de Chile que tiene como objetivo optimizar-administrar los recursos logísticos de la institución, su desarrollo es con base en el lenguaje JAVA.<sup>2</sup> El objetivo de esta herramienta informática, según el MAL-01007,<sup>3</sup> es *“automatizar los procesos logísticos institucionales con el fin de facilitar el control adecuado del flujo de bienes y servicios adquiridos por la Institución”* (CAF, 2019). Esta automatización abarca todo el ciclo de vida logístico y se basa en la catalogación de los elementos que se ingresan al sistema.
  
- **Technology Readiness Levels (TLR):** *“Corresponden a una descripción de la historia de un sistema, subsistema o componente en relación con la tecnología. Permite describir el estado de desarrollo de alguna tecnología y entregan una base sobre la madurez de ellas y de avance posible”* (Aldecoa, 2020, p. 3).
  
- **Demostrador tecnológico:** el demostrador tecnológico se sitúa en el nivel 7 de los Technology Readiness Levels (TRL), que corresponde a la demostración del sistema o prototipo en un entorno real. En este nivel, se lleva a cabo la validación y verificación del sistema o prototipo en condiciones operativas reales.
  
- **Fuerza de Paz Conjunta Combinada “Cruz del Sur”:** es una fuerza de paz conjunta, conformada por fuerzas de Argentina y Chile, y combinada, por estar integrada por las tres ramas de la Fuerzas Armadas de cada país. Su propósito es estar a disposición de la Organización de Naciones Unidas para ser empleada en operaciones de mantenimiento de la paz, bajo el sistema UNSAS (United Nations Standby Arrangement System), utilizado por la ONU para tener disponibles unidades militares y policiales.

---

<sup>2</sup> Java es un lenguaje de programación de alto nivel, orientado a objetos y generalmente considerado como *“write once, run anywhere”* (Oracle, 2023).

<sup>3</sup> MAL-01007 Manual de codificación de material por rubros en el Sistema de información y Gestión Logística del Ejército 2019 (CAF, 2019).

### 2.2.2. Estado del arte

Por otra parte, en el estudio del estado de arte pudimos observar que la implementación de herramientas informáticas en las organizaciones ha demostrado ser de gran utilidad, ya que facilita considerablemente el trabajo y permite una explotación más eficiente y efectiva de los recursos. Esta opinión es compartida por muchas organizaciones, lo que ha llevado a que en la actualidad haya una amplia variedad de proveedores que ofrecen servicios de gestión de activos computacionales, tanto de manera comercial como también de forma gratuita a través del desarrollo de software de código abierto.

### 2.2.3. Marco metodológico

Una vez establecido el marco teórico de la investigación, se definió el marco metodológico, el que se ha estructurado en base al objetivo general, que es diseñar preliminarmente un software de administración y gestión de activos para el Depósito de Operaciones de Paz (DCO), en base a cinco objetivos específicos que serán abordados. Cada uno de estos objetivos se enfocará en aspectos claves relacionados con el software de administración y gestión de activos para el Depósito de Cargo de Operaciones de Paz. Finalmente, como resultado de este proceso, se presentará un demostrador tecnológico validado, que será el producto final de la investigación y servirá como una representación tangible y funcional del software desarrollado para el Depósito de Cargo de Operaciones de Paz.

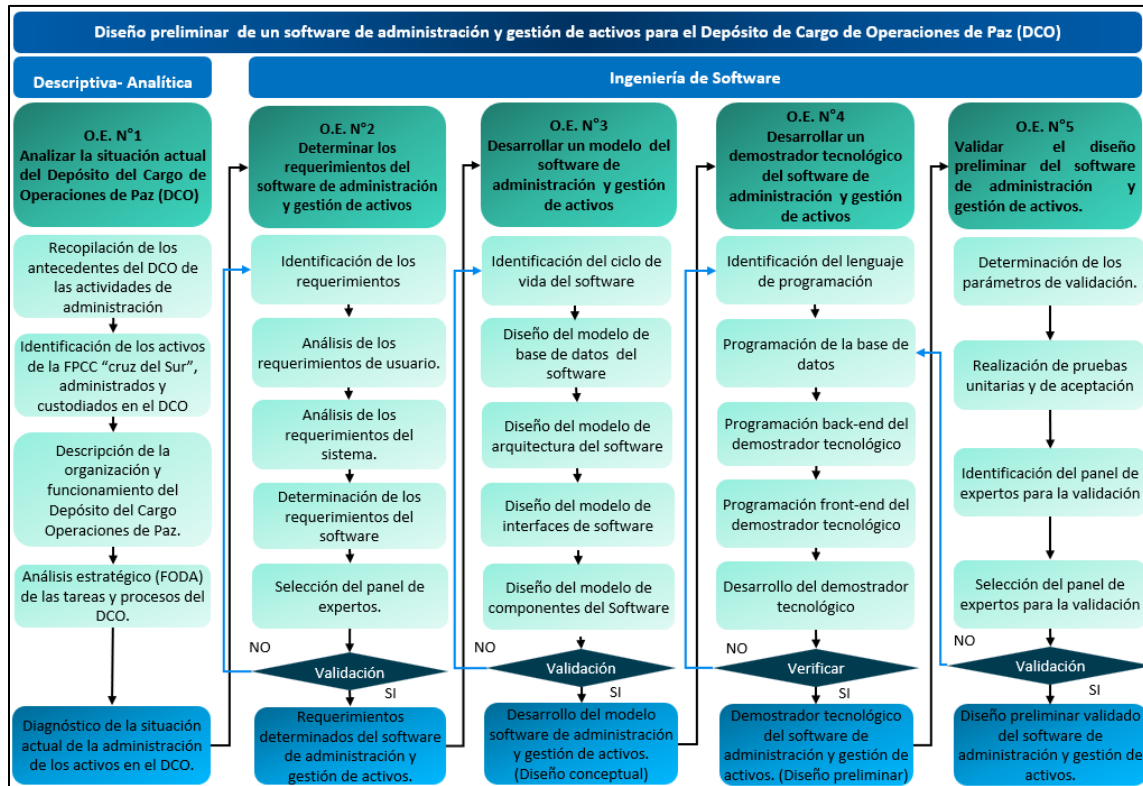


Figura N° 3: Secuencia metodológica.

Fuente: Elaboración propia (2023).

### 2.3 Análisis de la situación actual en el DCO y determinación de los requerimientos

Luego de establecer los marcos que guiarán la investigación, procede el análisis de la situación actual en el Depósito de Cargo de Operaciones de Paz. Esto fue posible gracias a la recopilación de antecedentes sobre las actividades realizadas por el DCO, así como la identificación de los activos y sus características diversas. Además, se examinó el funcionamiento y organización del Depósito de Cargo de Operaciones de Paz, incluyendo su estructura organizacional y la infraestructura física relacionada con los almacenes.

Mediante el uso de la herramienta FODA, se logró establecer un diagnóstico integral de la situación actual del DCO. Entre los aspectos relevantes a considerar se encuentran la complejidad en la administración de activos, la prioridad en el mantenimiento y conservación, y las limitaciones de personal. Por tanto, una de las estrategias detectadas es el desarrollo de un sistema informático propio que respalde las labores de administración y gestión en la unidad, lo que permitiría mayor control, trazabilidad y eficiencia en la gestión de los activos, facilitando la toma de decisiones y mejorando la eficacia del DCO.

Luego de lo anterior, se abordó el proceso de levantamiento de los requisitos fundamentales para el desarrollo de una solución. Para lograr este objetivo, en primer lugar, se identificaron los interesados (*stakeholders*). Luego, se seleccionó el "*proceso de adquisición y análisis de requisitos*" (Pressman, 2010) como herramienta, el cual implica una serie de actividades secuenciales e iterativas que permiten obtener una comprensión clara de las necesidades, expectativas, restricciones y limitaciones del sistema esperado. De esta manera, se logró finalmente establecer los requerimientos de usuario (de alto nivel) junto con los requerimientos de sistema (requisitos funcionales y no funcionales), lo que permite al desarrollador conocer qué debe hacer el sistema y cómo lo debe hacer en dependencias del Depósito de Cargo de Operaciones de Paz.



Figura N° 4: Bodega del Depósito de Cargo de Operaciones de Paz.  
Fuente: Depósito de Operaciones de Paz RLE N° 2 (2023).

## 2.4 Diseño del software de administración y gestión de activos

El siguiente paso consistió en establecer el diseño del software destinado a la administración y gestión de activos asociados a la FPCC "Cruz del Sur", a través de un enfoque metodológico y secuencial de ingeniería de software que cumpla con los requisitos definidos en el punto anterior.

En primer lugar, se identificó el ciclo de vida del software, seleccionándose el modelo en cascada como el ciclo de vida óptimo para el desarrollo del software, ya que garantiza un enfoque sistemático en cada fase. En segundo lugar, se procedió a identificar los diagramas de procesos del software, que representan las funciones generales que el software debe llevar a cabo para cumplir con los requerimientos del usuario. Posterior a ello y utilizando la metodología de ingeniería de software de Roger Pressman, se diseñó en base al modelo en capas que se presenta en la figura N°5.

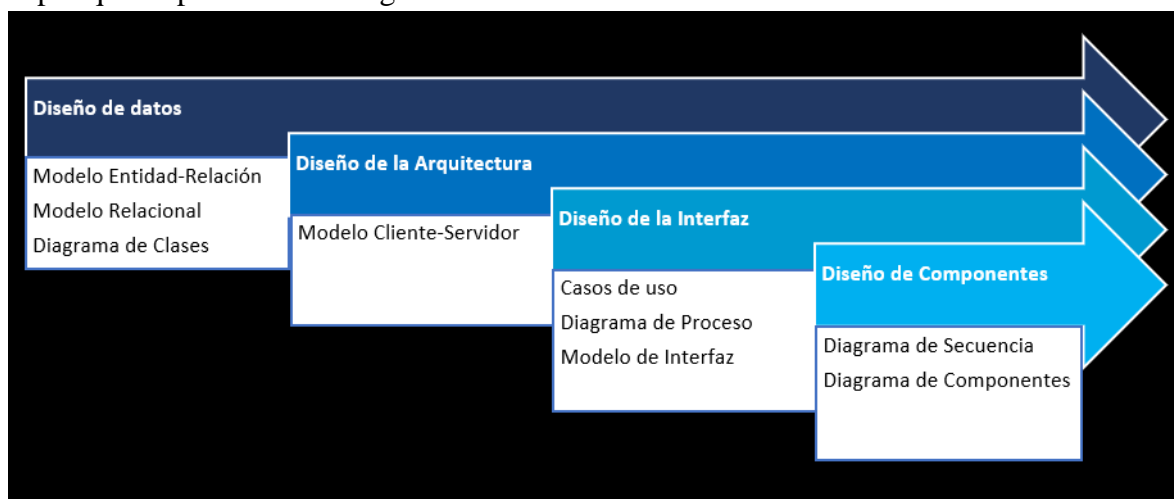


Figura N° 5: Modelo de diseño de 4 capas.

Fuente: Elaboración propia adaptada del libro "Ingeniería de software: un enfoque práctico".

### 2.4.1 Diseño del modelo de base de datos

En el proceso de desarrollo de sistemas de información, uno de los pasos cruciales es desarrollar la primera capa que consiste en el diseño del modelo de base de datos. Este componente es fundamental, ya que actúa como el almacén de datos que respalda toda la funcionalidad del sistema. Consecuente con lo anterior y teniendo en consideración los principios ACID, extraídos del texto "Fundamentos de Bases de Datos" del autor Abraham Silberschatz (2002), se diseñó el modelo de la base de datos en 2 etapas. En la primera se diseñó el modelo entidad-relación considerando los siguientes componentes:

- **Entidad:** representa un objeto o concepto del mundo real que se puede identificar y almacenar en una base de datos.
- **Atributo:** son las características o propiedades de una entidad.



- **Relación:** las relaciones representan cómo las entidades se relacionan entre sí en la base de datos.
- **Cardinalidad:** la cardinalidad en una relación especifica cuántas entidades se pueden relacionar entre sí. Estas pueden ser: 1:1 (uno a uno), 1:N (uno a muchos) y N:N (muchos a muchos).
- **Clave primaria:** cada entidad tiene una clave primaria que es un atributo único que identifica de manera particular a cada instancia de esa entidad.

El modelo obtenido se presenta en la figura N° 6:

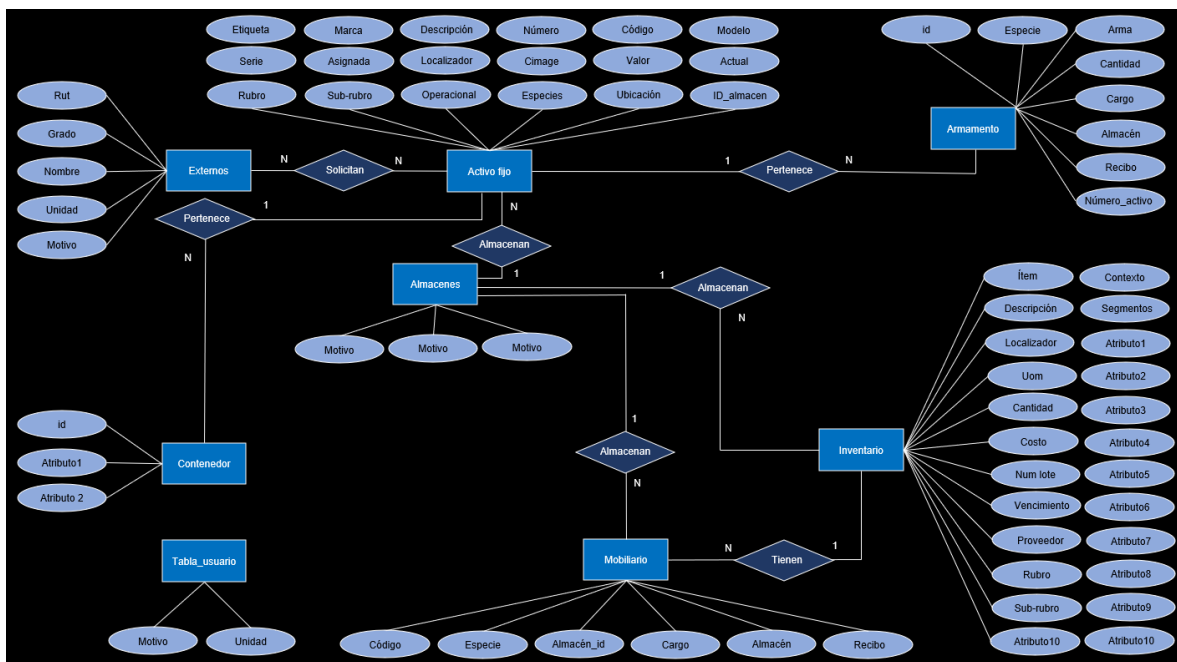


Figura N° 6: Modelo entidad-relación de la base de datos.

Fuente: Elaboración propia en base al libro “Ingeniería de software: un enfoque práctico”.

Para llevar a cabo la implementación del Modelo Entidad-Relación (MER) de manera efectiva, es esencial transformarlo en un modelo lógico que pueda ser comprendido y ejecutado por el ordenador. Esta transición se logra a través de la creación de un Modelo Relacional (MR). En el MR, las entidades, relaciones y atributos del MER se traducen en tablas, claves primarias y foráneas.

La creación de claves foráneas es un paso fundamental en la gestión de bases de datos, ya que estas claves permiten establecer conexiones lógicas entre conjuntos de datos, lo que facilita la recuperación de información coherente y relacionada. Estas relaciones son

esenciales para entender la estructura y el funcionamiento de la base de datos en su totalidad. Además, las claves foráneas desempeñan un papel crucial en la preservación de la integridad de los datos al imponer restricciones que aseguran que las relaciones entre las tablas sean coherentes y que no se produzcan inconsistencias. En última instancia, este proceso contribuye a mantener la consistencia de los datos en toda la base de datos, garantizando que la información sea precisa y utilizable de manera eficiente para las operaciones y análisis posteriores.

Finalmente, utilizando el enfoque orientado a objetos se diseñó el diagrama de clases, que permite visualizar las clases, sus atributos, relaciones y métodos que permitirán la construcción y codificación de la base de datos (ver figura N° 7).

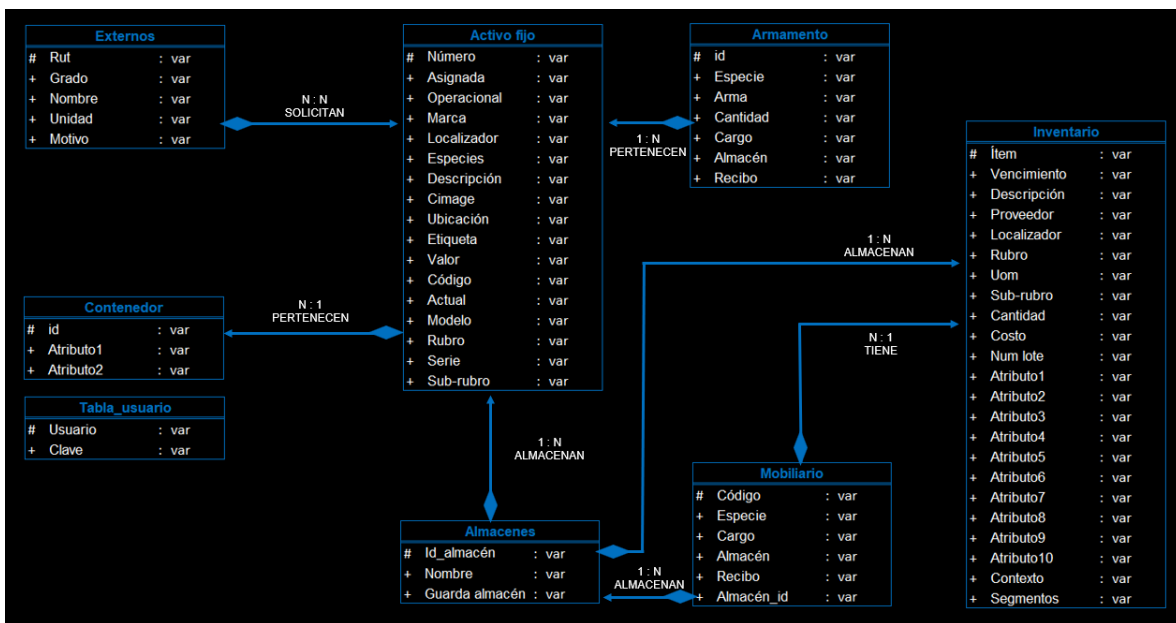


Figura N° 7: Diagrama de clases.

Fuente: Elaboración propia en base al libro “Ingeniería de software: un enfoque práctico”.

### 2.4.2 Diseño del modelo de arquitectura del software

La segunda capa del proceso es el diseño arquitectónico, que desempeña un papel fundamental al definir la estructura y las interacciones de funcionamiento entre el software, los equipos y las redes. Este diseño es esencial para establecer las bases del sistema y garantizar su eficacia.

El diseño de la arquitectura se elaboró teniendo en cuenta las políticas institucionales, que enfatizan la necesidad de utilizar lenguajes que permitan la escalabilidad y evitar la creación de herramientas de escritorio que no puedan controlarse en cuanto a sus actualizaciones, lo que podría generar riesgos informáticos.

En este contexto, la arquitectura desarrollada se alinea con el modelo cliente-servidor, reconocido por su diseño en capas claramente definidas, lo que facilita su integración con otros avances tecnológicos si fuera necesario. Además, esta arquitectura brinda un control efectivo sobre las actualizaciones de seguridad, asegurando un alto nivel de protección. En este modelo, cada capa desempeña una función específica que contribuye de manera incremental a la funcionalidad general del sistema, tal como se detalla a continuación en la figura N° 8.

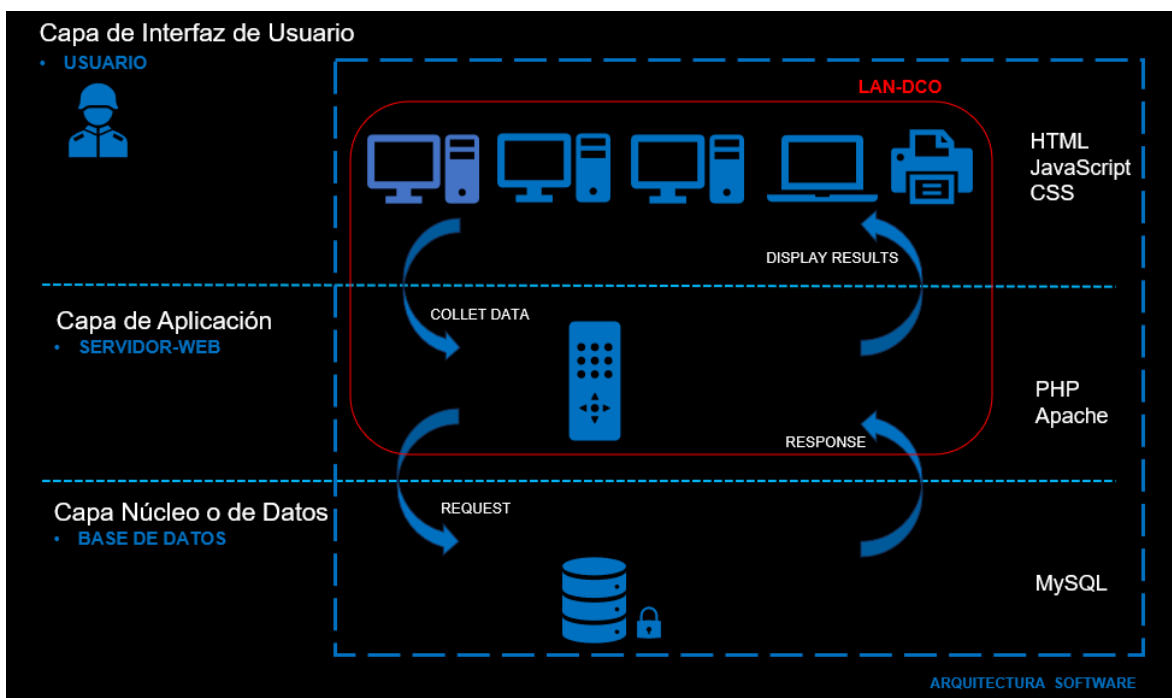


Figura N° 8: Modelo de arquitectura cliente-servidor.<sup>4</sup>

Fuente: Elaboración propia en base al libro “Ingeniería de software un enfoque práctico”.

### 2.4.3 Diseño del modelo de interfaces de software

Después de completar el diseño de las dos primeras capas, el siguiente paso es abordar la tercera capa, que se centra en el desarrollo del diseño de las interfaces del software. Esta etapa es de vital importancia, ya que estas interfaces son la cara visible que permite a los usuarios interactuar y percibir cómo se cumplen las funciones definidas en el levantamiento de los requerimientos. Utilizando la metodología de capas, es esencial transformar los diagramas de proceso previamente establecidos a una interfaz gráfica que se ajuste perfectamente a los requerimientos y especificaciones delineados en dicho paso.

### 2.4.4 Diseño del modelo de componentes del software

<sup>4</sup> HTML es sigla de lenguaje de marcado de hipertexto o HyperText Markup Language, CSS son las siglas de Cascading Style Sheets o hojas de estilo en cascada, PHP es de Hypertext Preprocessor, MySQL es abreviatura de Structured Query Language.

Según el autor R. Pressman “*un componente es un elemento funcional de un programa que incorpora la lógica del procesamiento, las estructuras de datos internas que se requieren para implementar la lógica del procesamiento y la interfaz que permite la invocación del componente*”. Por lo tanto, teniendo en cuenta los modelos de las capas anteriores, se procedió al diseño de la cuarta y última capa de componentes utilizando el paradigma orientado a objetos, que es el enfoque adecuado para el desarrollo de un software encargado de administrar y gestionar activos de diversos tipos.

Finalmente, con el diseño de estos 4 niveles se establecen las bases sólidas para el desarrollo y codificación del software de administración y gestión de activos, asegurando que el diseño cumple con los requerimientos y está respaldado por una evaluación imparcial de expertos y usuarios.

## **2.5 Desarrollo y programación del demostrador tecnológico del software**

Luego de contar con el diseño conceptual del software, el siguiente paso consistió en el desarrollo del demostrador tecnológico, que es esencial para validar la solución propuesta ante la problemática que afecta al DCO. La selección de los lenguajes de programación se llevó a cabo en colaboración con expertos, utilizando el Proceso Jerárquico Analítico (AHP) para identificar la combinación óptima, resultando esta ser PHP, HTML, CSS y MySQL.

La programación se dividió en tres partes: primero, la creación de la base de datos, luego el desarrollo del *back-end* y finalmente el *front-end*.

### **2.5.1 Programación de la base de datos**

La programación de la base de datos se efectuó utilizando el lenguaje MySQL, que permite trabajar con un conjunto organizado de datos relacionados que se almacenan y gestionan de manera estructurada. El propósito principal de la base de datos es permitir el almacenamiento eficiente, la recuperación y la gestión de información de manera sistemática.

### **2.5.2 Programación back-end**

El siguiente paso consistió en la programación del *back-end*, que implicó la creación de algoritmos encargados de ejecutar las funciones lógicas necesarias para facilitar la conexión entre las interfaces visibles para el usuario y la gestión de la información almacenada en la base de datos. Las principales funcionalidades programadas incluyeron las operaciones CRUD (*create, read, update y delete*). Para llevar a cabo esta codificación, el programador se apoyó en el software Visual Studio Code, un editor de código, y utilizó el lenguaje PHP que es un lenguaje de *scripting* del lado del servidor, lo que significa que el código PHP se ejecuta en el servidor web antes de que se envíe la página web al navegador del usuario, permitiendo la optimización en la ejecución de los procesos.

### 2.5.3 Programación front-end

A continuación, se realizó la programación del *front-end*, que abarcó la creación de las interfaces utilizadas por los usuarios para interactuar con el sistema. En esta etapa, se empleó el lenguaje HTML junto con CSS, lo que permitió el desarrollo de interfaces intuitivas y estéticamente atractivas, en consonancia con los requisitos establecidos por los usuarios. Es relevante destacar que, durante el desarrollo del *front-end*, se consideró la compatibilidad con los navegadores comunes, como Chrome, Edge y Firefox, que se utilizan en el entorno de trabajo (DCO). Esto se realizó teniendo en cuenta la ventaja principal de la arquitectura cliente-servidor, que permite aprovechar al máximo la capacidad de cada dispositivo, reduciendo así la carga de recursos en el servidor.

```

1
2 <!DOCTYPE html>
3 <html lang="en">
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Encuadramiento</title>
8   <link rel="stylesheet" type="text/css" href="CSS/styles_inicio3.css" />
9 </head>
10 <header>
11   <input type="checkbox" id="btn-menu">
12   <label for="btn-menu"></label>
13
14   <nav class="menu">
15     <ul>
16       <li><a href="inicio.html">INICIO</a></li>
17       <li><a href="administracion.html">ADMINISTRACIÓN DEL CARGO</a></li>
18       <li><a href="phpOPERACIONAL/0_operacionalidad.php">ESTADO OPERACIONAL</a></li>
19       <li><a href="phpHOJASVIDA/">OPERACIÓN Y MANTENIMIENTO</a></li>
20       <li><a href="phpPERSONAL/0_encuadramiento.php">PERSONAL</a></li>
21     </ul>
22   </nav>
23 </header>
24
25 <body>
26
27   <h1>Sistema de Administración y Gestión Cruz del Sur</h1>
28 <main>
29
30   <div class= columnas>

```

Figura N° 9: Código de la función inicio.

Fuente: Elaboración propia (2023).

Finalmente, el demostrador tecnológico se completó con éxito mediante la integración de la base de datos, el desarrollo *del back-end* y *front-end*, y la carga de datos desde el sistema SIGLE al demostrador tecnológico.

## 2.6 Validación del demostrador tecnológico del software

Finalmente se abordó la fase de validación del demostrador tecnológico, una etapa crítica en la que el proyecto fue sometido a un riguroso escrutinio. Esta validación se llevó a cabo en dos dimensiones clave:

En primer lugar, la validación de su lógica interna, que implicó pruebas unitarias para asegurar el funcionamiento correcto de cada componente del demostrador, como así también pruebas unitarias de la seguridad para corroborar la robustez del sistema. Pruebas que fueron aprobadas exitosamente.

En segundo lugar, la evaluación de su conformidad con los requisitos y expectativas establecidos por el usuario, realizada a través de pruebas de aceptación que involucraron a los usuarios del DCO, quienes, utilizando el demostrador tecnológico del software de administración y gestión de activos, lo evaluaron positivamente.

Por último, se procedió a la selección de un panel de expertos calificados para llevar a cabo la validación de esta etapa final. Su elección se basó en criterios rigurosos, incluyendo aspectos cruciales como su formación académica y su vasta experiencia en áreas claves relacionadas. Esta selección meticulosa garantiza la obtención de una perspectiva imparcial y objetiva durante todo el proceso de validación.



Figura N° 10: Interfaz de inicio del demostrador tecnológico. Diseño final.

Fuente: Elaboración propia (2023).

### 3. CONCLUSIONES

El presente artículo ha tenido como objetivo principal abordar la problemática que enfrenta la administración y gestión de los activos de la Fuerza de Paz Conjunta Combinada "Cruz del Sur", específicamente en el DCO, que forma parte del Regimiento Logístico N° 2 "Arsenales de Guerra". Para lograr esto, se aplicó la metodología CEPAL, que permitió identificar a las partes involucradas, analizar las causas y efectos del problema, y definir claramente el problema de investigación.

Utilizando la metodología antes mencionada podemos concluir que se exploraron y evaluaron diferentes soluciones con la participación de un grupo de expertos, empleando el

proceso jerárquico analítico para determinar la solución óptima, que consiste en el diseño de un software destinado a la administración y gestión de los activos de la FPCC "Cruz del Sur".

Posteriormente, se procedió a adquirir conocimientos teóricos sobre los activos en cuestión y sobre la ingeniería de software, así como de la importancia de las herramientas informáticas en la eficiencia organizacional. Se estableció un marco metodológico basado en cinco objetivos específicos para guiar el proceso de desarrollo del software, lo que permitió avanzar de manera ordenada en la obtención de un diagnóstico organizacional.

También es importante indicar, como conclusión al desarrollo de la investigación, que, gracias al uso de una secuencia metodológica, se abordó de buena forma el levantamiento de los requerimientos, un paso crítico que implicó trabajar en conjunto con el personal del DCO para identificar y documentar sus necesidades en un conjunto de Especificaciones de Requerimientos del Software.

Con los requerimientos claramente definidos, se procedió al diseño conceptual del software, siguiendo la metodología de capas propuesta por Roger S. Pressman. Esta metodología resultó en el diseño de capas que abarcan el modelo de datos, la arquitectura del software basada en un modelo cliente-servidor, las interfaces y los componentes del software. Este diseño sentó las bases sólidas para el desarrollo y codificación del software, asegurando que esté en consonancia con los requerimientos establecidos.

El siguiente paso consistió en construir y programar el demostrador tecnológico, para lo cual se utilizó la herramienta AHP para seleccionar las combinaciones óptimas de lenguajes de programación, resultando en la elección de PHP, HTML, CSS y MySQL. La programación se dividió en tres etapas: la creación de la base de datos, el desarrollo del *back-end* y el *front-end*.

Luego, se completó con éxito el demostrador tecnológico al integrar la base de datos, desarrollar el *back-end* y *front-end*, y cargar los datos desde el sistema SIGLE al demostrador. Este logro fue validado a través de pruebas realizadas por el autor, usuarios y un panel de expertos.

Finalmente, es importante resaltar que las herramientas informáticas están experimentando una evolución acelerada. Por tanto, resulta fundamental para las organizaciones tanto el empleo como la actualización constante de estas herramientas. Todo esto con el propósito de conservar una ventaja competitiva. En el caso de las unidades de la institución, este uso y actualización continua representan la clave para mantener una ventaja estratégica-operativa significativa.

## REFERENCIAS

- Aldecoa, J. M. (2020). Niveles de madurez de la tecnología. Una introducción.
- CAF. (2019). Manual de codificación del material por rubros en el sistema de información y gestión logística del Ejército. Santiago: IGM.
- Gutiérrez, A. (2018). Tecnologías de la Información. México: Marcombo.
- Ministerio de Defensa Nacional. (2017). Libro de la Defensa Nacional de Chile. Santiago.
- Oracle. (15 de mayo de 2023). Java. Obtenido de [https://www.java.com/es/download/help/whatis\\_java.html](https://www.java.com/es/download/help/whatis_java.html)
- Ortegón, E.; Pacheco, J. y Roura, H. (2005). Metodología general de identificación, preparación y evaluación de proyectos de inversión pública. Santiago, Chile: Naciones Unidas.
- Pressman, R. S. (2010). Ingeniería de Software, un enfoque práctico. México: McGraw Hill.
- Servicio de Impuestos Internos. (03 de noviembre de 2023). SII. Obtenido de [https://www.sii.cl/diccionario\\_tributario/dicc\\_a.htm](https://www.sii.cl/diccionario_tributario/dicc_a.htm)
- Silberschatz, A. (2002). Fundamentos de las bases de datos. Madrid: España.
- Sommerville, I. (2011). Software Engineering. Edinburgh: Pearson.