



Desarrollo de un Aplicativo Web que permita la gestión de inventario, mantenimiento y control de manera integrada en la bodega de comunicaciones de La Escuela de Formación de Soldados “Vencedores del Cenepa”

Sailema Jiménez, José Luis

Departamento de Eléctrica y Electrónica

Carrera de Tecnología Superior en Redes y Telecomunicaciones

Trabajo de Unidad de Integración Curricular, previo a la obtención del título de
Tecnólogo Superior en Redes y Telecomunicaciones

Ing. Caiza Caisabuano, José Rubén

13 de febrero de 2023

Latacunga

14/2/23, 6:57

Monografía

Informe de originalidad

NOMBRE DEL CURSO

Revisión Monografía

NOMBRE DEL ALUMNO

JOSE LUIS SAILEMA JIMENEZ

NOMBRE DEL ARCHIVO

JOSE LUIS SAILEMA JIMENEZ - Monografía

SE HA CREADO EL INFORME

13 feb 2023

Resumen

Fragmentos marcados	4	3 %
Fragmentos citados o entrecorillados	2	1 %
Coincidencias de la Web		
occamagenciadigital.com	2	2 %
infranetworking.com	1	1 %
concepto.de	1	0,5 %
hostinger.es	1	0,5 %
passeidireto.com	1	0,4 %

1 de 6 fragmentos

Fragmento del alumno MARCADO

Protocolo: Un protocolo es un conjunto de normas o reglas y pasos establecidos de manera clara y concreta sobre el flujo de información en una red estructurada. **Servicios:** Un servicio es un...

Mejor coincidencia en la Web

Protocolo: Un protocolo es un conjunto de normas o reglas y pasos establecidos de manera clara y concreta sobre el flujo de información en una red estructurada. **Servicios:** Un servicio es un conjunto...

Modelo cliente servidor: ¿Qué es? Características, Ventajas y ... <https://blog.infranetworking.com/modelo-cliente-servidor/>



Ing. Caiza Caizabuan, José Rubén Mgrt.

C.C: 050265429-6



Departamento de Eléctrica y Electrónica

Carrera de Tecnología Superior en Redes y Telecomunicaciones

Certificación

Certifico que el Trabajo de Unidad de Integración Curricular **“Desarrollo de un Aplicativo Web que permita la gestión de inventario, mantenimiento y control de manera integrada en la bodega de comunicaciones de La Escuela de Formación de Soldados “Vencedores del Cenepa.”** fue realizado por el señor **Sailema Jiménez, José Luis**; el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Latacunga, 13 febrero del 2023

Ing. Caiza Caizabuan, José Rubén MSc.

C.C.: 050265429-6



Departamento de Eléctrica y Electrónica

Carrera de Tecnología Superior en Redes y Telecomunicaciones

Responsabilidad de Autoría

Yo, **Sailema Jiménez, José Luis**, con cédula de identidad N° **1804981742**, declaro que el contenido, ideas y criterios del Trabajo de Unidad de Integración Curricular: **Desarrollo de un Aplicativo Web que permita la gestión de inventario, mantenimiento y control de manera integrada en la bodega de comunicaciones de La Escuela de Formación de Soldados “Vencedores del Cenepa”**, es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y refiriendo las citas bibliográficas.

Latacunga, 13 febrero del 2023



Sailema Jiménez, José Luis

C.C.: 180498174-2



Departamento de Eléctrica y Electrónica

Carrera de Tecnología Superior en Redes y Telecomunicaciones

Autorización de Publicación

Yo, **Sailema Jiménez, José Luis**, con cédula de identidad N° **1804981742**, autorizo a la Universidad de las Fuerzas Armadas ESPE, a publicar el Trabajo de Unidad de Integración Curricular: **Desarrollo de sistemas informáticos que permitan mejorar la gestión de inventarios, mantenimientos, infraestructuras, abastecimientos, material bélico, entre otros aspectos, de manera integrada** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Latacunga, 13 febrero del 2023

Sailema Jiménez, José Luis

C.C.: 180498174-2

Dedicatoria

Me permito dedicar primeramente el presente trabajo de investigación a Dios todopoderoso, siendo él, el que me da la fuerza para realizar todo lo que me propongo.

También agradezco a mi familia y mis familiares quiénes me han dado el apoyo y la ayuda para poder llevar a cabo esta investigación y que está haya sido terminada a tiempo con ayuda de sus esfuerzos y mi dedicación.

Sailema Jiménez, José Luis

Agradecimiento

En primer lugar, doy gracias a Dios por permitirme continuar con mis estudios y darme la fuerza necesaria para concretar este proyecto también agradece agradezco a todas las personas que le han ayudado en especial a mi familia, quiénes han sido la fuerza que utilizado para continuar la investigación.

Agradezco a mi tutor de tesis quién fue la persona que se encargó de darme las ideas y a portarme correcciones para poder realizar un trabajo que sea factible y que cumpla con las funciones con las cuales está siendo desarrollado.

Sailema Jiménez, José Luis

ÍNDICE DE CONTENIDO

Carátula	1
Reporte de verificación de contenido.....	2
Certificación	3
Responsabilidad de Autoría.....	4
Autorización de Publicación	5
Dedicatoria	6
Agradecimiento.....	7
Índice de contenido	8
Índice de figuras	11
Índice de tablas	13
Resumen.....	14
Abstract	15
Capítulo I: Introducción.....	16
Antecedentes	16
Justificación.....	19
Objetivos	20
<i>Objetivo General</i>	20
<i>Objetivos Específicos</i>	20
Alcance.....	20
Capítulo II: Marco teórico	22
Aplicación Web.....	22
<i>Arquitectura cliente-servidor</i>	23
<i>Aplicación Web Estática</i>	24

<i>Aplicación Web Dinámica</i>	24
Framework	25
Servidor Web	25
Base de datos	26
Lenguaje de programación	27
<i>HTML</i>	27
<i>CSS</i>	28
<i>JavaScript</i>	28
<i>MySQL</i>	29
Metodología	29
<i>Modalidad de investigación</i>	29
<i>Investigación Experimental</i>	29
<i>Investigación Aplicada</i>	29
<i>Modelo en Cascada</i>	30
<i>Análisis y procesamiento de la información</i>	31
Capítulo III: Desarrollo y resultados	32
Modelo Cascada: Análisis.....	32
<i>Requerimientos de la aplicación web</i>	32
<i>Diagramas de flujo</i>	33
Modelo Cascada: Diseño.	37
<i>Bosquejo de aplicación</i>	37
<i>Modelamiento de la base de datos</i>	42
Modelo Cascada: Desarrollo.....	45
<i>Inicio de sesión</i>	46

<i>Stock de equipos</i>	48
<i>Gestión de prendas, Prestamos</i>	51
<i>Gestión de prendas, Devolución</i>	52
<i>Alertas</i>	53
<i>Listado de Personal</i>	55
<i>Reportes</i>	56
Modelo Cascada: Pruebas.	58
<i>Pruebas de Funcionalidad</i>	58
Análisis de resultados	64
Capítulo IV: Conclusiones y recomendaciones	66
Conclusiones	66
Recomendaciones	67
Bibliografía	68
Anexos	70

ÍNDICE DE FIGURAS

Figura 1. <i>Esquema de conexión entre cliente y servidor (base de datos).</i>	23
Figura 2. <i>Esquema de conexión entre cliente y servidor (base de datos).</i>	26
Figura 3. <i>Extracto de código HTML.</i>	27
Figura 4. <i>Modelo de desarrollo en Cascada.</i>	31
Figura 5. <i>Flujograma del Sistema de Inventario.</i>	34
Figura 6. <i>Flujograma para el manejo de los equipos del inventario.</i>	35
Figura 7. <i>Diagrama de flujo para la prestación y devolución de prendas.</i>	36
Figura 8. <i>Diagrama de Casos de Uso.</i>	37
Figura 9. <i>Diseño conceptual de página de Log In.</i>	38
Figura 10. <i>Diseño conceptual de página de visualización de equipos.</i>	39
Figura 11. <i>Diseño conceptual de página de Registro de equipos.</i>	39
Figura 12. <i>Diseño conceptual de página de Devolución.</i>	40
Figura 13. <i>Diseño conceptual de página de Prestamos.</i>	40
Figura 14. <i>Diseño conceptual de página de Alertas de Mantenimiento para equipos.</i>	41
Figura 15. <i>Diseño conceptual de página de Configuración (gestión del personal).</i>	41
Figura 16. <i>Modelo Conceptual.</i>	42
Figura 17. <i>Modelo Lógico.</i>	43
Figura 18. <i>Modelo Físico.</i>	44
Figura 19. <i>Tablas de la base de datos.</i>	44
Figura 20. <i>Sistema MVC.</i>	46
Figura 21. <i>Contraseñas cifradas con Base64.</i>	47
Figura 22. <i>Clase UserSession, para la gestión de la sesión de usuario.</i>	48
Figura 23. <i>Validación de la sesión.</i>	48
Figura 24. <i>Llamada a todos los equipos, en la base de datos.:</i>	49
Figura 25. <i>Creación de objeto JSON, para enviar a PHP.</i>	50

Figura 26. <i>Funciones CRUD que se ejecutan del lado del servidor al recibir el objeto JSON.....</i>	50
Figura 27. <i>Consulta a la base de datos, con la declaración “LIKE”.....</i>	51
Figura 28. <i>Manejo del Objeto y acondicionamiento de los datos.</i>	52
Figura 29. <i>Registro de observaciones y eliminación de datos.</i>	53
Figura 30. <i>Función para llamar los registros.....</i>	54
Figura 31. <i>Lista de equipos fuera del inventario.</i>	54
Figura 32. <i>Botones para registrar información.....</i>	55
Figura 33. <i>Formulario de registro de personal.....</i>	55
Figura 34. <i>Botones de acción para el personal.....</i>	56
Figura 35. <i>Evento para generar reportes.</i>	57

ÍNDICE DE TABLAS

Tabla 1. <i>Validación de Credenciales</i>	58
Tabla 2. <i>Validación de Credenciales erróneas</i>	59
Tabla 3. <i>Registro de equipos</i>	59
Tabla 4. <i>Editar la información de los equipos</i>	60
Tabla 5. <i>Cerrar la sesión vigente</i>	60
Tabla 6. <i>Cargar lista de equipos</i>	61
Tabla 7. <i>Gestionar información del personal</i>	62
Tabla 8. <i>Visualizar equipos en préstamos</i>	63
Tabla 9. <i>Devolución equipos</i>	64

Resumen

Durante los últimos años considerando el incremento del uso de los sistemas informáticos se ha visto siempre la necesidad de utilizar sistemas digitales Mediante los cuales podamos llevar un control más preciso y ordenado sobre los distintos equipos o componentes en un inventario. Por lo cual, mediante el desarrollo de un sistema informático que se ejecutará en un servidor en línea, utilizando un programa para visualizar la información (navegador web), sin la necesidad de descargar o instalar elementos adicionales, se puede llevar a cabo esta y otras funciones, siendo que, la presente investigación pretende desarrollar una aplicación web, mediante la cual se puede acceder utilizando credenciales previamente registrada, para así manipular lo que es la información de los equipos que se encuentran disponibles en el inventario. La aplicación web que se ha desarrollado va a permitir registrar, modificar y eliminar la información perteneciente a los diferentes equipos, así como también la información perteneciente a los distintos miembros del personal que estará a cargo de la bodega, y el personal que hará el préstamo de diferentes equipos, así como su devolución al inventario. Siendo que, mediante la aplicación se va a llevar el control de acceso de las personas al sistema de inventario, manejar el listado de las personas que han llevado material, equipos fuera del inventario y el listado de equipos devueltos facilitando el ingreso de observaciones y demás funciones que van facilitan el proceso de gestión del inventario.

Palabras clave: JavaScript, Php, MySQL, gestión de inventario.

Abstract

During the last years considering the increase in the use of computer systems, it has always been seen the need to use digital systems through which we can have a more precise and orderly control over the different equipment or components in an inventory. Therefore, through the development of a computer system that will run on an online server, using a program to view the information (web browser), without the need to download or install additional elements, you can carry out this and other functions, being that, this research aims to develop a web application, which can be accessed using previously registered credentials, in order to manipulate what is the information of the equipment that are available in the inventory. The web application that has been developed will allow to register, modify and delete the information pertaining to the different equipment, as well as the information pertaining to the different staff members that will be in charge of the warehouse, and the personnel that will make the loan of different equipment, as well as its return to the inventory. The application will control the access of people to the inventory system, manage the list of people who have taken material, equipment out of inventory and the list of returned equipment, facilitating the entry of observations and other functions that will facilitate the inventory management process.

Keys words: javascript, Php, MySQL, inventory management.

Capítulo I

Introducción

Antecedentes

Considerando el desarrollo de las tecnologías y la notable necesidad de digitalizar hacer más eficientes los sistemas de gestión de activos de cualquier entidad, se han diseñado sistemas que van desde aplicaciones de computadoras hasta aplicaciones que se ejecutan en el navegador o teléfonos inteligentes. Siendo que estas aplicaciones en sus inicios controlaban los activos (su entrada y salida del almacén) siendo mejorados hasta la actualidad, vinculándose a una base de datos remota y portando variedad de aplicaciones, como calendarios, alertas entre otras.

Analizando las investigaciones de años anteriores, se plantea a, Morales & Torres (2015), diseñando un sistema de inventario para la empresa Service Lunch, basándose en el modelo Coso 1, el cual es un estándar para el sistema de control interno mediante el cual evalúan y mejoran su sistemas en base a las observaciones durante la operación, con lo cual se aseguró mejorar el rendimiento de las operaciones y aumentar la productividad de la empresa, respecto a la gestión de insumos y recursos. De lo cual, se concluyó que debido a las falencias de la empresa para gestionar los procesos en general, el sistema de inventario basado en el modelo Coso 1, asegura un incremento en la eficiencia e incluso en los procesos políticos y administrativos de la empresa.

Por su parte, Rodriguez (2016), estableció que a la fecha de su investigación los procesos de las industrias eran pobremente sistematizados lo que incurría en altas perdidas y baja eficiencia de los procesos, esto propio de los sistemas que se ejecutaban. Pr lo cual centrando su investigación en la empresa MAGREB S.A. actualizó y optimizó su viejo sistema de control de inventarios, lo que dio como resultados una mejora en el uso de la materia prima, siendo evidenciado a nivel financiero puesto que la optimización del sistema facilitó administrar

correctamente la materia prima y otros recursos, ahorrando así dinero y aligerando la carga en la maquinaria en general.

Quizhpi Campoverde (2018), desarrollo e implemento un sistema de inventario que permite de forma permanente controlar tanto los productos como el sistema de clientes de la empresa, en el cual se manejan criterios que varían según el producto, y los clientes, logrando así obtener un sistema estable y que refleja su eficiencia en los registros contables y las valoraciones de los clientes. Algo a destacar en la investigación es que se consideró los criterios de seguridad, dado que la información tanto de los productos como de los clientes es privada, se estableció un servidor local, facilitando su acceso desde la red de la empresa, dificultando en gran medida el acceso externo, si no se disponía de la clave de conexión.

Carreño Dueñas *et al* (2019), planteo que la empresa Pymes, si bien tiene un sistema de gestión adecuado se puede optimizar para controlar en mejor manera los recursos de producción, siendo que la optimización del sistema iniciaría en la producción de dulces y se expandiría a otras áreas según los resultados obtenidos. Parte del sistema contaba con una memoria de cálculo para dosificación y demás criterios de control, pero que en general facilitaban administrar todos los procesos, así como los recursos.

En base a lo antes expuesto, se considera que el sistema a desarrollar debe ser optimizado para así facilitar futuras actualizaciones, así como también su manejo intuitivo. De las investigaciones previas se plantea implementar el sistema con base de datos de forma local para mayor seguridad de los datos, ya que no es necesario acceder a ella desde otro lugar, así como también el sistema de memoria, pero orientado al manejo de calendarios para así establecer las alertas en el sistema.

Planteamiento del problema

Los sistemas de inventario para el control y gestión de activos son de gran importancia para mantener el orden, control de los bienes y su uso por parte del personal. Tal como es el

caso de la Escuela de Formación de Soldados "Vencedores del Cenepa", en donde se maneja un sistema de inventario físico, utilizando recibos y bitácoras para la prestación y devolución de activos.

Dichos activos son solicitados y prestados para la ejecución eventos o actividades militares, siendo estos ultimo equipo estándar Mil, el cual requiere un mayor seguimiento, considerando que son materiales con piezas de difícil remplazo, además que los daños incurren en un mal funcionamiento del equipo en general y por consiguiente malestares durante las operaciones.

El mayor problema se presenta ya que con este sistema actual se dificulta llevar un control total del inventario, de modo que, si no existe un orden en el manejo de los equipos se deberá revisar manualmente cada equipo para consultar o actualizar la información, sea esta sobre él está actual, la cantidad y/o disponibilidad del equipo.

En este punto también se consideran la duración de los periodos de préstamo, los cuales, al cumplirse, no es posible tener certeza de la devolución de los bienes, a menos que se revisen las bitácoras de registro, o en su defecto no existe una alerta acerca de las fechas superadas para la devolución de los equipos.

En base a lo antes expuesto se considera que, actualizar el sistema físico de inventario, a una aplicación digital web permitirá llevar un control total y ordenado de inventario de la Escuela de Formación de Soldados "Vencedores del Cenepa", puesto que mediante el sistema se controlará en tiempo real, la disponibilidad, cantidad, prestados, fechas por vencer, estado préstamo y devolución de los activos; todo gestionado mediante el uso de códigos en los equipos y en el personal en general, lo que agilizará los procesos y el rendimiento del inventario.

Justificación

El desarrollo e implementación del sistema de inventario utilizando una aplicación web, servirá para mejorar el rendimiento y control del inventario de la Escuela de Formación de Soldados "Vencedores del Cenepa", dado que se podrá acceder en línea digitalmente a la información del inventario, facilitando así conocer las unidades disponibles y las prestadas, además de conocer rápidamente a nombre de quien fueron prestadas. Un sistema de inventario digital tiene varias ventajas en su aplicación, siendo que, para este proyecto se hace énfasis en controlar los equipos prestados y llevar un seguimiento más cercano a las unidades estándar mil, las cuales presentan mayor complejidad al momento de requerir reparaciones.

El inventario digital tendrá opciones a seleccionar el equipo a retirar, ingresar datos de retiro y entrega, mediante el uso de una base de datos personal (perteneciente a la escuela) en la cual se almacena toda la información para usos posteriores. Con ello, será capaz de generar informes en base a las observaciones ingresadas al momento de devolver las prendas, sean estas estándar mil o equipo de comunicaciones, los cuales en cualquier momento se podrá acceder al sistema y obtener los registros, además de las alertas del inventario para así enviar a mantenimiento los equipos que lo requieran o reabastecer el stock para los equipos que se descarten o estén fuera de funcionamiento.

Como tal, los beneficiarios directos de esta investigación es la Escuela de Formación de Soldados "Vencedores del Cenepa", dado que un control eficiente de los activos ayuda a disponer de menor manera los recursos y distribuirlos estratégicamente. De forma indirecta es posible implementar este sistema en otras áreas, generando su propia base de datos y ajustando los parámetros del sistema para que se ajusta a las actividades y equipos que se manejan.

Objetivos

Objetivo General

- Desarrollar el Aplicativo Web que permita una adecuada gestión de inventario, mantenimiento y control de manera integrada en la bodega de comunicaciones de La Escuela de Formación de Soldados “Vencedores del Cenepa”.

Objetivos Específicos

- Construir la infraestructura de software funcional para el control del inventario.
- Implementar el módulo en el aplicativo web de validación de códigos y alertas en la gestión de los activos.
- Diseñar la interfaz sencilla e intuitiva para la gestión del inventario.

Alcance

El alcance de la investigación tiene como objetivo diseñar un sistema de gestión de activos para la Escuela de Formación de Soldados "Vencedores del Cenepa", el cual remplazará el actual sistema de gestión el cual se controla por medios físicos (bitácoras y recibos), con lo cual se busca aumentarla eficiencia y tener mayor control sobre los equipos almacenados y los que se encuentran en campo durante la ejecución de actividades.

El sistema de inventario permitirá navegar entre las diferentes opciones

- Préstamo de prendas
- Devolución de prendas
- Generador de reportes general o por tipo de equipo
- Revisión de Alertas
- Manipulación de datos desde la base de datos
- RespalDOS digitales de información

Las limitaciones de este proyecto se rigen a controlar equipos de comunicaciones y estándar mil, dándoles su respectiva prioridad a cada uno, pues se busca tener la mayor disponibilidad de los equipos, lo cual se consigue mediante un adecuado sistema de gestión.

Capítulo II

Marco teórico

Aplicación Web

Se plantea a una aplicación como un software informático diseñado para cumplir o ejecutar una función mediante el ingreso o manipulación de parámetros o hardware. Las aplicaciones webs tienen el mismo principio de funcionamiento, con la diferencia basada en el tipo de ejecución, siendo que, las aplicaciones de escritorio deben instalarse y funcionan principalmente con los recursos del computador, mientras que las aplicaciones web requieren de una configuración y conexión a de red (sea esta local, intranet o extranet) para funcionar (David, Jordi, & Carles, 2004).

Las aplicaciones web no se instalan ni requieren de espacio en la memoria, ya que se acceden a ellas mediante una dirección de red, vinculada a un servidor del cual se obtienen, leen, modifican y guardan los datos.

Las aplicaciones web son de populares considerando que son más flexibles que las aplicaciones de escritorio, esto analizando por el lado de las mejoras, depuración y actualización del código, ya que las actualizaciones se suben al servidor reemplazando los datos viejos, y automáticamente se tendrá mejoras visibles para los que accedan a la dirección de red. A comparación de las aplicaciones normales, las cuales requieren de la descarga de paquetes y su instalación para evidenciar las mejoras.

Sin embargo, al optar por una aplicación web se debe considerar la robustez, complejidad y demás componentes de la aplicación, puesto que, bajo ciertos protocolos y estándares, se obliga a optar por un sistema u otro.

Arquitectura cliente-servidor

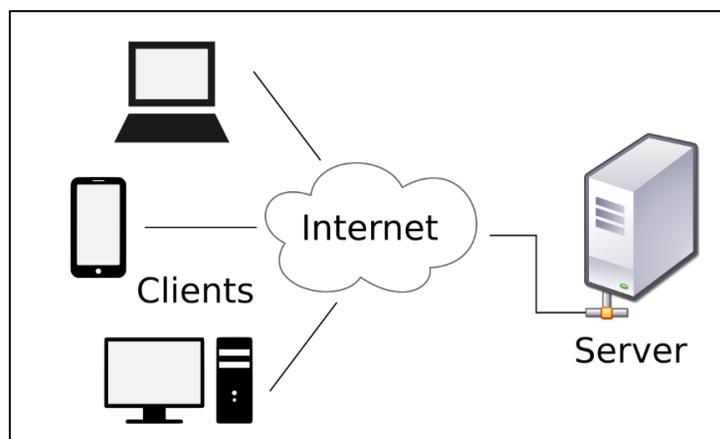
Consta de 2 partes. El cliente es el que inicia la solicitud de información, esto lo realiza mediante una página en el navegador. El servidor es cualquier recurso informático mediante el cual se da respuesta a las solicitudes del cliente. Los clientes están conectados a redes LAN, WAN, intranet o extranet (Arias Muñoz, 2018).

Internamente, también se consta de:

- **Protocolo:** Un protocolo es un conjunto de normas o reglas y pasos establecidos de manera clara y concreta sobre el flujo de información en una red estructurada.
- **Servicios:** Un servicio es un conjunto de información que busca responder las necesidades de un cliente, donde esta información pueden ser mail, música, mensajes simples entre software, videos, etc.
- **Base de datos:** Son bancos de información ordenada, categorizada y clasificada que forman parte de la red, que son sitios de almacenaje para la utilización de los servidores y también directamente de los clientes.

Figura 1

Esquema de conexión entre cliente y servidor (base de datos).



Nota. Arquitectura cliente-servidor para la conexión de la aplicación con la base de datos.

Aplicación Web Estática

Son páginas web diseñadas únicamente con la principal función de mostrar información, la cual suele ser constante o que presenta muy pocos cambios en un periodo de tiempo.

El diseño y estructuración de estas páginas, las hace pesadas de modificar, lo que conlleva tiempo considerable en su mantenimiento del código, además de que requiere acceder al servidor para efectuar los cambios.

Estas se caracterizan por tener poco o nula interactividad con el usuario, además de poseer pocas opciones de personalización y con funciones muy limitadas. Entre estas podemos encontrar, currículos online, portafolios digitales (Cerón Romo, 2005).

Aplicación Web Dinámica

A diferencia de las estáticas, estas se diseñan para mantener la interacción constante con el usuario, permitiendo manipular datos y personalizar parámetros, ya que estas acceden a la base de datos para presentarla información al cliente.

Las aplicaciones web dinámicas incluyen otros lenguajes además del HTML y CSS, para mejorar la interacción, tal como es PHP, NodeJs, Javascript, etc. Por lo cual, no se accede al servidor para dar actualización de la información, sino a la base de datos lo que implica una mejora en el rendimiento y los tiempos de consulta.

Por otro lado, al implicar los distintos lenguajes para mejorar la interacción, las aplicaciones web, tienen cierto nivel de robustez, que las vuelve complejas a nivel técnico, pero, aun así, poseen mayor flexibilidad que las estáticas (Cerón Romo, 2005).

Framework

Se plantea como un recurso mediante el cual se establece un orden y sistematización para el desarrollo de una aplicación o recursos informáticos; con estos es posible desarrollar un proyecto y dar facilidad de operación a los diferentes equipos que intervienen.

El Framework, según su proveedor puede contener diferentes librerías y API que faciliten el desarrollo de las aplicaciones informáticas, sin embargo, estas son diferentes a las librerías, ya que, el Framework ofrece una estructura base para el trabajo, y las librerías con básicamente complementos líneas de código que facilitan alguna funcionalidad que se pretende implementar (Cerón Romo, 2005).

Si bien, el uso de Framework ofrece diferentes ventajas como optimización del código, disposición de recursos y demás; este no es muy flexible lo que tiende a obligar a trabajar en un solo estilo con pocas personalizaciones, además la curva de aprendizaje de un Framework puede ser muy larga, ya que es necesario entender aspectos como la forma en la que se comunican los componentes, cómo se estructuran, entre otros (David, Jordi, & Carles, 2004).

Servidor Web

El servidor web es una aplicación informática la cual se encarga de enviar y recibir las solicitudes de información del cliente, esto se realiza mediante las direcciones IP y el DNS referencia del servidor y la página o aplicación web, con la cual saben de dónde viene la solicitud y hacia donde redirigir la información obtenida.

Los servidores web emplean para comunicarse diversos protocolos de datos, siendo el de uso común o estándar el HTTP (HyperText Transfer Protocol) o su variante de seguridad HTTPS (Hypertext Transfer Protocol Secure). Sin embargo, es posible también usar el término para referirse al computador en el que están guardados los archivos que componen un sitio web, junto al software necesario para cumplir con la conexión de datos web (MOSELEY, 2007).

Base de datos

Una base de datos es un almacenamiento de información de cualquier tipo, la cual se organiza según los requerimientos del programador o usuario final, para así enlazarlas y establecer coherencias o relaciones al momento de realizar consultas o modificar la información almacenada. En la base de datos es posible almacenar desde textos, números, fechas, hasta archivos de cualquier extensión como “.pdf”, “.mp4”, “.jpg”, etc.

Las bases de datos van enlazadas a las aplicaciones web permitiendo así modificar las tablas presentes en la base de datos y mantener el archivo origen en el servidor. Siendo una de sus principales ventajas que presenta, es la indexación de la información, lo que facilita busca y obtener resultados de manera eficiente (Pisco Gómez, 2017).

Figura 2

Esquema de conexión entre cliente y servidor (base de datos).



Nota. Modelo de comunicación entre el cliente (navegador) con el servidor de datos.

Si bien existen diferentes bases de datos en el mercado, todas emplean la misma sintaxis base para realizar las consultas, variando en sí, en los servicios adicionales que presentan, tal como capacidad de almacenamiento global y por registro, velocidad de consulta en simultaneo, protocolos de seguridad, etc. (Pisco Gómez, 2017).

Lenguaje de programación

El diseño y programación de las aplicaciones web, requieren del uso de diversos componentes para su desarrollo. Entre ellos se emplean diferentes lenguajes de programación con fines específicos, esto se hace con la finalidad de mantener separados los distintos componentes, como lo es la estructura, los estilos, el control y funciones, etc.

Es decir, para el desarrollo de la tesis en estudio, se utilizarán los lenguajes de programación web que se detallan en apartados posteriores, los cuales cumplirán con sus respectivas funciones como se detalla en cada uno (Núñez Miranda & Guerra Poalasin, 2021).

HTML

El lenguaje de marcado de hipertexto (HTML) es un lenguaje de programación sencillo planteado para la divulgación de información, es decir, para mostrar un texto e imágenes con fines específicos y con nula interacción del usuario, es decir, solo lectura.

El HTML consta de diferentes etiquetas con las cuales se realiza la maquetación de las páginas y aplicaciones web, generalmente se parte de este lenguaje para así establecer la disposición de la información y demás componentes de la página (Arias Muñoz, 2018).

Figura 3

Extracto de código HTML.

```
<p>Así es como se añade un párrafo en HTML.</p>
```

Nota. Ejemplo de código HTML, con el uso de etiquetas.

El HTML no es considerado un lenguaje de programación como tal, ya que no puede crear funcionalidades dinámicas. En su lugar, los usuarios web pueden crear y estructurar secciones, párrafos y enlaces mediante elementos, etiquetas y atributos.

CSS

CSS, en español «Hojas de estilo en cascada», es un lenguaje de “diseño gráfico”, su función principal es la de dar estilos a las páginas y aplicaciones web. Este lenguaje se usa en conjunto con HTML para así dar un mejor acabado al Front-End de la página web, pues como se especificó antes, su función es la de dar un mejor aspecto visual, y ayudar a optimizar los efectos que se han agregado (Arias Muñoz, 2018).

JavaScript

Es un lenguaje de alto nivel, mediante el cual se agregan funciones o comportamientos dinámicos a las páginas y aplicaciones ligadas a la web. Con la implementación de este lenguaje es posible agregar componentes dinámicos, mapas en tiempo real, notificaciones, animaciones interactivas, y las funcionalidades de otros lenguajes como la creación de clases, interfaces, funciones, herencias, etc.

En términos más simples, JavaScript cumpliría la función de un IDE como Java, dado que se pueden programar las mismas acciones, limitándose a los recursos disponibles del navegador en el cual se ejecute.

A esto, existen las denominadas interfaces de programación de aplicaciones (API) que son bloques de código JavaScript los cuales están diseñados para una función específica, los cuales se ajustarán a los requerimientos para su aplicación, su implementación ayuda a reducir el tiempo de diseño de componentes y depurar con mayor facilidad el código en general (MOSELEY, 2007).

MySQL

Considerando la potencia de las computadoras para manejar grandes cantidades de datos, los sistemas de administración de bases de datos desempeñan un papel central en la informática, como utilidades independientes o como parte de otras aplicaciones.

Siendo que, MySQL es un sistema de administración de base de datos, la cual se maneja (comúnmente) en conjunto con un servidor, se usa principalmente en aplicaciones web, y dado su simple pero eficaz lenguaje de programación es sencillo acoplarlo a los diferentes lenguajes para su ejecución (Pisco Gómez, 2017).

Metodología

La investigación se basa en las siguientes modalidades: experimental y aplicada.

Modalidad de investigación

Investigación Experimental

Modalidad de investigación dirigida a manipular información y variables relacionadas con el estudio en curso, con lo cual se controla y cuantifica los efectos del trabajo.

Con lo cual, se considera que esta modalidad de investigación se ajusta al tema ya que se debe plantear un modelo digital para manipular la información, siendo que, para que dicho modelo se efectúe, se deben plantear parámetros y funciones para las cuales se deben ejecutar diversas pruebas hasta obtener un funcionamiento adecuado y optimizada.

Investigación Aplicada

Se basa en dar una solución a problemas de índole prácticos, generando así un aporte al medio, ofreciendo documentación e información con un apoyo práctico y teórico. Este modelo de investigación se aplica ya que, en capítulos anteriores se estableció el problema existe, así como la solución, desencadenando en el desarrollo de la aplicación web, la cual ayudara a llevar un manejo más rápido y optimizado sobre el inventario de los equipos.

Modelo en Cascada

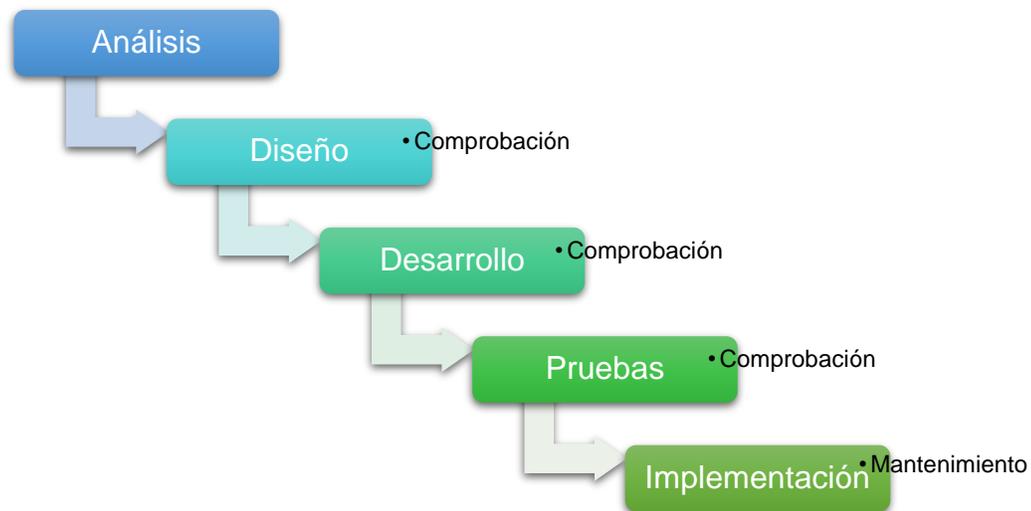
Es una forma de desarrollo de software en la que se siguen una serie de pasos estrictos y en orden. Esta metodología es muy detallada y deja poco margen de maniobra para el equipo de desarrollo. A menudo, se usa en proyectos grandes y complejos en los que es importante que todos los pasos se cumplan de forma estricta.

Se mantiene a lo largo de todo un ciclo de vida de un proyecto de software, y se caracteriza porque suele emplearse cuando los requisitos del sistema son bien conocidos y no se esperan grandes cambios durante el desarrollo.

Aunque existen enfoques en cascada para varios tipos de proyectos de software, esta metodología se adapta mejor a los proyectos estructurados, que se caracterizan por tener una planificación detallada, un enfoque en etapas, un análisis y diseño exhaustivos, un control de calidad y una fase de implementación en la que se usa un lenguaje de programación muy estructurado (Maida & Pacienza, 2015).

Figura 4

Modelo de desarrollo en Cascada.



Nota. Modelo en Cascada para el desarrollo de aplicaciones y software, basado en la implementación de características y comprobar su funcionamiento, teniendo la facilidad de regresar a etapas anteriores, para modificar parámetros.

Análisis y procesamiento de la información

Considerando que se emplea el modelo de desarrollo en cascada, en donde se realiza una comprobación del software con cada modificación y fase del proyecto, se plantean procedimientos para el manejo de la información:

- Recolección de la información basada en los requerimientos para el manejo del inventario a partir de una aplicación web.
- Análisis e interpretación de la información.
- Modelado de los casos de uso y diagramas de flujo en base a la información.
- Diseño de los procesos de la aplicación.

Capítulo III

Desarrollo y resultados

Modelo Cascada: Análisis.

Requerimientos de la aplicación web

El desarrollo de aplicaciones, en general, cumplen una serie de pasos mediante los cuales se asegura cumplir los requerimientos preliminares de la aplicación a desarrollar, de forma simplificada se considera:

Análisis de los requerimientos.

Destinado a determinar las plataformas en las cuales se ejecutará y por ende a estimar la robustez del sistema, con lo cual se determinan requerimientos del equipo como potencia del procesador, memoria RAM y espacio en disco. Dado que, la investigación busca desarrollar una aplicación web, esta se ejecuta en navegadores, de forma que su consumo de recursos es mínimo, permitiendo su ejecución en cualquier navegador web.

Diseño conceptual.

Se construye acorde a un esquema en el cual se representan los diferentes objetos de la aplicación, aquí se consideran tanto los objetos de Front-End (procedente de la parte interactiva y visual de la aplicación), como del Back-End (representa la codificación propia del servidor, es decir el encargado de comunicar el Front-End con la base de datos), siendo para este proyecto, los diferentes botones y los objetos procedentes de las consultas, este último teniendo en cuenta que las bases de datos devuelven objetos para cada consulta que se realiza. Con ello se añaden las relaciones y clases para gestionarla.

PROCESO

Diseño de navegación.

Con el diseño conceptual se establece el modelo de navegación a través de la aplicación, designando así a los respectivos componentes sus enlaces y funciones para así navegar entre paginas o los archivos de hipertexto. El diseño de la navegación básicamente establece los caminos a seguir en la aplicación, lo que ayuda al usuario a revisar los componentes que busca, e interactuar con los objetos que precisa, y además ayuda en la optimización de general de la página, cargando precisamente los elementos de la página cuando se accede a ella.

Diseño abstracto.

Todo el arte conceptual y diseño se plasma en código, en este tipo de aplicaciones se maneja la programación orientada a objetos, con los cuales se gestiona mejor la información manteniendo su velocidad de respuesta. Siendo que, la implementación del lenguaje del servidor (PHP), no afecta el diseño previo, ya que se ajusta a los objetos interactivos.

Con esto se asegura asignar los respectivos comportamientos a los botones, tablas y menú. Esta fase se basa en prueba y error a fin de asegurar una interacción agradable para el usuario.

Implementación.

En esta fase se desarrolla la parte gráfica y de diseño, se definen las paletas de colores, forma de los botones, aspectos de la interacción como señalización o enfoque de algún componente, fuentes, tamaños etc.

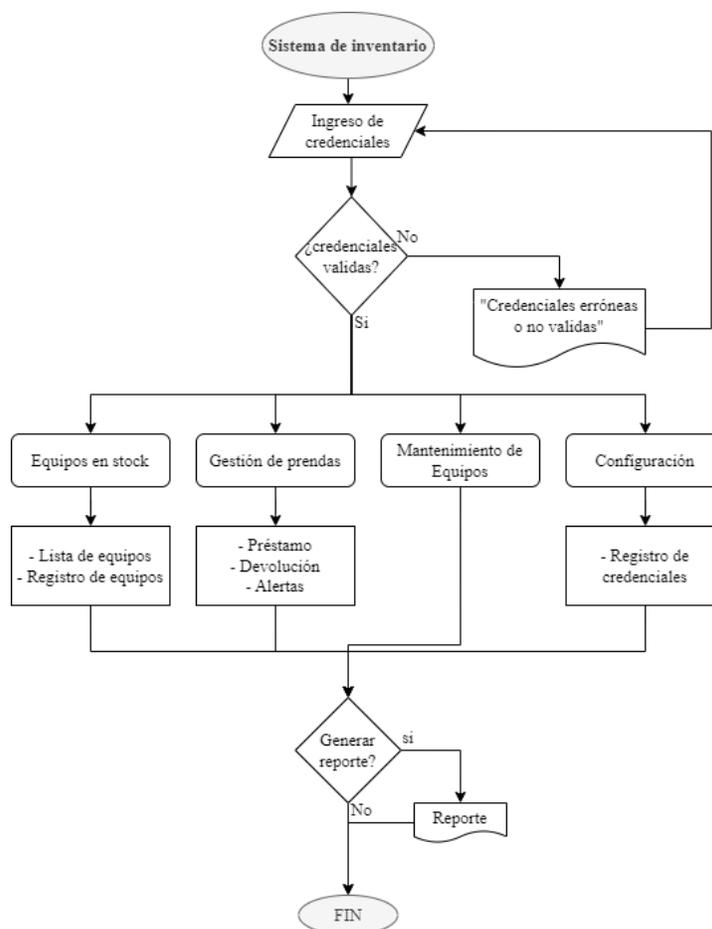
Diagramas de flujo

Considerando que los flujogramas son la representación gráfica de toma de decisiones, su implementación en el proyecto ayuda a definir las acciones que se pueden realizar y sus posibles repuestas.

Tal como se ve en la figura a continuación, en la cual se muestra el proceso de validación de credenciales, las cuales al no ser validas redirige el sistema al Log In de la página, o caso contrario la página principal, con los respectivos botones para cambiar de pestañas.

Figura 5

Flujograma del Sistema de Inventario.

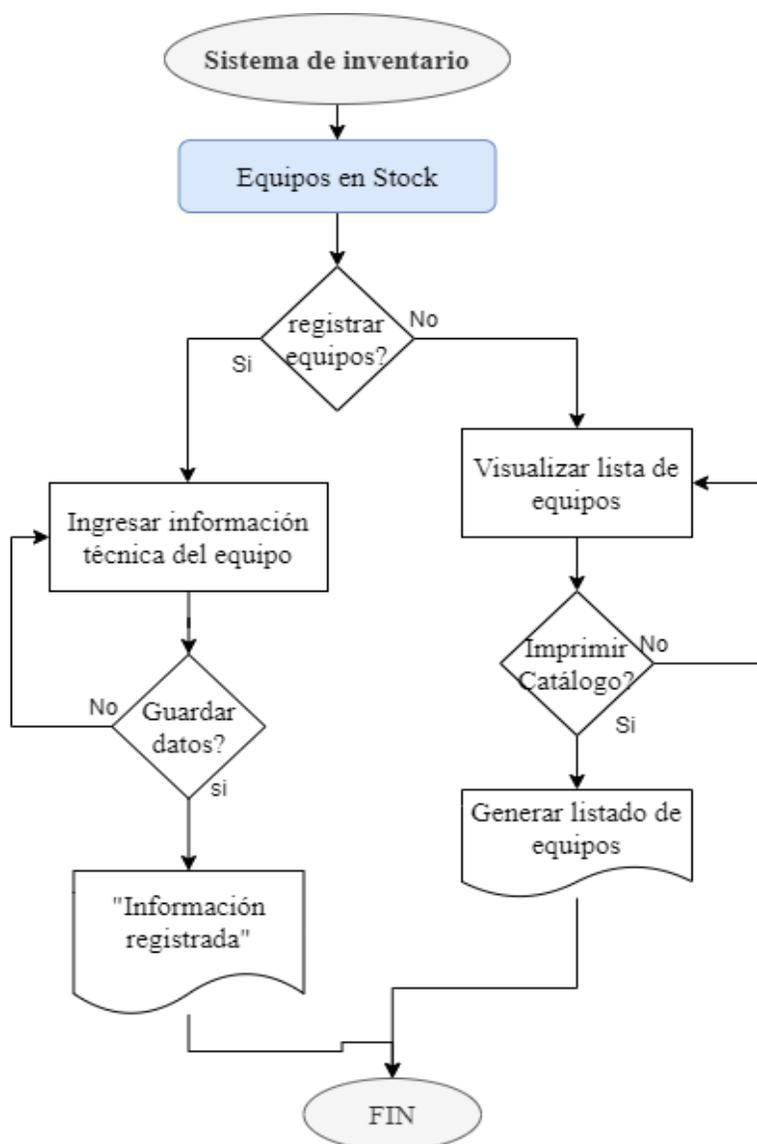


Nota. Diagrama de flujo acerca del funcionamiento de la aplicación.

La pestaña para el manejo de equipos se basa en 2 opciones, visualizar los datos ingresados hasta el momento o registrar equipos nuevos, finalizando en generar un documento “.pdf” con el listado de los equipos.

Figura 6

Flujograma para el manejo de los equipos del inventario.



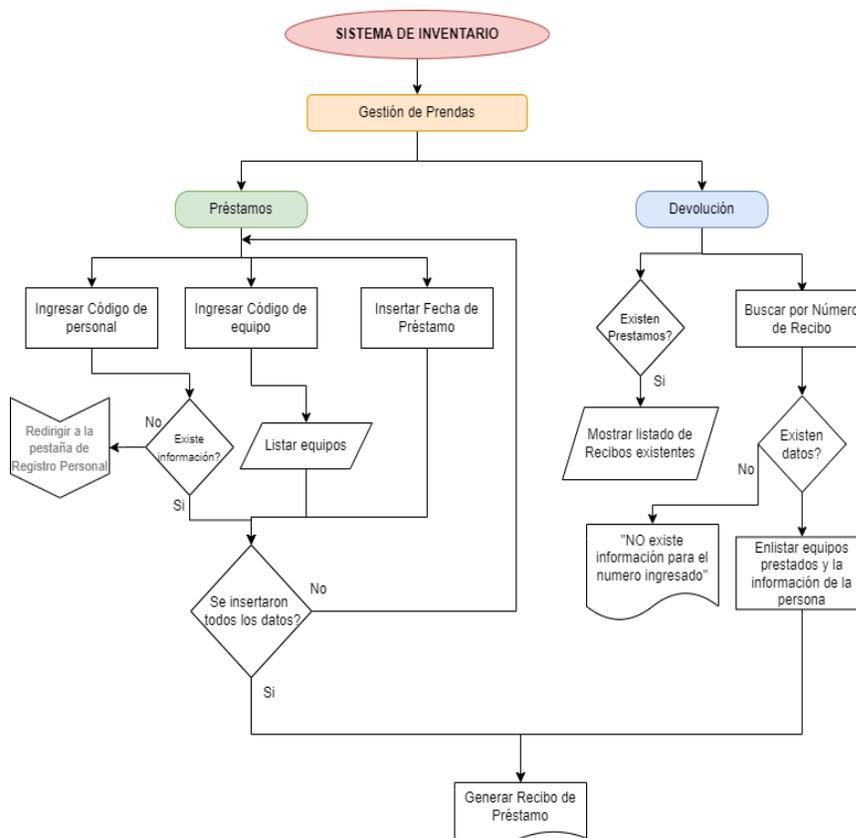
Nota. Diagrama de flujo acerca del funcionamiento de la aplicación.

El sistema para la gestión de prendas permite realizar el préstamo y devolución de las prendas, inicialmente al seleccionar “Préstamos”, se deben ingresar la información relacionada, como es el código de la persona que solicita las prendas, siendo que, mientras el código exista podrá continuar con el proceso, caso contrario el bodeguero tendrá la opción de registrar a dicha

persona en la aplicación para poder facilitarle los equipos. La búsqueda de equipos se realiza por el detalle o la especie del mismo, siendo que el sistema arroja todos los resultados que concuerden, siempre y cuando estén disponibles.

Figura 7

Diagrama de flujo para la prestación y devolución de prendas.



Nota. Diagrama de flujo acerca del funcionamiento de la aplicación.

Al momento de validar el préstamo, se verifica que toda la información haya sido ingresada, pues, de no ser el caso, se envían las alertas respectivas indicando que información no ha sido ingresada. Caso contrario se genera el recibo con toda la información ingresada.

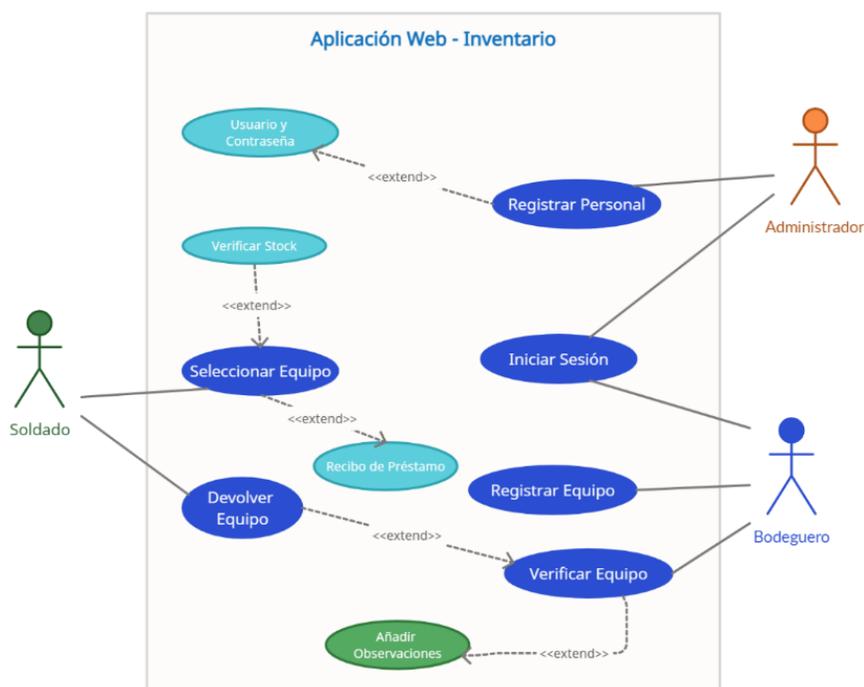
La página de “Devolución”, al iniciarse carga toda la información respecto a los préstamos realizados, pues muestra, un listado de los recibos generados (si existen), y al pulsar en alguno de ellos, se muestra su información respectiva. También existe la posibilidad de buscar directamente el código de recibo, lo cual arrojará la información que tiene asociada.

Modelo Cascada: Diseño.

Bosquejo de aplicación

Figura 8

Diagrama de Casos de Uso.



Nota. Casos de uso, aplicado a la aplicación web del proyecto.

Tal como se muestra en la figura 8, el sistema se maneja por 3 actores, siendo el bodeguero, el administrador, y la persona que solicita los equipos.

- El administrador posee todos los permisos para manipular la información.

- El bodeguero puede manipular los equipos y registrar personal para facilitar los préstamos, así como gestionar la devolución o revisión de préstamos vigentes.
- Por su parte el personal, únicamente puede proveer su información y el listado de equipos que requiere, siendo que es este último no debe manipular la aplicación.

Tal como se ve en las figuras 9-15, en base a los requerimientos de funcionamiento, orientado más la parte grafica se planteó una interfaz simple con pocos elementos, mediante los cuales se pueda tener una interacción intuitiva por parte del usuario.

Figura 9

Diseño conceptual de página de Log In.



The image shows a conceptual design for a login page. The page is titled "Iniciar Sesión" in the top right corner. The main content area is a dark green rectangle. At the top left of this rectangle is the ESPE logo, which includes a crest and the text "ESPE" in large green letters, followed by "UNIVERSIDAD DE LAS FUERZAS ARMADAS" and "INNOVACION PARA LA EXCELENCIA" in smaller white text. Below the logo, there are two input fields: "Usuario" with a placeholder "username" and "Contraseña" with a placeholder "password". Below the password field is a blue button with the text "Iniciar Sesión". At the bottom of the green rectangle is a dark green button with the text "No puedes iniciar sesión?".

Nota. Diseño conceptual de la UI (interfaz) de la aplicación web

Figura 10

Diseño conceptual de página de visualización de equipos.

ORD	Código	Nombre	Serie	Modelo	Marca-Raza...	Bodega	Ubicación
1	21143045	Equipos Prensa Radio y televisión	ESFORT1995	MOELESF2159	--	ESFORSE	Equipo de c...
2	21143045	Equipos Prensa Radio24 canales	ESFORT1995	MOELESF2159	--	ESFORSE	Equipo de c...
3	21143045	Equipos ONIX 24 MACKIE	ESFORT1995	MOELESF2159	--	ESFORSE	Equipo de c...
4	21143045	Equipos24 canales ONIX 24 MACKIE	ESFORT1995	MOELESF2159	--	ESFORSE	Equipo de c...

Nota. Diseño conceptual de la UI (interfaz) de la aplicación web

Figura 11

Diseño conceptual de página de Registro de equipos.

Nota. Diseño conceptual de la UI (interfaz) de la aplicación web

Figura 12

Diseño conceptual de página de Devolución.

The image shows a web application interface for 'Gestión de Prendas'. The top navigation bar includes 'Préstamo' and 'Devolución', with 'Devolución' being the active tab. On the left, there is a sidebar with the Ecuadorian coat of arms, '1925 ECUADOR', and menu items: 'Equipos en Stock', 'Gestión de Prendas', 'Mantenimientos', 'Configuración', and 'Cerrar Sesión'. The main content area is titled 'Formulario' and contains a 'Código de la orden' field with a checkmark icon. Below this is a table with columns: ORD, CANT., ESPECIE, SERIE, TIPO, and OBSERVACIONES. A 'Validar' button is located at the bottom right of the form area.

Nota. Diseño conceptual de la UI (interfaz) de la aplicación web

Figura 13

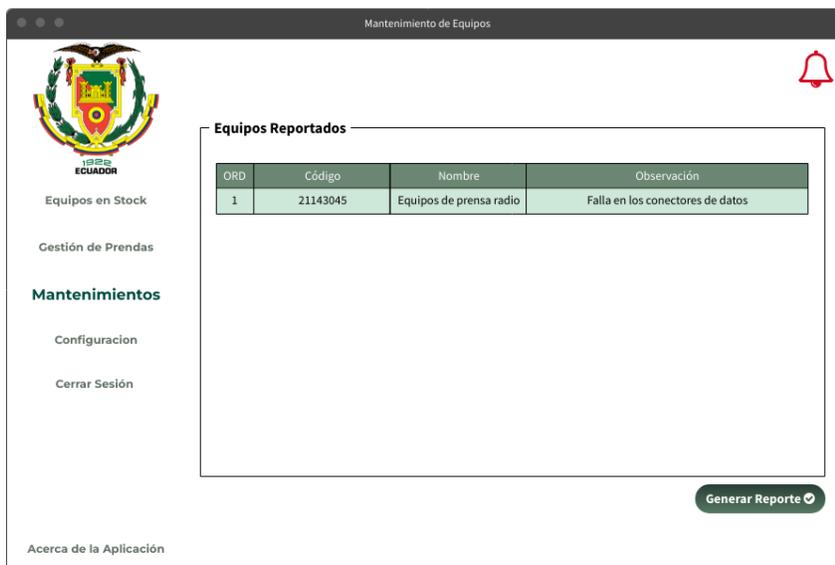
Diseño conceptual de página de Prestamos.

The image shows a web application interface for 'Gestión de Prendas'. The top navigation bar includes 'Préstamo' and 'Devolución', with 'Préstamo' being the active tab. The sidebar is identical to Figure 12. The main content area is titled 'Formulario' and contains two fields: 'Código Soldado' and 'Código Equipo', both with checkmark icons. To the right of these fields is a 'Detalle Equipo' section with an 'Agregar' button. Below the form is a table with columns: ORD, CANT., ESPECIE, SERIE, and TIPO. At the bottom right, there is a date field showing '12 Nov 2022' and a 'Generar Orden' button.

Nota. Diseño conceptual de la UI (interfaz) de la aplicación web

Figura 14

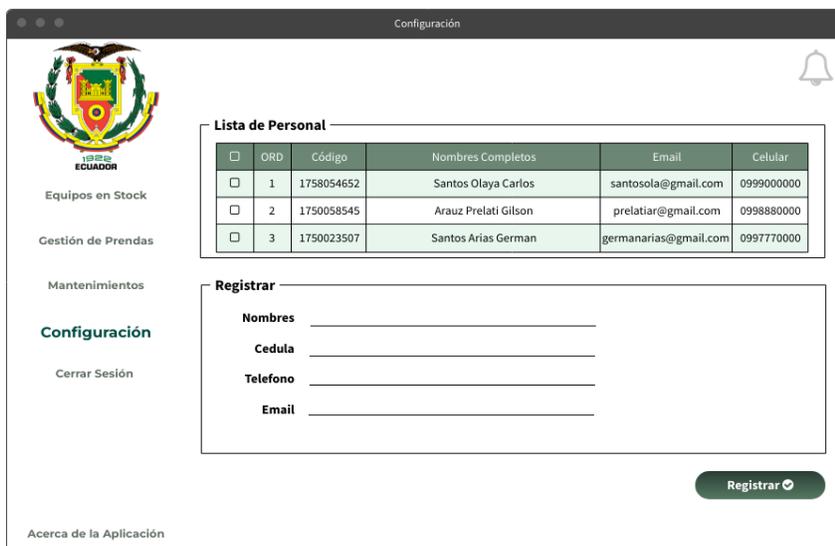
Diseño conceptual de página de Alertas de Mantenimiento para equipos.



Nota. Diseño conceptual de la UI (interfaz) de la aplicación web

Figura 15

Diseño conceptual de página de Configuración (gestión del personal).



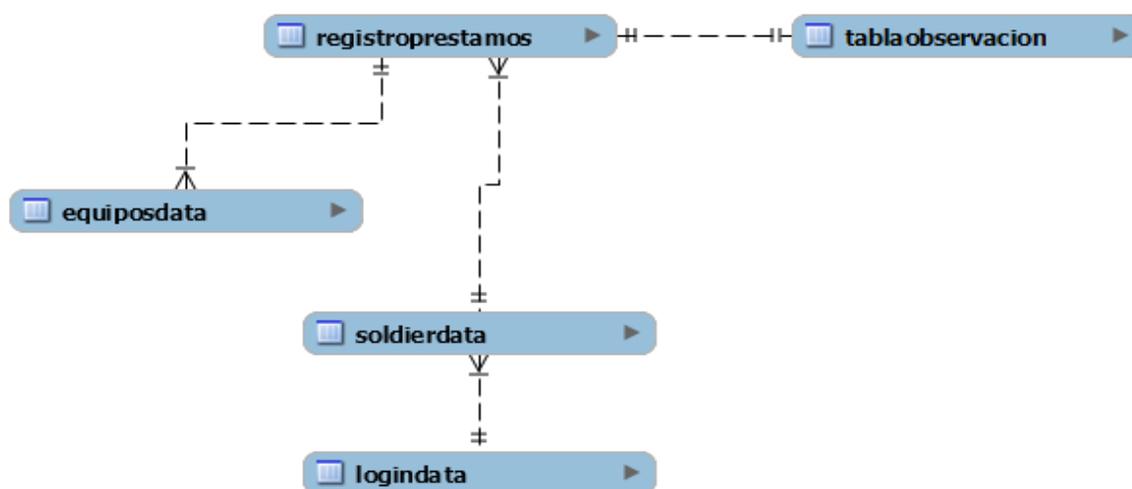
Nota. Diseño conceptual de la UI (interfaz) de la aplicación web

Modelamiento de la base de datos

Para el proyecto se utilizará una base de datos de MySQL, siendo de las más utilizadas en este tipo de aplicaciones.

Figura 16

Modelo Conceptual.



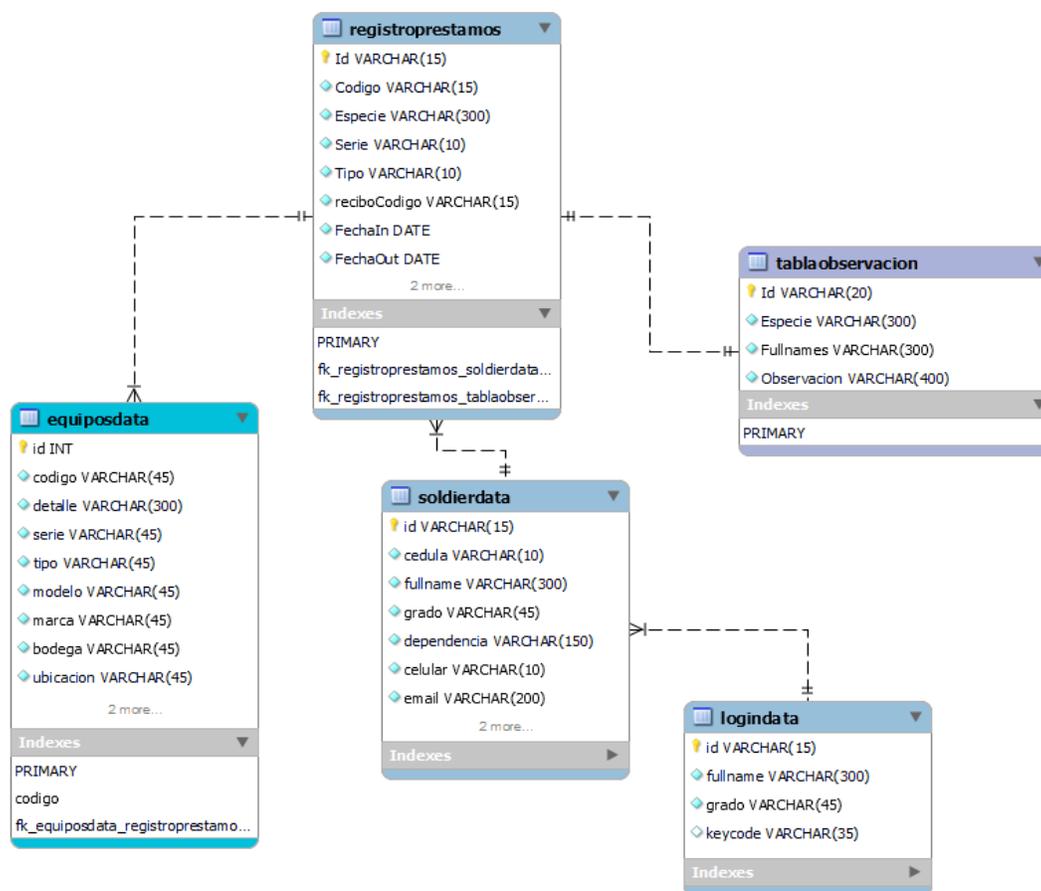
Nota. El modelo conceptual para la base de datos, ayuda a definir las necesidades de información de la base de datos.

La base de datos maneja una relación entre las claves primarias y foráneas existentes en cada tabla, ya sea, para el personal, o los equipos, lo que facilita las consultas al momento adquirir información relacionada a dichas llaves, para almacenarla en las modificarla en sus respectivas tablas, y procesarla para mostrarla al cliente para sus usos posteriores.

En base a la información recaudada procedente de bitácoras de gestión de equipos, sean estos de comunicaciones o de tipo militar, los recibos físicos que se manejan y las sugerencias del personal, se optó por manejar 5 tablas para la gestión de la información. Esto a modo de seguridad para evitar ambigüedades en el sistema, pues con algunos procesos al sobrescribir la información podrían generar errores o respuestas de tipo nulas.

Figura 17

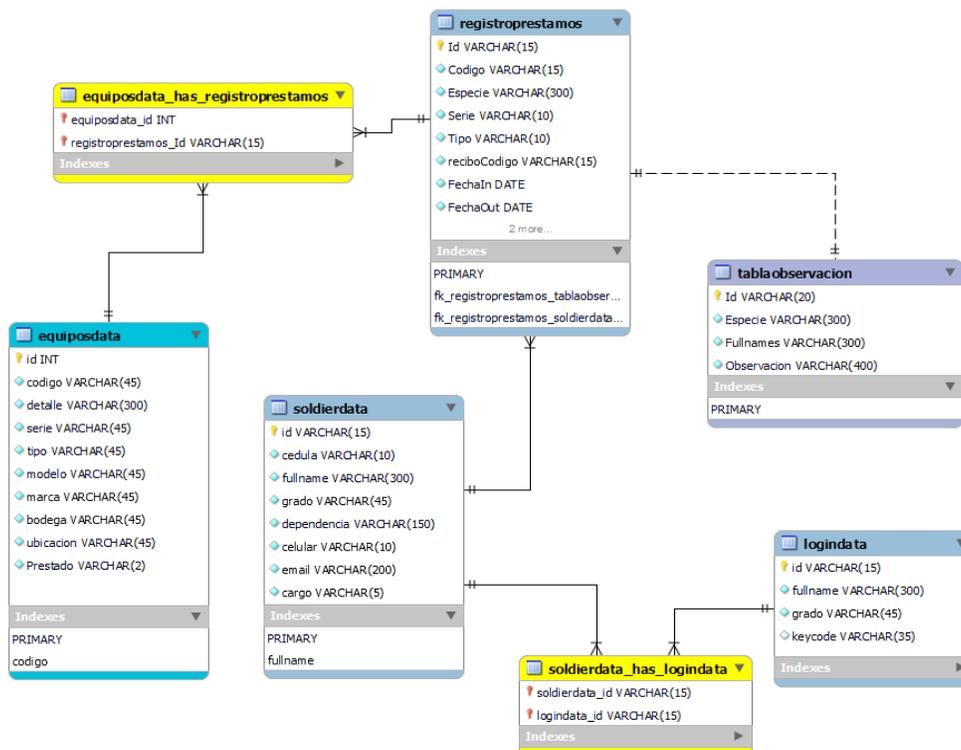
Modelo Lógico.



Nota. El modelo lógico de la base de datos, muestra la relación entre las tablas y sus datos.

Tal como se ve en la figura 16 y figura 17, las tablas modeladas presentan a detalle las columnas y el tipo de información que almacenará, así como la indexación entre ellas, al momento de acarrear la información, visualmente se observa que los enlaces indican la relación entre las tablas, tal como sería la tabla “*soldierdata*” y “*logindata*” que al existir la relación 1:1, por cada soldado registrado puede existir un usuario para iniciar sesión en la plataforma, sin embargo, internamente se limita la existencia de dichos usuarios. En otras tablas se maneja la relación “1: n”, donde por cada soldado se pueden tener varios equipos prestados.

Figura 18
Modelo Físico.



Nota. El modelo físico, ya presenta a detalle toda la información de la base de datos.

Figura 19
Tablas de la base de datos.

Filtros

Que contengan la palabra:

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño
<input type="checkbox"/> equiposdata	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	5	InnoDB	utf8mb4_general_ci	16.0 KB
<input type="checkbox"/> logindata	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	3	InnoDB	utf8mb4_general_ci	16.0 KB
<input type="checkbox"/> registroprestamos	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	12	InnoDB	utf8mb4_general_ci	16.0 KB
<input type="checkbox"/> soldierdata	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	5	InnoDB	utf8mb4_general_ci	16.0 KB
<input type="checkbox"/> tablaobservacion	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_general_ci	16.0 KB
5 tablas	Número de filas	25	InnoDB	utf8mb4_general_ci	80.0 KB

Nota. Tablas de la base de datos creadas en base al modelo físico.

Modelo Cascada: Desarrollo.

En esta etapa del proyecto, se implementa todo lo instanciado en apartados anteriores, siendo que se traduce todo el código y los aspectos conceptuales en lenguajes para la interacción del cliente o usuario, y el lenguaje encargado de ejecutar las consultas y manejo de la información.

Siendo que, para este proyecto se utilizaran los lenguajes antes mencionados, siendo:

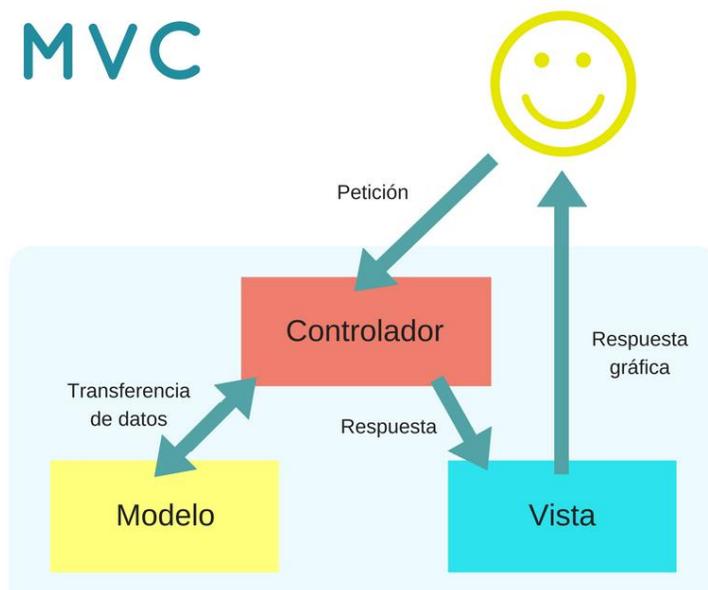
- HTML, para la maquetación de las distintas páginas de la aplicación.
- CSS, para aplicar estilos y formatos en la parte visual.
- JAVASCRIPT, para la interacción entre el usuario y la interfaz, así como también la manipulación de información
- PHP, para ejecutar las consultas, envía y recibir información.

En términos de programación, al estar orientada a objetos, se maneja el sistema MVC (Modelo Vista Controlador) el cual se basa en una arquitectura digital, con la cual se optimiza el manejo de la información, controlando las peticiones al servidor y haciendo una recesión de la información para su manejo, y posterior actualizar la base de datos, siempre y cuando sea necesario. Esto permite manejar grandes cantidades de información y a su vez permite acceder a la misma desde diferentes terminales.

- Modelo, gestiona la información procedente la base de datos. Este se encarga de ejecutar las consultas, aplicar filtros o actualizarla.
- Vista. Son las diferentes ventanas que tiene la aplicación en la cual se muestra la información y la interacción del usuario final.
- Controlador. Recibe las instrucciones y gestiona la información para presentarla, se mantiene en comunicación con la vista y accede al modelo, cuando es necesario.

Figura 20

Sistema MVC.



Nota. El MVC se emplea para optimizar tanto el desarrollo como el funcionamiento del software, haciéndolo más fácil de leer y mantener.

Inicio de sesión

Inicialmente se basa en recibir las credenciales y compararlas con la información en la base de datos si, existen los datos aprueba el ingreso y se instancia la página principal de la aplicación, siendo para este proyecto la página de listado de los equipos, caso contrario, emite una alerta y te redirección a la página de Log In.

A modo de seguridad, la aplicación maneja una codificación Base64, el cual es una codificación bidireccional alfanumérica. Si bien en comparación a otras no es la más segura, ofrece un nivel de seguridad aceptable para este tipo de aplicaciones.

Figura 21

Contraseñas cifradas con Base64.

grado	keycode
--	YWRtaW5Ac3VwcG9ydA==
Teniente	YWlyZ2V0bGFt
General	QWxsTWlnbGgyMDIy

Nota. Cifrado de contraseñas para asegurar mayor seguridad para el acceso a los datos.

Tal como se ve en la Figura 20, las claves son cifradas a caracteres alfanuméricos de hasta 32 caracteres, la codificación se vuelve más compleja según la complejidad y longitud de la contraseña normal.

Con esto definido, se utilizó una clase “*UserSession*” como se ve en la figura 18, con la cual se gestiona la creación de la sesión, el cierre de la sesión y la validación de la sesión en casa acceso. Esta clase se instancia a fin de evitar el acceso a algunas pestañas sin una contraseña, manteniendo así la seguridad de la información.

El extracto de código de la Figura 22, muestra cómo se recupera el usuario, siempre y cuando no se haya cerrado la sesión, según los datos almacenados, verifica las credenciales e instancia la última página que se visitó.

Figura 22

Clase *UserSession*, para la gestión de la sesión de usuario.

```

6 references | 0 implementations
class UserSession{

6 references | 0 overrides
public function __construct(){
    session_start();           //instancia una nueva session
}

0 references | 0 overrides
public function setCurrentUser($user){ //agrega un valor a la sesion actual
    $_SESSION['user'] = $user;
}

4 references | 0 overrides
public function getCurrentUser(){
    return $_SESSION['user'];
}

1 reference | 0 overrides
public function closeSession(){
    session_unset();           //borra las variables de la sesion
    session_destroy();         //destruye la sesion para evitar un ingreso sin credenciales
}

```

Nota. Funciones en PHP para el control de las sesiones en la aplicación web.

Figura 23

Validación de la sesión.

```

$userSession = new UserSession();
$user = new User();

if(isset($_SESSION['user'])){//Para volver al Home de la pagina
    $user->setUser($userSession->getCurrentUser());
    include_once BASE_DIR.'/vistas/settingsData.php';
}

```

Nota. Funciones en PHP para el control de las sesiones en la aplicación web.

Stock de equipos

En esta pestaña al cargarse se ejecuta la función listada en la figura 23, la cual llama todos los registros presentes en la tabla de “*equiposdata*”, y son ubicados en la tabla HTML para que sean visualizados por el usuario.

Figura 24

Llamada a todos los equipos, en la base de datos.

```
function callDataDB($DB){ //Llama todos los registros de los equipos
    $consulta = $DB->prepare("SELECT * FROM equiposdata");
    $consulta->execute();
    $listItems = $consulta->fetchAll();
    return $listItems;
}
```

Nota. Función para ejecutar consultas en la base de datos.

Para el registro, y modificación de la información se utilizan variables, en las cuales se hace el acondicionamiento de la información filtrándola de manera que, si existen datos, son cargados en las variables, caso contrario asigna un espacio vacío, esto debido a que la base de datos tiende a rebotar errores cuando se trata de insertar valores nulos en las tabas. Véase *Figura 24.*

Para enviar la información de HTML a PHP, se emplea JavaScript y JSON para transmitir la información mediante objetos. Si bien HTML posee componentes como formularios y los métodos “\$_POST” y “\$_GET”, para comunicarse con PHP, este método requiere de realizar las consultas constantemente, lo que incurre en un consumo de recursos y aumento de tráfico para acceder a la base de datos. Siendo que esto afecta la velocidad de consultas.

Utilizando JavaScript y los objetos JSON, se hace la comunicación con PHP sin la necesidad de actualizar la página, lo que permite mantener la información en objetos y acceder a ella las de forma indefinida y actualizarla solo cuando sea necesaria.

Con ello, se puede insertar la información y modificar las etiquetas de HTML con la información. Cabe recalcar que las aplicaciones que se ejecutan los navegadores, al refrescarse o actualizar la página, se ejecuta el código fuente desde cero, lo que vuelve a inicializar variables, objetos y componentes, eliminando así toda la información.

Figura 25

Creación de objeto JSON, para enviar a PHP.

```
var parametros = {
  "id"      : txtCodigo.value,
  "cedula"  : txtCedula.value,
  "name"    : txtNombres.value,
  "grado"   : txtGrado.value,
  "dependencia" : txtDependencia.value,
  "telf"    : txtTelefono.value,
  "email"   : txtEmail.value,
  "bodega"  : txtBodega.value,
  "consulta" : "Insertar Personal"
};
$.ajax({
  data: parametros,
  url: base_url+'serverInventario/MODELO/php/personalCRUD.php',
  type: 'POST',
```

Nota. Gestión de objetos JSON para comunicación entre PHP y JAVASCRIPT para su uso en el controlador del modelo.

El objeto JSON antes enunciado, son enviados a los diferentes métodos para ejecutar el CRUD en la base de datos, ejecutando cada función según el botón presionado, dado que cada uno maneja su identificador propio para evitar ambigüedades. Véase *Figura. 26*.

Figura 26

Funciones CRUD que se ejecutan del lado del servidor al recibir el objeto JSON.

```
1 reference
function DeleteRegistro($DB, $id){ //envia un query para borrar un registro de la tabla...
}
1 reference
function UpdateRegistro($DB, $id, $idDB,$detalle,$serie,$modelo,$marca,$bodega,$ubicacion){ //Actualiza los r
}
1 reference
function InsertRegistro($DB, $idDB,$detalle,$serie,$modelo,$marca,$bodega,$ubicacion){ //Inserta registros
}
```

Nota. Función para ejecutar consultas en la base de datos.

Gestión de prendas, Prestamos

El sistema de préstamos se basa en el manejo de objetos, de los cuales se obtienen la información, haciendo pocas solicitudes a la base de datos. El arreglo se actualiza según los criterios de búsqueda, utilizando la declaración “LIKE”, como se ve en la Figura 27.

Figura 27

Consulta a la base de datos, con la declaración “LIKE”.

```
function filledTable($conDB, $element){
    //obtencion de datos y converison a JSON
    $consulta = $conDB->prepare("SELECT * FROM equiposdata WHERE detalle LIKE '%$element%' AND Prestado='No'");
    $consulta->execute();
    $listItems = $consulta->fetchAll();

    $dataEquipo = array();
    foreach($listItems as $equipo){
        $data = array(
            'codigo' => $equipo['codigo'],
            'especie' => $equipo['detalle'],
            'serie' => $equipo['serie'],
            'tipo' => $equipo['ubicacion']
        );
        array_push($dataEquipo, $data);
    }
}
```

Nota. Función para ejecutar consultas en la base de datos.

Con esta sentencia y el uso de ‘% ... %’ se extraen los registros de determinada columna, mientras algunos caracteres coincidan. La información se acondiciona y se agrega a un objeto “*dataEquipo*”, para posterior convertirlo a un objeto manejable desde JavaScript. Al finalizar el proceso se actualizan los datos en las tablas, ya que se asigna in carácter a cada equipo prestado para evitar un registro duplicado.

Figura 28

Manejo del Objeto JSON y acondicionamiento de los datos.

```

//Creamos un objeto json en el cual guardamos la informacion de la tabla y despues lo pasamos a la DB
function itemSelect(inde){
  let date = new Date();
  var _ord = ('0' + count).slice(-2);
  var currentHora = date.getMinutes();

  if(nameSoldado != ''){ //Verificamos si se a agregado al soldado
    let newROW = TbodyPres.insertRow(-1); //Llenar la tabla con los datos
    let _column = _newROW.insertCell(0);
    _column.innerHTML = _ord;
    _column = _newROW.insertCell(1);
    _column.innerHTML = equipoJson[inde].codigo;
    _column = _newROW.insertCell(2);
    _column.innerHTML = equipoJson[inde].especie;
    _column = _newROW.insertCell(3);
    _column.innerHTML = equipoJson[inde].serie;
    _column = _newROW.insertCell(4);
    _column.innerHTML = equipoJson[inde].tipo;

    setPtrestamoItem(equipoJson[inde].codigo);

    dataObj.push({ //Copiar los datos al array
      "id": nameSoldado+"-"+_ord+"-"+currentHora,
      "codigo": equipoJson[inde].codigo,
      "especie": equipoJson[inde].especie,
      "serie": equipoJson[inde].serie,
      "tipo": equipoJson[inde].tipo,
      "comprobante": comprobante
    });
  }
}

```

Nota. Asignación de la información del objeto JSON en las diferentes celdas de la tabla.

En la figura 28, se observa el manejo del objeto “*equipoJson*” en JavaScript, de forma que mediante un índice se acceden a los datos y se asignan los campos necesarios para el recibo y a su vez, se crea un nuevo objeto que servirá para actualizar la base de datos.

Gestión de prendas, Devolución

El sistema de devolución manejo objetos al igual que los préstamos, con la diferencia, que utilizan un solo objeto para todo el proceso. En base al recibo del préstamo se buscan los datos y se muestran en pantalla, ya que no se busca acondicionar esa información, el objeto se destina para el ingreso de observaciones sobre el equipo (si es que se tiene alguna conservación).

De modo que, el objeto envía a la base de datos el código y nombre del equipo y su observación, y a la vez elimina los datos para ese recibo y asigna esos mismos equipos para que estén disponibles para ser prestados.

En la figura 29, se observa cómo se valida si el campo “observa” está vacío, en caso de no estarlo, se ejecuta la sentencia para guardar esa información. Y para el final se eliminan los datos, ya sea que existan o no observaciones, recalcando que ambos datos se manejan en diferentes tablas, pero relacionadas entre sí.

Figura 29

Registro de observaciones y eliminación de datos.

```

foreach($idSearch as $item){
    $code = $item['code'];
    $detail = $item['especie'];
    $observa = $item['observacion'];
    if($observa != ''){
        /*-----Registro Observacion-----*/
        $sql = "INSERT INTO tablaobservacion(Id, Especie, Observacion) VALUES (:code, :detail, :observa)";
        $consulta = $conexionDB->prepare($sql);
        $consulta->bindParam(':code', $code);
        $consulta->bindParam(':detail', $detail);
        $consulta->bindParam(':observa', $observa);
        $consulta->execute();
    }
    prestamoSet($conexionDB, $item['code']);
}
/*-----Eliminar DATos-----*/
$consulta = $conexionDB->prepare("DELETE FROM `registroprestamos` WHERE reciboCodigo='$numRecibo'");
$consulta->execute();

```

Nota. Función para ejecutar consultas en la base de datos.

Alertas

El sistema de alertas solo presenta la información referente a los equipos que han sido devueltos y presentan alguna novedad en su funcionamiento, y los equipos que hasta el momento se encuentran fuera del inventario.

Para lo cual se emplean 2 funciones con llamado a los préstamos y observaciones en la base de datos, tal como se ven la Figura 30.

Figura 30

Función para llamar los registros.

```
function _callRegistrosObservacion($conDB){
    //Llamada a todos los registros en la tabla = tablaobservacion
    $consulta = $conDB->prepare("SELECT * FROM tablaobservacion");
    $consulta->execute();
    $listItems = $consulta->fetchAll();

    $dataEquipo = array();
    foreach($listItems as $equipo){ //ejecutamos un bucle con todos los registros
        $sqlBB = "SELECT ObservacionEquipo FROM equiposdata WHERE codigo=:recibo";
        $consultaBB = $conDB->prepare($sqlBB);
        $consultaBB->bindParam(':recibo', $equipo['Id']);
        $consultaBB->execute();
        $resultBB = $consultaBB->fetch(PDO::FETCH_ASSOC); //información del detalle

        $data = array( //y los asignamos a un vectorArray
            'codigo' => $equipo['Id'],
            'especie' => $equipo['Especie'],
            'observacionEntrada' => $equipo['Observacion'],
            'observacionSalida' => $resultBB['ObservacionEquipo']
        );
        array_push($dataEquipo, $data); //añadís el vectorArray, al objeto de tipo array
    }
    echo json_encode($dataEquipo);
}
```

Nota. Función para ejecutar consultas en la base de datos.

Para estos últimos es posible visualizar la información de la persona que solicitó los equipos y a su vez el sistema se mantiene comparando las fechas de devolución con la fecha actual, de modo que presenta un resaltado rojo sobre los equipos que se han sobrepasado la fecha de devolución. Véase Figura 31.

Figura 31

Lista de equipos fuera del inventario.

Lista de Equipos fuera del Inventario

ORD	CÓDIGO	ESPECIE	TIPO	FECHA PRÉSTAMO	FECHA DEVOLUCIÓN	DEUDOR
1	21182505	Equipo prensa radio y television	Estandar Mil	2023-01-26	2023-01-20	
2	belt922208	Cinturón utilitario táctico militar equipado	Estandar Mil	2023-01-26	2023-01-20	
3	micA505002	Microfono de pedestal flexible Broadcast Plug	Equipos de Comunicación	2023-01-26	2023-01-20	
4	belt108246	Cinturón utilitario táctico militar con sistema modular de seguridad	Estandar Mil	2023-01-26	2023-01-28	

Nota. Visualización de los equipos fuera de inventario con su detalle.

Listado de Personal

La configuración de la aplicación, se modelo para registrar a todo el personal, y así facilitar y acelerar el proceso de préstamo y devolución de los equipos. Este sistema utiliza el mismo modelo empleado para registrar los equipos, habiendo diferencias en las tablas en las que se guarda la información y los diferentes campos a guardar. Para esto se accede al formulario desde el botón “Añadir”. Véase *Figura 32*.

Figura 32

Botones para registrar información.



Nota. Botones de la interfaz.

En dicho formulario se capturan los datos ingresados y son enviados a la base de datos. Para asegurar que los campos se llenen en su totalidad, HTML provee un atributo “*required*” el cual envía alertas cuanto algún campo del formulario está vacío o no cumple con el formato requerido, tal como sería el caso del correo electrónico, del cual el input es capaz de distinguir si cumple con el formato o no. Véase *Figura 33*.

Figura 33

Formulario de registro de personal.

 A screenshot of a registration form with the following fields:

- Nombres: Daniel Mendoza Palacios
- Grado: Sargento Primero (dropdown)
- Bodega: No (dropdown)
- Dependencia: Latacunga
- Teléfono: 0987522414
- Correo Electrónico: daniel_mendoza

 A red error message box is displayed below the email field, containing the text: "Incluye un signo '@' en la dirección de correo electrónico. La dirección 'daniel_mendoza' no incluye el signo '@'."

Nota. Formulario de registro.

Como tal, la ventana del personal, por cada persona registrada, muestra los botones para las respectivas acciones. Permitiendo editar, eliminar o ver los equipos prestados para cada persona. Internamente el sistema del inventario evalúa que cuentas de usuario es la que ha iniciado sesión, siendo que, el administrador, es la única cuenta capaz de interactuar con dichos botones, pues las demás cuentas solo podrán acceder al historial de prendas de dicha persona.

Figura 34

Botones de acción para el personal.

Teléfono	Correo	Bodega	Acción
0984555	alarcon@jose	No	
0988	altair@altair	Si	
000000123	test@jose@gmail.com	No	

Nota. Botones de la interfaz.

Como se ve en la Figura 32, el botón “Claves”, se tinta de rojo, para indicar que existe personal de bodega que no tiene registrada una contraseña. Esto se logra ejecutando un segundo llamado a la base de datos, en base al dato seleccionado en el punto “Bodega” del formulario en la Figura 31, al seleccionarse “Si”, automáticamente se agregan los datos para generar un usuario, y la contraseña se agrega a posterior. Si todo el personal tiene una contraseña ingresada, entonces el botón retorna a con color verde.

Reportes

El sistema de reportes se basa en imprimir las tablas y datos visibles de la página en cuestión, al presionar los botones se desencadena el evento “onclick” con el cual, se llama la función encargada de generar el reporte, con extensión en “.pdf”, pues, utilizando este método, se continúa trabajando con objetos y se limitan los accesos a la base de datos, lo que resulta en optimización del sistema.

Esto se realiza mediante una API o librería externa “DOMPDF”, la cual genera documentos *pdf*, de forma sencilla sin consumir muchos recursos, esta librería emplea una serie de funciones, las cuales toman los datos ya existentes y los plasma en el documento, con una serie de personalizaciones, como el tamaño, orientación de la página, fuentes, nombre del archivo, etc.

Así, se puede utilizar el mismo código para todos los reportes, cambiando el encabezado para cada situación.

Figura 35

Evento para generar reportes.

```
$html = ob_get_clean();  
//echo $html;  
  
require_once '../dompdf/autoload.inc.php';  
  
use Dompdf\Dompdf;  
$dompdf = new Dompdf();  
  
$options = $dompdf->getOptions();  
$options->set(array('isRemoteEnabled' => true));  
$dompdf->setOptions($options);  
  
$dompdf->loadHtml($html);  
$dompdf->setPaper('A4', 'portrait');  
  
$dompdf->render();  
$dompdf->stream("listado_equipos.pdf", array("Attachment" => false));
```

Nota. Codificación para copiar la información de pantalla y enmaquetarla en un documento pdf.

Modelo Cascada: Pruebas.

Las pruebas realizadas a la aplicación web, tratan de demostrar el funcionamiento de la aplicación y las diferentes funciones implementadas, evidenciando que estos son capaces cumplir efectivamente con las diferentes tareas ya programadas.

Si bien, con el modelo Cascada, las pruebas se realizan de forma continua, los resultados presentados a continuación corresponden a la fase anterior a la implementación o puesta en funcionamiento real de la aplicación.

Pruebas de Funcionalidad

Tabla 1

Validación de Credenciales.

Sistema	Iniciar Sesión
Nombre de la prueba	Validación de Credenciales existentes
Descripción de la acción	El bodeguero deberá ingresar a la aplicación con su código personal y la contraseña.
Condición de ejecución	Los datos del bodeguero deben estar almacenados en la base de datos, previo a la ejecución de la prueba.
Proceso:	a) Ingresar código y contraseña en los respectivos inputs. b) Dar clic en el botón "INICIAR SESION"
Resultado Obtenido:	Las credenciales al ser válidas muestran un mensaje de aprobación y se carga la página principal, mostrando datos de la sesión.
Evaluación del resultado:	El sistema funciona correctamente.

Tabla 2

Validación de Credenciales erróneas.

Sistema	Iniciar Sesión
Nombre de la prueba	Validación de Credenciales erróneas
Descripción de la acción	El bodeguero deberá ingresar a la aplicación con su código personal y la contraseña erróneas o que no estén registrados
Condición de ejecución	Los datos por ingresar no deben constar en la base de datos.
Proceso:	a) Ingresar código y contraseña en los respectivos inputs. b) Dar clic en el botón "INICIAR SESION"
Resultado Obtenido:	Las credenciales durante la validación arrojan un valor nulo, por lo tanto, no existen resultados coincidencias, y se muestra una alerta con dicho enunciado.
Evaluación del resultado:	El sistema funciona correctamente.

Tabla 3

Registro de equipos.

Sistema	Equipos en Stock
Nombre de la prueba	Registro de equipos
Descripción de la acción	El bodeguero deberá rellenar el formulario con la información del equipo.
Condición de ejecución	El código del equipo no debe constar en la base de datos
Proceso:	a) Dar clic en el botón "Añadir" b) Rellenar el formulario c) Dar clic en "Generar Registro"
Resultado Obtenido:	El código ingresado al no existir en la base de datos permitió insertar la información del formulario.
Evaluación del resultado:	El sistema funciona correctamente.

Tabla 4

Editar la información de los equipos.

Sistema	Equipos en Stock
Nombre de la prueba	Editar la información de los equipos
Descripción de la acción	El bodeguero deberá actualizar algún dato del equipo y posterior eliminar dicho registro.
Condición de ejecución	Debe existir equipos registrados con anterioridad.
Proceso:	<ul style="list-style-type: none"> a) Dar clic en el botón “editar” y modificar los datos en el formulario b) Dar clic en el botón “Actualizar Registro” c) Dar clic en el botón “Borrar”
Resultado Obtenido:	La información actualizada es enviada y se refleja en la tabla, ya que se actualiza al finalizar él envío de la información, de igual manera ocurre al eliminar un registro. Todo esto con sus respectivas alertas sobre las acciones que se realizan
Evaluación del resultado:	El sistema funciona correctamente.

Tabla 5

Cerrar la sesión vigente.

Sistema	Cerrar Sesión
Nombre de la prueba	Terminar la sesión vigente
Descripción de la acción	El bodeguero deberá cerrar sesión.
Condición de ejecución	Debe existir una sesión vigente.
Proceso:	<ul style="list-style-type: none"> a) Dar clic en el botón “Cerrar Sesión” b) La aplicación cargar la página de Inicio de sesión c) Dar clic en el botón “Regresar” del navegador d) El navegador deberá redirigir a página de Inicio de sesión
Resultado Obtenido:	El navegador pide confirmación al momento de dar clic en regresar, sin embargo, al actualizar, se mantiene en la página de inicio de sesión.
Evaluación del resultado:	El sistema funciona correctamente.

Tabla 6*Cargar lista de equipos.*

Sistema	Alertas
Nombre de la prueba	Cargar lista de equipos con novedades Cargar lista de equipos fuera del inventario
Descripción de la acción	El bodeguero deberá acceder a la página de Alertas, y observar la información que aparece.
Condición de ejecución	Previamente deberán existir equipos que hayan sido devueltos y se haya ingresado alguna observación, sobre el mismo. Y a su vez, existir prestamos vigentes.
Proceso:	a) Dar clic en el botón “alertas” b) Observar la tabla c) Acceder a la información de la persona a cargo de las prendas
Resultado Obtenido:	Las tablas cargan la información correctamente la cual, al compararse con los datos de las tablas en la base de datos, se evidencia que la información está siendo filtrada correctamente.
Evaluación del resultado:	El sistema funciona correctamente.

Tabla 7

Gestionar información del personal.

Sistema	Listado del Personal
Nombre de la prueba	Añadir, editar y eliminar personal de la base de datos de la aplicación
Descripción de la acción	El bodeguero deberá acceder a la página de Listado del personal, y añadir personal y editar su información, mediante el formulario que se despliega
Condición de ejecución	No se tienen condiciones previas
Proceso:	<ul style="list-style-type: none"> a) Dar clic en el botón “Listado del personal” b) Dar clic en el botón “Añadir” c) Llenar el formulario, y enviar los datos para registrar. d) Actualizar la pagina e) Observar la información agregada f) Dar clic en el botón “editar”, en la columna “Acción” para la persona a modificar sus datos. g) Rellenar el formulario con datos actualizados, y enviar formulario h) Actualizar la pagina i) Observar la información agregada j) Dar clic en el botón “eliminar”, en la columna “Acción” para la persona a eliminar sus datos. k) Actualizar la pagina l) Observar la información agregada
Resultado Obtenido:	<p>La información se añade y se modifica de manera exitosa, siendo que no se pierden caracteres, y se añaden en los campos correspondientes.</p> <p>Ya que se maneja un objeto de información pequeño, la aplicación no presenta ninguna dificultad al momento de utilizarla.</p>
Evaluación del resultado:	El sistema funciona correctamente.

Tabla 8

Visualizar equipos en préstamos.

Sistema	Listado del Personal
Nombre de la prueba	Visualizar los equipos que mantiene cada persona, en la aplicación
Descripción de la acción	El bodeguero deberá acceder a la página de Listado del personal y observar los equipos que cada persona a prestado en bodega
Condición de ejecución	Deberán existir al menos una persona registrada en el sistema, y la misma deberá mantener préstamos vigentes durante la prueba.
Proceso:	<ul style="list-style-type: none"> a) Dar clic en el botón “prendas”, en la columna “Acción” para la persona a visualizar sus datos. b) Se despliega un modal con información. c) Observar la información d) Comparar la información con el recibo emitido con anterioridad e) Presionar escape, para cerrar.
Resultado Obtenido:	La lista equipos y su información, concuerda efectivamente con la información presente en el recibo
Evaluación del resultado:	El sistema funciona correctamente.

Tabla 9*Devolución equipos.*

Sistema	Devolución
Nombre de la prueba	Devolución de equipos, por código de Recibo
Descripción de la acción	El bodeguero deberá acceder a la página de Devolución, y buscar el código del recibo a devolver, añadir alguna observación o detalle, si es necesario y validar los datos.
Condición de ejecución	Deberán existir al menos una persona registrada en el sistema, y la misma deberá mantener prestamos vigentes durante la prueba.
Proceso:	<ul style="list-style-type: none"> a) Dar clic en el botón "Devolución" b) En el campo de búsqueda, añadir el código del recibo y buscar. c) Revisar cada equipo, y añadir una novedad, si es que la tiene. d) Validar los datos.
Resultado Obtenido:	La búsqueda de la información se realiza de manera efectiva, sin embargo, al momento de validar las observaciones ingresadas, el sistema evaluar que los strings contengas, ciertas palabras para poder añadirlos a las tablas correspondientes, la ortografía, puede generar problemas, haciendo que la aplicación ejecute acciones por defecto, por palabras mal escribe
Evaluación del resultado:	El sistema funciona correctamente.

Análisis de resultados

La aplicación desarrollada al ejecutarse en un navegador web, opera con una arquitectura de cliente - servidor, siendo el cliente el personal de bodega el encargado de enviar las solicitudes de información y obtener una respuesta, mientras que el servidor (PHP) se encarga de obtener la información de la base de datos y gestionarla

Con las pruebas realizadas, se evidencia que la aplicación web tiene un funcionamiento eficiente, ya que maneja y acondiciona la información en función de las solicitudes, logrando así que, sea eficiente de fácil implementación y utilización.

Si bien, al ser desarrollada para navegadores, debido a la particularidad entre los distintos navegadores y con compatibilidad con la última versión de PHP 8.0, el rendimiento se ve afectado en algunos pocos navegadores.

En términos generales, las consultas y posicionamiento de la información se realizan en el aplicativo web, pero los tiempos de respuesta en cada navegador varían en cierta medida. Siendo Chrome, Opera y Brave, los navegadores con las menores diferencias en la ejecución de aplicaciones de este nivel y arquitectura.

Capítulo IV

Conclusiones y recomendaciones

Conclusiones

- El desarrollo del aplicativo web, sirve para automatizar y simplificar el proceso para el control, gestión, préstamos y devolución de activos fijos fungibles y no fungibles, para el inventario de la Escuela de Formación de Soldados "Vencedores del Cenepa" va a permitir acceder a la información y llevar un control de mayor precisión en los activos, pues en sus diferentes pestañas la aplicación proveerá de la información y datos para saber con precisión el estados de las prendas y la persona a cargo de ellas.
- EL aplicativo web se desarrolló con el modelo en cascada y la arquitectura cliente servidor, ejecuta las distintas solicitudes de información a la base de datos, la cual gestiona todos los datos mediante objetos de información (Programación Orientada a Objetos) de forma funcional y eficiente utilizando pocos recursos y permitiendo la reutilización de los objetos.
- Dado que se implementó un sistema basado en códigos, tanto para el personal como los equipos, para identificarlos de forma activa, la arquitectura de la aplicación se mantiene validando dichos códigos para así proveer la información correspondiente que se ha solicitado. De igual manera se emplean dichos códigos y la información almacenada, para proveer el bodeguero de alertas visuales sobre el personal o los equipos que mantienen haberes fuera de plazo.

- La interfaz diseñada para la aplicación maneja un esquema simple, centrando los mayores detalles en las botones de acción, siendo esta, la forma de indicar visualmente al usuario cuales son los puntos de interacción con la aplicación.

Recomendaciones

- Ya que la aplicación fue desarrollada con el modelo en Cascada y una modalidad no destructiva, es posible acceder al código fuente de la aplicación y ajustarla para que cumpla con requisitos de funcionamiento adicionales, siendo que, se puede conseguir un sistema más personalizado y capaz de cumplir con su función principal.
- Se utilizó una arquitectura cliente - servidor utilizando PHP, es recomendable realizar un mantenimiento del código cuando se lancen nuevas actualizaciones para PHP, puesto, sus desarrolladores deprecian algunas funciones y las reemplazan con funciones más eficientes y compatibles con otras tecnologías, dejando poco funcionales las aplicaciones con las tecnologías anteriores.
- Se recomienda que, al ser una aplicación de código abierto, añadirle módulos y demás componentes, para maximizar su funcionamiento y proveer de otras funcionalidades, para los usuarios, mejorando notablemente la gestión del inventario.

Bibliografía

- Arias Muñoz, M. A. (Agosto de 2018). *Desarrollo de una aplicación web para la mejora del control de asistencia de personal en la Escuela Tecnológica Superior de la Universidad Nacional de Piura*. Obtenido de <http://repositorio.uigv.edu.pe/handle/20.500.11818/2930>
- Cerón Romo, F. A. (2005). *Desarrollo Profesional de Aplicaciones Web con ASP.NET*. ISBN 84-609-4245-7.
- David, M. J., Jordi, M., & Carles, M. (2004). *Desarrollo de aplicaciones web*. Barcelona, Catalunya: r Eureka Media, SL, Primera edición: marzo 2004. ISBN: 84-9788-118-4.
- Diego Andrés, C. D., Luis Felipe, A. G., Erika Tatiana, R. O., & Felipe Javier, T. (febrero de 2019). *Diseño de un sistema para la gestión de inventarios de las pymes en el sector alimentario*. Obtenido de <https://www.redalyc.org/journal/816/81661270007/html/>
- ISO/IEC-JTC-1/SC-7. (2001). *ISO/IEC 9126-1:2001, Software engineering — Product quality — Part 1: Quality model*. ISO/IEC JTC 1/SC 7 Software and systems engineering.
- Maida, E. G., & Pacienza, J. (2015). *Metodologías de desarrollo de Software*. Obtenido de <https://repositorio.uca.edu.ar/handle/123456789/522>
- Morales Escalante, L. C., & Torres Villanueva, M. A. (2015). *Diseño de un sistema de control interno de inventario basado en el modelo Coso I para la empresa service Lunch en la ciudad de Guayaquil*. Obtenido de <http://bibliotecavirtualoducal.uc.cl/vufind/Record/oai:localhost:123456789-1447882>
- MOSELEY, R. (2007). *DESARROLLO DE APLICACIONES WEB (MANUAL AVANZADO)*. España: ANAYA MULTIMEDIA, Primera edición. ISBN 9788441522657.

Núñez Miranda, C. I., & Guerra Poalasin, D. O. (marzo de 2021). *Desarrollo de una aplicación Web progresiva para la gestión de pedidos aplicada a la empresa de ropa Hoxton.*

Obtenido de <https://repositorio.uta.edu.ec/handle/123456789/32673>

Pisco Gómez, Á. R. (2017). *Fundamento sobre la Gestio de bases de datos.* Obtenido de Editorial

Área de Innovación y Desarrollo,S.L. Primera Edicion. 2017:

[http://repositorio.unesum.edu.ec/bitstream/53000/2077/1/Fundamento%20sobre%20la%](http://repositorio.unesum.edu.ec/bitstream/53000/2077/1/Fundamento%20sobre%20la%20Gestio%20de%20bases%20de%20datos.pdf)

[20Gestio%20de%20bases%20de%20datos.pdf](http://repositorio.unesum.edu.ec/bitstream/53000/2077/1/Fundamento%20sobre%20la%20Gestio%20de%20bases%20de%20datos.pdf)

Quizhpi Campoverde, D. F. (marzo de 2018). *Diseño de un sistema de control de inventario y*

organización de las bodegas de producto terminado de la Empresa ECUAESPUMAS-

LAMITEX S.A. Obtenido de <http://dspace.ups.edu.ec/handle/123456789/15717>

Rodriguez Espinoza, J. R. (noviembre de 2016). *Diseño y Mejoramiento del Sistema de Inventario*

Informático de la Empresa MAGREB S.A. Obtenido de

[http://repositorio.ug.edu.ec/bitstream/redug/19851/1/TESIS%20DE%20CPA%20JOSE%](http://repositorio.ug.edu.ec/bitstream/redug/19851/1/TESIS%20DE%20CPA%20JOSE%20RODRIGUEZ%20ESPINOZA%202016.pdf)

[20RODRIGUEZ%20ESPINOZA%202016.pdf](http://repositorio.ug.edu.ec/bitstream/redug/19851/1/TESIS%20DE%20CPA%20JOSE%20RODRIGUEZ%20ESPINOZA%202016.pdf)

Sanchez de la Cruz, J. J., & Villa Pérez, C. I. (2015). *Desarrollo de una aplicación web, para el*

proceso de registro nominal de pacientes hipertensos y diabéticos en la unidad ejecutora

401 Hospital San José de Chincha. Obtenido de

https://alicia.concytec.gob.pe/vufind/Record/AUIC_dfcb7e89cc45edbf77662a72c13a166

Anexos