



**UNIVERSIDAD NACIONAL DE CHIMBORAZO
FACULTAD INGENIERIA
CARRERA DE TECNOLOGIAS DE LA INFORMACION**

Título

**Aplicación web para la gestión de inventarios de la empresa
Servicompu utilizando la metodología Design Thinking**

**Trabajo de Titulación para optar al título de Ingeniero en
Tecnologías de la información**

Autor:

Chimbo Chumaina Juan Carlos

Tutor:

Ing. Miryan Estela Narváez Vilema

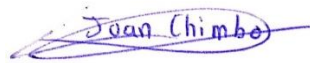
Riobamba, Ecuador. 2024

DECLARATORIA DE AUTORÍA

Yo, Juan Carlos Chimbo Chumaina, con cédula de ciudadanía 0302578398, autor del trabajo de investigación titulado: Aplicación web para la gestión de inventarios de la empresa Servicompu utilizando la metodología Design Thinking, certifico que la producción, ideas, opiniones, criterios, contenidos y conclusiones expuestas son de mí exclusiva responsabilidad.

Asimismo, cedo a la Universidad Nacional de Chimborazo, en forma no exclusiva, los derechos para su uso, comunicación pública, distribución, divulgación y/o reproducción total o parcial, por medio físico o digital; en esta cesión se entiende que el cesionario no podrá obtener beneficios económicos. La posible reclamación de terceros respecto de los derechos de autor de la obra referida, será de mi entera responsabilidad; librando a la Universidad Nacional de Chimborazo de posibles obligaciones.

En Riobamba, a la fecha de su presentación.



Juan Carlos Chimbo Chumaina

C.I: 0302578398

DICTAMEN FAVORABLE DEL PROFESOR TUTOR



Dirección
Académica
VICERRECTORADO ACADÉMICO

en movimiento



UNACH-RGF-01-04-08.11
VERSIÓN 01: 06-09-2021

ACTA FAVORABLE - INFORME FINAL DEL TRABAJO DE INVESTIGACIÓN

En la ciudad de Riobamba, a los 5 días del mes de Febrero de 2024, luego de haber revisado el Informe Final del Trabajo de Investigación presentado por el estudiante **JUAN CARLOS CHIMBO CHUMAINA** con CC: **0302578398**, de la carrera **TECNOLOGIAS DE LA INFORMACION** y dando cumplimiento a los criterios metodológicos exigidos, se emite el **ACTA FAVORABLE DEL INFORME FINAL DEL TRABAJO DE INVESTIGACIÓN** titulado **"APLICACIÓN WEB PARA LA GESTIÓN DE INVENTARIOS DE LA EMPRESA SERVICOMPU UTILIZANDO LA METODOLOGÍA DESIGN THINKING "**, por lo tanto se autoriza la presentación del mismo para los trámites pertinentes.



MIRYAN ESTELA
NARVAEZ VILEMA

PhD. Miryan Estela Narváez Vilema
TUTORA

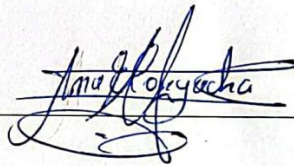
CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

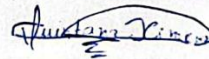
Quienes suscribimos, catedráticos designados Miembros del Tribunal de Grado para la evaluación del trabajo de investigación Aplicación web para la gestión de inventarios de la empresa Servicompu utilizando la metodología Design Thinking, presentado por Juan Carlos Chimbo Chumaina, con cédula de identidad número 0302578398, bajo la tutoría de PhD. Miryan Estela Narváez Vilema; certificamos que recomendamos la APROBACIÓN de este con fines de titulación. Previamente se ha evaluado el trabajo de investigación y escuchada la sustentación por parte de su autor; no teniendo más nada que observar.

De conformidad a la normativa aplicable firmamos, en Riobamba a los 4 días del mes de abril de 2024.

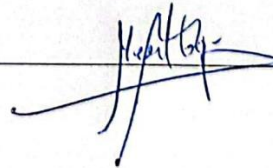
Mgs. Ana Congacha
PRESIDENTE DEL TRIBUNAL DE GRADO



PhD. Ximena Quintana
MIEMBRO DEL TRIBUNAL DE GRADO



Mgs. Milton López
MIEMBRO DEL TRIBUNAL DE GRADO



CERTIFICADO ANTIPLAGIO



Dirección
Académica
VICERRECTORADO ACADÉMICO



UNACH-RGF-01-04-08.17
VERSIÓN 01: 06-09-2021

CERTIFICACIÓN

Que, **CHIMBO CHUMAINA JUAN CARLOS** con CC: **0302578398**, estudiante de la Carrera **INGENIERÍA EN TECNOLOGÍA DE LA INFORMACIÓN**, Facultad de **INGENIERÍA**; ha trabajado bajo mi tutoría el trabajo de investigación titulado "**APLICACIÓN WEB PARA LA GESTIÓN DE INVENTARIOS DE LA EMPRESA SERVICOMPU UTILIZANDO LA METODOLOGÍA DESIGN THINKING**", cumple con el 2 %, de acuerdo al reporte del sistema Anti plagio **TURNITIN**, porcentaje aceptado de acuerdo a la reglamentación institucional, por consiguiente autorizo continuar con el proceso.

Riobamba, 5 de marzo de 2024



codigo_electronicometa.pdf:
MIRYAN ESTELA
NARVAEZ VILEMA

PhD. Miryan Narváz
TUTORA

DEDICATORIA

Esta tesis está dedicada a mis padres, Graciana y Manuel, quienes siempre me han brindado un apoyo incondicional a lo largo de mi vida. También quiero dedicar este trabajo a mis hermanos María, Antonio, Francisco, Delfina, así como a toda mi familia, amigos y seres queridos. A cada uno de ellos les agradezco por impulsarme a perseguir mis sueños y metas.

Juan Carlos Chimbo

AGRADECIMIENTO

El agradecimiento de este trabajo de titulación dedicado a Dios por haberme brindado lo necesario para cumplir esta meta, me gustaría expresar mi profunda gratitud a mi tutora Ing. Estela Narváez por su orientación experta, su apoyo constante y sus valiosas sugerencias a lo largo de todo el proceso de investigación. Su conocimiento y experiencia han sido fundamentales para el desarrollo de este trabajo.

También quiero agradecer a mi familia por su amor incondicional, su apoyo emocional y su comprensión durante esta etapa de mi vida. A mis padres Graciana y Manuel, les agradezco por su constante estímulo para superarme. A mis hermanos María, Antonio, Francisco, Delfina les agradezco por su aliento y por estar siempre a mi lado.

No puedo dejar de mencionar a mis amigos y seres queridos, quienes han sido una fuente de inspiración y motivación en cada paso del camino. Su aliento han sido un impulso constante para mí. Agradezco sinceramente su apoyo y comprensión.

ÍNDICE GENERAL

DECLARATORIA DE AUTORÍA

DICTAMEN FAVORABLE DEL PROFESOR TUTOR

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

CERTIFICADO ANTIPLAGIO

DEDICATORIA

AGRADECIMIENTO

ÍNDICE GENERAL

ÍNDICE DE TABLA

ÍNDICE DE FIGURAS

RESUMEN

CAPÍTULO I. INTRODUCCION..... 15

1 PLANTAMIENTO DEL PROBLEMA 17

1.1 Problema y Justificación 17

1.1.1 Formulación del Problema 17

1.2 Objetivos 18

1.2.1 General 18

1.2.2 Específicos 18

CAPÍTULO II. MARCO TEÓRICO..... 19

2.1 Aplicaciones Web..... 19

2.1.1 Aplicaciones web estáticas..... 19

2.1.2 Aplicaciones web Dinámicas 19

2.2 Framework y bibliotecas para el desarrollo 20

2.2.1 Laravel 20

2.2.2 ReactJS 20

2.3 Herramientas de desarrollo..... 22

2.3.1 Visual Studio Code 22

2.3.2 Xampp..... 22

2.3.3	PhpMyAdmin.....	22
2.3.4	Figma	23
2.4	Metodología Design Thinking.....	24
2.4.1	Metodología Design Thinking vs metodología Scrum	27
2.5	Rendimiento	28
2.5.1	JMeter.....	29
2.6	Arquitectura lógica	29
2.6.1	MVC.....	29
CAPÍTULO III. METODOLOGIA		31
3.1	Metodología.....	31
3.2	Tipo de Investigación	31
3.2.1	Investigación bibliográfica.....	31
3.3	Método y diseño de la investigación	31
3.4	Técnicas de recolección de datos	32
3.5	Población y Muestra	32
3.6	Técnicas de análisis e interpretación de la información.....	32
3.7	Identificación de variables.....	32
3.7.1	Variable dependiente.....	32
3.7.2	Variable independiente	32
3.8	Operacionalización de variables.....	33
Metodología de desarrollo		34
4.1	Levantamiento de requerimientos funcionales y no funcionales	34
4.1.1	Requerimientos funcionales	34
4.1.2	Requerimientos No Funcionales	36
4.2	Diagrama de caso de uso	37
4.3	Estructura de la base de datos.....	39
4.4	Desarrollo GUI para la aplicación web	40

4.4.1 Sprint 1	40
4.5 Desarrollo de la aplicación web	46
4.5.1 Desarrollo de los módulos.....	46
4.5.2 Desarrollo de los módulos Back-end	46
4.5.3 Desarrollo de los Módulos (Reportes, Ventas, Clientes, Categorías)	47
CAPÍTULO IV. RESULTADOS Y DISCUSIÓN	48
5.1 Resultados	48
5.1.1 Análisis de tiempo de respuesta	48
5.1.2 Análisis de uso de recursos	49
5.1.3 Análisis de Resultados	50
5.2 Discusión	50
CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES.....	52
6.1 Conclusión.....	52
6.2 Recomendaciones	53
BIBLIOGRAFÍA	54
ANEXOS.....	57

ÍNDICE DE TABLAS

Tabla 1: Funcionalidades de Laravel.....	20
Tabla 2: Diferencia entre ReactJS y React Native	21
Tabla 3: Metodología Design Thinking vs metodología Scrum.....	27
Tabla 4: Atributos del Rendimiento	28
Tabla 5: Operacionalización de Variables	33
Tabla 6: Requerimiento Funcional 1	34
Tabla 7: Requerimiento funcional 2	34
Tabla 8: Requerimiento funcional 3	34
Tabla 9: Requerimiento funcional 5	35
Tabla 10: Requerimiento funcional 6	35
Tabla 11: Requerimiento funcional 7	35
Tabla 12: Requerimiento funcional 8	35
Tabla 13: Requerimiento funcional 9	35
Tabla 14: Requerimiento funcional 10	36
Tabla 15: Requerimiento funcional 12	36
Tabla 16: Requerimiento no Funcional 1	36
Tabla 17: Requerimiento no Funcional 2	36
Tabla 18: Mapa de empatía	41
Tabla 19: Indicadores de evaluación	48
Tabla 20: Características del dispositivo	48
Tabla 21: Uso de Recursos	49
Tabla 23: Resultados	50
Tabla 24: Brief general	57

ÍNDICE DE FIGURAS

Figura 1: Beneficios de React	21
Figura 2: Panel de control de Xampp	22
Figura 3: Interfaz de trabajo de phpMyAdmin.....	23
Figura 4: Entorno de trabajo de Figma.....	23
Figura 5: Fases de la Metodología Design Thinking	24
Figura 6: Interfaz de JMeter	29
Figura 7: Patrones de arquitectura MVC.....	30
Figura 8: Diagrama General	37
Figura 9: Diagrama Administrador	38
Figura 10: Diagrama Personal	38
Figura 11: Diagrama Cliente	39
Figura 12: Diagrama de Base de Datos - Entidad Relación	39
Figura 13: Brainstorming	42
Figura 14: Módulos de Información, Servicios, Login	43
Figura 15: Módulos de Productos, Categorías, General.....	44
Figura 16: Módulos de Clientes, Personal, Ventas, Proveedores.....	44
Figura 17: Módulos de Servicio Técnico, Servicios, Reportes	45
Figura 18: Desarrollo del Panel Administrativo, Información, Login, Productos .	46
Figura 19: Desarrollo de Proveedores, Servicio Técnico, Servicios y Personal	47
Figura 20: Desarrollo de Reportes, Ventas, Categorías y Clientes	47
Figura 21: Tiempo de respuesta	49
Figura 22: Uso de Recursos.....	50
Figura 23: Pruebas al Usuario Sprint 1	59
Figura 24: JMeter Pruebas Tiempo de respuesta.....	60
Figura 25: JMeter uso de RAM.....	60
Figura 26: JMeter uso de almacenamiento.....	61

RESUMEN

El propósito de esta investigación es analizar la metodología Design Thinking, cuyo objetivo es centrarse en el usuario buscando comprender profundamente las necesidades y perspectivas, con el fin de elaborar un prototipo de aplicación web para mejorar la gestión de inventarios de la empresa Servicompu ubicada en la ciudad de Riobamba.

Para el desarrollo de la aplicación web, se utilizó la metodología Design Thinking, la misma que permite empatizar, definir, idear, crear prototipos y evaluar la lluvia de ideas generadas con el fin de mejorar la calidad de la aplicación y que posea un buen rendimiento. Para el análisis del rendimiento de la aplicación web, se focalizó en tres subcaracterísticas, como comportamiento temporal, tiempo de respuesta y utilización de recursos, memoria RAM y memoria de almacenamiento, proporcionando resultados que permitan mejorar la aplicación web.

Las pruebas de rendimiento a la aplicación web, se efectuó utilizando la herramienta Apache JMeter por medio de simulaciones, se enviaron peticiones en tiempo real, obteniendo así un informe del comportamiento de la aplicación web. Evidenciando un rendimiento satisfactorio con un tiempo de respuesta promedio de 0.508 segundos, el uso de memoria RAM del 39% y el bajo consumo de almacenamiento del 1%, sugieren una gestión eficiente de recursos.

Palabras claves: Design Thinking, Rendimiento, Aplicaciones web, Gestión de Inventarios.

ABSTRACT

Abstract

The purpose of this research is to analyze the Design Thinking methodology, which aims to focus on the user seeking to deeply understand the needs and perspectives to develop a prototype web application to improve the inventory management of the company Servicompu located in Riobamba City. The Design Thinking methodology was used to develop the web application, which allows empathizing, defining, devising, creating prototypes, and evaluating the brainstorming generated to improve the application's quality and performance. For the analysis of the web application's performance, we focused on three sub-characteristics, such as temporal behavior, response time and resource utilization, RAM memory, and storage memory, providing results that allow for web application improvement. The performance tests of the web application were carried out using the Apache JMeter tool by means of simulations, sending requests in real-time, thus obtaining a report of the behavior of the web application. Evidencing a satisfactory performance with an average response time of 0.508 seconds, RAM memory usage of 39%, and low storage consumption of 1%, suggesting efficient resource management.

Keywords: Design Thinking, Performance, Web applications, Inventory Management.



Reviewed by:
Lic. Jenny Freire Rivera
ENGLISH PROFESSOR
C.C. 0604235036

CAPÍTULO I. INTRODUCCION

Las aplicaciones web a través de diversos contextos, como comercio electrónico, productividad, colaboración y educación, son fundamentales en la vida contemporánea, han cambiado la forma de interacción en línea. El progreso de la tecnología ha posibilitado una mayor personalización e interactividad, dando como resultado tecnologías más sofisticadas y utilizadas. Esto es crucial para las empresas y organizaciones porque les permite automatizar procesos comerciales como facturación, servicio al cliente, control de inventario y ventas (Ramos, 2018).

La metodología Design Thinking, está enfocada en el usuario e involucra a todas las partes interesadas en el proceso de diseño. La finalidad es comprender a fondo los requisitos específicos de los usuarios y crear una interfaz de usuario intuitiva y fácil de usar, que permita una gestión eficiente del inventario. Este método ayuda a encontrar oportunidades para mejorar la aplicación y transforma su solución en un producto final funcional y confiable. En el desarrollo, se aplican y emplean las tecnologías más sofisticadas para asegurar la escalabilidad, seguridad y compatibilidad de las soluciones propuestas.

Design Thinking es un conjunto de enfoques para la resolución de problemas que tiene como objetivo reducir los riesgos y al mismo tiempo aumentar las posibilidades de éxito pensando como un diseñador, es decir, pensando en cómo desarrollar una propuesta de solución. Separados en 5 fases: empatizar, idear, definir, prototipar y evaluar. Proceso de innovación colaborativa en un entorno seguro que fomenta la creatividad y la exploración de opciones alternativas, el resultado final es un producto o servicio único y de alto valor (Pérez, 2020).

Establecida por la Apache Software Foundation, JMeter es un instrumento de código abierto que se diseñó originalmente para examinar la función de aplicaciones web, convirtiéndose en una herramienta versátil que se utiliza principalmente para pruebas de rendimiento y carga. Diseñado para simular cargas de usuarios, enviar solicitudes HTTP y medir cómo responde la aplicación a diferentes niveles de carga, se usa ampliamente en el desarrollo de aplicaciones web para garantizar un rendimiento óptimo e identificar posibles problemas de escalabilidad, con el fin de ser corregidos.

Servicompu gestiona su línea de productos de forma manual, generando dificultades para la gestión de la mercadería. Para abordar el problema, se sugiere desarrollar una aplicación

web que permita la automatización eficiente de procesos. El desarrollo se realizará utilizando los frameworks Laravel y librerías de React, conocidos por su alta escalabilidad, así como MySQL como administrador de datos, para asegurar la eficacia del proyecto, se implementará la metodología Design Thinking, posibilita la participación del usuario en todas las fases del proceso de desarrollo y asegura que el producto final cumpla con las expectativas y necesidades. Finalmente, se utilizó la herramienta Apache JMeter para evaluar el rendimiento.

1 PLANTAMIENTO DEL PROBLEMA

1.1 Problema y Justificación

La acogida y rápida adopción de Internet ha llevado a que las aplicaciones web sean indispensables en la vida cotidiana, en la actualidad las personas utilizan estas aplicaciones para llevar a cabo una variedad de actividades, como realizar compras, ventas, pagos, trabajar, solicitar servicios y comunicarse. Este cambio ha provocado una transformación significativa en las sociedades de todo el mundo, generando un cambio en las formas tradicionales de obtener ingresos.

La necesidad de llegar a una mayor cantidad de clientes para aumentar las ventas, las pequeñas y medianas empresas (PYMEs) se han visto obligadas en una transformación digital en la actualidad. Estas buscan automatizar procesos como la gestión de inventario, ventas y facturación con el fin de mejorar. La empresa Servicompu de la ciudad de Riobamba, dedicada a la importación, distribución y venta de repuestos para computadoras y laptops. Actualmente la empresa gestiona su inventario de forma manual, generando errores en el control de productos disponibles, discrepancias en el saldo de la caja y pérdida de información. Es fundamental contar con un sistema de control de inventario preciso para revisar periódicamente y realizar una reposición a tiempo en función de la demanda, para garantizar la disponibilidad constante de los productos que los clientes requieren.

Como se mencionó anteriormente, el objetivo de esta investigación es implementar una solución tecnológica (aplicación web) utilizando la metodología Design Thinking, se enfoca principalmente en comprender las necesidades y deseos de los usuarios para producir ideas y soluciones que satisfagan dichas necesidades. Esto agilizará el proceso de gestión de inventario al automatizar tareas que actualmente se realizan a mano.

1.1.1 Formulación del Problema

¿En qué medida la metodología Design Thinking influirá en el rendimiento de la aplicación web para la gestión de inventarios de la empresa Servicompu?

1.2 Objetivos

1.2.1 General

Desarrollar una aplicación web para la gestión de inventarios de la empresa Servicompu, utilizando la metodología Design Thinking.

1.2.2 Específicos

- Investigar la metodología Design Thinking para el desarrollo de aplicaciones web.
- Crear una Aplicación web para la gestión de inventarios de la empresa Servicompu.
- Evaluar el rendimiento de la aplicación web, utilizando la herramienta JMeter.

CAPÍTULO II. MARCO TEÓRICO

2.1 Aplicaciones Web

Es un tipo de software al que se puede acceder a través de un navegador web y se ejecuta en un servidor remoto. Aplicaciones, ésta no requiere instalación en el dispositivo del usuario. Todas las funciones y datos se almacenan en el servidor, brinda a los usuarios flexibilidad al permitirles acceder a aplicaciones web desde una variedad de dispositivos y sistemas operativos (Paucar, 2022).

Pueden proporcionar una amplia gama de funcionalidades, desde sencillos sitios web interactivos hasta complejas aplicaciones comerciales. Plataformas de comercio electrónico, aplicaciones de productividad, herramientas de colaboración en línea y sistemas de gestión de contenidos son algunos ejemplos comunes. la infraestructura de la nube para el procesamiento y archivo de datos, que ofrece mayor flexibilidad y escalabilidad.

2.1.1 Aplicaciones web estáticas

Muestra una cantidad limitada de información y está diseñada para no añadir nuevos contenidos con frecuencia. Generalmente, estas aplicaciones se desarrollan con HTML y CSS y pueden incluir elementos multimedia como videos, banners y GIFS, actualizar el contenido puede ser complicado, requiere descargar el código, modificarlo y volver a subirlo, no es práctico. Este tipo de aplicación es adecuado para proyectos que no necesitan actualizar la información con frecuencia o para aquellos que desean dar sus primeros pasos en el mundo digital (Zambrano, 2019).

2.1.2 Aplicaciones web Dinámicas

Mucho más compleja que las estáticas. Se basan en el uso de bases de datos para cargar información y los contenidos se actualizan en tiempo real cada vez que el usuario accede a la aplicación. Suelen contar con un panel de administración o CMS desde el cual se gestionan y publican los contenidos como noticias, publicaciones, imágenes, videos, etc. Existen diversos lenguajes de programación para aplicaciones web dinámicas como PHP y ASP, que permiten una buena estructuración del contenido, la actualización de contenidos es muy sencilla y se pueden agregar foros o bases de datos. También es posible cambiar y retocar el diseño de la aplicación web (Lazo, 2022).

2.2 Framework y bibliotecas para el desarrollo

2.2.1 Laravel

Es un framework de PHP que facilita el desarrollo de aplicaciones. Proporciona un sistema de paquetes y sigue el patrón de diseño MVC. Permite a los desarrolladores concentrarse en el desarrollo de la aplicación en lugar de preocuparse por aspectos más técnicos como la instanciación de clases y métodos repetitivos en todo el código. Esto resulta en una mayor eficiencia y un menor gasto de tiempo en la creación y modificación del código de la aplicación (Vera, 2021).

Tabla 1: Funcionalidades de Laravel

Nombre	Función
Blade	Proporciona un sistema de plantillas predefinidas y la posibilidad de incluir secciones de otras vistas. Esto significa que puedes acceder a plantillas ya creadas y utilizar variables con la opción de incorporar código PHP en ellas. Esta funcionalidad te permite estructurar y reutilizar de manera eficiente el código de tu aplicación, facilitando el diseño y la gestión de la interfaz de usuario.
Middlewares	Son controladores que agregan funcionalidad adicional antes y después del procesamiento de una solicitud en el servidor.
Routing	Con esta función, puedes configurar de manera sencilla las rutas web y las rutas de API que serán visitadas por los usuarios de tu aplicación.

Fuente: (Vera, 2021)

Se utiliza principalmente en el desarrollo backend, también es una opción popular para el desarrollo Full Stack en PHP. Con Laravel, los programadores pueden implementar funciones como el inicio de sesión de usuarios y otras capas de datos que no son directamente visibles para los usuarios finales. Las características avanzadas y amigables permiten a los desarrolladores crear aplicaciones web robustas y escalables de manera eficiente.

2.2.2 ReactJS

Es una biblioteca de código abierto de JavaScript, que se utiliza para desarrollar interfaces de usuario (UI). Fue creado por Facebook y su objetivo es simplificar y acelerar la creación de aplicaciones web complejas y escalables, se basa en un enfoque declarativo en el que los desarrolladores describen cómo debería ser la interfaz de usuario, y React se encarga de

actualizar la interfaz en función de los cambios de estado, de esta manera, los desarrolladores pueden centrarse en la lógica de la aplicación y no en la manipulación directa del Document Object Model (DOM) (Rodrigues, 2022).

Tabla 2: Diferencia entre ReactJS y React Native

ReactJS	React Native
Una librería de JavaScript.	Un framework de JavaScript.
Ideal para construir aplicaciones web dinámicas.	Da una sensación nativa a la interfaz de usuario de las aplicaciones móviles.
Utiliza el DOM virtual para renderizar el código del navegador.	Utiliza las APIs nativas para renderizar el código en los dispositivos móviles.
Soporta CSS para crear animaciones.	Requiere la API animada para animar los componentes.
Utiliza etiquetas HTML.	No utiliza etiquetas HTML debido a la falta de soporte de DOM.
Utiliza CSS para el estilo.	Utiliza la hoja de estilos JS para el estilo.

Fuente: (Aguirre, 2023)

React ha ganado una gran popularidad en el desarrollo web gracias a su enfoque en el modularidad, la reutilización de código y el rendimiento, se logra mediante la construcción de componentes independientes y reutilizables, lo que facilita la creación y el mantenimiento de aplicaciones. Utiliza un algoritmo de reconciliación eficiente que minimiza los cambios en el DOM, lo que resulta en una actualización más rápida y suave de la interfaz de usuario.

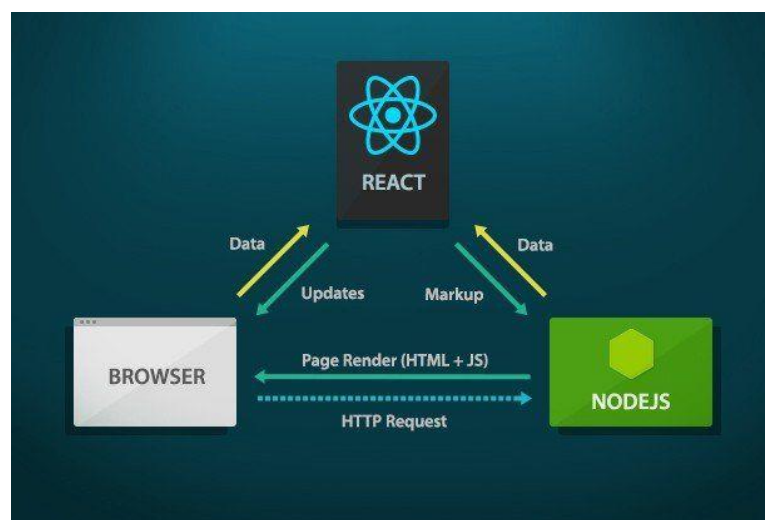


Figura 1: Beneficios de React

Fuente: <https://ghost.codersera.com/blog/starting-your-react-app-and-benefits/>

2.3 Herramientas de desarrollo

2.3.1 Visual Studio Code

Es un popular editor de código fuente desarrollado por Microsoft, ampliamente utilizado en el desarrollo de software. Destaca por su interfaz intuitiva y altamente personalizable, permite a los usuarios adaptar su entorno de desarrollo a sus necesidades. Además, se destaca por su capacidad de extensión mediante la instalación de complementos, que brindan funciones adicionales y soporte para diferentes lenguajes y tecnologías (Velasco, 2021).

2.3.2 Xampp

Es un paquete de software de código abierto que facilita la creación de un entorno de desarrollo web local, incluye componentes esenciales como Apache, MySQL, PHP y Perl, que permiten a los desarrolladores crear y probar aplicaciones web en su propia computadora sin necesidad de una conexión a Internet o un servidor. Compatible con múltiples sistemas operativos y proporciona una interfaz de usuario intuitiva para administrar los servicios web y las bases de datos, simplifica la configuración de un entorno de desarrollo local, lo que facilita el desarrollo y la prueba de aplicaciones web (Granizo, 2022).

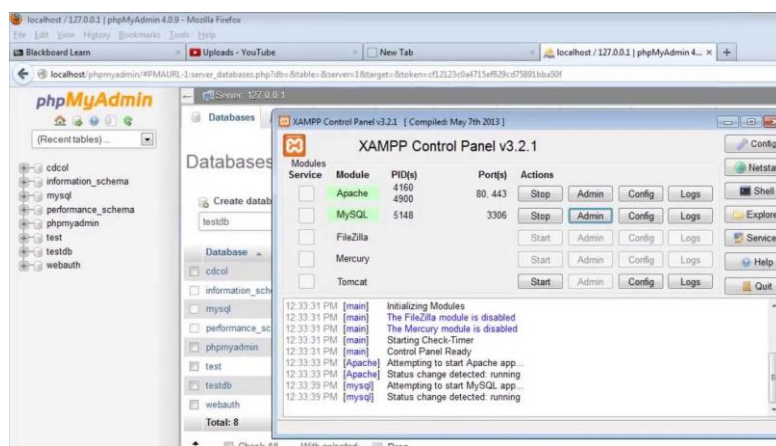


Figura 2: Panel de control de Xampp

Fuente: <https://vsetu.weebly.com/blog/using-xampp>

2.3.3 PhpMyAdmin

Herramienta de código abierto, facilita la gestión y la administración de bases de datos MySQL, posee una interfaz gráfica fácil de entender, permite a los usuarios realizar tareas como crear, modificar y eliminar tablas, ejecutar consultas SQL, administrar usuarios y realizar copias de seguridad, la mayoría de las personas lo utilizan cuando desarrollan y

administran sitios web y aplicaciones, facilita el acceso y la manipulación de datos y proporciona opciones avanzadas para la configuración y optimización de la base de datos. Su comunidad de desarrolladores activa brinda actualizaciones y soporte constante (Zúñiga, 2021).

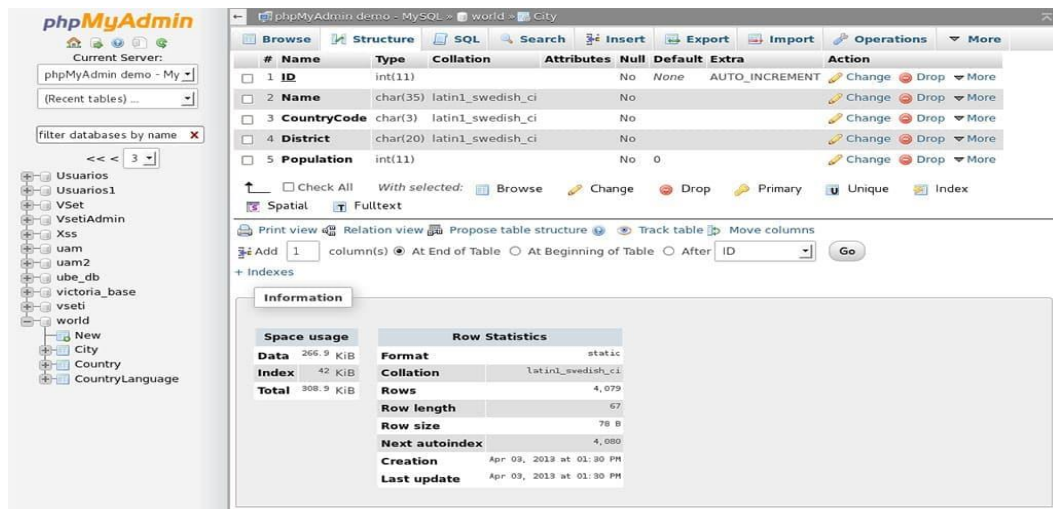


Figura 3: Interfaz de trabajo de phpMyAdmin

Fuente: (Zúñiga, 2021)

2.3.4 Figma

Es una herramienta en línea que ayuda a los diseñadores a crear interfaces, wireframes y prototipos interactivos. Permite a los usuarios colaborar en tiempo real, editar gráficos vectoriales, utilizar componentes reutilizables y crear prototipos, ofrece funciones de control de versiones, facilita la entrega de diseños a los desarrolladores y se integra con otras herramientas mediante complementos (Blandino, 2023).

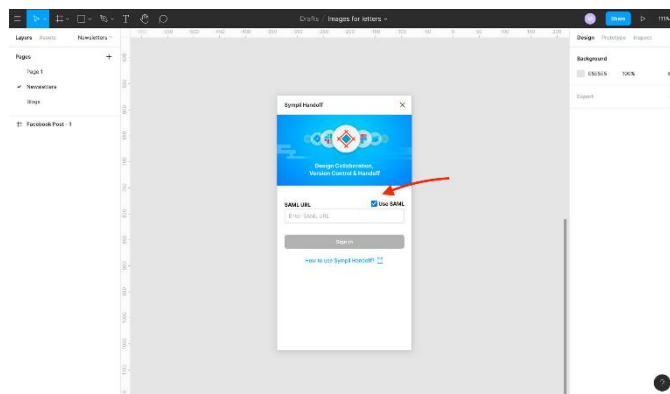


Figura 4: Entorno de trabajo de Figma

Fuente: <https://symply.io/blog/how-to-use-symply-handoff-with-figma-in-2021>

Una de las características apreciadas de Figma es su naturaleza basada en la web, significa que se puede acceder a la herramienta directamente a través del navegador sin necesidad de instalar o actualizar programas, ni adquirir licencias. Esta funcionalidad permite que todos los miembros del equipo puedan trabajar desde cualquier sistema operativo sin preocuparse por las fuentes instaladas o tener que cambiar de computadora, proporciona una mayor flexibilidad y facilidad de uso, eliminando las barreras asociadas con la instalación de software y las limitaciones de compatibilidad.

2.4 Metodología Design Thinking

Surgió en IDEO, una empresa de diseño fundada por David Kelley, Bill Moggridge y Mike Nuttall en 1991, originalmente con el objetivo de diseñar productos para el consumidor, ha expandido su enfoque a la creación de experiencias para el consumidor, trabajando con un enfoque centrado en el ser humano en respuesta a diversos proyectos encargados a la empresa (Fisas, 2019), el proceso de design thinking se organiza en un sistema de tres espacios que se superponen entre sí, en lugar de seguir una secuencia lineal de pasos. Estos tres espacios no se ejecutan de manera secuencial y se puede volver a ellos varias veces a medida que evolucionan las ideas.

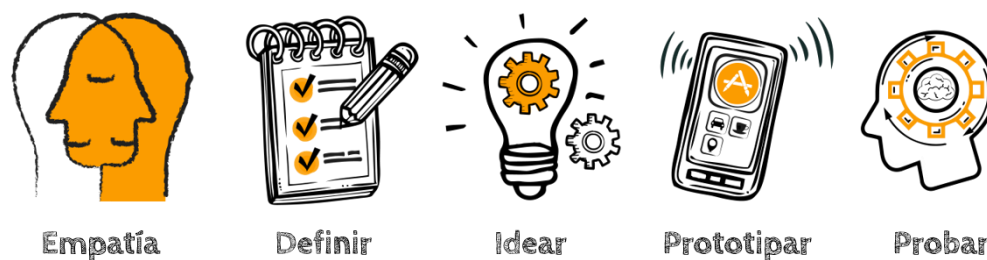


Figura 5: Fases de la Metodología Design Thinking

Fuente: <https://www.ovtt.org/recursos/design-thinking/>

En el espacio de inspiración dentro de la metodología Design Thinking, se busca entender el problema desde múltiples perspectivas y obtener una comprensión profunda de las necesidades y deseos de las personas afectadas por el problema. Por otro lado, en el espacio de ideación, el objetivo es generar una gran cantidad de ideas y soluciones creativas para el problema planteado, en el espacio de implementación, se trabaja para convertir estas ideas y soluciones en acciones concretas y aplicables en la vida real, es importante destacar que estos

tres espacios no se ejecutan de manera secuencial y pueden volver a ser abordados varias veces a medida que se van desarrollando nuevas ideas y soluciones (Marybeth, 2021).

Fase 1 Empatía

Es importante comprender a las personas y a los usuarios en el desarrollo de productos y servicios. Ponerse en los zapatos del usuario final implica adoptar una perspectiva empática, buscando entender al cliente no solo como un cliente, sino como un ser humano con experiencias, necesidades y contexto único. Al tener esta comprensión, se pueden identificar las necesidades reales del usuario y encontrar formas efectivas de satisfacerlas. Esta perspectiva humanizada permite crear soluciones más relevantes y significativas, que se ajusten a las necesidades y expectativas de las personas, a continuación, lo esencial para ser empático recuperado de (Pesantez, 2021).

- Observar: implica mirar a los usuarios y sus comportamientos dentro del contexto de sus vidas. Es importante adoptar una perspectiva externa, sin entrometerse en su experiencia, para poder comprender y obtener información relevante.
- Involucrarse: implica generar una conversación activa con los usuarios. Esto puede ser a través de preguntas informales en un encuentro casual o mediante una conversación más estructurada.
- Es importante hacer la pregunta "¿Por qué?" de manera constante, permite descubrir nuevos significados y comprender más a fondo una situación. Al cuestionar repetidamente con "¿Por qué?", buscando ir más allá de las respuestas superficiales y descubrir las causas subyacentes o los motivos detrás de un comportamiento o una decisión.

Fase 2 Definir

En esta etapa, es crucial establecer de manera clara el problema que se desea resolver y la necesidad que se busca satisfacer. Es importante ser preciso en la definición del problema para enfocar correctamente los esfuerzos y recursos en la búsqueda de una solución efectiva, al comprender claramente el problema, se puede comenzar a explorar diferentes enfoques y aplicar la creatividad para generar ideas innovadoras (Quistan, 2022).

Una vez que se ha definido el problema, es fundamental también definir la solución deseada. Esto implica tener una visión clara de cómo se espera que sea la solución final y cómo se

alineará con los objetivos y necesidades identificadas, definir la solución ayuda a establecer un punto de referencia y una dirección clara para el desarrollo y la implementación de las acciones necesarias.

Fase 3 Ideas

En la fase de ideación, se generan ideas y soluciones para abordar el problema previamente definido y se inicia el proceso de diseño. Esta etapa combina una fase divergente, se fomenta la generación libre de alternativas e ideas, con una fase posterior de convergencia, en la cual se buscan patrones y se reducen el número de opciones (Bravo, 2022).

Fase 4 Prototipar

En la fase de prototipado, se lleva a cabo la ejecución y materialización de las ideas y conceptos generados en la fase de ideación, se busca poner a prueba la visión del proyecto y convertirla en una realidad tangible. Se crean prototipos que representan a pequeña escala la idea de solución, ya sea a través de dibujos, objetos tangibles u otros medios que se acerquen a la representación de la idea.

A lo largo de esta etapa, se realizan iteraciones y pruebas de los prototipos creados. Esto implica evaluar y probar los prototipos para garantizar que funcionan, sean efectivos y cumplan con los requisitos. Estas pruebas generalmente producen nuevas ideas y validaciones, lo que resulta en un ciclo de iteraciones continuas entre la ideación y la validación. El objetivo principal es interactuar con los prototipos y recopilar información útil. Con la ayuda de esta evaluación, se pueden realizar cambios, mejoras y refinamientos en el diseño y la funcionalidad del producto o solución que se ha propuesto. En esta etapa, el ciclo de iteraciones permite perfeccionar y optimizar la idea inicial para lograr los objetivos y satisfacer las necesidades (Karina, 2020).

Fase 5 Evaluar

Durante la fase de evaluación y retroalimentación, se lleva a cabo un proceso clave para el desarrollo del proyecto o idea. En esta etapa, se busca identificar mejoras significativas, fallos y posibles carencias y deficiencias en el prototipo desarrollado, la retroalimentación de los usuarios desempeña un papel fundamental, proporciona información valiosa sobre la adecuación de la solución propuesta. Mediante la interacción con el prototipo, se recopila feedback directo de los usuarios, permite evaluar la efectividad y usabilidad de la solución.

Esta retroalimentación ayuda a identificar áreas de mejora, detectar posibles problemas y validar si el prototipo satisface las necesidades y expectativas de los usuarios (Eras, 2021).

Durante esta fase se establece una conexión emocional con los usuarios al tener en cuenta sus necesidades y diseñar el prototipo pensando en ellos. Esta empatía contribuye a que la idea evolucione y se ajuste de manera efectiva hasta convertirse en la solución esperada.

2.4.1 Metodología Design Thinking vs metodología Scrum

Aunque SCRUM y Design Thinking son dos enfoques diferentes, comparten algunas similitudes, se centran en la colaboración y en trabajar en equipo, así como en la iteración continua para mejorar el resultado, Involucran a los usuarios o clientes en el proceso de desarrollo, garantiza que el resultado final satisfaga sus necesidades y expectativas. A pesar de estas similitudes, es importante tener en cuenta que SCRUM se enfoca más en la gestión de proyectos y en la entrega de productos de alta calidad en el menor tiempo posible, mientras que Design Thinking se enfoca en la innovación y en la solución de problemas complejos (Beltrán, 2020).

Tabla 3: Metodología Design Thinking vs metodología Scrum

Metodología	Design Thinking	Scrum
Enfoque	Centrado en el usuario, solución de problemas complejos	Orientado a la gestión de proyectos ágiles
Etapas	Empatizar, definir, idear, prototipar y testear	Planificación, desarrollo, pruebas y entrega
Objetivo	Desarrollar soluciones innovadoras para problemas complejos	Entregar un producto o servicio de manera eficiente
Enfoque de equipo	Colaborativo, multidisciplinario y con roles flexibles	Roles y responsabilidades definidos y enfocados en la entrega del proyecto
Resultados	Soluciones creativas y enfocadas en el usuario	Entrega de un producto o servicio funcional y eficiente
Herramientas y técnicas	Entrevistas con usuarios, mapas de empatía, lluvia de ideas, prototipado, testing	Reuniones diarias, sprint planning, scrum board, retrospectives

Fuente: (Beltrán, 2020)

2.5 Rendimiento

Se relaciona con la eficiencia y la velocidad de un sistema o aplicación en la realización de tareas. Se evalúa a través de indicadores como el tiempo de respuesta, la velocidad de procesamiento, el uso de recursos y la capacidad de escalabilidad, un buen rendimiento implica respuestas rápidas, procesamiento eficiente, uso adecuado de los recursos y capacidad para manejar cargas de trabajo más grandes, mejorar el rendimiento implica optimizar el sistema y sus componentes para lograr mayor eficiencia y productividad (Cougil, 2019).

Tabla 4: Atributos del Rendimiento

Atributo	Concepto
Capacidad de escalabilidad	La capacidad del sistema para manejar cargas de trabajo más grandes y adaptarse a un aumento en la demanda sin perder rendimiento
Estabilidad y fiabilidad	La capacidad del sistema para mantener un rendimiento constante y confiable en diferentes situaciones y condiciones.
Eficiencia energética	La capacidad del sistema para minimizar el consumo de energía mientras mantiene un rendimiento óptimo.

Fuente: (Cougil, 2019).

Mejorar el rendimiento es un objetivo muy buscado en el desarrollo de software y la optimización de sistemas. Implica aplicar diversas estrategias y técnicas para aumentar la eficiencia y velocidad de un sistema o aplicación. Algunas formas comunes de lograr esto incluyen optimizar el código fuente, mejorar los algoritmos utilizados, asignar eficientemente recursos como memoria y procesadores, usar técnicas de almacenamiento en caché, paralelizar tareas y reducir las operaciones de entrada y salida. (Kanjila, 2022).

Estos esfuerzos se pueden realizar en varios niveles, desde el diseño arquitectónico hasta la implementación y el refinamiento del código. El objetivo final es crear un sistema que responda rápidamente a las solicitudes de los usuarios, utilice los recursos de manera eficiente y proporcione una experiencia de usuario fluida y satisfactoria.

2.5.1 JMeter

Es ampliamente reconocido como la herramienta más popular para llevar a cabo pruebas de carga. Inicialmente, se diseñó con el propósito de realizar pruebas de estrés específicamente en aplicaciones web, pero en la actualidad se ha ampliado su alcance para permitir el diseño de pruebas en bases de datos, FTP y prácticamente cualquier otro tipo de sistema.

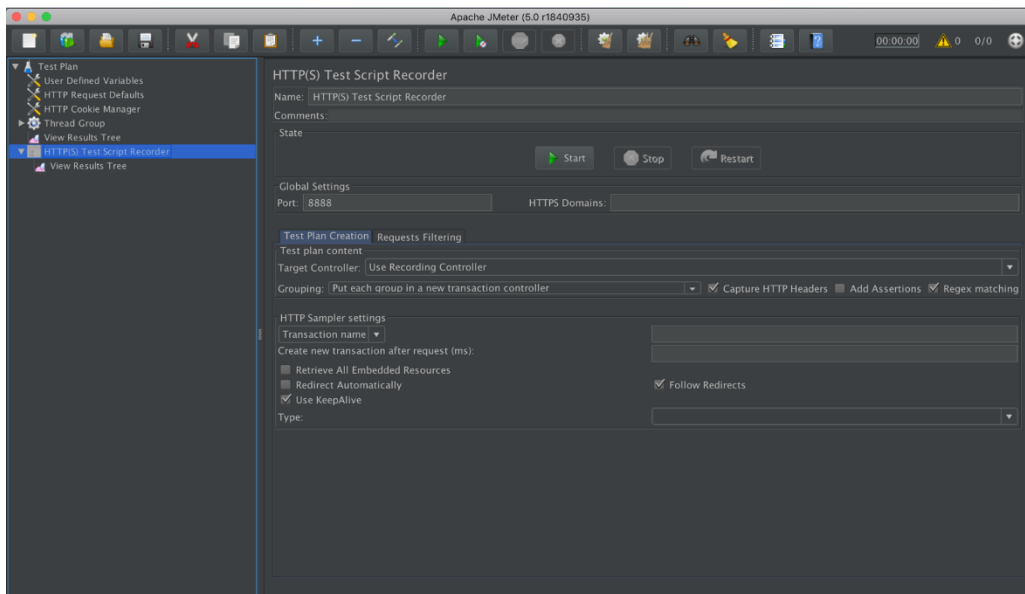


Figura 6:Interfaz de JMeter

Fuente: <https://www.appvizer.co.uk/it/end-to-end-testing/apache-jmeter>

Una de las características principales de JMeter es su versatilidad en cuanto a las pruebas que se pueden realizar, es posible realizar desde componentes simples, como solicitudes individuales, hasta secuencias complejas que permiten analizar el comportamiento de una aplicación en detalle, tiene la capacidad de manipular y almacenar secuencias de pruebas para su reutilización en otros escenarios. Esto facilita la creación de pruebas personalizadas y la optimización del tiempo y los recursos necesarios para realizar pruebas adicionales (Banda, 2022).

2.6 Arquitectura lógica

2.6.1 MVC

Es un patrón de diseño de software que permanece independiente y utiliza tres componentes principales: un modelo, una vista y un controlador. Este método se utiliza ampliamente en componentes gráficos básicos y sistemas comerciales. Además, es un enfoque muy popular

en la mayoría de las arquitecturas estructurales modernas y se utiliza ampliamente en la industria para crear diseños modulares y extensibles (Castillo, 2020).

Patrones de Arquitectura MVC

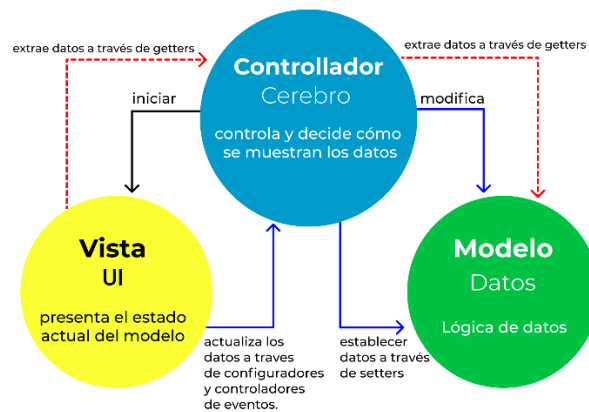


Figura 7: Patrones de arquitectura MVC

Fuente: <https://www.freecodecamp.org/espanol/news/el-modelo-de-arquitectura-view-controller-pattern/>

La arquitectura MVC intenta que cada componente del código tenga un propósito claro y diferenciado y que interactúen de forma que facilite el mantenimiento futuro. El objetivo principal es respaldar modelos funcionales y mapas mentales con información relevante para que los usuarios puedan familiarizarse y manipular fácilmente los datos. Esto se logra separando responsabilidades y estructurando componentes en modelos, vistas y controladores.

CAPÍTULO III. METODOLOGIA

3.1 Metodología

El objetivo de esta investigación es aplicar la metodología Design Thinking para desarrollar una aplicación web de gestión de inventarios para la empresa Servicompu, con el fin de mejorar la eficiencia y efectividad, proporcionando una herramienta útil y adaptada a las necesidades de los usuarios. El estudio se planteó con un enfoque cuantitativo, se evaluó el rendimiento en función de los indicadores definidos.

3.2 Tipo de Investigación

3.2.1 Investigación bibliográfica

Es una etapa fundamental en cualquier estudio o proyecto académico. Consiste en realizar una búsqueda minuciosa y exhaustiva de información en diversas fuentes bibliográficas, como libros, revistas científicas, artículos, informes y otros recursos relevantes, el objetivo principal es recopilar una amplia gama de información relacionada con el tema de estudio, Con el fin de obtener una base sólida y fundamentada para el desarrollo del trabajo.

En esta etapa se emplean diferentes técnicas de búsqueda de información, como la consulta de catálogos de bibliotecas, bases de datos en línea y motores de búsqueda especializados. Se revisan los títulos, resúmenes y contenidos, evaluando su pertinencia y relevancia, esto permite obtener conocimientos previos sobre el tema, identificar las teorías, enfoques y metodologías existentes, conocer las opiniones y hallazgos de otros investigadores en el área, proporcionando una base sólida para el marco teórico del estudio, permitiendo establecer conexiones y fundamentar las afirmaciones o conclusiones que se presentarán posteriormente.

3.3 Método y diseño de la investigación

El estudio se fundamentó en una investigación aplicada donde el objetivo es resolver un problema en específico, desarrollando un prototipo para la aplicación web, con el fin de dar como resultado un proceso optimizado de gestión de inventarios de la empresa.

3.4 Técnicas de recolección de datos

Según el enfoque de investigación planteado, se llevan a cabo simulaciones mediante la herramienta JMeter. Los resultados obtenidos se analizan para evaluar la eficiencia del rendimiento de la aplicación web y probar cómo responde ante las solicitudes de los usuarios.

- **Observación:** es una técnica de recolección de datos utilizada en diferentes áreas. Implica la recopilación de información mediante una observación directa y sistemática de eventos, comportamientos, situaciones o fenómenos en un contexto particular. Esta ayudará de manera inicial para identificar el problema que presenta la empresa.

3.5 Población y Muestra

En función del tipo de investigación planteada, se considera que la población es infinita. Para recopilar datos según los indicadores especificados, se utilizó la herramienta JMeter mediante la cual se simulaban diversas pruebas en base a los indicadores de rendimiento.

3.6 Técnicas de análisis e interpretación de la información

Las mediciones de rendimiento se efectuaron con la ayuda de la aplicación JMeter, esta aplicación proporciona valores numéricos relacionados a la eficacia, tiempo de respuesta y a la vez entrega gráficas de los valores obtenidos.

3.7 Identificación de variables

3.7.1 Variable dependiente

Aplicación web para la gestión de inventarios de la empresa Servicompu.

3.7.2 Variable independiente

Rendimiento de la aplicación web.

3.8 Operacionalización de variables

Tabla 5: Operacionalización de Variables

Pregunta de investigación	Tema	Objetivos	Variables	Conceptualización	Indicadores
¿En qué medida la metodología Desing Thinking influirá en el rendimiento de la aplicación web para la gestión de inventarios de la empresa Servicompu?	Aplicación web para la gestión de inventarios de la empresa Servicompu utilizando la metodología Design Thinking	General	Independiente	Busca simplificar y que permita un seguimiento preciso, eficiente y actualizado de los productos, así como la generación de informes y análisis. (Laoyan, 2022).	<ul style="list-style-type: none"> • Número de requerimientos generados. • Número de módulos generadas.
		Específicos	Dependiente	Implica mejorar el tiempo de carga, la capacidad de respuesta, la eficiencia en el uso de recursos, mediante técnicas de optimización.	Como criterios del rendimiento <ul style="list-style-type: none"> • Consumo de recursos. • Tiempo de respuesta.

Elaborado por: Autor

Metodología de desarrollo

Design Thinking es una metodología centrada en los usuarios y tiene como objetivo solucionar problemas de manera creativa e innovadora, basándose en un enfoque iterativo y colaborativo, combina diferentes disciplinas y perspectivas para generar soluciones efectivas. Se centra en analizar el software directamente con el beneficiario en cada ciclo de vida. En el proyecto actual, se incluyen las siguientes fases: empatizar, definir, idear, prototipar y evaluar.

4.1 Levantamiento de requerimientos funcionales y no funcionales

4.1.1 Requerimientos funcionales

Para el desarrollo de la aplicación web, se enfocó en los siguientes requerimientos.

Tabla 6: Requerimiento Funcional 1

Identificación del Requerimiento	RF001
Nombre del requerimiento	Mostrar información del establecimiento
Característica	Información
Prioridad de requerimiento	Media

Elaborado por: Autor

Tabla 7: Requerimiento funcional 2

Identificación del Requerimiento	RF002
Nombre del requerimiento	Autenticación de los usuarios
Característica	Autenticar usuario
Prioridad de requerimiento	Alta

Elaborado por: Autor

Tabla 8: Requerimiento funcional 3

Identificación del Requerimiento	RF003
Nombre del requerimiento	Gestión de Servicios
Característica	Gestión
Prioridad de requerimiento	Alta

Elaborado por: Autor

Tabla 9: Requerimiento funcional 5

Identificación del Requerimiento	RF005
Nombre del requerimiento	Gestión de usuarios internos
Característica	Gestión
Prioridad de requerimiento	Alta

Elaborado por: Autor

Tabla 10: Requerimiento funcional 6

Identificación del Requerimiento	RF006
Nombre del requerimiento	Gestión de clientes
Característica	Gestión
Prioridad de requerimiento	Media

Elaborado por: Autor

Tabla 11: Requerimiento funcional 7

Identificación del Requerimiento	RF007
Nombre del requerimiento	Gestión de proveedores
Característica	Gestión
Prioridad de requerimiento	Alta

Elaborado por: Autor

Tabla 12: Requerimiento funcional 8

Identificación del Requerimiento	RF008
Nombre del requerimiento	Gestión de categorías
Característica	Gestión
Prioridad de requerimiento	Media

Elaborado por: Autor

Tabla 13: Requerimiento funcional 9

Identificación del Requerimiento	RF009
Nombre del requerimiento	Gestión de Productos
Característica	Gestión
Prioridad de requerimiento	Alta

Elaborado por: Autor

Tabla 14: Requerimiento funcional 10

Identificación del Requerimiento	RF010
Nombre del requerimiento	Gestión de ventas
Característica	Gestión
Prioridad de requerimiento	Alta

Elaborado por: Autor

Tabla 15: Requerimiento funcional 12

Identificación del Requerimiento	RF012
Nombre del requerimiento	Gestión del servicio técnico
Característica	Administración
Prioridad de requerimiento	Alta

Elaborado por: Autor

4.1.2 Requerimientos No Funcionales

Los requisitos no funcionales son las características y condiciones que describen cómo debe funcionar en general el aplicativo web.

Tabla 16: Requerimiento no Funcional 1

Identificación del Requerimiento	RNF001
Nombre del requerimiento	Seguridad
Característica	Protección de acceso
Descripción del requerimiento	La aplicación web debe garantizar a los usuarios la seguridad con la autenticación para su uso.
Prioridad de requerimiento	Media

Elaborado por: Autor

Tabla 17: Requerimiento no Funcional 2

Identificación del Requerimiento	RNF002
Nombre del requerimiento	Eficiencia en el uso de recursos
Característica	La aplicación debe utilizar los recursos del sistema de manera eficiente.
Descripción del requerimiento	Debe consumir los recursos de manera eficiente como la memoria, el ancho de banda y el espacio de almacenamiento, para garantizar un buen rendimiento.
Prioridad de requerimiento	Alto

Elaborado por: Autor

4.2 Diagrama de caso de uso

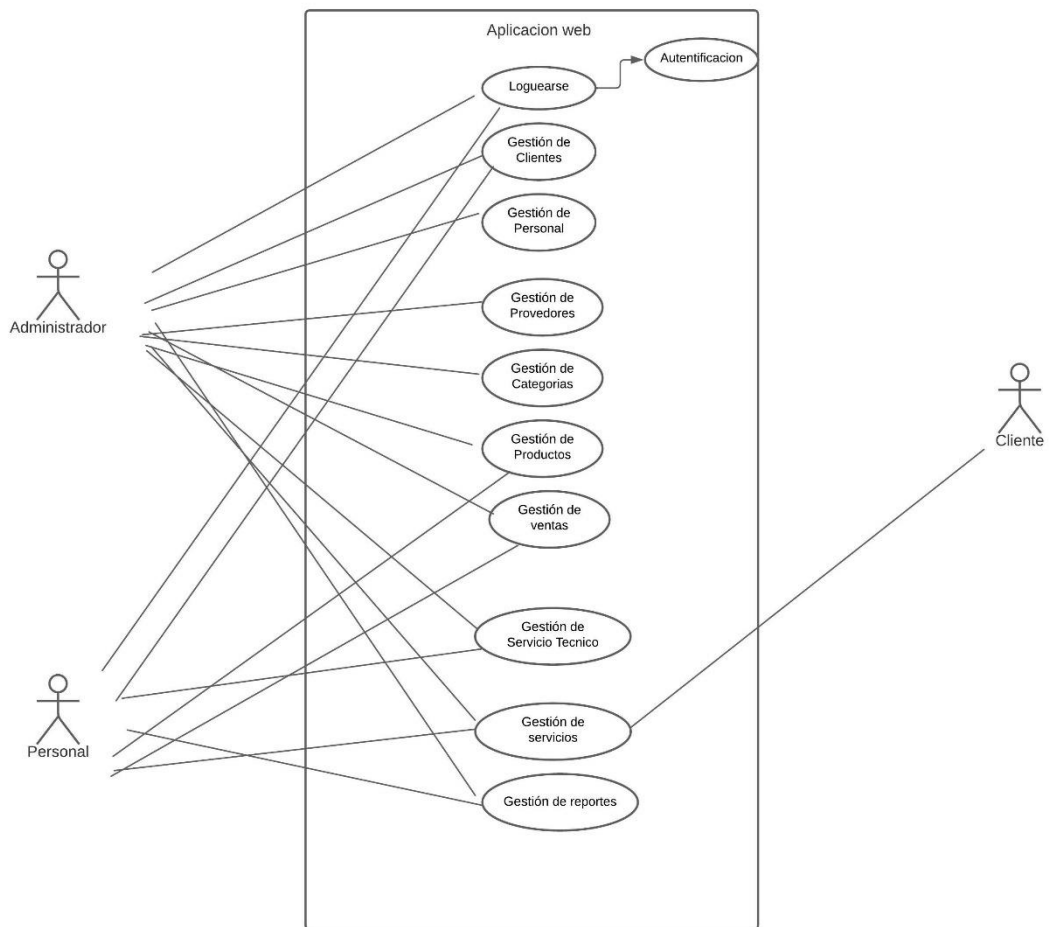


Figura 8: Diagrama General

Elaborado por: Autor

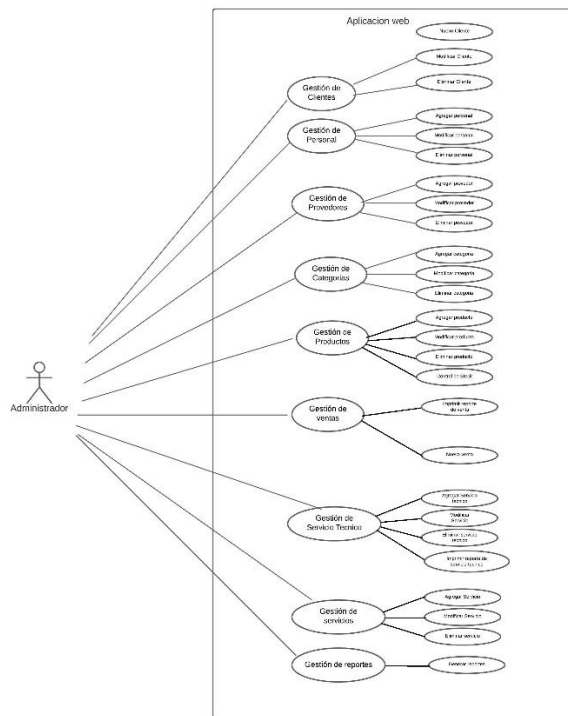


Figura 9: Diagrama Administrador

Elaborado por: Autor

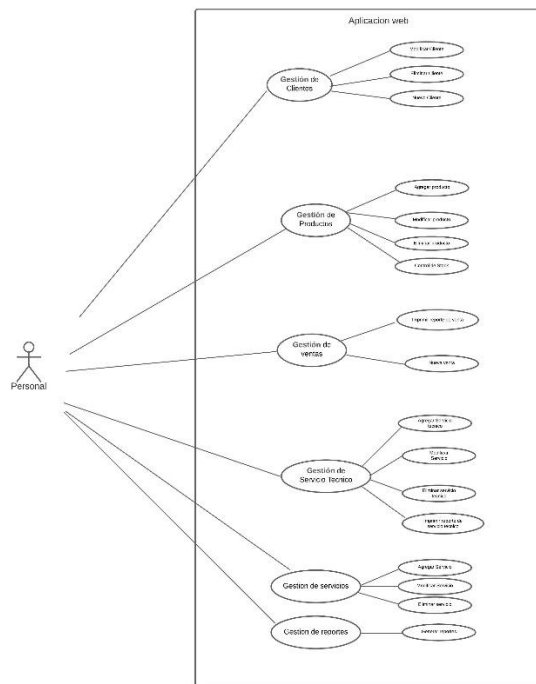


Figura 10: Diagrama Personal

Elaborado por: Autor

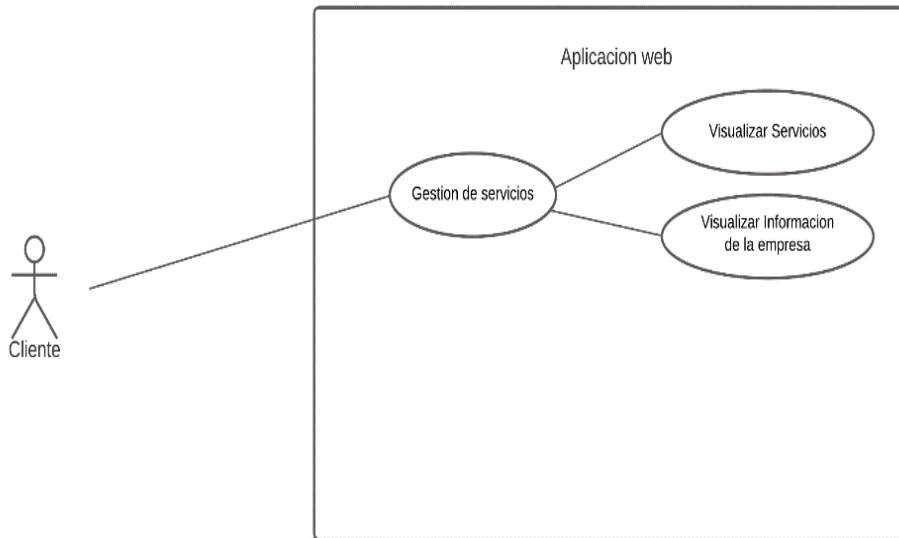


Figura 11: Diagrama Cliente

Elaborado por: Autor

4.3 Estructura de la base de datos

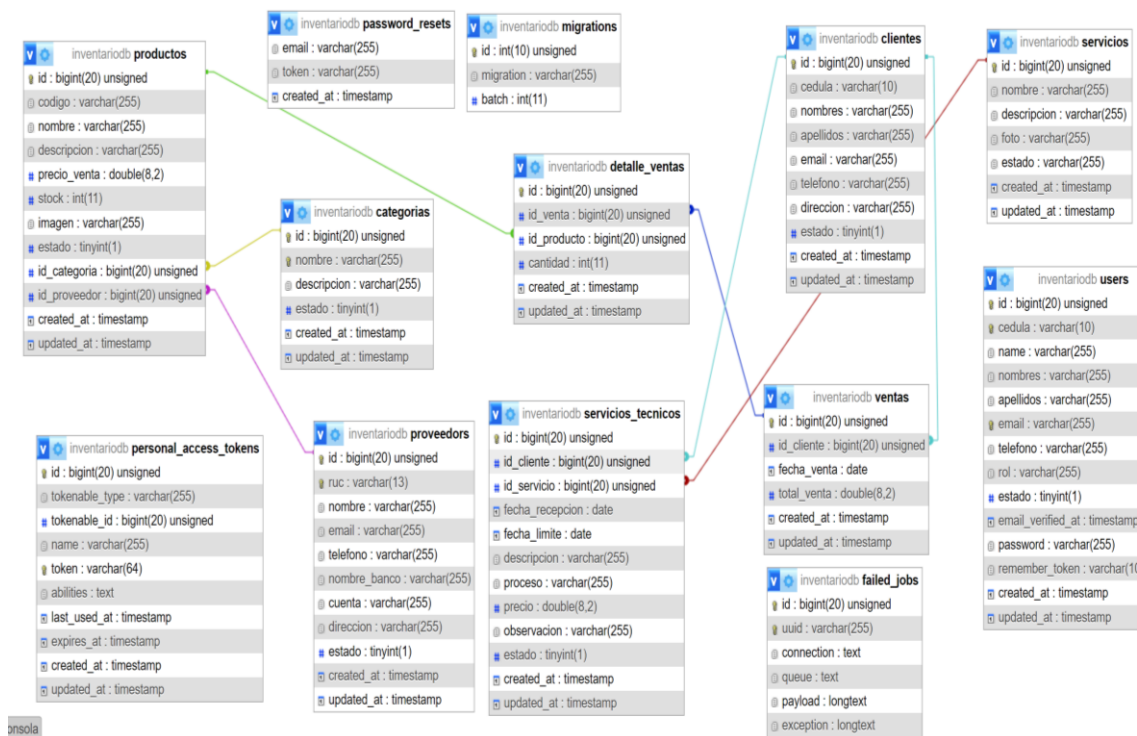


Figura 12: Diagrama de Base de Datos - Entidad Relación

Elaborado por: Autor

4.4 Desarrollo GUI para la aplicación web

Para construir las interfaces gráficas de la aplicación web se utilizó la herramienta figma, permite crear prototipos, facilitando el diseño de cada interfaz. En el desarrollo front end se empleó Laravel y React, proporciona una plantilla con una arquitectura MVC y una biblioteca de JavaScript respectivamente. Para la generación de la aplicación web se realizó un sprint, englobado en la metodología Design Thinking.

4.4.1 Sprint 1

Fase 1: Empatizar

En esta fase se busca recopilar información y lograr una comprensión más profunda del contexto en el que existe el problema y las características de los individuos. Esto implica realizar investigaciones, entrevistas y observaciones para recopilar información relevante.

Se utiliza un plan de trabajo para definir el contenido de las tareas a realizar, permite organizar los objetivos y desarrollar una estrategia de solución adecuada. Este esquema se incluye en un Brief o resumen creativo (ver Anexo 1) como guía para el proceso de diseño. Obtener y organizar información sobre el estado actual de la empresa.

Estas reuniones son muy importantes para comprender el contexto de la empresa, conocer su funcionamiento interno y comprender sus objetivos. La información recopilada durante estas reuniones permite orientar de manera precisa y adecuada el desarrollo de soluciones, para satisfacer las necesidades y objetivos de la empresa.

Para cubrir estas necesidades se implementó la aplicación web con tecnología Laravel, React y MySql como gestor de base de datos. Esta elección se realizó con el objetivo de lograr una visualización atractiva de la información de la empresa y proporcionar una experiencia moderna y fácil de usar (ver anexo 1).

Fase 2: Definir

En esta etapa del proceso, el objetivo es desarrollar una comprensión detallada de los usuarios y del diseño del producto. Esto incluye crear un mapa de empatía que le ayude a comprender sus necesidades, deseos, motivaciones y frustraciones.

Mapa de empatía

Se llevó a cabo una sesión de lluvia de ideas con los usuarios de la empresa Servicompu en un esfuerzo por identificar los inconvenientes de negocio centrado en el usuario. El objetivo es comprender mejor el entorno, comportamiento, preocupaciones y preferencias. Esta herramienta puede recopilar información relevante para comprender mejor a los usuarios. Durante este encuentro se generó un mapa de empatía que reflejaba pensamientos, sentimientos, necesidades y desafíos. Esto proporciona una comprensión más profunda y detallada de los clientes potenciales, permite comprender sus motivaciones y diseñar soluciones adaptadas a sus necesidades.

Tabla 18: Mapa de empatía

<p>¿Qué ve?</p> <ul style="list-style-type: none"> -Falta dinero al cierre de caja -A veces falta productos por que no se tienen el stock actualizado -Existen muchos libros para la gestión de inventario -Hay muchos papeles en la oficina -Pérdida de información. 	<p>¿Qué oye?</p> <ul style="list-style-type: none"> -Quejas -Existen quejas de los clientes habituales. -El propietario del negocio preocupado por las ventas. -El gerente se molesta al no contar con información actualizada. -Dificultad para analizar y planificar -Tiempo y esfuerzo requerido
<p>Personal</p>	
<p>¿Qué piensa y siente?</p> <ul style="list-style-type: none"> -La falta de control de stock de productos. -No se cuenta con un control preciso de ventas -Los inventarios no llegan a cuadrar. -No cuenta con información en tiempo real. -No cuenta con información de productos que deben ser reabastecidos. -No cuenta con registro de ventas -Pérdida de tiempo buscando de manera manual. 	<p>¿Qué dice y hace?</p> <ul style="list-style-type: none"> -Dan apoyo en la actualización de inventarios. -Solicitar reportes con tiempo -Dar seguimientos al cliente en la adquisición de su compra. -Realizo cuadros de caja. -Hago una actualización semanal del stock de los productos.
<p>Esfuerzos</p> <ul style="list-style-type: none"> -No me gusta no poder contar con un reporte de productos en tiempo real. 	<p>Resultados</p> <ul style="list-style-type: none"> -Facilidad para obtener información -Usuarios contentos -Gestión de inventarios -Una nueva experiencia

Elaborado por: Autor.

Fase 3: Idear

El objetivo principal de esta fase es generar múltiples ideas y evitar que la empresa se liquide. Se adopta el método de Brainstorming para facilitar la generación de un gran número de ideas en un entorno colaborativo y tratar de estimular el entusiasmo del equipo. Creatividad y pensamiento divergente, se anima a todos los participantes a compartir libremente sus

ideas, sin importar cuán locas o poco convencionales puedan parecer. Céntrese en la cantidad y variedad de ideas generadas y evitando cualquier forma de juicio o crítica.

Brainstorming

Esta actividad fomenta la generación de varias ideas con el objetivo de ofrecer varias soluciones alternativas, la meta es estimular la creatividad y crear amplias posibilidades que puedan resolver el problema o necesidad identificada. Las ideas proporcionan diferentes enfoques y perspectivas, permitiendo la exploración de soluciones innovadoras adaptadas a las necesidades específicas de los usuarios. Al brindar diversas recomendaciones se facilita el proceso de evaluación y selección de la mejor solución a desarrollar en las siguientes fases.

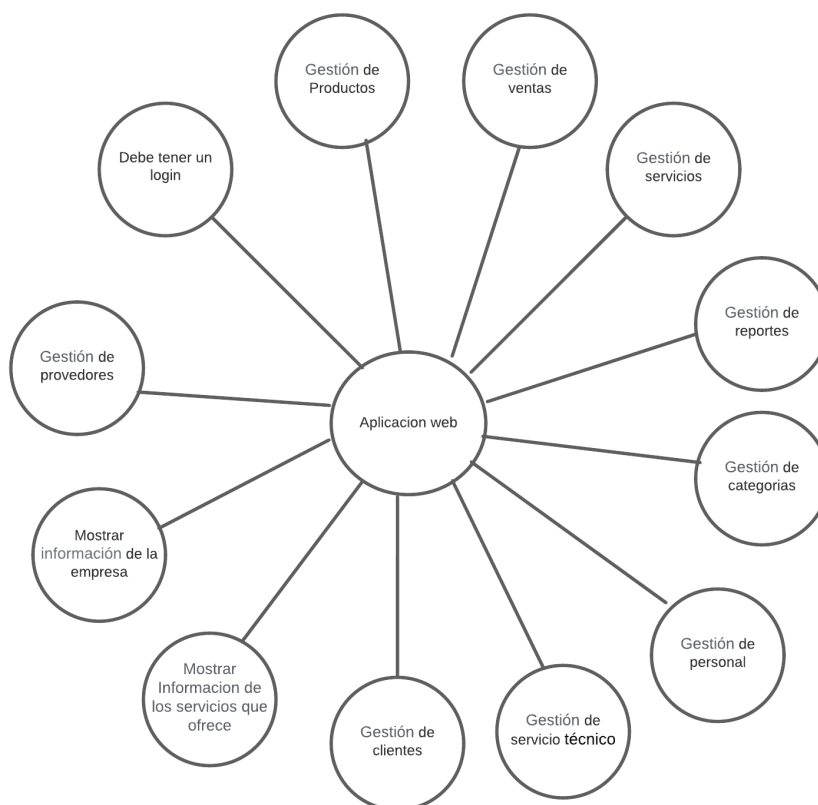


Figura 13: Brainstorming

Elaborado por: Autor

Fase 4: Prototipar

En esta etapa, todas las ideas se implementaron mediante la creación de prototipos. Esto significa una variedad de propuestas, que requieren una mejora continua hasta llegar al producto final.

Prototipos diseñados en Figma

A partir de las ideas creadas previamente se inicia el desarrollo de la interfaz o interfaces del módulo de aplicación web. Este proceso implica crear y diseñar las partes visuales e interactivas de la aplicación, centrándose en el aspecto y la utilidad de cada módulo.

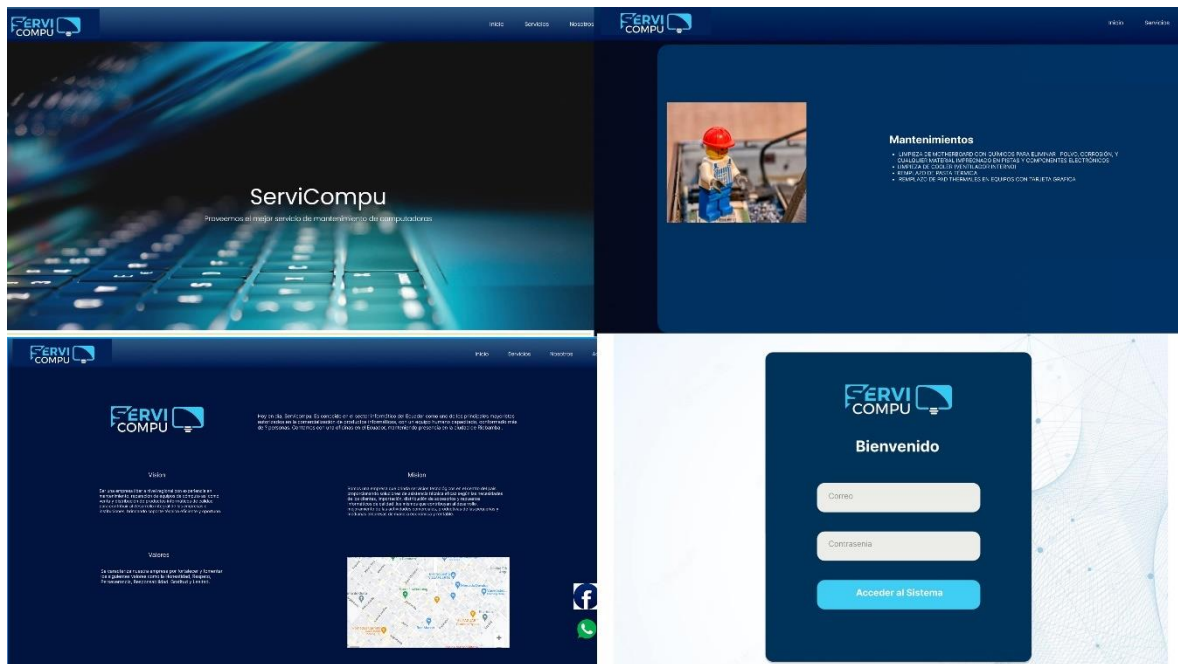


Figura 14: Módulos de Información, Servicios, Login

Elaborado por: Autor

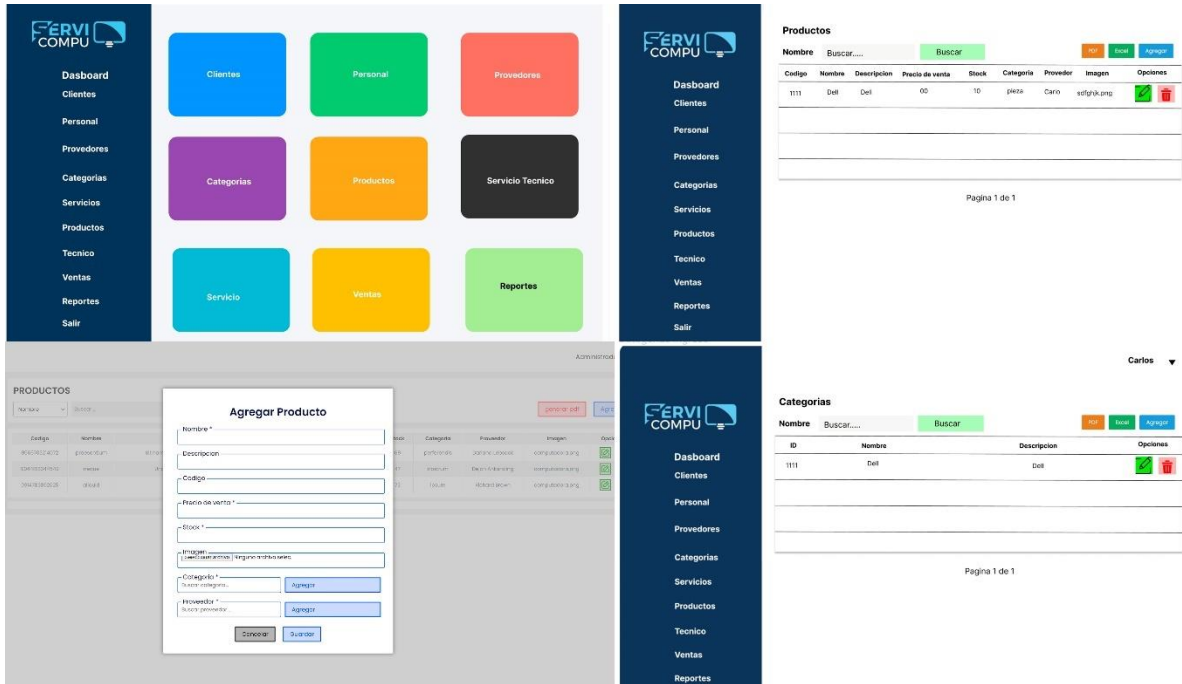


Figura 15: Módulos de Productos, Categorías, General

Elaborado por: Autor

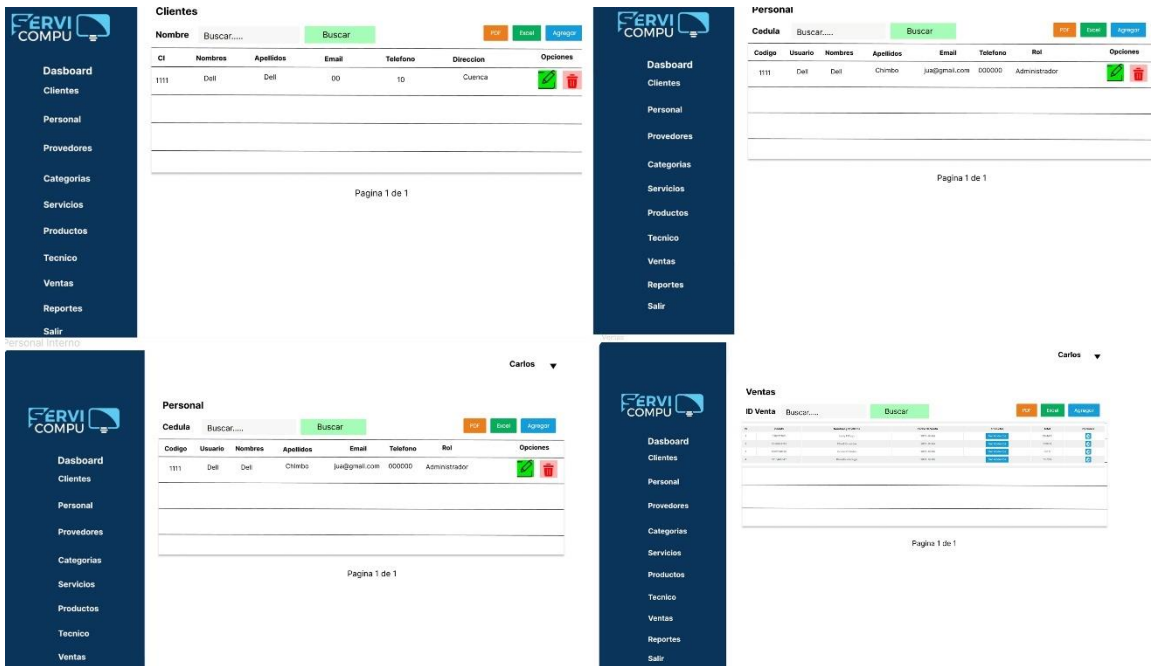


Figura 16: Módulos de Clientes, Personal, Ventas, Proveedores

Elaborado por: Autor

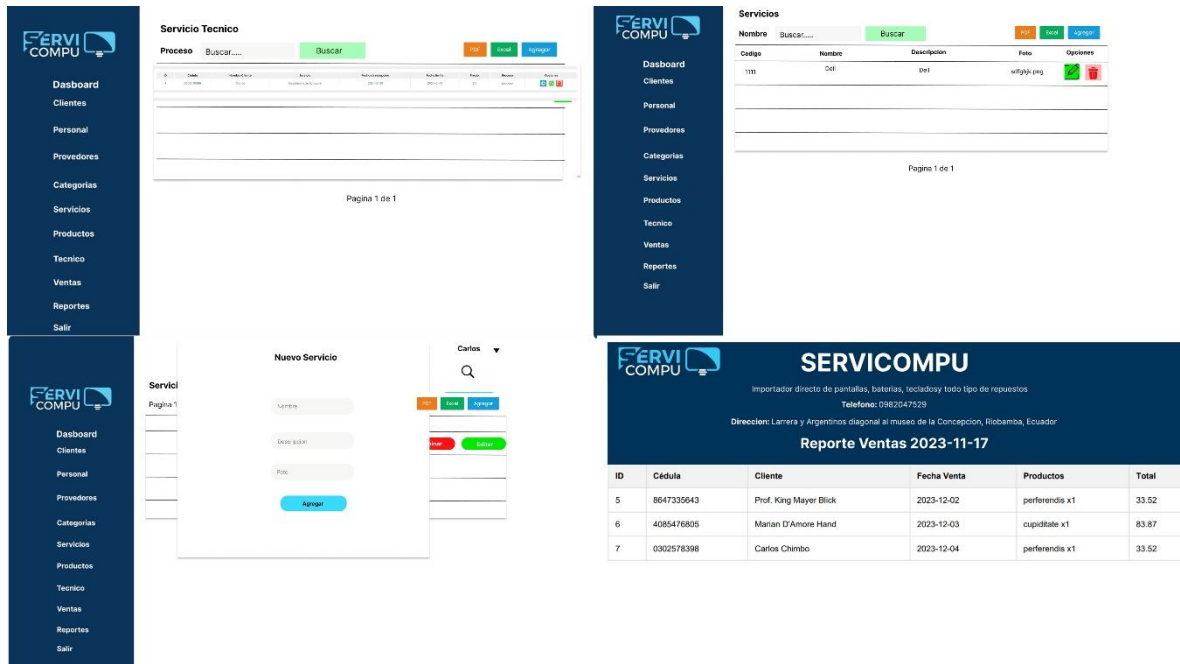


Figura 17: Módulos de Servicio Técnico, Servicios, Reportes

Elaborado por: Autor

Fase 5: Evaluar

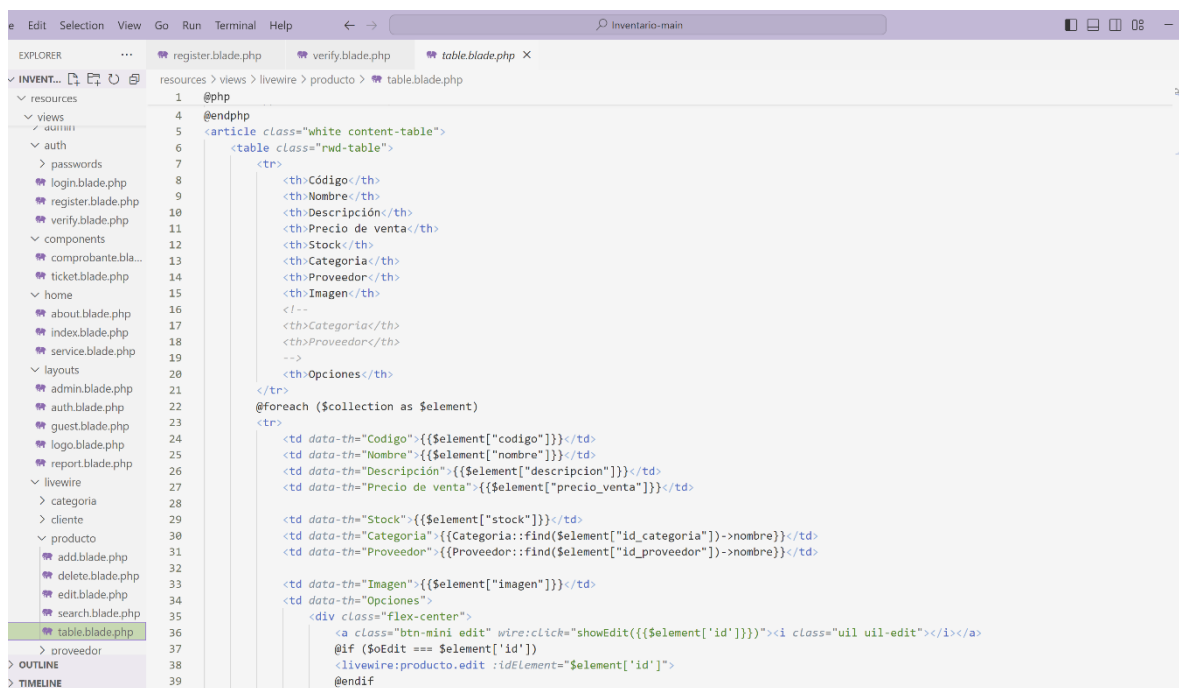
El usuario revisó y evaluó los prototipos, ayudó a identificar áreas de mejora y resolver posibles errores, al tomar las ideas, adaptarlas a las necesidades y garantizar que la aplicación web final cumpla con los requisitos y expectativas, los comentarios del usuario han brindado información valiosa para realizar ajustes y optimizaciones para garantizar una experiencia satisfactoria. Esto muestra cuán importante es la participación del usuario en el proceso de desarrollo para llegar a una solución final que sea realmente útil y satisfactorio de usar (Ver Anexo 2).

4.5 Desarrollo de la aplicación web

Durante el desarrollo del software, se utilizaron los prototipos previamente creados el front-end. Se continuó con el avance del back-end utilizando tecnologías como Laravel, React y MySQL como gestor de base de datos. Asimismo, se implementó una arquitectura lógica MVC para la aplicación web, permitió una estructura organizada y modular, separando las responsabilidades de los componentes y facilitando el mantenimiento del código.

4.5.1 Desarrollo de los módulos

Como objetivo se propuso el desarrollo back-end de los módulos de información, login, panel administrativo y productos para la gestión del inventario de la aplicación web, se trabajó con los prototipos aprobados y los requerimientos, con la finalidad de contar con un código limpio, eficiente y entendible.



```
1 @php
2
3
4 @endphp
5 <article class="white content-table">
6 <table class="rwd-table">
7
8 <tr>
9 <th>Código</th>
10 <th>Nombre</th>
11 <th>Descripción</th>
12 <th>Precio de venta</th>
13 <th>Stock</th>
14 <th>Categoría</th>
15 <th>Proveedor</th>
16 <th>Imagen</th>
17 </tr>
18 <tr>
19 <th>Categoría</th>
20 <th>Proveedor</th>
21 </tr>
22 <th>Opciones</th>
23 </tr>
24 @foreach ($collection as $element)
25 <tr>
26 <td data-th="Codigo">{{ $element["codigo"] }}</td>
27 <td data-th="Nombre">{{ $element["nombre"] }}</td>
28 <td data-th="Descripción">{{ $element["descripcion"] }}</td>
29 <td data-th="Precio de venta">{{ $element["precio_venta"] }}</td>
30 <td data-th="Stock">{{ $element["stock"] }}</td>
31 <td data-th="Categoría">{{ $element["id_categoria"]->nombre }}</td>
32 <td data-th="Proveedor">{{ $element["id_proveedor"]->nombre }}</td>
33 <td data-th="Imagen">{{ $element["imagen"] }}</td>
34 <td data-th="Opciones">
35 <div class="flex-center">
36 <a class="btn-mini edit" wire:click="{{ $element["id"] }}"><i class="uil uil-edit"></i></a>
37 @if ($oEdit == $element["id"])
38 <livewire:producto.edit :idElement="$element["id"]">
39 @endif
```

Figura 18: Desarrollo del Panel Administrativo, Información, Login, Productos

Elaborado por: Autor

4.5.2 Desarrollo de los módulos Back-end

De la misma manera, se planteó el desarrollo back-end de los módulos de proveedores, servicio técnico, servicios y personal de la aplicación web.

```

1 <article class="white content-table">
2 <table class="rw-table">
3
4 <tr>
5 <th>Cédula</th>
6 <th>Usuario</th>
7 <th>Nombres</th>
8 <th>Apellidos</th>
9 <th>Email</th>
10 <th>Teléfono</th>
11 <th>Rol</th>
12 <th>Opciones</th>
13 </tr>
14 @foreach ($collection as $element)
15 <tr>
16 <td data-th="Cédula">{{ $element["cedula"] }}</td>
17 <td data-th="User Name">{{ $element["name"] }}</td>
18 <td data-th="Nombres">{{ $element["nombres"] }}</td>
19 <td data-th="Apellidos">{{ $element["apellidos"] }}</td>
20 <td data-th="Correo">{{ $element["email"] }}</td>
21 <td data-th="Teléfono">{{ $element["telefono"] }}</td>
22 <td data-th="Rol">{{ $element["rol"] }}</td>
23 <td data-th="Opciones">
24 <div class="flex-center">
25 <a class="btn-mini edit" wire:click="showEdit('{{ $element['id'] }}')"><i class="uil uil-edit"></i></a>
26 @if ($oEdit == $element['id'])
27 <livewire:user.edit :idElement="$element['id']">
28 @endif
29 <a class="btn-mini delete" wire:click="openDelete('{{ $element['id'] }}')"><i class="uil uil-trash-alt"></i></a>
30 @if ($viewDelete == $element['id'])
31 <livewire:user.delete :idElement="$element['id']">
32 @endif
33 </div>
34 </td>
35 </tr>
36 @endforeach
37 </table>
38 </article>

```

Figura 19: Desarrollo de Proveedores, Servicio Técnico, Servicios y Personal

Elaborado por: Autor

4.5.3 Desarrollo de los Módulos (Reportes, Ventas, Clientes, Categorías)

Finalmente, el desarrollo de los últimos módulos reportes, ventas, categorías y clientes de la aplicación web.

```

22 <table>
23 <thead>
24 <tr>
25 <th>ID</th>
26 <th>Cédula</th>
27 <th>Cliente</th>
28 <th>Fecha Venta</th>
29 <th>Producto</th>
30 <th>Cantidad</th>
31 <th>Precio Unitario</th>
32 <th>Total</th>
33 </tr>
34 </thead>
35 <tbody>
36 @foreach($data as $item)
37 <tr>
38 <td>{{ $item['id'] }}</td>
39 <td>{{ Cliente::find($item['id_cliente'])->cedula }}</td>
40 <td>{{ Cliente::find($item['id_cliente'])->nombres . "\n" . Cliente::find($item['id_cliente'])->apellidos }}</td>
41 <td>{{ $item['fecha_venta'] }}</td>
42 <td>
43 @foreach (DetalleVenta::where('id_venta', $item['id'])->get() as $element)
44 <div>
45 {{ Producto::find($element->id_producto)->nombre }}
46 </div>
47 @endforeach
48 </td>
49 <td>
50 @foreach (DetalleVenta::where('id_venta', $item['id'])->get() as $element)
51 <div>
52 {{ $element->cantidad }}
53 </div>
54 @endforeach
55 </td>
56 </tr>
57 </tbody>
58 </table>

```

Figura 20: Desarrollo de Reportes, Ventas, Categorías y Clientes

Elaborado por: Autor

CAPÍTULO IV. RESULTADOS Y DISCUSIÓN

En el estudio se propuso como tercer objetivo la evaluación del rendimiento de la aplicación web, utilizando el software Apache JMeter. Basándose específicamente en analizar la utilización de recursos y el tiempo de respuesta.

Por lo que se plantea asignarle un porcentaje a cada indicador con un valor de un 50%.

Tabla 19: Indicadores de evaluación

Características	Indicadores	Porcentaje %	Total
Tiempo	Tiempo de respuesta	50%	50%
Uso de recursos	RAM	25%	50%
	Almacenamiento	25%	

Elaborado por: Autor

Dispositivo usado para la evaluación

Tabla 20: Características del dispositivo

	Windows
Dispositivo	Windows 10 Home
RAM	16GB
Almacenamiento	500GB
Versión del SO	22H2
Procesador	11th Gen Intel(R) Core (TM) i7-1165G7 @ 2.80GHz 2.80 GHz

Elaborado por: Autor

5.1 Resultados

Para las pruebas se empleó la herramienta Apache JMeter, con un total 40 simulaciones realizadas en un segundo. El uso de múltiples simulaciones permitió evaluar la capacidad de respuesta y la estabilidad, proporcionando información detallada sobre el rendimiento de la aplicación web bajo cargas pesadas.

5.1.1 Análisis de tiempo de respuesta

La Figura 21, muestra las mediciones del tiempo de respuesta de la aplicación web. Los resultados indican que, con 40 simulaciones en un segundo, se alcanzó un tiempo promedio de 0.508 segundos, Para garantizar una comprensión más completa de los datos, en el Anexo 3 se proporciona un análisis detallado de los tiempos de respuesta.

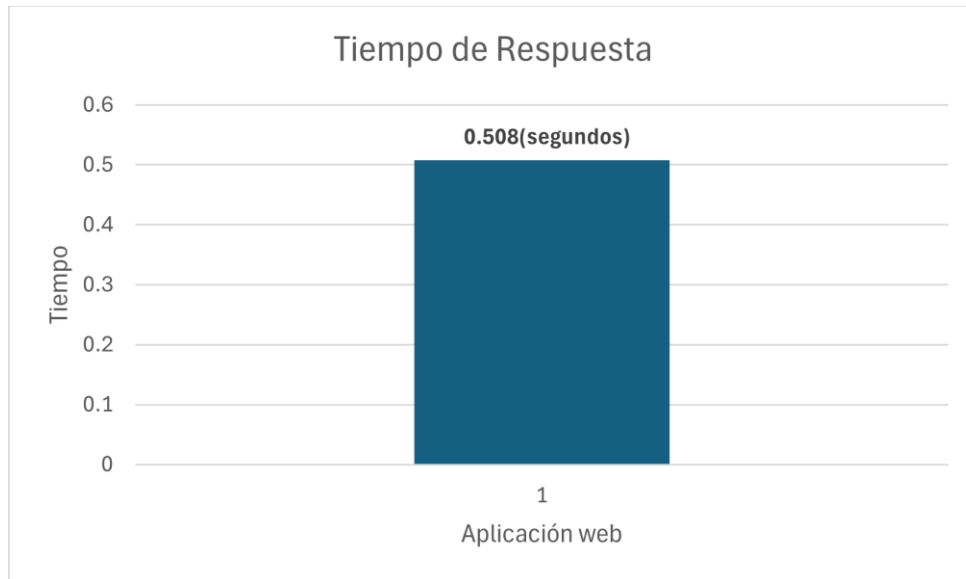


Figura 21: Tiempo de respuesta

Elaborado por: Autor

5.1.2 Análisis de uso de recursos

Para evaluar el indicador uso de recursos, se utilizó el componente de monitoreo de JMeter, Server Performance Monitoring (PerfMon), tiene la capacidad de evaluar el consumo de RAM y almacenamiento durante las simulaciones, mostrando los resultados en tiempo real. La Tabla 21 y la Figura 22, muestran el porcentaje de recursos del servidor consumidos, después de ejecutar la aplicación web para la gestión de inventarios.

Tabla 21: Uso de Recursos

Dimensión	Indicador	Porcentaje
Uso de recursos	Memoria RAM	39%
	Almacenamiento	1%

El uso promedio de la memoria RAM fue el 39%, y 1% del uso promedio del almacenamiento, demostrando que tiene una buena gestión de recursos. El Anexo 4 proporciona información adicional sobre el uso promedio de los recursos.

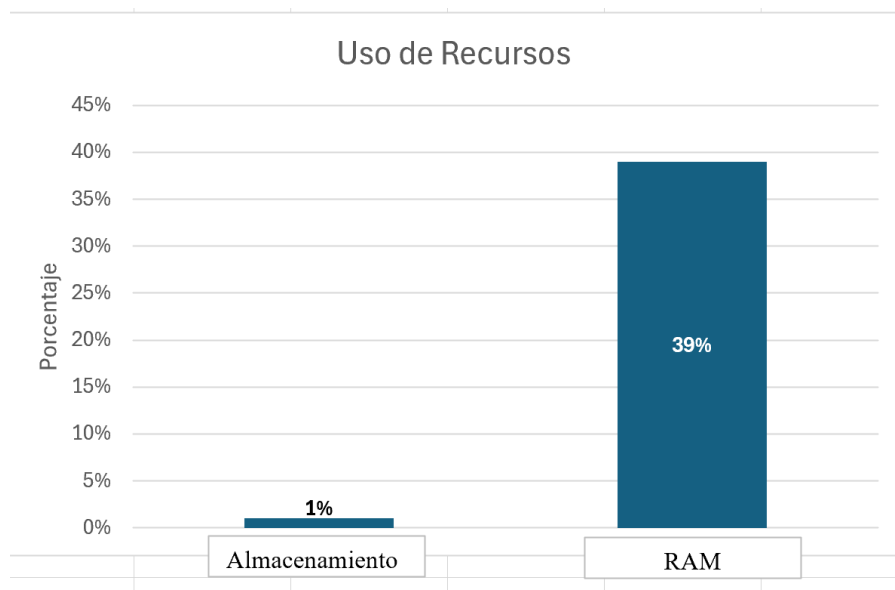


Figura 22: Uso de Recursos

Elaborado por: Autor

5.1.3 Análisis de resultados

Al finalizar las pruebas empleando la herramienta Apache JMeter los resultados arrojaron valores cuantitativos como se muestra en la Tabla 23.

Tabla 22: Resultados

Métricas	Windows	
	Resultados	Valor
Tiempo de respuesta	508 milisegundos	0.508 segundos
RAM	39%	6.24 GB
Almacenamiento	1%	5 GB

Elaborado por: Autor

Estos resultados muestran un rendimiento perfecto en términos de tiempo de respuesta, consumo de RAM y espacio de almacenamiento, demostrando que la aplicación web posee un buen funcionamiento bajo las condiciones especificadas de la prueba.

5.2 Discusión

Con anterioridad se mencionó que la metodología Design Thinking y su aplicación en el desarrollo de aplicaciones web, demostró ser muy efectiva para comprender a profundidad las necesidades y crear soluciones innovadoras centradas en el usuario, fomentando la colaboración y priorizando la iteración rápida para crear prototipos y obtener retroalimentación continua. Se obtiene una aplicación web que está completamente alineada

con las necesidades y expectativas de los usuarios. La efectividad de adaptarse mejor a las circunstancias cambiantes del proyecto, aumentando significativamente su usabilidad y rendimiento.

Según Fonseca (2021), Design Thinking ha mostrado resultados positivos en la investigación académica. Su uso es prometedor debido a su enfoque centrado en el usuario, resulta en una mayor creatividad para abordar diversos problemas encontrados durante el desarrollo. Un método importante para diseñar productos o servicios utilizando estrategias de carácter emocional y experimental para comprender a fondo las necesidades de los usuarios, reduciendo errores y alcanzando el éxito, utilizando de manera correcta sus etapas como empatía, definir, idear, prototipar y evaluar.

De igual manera, según los autores Castaño y Gonzáles (2020), es muy adecuado para proyectos centrados en las necesidades del usuario. Proporciona una buena estructura para aplicaciones y se puede utilizar de forma segura en otros proyectos de desarrollo de software. Beneficiará tanto a estudiantes de tecnologías como a empresas enfocadas en la resolución de problemas del usuario en la era digital. Design Thinking es un candidato prometedor para construir aplicaciones que proporcionan una alta satisfacción al usuario.

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusión

- Al usar una metodología ágil como Design Thinking para desarrollar una aplicación web, facilita adaptar los requerimientos del usuario demostrando ser altamente beneficioso, logrando una comprensión profunda de las necesidades, conduciendo a soluciones innovadoras, posibilitando obtener un software en poco tiempo y con una calidad verificada.
- Al desarrollar la aplicación web para la gestión de inventarios, permite al personal de la empresa optimizar y acelerar las actividades cotidianas, contribuyendo así el crecimiento y la sostenibilidad a largo plazo, la combinación de las herramientas Laravel y React permiten reducir el tiempo de desarrollo del software, la utilización del modelo MVC facilito la organización, mantenimiento y eficiencia de la aplicación web.
- Gracias a la herramienta Apache JMeter las pruebas indican un resultado generalmente satisfactorio de la aplicación web, con un tiempo de respuesta promedio de 0.508 segundos, demostrando eficiencia en el procedimiento de solicitudes, el uso de memoria RAM 39%, y el bajo consumo de almacenamiento 1% sugieren una gestión eficiente de recursos.

6.2 Recomendaciones

- Es importante utilizar una metodología ágil para la implementación de un software, permitiendo acelerar la finalización y resolución de fases en un corto periodo de tiempo. Haciendo uso de buenas prácticas de desarrollo. Aprovechando las funciones e información que ofrece Laravel y React.
- Se recomienda a la empresa Servicompu incorporar la funcionalidad de marketing digital y facturación electrónica a la aplicación web. Esta mejora permitirá agilizar y modernizar los procesos, brindando beneficios tanto a la empresa como a sus clientes.
- Debido a posibles cambios en la carga de la aplicación web, se recomienda un monitoreo continuo para garantizar un rendimiento óptimo en entornos de producción, garantizando al usuario una experiencia consistente y satisfactoria.

BIBLIOGRAFÍA

- Aguirre, D. (2023). Qué es React. Obtenido de Hostinger: <https://www.hostinger.es/tutoriales/que-es-react>
- Banda, R. (2022). AUTOMATIZACIÓN PRUEBAS DE CALIDAD. Obtenido de Repositorio.usil: <https://repositorio.usil.edu.pe/server/api/core/bitstreams/71014249-7c5a-4971-b2e0-f9435209da2d/content>
- Beltrán, S. (2020). SIMILITUDES Y DIFERENCIAS DE SCRUM como Design Thinking. Obtenido de Repository.unimilitar: <https://repository.unimilitar.edu.co/bitstream/handle/10654/41094/BELTRA%CC%81NGONZA%CC%81LEZSERGIOALEJANDRO2022.pdf?sequence=1&isAllowed=y#:~:text=SCRUM%20es%20un%20marco%20para,usuario%20para%20resolver%20problemas%20complejos>.
- Blandino, G. (2023). Figma: qué es y cómo funciona. Obtenido de Pixartprinting: <https://www.pixartprinting.es/blog/figma-que-es/>
- Bravo, M. (2022). Fases y redes en la metodología del Design Thinking. Obtenido de Scielo.org: http://www.scielo.org.ar/scielo.php?script=sci_arttext&pid=S1853-35232020000100091
- Castaño, F., & González, N. (2020). Creación de una aplicación móvil híbrida que aplique la metodología del Design Thinking y documente el producto de cada una las fases. Obtenido de Repositorio.uniagustiniana: <https://repositorio.uniagustiniana.edu.co/bitstream/handle/123456789/1389/GonzalezMorales-Natalia-2020.pdf?sequence=4>
- Castillo, P. (2020). Implementación del patrón arquitectónico MVC. Obtenido de Anuarioinvestigacion: <http://anuarioinvestigacion.um.edu.mx/index.php/a2020/article/view/118/101>
- Cougil, R. (2019). Rendimiento en Aplicaciones Web — Fundamentos. Obtenido de Medium.com: <https://medium.com/@rcougil/rendimiento-en-aplicaciones-web-fundamentos-50a004d33a19>
- Eras, C. (2021). Aplicación del design thinking en el desarrollo de soluciones web. Obtenido de Dspace.utpl: <https://dspace.utpl.edu.ec/handle/20.500.11962/28576>
- Fisas, A. (2019). La metodología design thinking y su implementación. Obtenido de Semantic Scholar:

- <https://pdfs.semanticscholar.org/efb8/5fd4a30d607210a9aa1b928c32724621d412.pdf>
- Fonseca, D. (2021). APLICACIÓN DE LA METODOLOGÍA DESIGN THINKING PARA EL MEJORAMIENTO DEL PROCESO DE TITULACION DE LA CARRERA INDUSTRIAL DE LA UNIVERSIDAD TECNICA DE COTOPAXI. Obtenido de repositorio.utc: <https://repositorio.utc.edu.ec/bitstream/27000/8306/1/PI-001825.pdf>
- Granizo, J. (2022). Conoce qué es Xampp. Obtenido de Dongee: <https://www.dongee.com/tutoriales/que-es-xampp/>
- Kanjila, J. (2022). Métricas de rendimiento de aplicaciones. Obtenido de Computerweekly.com: <https://www.computerweekly.com/es/consejo/10-metricas-de-rendimiento-de-aplicaciones-y-como-medirlas>
- Karina, A. (2020). METODOLOGÍA DESIGN THINKING. Obtenido de Repositorio.upn: <https://repositorio.upn.edu.pe/bitstream/handle/11537/31013/Ayala%20Murga%2c%20Karina%20Yajaira-Villarreal%20Acosta%2c%20Patricia%20Lucia.pdf?sequence=1&isAllowed=y>
- Laoyan, S. (2022). Design thinking. Obtenido de Asana: <https://asana.com/es/resources/design-thinking-process>
- Lazo, E. (2022). Tipos de aplicaciones web. Obtenido de Bambu-mobile.: <https://www.bambu-mobile.com/tipos-de-aplicaciones-web/>
- Marybeth, M. (2021). Design thinking para el desarrollo del pensamiento creativo. Obtenido de Renati.sunedu: <https://renati.sunedu.gob.pe/handle/sunedu/2847121>
- Paucar, J. (2022). Aplicación web para la mejora de la gestión. Obtenido de Repositorio.ucv: <https://repositorio.ucv.edu.pe/handle/20.500.12692/94752>
- Pérez, Y. (2020). Redalyc. Obtenido de Design Thinking en la Planificación de Pruebas de Software: <https://www.redalyc.org/journal/6738/673870835004/673870835004.pdf>
- Pesantez, C. (2021). Importancia de la aplicabilidad del “Design Thinking” . Obtenido de Repositorio.sangregorio: <http://repositorio.sangregorio.edu.ec/bitstream/123456789/2411/1/TESIS%20FINAL%20%20DESIGN%20THINKING%2021%20DE%20OCTUBRE%202021%20%281%29.pdf>
- Quistan, G. (2022). MetodologíaDesign. Obtenido de Repositorio.utp: https://repositorio.utp.edu.pe/bitstream/handle/20.500.12867/5482/G.Quistan_Trabajo_de_Investigacion_Bachiller_2022.pdf?sequence=1&isAllowed=y

- Ramos, M. (2018). Aplicaciones Web 2.^a edición. Obtenido de Books.google:
<https://books.google.es/books?hl=es&lr=&id=43G6AwAAQBAJ&oi=fnd&pg=PA1&dq=aplicaciones+web&ots=Dhb9sZu8FI&sig=tOkbk4n1UFisGKEw8GW--T1FWDE#v=onepage&q=aplicaciones%20web&f=false>
- Rodrigues, H. (2022). Qué es React.js. Obtenido de kinsta: <https://kinsta.com/es/base-de-conocimiento/que-es-react-js/>
- Velasco, R. (2021). Visual Studio Code. Obtenido de Softzone:
<https://www.softzone.es/programas/utilidades/visual-studio-code/>
- Vera, R. (2021). Qué es Laravel. Obtenido de Openwebinars:
<https://openwebinars.net/blog/que-es-laravel-caracteristicas-y-ventajas/>
- Zambrano, J. (2019). Aplicación web para gestionar los procesos administrativos . Obtenido de [Revistas.itsup:](https://revistas.itsup.edu.ec/index.php/sinapsis/article/view/200/294)
<https://revistas.itsup.edu.ec/index.php/sinapsis/article/view/200/294>
- Zúñiga, F. (2021). Qué es phpMyAdmin. Obtenido de Arsys:
<https://www.arsys.es/blog/phpmyadmin#:~:text=PhpMyAdmin%20es%20una%20aplicaci%C3%B3n%20web,muy%20popular%20basado%20en%20PHP.>

ANEXOS

Anexo 1. Brief general

Tabla 23: Brief general

Acerca de la empresa y del personal	
Nombre de la empresa	Servicompu
Contacto directo	https://www.facebook.com/accesorioslaptops
Nombre del Contacto	Roberto Villa.
Telefono	0982077529
E-mail	servicompu1@outlook.com
1. ¿Cuál es su negocio? ¿A qué se dedica su empresa?	La empresa Servicompu se dedica a la importación y distribución de partes y piezas de laptops y pc, con el objetivo de contribuir con repuestos de calidad.
2. ¿Cuál es el público objetivo con el que desea comunicarse?	Personas interesadas en adquirir partes o piezas de laptops o pc.
3. ¿Actualmente cuenta con alguna tecnología existente? (sitio web, aplicación móvil)	https://www.facebook.com/accesorioslaptops
4. ¿Tiene definida la estructura de contenidos y de qué tipo de contenidos se trata? ¿Cuántas páginas?	Ideas: - Algo interactivo con los usuarios - Mostrar información de la empresa - Mostrar los módulos
5. ¿Qué funciones le gustaría incluir? (Imágenes, menú, módulos).	- Información de la empresa - Interacción con botones para ver productos - Modulo de Productos - Módulo de imprimir Reportes - Módulo de gestion servicios - Módulo de gestión de proveedores - Módulo de gestión de usuarios - Módulo de gestión de ventas - Módulo de servicio técnico
6. ¿Cuáles son sus principales competidores en el mercado?	- SysteMarket - tecnomegacst - pclaptop
7. ¿Necesita fotografías, ilustraciones, infografías, videos, modelados 3D o algún tipo de creatividad? ¿Usted ya cuenta con ello?	Necesita: - Iconografía para inicio - Diseño de la aplicación Cuenta con: - Fotos de productos - Logotipo del establecimiento
8. ¿Se dispone de un equipo o personal para actualizar el contenido?	Si se contara con un administrador para actualizar contenidos.
9. ¿Para qué dispositivos lo tiene visualizado? (Móvil, tableta y/o PC)	- Laptops - Pc
10. ¿Su empresa cuenta con una imagen corporativa y diseños?	Cuenta con: - Logotipo - Paleta de colores
11. ¿Cómo visualiza la aplicación web? ¿Tiene alguna referencia de alguna aplicación?	- Como un panel de administración moderno - No se cuenta con referencias
12. ¿Cuándo le gustaría disponer de la aplicación? ¿Está trabajando en un marco de	Sin fecha especifica


tiempo estricto o tiene una fecha de lanzamiento?

13. Comentarios generales.

- Se cuenta con un administrador para el manejo de la aplicación web.
- Se pretende que se utilice con acceso a internet.
- Se busca algo nuevo e innovador.
- Se busca mostrar los servicios disponibles.
- Mostrar de manera visual y atractiva.

Elaborado por: Autor.

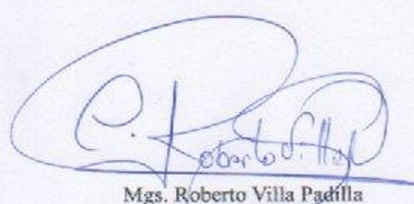
Anexo 2. Pruebas al usuario Sprint 1

 Empresa	Nombre del entrevistado:		Sprint	1
	Roberto Villa Mgs.			
Cargo:	Administrador de la empresa Servicompu			
Elaborado:	si	Revisado:		Aprobado:
Fecha:	27/06/2023	Fecha:	28/06/2023	Fecha:
Objetivo:	Puntuación del Sprint 1 correspondiente a los módulos (inicio, información de la empresa, login, servicios, personal, clientes, proveedores, categorías, productos, ventas, servicio técnico, reportes)			
Nombre Entrevistador:	Juan Carlos Chimbo Chumaina			

Calificar según su criterio con una X por pregunta. Donde SI es 1 y NO es 0 puntos.

Nivel Usuario

N	PREGUNTAS	SI	NO	OBSERVACION
1	¿La gama de colores crea un ambiente cómodo para el usuario?	X		
2	¿Los botones y navegabilidad están bien integrados en la aplicación web?	X		
3	¿La aplicación web es agradable para cualquier tipo de usuario?	X		
4	¿Las diferentes funciones de la aplicación web se encuentran muy bien integradas?	X		
5	¿Le gustaría utilizar la aplicación web frecuentemente?	X		
6	¿Puede utilizar la aplicación web sin ayuda de soporte técnico?	X		
7	¿La aplicación web le resulta fácil de comprender y utilizar?	X		
8	¿Se sintió cómodo al utilizar la aplicación web?	X		
9	¿Se sintió seguro al utilizar la aplicación web?	X		
10	¿La aplicación web le resulta muy acorde con los requerimientos para su diseño?	X		
Total		10/10		



Mgs. Roberto Villa Padilla
CI. 0602992109

Figura 23: Pruebas al Usuario Sprint 1

Elaborado por: Autor

Anexo 3. Tiempo de respuesta

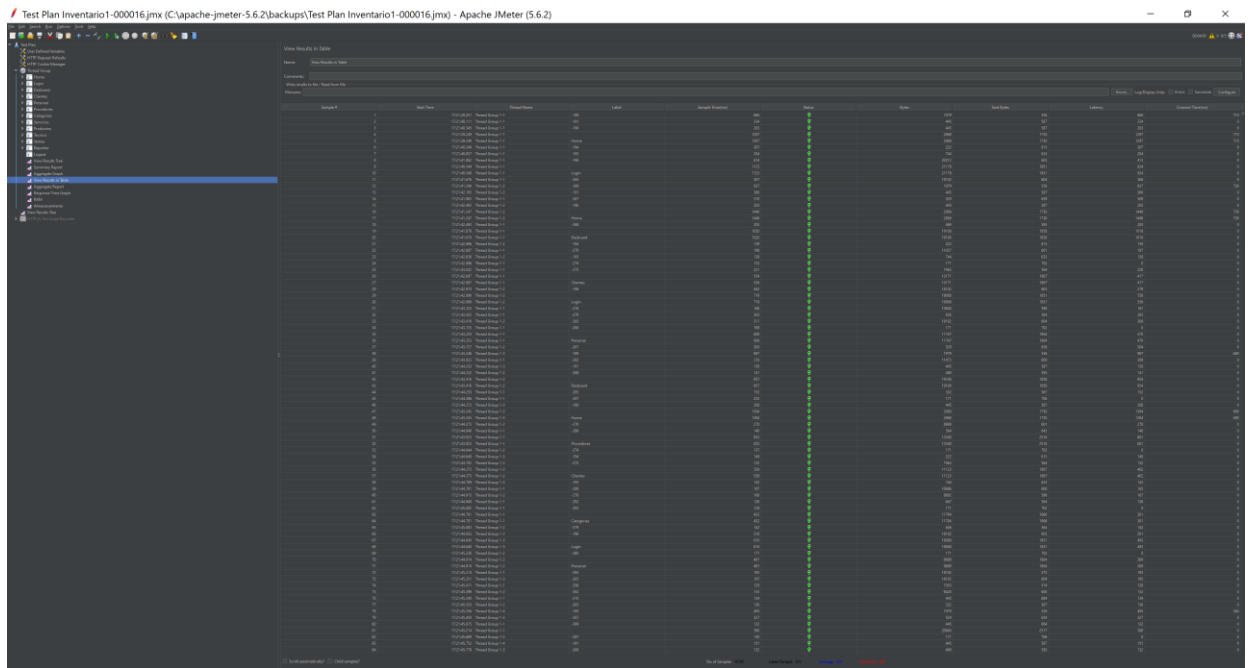


Figura 24: JMeter Pruebas Tiempo de respuesta

Elaborado por: Autor

Anexo 4. Uso de recursos

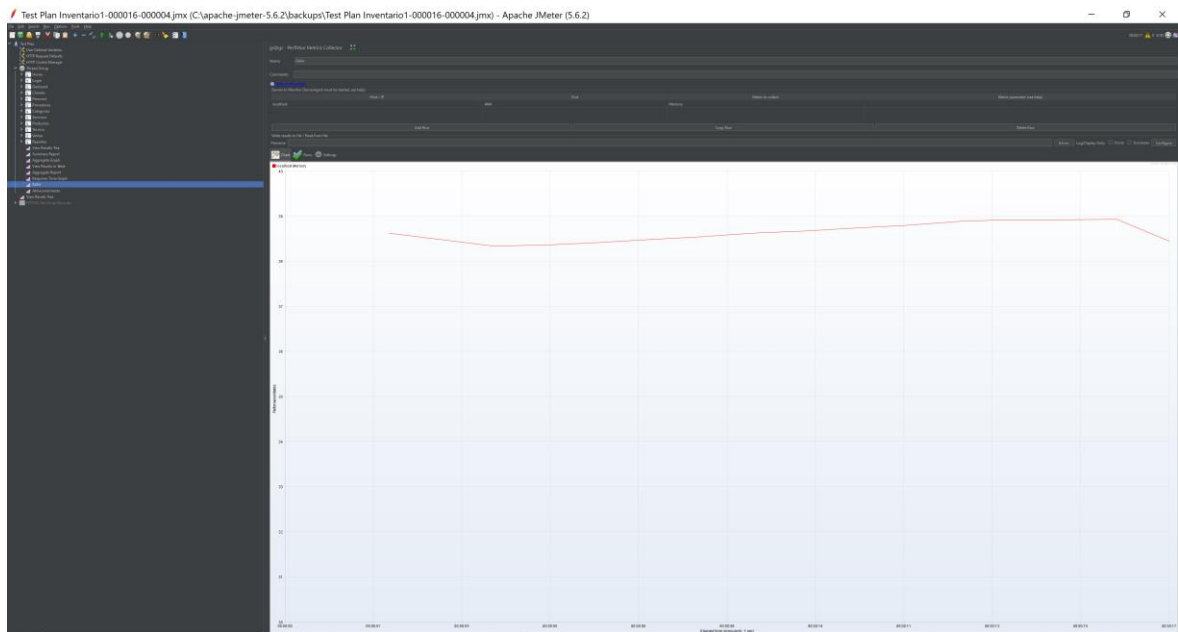


Figura 25: JMeter uso de RAM

Elaborado por: Autor

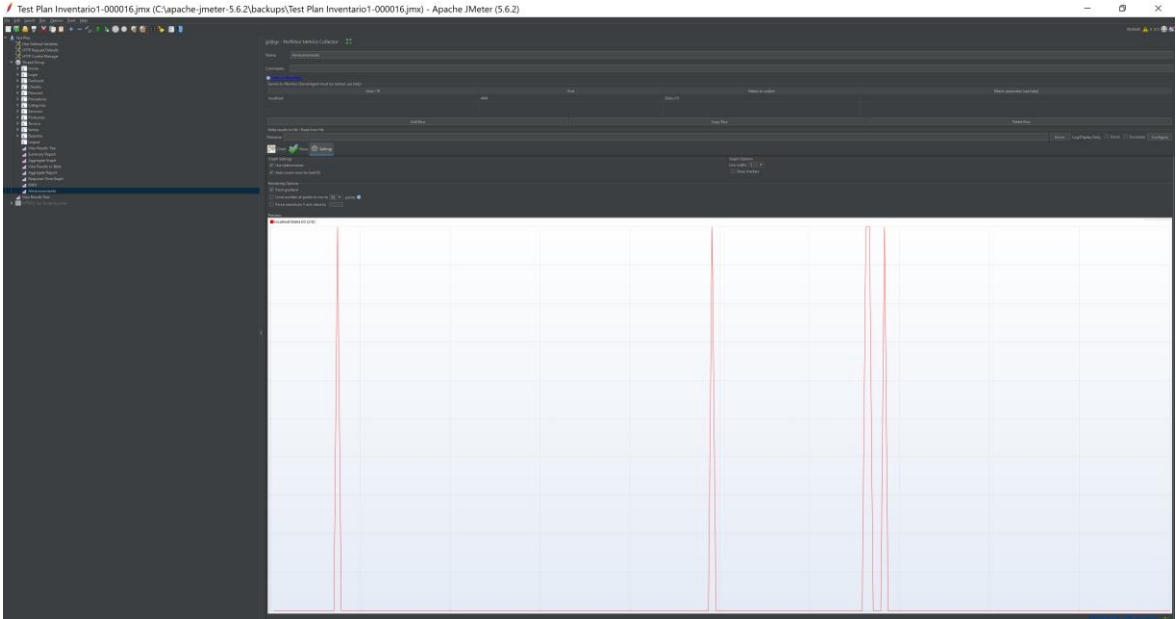


Figura 26: JMeter uso de almacenamiento

Elaborado por: Autor