



UNIVERSIDAD NACIONAL DE CHIMBORAZO
FACULTAD DE INGENIERIA
CARRERA DE SISTEMAS Y COMPUTACIÓN

Desarrollo de una aplicación web para el control de inventario de pymes
utilizando el framework Laravel

**Trabajo de Titulación para optar al título de Ingeniero en Sistemas y
Computación**

Autor:

Muñoz Molina, Luis Vinicio

Tutor:

Ing. Narváez Vilema, Miryan Estela

Riobamba, Ecuador. 2025

DERECHO DE AUTORÍA

Yo, **Luis Vinicio Muñoz Molina**, con cédula de ciudadanía **0604074245**, autor del trabajo de investigación titulado: **Desarrollo de una aplicación web para el control de inventario de pymes utilizando el framework Laravel**, 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, el 14 de abril del 2025.



Luis Vinicio Muñoz Molina

C.I: 0604074245



ACTA FAVORABLE - INFORME FINAL DEL TRABAJO DE INVESTIGACIÓN CARRERAS NO VIGENTES

En la ciudad de Riobamba, a los 14 días del mes de abril de 2025, luego de haber revisado el Informe Final del Trabajo de Investigación presentado por el estudiante **LUIS VINICIO MUÑOZ MOLINA** con CC: **0604074245**, de la carrera **INGENIERÍA EN SISTEMAS Y COMPUTACIÓN** y dando cumplimiento a los criterios metodológicos exigidos, se emite el **ACTA FAVORABLE DEL INFORME FINAL DEL TRABAJO DE INVESTIGACIÓN** titulado **"DESARROLLO DE UNA APLICACIÓN WEB PARA EL CONTROL DE INVENTARIO DE PYMES UTILIZANDO EL FRAMEWORK LARAVEL"**, por lo tanto se autoriza la presentación del mismo para los trámites pertinentes.



PhD. Miryan Narváez
TUTORA

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

Quienes suscribimos, catedráticos designados Miembros del Tribunal de Grado para la evaluación del trabajo de investigación **Desarrollo de una aplicación web para el control de inventario de pymes utilizando el framework Laravel**, presentado por **Luis Vinicio Muñoz Molina**, con cédula de identidad número **0604074245**, bajo la tutoría de la **Ing. 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 el 28 de mayo del 2025.

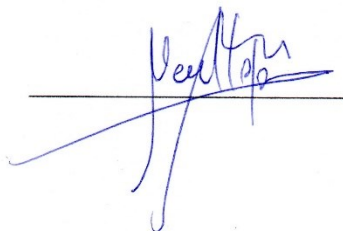
Presidente del Tribunal de Grado
PhD. Ximena Alexandra Quintana López



Miembro del Tribunal de Grado
Mgs. Lady Marieliza Espinoza Tinoco



Miembro del Tribunal de Grado
Mgs. Milton Paul López Ramos





CERTIFICACIÓN

Que, **MUÑOZ MOLINA LUIS VINICIO** con CC: **0604074245**, estudiante de la carrera **INGENIERÍA EN SISTEMAS Y COMPUTACIÓN, NO VIGENTE**, Facultad de **INGENIERÍA**; ha trabajado bajo mi tutoría el trabajo de investigación titulado "**DESARROLLO DE UNA APLICACIÓN WEB PARA EL CONTROL DE INVENTARIO DE PYMES UTILIZANDO EL FRAMEWORK LARAVEL**", cumple con el 3 %, de acuerdo al reporte del sistema Anti plagio **COMPILATIO**, porcentaje aceptado de acuerdo a la reglamentación institucional, por consiguiente autorizo continuar con el proceso.

Riobamba, 14 de abril de 2025



Firmado electrónicamente por:
**MIRYAN ESTELA
NARVAEZ VILEMA**

PhD. Miryan Narváez
TUTORA TRABAJO DE INVESTIGACIÓN

DEDICATORIA

Dedico el presente trabajo a mis maravillosos padres, Luis Muñoz Miranda y Livia Molina Ávila, quienes, con sus inalcanzables sacrificios, muestras de amor, me inspiran a ser una mejor persona, con el compromiso de seguir mejorando día con día, y enseñarme que todo es posible con la bendición de Dios.

A mis queridos hermanos Eduardo, María, Ana y Mariela por siempre brindarme su apoyo incondicional, por siempre brindarme la fuerza necesaria para seguir hacia adelante con sus palabras de apoyo, sus tan sentidos consejos, por su calidez humana.

A mi querido tío Holger Molina, que desde el cielo me sigue enviando sus bendiciones, cuya presencia sigue intacta en mis recuerdos. Gracias por enseñarme que las cualidades y valores humanos van de la mano con los logros académicos, a no rendirme, a tomarme la vida con alegría y ser agradecido por las pequeñas cosas que me da la vida.

Luis Muñoz

AGRADECIMIENTO

A Dios que con su grandiosa misericordia siempre me ayuda en cada etapa de mi vida, por darme la entereza necesaria para culminar mi carrera universitaria. y por todas las bendiciones que recibo día tras día.

Deseo expresar mi más profundo agradecimiento a cada una de las personas que hicieron posible la elaboración de este trabajo de tesis.

A mi tutor de tesis, Ing. Estela Narváez, por su amabilidad, paciencia, orientación y su contante disposición para apoyarme con su guía durante este proceso.

A todo el personal docente de la Facultad de Ingeniería de la carrera de Ingeniería en Sistemas y Computación, por compartir sus valiosos conocimientos, y por contribuir a mi crecimiento como ser humano a lo largo de toda mi carrera universitaria.

A todos mis compañeros de aula, quienes con su valiosa compañía hicieron de este camino una experiencia única y enriquecedora.

A toda mi grandiosa familia por ser un pilar fundamental en mi vida, agradecerles infinitamente por su amor, apoyo y comprensión en los momentos más difíciles.

Finalmente darle las gracias a mi querida U.N.A.C.H por acogerme en sus instalaciones y permitirme estudiar la carrera de Ingeniería en Sistemas y Computación.

Luis Muñoz

ÍNDICE GENERAL

DERECHO DE AUTORÍA.....	
DICTAMEN FAVORABLE DEL TUTOR Y MIEMBROS DE TRIBUNAL	
CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL	
CERTIFICADO ANTIPLAGIO.....	
DEDICATORIA	
AGRADECIMIENTO	
RESUMEN	
ABSTRACT	
CAPÍTULO I. INTRODUCCIÓN.....	14
1.1 PLANTEAMIENTO DEL PROBLEMA	15
1.1.1 Formulación del problema	15
1.1.2 Justificación	15
1.2 OBJETIVOS	16
1.2.1 Objetivo general	16
1.2.2 Objetivos específicos	16
CAPITULO II. MARCO TEÓRICO.....	17
2.1 Arquitectura de software.....	17
2.2 Modelo-Vista-Controlador (MVC)	17
2.3 Framework	18
2.3.1 Ventajas de un Framework.....	19
2.4 Laravel	19
2.4.1 Características del framework Laravel	20
2.4.2 Ventajas y desventajas de Laravel.....	21
2.4.3 El ecosistema Laravel	21
2.4.4 Composer.....	22
2.4.2 Artisan	23
2.4.3 Blade.....	23
2.4.4 ORM (Object Relational Mapping (Mapeo Objeto Relacional)).....	23
2.4.5 Eloquent ORM.....	24
2.5 Base de datos relacionales	24
2.5.1 Base de datos ventajas e inconvenientes.....	25
2.6 Sistema gestor de base de datos MySQL.....	26
2.6.1.1 Características de MySQL	26

2.6.1.2	Ventajas de MySQL	27
2.5.2	MySQL y Laravel.....	28
2.7	Apache Jmeter.....	28
2.8	API REST y RestFul.....	29
2.9	Metodología ágil para el desarrollo de software	30
2.10	Metodología ágil Kanban.....	30
CAPITULO III. METODOLOGÍA		33
3.1	Tipo y diseño de la investigación.....	33
3.1.1	Según el tipo de variable.....	33
3.1.2	Según el objeto de estudio.....	33
3.2	Población y muestra.....	33
3.3	Identificación de variables	33
3.4	Operacionalización de variables.....	33
3.5	Metodología de desarrollo	34
3.5.1	Análisis de requerimientos	34
3.5.2	Diseño y desarrollo de la aplicación web	36
3.5.3	Implementación del proyecto con Laravel	39
3.5.4	Desarrollo de módulos, controladores y vistas	39
3.5.5	API RESTful agrega la pantalla del api	50
3.5.6	Pruebas	51
3.6	Pruebas de carga y estrés con la herramienta Jmeter	52
CAPÍTULO IV. RESULTADOS Y DISCUSIÓN		55
4.1	Resultados	55
CAPITULO V. CONCLUSIONES y RECOMENDACIONES		60
5.1	CONCLUSIONES.....	60
5.2	RECOMENDACIONES.....	60
BIBLIOGRAFÍA		61
ANEXOS.....		64

ÍNDICE DE TABLAS

Tabla 1: Características del framework Laravel.....	20
Tabla 2: Ventajas y desventajas del framework Laravel.	21
Tabla 3: Herramientas para facilitar el desarrollo de sistemas con Laravel.....	22
Tabla 4: Operacionalización de Variables.	33
Tabla 5: Metodología Kanban aplicada a la fase de análisis.....	34
Tabla 6: Tabla de requerimientos funcionales.....	35
Tabla 7: Tabla de requerimientos no funcionales.....	36
Tabla 8: Metodología Kanban aplicada a la fase de diseño.	36
Tabla 9: Metodología Kanban aplicada a la fase de desarrollo.....	39
Tabla 10: Protocolos para comunicación con API RESTfull.....	50
Tabla 11: Metodología Kanban aplicada a la fase de pruebas	52
Tabla 12: Prueba de carga y estrés con 25 hilos.....	55
Tabla 13: Prueba de carga y estrés con 40 hilos.....	56
Tabla 14: Prueba de carga y estrés con 100 hilos.....	57

ÍNDICE DE FIGURAS

Figura 1: Patrones arquitectónicos de software.....	17
Figura 2: Patrón arquitectónico Modelo-Vista-Controlador.....	18
Figura 3: Flujo de trabajo del framework laravel.....	19
Figura 4: Manejador de dependencias de laravel.	22
Figura 5: Flujo de trabajo de Eloquent ORM.....	24
Figura 6: Modelo Entidad Relación.....	25
Figura 7: MySQL se basa en el modelo cliente-servidor.....	27
Figura 8: Estructura de Laravel.	28
Figura 9: Características de Apache Jmeter.	29
Figura 10: HTTP (Protocolo de Transferencia de Hipertextos).....	29
Figura 11: Modelos de proceso de desarrollo de software.....	30
Figura 12: Tablero Kanban.	31
Figura 13: Esquema de funcionamiento de Jmeter.....	32
Figura 14: Requerimientos Funcionales.	35
Figura 15: Diagrama Entidad Relación.	37
Figura 16: Diseño de Login.....	37
Figura 17: Diseño del panel principal.....	38
Figura 18: Diseño de formularios de ingreso de datos.	38
Figura 19: Git bash para construir el entorno.....	39
Figura 20: Vista de login de acceso a la aplicación web.	40
Figura 21: Panel de administración de usuarios.....	41
Figura 22: Vista de creación de usuarios.	41
Figura 23: Vista de edición de usuarios.....	42
Figura 24: Vista de usuario único.....	42
Figura 25: Panel de administración de empresas.	43
Figura 26: Vista de creación de empresas.....	43

Figura 27: Vista de edición de empresas.	44
Figura 28: Panel de administración de categorías.	44
Figura 29: Vista de formulario de creación de categorías.	45
Figura 30: Vista de edición de categorías.	45
Figura 31: Panel de administración de productos.....	46
Figura 32: Formulario de ingreso de productos.	46
Figura 33: Formulario de actualización del producto.....	47
Figura 34: Vista de gestión de movimientos.....	47
Figura 35: Ver formulario de creación de movimientos.....	48
Figura 36: Vista de edición de movimientos.....	48
Figura 37: Visita de gestión de Kardex	49
Figura 38: Vista de detalles del Kardex de un producto.....	49
Figura 39: Vista del servicio Restful.	50
Figura 40: JSON compartido a través de un API RESTful.....	51
Figura 41: Diagrama relacional de la base de datos en PhpMyAdmin.....	51
Figura 42: Panel general de la aplicación web.	52
Figura 43: Prueba de carga y estrés con 100 threads.....	53
Figura 44: Grafica de tiempo de respuesta con 100 hilos.....	54
Figura 45: Gráfica de tiempo de respuesta con 25 threads.	55
Figura 46: Grafico de tiempo de respuesta con 40 threads.....	57
Figura 47: Grafico de tiempo de respuesta con 100 threads.....	58

RESUMEN

En la actualidad, la automatización de procesos ha transformado la gestión empresarial de pequeñas y grandes empresas (Pymes), las aplicaciones web modernas ofrecen una ventaja significativa al mejorar la independencia de actividades y optimizar procesos administrativos. En particular, la gestión de inventarios es una de las áreas más críticas dentro de las organizaciones, ya que requiere un enfoque integral para la supervisión de mercadería, la actualización de datos y el control eficiente de los recursos disponibles.

Este proyecto de investigación tuvo como objetivo desarrollar una aplicación web para la gestión de inventarios de pequeñas y medianas empresas utilizando Laravel. Este framework se caracteriza por ser escalable y mantenible, lo que garantiza una adecuada evolución del sistema a lo largo del tiempo. Laravel emplea el patrón arquitectónico Modelo-Vista-Controlador (MVC), que facilita la separación de responsabilidades dentro del código y asegura una correcta gestión de solicitudes, procesamiento de datos y presentación de información al usuario. Además, su modularidad permite la integración de funcionalidades específicas, así como la comunicación con otras aplicaciones a través de servicios RESTful, lo que amplía sus posibilidades de interoperabilidad.

Para el desarrollo de la aplicación web, se empleó la metodología ágil Kanban, permitiendo un control eficiente de las actividades mediante la visualización y gestión de tareas en un flujo de trabajo optimizado. Su flexibilidad y enfoque en la mejora continua facilitaron la adaptación a cambios y aseguraron el éxito en cada etapa del desarrollo. Además, se realizó una evaluación del rendimiento con Apache Jmeter para medir la capacidad de respuesta del sistema bajo distintos niveles de carga. Los resultados demostraron que la aplicación mejora significativamente la gestión de inventarios, reduciendo el tiempo de actualización de datos y aumentando la precisión en la supervisión de mercadería.

Palabras claves: Aplicación web, gestión de inventarios, Laravel, metodología Kanban, optimización de procesos.

ABSTRACT

Nowadays, the automation process has transformed the business management of small and large enterprises (SMEs). Modern web applications provide a meaningful advantage by enhancing independent activity and optimizing administrative processes. In particular, inventory management is the most critical area within organizations. It requires a comprehensive approach to merchandise supervision, data updating, and efficient control of available resources.

This research project was to develop a web application for inventory management for small and medium enterprises using "Laravel". This framework is characterized as being scalable and maintainable, ensuring the proper evolution of the system over time. "Laravel" employs the Model-View-Controller (MVC) architectural pattern, which facilitates the separation of responsibilities within the code and ensures the correct request management, data processing, and presentation of information to the user. Furthermore, its modularity allows the integration of specific functionalities and communication with other applications through RESTful services, which expands its interoperability possibilities.

The "Agil Kanban" methodology was applied to develop the web application, enabling efficient control activities through visualization and management tasks in an optimized workflow. Its flexibility and focus on continuous improvement facilitated change adaptation and ensured success at every stage of development. In addition, a performance evaluation was done using "Apache JMeter" to measure the system's answers under different charge levels. The results demonstrated that the application significantly improves inventory management, reducing data update time and increasing merchandise monitoring precision.

Keywords: Web application, inventory management, Laravel, Kanban methodology, process optimization.



Firmado electrónicamente por:
**JENIFFER VANESSA
PALACIOS MORENO**

Reviewed by: Mgs. Vanessa Palacios
ENGLISH PROFESSOR
C.C. 0603247487

CAPÍTULO I. INTRODUCCIÓN

En la actualidad la automatización de procesos surgió con la finalidad de reducir costos y el tiempo de servicio, dando como resultado beneficios para el cliente y la empresa. Las aplicaciones web tienen una gran ventaja en cuanto a independencia, ya que solo necesitan tener acceso a la red, un buen navegador web y un dispositivo hardware para acceder al mismo.

En el desarrollo de aplicaciones web existe una variedad de lenguajes de programación y frameworks, la mayoría de código libre, como de PHP (Preprocesador de hipertexto) cuya estructura es similar a JavaScript y a Python, lenguajes popularmente utilizados en el desarrollo web moderno, a diferencia de los lenguajes mencionados, PHP trabaja con más eficiencia en el lado del servidor, realizando peticiones asíncronas a la base de datos y enviándolas al lado del cliente a través de peticiones HTTP (Deyimar, 2024).

Las aplicaciones web desarrolladas con PHP suelen incluir la incorporación de librerías externas y frameworks que reducen la necesidad de insertar de código y lo reemplazan con módulos que se pueden reutilizar n veces. Laravel es un framework escalable con bases de código de fácil mantenimiento y reutilización, a su vez, Laravel tiene un sistema de empaquetado modular y una gestión de dependencias bastante sólida. El desarrollo de aplicaciones web con Laravel y PHP se refuerzan con una arquitectura que sigue el patrón Modelo-Vista-Controlador, lo que permite separar las actividades y dividirlas en módulos independientes y de fácil mantenimiento. Los controladores llevan a cabo la tarea de gestionar solicitudes desde la interfaz de usuario, procesan los datos del modelo y finalmente devuelven la información estructurada y legible al usuario (Vera, 2021).

Laravel es un framework que permite la integración con múltiples bases de datos relacionales y no relacionales. Para este proyecto se utilizó MySQL por su alta confiabilidad en el manejo de datos estructurados y el soporte en la creación de aplicaciones web expansivas; su principal objetivo es mejorar el rendimiento de las aplicaciones mediante la implementación de diversas funcionalidades, permitiendo que la aplicación sea manejable, segura y de fácil acceso para los usuarios (Huillcen, Soria, & Palomino, 2022).

Para el desarrollo de la aplicación web, se utilizó la metodología ágil Kanban, la cual permite un mejor control de las actividades de desarrollo mediante la visualización y gestión de tareas en un flujo de trabajo optimizado. Esta metodología se escogió debido

a su flexibilidad y enfoque en la mejora continua, lo que facilita la adaptación a cambios y garantiza la entrega de un producto funcional en cada etapa del desarrollo. Adicionalmente, se llevó a cabo una evaluación del rendimiento de la aplicación utilizando la herramienta Apache JMeter, con el fin de medir la capacidad de respuesta del sistema bajo diferentes niveles de carga. Los resultados obtenidos evidenciaron que la aplicación web optimiza significativamente los procesos de gestión de inventarios, reduciendo el tiempo empleado en la actualización de datos y mejorando la precisión en la supervisión de mercadería.

1.1 PLANTEAMIENTO DEL PROBLEMA

1.1.1 Formulación del problema

La gestión de inventarios en pequeñas y medianas empresas ha presentado deficiencias que impactan negativamente en la eficiencia operativa, la toma de decisiones y la rentabilidad del negocio. Entre los principales problemas se encuentran la falta de trazabilidad en los movimientos de stock, la desactualización de registros, la sobreacumulación o escasez de productos, y la ausencia de integración con otros sistemas de gestión empresarial.

Los métodos tradicionales de control de inventario, basados en hojas de cálculo o registros manuales, generan inconsistencias en la información, dificultando la identificación de tendencias y la optimización de los recursos. La falta de automatización también limita la capacidad de respuesta ante variaciones en la demanda, afectando la planificación de compras y la distribución de productos.

1.1.2 Justificación

Dada la necesidad de soluciones tecnológicas adaptables y escalables, se propuso el desarrollo de una aplicación web para la gestión de inventarios de Pymes utilizando Laravel, un framework PHP que facilita la implementación de arquitecturas modernas como MVC. Laravel proporciona herramientas avanzadas para la gestión de bases de datos, implementación de APIs RESTful, autenticación de usuarios, y despliegue en entornos en la nube, lo que garantiza una solución robusta y segura.

Este proyecto aborda la implementación de una aplicación web que permite a las Pymes gestionar su inventario en tiempo real, minimizando errores humanos, mejorando la visibilidad del stock. Al implementar una plataforma eficiente y accesible, las Pymes

podrán optimizar sus operaciones, adaptarse rápidamente a cambios en la demanda y mejorar sus márgenes de rentabilidad.

¿Cómo influye una aplicación web para la gestión de inventarios en Pymes desarrollado con Laravel, en el rendimiento general del sistema y en la toma de decisiones?

La implementación de una aplicación web para la gestión de inventarios en Pymes, desarrollada con Laravel, tiene un impacto significativo en el rendimiento general del sistema y en la toma de decisiones empresariales. Laravel es un marco de trabajo elegante y expresivo, primordial en el desarrollo de aplicaciones web que no exigen demasiados recursos tecnológicos. Las aplicaciones web desarrolladas con Laravel son robustas y escalables, capaces de manipular grandes cantidades de registros como compras, inventarios e incluso información de productos en tendencia, todo lo que una Pymes necesita para su administración.

Otra ventaja que una aplicación web con Laravel ofrece es su independencia, puede ser ejecutado a través de cualquier sistema operativo, no necesita de un servidor local ya que la aplicación final se aloja en un servidor remoto, esta ventaja se ve reflejada en el costo y la rapidez con la que Pymes recuperan información de sus actividades comerciales.

1.2 OBJETIVOS

1.2.1 Objetivo general

Desarrollar una aplicación web para el control de inventario de Pymes utilizando el framework Laravel.

1.2.2 Objetivos específicos

- Investigar el framework Laravel en el desarrollo de aplicaciones web de gestión de inventarios.
- Desarrollar la aplicación web para gestionar de manera eficiente los inventarios de Pymes utilizando Laravel.
- Evaluar el rendimiento de la aplicación web utilizando la herramienta Jmeter.

CAPITULO II. MARCO TEÓRICO

2.1 Arquitectura de software

Según Arias & Durango (2014), se conoce como arquitectura de software a la estructura fundamental de un sistema, sus componentes y las relaciones entre ellos, además, las propiedades que resultan de sus interacciones. Para lograr que un software tenga una arquitectura que comparta responsabilidades con el resto de componentes integrados, es necesario comprender los requisitos que deben cumplirse durante todas las fases de vida del software. También, es fundamental llevar una documentación clara y consistente para compartirla con el equipo de desarrollo, un código que no está debidamente documentado perderá su objetivo inicial, y, por lo tanto, no será escalable.



Figura 1: Patrones arquitectónicos de software.
Obtenido de: <https://canvia.com/arquitectura-software/>

2.2 Modelo-Vista-Controlador (MVC)

Los autores Fernández & Díaz (2012) argumentan que el patrón MVC es una arquitectura de software que segmenta una aplicación en tres componentes interrelacionados: el Modelo, Vista y Controlador. Esta separación facilita la gestión y escalabilidad del software al asignar responsabilidades específicas a cada componente.

Este modelo de arquitectura de software se caracteriza por segmentar en tres componentes relacionados entre sí:

- **Modelo:** lleva la lógica de negocio y maneja los datos de la aplicación web, permite su acceso y manipulación, además mantiene la integridad y claridad de los datos.
- **Vista:** cuando los datos son procesados por la lógica de negocios son mostrados en formato legible para el usuario a través de la interfaz gráfica.
- **Controlador:** procesa las solicitudes que el usuario realiza a través de la interfaz gráfica, las encamina hacia la lógica de negocio y finalmente las regresa a las vistas.

Este patrón promueve la separación de responsabilidades, facilitando el mantenimiento y la escalabilidad de las aplicaciones. Al mantener independientes la lógica de negocio, la presentación y el control de flujo, es más sencillo realizar modificaciones o añadir nuevas funcionalidades sin afectar otras partes del sistema.

El modelo MVC aplica la separación de responsabilidades de manera efectiva para aplicaciones básicas, pero con potencial de escalabilidad. Es muy común utilizar este tipo de arquitectura en frameworks como: Angular, Node, o Laravel, porque son muy populares en el entorno de desarrollo web moderno.

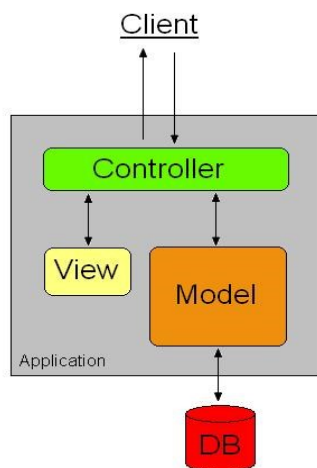


Figura 2: Patrón arquitectónico Modelo-Vista-Controlador.
Obtenido de: https://commons.wikimedia.org/wiki/File:MVC_Diagram.jpg

2.3 Framework

Según Heredia (2022) un framework es un subsistema integrado por funciones y métodos que reducen la cantidad de código y permiten su reutilización. Un framework es también un stack tecnológico que fue creado para reducir el código espagueti, los desarrolladores de frameworks siempre están en constante actualización, beneficiando la escalabilidad y

el soporte a largo plazo. Sin embargo, un framework no está hecho para todo proyecto, generalmente, utilizan para sistemas con potencial de escalabilidad, aplicaciones empresariales grandes que no escatiman en uso de recursos como memoria o almacenamiento remoto.

2.3.1 Ventajas de un Framework

Entre las ventajas que el autor destaca se encuentran los siguientes:

- Marcos de desarrollo que previamente están definidos.
- Reutilización de código en varias secciones del proyecto.
- Mayor velocidad y alcance.
- La gran mayoría de patrones arquitectónicos se basan en frameworks.
- Cuenta con una amplia comunidad de soporte.
- Son adaptables y fáciles de mantener.

2.4 Laravel

El autor Cíceri (2018) plantea que, el framework Laravel es considerado el más popular para el lenguaje de programación PHP, se originó a partir del año 2011 y en un principio se focalizaba a tareas de autenticación. Su segunda versión adoptó el patrón arquitectónico MVC con una sintaxis simple y previniendo el famoso código espagueti. A partir del año 2015 Laravel empezó la distribución de versiones LTS (Long Term Support), esto significaba que su diseño tendría soporte durante periodos más largos, garantizando la continuidad en la creación de proyectos a largo plazo.

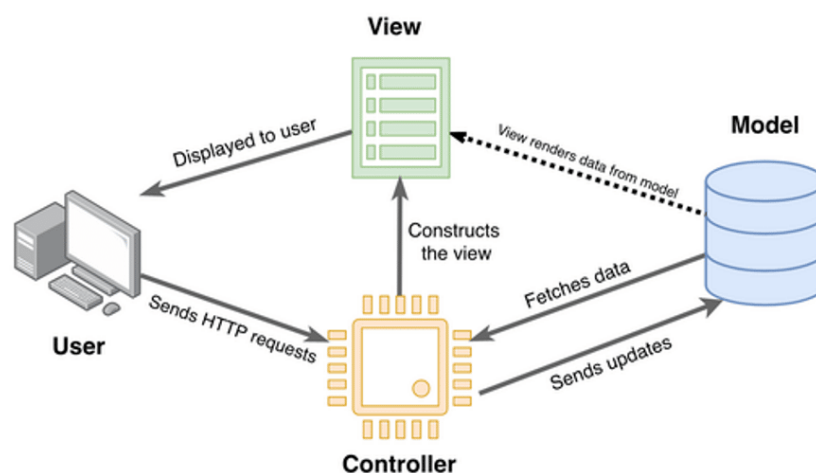


Figura 3: Flujo de trabajo del framework laravel.
Obtenido de: <https://kinsta.com/es/blog/laravel-vs-node/>

2.4.1 Características del framework Laravel

Según la investigación del autor Hurtado (2022), Laravel es un poderoso framework PHP que simplifica el desarrollo de aplicativos webs, proporcionando herramientas flexibles y poderosas para la creación de proyectos escalables con el tiempo y robustos. Laravel permite construir aplicaciones modernas con un código mantenible y limpio, evitando el código espagueti.

Tabla 1: Características del framework Laravel.

Característica	Descripción
Modular y extensible	Laravel es un framework que facilita la incorporación de cualquier funcionalidad adicional, ya que dispone de un repositorio denominado Packalyst, el cual alberga más de 5,500 paquetes. En este, se pueden encontrar diversas soluciones ya implementadas para cubrir las necesidades del proyecto.
Tareas automatizadas	A través de la API Elixir de Laravel, es posible crear tareas de Gulp que permiten configurar el uso de preprocesadores para la compresión de archivos CSS (Hojas de Estilo en Cascada) y JavaScript.
Http routing	Laravel dispone de un sistema de enrutamiento ágil y eficaz, similar al utilizado en Ruby on Rails. Este sistema facilita la conexión de las diferentes secciones de la aplicación con las rutas que el usuario introduce en el navegador.
Http Middleware	Permite el uso de Middleware, que se encarga de examinar y filtrar las solicitudes HTTP en el servidor.
Caché	Cuenta con un sistema de caché robusto y configurable.
Encriptación	Utiliza la seguridad de OpenSSL y el cifrado AES-256-CBC.
Cola de tareas (Queues)	Ofrece la opción de ejecutar procesos largos y complejos en segundo plano mediante el uso de listas de tareas.
Integra un sistema ORM	Cuenta con un sistema de mapeo de datos relacional denominado Eloquent, y, además, permite la construcción de consultas directas a la base de datos mediante su Query Builder.
Autenticación	Se lleva a cabo a través de un inicio de sesión con usuario y contraseña, lo que permite restringir el acceso y definir los permisos según el tipo de usuario.

Fuente: (Hurtado, 2022).

2.4.2 Ventajas y desventajas de Laravel

Tabla 2: Ventajas y desventajas del framework Laravel.

Ventajas	Desventajas
• Laravel se considera un framework accesible y sencillo de aprender, debido a que está desarrollado en el lenguaje de programación PHP. • Facilita la colaboración en equipo, además de mejorar la claridad, el mantenimiento y la reutilización del código. • Facilita la gestión de bases de datos como MySQL, PostgreSQL, SQLite y SQL Server, permitiendo la manipulación de tablas directamente desde el código y asegurando un control de versiones a través de su sistema de migraciones. • Hace uso del motor de plantillas Blade para agilizar el proceso de compilación, lo que permite a los usuarios añadir nuevas funcionalidades de forma sencilla. • Laravel simplifica la expansión de funcionalidades a través de paquetes o bibliotecas externas, lo que permite integrar fácilmente diversas librerías, agilizando el desarrollo de aplicaciones y reduciendo el tiempo de programación. • Incluye un intérprete de línea de comandos denominado Artisan, que ofrece herramientas avanzadas para realizar tareas y migraciones. • Dispone de una muy amplia documentación	• Resulta complicado de optimizar. • Dado que es un lenguaje interpretado en tiempo de ejecución, en ciertos casos puede ser un inconveniente que el código fuente no pueda ser ocultado. • Existen muchos métodos incluidos en el enrutamiento inverso que son complejos

Fuente: (Hurtado, 2022).

2.4.3 El ecosistema Laravel

Laravel tiene una de los ecosistemas más amigables para desarrolladores web, se destaca las siguientes herramientas integradas:

Tabla 3: Herramientas para facilitar el desarrollo de sistemas con Laravel.

Característica	Descripción
Homestead	Máquina virtual para ejecutar en un ambiente virtual los proyectos de Laravel
Lumen	Usada para la creación de servicios web, más ligeros y más rápido de desarrollar.
Laravel cashier	Creado para acoplarse con sistemas externos para suscripciones o métodos de pago en línea.
Statamic	Un administrador de contenido desarrollado por Laravel.

Fuente: (Cíceri, 2018).

2.4.4 Composer

Es un software encargado de la gestión de paquetes y librerías de PHP, se encarga de verificar la compatibilidad, manteniendo la legibilidad del código a largo plazo. Se instala de manera global en el sistema operativo anfitrión y es recomendable actualizarlo de manera periódica.

Una de las ventajas de este gestor de dependencias es su facilidad de uso, para crear un nuevo proyecto, se utiliza el comando ‘composer init’ desde el símbolo del sistema o con git bash, después de ejecutar el comando se crea el historial de cambios del proyecto dentro de un archivo llamado ‘composer.json’.

Para crear un proyecto usando el framework Laravel y Composer se utiliza la misma estructura, incorporando la dependencia Laravel de esta manera: ‘composer create-project --prefer-dist laravel/laravel’, con este comando se descarga la última versión estable de Laravel junto con sus dependencias (Equipo de Contenidos de GoDaddy, 2024).

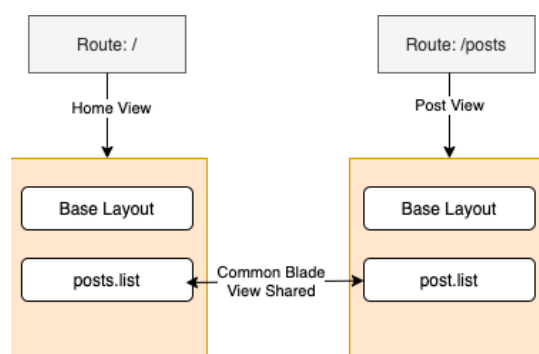


Figura 4: Manejador de dependencias de laravel.

Obtenido de: <https://www.learn2torials.com/a/laravel-view-composers>

2.4.2 Artisan

Es una interfaz de línea de comandos (CLI) que viene acompañada de un proyecto con Laravel y PHP. Cuenta con muchas funcionalidades, como la automatización de tareas, gestión de migraciones a base de datos, crear plantillas de código, entre otras (Willis, 2024).

Los comandos más comunes en el desarrollo web con Laravel son los siguientes:

- **php artisan list:** muestra la lista general de comandos disponibles en la versión del proyecto.
- **php artisan help [command_name]:** muestra las instrucciones de uso para un comando en específico.
- **php artisan make:controller [ControllerName]:** genera un controlador en el proyecto.
- **php artisan migrate:** utilizado para ejecutar las migraciones a la base de datos.
- **php artisan tinker:** tinker es también una línea de comandos que comunica a toda la aplicación, se utiliza para inserción de datos iniciales o de configuración global.

2.4.3 Blade

Es el motor de plantillas de vistas que más se utiliza en proyectos creados con Laravel, su sintaxis es simple y flexible por su orgánica combinación entre código PHP y etiquetas HTML, muy indispensable para el mantenimiento de código. Blade tiene muchas ventajas, como evitar la duplicación de código al heredar plantillas, sus directivas permiten manejar estructuras condicionales desde las vistas, un enfoque que beneficia mucho al rendimiento de la aplicación (10Code Software Design, 2024).

2.4.4 ORM (Object Relational Mapping (Mapeo Objeto Relacional))

De acuerdo a lo indicado por Yanes & Gracia del Busto (2011), el mapeo objeto-relacional es un enfoque de programación que se utiliza para transformar los datos de un sistema basado en objetos, en un lenguaje de programación hacia un formato compatible con una base de datos relacional. En términos prácticos, da lugar a una base de datos virtual orientada a objetos sobre una estructura de base de datos relacional. Gracias a esta técnica, se puede aprovechar las características propias de la programación orientada a objetos, como la herencia y el polimorfismo.

2.4.5 Eloquent ORM

Según el autor, (Jound, s.f.) Eloquent ORM que se incluye en Laravel utiliza la implementación del patrón Active Record para acceder a la base de datos. Este patrón permite que las bases de datos relacionales se representen mediante código orientado a objetos. En la estructura MVC, cada clase de modelo creada se corresponde con una tabla en la base de datos, y la instancia de dicho modelo está asociada a una única fila de esa tabla. Los atributos del modelo se vinculan con los campos de la tabla. Cuando se crea y guarda un objeto, se añade una nueva fila a la tabla correspondiente. Al cargar un objeto, se recupera la información desde la base de datos, y cuando el objeto se actualiza, la fila correspondiente en la tabla también se modifica.

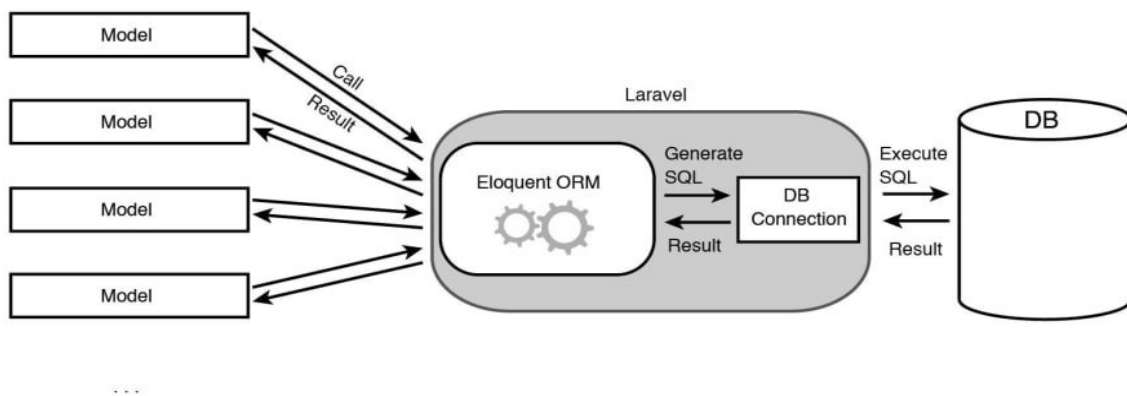


Figura 5: Flujo de trabajo de Eloquent ORM

Obtenido de: <https://www.diva-portal.org/smash/get/diva2:1014983/FULLTEXT02.pdf>

2.5 Base de datos relacionales

De acuerdo con Benítez (2015), las empresas en la actualidad enfrentan una demanda cada vez mayor en la presentación y tratamiento de datos. Una base de datos relacional es un sistema que gestiona los datos y la estructura de tal manera que, su acceso solo puede realizarse a través de consultas estructuradas.

Las bases de datos tienen la ventaja de que ser independientes a los cambios de código de aplicaciones, previenen la redundancia de información, y protegen el acceso no autorizado de manera concurrente.

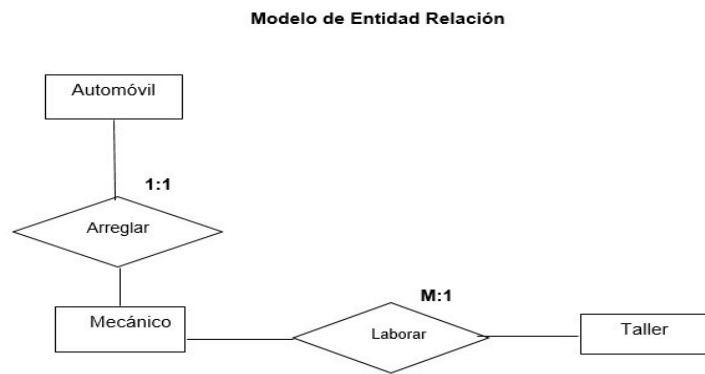


Figura 6: Modelo Entidad Relación.

Obtenido de: (https://commons.wikimedia.org/wiki/File:Modelo_de_Entidad_Relaci%C3%B3n.jpg)

2.5.1 Base de datos ventajas e inconvenientes

De acuerdo con Cabello (2010), las bases de datos contribuyen a mejorar la calidad de las prestaciones de los sistemas informáticos y a maximizar su rendimiento.

Ventajas

- **Independencia de los datos y los programas y procesos:** Esto posibilita la modificación de los datos sin necesidad de alterar el código de las aplicaciones.
- **Menor redundancia:** Esto permite cambiar los datos sin tener que modificar el código de las aplicaciones.
- **Integridad de los datos:** Mayor dificultad para perder los datos o generar incoherencias en los mismos.
- **Coherencia de los resultados:** Al recopilarse y almacenarse la información una sola vez, en los procesos se emplean siempre los mismos datos, lo que garantiza la coherencia de los resultados.
- **Mayor seguridad de los datos:** Al permitir limitar el acceso a los usuarios, se puede establecer que cada tipo de usuario tenga acceso solo a las funcionalidades o datos específicos correspondientes a su perfil.
- **Datos más documentados:** Gracias a los metadatos, que facilitan la descripción de la información en la base de datos, se puede gestionar y organizar mejor los datos almacenados.
- **Acceso a los datos más eficiente:** La correcta organización de los datos genera un rendimiento más óptimo en el sistema.

- **Reducción del espacio de almacenamiento:** Gracias a una mejor estructuración de los datos, se logra un rendimiento más eficiente en el sistema.
- **Acceso simultáneo a los datos:** Es más sencillo gestionar el acceso de usuarios de manera concurrente.

Desventajas

En un sistema de gestión de base de datos se presentan las siguientes desventajas:

- **Instalación costosa:** La administración y el control de bases de datos requieren de un software y hardware de alto costo.
- **Requiere personal cualificado:** Debido a la complejidad en la gestión de este tipo de sistemas.
- **Implantación larga y difícil:** Debido a los puntos mencionados anteriormente, la adaptación del personal se vuelve más compleja y requiere un tiempo considerable.
- **Ausencia de estándares reales:** Esto significa que exista una dependencia excesiva de los sistemas comerciales disponibles en el mercado. Sin embargo, en la actualidad, gran parte de esta tecnología se ha consolidado como un estándar de facto.
- **Falta de rentabilidad a corto plazo:** Debido al costo del personal y los equipos, además del tiempo que se requiere para que esté operativa.

2.6 Sistema gestor de base de datos MySQL

Según Cobo et al. (2005), MySQL es un sistema de gestión de bases de datos relacionales rápido, robusto y flexible. Es ideal para desarrollar bases de datos con acceso desde páginas web dinámicas, crear sistemas de transacciones en línea o cualquier otra solución profesional que implique el almacenamiento de datos, permitiendo realizar consultas múltiples y rápidas.

2.6.1.1 Características de MySQL

Cobo et al. (2005) afirman que, MySQL utiliza un lenguaje estructurado para ejecutar las consultas y acceder a datos relacionados, su arquitectura se basa en cliente/servidor, permitiendo el trabajo multi usuario y procesamiento múltiple. MySQL también destaca por su portabilidad, significa que, está disponible en varias plataformas y distribuciones para Linux. La mayoría de aplicaciones web utilizan MySQL por estas razones, además de tener un soporte amplio y una comunidad de desarrollo en constante crecimiento.

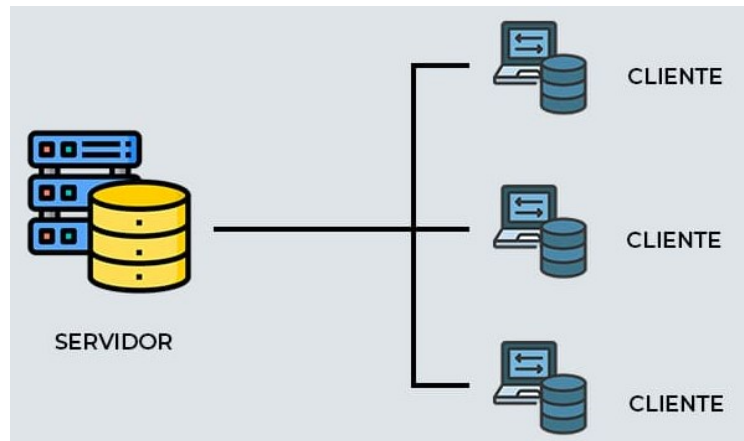


Figura 7: MySQL se basa en el modelo cliente-servidor
Obtenido de: <https://raiolanetworks.com/blog/mysql-que-es-y-para-que-sirve/>

2.6.1.2 Ventajas de MySQL

De acuerdo a lo señalado por Fossati (2014) las ventajas de MySQL son las siguientes:

- Dado que utiliza el modelo cliente-servidor, permite realizar copias de seguridad sin necesidad de cerrar los objetos bloqueados por los usuarios.
- Mientras que Access tiene un límite de 4 GB por tabla, MySQL puede manejar hasta 1 Terabyte.
- Es compatible con múltiples plataformas.
- Existen numerosos ejemplos en línea de cómo acceder a la base de datos utilizando JSP o Servlets.
- Su configuración es bastante sencilla.
- Maximiza el rendimiento de los sistemas multiprocesador gracias a su implementación de multihilos, lo que la convierte en una base de datos muy rápida.
- Soporta una amplia variedad de tipos de datos en las columnas.
- Ofrece APIs para una gran cantidad de lenguajes de programación (PHP, C++, PERL, JAVA, entre otros).
- Tiene una excelente compatibilidad entre diferentes sistemas.
- Admite hasta 32 índices por tabla.
- Cuenta con un robusto sistema de gestión de usuarios y contraseñas, garantizando un alto nivel de seguridad en los datos.

2.5.2 MySQL y Laravel

Las aplicaciones realizadas con el framework Laravel tienen la particularidad de crear una base de datos al ejecutar las migraciones, no obstante, siempre se recomienda iniciar las primeras configuraciones de manera manual. Una base de datos MySQL tiene muchos gestores para administración de datos, por ejemplo, PhPMyAdmin, DBeaver, el desarrollador es libre de escoger el gestor que mejor se adapte a sus necesidades (Herd Services, s.f.).

La conexión entre Laravel y MySQL se controla desde los archivos que manejan las variables de entorno de la aplicación, por lo general están nombrados como ‘.env’ o ‘production.env’. Estos archivos contienen información confidencial y deben ser protegidos, esta es una configuración básica de la conexión entre Laravel y MySQL:

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3307
DB_DATABASE=laravel # el mismo nombre en la base de datos
DB_USERNAME=root
DB_PASSWORD=
```

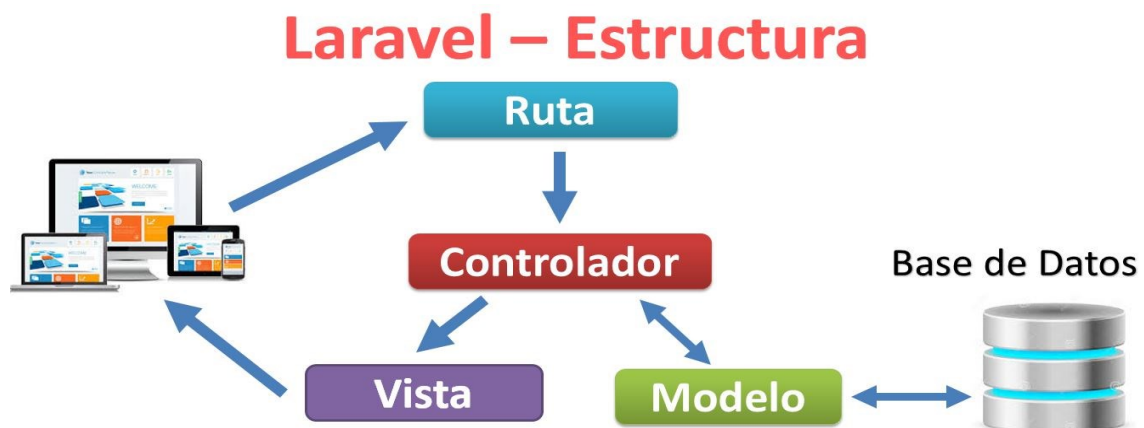


Figura 8: Estructura de Laravel.

Obtenido de: <https://www.incanatoit.com/2016/07/sistema-web-php-laravel-mysql-rutas-modelo.html>

2.7 Apache Jmeter

Según el autor Gillis (2024), Jmeter es un software open source utilizado para pruebas de carga y estrés en sitios web. Las pruebas simulan como se comportará la aplicación web cuando es sometida a visitas de múltiples usuarios al mismo tiempo. La interfaz de Jmeter es muy similar a la de un navegador web, su comportamiento es muy similar, sin embargo, esta herramienta también incluye un CLI para un control más eficiente.

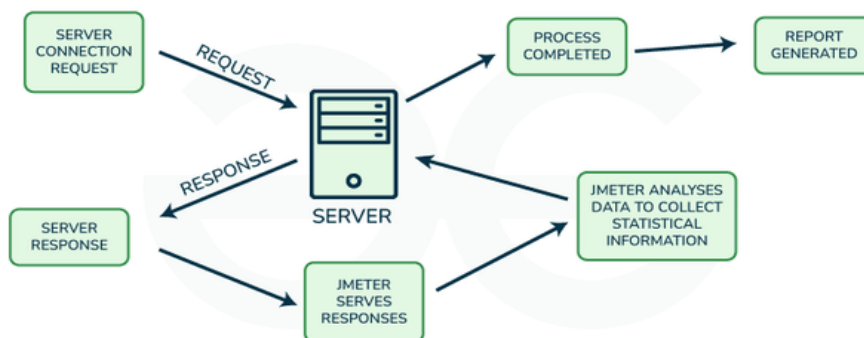


Figura 9: Características de Apache Jmeter.

Obtenido de: <https://www.geeksforgeeks.org/apache-jmeter-an-introduction/>.

2.8 API REST y RestFul

API. - Se han convertido en una forma estándar y omnipresente de intercambiar datos en la nube, en plataformas SaaS y entre los sistemas internos de las empresas. Aunque existen diversos tipos de interfaces API en la web, las que se desarrollan utilizando HTTP son las más relevantes, ya que es el tipo más utilizado y el más común en la web actual (Reis & Housley, 2022).

REST. - Basado en Reis & Housley (2022), REST (Transferencia de Estado Representacional) es el paradigma de API más utilizado en la actualidad. REST, fue sugerido por Roy Fielding en su tesis doctoral en el año 2000. Este conjunto de principios y filosofías se centra en el uso de los verbos HTTP, como **GET** y **PUT**. En la práctica, REST moderno emplea solo un número reducido de los mapeos de verbos que Fielding describió en su tesis original.

API REST. - De acuerdo a, (Yoris, s.a), una API REST es simplemente una API que sigue los principios de diseño de REST y emplea las solicitudes HTTP (GET, POST, PUT, PATCH, DELETE) para la manipulación y gestión de datos.

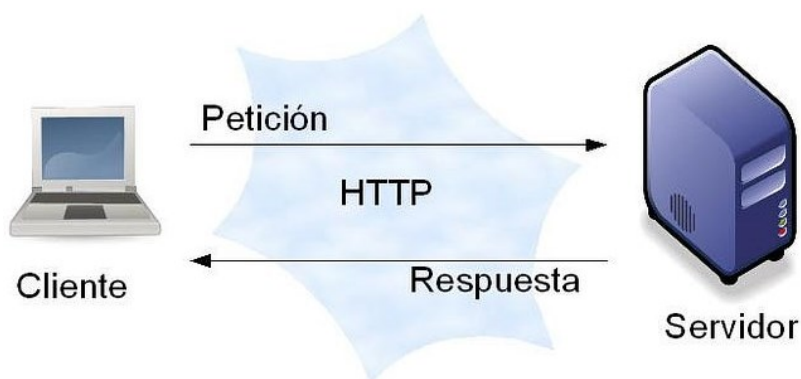


Figura 10: HTTP (Protocolo de Transferencia de Hipertextos)

Obtenido de: <https://medium.com/@germancutraro/servidor-http-node-js-4725004bf386>

2.9 Metodología ágil para el desarrollo de software

De acuerdo con (Porras, s.a), el desarrollo de software sigue un ciclo de vida que comienza cuando surge una necesidad, se definen los requisitos y se realiza la especificación correspondiente. Cada etapa del ciclo de vida consta de diversas actividades, y cada actividad está formada por tareas que son llevadas a cabo por personas que desempeñan roles específicos. Estas personas tienen la responsabilidad de entregar un artefacto en un tiempo determinado.



Figura 11: Modelos de proceso de desarrollo de software.

Obtenido de:

https://www.google.com.ec/books/edition/Metodolog%C3%ADas_%C3%A1giles_para_el_desarrollo/JfXBEAAQBAJ?hl=es&gbpv=1&dq=metodologia+de+desarrollo+de+software&printsec=frontcover

2.10 Metodología ágil Kanban

Según Anzuales & Moya (2024), la metodología ágil Kanban se basa en el uso de tableros dinámicos que organizan el trabajo del proyecto en tres columnas: tareas pendientes, tareas en progreso y tareas completadas. La visualización del trabajo en curso es un aspecto clave en Kanban, ya que permite a los equipos obtener una visión clara de sus actividades actuales.

La implementación de la metodología ágil Kanban en equipos de desarrollo de software tiene el potencial de optimizar significativamente la comunicación, transparencia, y colaboración en equipo. Además, su adopción ha demostrado incrementar la eficiencia y productividad al reducir el tiempo dedicado a tareas no productivas.

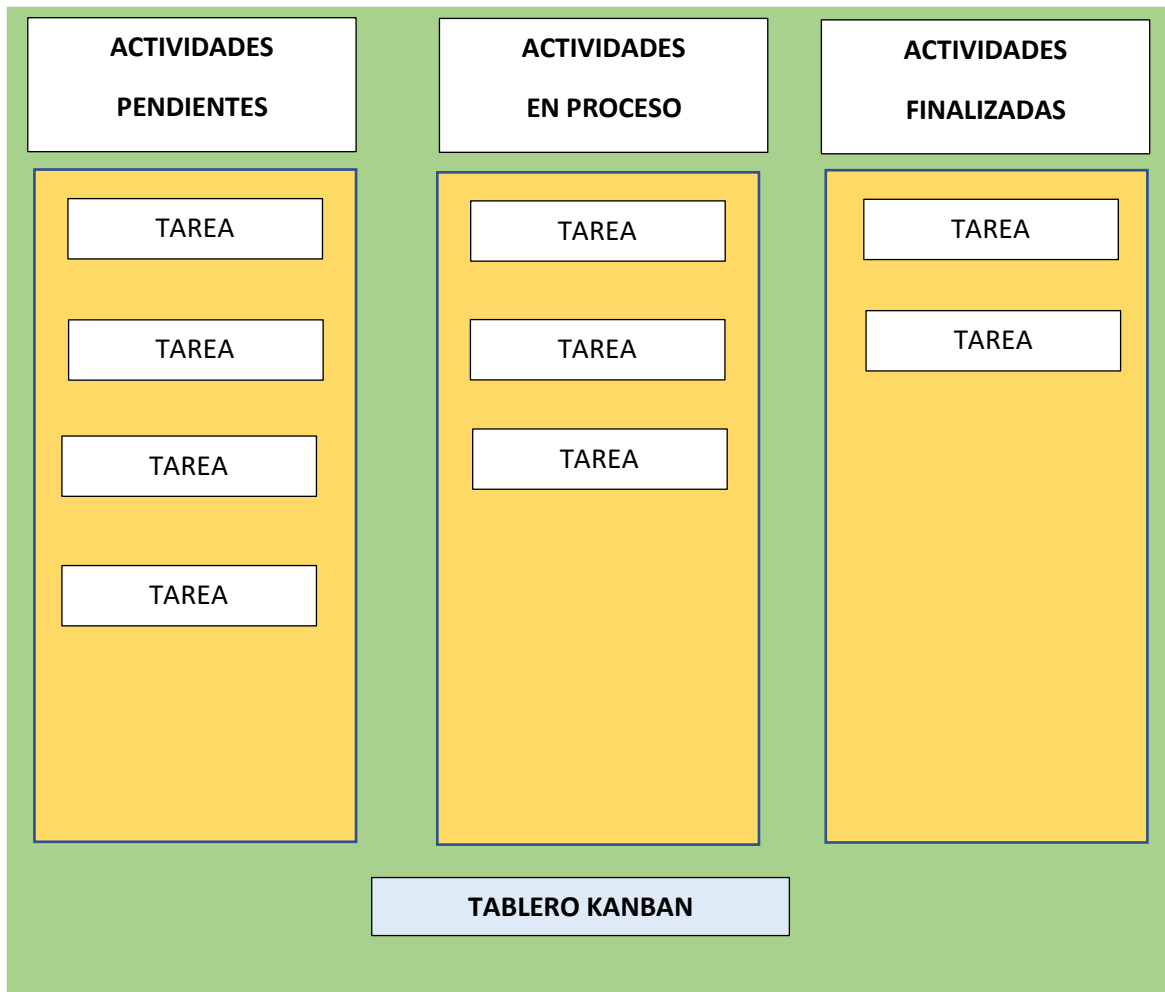


Figura 12: Tablero Kanban.

2.11 Rendimiento

Las pruebas de rendimiento con Jmeter evalúan el comportamiento y la eficiencia de una aplicación web, bajo todo tipo de carga. Se muestran los diferentes tipos de pruebas de rendimiento:

- **Prueba de carga:** evalúa el comportamiento del aplicativo con un número de usuarios de forma creciente.
- **Prueba de estrés:** comprueba el rendimiento del sistema en condiciones extremas.
- **Prueba de escalabilidad:** verifica que el sistema maneje un aumento progresivo de usuarios.
- **Prueba de resistencia:** mide el rendimiento del sistema en periodos prolongados de actividad (Alvarez, 2014).

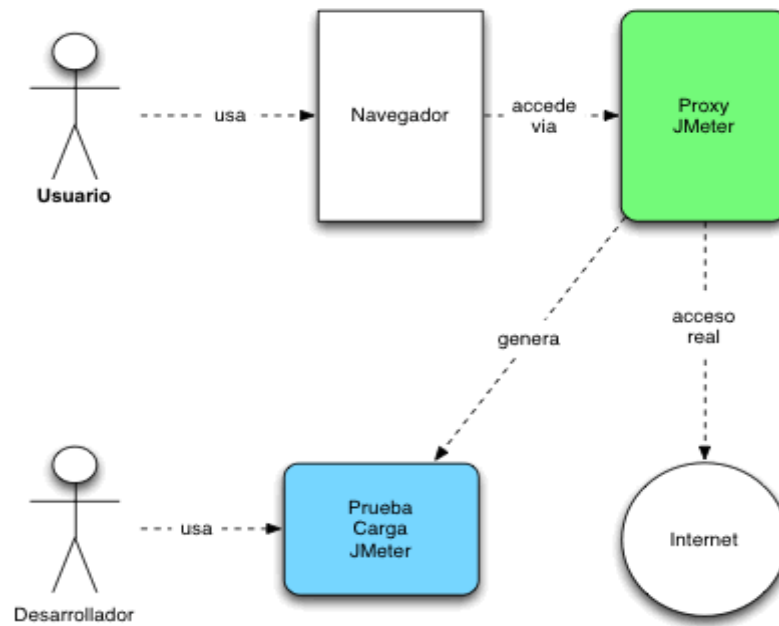


Figura 13: Esquema de funcionamiento de Jmeter.
Obtenido de: <https://www.arquitecturajava.com/jmeter-test/>

CAPITULO III. METODOLOGÍA

La investigación sigue un enfoque cuantitativo, se evaluó el rendimiento con la herramienta Apache Jmeter.

3.1 Tipo y diseño de la investigación

3.1.1 Según el tipo de variable

- **Cuantitativa:** se utilizó para obtener un tiempo promedio de respuesta, métricas de rendimiento, índices de error y resultados en las pruebas de carga y estrés.

3.1.2 Según el objeto de estudio

- **Aplicada:** el desarrollo de la aplicación web con Laravel busca implementar una solución eficiente, escalable y que cumpla con las necesidades operativas de las Pymes.
- **Experimental:** incluye pruebas de rendimiento de la aplicación (tiempos de respuesta, capacidad de carga, escalabilidad), además de probar sus funcionalidades en un entorno real para verificar su funcionamiento y usabilidad.

3.2 Población y muestra

Acorde al tipo de investigación planteada la población es infinita, por lo tanto, se va a utilizar la herramienta Apache Jmeter para evaluar el rendimiento de la aplicación web.

3.3 Identificación de variables

Variable independiente

Aplicación web

Variable dependiente

Rendimiento de la aplicación web

3.4 Operacionalización de variables

Tabla 4: Operacionalización de Variables.

Variable	Tipo	Definición conceptual	Dimensión	Indicadores
Aplicación web	Independiente	Sistema desarrollado con el framework Laravel para gestionar inventarios en Pymes,	Metodología	• Módulos de gestión de inventario. • Generación de reportes.

Rendimiento de la aplicación web	Dependiente	Capacidad de la aplicación web para responder eficientemente a las solicitudes de los usuarios	Rendimiento	• Administración de usuarios y roles • Tiempo promedio de carga de las páginas. • Tiempo de procesamiento de solicitudes.
----------------------------------	-------------	--	-------------	---

3.5 Metodología de desarrollo

El desarrollo de la aplicación web siguió los lineamientos de la metodología ágil Kanban enfocada a las fases de desarrollo de software.

Tabla 5: Metodología Kanban aplicada a la fase de análisis.

Actividades Pendientes	Actividades en Desarrollo	Actividades Terminadas
Diseño de la base de datos	Revisión de la documentación de ELOQUENT ORM	Revisión de la documentación de Laravel
Prueba de la base de datos	Análisis de requerimientos	
Diseño de la interfaz grafica		
Implementación del proyecto con Laravel		
Desarrollo de módulos		
Desarrollo de controladores		
Desarrollo de vistas		
Pruebas de integración		
Desarrollo de Api RESTful		
Pruebas de integración con PostMan		
Pruebas de rendimiento con Apache Jmeter		

3.5.1 Análisis de requerimientos

En esta fase, se identificaron las necesidades del sistema y se definieron los requerimientos funcionales y no funcionales.

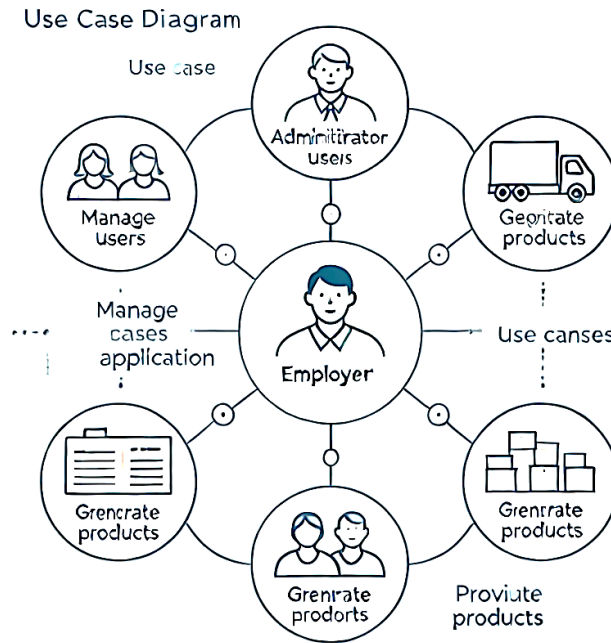


Figura 14: Requerimientos Funcionales.

Requerimientos funcionales

Requerimientos específicos e importantes para el desarrollo de la aplicación web.

Tabla 6: Tabla de requerimientos funcionales.

Ítem	Descripción	Concepto	Prioridad
RF.1	Gestión de usuarios	La aplicación debe gestionar permitir el registro, lectura, actualización y eliminación de usuarios	Alta
RF.2	Gestión de almacenes	La aplicación debe gestionar permitir el registro, lectura, actualización y eliminación de almacenes.	Alta
RF.3	Gestión de categorías	La aplicación debe gestionar permitir el registro, lectura, actualización y eliminación de categorías de productos.	Alta
RF.4	Gestión de productos	La aplicación debe gestionar permitir el registro, lectura, actualización y eliminación de productos.	Alta
RF.5	Generación de reportes	La aplicación debe emitir reportes finales de productos.	Alta
RF.6	Generación de API RESTful	La aplicación debe gestionar permitir que otras aplicaciones accedan a la información a través de un API RESTful.	Alta

Requerimientos no funcionales

Requerimientos que se enfocan en el diseño y seguridad de la aplicación web, pero no comparten una estrecha relación con las funcionalidades.

Tabla 7: Tabla de requerimientos no funcionales.

Ítem	Descripción	Concepto	Prioridad
RNF.1	Usabilidad	Interfaz intuitiva y responsive	Alta
RF.2	Seguridad	Protección de rutas y prevención de autoeliminación	Alta
RF.3	Escalabilidad	Diseño modular para futuras versiones	Alta

3.5.2 Diseño y desarrollo de la aplicación web

Tabla 8: Metodología Kanban aplicada a la fase de diseño.

Actividades Pendientes	Actividades en Desarrollo	Actividades Terminadas
Implementación del proyecto con Laravel	Diseño de la base de datos	Revisión de la documentación de Laravel
Desarrollo de módulos	Prueba de la base de datos	Revisión de la documentación de ELOQUENT ORM
Desarrollo de controladores	Diseño de la interfaz grafica	Análisis de requerimientos
Desarrollo de vistas		
Pruebas de integración		
Desarrollo de Api RESTful		
Pruebas de integración con PostMan		
Pruebas de rendimiento con Apache Jmeter		

Diseño de la base de datos

Se diseñó un modelo relacional normalizado con la herramienta en línea DBDIAGRAM para garantizar la integridad y eficiencia de los datos.

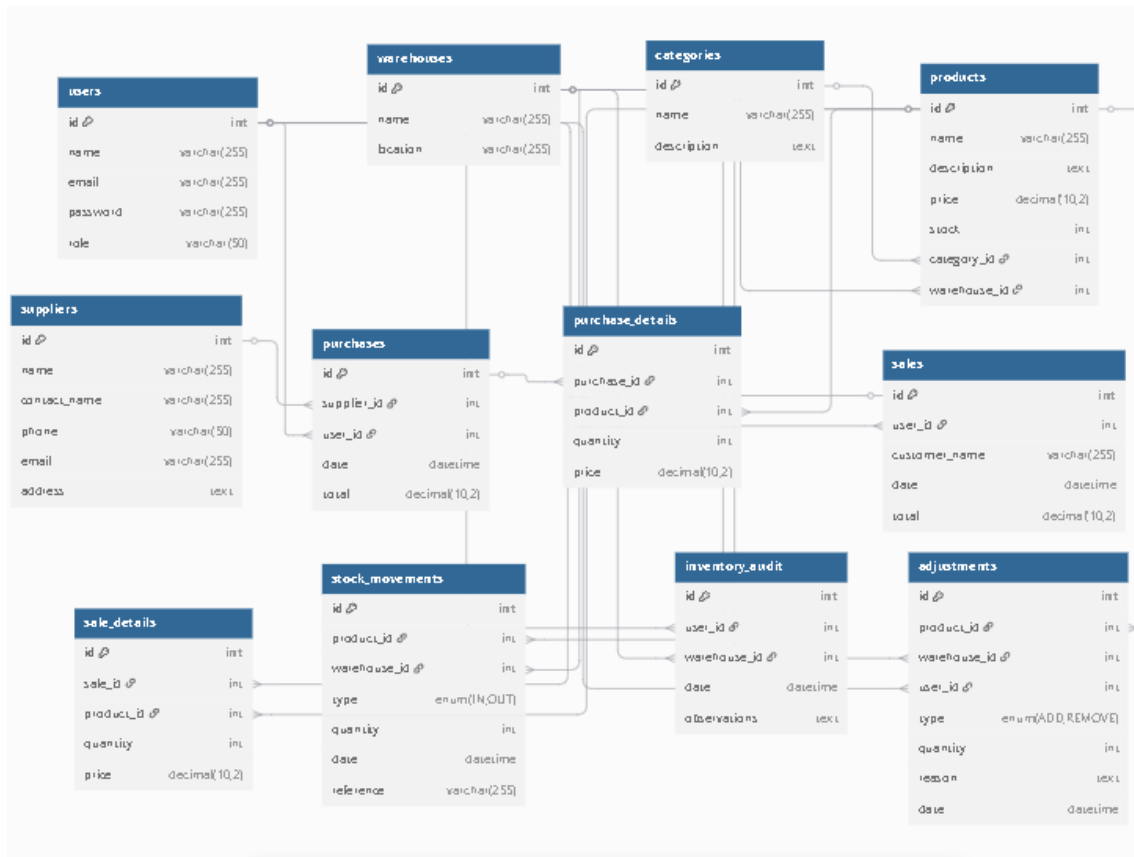


Figura 15: Diagrama Entidad Relación.

Diseño de la interfaz gráfica de usuario

Con la herramienta en línea Pencil, se desarrolló el mockup de la aplicación web, tomando en cuenta que los módulos y vistas que Laravel genera pueden variar con el boceto inicial propuesto.

Vista de Login

Las bibliotecas externas de Laravel permiten generar una vista genérica de un login de ingreso, permite personalización utilizando código PHP junto con etiquetas HTML.

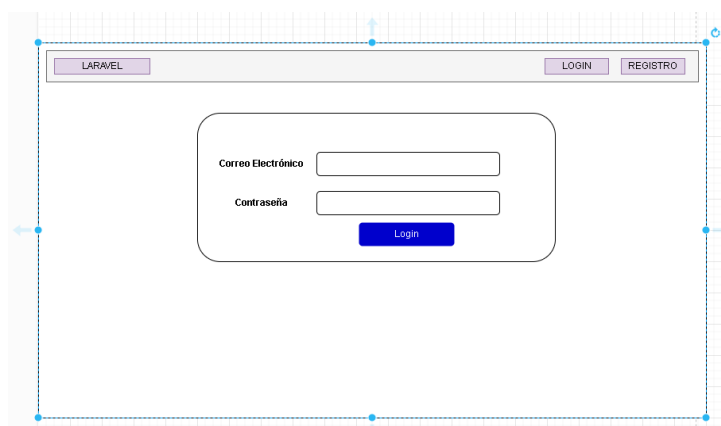


Figura 16: Diseño de Login.

Vista del panel de administración

Es la vista principal del usuario, se muestra gráficos interactivos que resumen las actividades realizadas con la aplicación web. En la barra lateral izquierda se ubican los botones de redireccionamiento a las demás vistas de la aplicación.

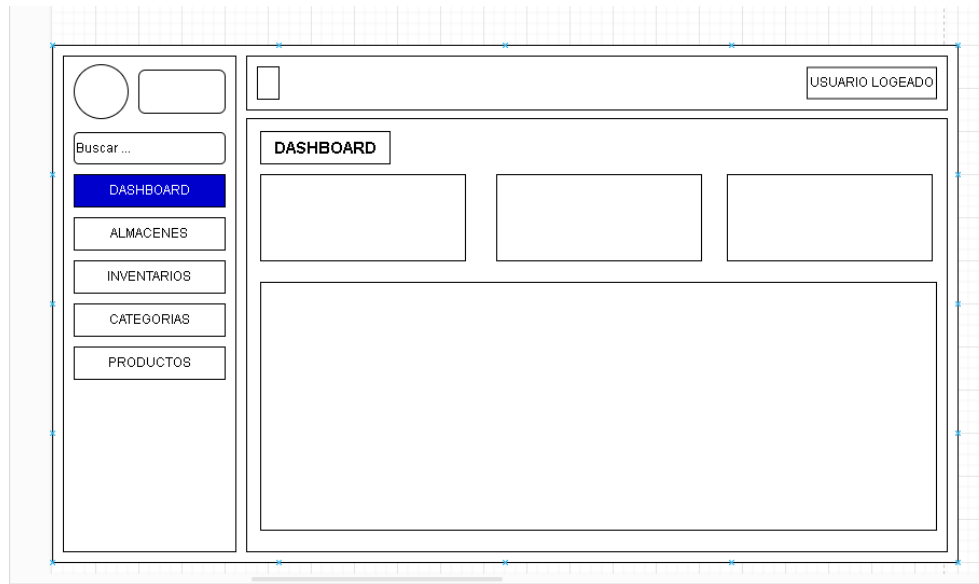


Figura 17: Diseño del panel principal.

Vista formulario de ingreso de productos

Todos los formularios utilizados para el ingreso de datos siguen una misma estructura para reutilización de código. Toda la información pasa por un filtro de JavaScript antes de ser almacenada en la base de datos, por lo que, si existe errores de tipado, el formulario se encarga de validarlo.

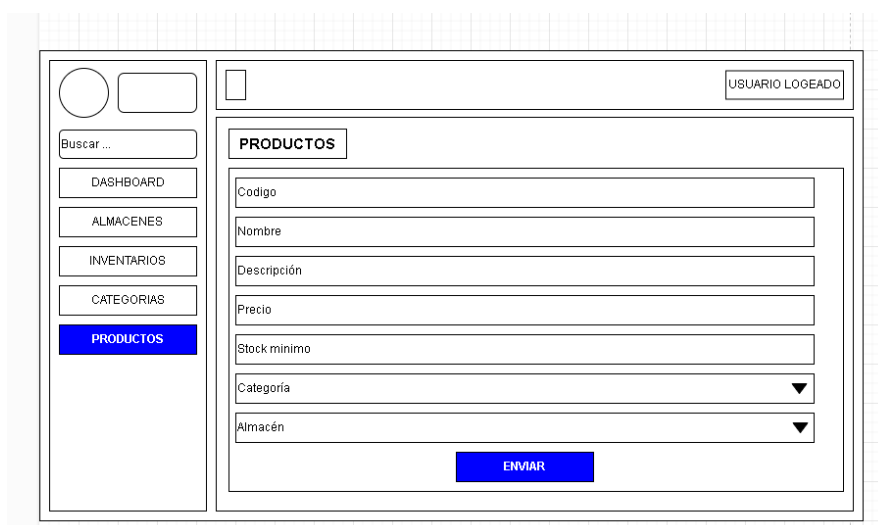


Figura 18: Diseño de formularios de ingreso de datos.

3.5.3 Implementación del proyecto con Laravel

Tabla 9: Metodología Kanban aplicada a la fase de desarrollo.

Actividades Pendientes	Actividades en Desarrollo	Actividades Terminadas
Pruebas de integración	Implementación del proyecto con Laravel	Revisión de la documentación de Laravel
Desarrollo de Api RESTful	Desarrollo de módulos	Revisión de la documentación de ELOQUENT ORM
Pruebas de integración con PostMan	Desarrollo de controladores	Análisis de requerimientos
Pruebas de rendimiento con Apache Jmeter	Desarrollo de vistas	Diseño de la base de datos
		Prueba de la base de datos
		Diseño de la interfaz gráfica
		Diseño de la base de datos

Después de tener un modelo inicial como guía para la construcción de la interfaz, se procedió con la estructuración del proyecto utilizando composer. Desde la terminal de línea de comandos, para este proyecto Bash, se inicia con la descarga y configuración del entorno de trabajo.

Aplicando los comandos `composer create-project --prefer-dist laravel/laravel inventario_manager "10. *"` se procede con la construcción del entorno de la aplicación web.

```
n/Luis Muñoz - Unach/Aplicacion Web (master)
code .

antiago@DESKTOP-E490I23 MINGW64 ~/Desktop/srhs_projects/Trabajos de investigaci
n/Luis Muñoz - Unach/Aplicacion Web (master)
php artisan serve
ould not open input file: artisan

antiago@DESKTOP-E490I23 MINGW64 ~/Desktop/srhs_projects/Trabajos de investigaci
n/Luis Muñoz - Unach/Aplicacion Web (master)
cd inventario_app/

antiago@DESKTOP-E490I23 MINGW64 ~/Desktop/srhs_projects/Trabajos de investigaci
n/Luis Muñoz - Unach/Aplicacion Web/inventario_app (master)
php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Sun Feb 23 20:03:22 2025] PHP 8.2.12 Development Server (http://127.0.0.1:8000)
started
[Sun Feb 23 20:03:33 2025] 127.0.0.1:22276 Accepted
[Sun Feb 23 20:03:34 2025] 127.0.0.1:22276 Closing
[Sun Feb 23 20:03:34 2025] 127.0.0.1:22277 Accepted
[Sun Feb 23 20:03:36 2025] 127.0.0.1:22277 [200]: GET /favicon.ico
[Sun Feb 23 20:03:36 2025] 127.0.0.1:22277 Closing
```

Figura 19: Git bash para construir el entorno.

3.5.4 Desarrollo de módulos, controladores y vistas

Autenticación

- **Instalar Laravel UI:** Laravel UI es un paquete oficial que facilita la generación de vistas y configuraciones iniciales para autenticación.

- **Generar la interfaz de usuario y las vistas de autenticación:** para este proyecto se utilizó las vistas generadas con Blade.
- **Compilar los assets front-end:** Una vez que se generan las vistas, necesitas instalar y compilar los recursos front-end.
- **Configurar la base de datos:** Como se mencionó anteriormente, Laravel incluye comandos de migración de tablas a la base de datos, después de generar los módulos respectivos se procede a ejecutar una migración.
- **Proteger rutas con middleware:** Laravel usa middleware para asegurarse de que solo los usuarios autenticados puedan acceder a ciertas rutas.

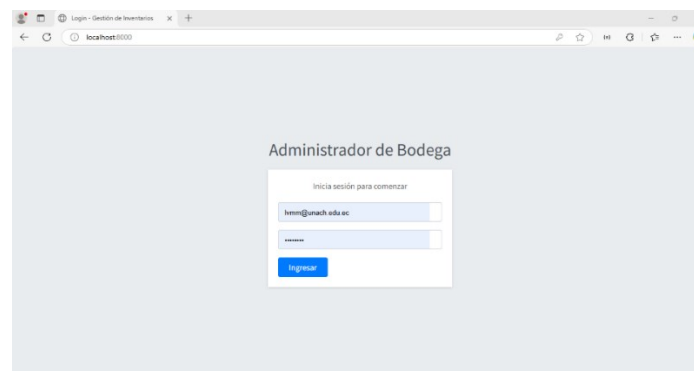


Figura 20: Vista de login de acceso a la aplicación web.

Gestión de usuarios

La aplicación web permite el registro de nuevos usuarios desde el login y desde el panel de administración, al registrarse desde la interfaz del login el usuario es asignado con el rol de vendedor, por defecto.

Vista gestión de usuarios

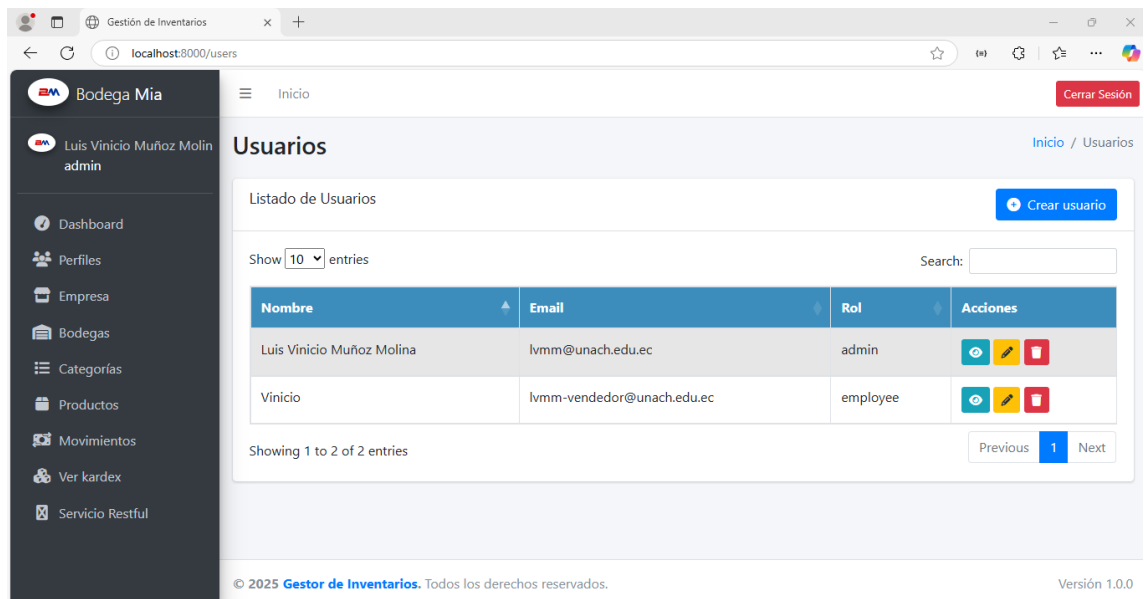


Figura 21: Panel de administración de usuarios.

Vista formulario de creación de usuarios

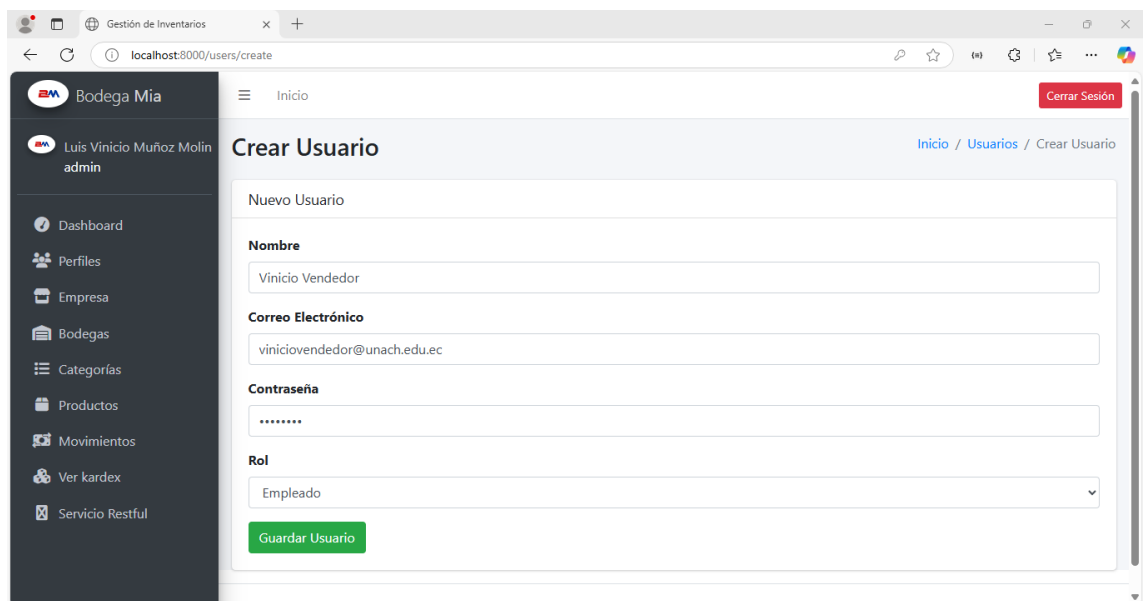


Figura 22: Vista de creación de usuarios.

Vista edición de usuarios

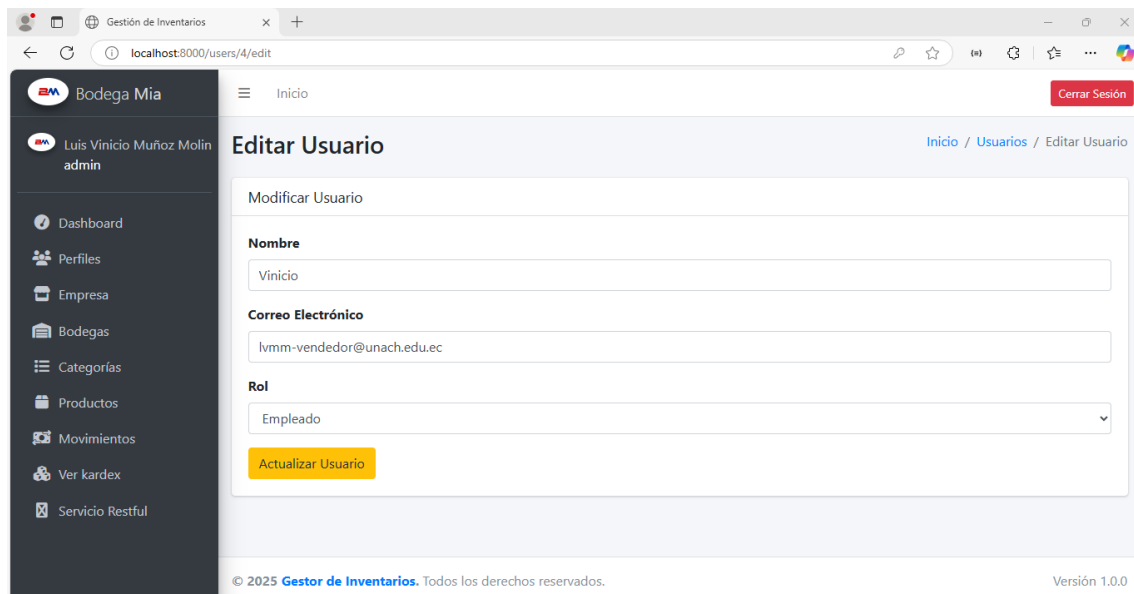


Figura 23: Vista de edición de usuarios.

Vista de usuario único

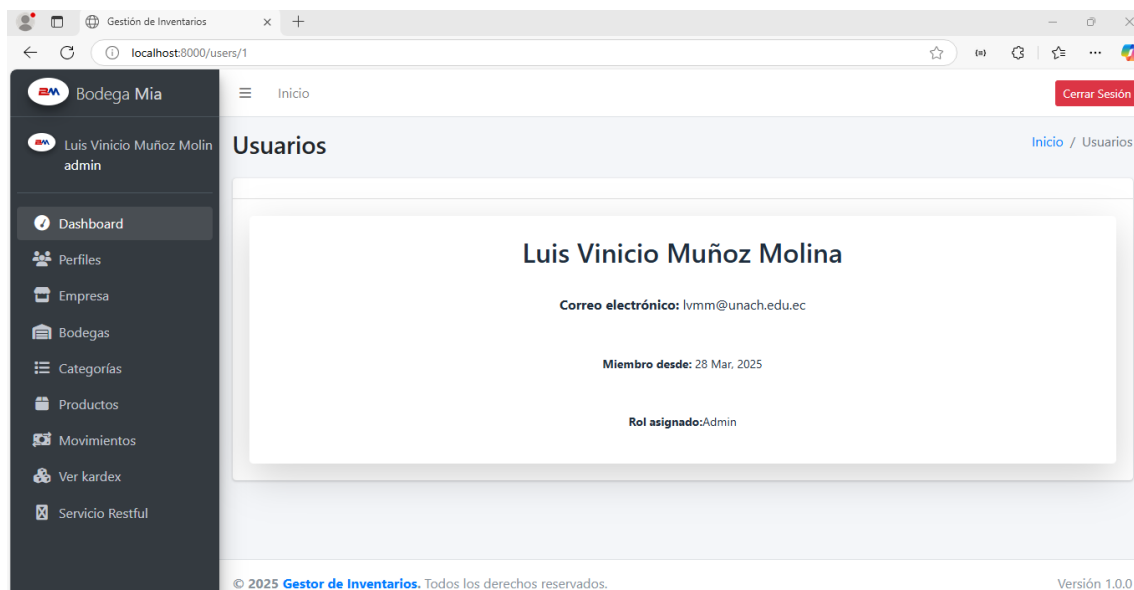


Figura 24: Vista de usuario único.

Cuando un usuario con rol de Administrador ingresa al panel principal podrá observar y tener acceso a todas las funcionalidades del sistema. Los permisos asignados según el rol son:

- **Administrador:** tiene todos los permisos para operaciones CRUD en la aplicación.
- **Vendedor:** tiene permisos de edición y lectura de los módulos de inventarios y almacenes.

Gestión de empresas

La aplicación web está desarrollada para gestionar diferentes almacenes con diferentes productos e inventarios independientes. Esto permite que el usuario pueda gestionar a la vez diferentes tipos de sucursales o negocios independientes.

Vista de listado de empresas

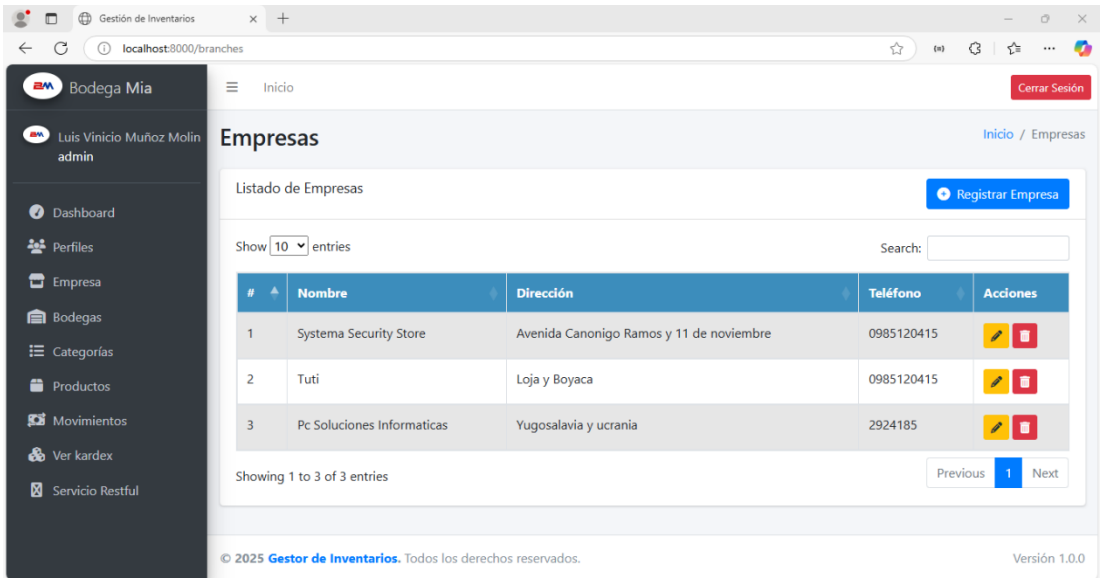


Figura 25: Panel de administración de empresas.

Vista formulario de creación de empresas

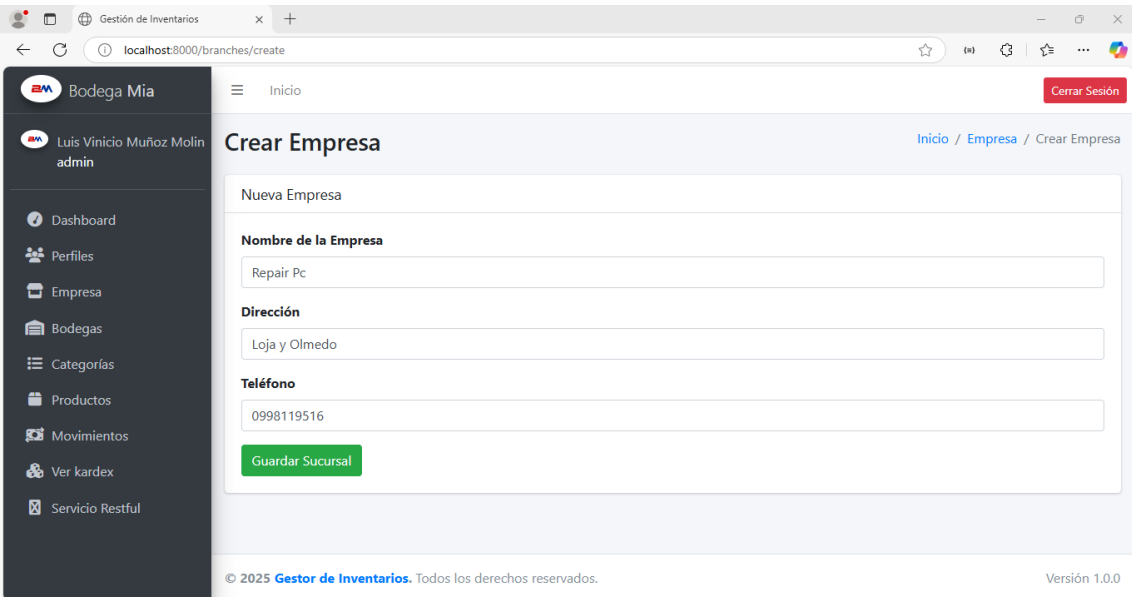


Figura 26: Vista de creación de empresas.

Vista edición de empresas

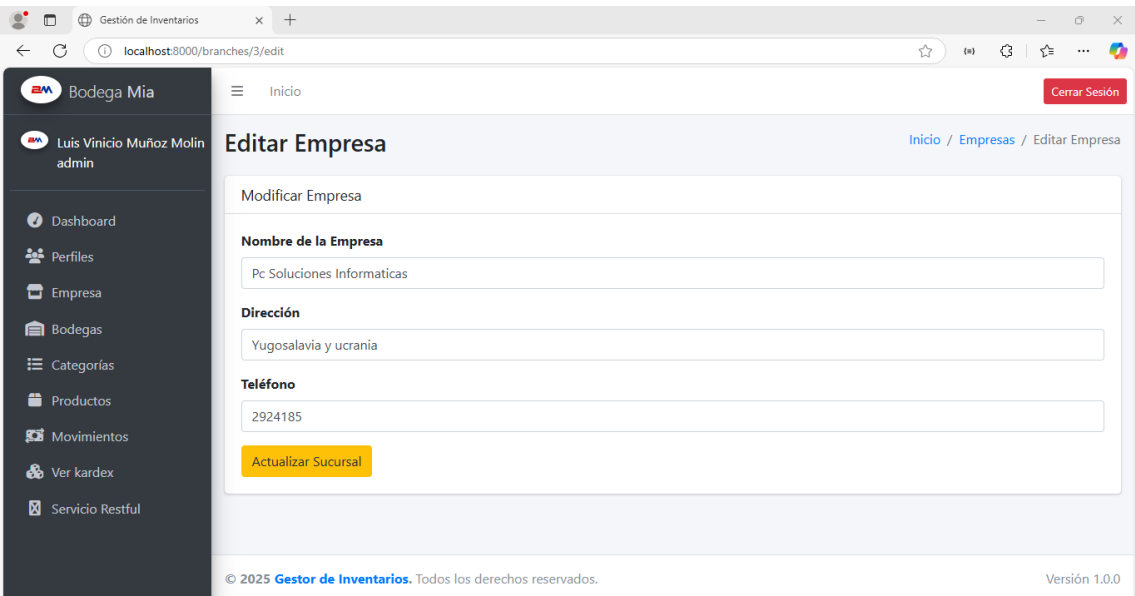


Figura 27: Vista de edición de empresas.

Gestión de categorías

Cada producto del inventario está debidamente categorizado, esto facilita la administración de consultas y filtrado de productos.

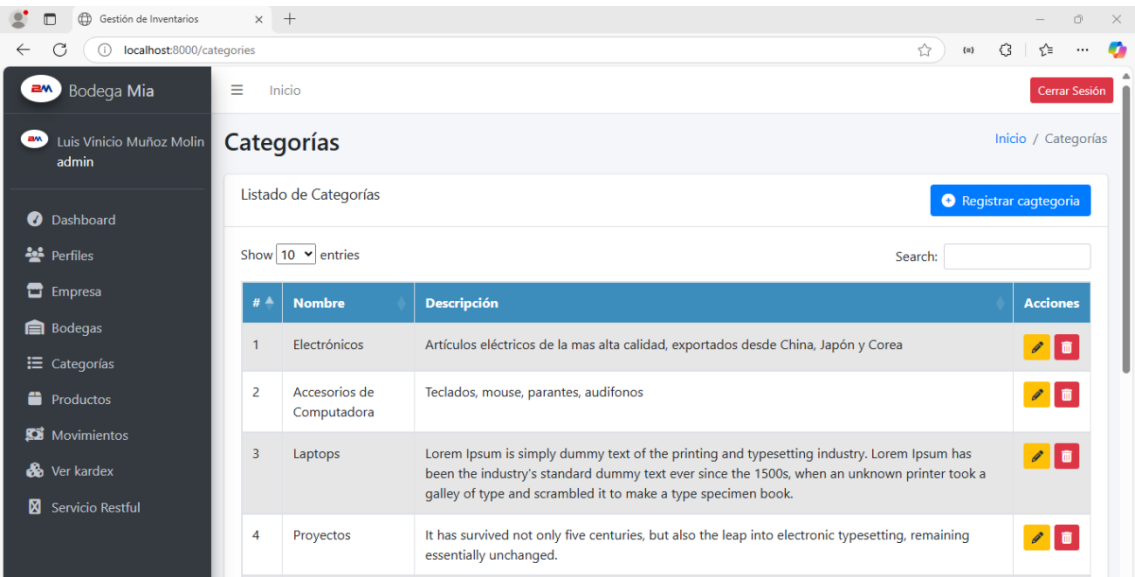


Figura 28: Panel de administración de categorías.

Vista formulario de creación de categorías

The screenshot shows a web application interface for 'Bodega Mia'. The left sidebar contains navigation links: Dashboard, Perfiles, Empresa, Bodegas, Categorías, Productos, Movimientos, Ver kardex, and Servicio Restful. The main content area is titled 'Crear Categoría' and includes a breadcrumb trail: Inicio / Categorías / Crear Categoría. The form itself has a title 'Nueva Categoría' and two input fields: 'Nombre de la Categoría' (containing 'Fuentes de poder') and 'Descripción' (containing 'Fuentes de poder de 500w'). A green 'Guardar Categoría' button is at the bottom of the form. The footer shows '© 2025 Gestor de Inventarios. Todos los derechos reservados.' and 'Versión 1.0.0'.

Figura 29: Vista de formulario de creación de categorías.

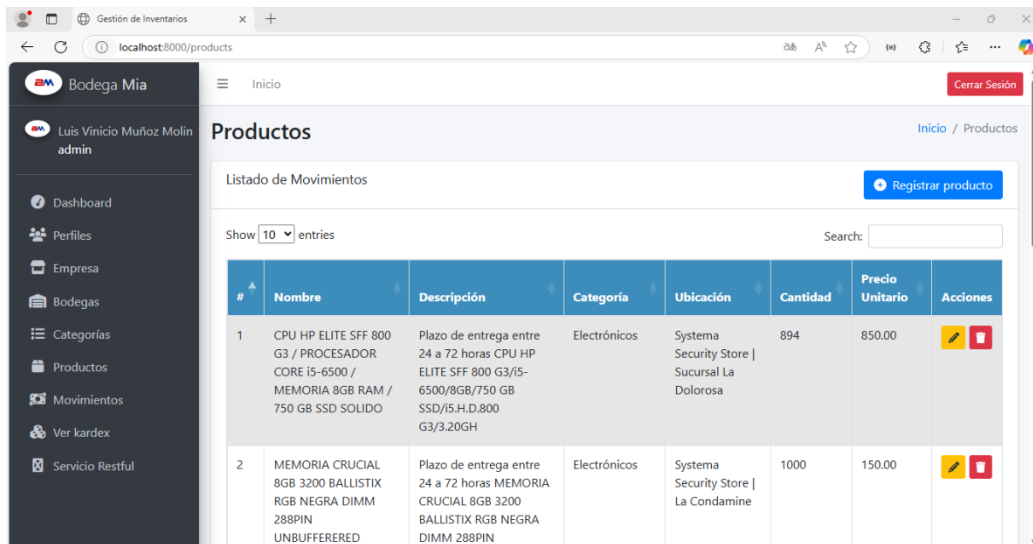
Vista de edición de categorías

The screenshot shows the 'Editar Categoría' (Edit Category) form in the 'Bodega Mia' application. The left sidebar is identical to the previous screenshot. The main content area has a breadcrumb trail: Inicio / Categorías / Editar Categoría. The form is titled 'Modificar Categoría' and contains two input fields: 'Nombre de la Categoría' (containing 'Accesorios de Computadora') and 'Descripción' (containing 'Teclados, mouse, parantes, audifonos'). A yellow 'Actualizar Categoría' button is at the bottom of the form. The footer shows '© 2025 Gestor de Inventarios. Todos los derechos reservados.' and 'Versión 1.0.0'.

Figura 30: Vista de edición de categorías.

Gestión de productos

Los productos que se almacenan en la base de datos llevan un identificador de categoría y un identificador de almacén correspondiente.



	Nombre	Descripción	Categoría	Ubicación	Cantidad	Precio Unitario	Acciones
1	CPU HP ELITE SFF 800 G3 / PROCESADOR CORE i5-6500 / MEMORIA 8GB RAM / 750 GB SSD SOLIDO	Plazo de entrega entre 24 a 72 horas CPU HP ELITE SFF 800 G3/i5-6500/8GB/750 GB SSD/i5.H.D.800 G3/3.20GH	Electrónicos	Systema Security Store Sucursal La Dolorosa	894	850.00	 
2	MEMORIA CRUCIAL 8GB 3200 BALLISTIX RGB NEGRA DIMM 288PIN UNBUFFERED	Plazo de entrega entre 24 a 72 horas MEMORIA CRUCIAL 8GB 3200 BALLISTIX RGB NEGRA DIMM 288PIN	Electrónicos	Systema Security Store La Condamine	1000	150.00	 

Figura 31: Panel de administración de productos.

Vista formulario creación de producto

Para añadir un nuevo producto se realiza a partir del botón ubicado debajo del título principal de la vista.

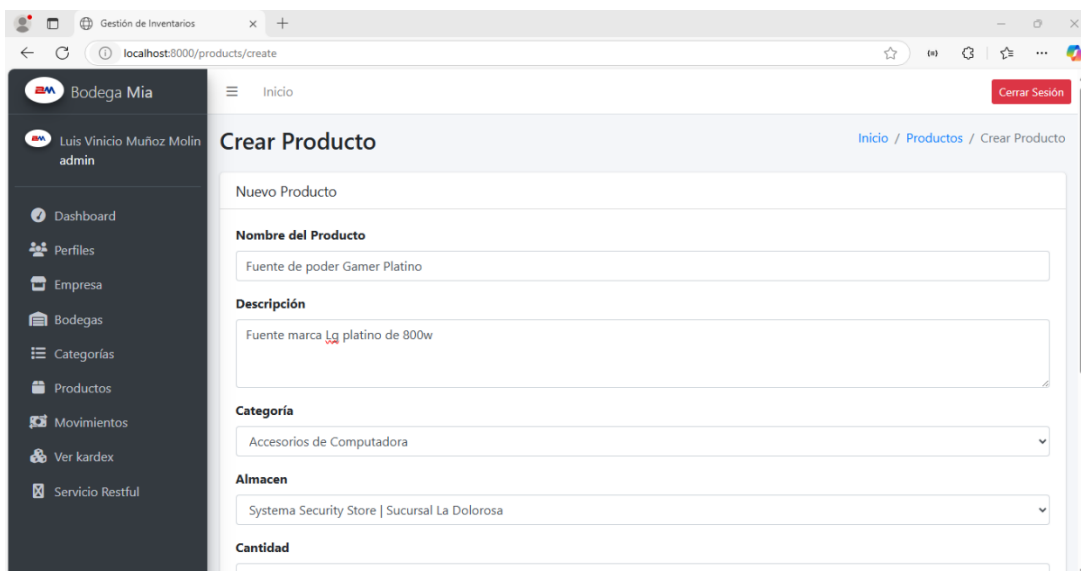


Figura 32: Formulario de ingreso de productos.

Vista de edición de producto

Para editar un producto se redirige a la vista principal del módulo Productos, se selecciona el producto a editar. El formulario de edición captura los datos registrados a través de una consulta asíncrona a la base de datos.

Editar Producto

Modificar Producto

Nombre del Producto

Fuente de poder

Descripción

FUENTE DE PODER QUASAD MINI 850W 24PIN 2SATA P4 110V

Categoría

Accesorios de Computadora

Ubicación

Pc Soluciones | Centro - principal

Cantidad

1

Figura 33: Formulario de actualización del producto.

Vista de gestión de movimientos

Movimientos

Listado de Movimientos

Registrar Movimiento

Show 10 entries

Search:

ID	Tipo	Detalles	Producto	Almacén	Cantidad	Fecha	Usuario	Acciones
1	egreso	Salida de mercadería a Sucursal La Condamine	CPU HP ELITE SFF 800 G3 / PROCESADOR CORE i5-6500 / MEMORIA 8GB RAM / 750 GB SSD SOLIDO	Systema Security Store Sucursal La Dolorosa	40	2025-03-01	Luis Vinicio Muñoz Molina	 
2	ingreso	Vuelve una mercadería	CPU HP ELITE SFF 800 G3 / PROCESADOR CORE i5-6500 / MEMORIA 8GB RAM / 750 GB SSD SOLIDO	Systema Security Store Sucursal La Dolorosa	100	2025-03-02	Luis Vinicio Muñoz Molina	 
3	egreso		COOLER ANTEC T120	Systema	500	2025-	Luis Vinicio	 

Figura 34: Vista de gestión de movimientos.

Ver formulario de creación de movimientos

The screenshot shows a web browser window with the URL `localhost:8000/movements/create`. The page title is "Registrar Movimiento". On the left is a dark sidebar with the "Bodega Mia" logo and a user profile for "Luis Vinicio Muñoz Molina" with the role "admin". The sidebar contains a menu with items: Dashboard, Perfiles, Empresa, Bodegas, Categorías, Productos, Movimientos, Ver kardex, and Servicio Restful. The main content area has a breadcrumb trail: "Inicio / Movimientos / Registrar Movimiento". A red "Cerrar Sesión" button is in the top right. The form itself is titled "Nuevo Movimiento" and contains several fields: "Tipo de Movimiento" (a dropdown menu showing "Ingreso"), "Detalle del movimiento" (a text area containing "Venta de mercadería a una empresa x"), "Producto" (a dropdown menu showing "CPU HP ELITE SFF 800 G3 / PROCESADOR CORE i5-6500 / MEMORIA 8GB RAM / 750 GB SSD SOLIDO | Qty: 894"), "Almacén" (a dropdown menu showing "Systema Security Store | Sucursal La Dolorosa"), and "Responsable" (a dropdown menu showing "Luis Vinicio Muñoz Molina").

Figura 35: Ver formulario de creación de movimientos.

Vista de edición movimientos

The screenshot shows a web browser window with the URL `localhost:8000/movements/1/edit`. The page title is "Editar Movimiento". The sidebar and breadcrumb trail are identical to the previous screenshot. The form is titled "Modificar Movimiento" and contains the same fields as the creation form: "Tipo de Movimiento" (dropdown showing "Egreso"), "Detalle del movimiento" (empty text area), "Producto" (dropdown showing "CPU HP ELITE SFF 800 G3 / PROCESADOR CORE i5-6500 / MEMORIA 8GB RAM / 750 GB SSD SOLIDO"), "Almacén" (dropdown showing "Systema Security Store | Sucursal La Dolorosa"), and "Responsable" (dropdown showing "Luis Vinicio Muñoz Molina").

Figura 36: Vista de edición de movimientos.

Vista de gestión de Kardex

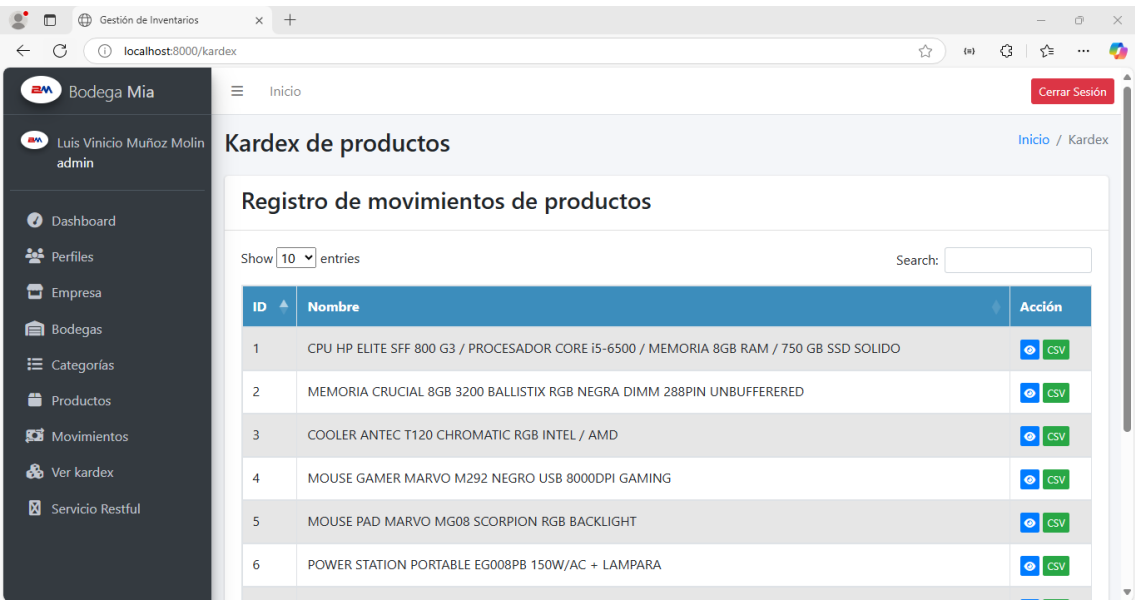


Figura 37: Visita de gestión de Kardex

Vista de detalles del Kardex

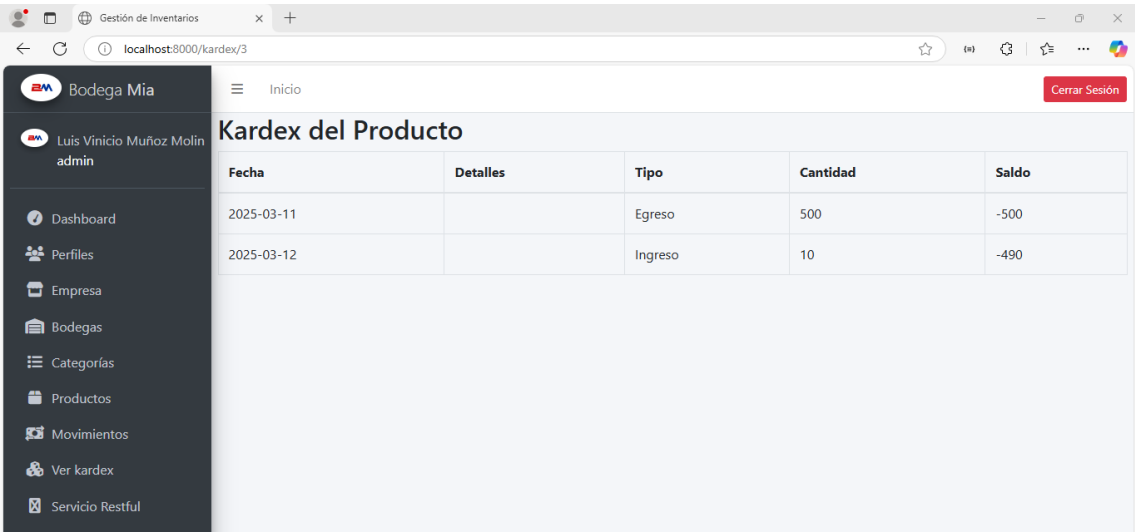


Figura 38: Vista de detalles del Kardex de un producto.

Vista del servicio Restful

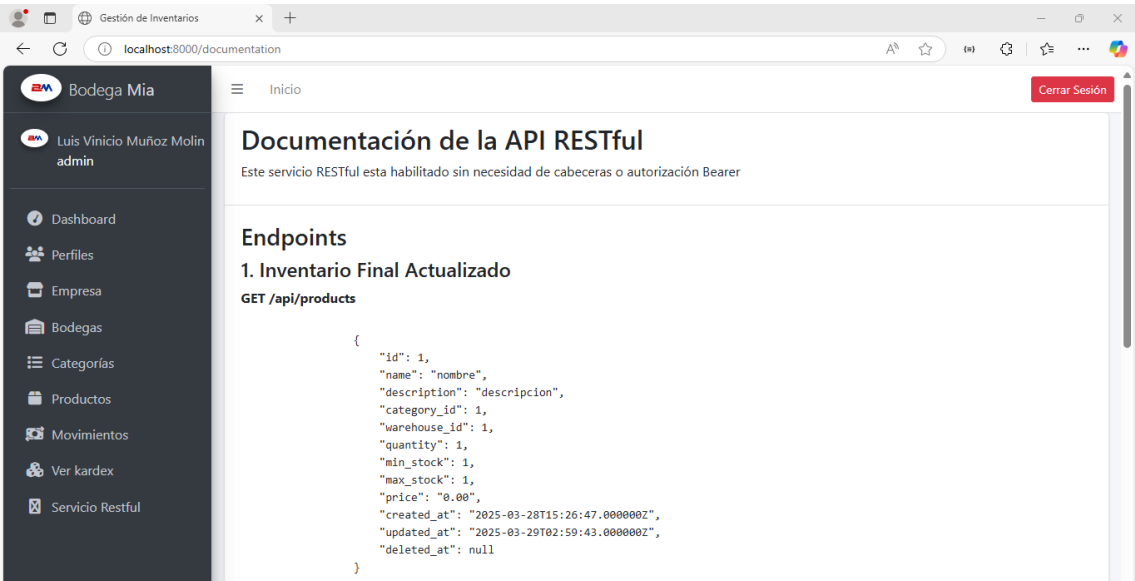


Figura 39: Vista del servicio Restful.

3.5.5 API RESTful agrega la pantalla del api

Se desarrolló un API RESTful que facilite la integración con otros dispositivos, a través de protocolos HTTP. El objetivo es permitir a futuro la integración con aplicaciones empresariales CRM o facilitar la emisión de facturas electrónicas. Con estos protocolos cualquier dispositivo conectado a la red puede realizar peticiones.

Tabla 10: Protocolos para comunicación con API RESTfull.

URL	Protocolo	Método	Descripción
http://localhost:8000/api/products	HTTP	GET	Consulta
http://localhost:8000/api/categories	HTTP	GET	Consulta

```

array [5]
  0 {11}
    id : 1
    name : Botella con recubrimiento de Rubber de 500 ml
    description : Solo en color camel
    price : 12.50
    stock : 1000
    category_id : 1
    warehouse_id : 1
    created_at : 2025-03-23T02:33:57.000000Z
    updated_at : 2025-03-23T02:33:57.000000Z
    category {5}
    warehouse {5}
  1 {11}
    id : 2
    name : Vaso termico para cafe de 250 ml
    description : Colores avena, negro y rojo
    price : 5.25
    stock : 500
    category_id : 1
    warehouse_id : 1
    created_at : 2025-03-23T02:34:57.000000Z
    updated_at : 2025-03-23T02:34:57.000000Z
    category {5}
    warehouse {5}

```

Figura 40: JSON compartido a través de un API RESTful

3.5.6 Pruebas

Una vez concluida la fase de desarrollo se realizó las pruebas de funcionalidad de la aplicación web.

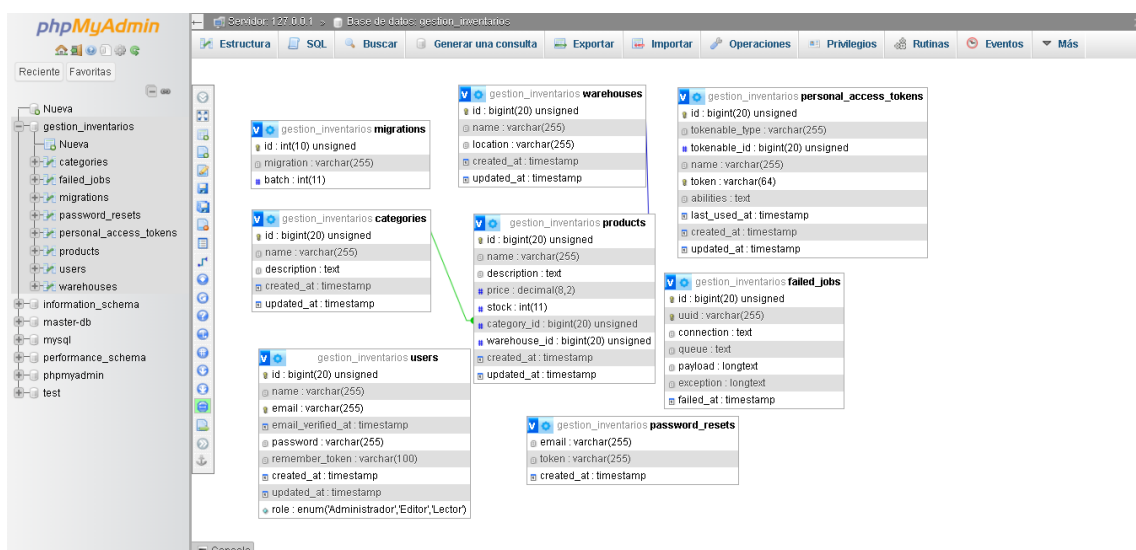


Figura 41: Diagrama relacional de la base de datos en PhpMyAdmin.

Vista general de la aplicación web

Los datos ingresados se presentan en tarjetas visuales para mejor interpretación del usuario.

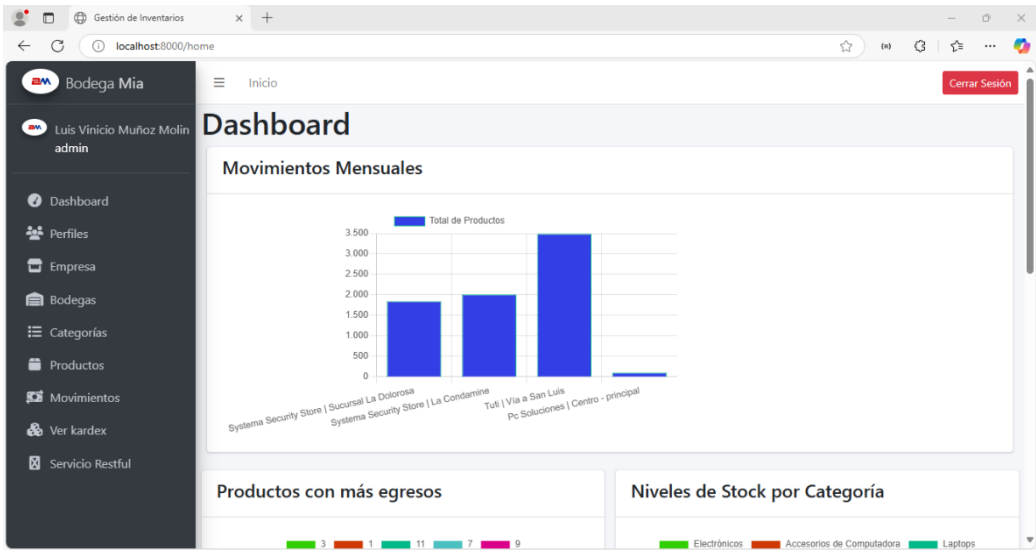


Figura 42: Panel general de la aplicación web.

Tabla 11: Metodología Kanban aplicada a la fase de pruebas

Actividades Pendientes	Actividades en Desarrollo	Actividades Terminadas
Pruebas de rendimiento con Apache Jmeter	Pruebas de integración	Revisión de la documentación de Laravel
	Desarrollo de Api RESTful	Revisión de la documentación de ELOQUENT ORM
	Pruebas de integración con PostMan	Análisis de requerimientos
		Diseño de la base de datos
		Prueba de la base de datos
		Diseño de la interfaz grafica
		Diseño de la base de datos
		Implementación del proyecto con Laravel
		Desarrollo de módulos
		Desarrollo de controladores
		Desarrollo de vistas

3.6 Pruebas de carga y estrés con la herramienta Jmeter

Prueba de carga y estrés con 100 threads

El programa Jmeter sobrecarga el número de peticiones de ingreso al sistema por una cantidad X determinada de usuarios, cabe recalcar que los usuarios están realizando peticiones GET al mismo tiempo. En la Figura 43, se aprecia los resultados obtenidos al sobrecargar el sistema con 100 usuarios, se observa un resultado promedio 6,78% en el

manejo de errores. Cabe recalcar que los resultados pertenecen al desempeño total del sistema, no al desempeño del servidor físicos en donde se ubique la base de datos.

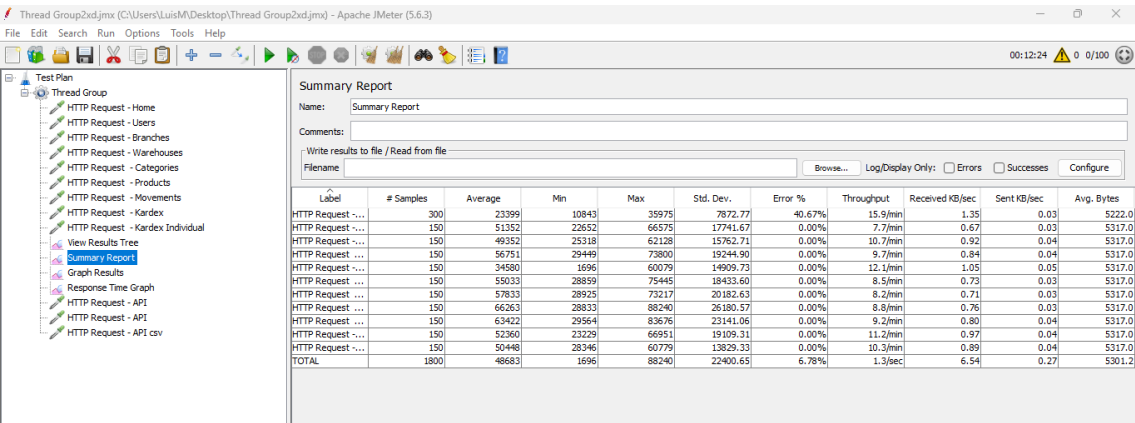


Figura 43: Prueba de carga y estrés con 100 threads.

Gráfica de tiempo de respuesta con 100 thread

En la Figura 44, se muestran los tiempos de respuesta a lo largo del tiempo para varias solicitudes HTTP bajo una carga de 100 threads o hilos, el eje vertical (y) representa los tiempos de respuesta en milisegundos (ms). El eje Horizontal (x) representa el número de muestras o el tiempo transcurrido durante la prueba. A medida que aumentan las muestras (o el tiempo), se observan los tiempos de respuesta en función de la carga y tiempo de ejecución.

Las diferentes lianas de colores representan los tiempos de respuesta de diferentes solicitudes HTTP que se han ejecutado:

La línea de color azul, representa las solicitudes que están en un grupo específico de peticiones, las líneas roja, verde o negra pueden corresponder a diferente tipo de peticiones.

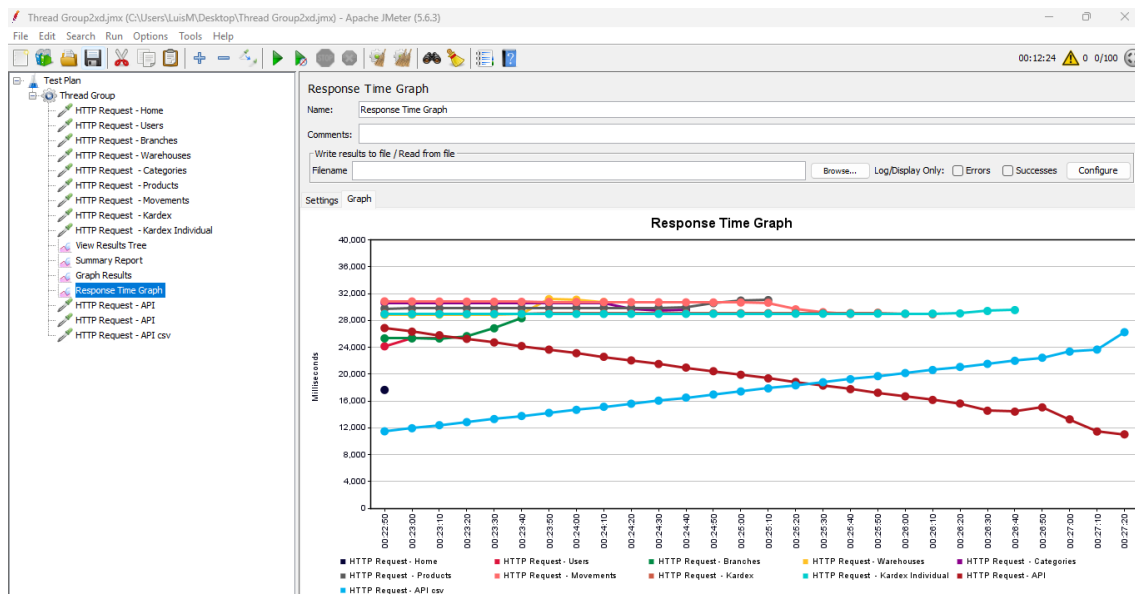


Figura 44: Grafica de tiempo de respuesta con 100 hilos.

CAPÍTULO IV. RESULTADOS Y DISCUSIÓN

4.1 Resultados

Análisis de las pruebas de carga y estrés con 25 threads

Tabla 12: Prueba de carga y estrés con 25 hilos.

Componente	Threads (usuarios)	Media (ms)	Tiempo (Min.)	Tiempo (Max.)	Rendimiento (peticiones/segundo)	Envío (kb/segundo)	Recepción (Kb/segundo)	Error (%)
Http Request – API	2	3523	611	6436	18.6/minuto	0.00	0.84	100.00%
Http Request – Brnaches	25	13906	11612	15659	38.3/minuto	0.15	3.31	0.00%
Http Request – Categories	25	13799	12301	15452	36.2/minuto	0.14	3.13	0.00%
Http Request – Home	25	7797	736	11755	1.5/segundo	0.34	7.84	0.00%
Http Request – Kardex	25	11580	6937	14254	47.3/minuto	0.12	3.74	56.00%
Http Request – Kardex Individual	11	6033	34	14266	34.2/minuto	0.03	1.90	81.82%
Http Request – Movements	25	12442	12195	13629	39.1/minuto	0.15	3.38	0.00%
Http Request – Products	25	12960	12414	14252	39.1/minuto	0.15	3.38	0.00%
Http Request – Users	25	11328	4441	12165	53.6/minuto	0.20	4.64	0.00%
Http Request - Warehouses	25	13789	12277	15717	37.8/minuto	0.15	3.28	0.00%
TOTAL	213	11800	34	15715	2.1/segundo	0.44	10.39	11.74%

Se realizó una prueba inicial con una carga de 25 threads en un periodo de 60 segundos, donde se notó que los endpoints: Branches, Categories y Warehouses mantienen un rendimiento estable entre 36 y 39 peticiones por minuto, con 0% de errores indicando robustez en operaciones de lectura y gestión de datos.

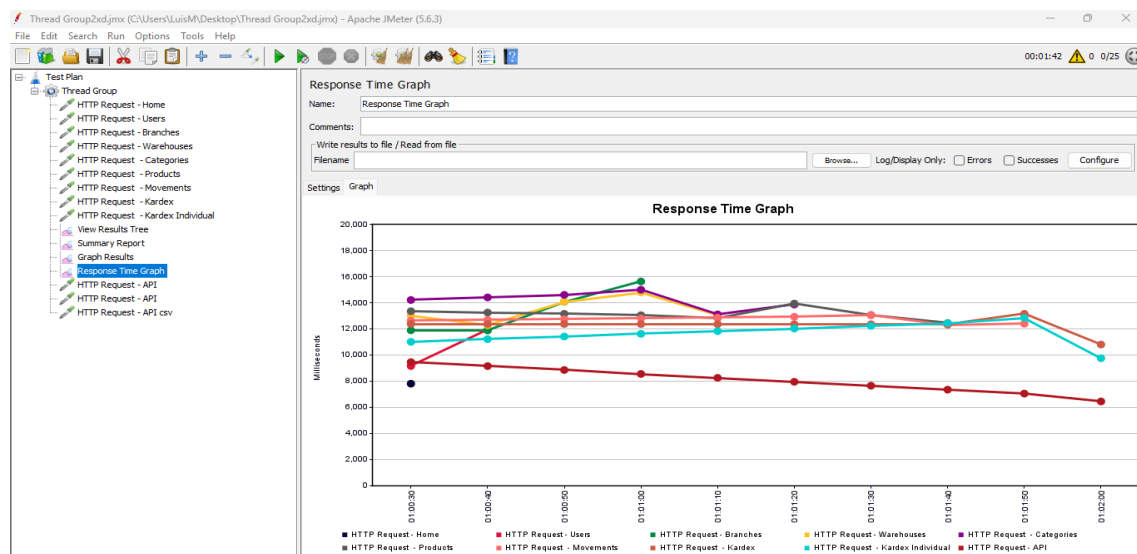


Figura 45: Gráfica de tiempo de respuesta con 25 threads.

También, se registró una baja desviación estándar en los siguientes endpoints: Movements y Products, con una desviación estándar mínima de 333.71 ms y 146.19 ms respectivamente, lo que sugiere tiempos de respuesta predecibles bajo carga.

Finalmente, el endpoint Movements destaca con el menor tiempo promedio de respuesta de 7797 ms y un alto rendimiento de 7.84 KB/s, optimizado para transacciones rápidas.

Análisis de las pruebas de carga y estrés con 40 threads

Tabla 13: Prueba de carga y estrés con 40 hilos.

Componente	Threads (usuarios)	Media (ms)	Tiempo (Min.)	Tiempo (Max.)	Rendimiento (peticiones/segundo)	Envío (kb/segundo)	Recepción (Kb/segundo)	Error (%)
Http Request – API	25	3523	611	6436	18.6/minuto	0.00	0.84	100.00%
Http Request – Brnaches	40	21651	19144	22497	45.4/minuto	0.18	3.93	0.00%
Http Request – Categories	18	6654	10	19612	47.7/minuto	0.03	2.61	94.44%
Http Request – Home	40	17282	1324	23881	1.4/segundo	0.32	7.22	0.00%
Http Request – Kardex	25	11580	6937	14254	47.3/minuto	0.12	3.74	56.00%
Http Request – Kardex Individual	11	6033	34	14266	34.2/minuto	0.03	1.90	81.82%
Http Request – Movements	25	12442	12195	13629	39.1/minuto	0.15	3.38	0.00%
Http Request – Products	1	3031	3031	3031	19.8/minuto	0.89	0.0	100.00%
Http Request – Users	40	19930	16261	22735	50.1/minuto	0.19	4.34	0.00%
Http Request – Warehouses	40	17503	11477	22474	53.9/minuto	0.15	4.40	55.00%
TOTAL	179	17751	10	23881	2.2/segundo	0.43	10.75	22.35%

Para una carga de 40 threads con un periodo de 5 minutos entre cada petición se notó que: los endpoints Branche y Users muestran 0% de errores, con tiempos de respuesta consistentes y una desviación estándar baja de 792.70 ms y 1188.80 ms respectivamente, indicando robustez en operaciones de gestión de datos.

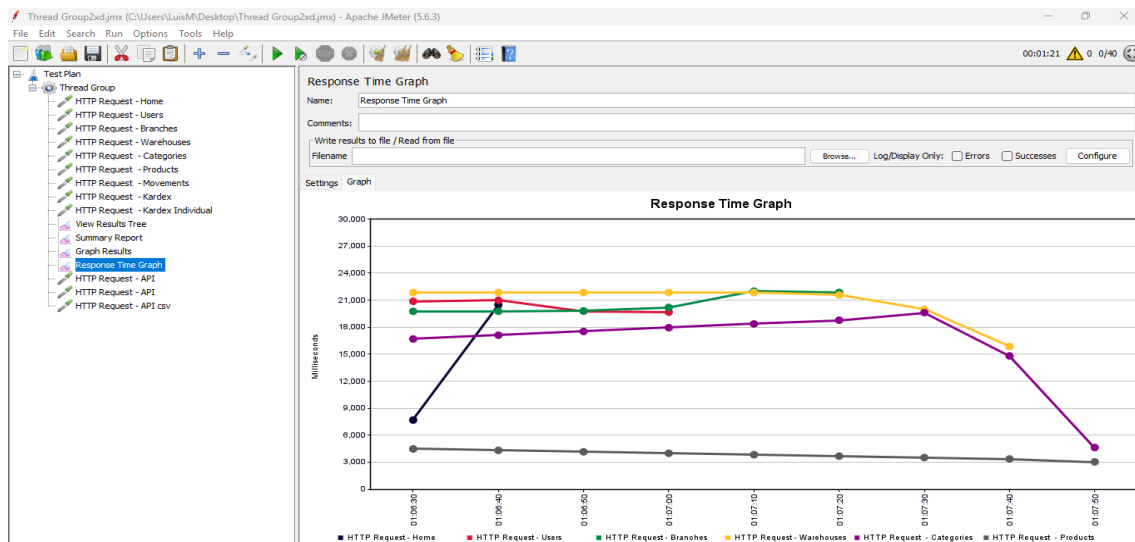


Figura 46: Grafico de tiempo de respuesta con 40 threads.

El resultado del rendimiento con esta carga oscila entre 45 y 50 peticiones por minuto para estos endpoints, adecuado para cargas moderadas.

Análisis de las pruebas de carga y estrés con 100 threads

Tabla 14: Prueba de carga y estrés con 100 hilos.

Componente	Threads (usuarios)	Media (ms)	Tiempo (Min.)	Tiempo (Max.)	Rendimiento (peticiones/segundo)	Envío (kb/segundo)	Recepción (Kb/segundo)	Error (%)
Http Request - API	300	23399	10843	35975	15.9/minuto	0.03	1.35	40.67%
Http Request - Brnaches	150	51352	22652	66575	7.7/minuto	0.03	0.67	0.00%
Http Request - Categories	150	49352	25318	62128	10.7/minuto	0.04	0.92	94.44%
Http Request - Home	150	56751	29449	73800	9.7/segundo	0.04	0.84	0.00%
Http Request - Kardex	150	34580	1696	60079	12.1/minuto	0.05	1.05	56.00%
Http Request - Kardex Individual	150	55033	28859	75445	8.5/minuto	0.03	0.73	81.82%
Http Request - Movements	150	57833	28925	73217	8.2/minuto	0.03	0.71	0.00%
Http Request - Products	150	66263	28833	88240	8.8/minuto	0.03	0.76	100.00%
Http Request - Users	150	63422	29564	83676	9.2/minuto	0.04	0.80	0.00%
Http Request - Warehouses	150	52360	23229	66951	11.2/minuto	0.4	0.97	55.00%
TOTAL	1800	48638	1696	88240	1.3/segundo	0.27	6.54	6.78%

En la prueba con 100 threads durante un lapso de 5 minutos entre cada prueba se obtuvo que, 9 de 11 endpoints muestran 0% de errores incluso bajo carga pesada, demostrando robustez en operaciones críticas.

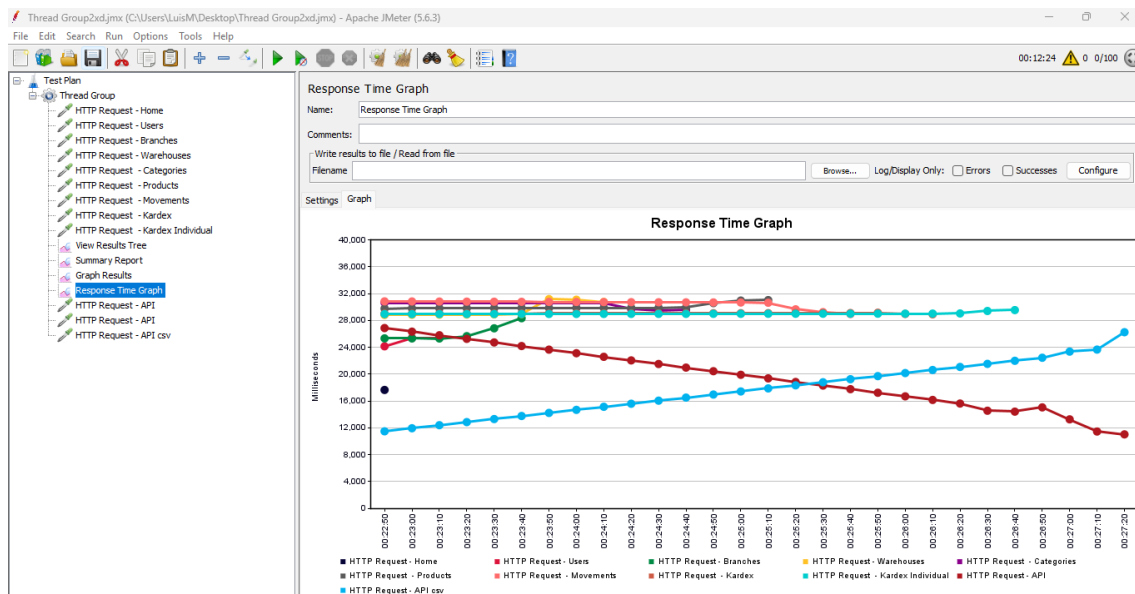


Figura 47: Grafico de tiempo de respuesta con 100 threads.

El rendimiento se mantiene estable con resultados entre 8 y 12 peticiones por minuto en cada endpoint, también se obtuvieron resultados de tiempo promedio de respuesta del servidor entre 56751 ms y 55830 ms, indicando capacidad para manejar cargas sostenidas.

4.2 Discusión

Laravel permitió construir módulos independientes que facilitaron la separación de la lógica de negocio y la lógica de presentación. Esto aseguró una aplicación web bien estructurada y funcional para las necesidades de las Pymes. Además, las pruebas de rendimiento realizadas con la herramienta Apache JMeter revelaron un margen de error de 0% en cargas de hasta 100 peticiones.

En comparación con el artículo publicado por la agencia de marketing online y desarrollo web (Entra en la Red, 2024) se refuerza la percepción de Laravel como una herramienta confiable para el desarrollo de sistemas de inventario. La agencia, con más de 24 años de experiencia en el mercado, destaca que Laravel es ideal para crear sistemas personalizados y automatizar procesos, como la generación de reportes y alertas. Estas cualidades se alinean con los resultados obtenidos en este proyecto, donde se confirma la capacidad del framework para ofrecer estabilidad, facilidad de uso y optimización del control de inventarios.

Según el autor Guevara (2017), Laravel presenta una solución eficiente para las pequeñas y medianas empresas (pymes). Ofrece una estructura bien organizada, funciones de seguridad integradas y herramientas que hacen que el desarrollo de aplicaciones web personalizadas sea mucho más ágil. En una investigación llevada a cabo en la empresa Cuenca Llantas, implementan un sistema web desarrollado con Laravel, permitió automatizar los procesos manuales de gestión de inventario, reducir errores operativos y generar reportes confiables que facilitaron la toma de decisiones. Gracias a estas mejoras, la empresa pudo optimizar su productividad y control operativo sin tener que hacer grandes inversiones en infraestructura tecnológica. Este caso ilustra que Laravel es una opción viable para las pymes que buscan soluciones tecnológicas escalables, seguras y económicas para gestionar sus procesos internos.

En conclusión, Laravel es una herramienta adecuada para la gestión de inventarios. Su flexibilidad, estabilidad y capacidad para manejar procesos complejos lo convierten en una opción fiable para Pymes que buscan mejorar su operatividad mediante soluciones tecnológicas avanzadas.

5 CAPITULO V. CONCLUSIONES y RECOMENDACIONES

5.1 CONCLUSIONES

Se concluye que el framework Laravel es una herramienta adecuada para el desarrollo de aplicaciones orientadas a la gestión de inventarios. Su capacidad para gestionar consultas básicas de forma estable y eficiente, y con una cantidad mínima de errores, fue respaldada por pruebas realizadas con herramientas como Apache JMeter. El rendimiento general en operaciones comunes demostró ser sólido, con tiempos de respuesta entre 2 y 12 segundos, una solución fiable para aplicaciones de este tipo.

Se desarrollaron de manera independiente los módulos, controladores y vistas, cumpliendo con el propósito de gestionar la transaccionalidad y el manejo de peticiones de manera eficiente. Cada módulo fue diseñado para desempeñar una función específica, lo que permitió evitar la sobrecarga de funcionalidades y garantizar una separación clara entre la lógica de negocio y la lógica de presentación.

Las pruebas de rendimiento de la aplicación web desarrollada con Laravel mostraron resultados satisfactorios, con un margen de error de 0% para en cargas masivas de hasta 100 peticiones en lapsos de 60 segundos.

5.2 RECOMENDACIONES

Se recomienda utilizar el framework Laravel como base para el desarrollo de aplicaciones orientadas a la gestión de inventarios, considerando su rendimiento estable y eficiente en consultas básicas. Adicionalmente, se sugiere explorar herramientas complementarias de Laravel, como paquetes de optimización y caching, para garantizar un desempeño superior en operaciones más complejas.

Mantener la estructura modular en el desarrollo de futuras aplicaciones, asegurando una separación clara entre la lógica de negocio y la lógica de presentación. Esta estrategia debe implementarse junto con una documentación detallada para facilitar el mantenimiento y futuras actualizaciones del sistema, garantizando una gestión organizada y eficiente.

Se sugiere realizar pruebas de carga adicionales y ajustes en la arquitectura para mejorar la escalabilidad del sistema. Es importante optimizar las consultas SQL y explorar mejoras en el diseño de la infraestructura tecnológica, con el objetivo de garantizar un rendimiento consistente incluso en escenarios de alta demanda y cargas extremas.

BIBLIOGRAFÍA

- 10Code Software Design. (2024). *10Code Software Design*. Obtenido de Blade, el Motor de Plantillas Oficial de Laravel para Crear Vistas Dinámicas: <https://10code.es/blade-laravel/#:~:text=Blade%20es%20una%20herramienta%20poderosa,y%20construir%20aplicaciones%20m%C3%A1s%20eficientes.>
- Alvarez, C. (2014). *Arquitectura Java*. Obtenido de JMeter Test: <https://www.arquitecturajava.com/jmeter-test/>
- Anzuales, M. C., & Moya, J. G. (2024). *Kanban: Una metodología ágil para la gestión eficiente del flujo de trabajo en el desarrollo de software, una revision sistematica*. Quito. Obtenido de <https://dspace.itsjapon.edu.ec/jspui/bitstream/123456789/4587/1/2.%2bV3%2c%2bN1%2c%2b2024-17-28.pdf>
- Arias, A., & Durango, A. (2014). *Ingeniería y Arquitectura del Software*. EISENBRAUNS. Obtenido de https://www.google.com.ec/books/edition/Ingenier%C3%ADa_y_Arquitectura_del_Software/uADPCgAAQBAJ?hl=es&gbpv=0&kptab=overview
- Benítez, M. (2015). *Curso de Introducción a la Administración de Bases de Datos*. IT campus Academy. Obtenido de https://books.google.es/books?hl=es&lr=&id=NUSiCgAAQBAJ&oi=fnd&pg=PA1&dq=Curso+de+Introducci%C3%B3n+a+la+Administraci%C3%B3n+de+Bases+de+Datos&ots=msJLnaCD0R&sig=FdA4aHCR-C_DrxTTu16Pq2RHWE#v=onepage&q=Curso%20de%20Introducci%C3%B3n%20a%20la%20Administrac
- Cabello, M. (2010). *INTRODUCCIÓN A LAS BASES DE DATOS RELACIONALES*. Madrid: Vision Libros. Obtenido de https://www.google.com.ec/books/edition/Introducci%C3%B3n_a_las_Bases_de_Datos_relac/0lUpB1lNUdIC?hl=es&gbpv=1&printsec=frontcover
- Ciceri, M. (2018). *Introducción a Laravel*. Buenos Aires: Six Ediciones. Obtenido de <https://books.google.es/books?hl=es&lr=&id=sPylDwAAQBAJ&oi=fnd&pg=PA4&dq=laravel&ots=mZuBuzem5e&sig=4UaU3qi8KrQ0xHi0tJ5S19Z8Qp4#v=onepage&q=laravel&f=true>
- Cobo, A., Gomez, P., Perez, D., & Rocha, R. (s.f.).
- Cobo, A., Gómez, P., Pérez, D., & Rocha, R. (2005). *PHP y MySQL Tecnologías para el desarrollo de aplicaciones web*. España: Díaz de Santos. Obtenido de https://www.google.com.ec/books/edition/PHP_y_MySQL/zMK3GOMOpQ4C?hl=es&gbpv=1&dq=MySql&pg=PA384&printsec=frontcover
- Deyimar, A. (03 de 09 de 2024). *Hostinger Tutoriales*. Obtenido de ¿Qué es PHP? Aprende todo sobre el lenguaje de scripting?: https://www.hostinger.com/es/tutoriales/que-es-php?ppc_campaign=google_search_generic_hosting_all&bidkw=defaultkeyword&lo=9069555#%C2%BFPor_que_usar_PHP
- Entra en la Red. (2024). *Entra en la Red*. Obtenido de Gestión de Inventario en Laravel: <https://www.entraenlared.com/blog/gestion-de-inventario-en-laravel>

- Equipo de Contenidos de GoDaddy. (2024). *GoDaddy*. Obtenido de Guía sobre Composer: Cómo usar este gestor de dependencias para PHP: <https://www.godaddy.com/resources/es/crearweb/composer-gestor-que-es>
- Fernández, Y., & Díaz, Y. (2012). Patrón Modelo-Vista-Controlador. *Telemática (La Habana)*, 11(1), págs. 47-57. Obtenido de <https://biblat.unam.mx/es/revista/telematica-la-habana/articulo/patron-modelo-vista-controlador>
- Fossati, M. (2014). *Todo sobre MySQL*. Obtenido de https://www.google.com.ec/books/edition/Todo_sobre_MySQL/GS3kAgAAQBAJ?hl=es&gbpv=1
- Gillis, A. (2024). *TechTarget*. Obtenido de What is Apache JMeter?: <https://www.techtarget.com/searchsoftwarequality/definition/Apache-JMeter>
- Guevara, C. (2017). *Desarrollo de un sistema en entorno web para el control de la gestión del inventario de la Empresa Cuenca Llantas, utilizando como framework de Desarrollo Laravel*. Universidad de Guayaquil. Recuperado de: <https://repositorio.ug.edu.ec/items/9f9edbef-9253-40b2-94fb-5804d285170f>
- Herd Services. (s.f.). *Laravel Herd*. Obtenido de Installing MySQL via Herd Pro: <https://herd.laravel.com/docs/macOS/herd-pro-services/mysql#installing-mysql-via-herd-pro>
- Heredia, S. (2022). Sistema web para la gestión de ventas del taller de reparación y mantenimiento automotriz Heredia utilizando el framework Mean Stack. Obtenido de http://dspace.unach.edu.ec/bitstream/51000/10049/1/Santiago%20R.%2c%20Heredia%20S.%20%282021%29_Sistema%20web%20para%20la%20gesti%20%b3n%20de%20ventas%20del%20taller%20de%20reparaci%20%b3n%20y%20mantenimiento%20automotriz.pdf
- Huillcen, H., Soria, I., & Palomino, F. (2022). *Introducción a las Bases de Datos con MySQL*. Arequipa: Huillcen Baca, Herwin Alayn. Obtenido de <https://books.google.com.pe/books?id=xq5wEAAAQBAJ>
- Hurtado, R. T. (2022). Análisis comparativo para la evaluación de frameworks usados en el desarrollo de aplicaciones web. *Cedamaz*, 11(2), 133-141. Obtenido de <https://dialnet.unirioja.es/servlet/articulo?codigo=9355058>
- Jound, I. (s.f.). *Comparison of performance between Raw*. Karlskrona. Obtenido de <https://www.diva-portal.org/smash/get/diva2:1014983/FULLTEXT02>
- Porras, A. (s.f.). *Metodologías ágiles para el desarrollo de software*. Obtenido de https://www.google.com.ec/books/edition/Metodolog%C3%ADas_%C3%A1giles_para_el_desarrollo/JfXBEAAAQBAJ?hl=es&gbpv=1&dq=metodologia+de+desarrollo+de+software&printsec=frontcover
- Reis, J., & Housley, M. (2022). *Fundamentos de ingeniería de datos Planifique y desarrolle sistemas robustos de datos*. Obtenido de https://www.google.com.ec/books/edition/Fundamentos_de_ingenier%C3%ADa_de_datos/bP3DEAAAQBAJ?hl=es&gbpv=1&dq=API+REST+QUE+ES&pg=PT293&printsec=frontcover
- Vera, R. (2021). *OpenWebinar*. Obtenido de Qué es Laravel: Características y ventajas: <https://openwebinars.net/blog/que-es-laravel-caracteristicas-y-ventajas/>

- Willis, D. (2024). *InMotion Hosting*. Obtenido de PHP Artisan – A Comprehensive Guide: <https://www.inmotionhosting.com/support/edu/laravel/php-artisan/#what>
- Yanes, O., & Gracia del Busto, H. (2011). Mapeo Objeto / Relacional (ORM). *TELEMATICA*, 10(3), 1-7. Obtenido de https://web.archive.org/web/20161104065918id_/http://revistatelematica.cujae.edu.cu:80/index.php/tele/article/viewFile/23/21
- Yoris, A. (s.f.). *DJANGO 5 Aqui continua tu camino en el desarrollo de aplicaciones web en Python con Django*. Obtenido de https://www.google.com.ec/books/edition/Primeros_pasos_con_Django_5/On58EAAAQBAJ?hl=es&gbpv=1&dq=API+REST+QUE+ES&pg=PA148&printsec=frontcover

ANEXOS

ANEXO 1 MANUAL DE USUARIO



Universidad
Nacional de Chimborazo



MANUAL DE USUARIO

BODEGA MIA

Elaborado por:
Luis Muñoz M

Riobamba - 2025

Contenido

INTRODUCCION	1
Login de acceso	2
GESTIÓN DE USUARIOS.....	2
Registrar usuario	2
Editar usuario	4
Eliminar usuario	5
Ver usuario.....	5
GESTIÓN DE EMPRESAS.....	6
Registrar empresa.....	6
Editar empresa.....	8
Eliminar empresa.....	9
GESTIÓN DE ALMACENES	9
Registrar almacén	9
Editar empresa.....	11
Visualizar el reporte de almacén	12
Exportar reporte de almacén.....	13
Eliminar almacén	13
GESTIÓN DE CATEGORÍAS.....	14
Registrar categoría.....	14
Editar categoría.....	15
Eliminar almacén	16
GESTIÓN DE PRODUCTOS	17
Registrar producto.....	17
Editar producto.....	18
Eliminar producto.....	19
GESTIÓN DE MOVIMIENTOS.....	20
Registrar movimiento	20
Editar movimiento	21
Eliminar movimiento	22
GENERACIÓN DE KARDEX	23
Visualización de kardex	23
Ver kardex.....	24
Exportar Kardex.....	25
GENERACIÓN DE REPORTES	25

API RESTFUL.....	27
Visualización de Servicio Restful.....	27

INTRODUCCIÓN

Bienvenido al manual de usuario de la aplicación web Bodega Mia de gestión de inventarios para PYMEs, una herramienta diseñada para optimizar y simplificar el control de existencias en pequeñas y medianas empresas. Esta aplicación ha sido desarrollada utilizando el framework Laravel y la base de datos MySQL, lo cual garantiza una plataforma robusta, segura y escalable para la administración eficiente de productos e insumos.

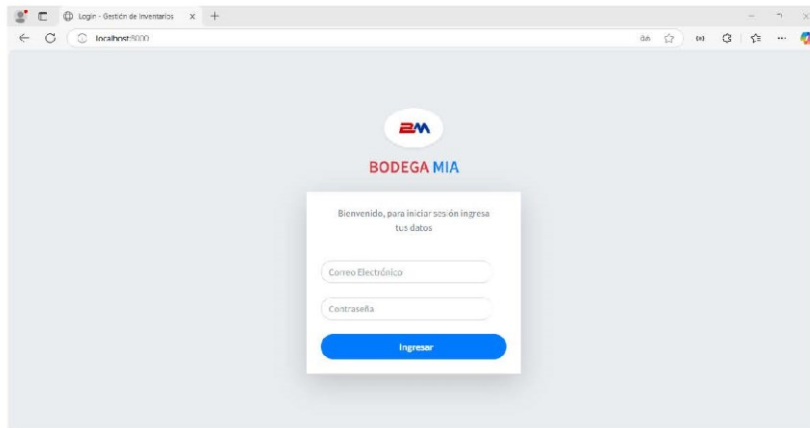
El objetivo principal de esta aplicación es proporcionar a los usuarios un sistema ágil e intuitivo para llevar un control detallado de su inventario, permitiendo registrar productos, realizar entradas y salidas de stock, establecer niveles mínimos de inventario, generar reportes, y facilitar la toma de decisiones basada en datos actualizados en tiempo real.

Este manual está dirigido a los usuarios finales del sistema, incluyendo personal de almacén, administrativos y gerentes, y tiene como propósito guiar paso a paso en el uso de las funcionalidades principales de la aplicación, desde el acceso al sistema hasta la generación de reportes.

A lo largo del documento se incluirán capturas de pantalla y ejemplos prácticos que facilitarán la comprensión del uso de cada módulo. Se recomienda leer este manual detenidamente antes de comenzar a utilizar la aplicación, con el fin de aprovechar al máximo todas sus funcionalidades.

Login de acceso

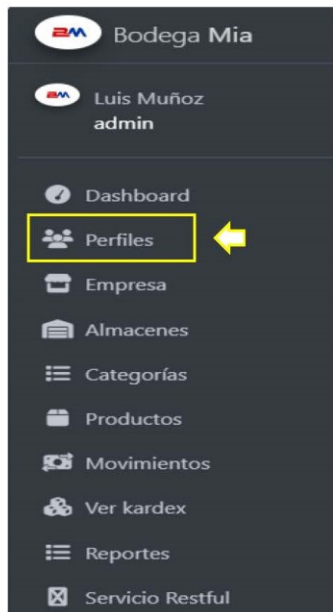
Para ingresar al sistema escribir un correo electrónico y contraseña registrados.



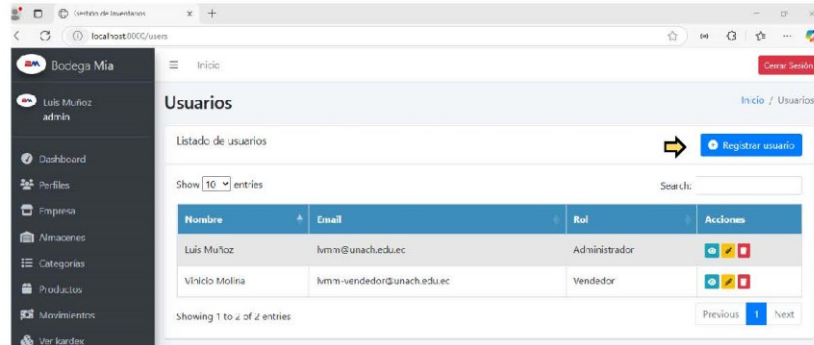
GESTIÓN DE USUARIOS

Registrar usuario

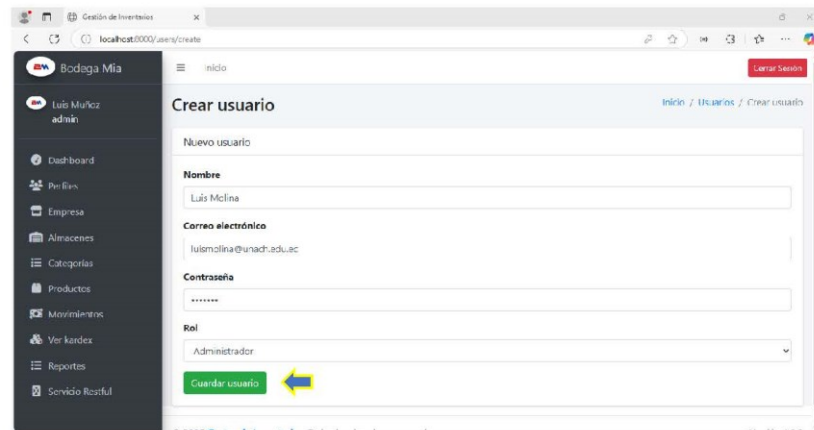
Para ingresar un usuario seleccionar del menú lateral izquierdo el icono de perfiles.



Dar click en el botón azul Registrar usuario.

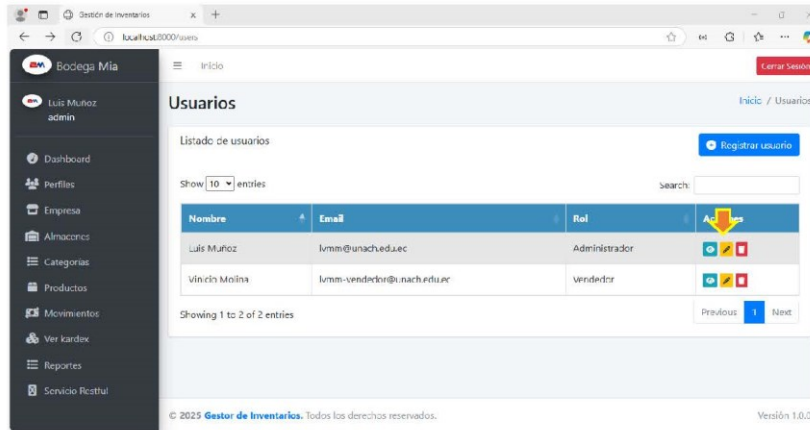


Completar todos los campos solicitados y dar click en el botón Guardar usuario.

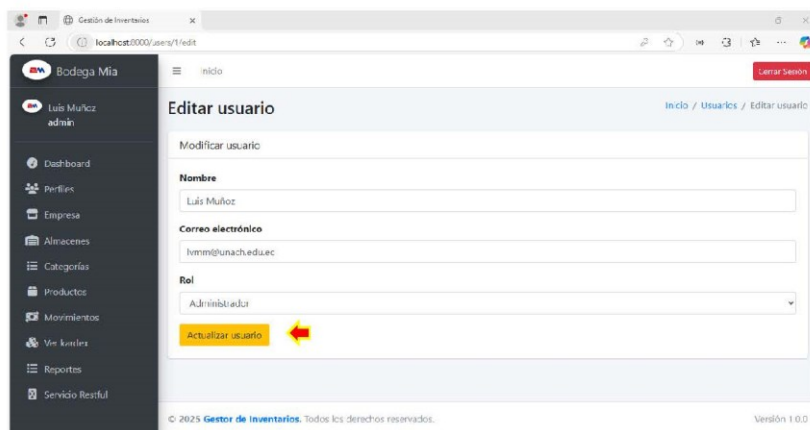


Editar usuario

Dar click en el icono de color amarillo con forma de lápiz.

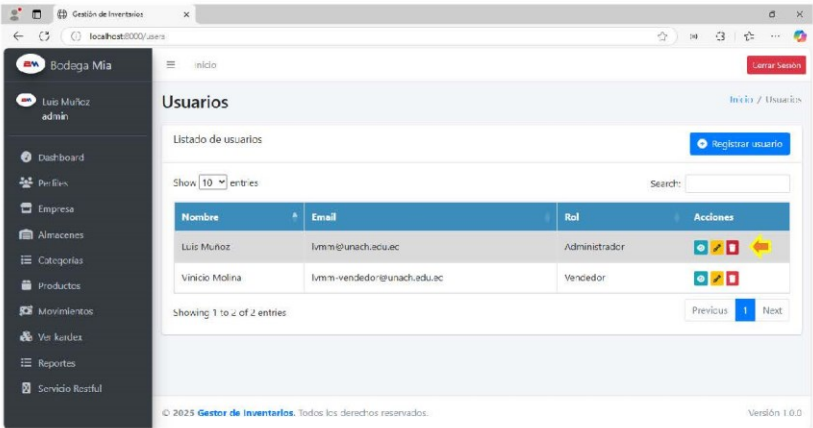


Es necesario completar todos los campos a actualizar del formulario y dar click en el botón de Actualizar usuario.



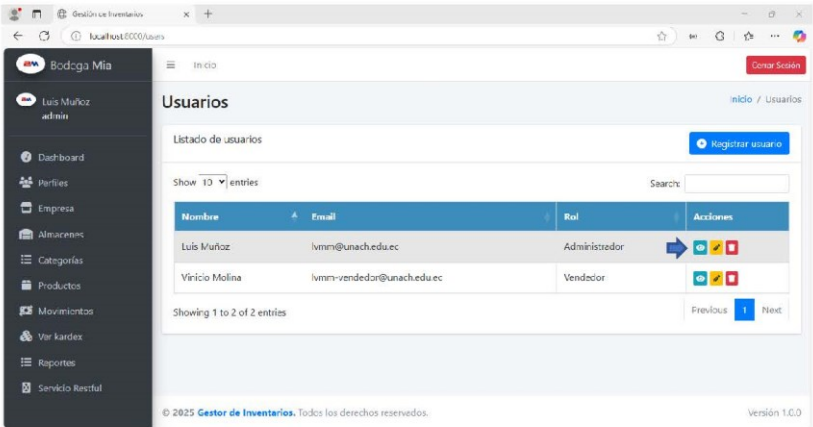
Eliminar usuario

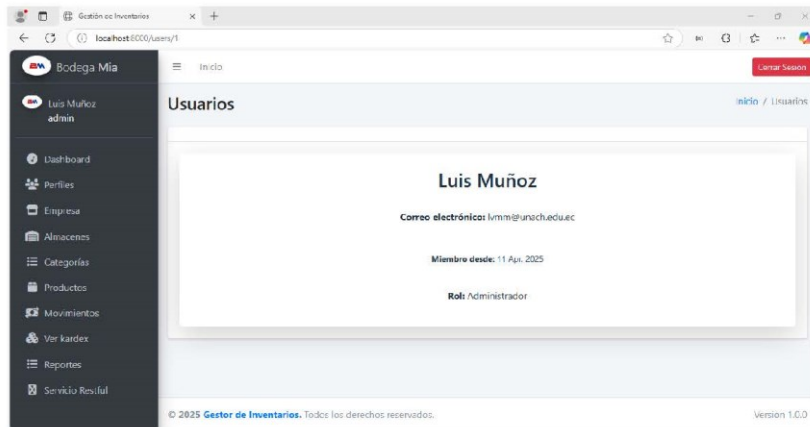
Para eliminar un usuario, elegir el registro correspondiente y dar click en el botón rojo con un icono de cesto de basura.



Ver usuario

Para visualizar un usuario, dentro del listado de usuarios dar click en el botón azul con un icono de un ojo.

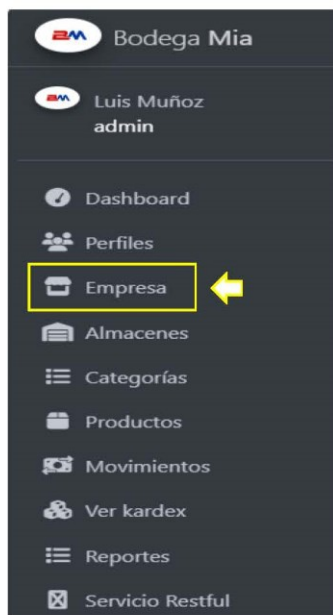




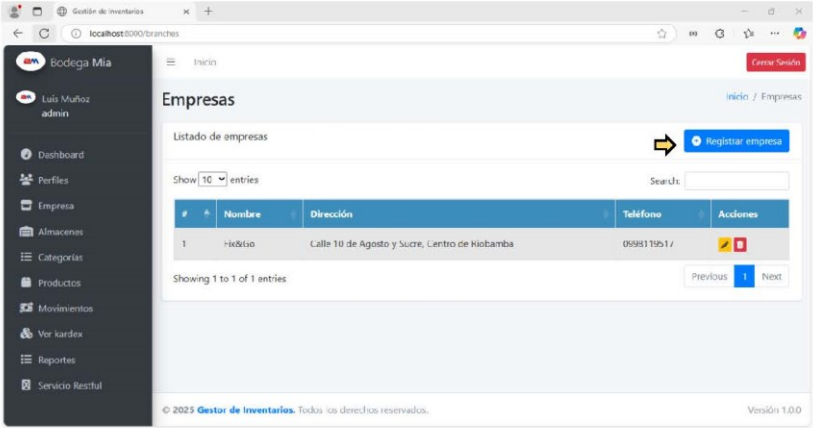
GESTIÓN DE EMPRESAS

Registrar empresa

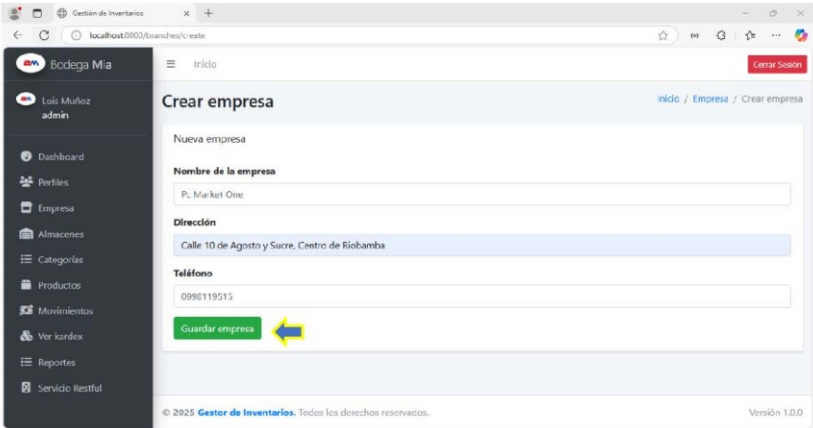
Para ingresar un usuario dirigir al menú lateral y dar click en el icono de Empresa.



Dar click en el botón azul Registrar empresa.

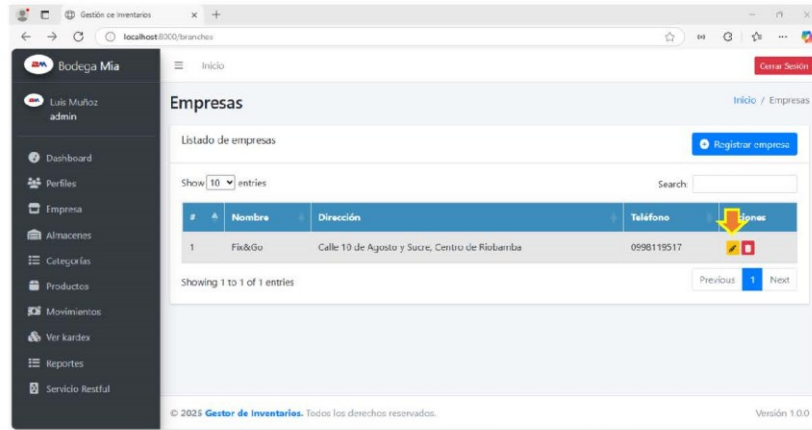


Completar todos los campos solicitados y dar click en el botón Guardar empresa.

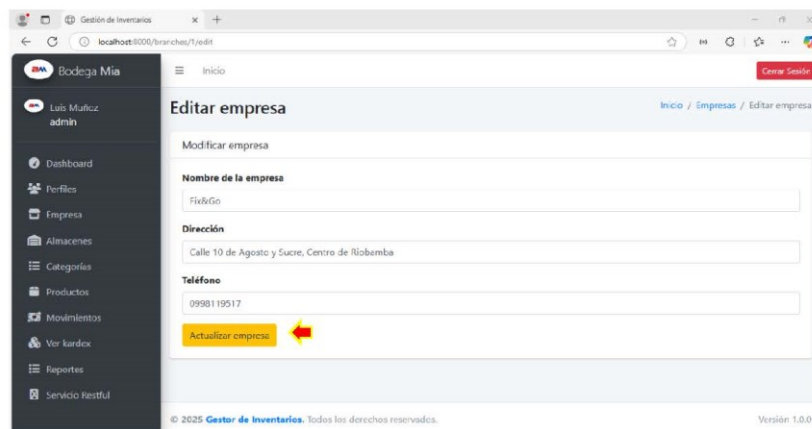


Editar empresa

Dar click en el icono de color amarillo con forma de lápiz.

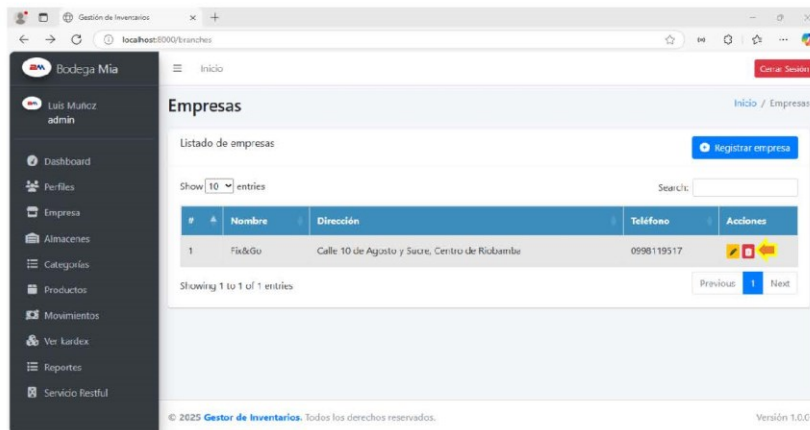


Es necesario completar todos los campos a actualizar del formulario y dar click en el botón de Actualizar empresa.



Eliminar empresa

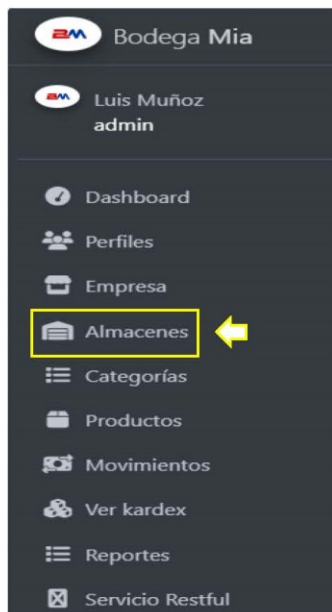
Para eliminar una empresa, elegir el registro correspondiente y dar click en el botón rojo con un icono de cesto de basura.



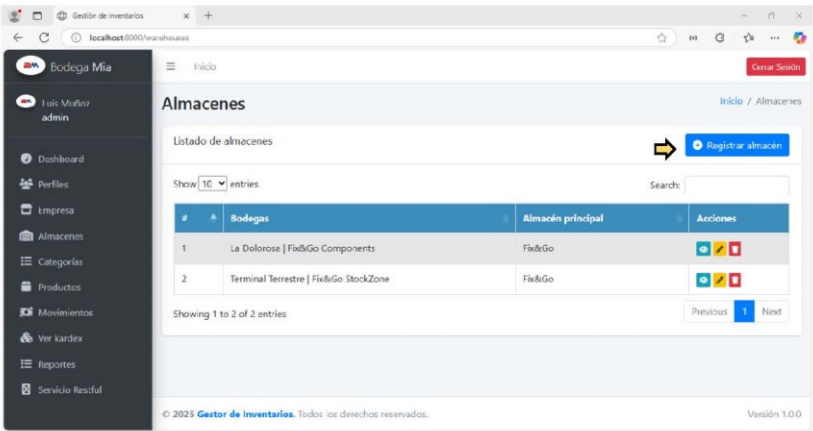
GESTIÓN DE ALMACENES

Registrar almacén

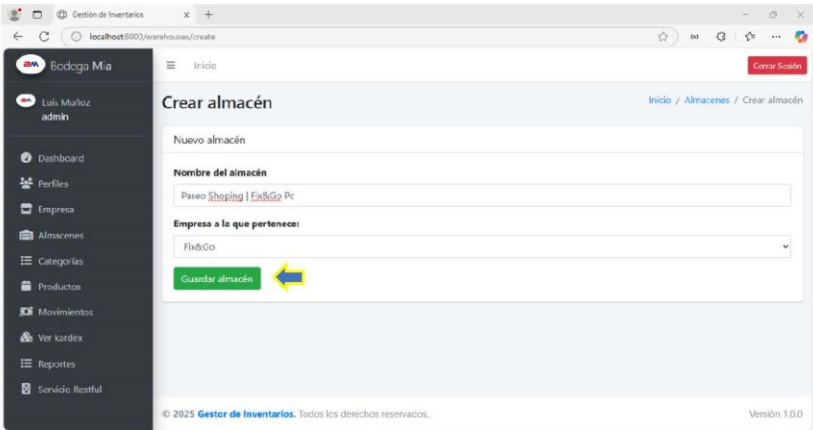
Para ingresar un usuario dirigir al menú lateral y dar click en el icono de Almacenes.



Dar click en el botón azul Registrar almacén.

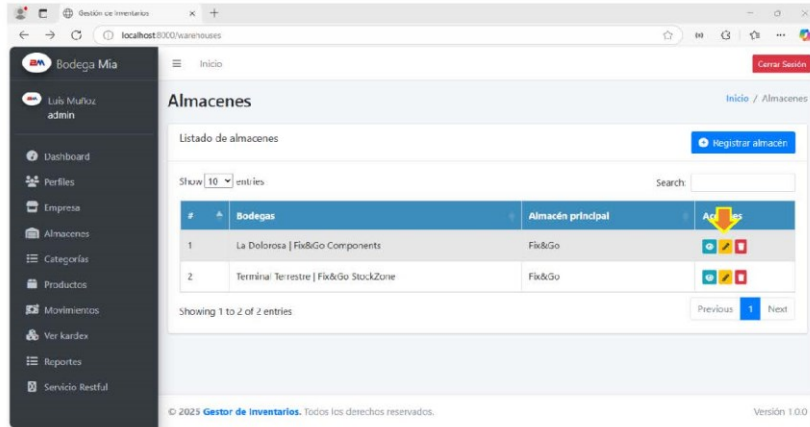


Completar todos los campos solicitados y dar click en el botón Guardar almacén.

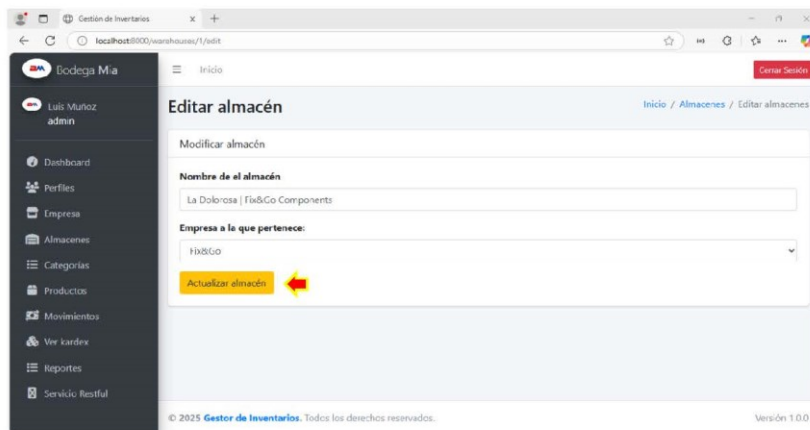


Editar empresa

Dar click en el icono de color amarillo con forma de lápiz.

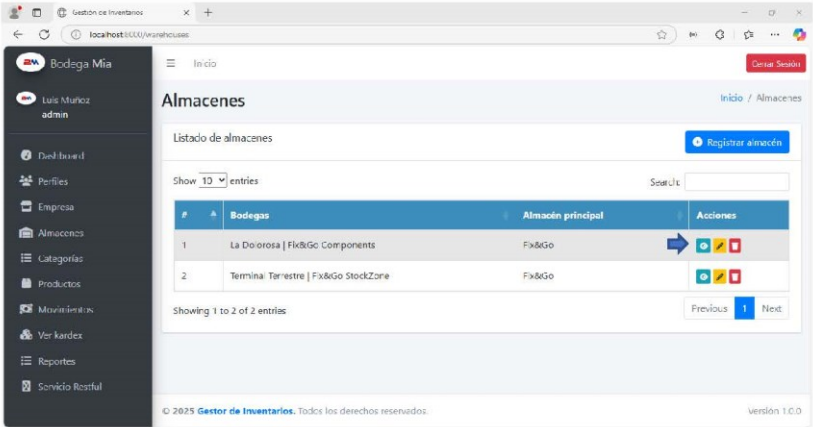


Es necesario completar todos los campos a actualizar del formulario y dar click en el botón de Actualizar almacén.

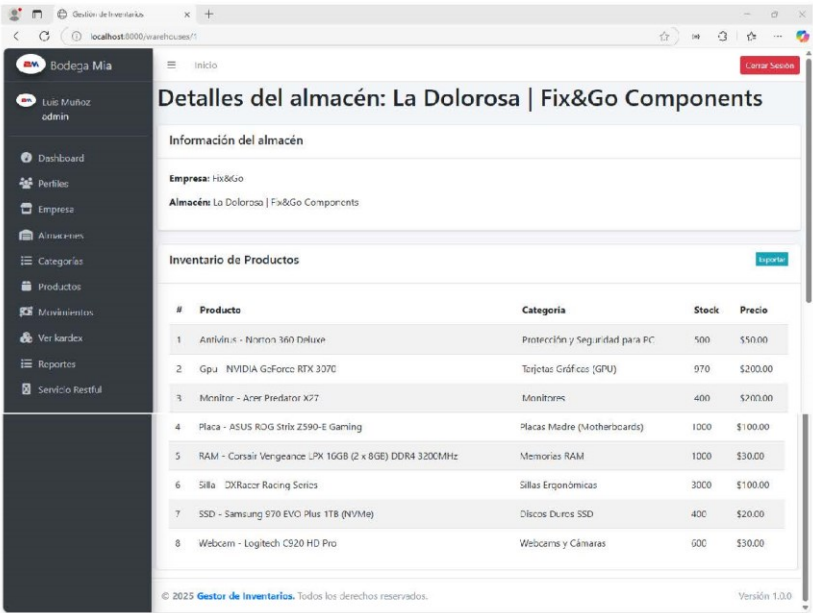


Visualizar el reporte de almacén

Para visualizar el inventario, dentro del listado de almacenes dar click en el botón azul con un icono de un ojo.

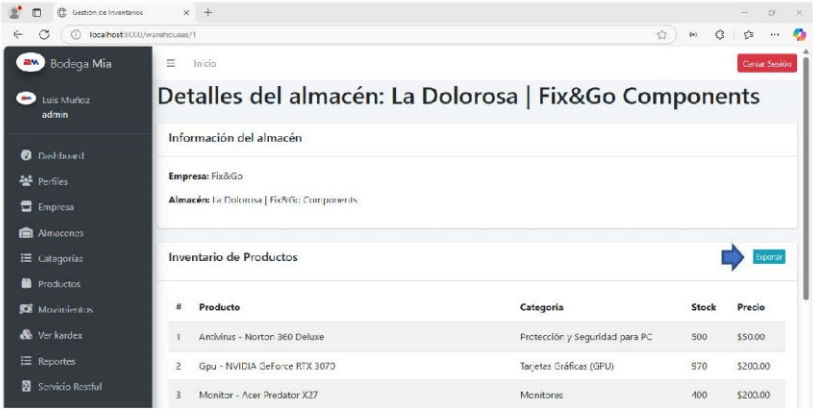


El sistema permite visualizar el inventario completo de cada almacén.



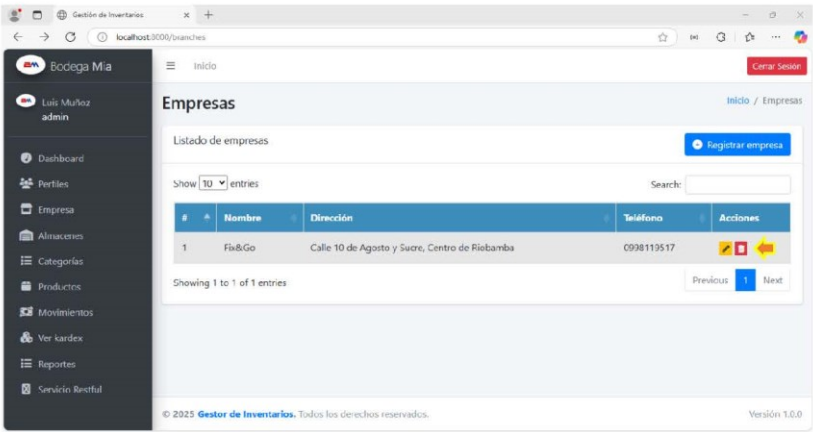
Exportar reporte de almacén

Dar click en el botón exportar de color azul y se descargara un archivo csv con el inventario completo.



Eliminar almacén

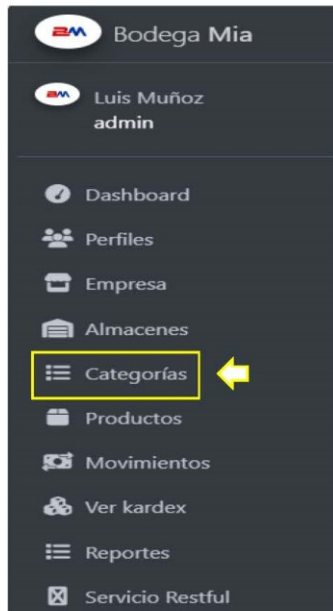
Para eliminar un almacén, elegir el registro correspondiente y dar click en el botón rojo con un icono de cesto de basura.



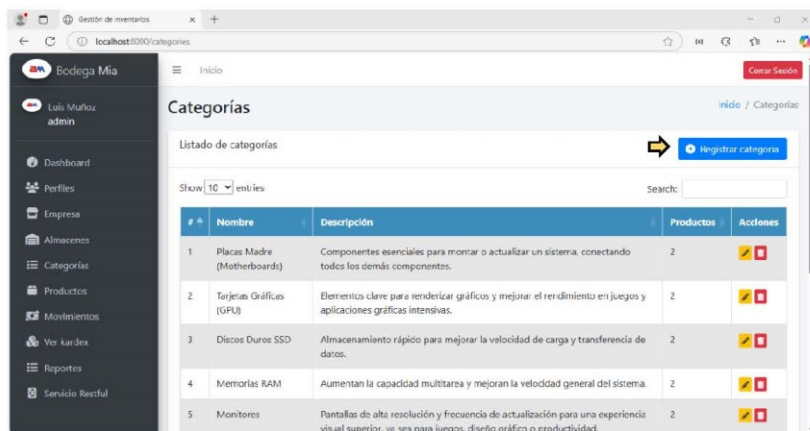
GESTIÓN DE CATEGORÍAS

Registrar categoría

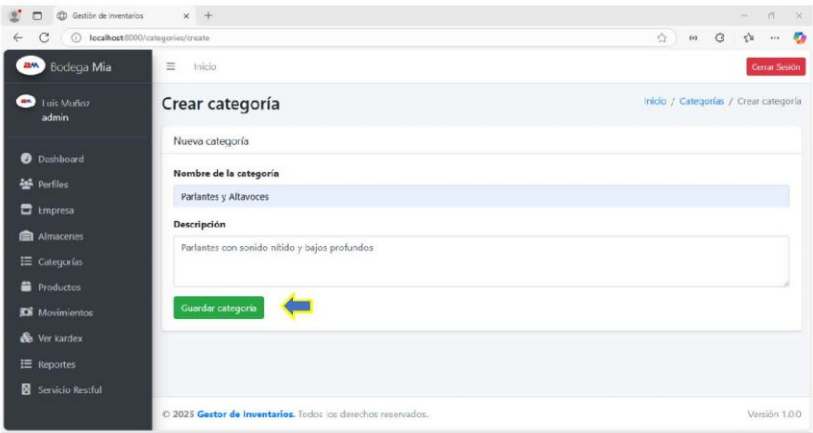
Para ingresar una categoría dirigir al menú lateral y dar click en el icono de Categorías.



Dar click en el botón azul Registrar categoría.

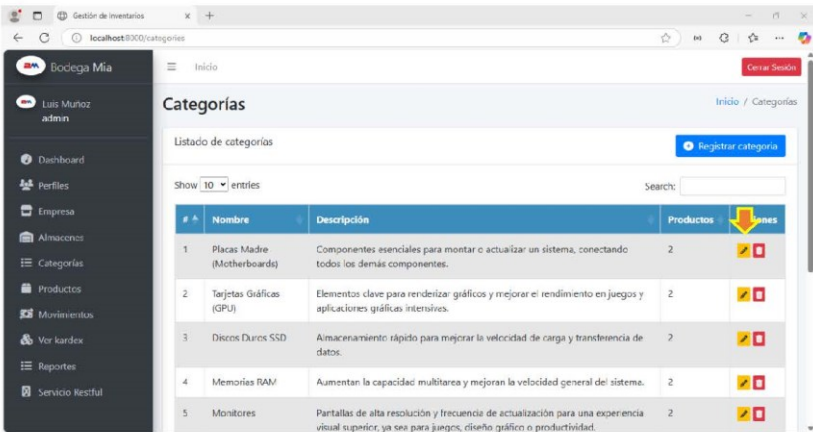


Completar todos los campos solicitados y dar click en el botón Guardar categoría.

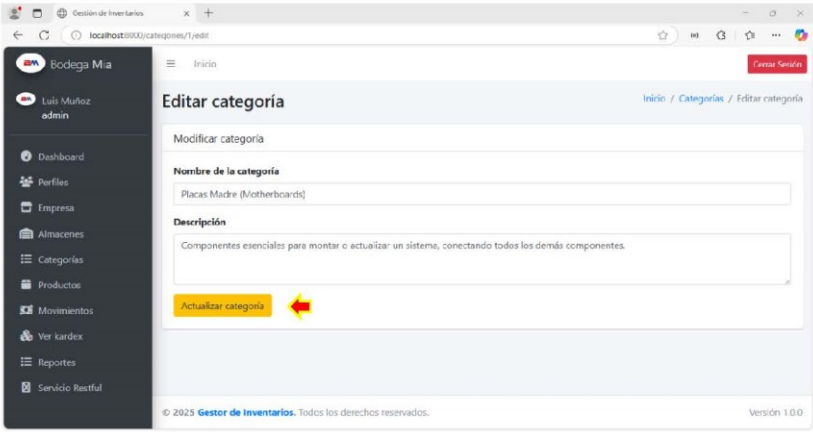


Editar categoría

Dar click en el icono de color amarillo con forma de lápiz

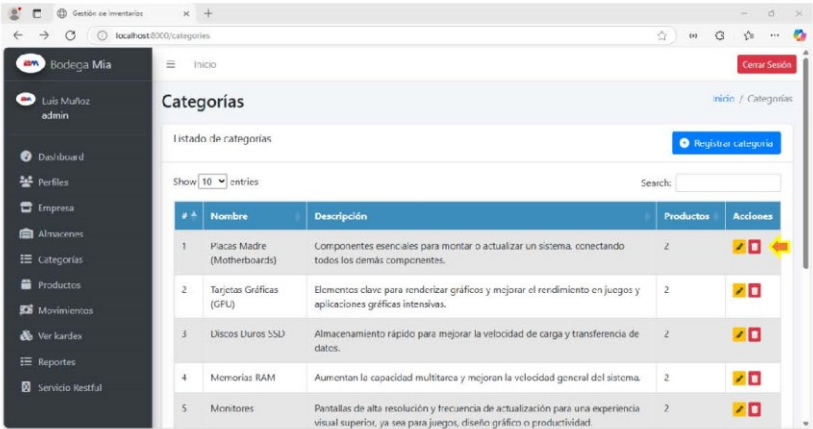


Es necesario completar todos los campos a actualizar del formulario y dar click en el botón de Actualizar categoría.



Eliminar almacén

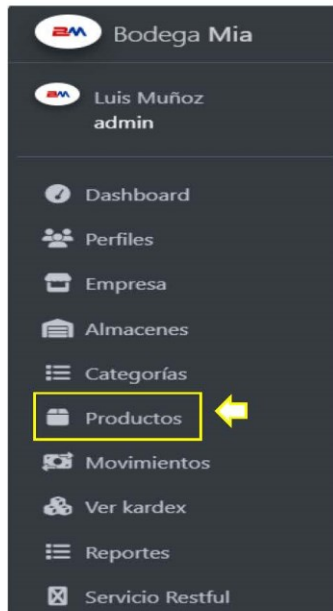
Para eliminar un almacén, elegir el registro correspondiente y dar click en el botón rojo con un icono de cesto de basura.



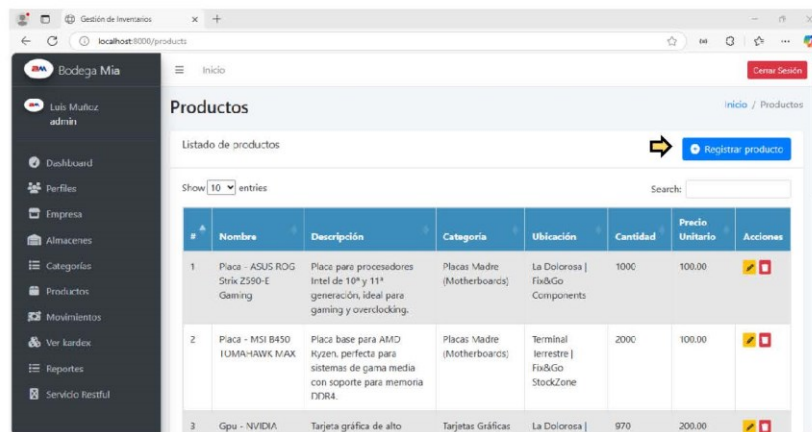
GESTIÓN DE PRODUCTOS

Registrar producto

Para ingresar un producto dirigir al menú lateral y dar click en el icono de Productos.



Dar click en el botón azul Registrar producto.



Completar todos los campos solicitados, elegir la categoría, y el almacén al que pertenece y dar click en el botón Guardar producto.

Crear Producto

Nuevo producto

Nombre del producto

Parlantes | El SW-HPS-146000

Descripción

tiene un subwoofer de graves profundos de 45 vatios con un controlador de 6.5" y un altavoz central individual que le brinda un rango de sonido medio excelente y claro

Categoría

Parlantes y Altavoces

Almacén

La Dolorosa | Fix&Go Components

Cantidad

1000

Stock mínimo

50

Stock máximo

2000

Precio unitario

20

Guardar producto

Editar producto

Dar click en el icono de color amarillo con forma de lápiz.

Productos

Listado de productos

Show 10 entries

#	Nombre	Descripción	Categoría	Ubicación	Cantidad	Precio Unitario	Acciones
1	Placa - ASUS ROG Strix Z590-E Gaming	Placa para procesadores Intel de 10ª y 11ª generación, ideal para gaming y overclocking.	Placas Madre (Motherboards)	La Dolorosa Fix&Go Components	1000	100.00	 
2	Placa - MSI B450 TOMAHAWK MAX	Placa base para AMD Ryzen, perfecta para usuarios de gama media con soporte para memoria DDR4.	Placas Madre (Motherboards)	Terminal Teniente Fix&Go StockZone	2000	100.00	 
3	Gpu - NVIDIA	Tarjeta grafica de alto	Tarjetas Graficas	La Dolorosa	970	200.00	 

Es necesario completar todos los campos a actualizar del formulario y dar click en el botón de Actualizar categoría.

Editar producto

Modificar producto

Nombre del producto
Placa - ASUS ROG Strix Z590-E Gaming

Descripción
Placa para procesadores Intel de 10ª y 11ª generación, ideal para gaming y overlocking.

Categoría
Placas Madre (Motherboards)

Ubicación
La Dolorosa | Fix&Go Components

Cantidad
1000

Stock mínimo
50

Stock máximo
1000

Precio unitario
100,00

Actualizar producto

Eliminar producto

Para eliminar un producto, elegir el registro correspondiente y dar click en el botón rojo con un icono de cesto de basura.

Productos

Listado de productos

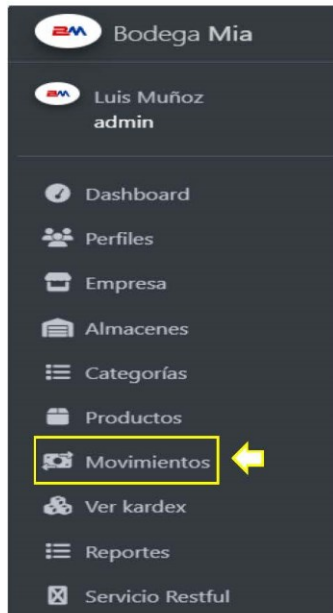
Show 10 entries

#	Nombre	Descripción	Categoría	Ubicación	Cantidad	Precio Unitario	Acciones
1	Placa - ASUS ROG Strix Z590-E Gaming	Placa para procesadores Intel de 10ª y 11ª generación, ideal para gaming y overlocking.	Placas Madre (Motherboards)	La Dolorosa Fix&Go Components	1000	100.00	
2	Placa - MSI B450 TOMAHAWK MAX	Placa base para AMD Ryzen, perfecta para sistemas de gama media con soporte para memoria DDR4.	Placas Madre (Motherboards)	Terminal Temestre Fix&Go StockZone	2000	100.00	
3	Gpu - NVIDIA	Tarjeta gráfica de alto	Tarjetas Gráficas	La Dolorosa	970	200.00	

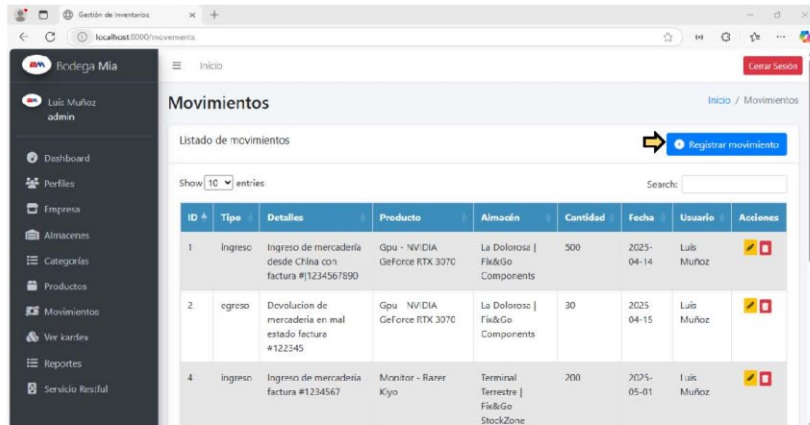
GESTIÓN DE MOVIMIENTOS

Registrar movimiento

Para registrar un movimiento dirigir al menú lateral y dar click en el icono de Productos.



Dar click en el botón azul Registrar movimiento.



Completar todos los campos solicitados, seleccionar si es ingreso o egreso y dar click en el botón Guardar movimiento.

Registrar movimiento

Nuevo movimiento

Tipo de movimiento: Ingreso

Detalle del movimiento: Ingreso de mercadería desde Japón con factura #321332761325

Producto: Silla - Secretlab TITAN Evo 2022 | Qty: 30

Almacén: Terminal Terrestre | Fix&Go StockZone

Cantidad: 1000

Fecha: 15/04/2025

Guardar movimiento

Editar movimiento

Dar click en el icono de color amarillo con forma de lápiz.

Movimientos

Lista de movimientos

Show 10 entries

ID	Tipo	Detalles	Producto	Almacén	Cantidad	Fecha	Usuario	Acciones
1	ingreso	Ingreso de mercadería desde China con factura #1234567890	Gpu - NVIDIA GeForce RTX 3070	La Dolorosa Fix&Go Components	500	2025-04-14	Luis Muñoz	
2	egreso	Devolucion de mercadería en mal estado factura #123456	Gpu - NVIDIA GeForce RTX 3070	La Dolorosa Fix&Go Components	30	2025-04-15	Luis Muñoz	
4	ingreso	Ingreso de mercadería factura #1234567	Monitor - Razor Kiyo	Terminal Terrestre Fix&Go StockZone	200	2025-05-01	Luis Muñoz	

Es necesario completar todos los campos a actualizar del formulario y dar click en el botón de Actualizar movimiento.

Editar movimiento

Modificar movimiento

Tipo de movimiento
Ingreso

Detalle del movimiento
Ingreso de mercadería desde China con factura #1234567890

Producto
Gpu - NVIDIA GeForce RTX 3070

Almacén
La Dolorosa | Fix&Go Components

Cantidad
500

Fecha
14/04/2025

Actualizar movimiento

Eliminar movimiento

Para eliminar un movimiento, se debe elegir el registro correspondiente y dar click en el botón rojo con un icono de cesto de basura.

Movimientos

Listado de movimientos

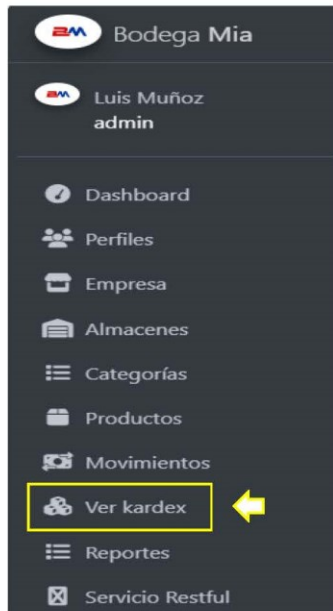
Show 10 entries

ID	Tipo	Detalles	Producto	Almacén	Cantidad	Fecha	Usuario	Acciones
1	ingreso	ingreso de mercadería desde China con factura #1234567890	Gpu - NVIDIA GeForce RTX 3070	La Dolorosa Fix&Go Components	500	2025-04-14	Luis Muñoz	[icono de basura]
2	egreso	Devolución de mercadería en mal estado factura #122345	Gpu - NVIDIA GeForce RTX 3070	La Dolorosa Fix&Go Components	30	2025-04-15	Luis Muñoz	[icono de basura]
4	ingreso	ingreso de mercadería factura #1234567	Monitor - Razer Kiyo	Terminal Terrestre Fix&Go StockZone	200	2025-05-01	Luis Muñoz	[icono de basura]

GENERACIÓN DE KARDEX

Visualización de kardex

Para visualizar la kardex dirigir al menú lateral y dar click en el icono de Ver kardex.



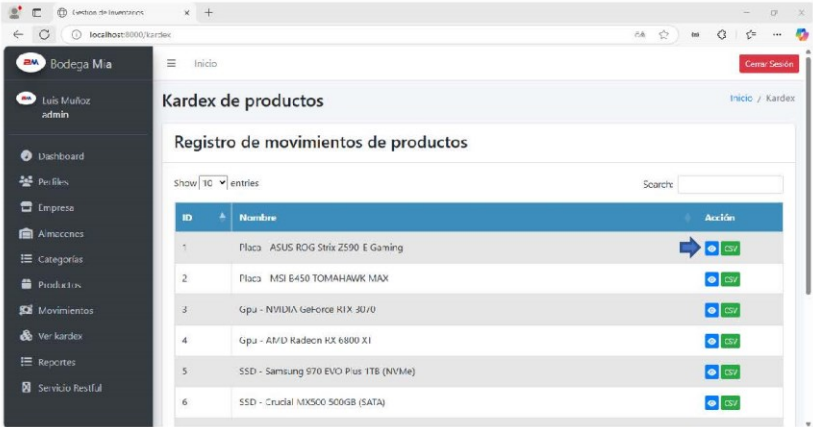
Se pueden visualizar los diferentes kardex de cada producto ingresado.

The image shows a web browser displaying the 'Kardex de productos' page. The page has a sidebar menu on the left with the same items as the mobile app. The main content area is titled 'Kardex de productos' and 'Registro de movimientos de productos'. It includes a 'Show 10 entries' dropdown and a search bar. Below is a table with 6 rows of product movement data.

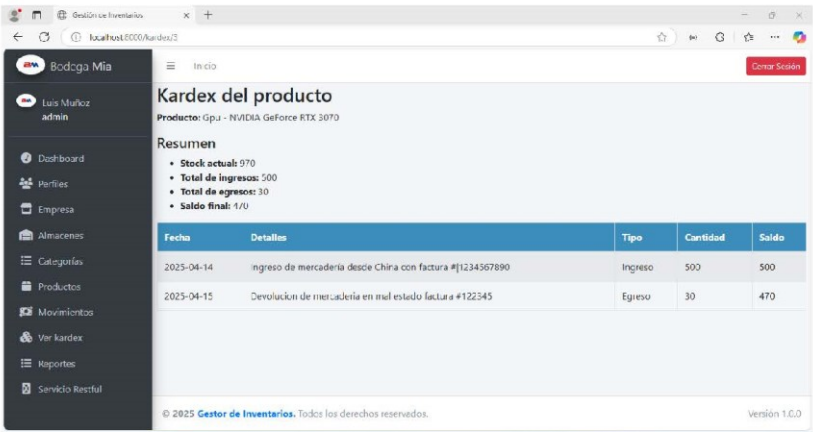
ID	Nombre	Acción
1	Placa - ASUS ROG Strix Z590-E Gaming	[Icono] [CSV]
2	Placa - MSI B450 TOMAHAWK MAX	[Icono] [CSV]
3	Gpu - NVIDIA GeForce RTX 3070	[Icono] [CSV]
4	Gpu - AMD Radeon RX 6800 XT	[Icono] [CSV]
5	SSD - Samsung 970 EVO Plus 1TB (NVMe)	[Icono] [CSV]
6	SSD - Crucial MX600 500GB (SATA)	[Icono] [CSV]

Ver kardex

Para visualizar un kardex, dentro del listado de usuarios dar click en el botón azul con un icono de un ojo.

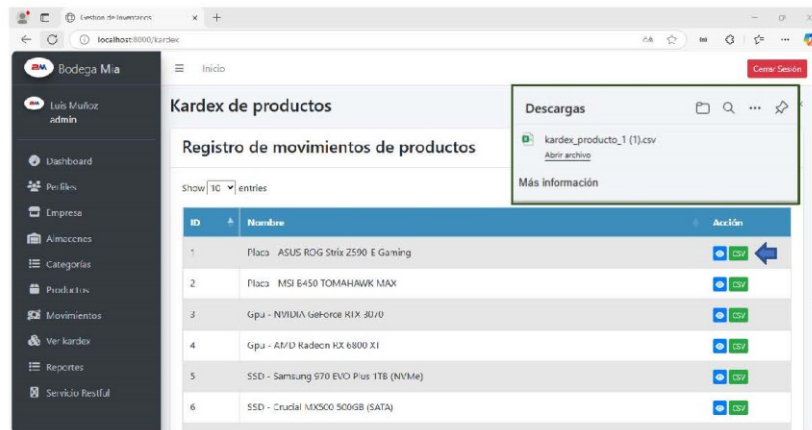


Se pueden visualizar los diferentes detalles del Kardex del producto elegido.



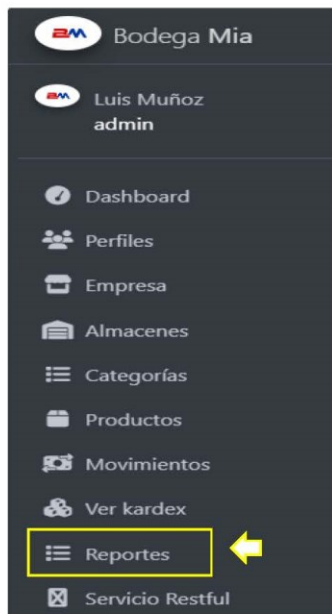
Exportar Kardex

Para exportar la kardex del producto elegido, se debe dar click en el botón de color verde csv.



GENERACIÓN DE REPORTES

Para visualizar el apartado de reportes dirigir al menú lateral y dar click en el icono de Reportes.



Reporte de movimientos

Filtros

Tipo de Movimiento: Todos Año: Todos Mes: Todos **Filtrar** **Limpiar**

Resultados

Tipo	Producto	Almacén	Usuario	Cantidad	Fecha
Ingreso	Monitor - Razer Kiyo	Terminal Terrestre Fix&Go StockZone	Luis Muñoz	200	01/05/21
Egreso	Gpu - NVIDIA GeForce RTX 3070	La Dolorosa Fix&Go Components	Luis Muñoz	30	15/04/21
Egreso	Silla - Secretlab TITAN Evo 2022	Terminal Terrestre Fix&Go StockZone	Luis Muñoz	70	15/04/21
Ingreso	Gpu - NVIDIA GeForce RTX 3070	La Dolorosa Fix&Go Components	Luis Muñoz	500	14/04/21

Para filtrar por movimiento de ingreso o egreso, por año o por meses y dar click en el botón filtrar.

Reporte de movimientos

Filtros

Tipo de Movimiento: Egreso Año: 2025 Mes: April **Filtrar** **Limpiar**

Resultados

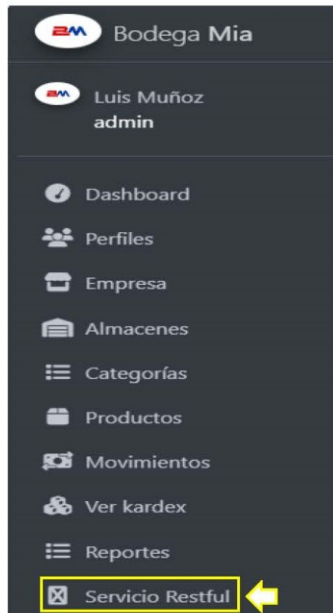
Tipo	Producto	Almacén	Usuario	Cantidad	Fecha
Egreso	Gpu - NVIDIA GeForce RTX 3070	La Dolorosa Fix&Go Components	Luis Muñoz	30	15/04/2025
Egreso	Silla - Secretlab TITAN Evo 2022	Terminal Terrestre Fix&Go StockZone	Luis Muñoz	70	15/04/2025
Egreso	SSD - Samsung 970 EVO Plus 1TB (NVMe)	La Dolorosa Fix&Go Components	Luis Muñoz	100	11/04/2025

© 2025 Gestor de Inventarios. Todos los derechos reservados. Versión 1.0.0

API RESTFUL

Visualización de Servicio Restful

Para visualizar el servicio Restful dirigir al menú lateral y dar click en el icono de Servicio RestFul.



El sistema permite visualizar los diferentes endpoints para que otra aplicación los pueda consumir o interactuar.

