UNIVERSIDAD DE CARABOBO

Facultad Experimental de Ciencias y Tecnología

Licenciatura en Computación

Aplicación para comparar precios de la competencia del sector *retail* de alimentos venezolano, con *Web Scraping* y Aprendizaje Automático. Un enfoque de Inteligencia de Negocio

Trabajo Especial de Grado presentado ante la ilustre Universidad de Carabobo como credencial de mérito para optar al título de Licenciado en Computación

Autor:

Br. Jesús David Machado

Br. José David López

Tutor:

Prof. Marylin Giugni

Prof. Mirella Herrera

Naguanagua, noviembre 2025

Dedicatoria

A nuestras familias y amigos, por ser el pilar fundamental y la inspiración detrás de cada página. Su apoyo incondicional y su compañía constante fueron el motor que nos impulsó a alcanzar esta meta tan anhelada. Este logro es tan suyo como nuestro.

Agradecimientos

En primer lugar, quiero expresar mi más profundo agradecimiento a Dios, por ser la luz que ha guiado mi camino y la fortaleza que me ha sostenido en cada momento de debilidad. Su presencia constante me ha dado la fe y la perseverancia necesarias para culminar esta importante etapa de mi vida.

A mis padres, pilares de mi existencia y mi más grande fuente de inspiración. Les agradezco por su amor incondicional, su sacrificio incansable y por inculcar en mí el valor del esfuerzo y la dedicación. Gracias por su apoyo incondicional y por estar siempre a mi lado en cada nuevo desafío.

A mi hermana, Daniery Machado, por ser mi cómplice, mi confidente y mi alegría. Su apoyo incondicional y sus palabras de aliento han sido un motor fundamental para seguir adelante, recordándome siempre que no estaba solo en este viaje.

A mis compañeros Luis Augusto, José Aguirre, Angelica, Gerardo, Ulises, José López y Jeremi por su compañía y motivación durante este desafio.

A mis tutoras, la Prof. Marylin y la Prof. Mirella, por su invaluable guía, paciencia y dedicación. Les agradezco profundamente por su compromiso, por compartir sus conocimientos y por su fe en mi potencial. Su orientación fue fundamental para la materialización de este trabajo.

Finalmente, un agradecimiento especial a Frédéric Chopin, cuya música me acompañó mucho antes de que supiera su nombre. Sus melodías fueron un refugio y una fuente de inspiración en largas noches de estudio, brindándome la calma y la concentración necesarias para transformar las ideas en palabras.

Jesús Machado

En primer lugar, quiero agradecer a Dios, por haberme permitido llegar hasta este punto y lograr esta meta de vida tan importante.

A mis padres, Arnaldo López y Melba Ríos que a lo largo de toda la vida me han dado un amor y apoyo incondicional, así como también han sido grandes mentores, que me han enseñado a no darme por vencido.

A mi hermano, Arnaldo Xavier, por su apoyo y enseñanzas, además de introducirme a la informática.

A mi abuela María y mi tía Ylse, por su paciencia para escucharme y por sus palabras de aliento que siempre me animaron.

A mis compañeros, Aimara, Gerardo, Harold, Jesús, Luis y Ulises por su compañía y apoyo en este trayecto universitario.

A nuestras tutoras, Marylin Giugni y Mirella Herrera por todas sus enseñanzas, y por todo su apoyo tanto académico, como profesional.

José López

Índice General

Índice de Figuras

Índice de Tablas

UNIVERSIDAD DE CARABOBO
Facultad Experimental de Ciencias y Tecnología
Licenciatura en Computación

Aplicación para comparar precios de la competencia del sector *retail* de alimentos venezolano, con *Web Scraping* y Aprendizaje Automático. Un enfoque de Inteligencia de Negocio

Resumen

En el entorno empresarial venezolano, caracterizado por una alta volatilidad económica y un creciente comercio electrónico, las empresas del sector *retail* de alimentos enfrentan el desafío de tomar decisiones de precios informadas y ágiles. En la actualidad, los métodos tradicionales de análisis de precios de la competencia son lentos, costosos y limitados en su capacidad de respuesta. Ante esta problemática, este trabajo propone el desarrollo de una aplicación que integra *price scraping*, aprendizaje automático y herramientas de *Business Intelligence* (BI), para la visualización interactiva de la información. El objetivo es transformar datos brutos de precios, a menudo dispersos y desestructurados en la web, en conocimiento estratégico para que las empresas que operan en el sector minorista de alimentos en Venezuela, tomen decisiones de precios más rápidas, informadas y efectivas, lo que podría traducirse en una mejora de su competitividad en el mercado. Para lograrlo, se utilizará la metodología Investigación-Acción en combinación con XP (*eXtreme Programming*) para el desarrollo ágil de software y KDD (*Knowledge Discovery in Databases*) para llevar a cabo el procesamiento, limpieza y preparación de los datos extraídos, que serán utilizados en las etapas posteriores de análisis y modelado con aprendizaje automático.

Palabras Claves: *Price Scraping*, Inteligencia de Negocios, Aprendizaje Automático, *Knowledge Discovery in Databases* (KDD).

UNIVERSIDAD DE CARABOBO
Facultad Experimental de Ciencias y Tecnología
Licenciatura en Computación

An application for comparing prices of competitors in the Venezuelan food retail sector using Web Scraping and Machine Learning. A Business Intelligence approach.

Abstract

In the Venezuelan business environment, characterized by high economic volatility and growing e-commerce, companies in the food retail sector face the challenge of making informed and agile pricing decisions. Currently, traditional methods of competitor price analysis are slow, costly, and limited in their responsiveness. Faced with this problem, this work proposes the development of an application that integrates price scraping, machine learning, and Business Intelligence (BI) tools for interactive information visualization. The objective is to transform raw price data, often scattered and unstructured on the web, into strategic knowledge so that companies operating in the food retail sector in Venezuela can make faster, more informed, and more effective pricing decisions, which could translate into improved market competitiveness. To achieve this, the Action Research methodology will be used in combination with XP (eXtreme Programming) for agile software development and KDD (Knowledge Discovery in Databases) to carry out the processing, cleaning and preparation of the extracted data, which will be used in the subsequent stages of analysis and modeling with machine learning

Keywords: *Price Scraping*, Business intelligence, Machine learning, *Knowledge Discovery in Databases* (KDD).

Introducción

En el dinámico y competitivo panorama empresarial actual, la capacidad de una organización para fundamentar sus decisiones en información precisa y oportuna se ha convertido en un factor crítico para asegurar su permanencia y consolidación en el mercado. Dentro de este contexto, el análisis exhaustivo de las estrategias de precios implementadas por la competencia emerge como una variable de especial relevancia, con un impacto directo y significativo en la rentabilidad y la posición competitiva de las empresas (Tatikonda, Vemuri & Thaneeru, 2023).

Tradicionalmente, las empresas han recurrido a métodos de análisis de mercado que, si bien proporcionan información valiosa, a menudo se basan en estructuras de precios estáticas y procesos manuales que limitan su capacidad de adaptación ante la rápida evolución de las condiciones del mercado. Técnicas convencionales como encuestas y análisis de costos internos, aunque útiles, suelen ser procesos que consumen tiempo y recursos considerables, además de ofrecer una respuesta limitada a la velocidad con la que cambian las dinámicas del mercado.

Sumado a esto, las organizaciones se enfrentan al desafío de integrar y analizar grandes volúmenes de datos dispersos y desestructurados para obtener una comprensión profunda de las estrategias de precios de sus competidores (Valecillos, 2019; Nagle & Müller, 2018). La dificultad inherente a la extracción y el análisis manual de esta información relevante a menudo resulta en una falta de conocimiento detallado sobre los precios y las tácticas de la competencia, generando una desventaja competitiva significativa que puede traducirse en la pérdida de oportunidades y cuota de mercado (Nuñez Cartolin, 2021).

En el contexto específico de la gestión empresarial en Venezuela, la necesidad de información precisa y oportuna sobre los precios de la competencia adquiere una relevancia aún mayor, dada la marcada volatilidad económica y el crecimiento exponencial del comercio electrónico (Observatorio Venezolano de Finanzas - OVF, 2025). A pesar de que muchas organizaciones en el país poseen grandes cantidades de datos, la carencia de sistemas

eficientes para su análisis (Torres Benítez, 2020) dificulta la toma de decisiones estratégicas acertadas y oportunas.

En este escenario, las herramientas de inteligencia de negocios (*Business Intelligence*, BI) se presentan como soluciones fundamentales para transformar datos brutos en información valiosa y comprensible, permitiendo a la gerencia visualizar información relevante y obtener una ventaja competitiva. En esta línea, el *price scraping* emerge como una técnica eficaz para la extracción automatizada de datos de precios desde sitios web de la competencia. Sin embargo, para convertir estos datos en *insights* estratégicos y facilitar la toma de decisiones, se requiere la aplicación de aprendizaje automático dentro de un enfoque de inteligencia de negocio (Cobo et al., 2025).

Esta investigación se centra en la necesidad de desarrollar una aplicación para el sector *retail* de alimentos en Venezuela, que integre *price scraping*, aprendizaje automático y herramientas de inteligencia de negocio, con el objetivo de recopilar, analizar y presentar información de precios de la competencia de manera automatizada. Esta solución busca transformar datos brutos en conocimiento estratégico que permita a las empresas del sector *retail* tomar decisiones de precios rápidas y efectivas. La implementación de esta aplicación no solo representa un avance tecnológico para los *retail* venezolanos, sino también un aporte significativo al campo de la computación aplicada en el contexto de la inteligencia de negocios.

Para concretar, el presente trabajo se organiza en cinco capítulos. El Capítulo I define el problema, su justificación y los objetivos. El Capítulo II establece el marco teórico y los trabajos relacionados. El Capítulo III detalla la metodología de investigación y desarrollo de software. El Capítulo IV presenta los resultados, incluyendo la arquitectura del sistema, la metodología de desarrollo de software *eXtremeProgramming* (*XP*) y el pipeline para la modelación *Knowledge Discovery in Databases* (*KDD*) y la evaluación de rendimiento de modelo. Finalmente, el Capítulo V expone las conclusiones, limitaciones y recomendaciones, cerrando el documento con las Referencias Bibliográficas.

Capítulo I. El Problema.

1.1. Planteamiento del problema

En un entorno empresarial cada vez más competitivo, la capacidad de una organización para tomar decisiones informadas es crucial para asegurar y consolidar su posición en el mercado. Entre las variables críticas que influyen directamente en la rentabilidad y la competitividad se encuentran las estrategias de precios de los competidores. Tradicionalmente, las empresas han dependido de métodos de análisis de mercado que, como argumentan Tatikonda, Vemuri y Thaneeru (2023), a menudo se basan en estructuras de precios estáticas, limitando su agilidad para adaptarse a las dinámicas del mercado. Métodos convencionales como encuestas y análisis de costos internos, si bien útiles, suelen ser procesos lentos, costosos y con una capacidad de respuesta limitada ante la rápida evolución de las condiciones del mercado.

Adicionalmente, las empresas se enfrentan al desafío de integrar datos dispersos y desestructurados en sus procesos de toma de decisiones. Como señala Valecillos (2019), la extracción y el análisis de información relevante a partir de grandes volúmenes de datos han sido históricamente procesos complejos y tediosos. Esta dificultad a menudo resulta en una falta de conocimiento profundo sobre los precios y las estrategias de la competencia (Nagle & Müller, 2018), generando una desventaja competitiva significativa que puede llevar a la pérdida de oportunidades y cuota de mercado.

En este sentido, Nuñez Cartolin (2021) subraya las ineficiencias de la consulta manual de datos, que conlleva tiempos de espera prolongados, informes inexactos y baja disponibilidad de información, afectando directamente la agilidad en la toma de decisiones. En concreto, su investigación revela que el 35% de los reportes se entregan fuera del plazo estimado, que en promedio es de 4 horas, un 27% son inexactos y un 23% están desactualizados, lo que inevitablemente conduce a decisiones basadas en datos obsoletos.

Más aún, el mismo autor destaca el enorme costo en tiempo que implica la búsqueda manual de precios y datos, estimando que se consumen alrededor de 160 horas al mes para generar 40 reportes, lo que equivale a 20 días laborales perdidos anualmente en tareas

repetitivas. En marcado contraste, la implementación de soluciones de BI puede reducir el tiempo de generación de cada reporte a tan solo 8 segundos. Esta mejora en eficiencia se traduce en un ahorro económico significativo, pasando de un costo mensual aproximado de 7000 USD a una inversión única de 4000 USD.

Ahora bien, en el contexto específico de la gestión empresarial en Venezuela, la necesidad de información precisa y oportuna sobre los precios de la competencia es aún más crítica. A pesar de que muchas organizaciones poseen grandes cantidades de datos, como advierte Torres Benítez (2020), a menudo carecen de sistemas eficientes para su análisis, lo que ralentiza las operaciones gerenciales y dificulta la toma de decisiones estratégicas acertadas.

En este escenario, las herramientas de inteligencia de negocios o *Business Intelligence* (BI), se presentan como soluciones fundamentales para transformar datos en información valiosa y comprensible, permitiendo a la gerencia visualizar la información relevante a través de *dashboards* interactivos para obtener una ventaja competitiva. En esta línea, el *price scraping* emerge como una técnica eficaz para la extracción automatizada de datos de precios desde sitios web de la competencia. Sin embargo, para convertir estos datos brutos en *insights* estratégicos y facilitar la toma de decisiones proactivas, se requiere la aplicación de aprendizaje automático dentro de un enfoque de inteligencia de negocio. La combinación de estas tecnologías permite no solo comparar precios en tiempo real, sino también identificar patrones de precios, predecir tendencias y comprender mejor las estrategias de la competencia, tal como lo demuestra el trabajo de Cobo et al. (2025).

A pesar de los beneficios demostrados de estos sistemas, la adopción en empresas venezolanas aún enfrenta desafíos significativos. Investigaciones como la de Torres Benítez (2020) en su estudio sobre modelos de inteligencia de negocios en el ámbito gerencial en Venezuela, señalan que muchas organizaciones, a pesar de poseer datos, carecen de sistemas de análisis, lo cual puede reflejar una falta de conocimiento, recursos o personal capacitado para implementar soluciones avanzadas de análisis de datos e inteligencia de negocios. Esta situación, como se evidencia en el diagnóstico de la transformación digital empresarial en Venezuela realizado por Nouel y Rodríguez (2024), perpetúa la dependencia de métodos manuales ineficientes, manteniendo a las empresas en una desventaja competitiva.

Esta problemática se manifiesta de forma directa en el caso de una empresa del sector de alimentos con la que se ha establecido una colaboración para esta investigación. Por motivos de confidencialidad y para proteger datos estratégicos, en adelante se hará referencia a esta entidad como el "*retailer* colaborador". Dicho *retailer* confirmó que su capacidad para reaccionar a los movimientos del mercado se ve directamente afectada por la dependencia en los procesos manuales de monitoreo de precios previamente descritos. En respuesta a esta necesidad concreta y validada, surge el propósito de desarrollar una aplicación que integre *price scraping,* aprendizaje automático y herramientas de inteligencia de negocio. El objetivo es proporcionar a este *retailer*, y por extensión a otras empresas del sector en Venezuela, una herramienta para recopilar, analizar y presentar información de la competencia de manera automatizada, transformando los datos en conocimiento accionable que permita tomar decisiones de precios rápidas y efectivas.

1.2. Objetivos

1.2.1. Objetivo General

Desarrollar una aplicación para comparar precios de la competencia del sector *retail* de alimentos venezolano, con *Web Scraping* y Aprendizaje Automático, utilizando inteligencia de negocio para la visualización y análisis de la información obtenida.

1.2.2. Objetivos Específicos

1. Realizar una revisión bibliográfica de las técnicas de *price scraping*, sus herramientas asociadas, procesamiento y almacenamiento de datos, aprendizaje automático e inteligencia de negocios, para obtener el marco conceptual de la investigación.
2. Seleccionar las técnicas de *price scraping*, algoritmos de aprendizaje automático y herramientas de BI adecuadas para la solución del problema.
3. Diseñar la arquitectura de software modular de la aplicación para la definición de sus componentes clave y la forma en que interactúan entre sí.

4. Desarrollar el módulo de *price scraping* que permita extraer precios de productos específicos de los sitios web de empresas de *retail* de alimentos seleccionadas en Venezuela.
5. Integrar los algoritmos de aprendizaje automático seleccionados sobre los datos de precios recolectados, para la identificación de patrones significativos en la variación de precios de productos.
6. Desarrollar el módulo de visualización y análisis de precios con herramientas de BI, que presente de forma integrada los datos obtenidos mediante price scraping y los resultados del análisis realizado con aprendizaje automático.
7. Evaluar el rendimiento y precisión de la aplicación para identificar áreas de mejora y sentar una base sólida para desarrollos futuros.

1.3. Justificación

En el panorama empresarial contemporáneo, la información se ha erigido como un activo estratégico fundamental para aquellas organizaciones que aspiran a mantener su rentabilidad y expandir su cuota de mercado. La calidad, la accesibilidad y la oportunidad en la obtención de información definen la agilidad y la eficacia de las decisiones empresariales, factores cruciales para sostener la competitividad en un entorno dinámico y globalizado.

Esta realidad adquiere una relevancia particular en Venezuela, una economía que, según datos del Observatorio Venezolano de Finanzas (OVF, 2025) indica que la tasa de inflación en febrero de 2025 registró una significativa aceleración al marcar el aumento de precios mensual 12,8% y anualizado 117%. Este escenario económico, caracterizado por un mercado volátil e inestable, aunado al crecimiento exponencial del comercio electrónico y las tiendas virtuales, evidencia la necesidad que tienen las empresas del sector *retail* alimenticio, y otros sectores económicos de ajustar sus precios con una frecuencia mucho mayor que en mercados estables.

Entonces, para las organizaciones que operan en este contexto venezolano, mantenerse informadas sobre las dinámicas del mercado representa un desafío considerable, especialmente ante la fuerte fluctuación económica. La necesidad de procesar grandes volúmenes de datos y discernir patrones que permitan comprender las tendencias del sector

retail de alimentos puede convertirse en una tarea abrumadora si se realiza de forma manual. Por consiguiente, se torna indispensable la adopción de herramientas tecnológicas que permitan responder con agilidad a las fluctuaciones del mercado, recopilando y analizando la información relevante disponible en la web de manera eficiente y presentándola de forma consolidada y accesible. Esto libera recursos valiosos y facilita una toma de decisiones más fundamentada y oportuna.

Es precisamente en este contexto donde el presente trabajo se presenta como una oportunidad valiosa, que va más allá de la simple comparación de precios, al integrar el *price scraping* para la extracción automatizada de datos de precios de competidores, aplicar algoritmos de aprendizaje automático sobre estos datos recolectados para identificar patrones, predecir tendencias y comprender mejor las estrategias de fijación de precios, para luego presentar esta información de forma intuitiva a través de *dashboards* de BI, facilitando una toma de decisiones proactiva y basada en evidencia.

La combinación de las tecnologías mencionadas previamente facilita la generación de visualizaciones en tiempo real y la conversión de datos brutos en información clave proactiva, dotando a las empresas de la capacidad para ajustar sus estrategias de precios y reaccionar de manera eficaz a las dinámicas del mercado.

Algunas de las ventajas de este tipo de soluciones son: Optimización de precios y de la toma de decisiones, al tener información más certera. Ahorro de tiempo y de recursos, lo que repercute en una reducción de costes y en una mayor productividad de la compañía. Mayor comprensión del comportamiento de los usuarios y de su proceso de compra. Posibilidad de centrar la estrategia de ventas en el cliente y ofrecerle una mejor experiencia de compra, más personalizada y acorde con sus necesidades.

La implementación de una aplicación como la propuesta en este trabajo, genera múltiples ventajas significativas para las empresas. Permite una optimización dinámica de precios y decisiones estratégicas al proporcionar información precisa del mercado (Intelligence Node, s.f.), así como un incremento en la eficiencia y una reducción de costos mediante la automatización de la recopilación y el análisis de datos (Boring Owl, 2023). Además, facilita un amplio conocimiento del comportamiento del cliente y del mercado al identificar patrones y tendencias (Kaizen Institute, s.f.), lo que a su vez posibilita el desarrollo

de estrategias de venta centradas en el cliente y experiencias personalizadas (ResearchGate, 2019).

Finalmente, este trabajo no solo representa un aporte significativo al ámbito empresarial venezolano, sino también al académico dentro del Departamento de Computación de la Facultad de Ciencias y Tecnología de la Universidad de Carabobo. Al explorar la aplicación de técnicas avanzadas como el *price scraping* y el aprendizaje automático en el contexto de la inteligencia de negocios, se establece una base sólida sobre la cual futuras investigaciones podrían desarrollarse, impulsando así el avance del conocimiento en la región.

Capítulo II. Marco Teórico

El propósito de este capítulo es introducir una recopilación de antecedentes, proyectos y trabajos similares, los cuales sirvieron de apoyo y sustento a esta investigación, para seguidamente agregar las bases teóricas y términos usados en este ejercicio académico.

2.1. Trabajos relacionados

Como parte de la elaboración de este trabajo, se realizó una revisión bibliográfica de estudios previos referentes al tópico de esta investigación. Esto con el fin de identificar los avances más recientes de las herramientas y métodos usados en la extracción de datos mediante *price scraping* y, además, reconocer los elementos que podrían ocasionar problemas e inconvenientes durante el desarrollo de la aplicación. Las experiencias obtenidas de estos trabajos nutren la presente investigación, puesto que sirven como guía para la realización exitosa de los objetivos propuestos.

Específicamente en relación con los desafíos legales y éticos, Khder (2021) en su artículo titulado "*Web* Scraping *or Web Crawling; State of Art, Techniques, Approaches and Application*" presenta una visión única e integral sobre los desafíos legales y éticos inherentes al *web scraping*, enfatizando que, aunque es una poderosa herramienta para el análisis competitivo, su implementación debe respetar normativas como los términos de servicio de los sitios web, las políticas de robots.txt y leyes como el *Computer Fraud and Abuse ACT* (CFAA). También advierte sobre la extracción de datos personales sin consentimiento, la infracción de derechos de autor y el impacto por sobrecarga en los servidores.

Su contribución fue primordial para el trabajo propuesto, ya que establece un marco de responsabilidad legal y ética, lo cual es primordial para el desarrollo de una solución sostenible y conforme a los estándares emergentes en el mundo digital.

Asimismo, Brown et al. (2024) en su trabajo "*Web Scraping for Research: Legal, Ethical, Institutional, and Scientific Considerations*" destaca que el *web scraping* en investigación demanda una reflexión profunda sobre sus dimensiones éticas e institucionales, especialmente en lo que respecta al respeto por la privacidad y protección de las comunidades afectadas. Subraya la importancia de adoptar prácticas proactivas para minimizar impactos

negativos, como evitar la sobrecarga de servidores mediante técnicas de *rate limiting*, y asegurar que los datos recolectados sean manejados con altos estándares de seguridad para impedir filtraciones o usos indebidos. Por otra parte, Aishwarya Lakshmi (2025) en su artículo web "*Ethical Web Scraping: A Practical Guide to Responsible Data Collection*" enfatiza que una práctica ética de *scraping* implica no solo el cumplimiento normativo, sino también transparencia en la recolección y el uso responsable de los datos, recomendando la implementación de mecanismos de consentimiento siempre que sea posible y la exclusión explícita de datos sensibles o personales para proteger a los usuarios.

Ambos enfoques aportaron al presente proyecto un marco ético y técnico que garantiza una extracción de datos responsable y segura, protegiendo la privacidad y maximizando la confiabilidad de la información recolectada. Lo cual facilita el desarrollo de una solución ajustada a estándares legales y sociales, asegurando su aceptación y sostenibilidad en contextos de investigación.

Por otra parte, se tiene el estudio de Abodayeh et al. (2023) titulado "*Web Scraping for Data Analytics: A BeautifulSoup Implementation*". Los autores presentan la implementación de un *web* scraper basado en la biblioteca BeautifulSoup de Python, con el objetivo de recopilar y analizar información de productos disponibles en la plataforma Amazon. Su metodología comprendió la extracción de datos relevantes de hasta cinco páginas de resultados para productos específicos, seguida del despliegue de los hallazgos a través de una interfaz gráfica de usuario desarrollada con PySimpleGUI para facilitar la interacción, y la visualización de los datos mediante gráficos generados con la biblioteca Matplotlib.

Sus resultados demostraron la aplicabilidad y la efectividad de las técnicas utilizadas para identificar los productos más económicos y mejor calificados dentro de las categorías analizadas, proporcionando una herramienta útil para la toma de decisiones de compra por parte de los usuarios. Asimismo, la implementación de una interfaz de usuario para la interacción con los resultados y la utilización de herramientas de visualización como Matplotlib son aspectos que guardan relación y contribuyen con este trabajo.

Seguidamente, el estudio de D'Souza, Agrawal, Desai y Joshi (2024), titulado "*Web Scraping based Product Comparison Model for E-Commerce Website*", lleva a cabo un modelo de comparación de precios de productos en plataformas de *e-commerce* mediante

web scraping. Incluyendo BeautifulSoup, Selenium y MongoDB, los autores integraron datos de múltiples sitios, generando visualizaciones interactivas y alertas personalizadas a los usuarios.

Asimismo, la relevancia del trabajo de D'Souza et al. radica en la prueba de cómo la combinación de múltiples herramientas de scraping permiten superar desafíos técnicos como la variabilidad de las estructuras de los sitios web evidenciando a su vez la importancia de una interfaz intuitiva para facilitar la toma de decisiones basadas en datos.

Adicionalmente, el trabajo realizado por Cobo, Benítez Baldion, Perdomo González y Novoa Mendoza (2025) titulado "*Web scraping* en supermercados para el seguimiento de precios de la cesta básica alimentaria", el cual desarrolló una solución basada en *web scraping* e inteligencia de negocios para monitorear los precios de la canasta básica alimentaria en cadenas de supermercados colombianas. Utilizando la librería Selenium en Python, recolectaron datos de precios, nombres de productos y captura durante cien días en cinco cadenas de *retail* (Éxito, Jumbo, D1, Makro y Carulla). Los datos, que previamente fueron almacenados en CSV (*Comma Separated Value*) se llevaron a hojas de cálculo y se analizaron mediante técnicas de aprendizaje automático para predecir tendencias. Además, se diseñaron tableros interactivos para visualizar información relevante junto con pronósticos a 10 días, facilitando la comparación entre tiendas y el análisis histórico.

En tal sentido, la contribución de dicho trabajo es clave, puesto que demostró las ventajas y la utilidad de la automatización eficiente de datos mediante *web scraping*, aportando también principios teóricos/prácticos sobre las herramientas y técnicas disponibles, así como un panorama actualizado de su desarrollo. Además, los resultados son una prueba práctica de cómo la visualización de datos mediante *dashboards* de *Power BI,* una herramienta de inteligencia de negocios, pueden ser implementados para la visualización y análisis de precios, superando la complejidad inherente a este tipo de información. Además, evidenciaron que la interactividad proporcionada por estos tableros facilita la comprensión y el uso de la información.

Por otra parte, los desafíos identificados en su investigación, representaron una advertencia crucial para la solución planteada. En concreto, la extracción de datos y su preprocesado llevaron consigo dificultades tales como la identificación de marcas y la

estandarización de unidades de medidas, que fueron abordadas en etapas tempranas del desarrollo del presente proyecto. Esto con el fin de asegurar la confiabilidad y consistencia de los datos recolectados.

Finalmente, el proyecto disponible en Kaggle titulado "*Retail Price Optimization*" (Harsh Singh, 2023) fue de gran utilidad para el desarrollo del modelo predictivo utilizado en esta investigación. Este trabajo ofreció un enfoque actualizado y práctico para el análisis y la optimización de precios minoristas mediante técnicas de *machine learning* y modelado predictivo, utilizando conjuntos de datos ya existentes. La metodología y estructura de este proyecto sirvieron como base para la implementación de los modelos analíticos en el presente estudio, favoreciendo la capacidad del sistema para responder a las exigencias del mercado en la optimización de precios dentro de entornos competitivos y dinámicos.

2.2. Bases teóricas

Con el objetivo de dar soporte al desarrollo de esta investigación, se presenta un resumen de los fundamentos teóricos que sustentan las funcionalidades y características del sistema propuesto. A continuación, se procede a describir cada uno de ellos.

2.2.1. *Web Scraping*.

Según lo que expresa D'Souza et al. (2024), se refiere a una técnica de extracción de datos, usada para recolectar datos de sitios web mediante algoritmos y *software*. Dichos datos, se almacenan en un formato estructurado, tales como hojas de cálculo o bases de datos. En otras palabras, esta tecnología permite recolectar datos, y almacenarlos de forma eficiente.

El *web scraping* se hace mediante un *web scraper* el cual rastrea y recoge mediante consultas HTML la información almacenada en la *web*, seguidamente la guarda en archivos CSV, que podrán ser analizados posteriormente. Dicho proceso parte desde la identificación del URL semilla de la cual se extrae la información. Luego, se hace uso de algún scraper como BeautifulSoup, Selenium o Scrapy, los cuales harán las consultas a las páginas mediante HTML. Para así, finalmente, realizar una búsqueda de toda la información deseada,

la cual se guardará en algún formato estructurado para su posterior análisis o almacenamiento en bases de datos (Cobo et al. 2024).

2.2.2. Price Scraping

Es una subárea del *web scraping* que se enfoca en la extracción de datos de precios de productos o servicios de la competencia desde diferentes sitios *web*. Todo esto, con el objetivo de monitorear el mercado y la competencia para así poder tener una visión completa de las dinámicas que rigen el mercado. También se le conoce como el *web scraping* de precios, porque se especializa en extraer datos específicos de precios de productos publicados en sitios web (Price2Spy, 2024).

2.2.3. Inteligencia de negocios o *Business Intelligence* (BI)

Se refiere a un conjunto de tácticas y recursos tecnológicos enfocados en la administración y el desarrollo de conocimiento a través del análisis de los datos que una organización ya posee. Esto significa que una empresa puede gestionarse tomando como base la información que genera su propia actividad. El *BI* busca satisfacer las necesidades de información tanto de los líderes como de los analistas, con el fin de ampliar la comprensión de cómo opera la empresa y así poder definir estrategias de negocio más acertadas. Por ejemplo, el *BI* permite guardar, juntar y examinar datos de los clientes para prever ventas o descubrir tendencias y patrones que podrían ser ventajosos, todo esto dentro de un sistema que facilita que la información se comparta entre los diferentes departamentos de la empresa (Joyanes Aguilar, L., 2020).

2.2.4. Dashboard

Few (2004, como se cita en Papadopoulou, 2023, p. 57) lo define como "una pantalla visual de la información más importante necesaria para alcanzar uno o más objetivos; consolidada y organizada en una sola pantalla para que la información pueda ser monitoreada de un vistazo". Los *Dashboards* son herramientas cuyo propósito es permitir a los usuarios

tomar decisiones informadas en un tiempo reducido, debido a que, resume toda la información recibida en gráficos interactivos que otorgan a los usuarios la capacidad de centrarse sólo en indicadores claves y métricas importantes que son necesarias para llevar a una organización al éxito.

2.2.5. Aprendizaje Automático

Para Pertuz (2022) “el aprendizaje automático es una rama de la inteligencia artificial que busca que un programa de computador aprenda de un conjunto de datos con los cuales se entrenó, y buscará identificar un patrón con el que puede realizar predicciones sobre nuevos datos”

2.2.6. Consideraciones éticas del *Web Scraping*

Mas allá de la implementación técnica, el *web scraping* ético se fundamenta en un conjunto de prácticas responsables que buscan minimizar el impacto negativo y respetar las normativas vigentes. Como señalan Khder (2021) y Brown et al. (2024), esto implica no solo adherirse a los términos del servicio de los sitios web, sino también proteger la privacidad de los datos, evitar la sobrecarga de los servidores y ser transparente en la recolección de información. A continuación, se definen mecanismos técnicos clave para implementar un *scraping* responsable

2.2.6.1. Robots.txt

Como señalan Chang y He (2025) es un archivo de texto utilizado por los administradores de sitios web para proveer instrucciones sobre el sitio web a *crawlers* o *scrapers* que se encuentren realizando acciones automatizadas. Su función principal es gestionar la carga en los servidores y proteger áreas sensibles, aunque su implementación conlleva desafíos técnicos y legales que deben ser considerados en procesos de *web scraping*.

2.2.6.2. *Rate limiting*

Es una técnica que consiste en controlar la frecuencia con la que un *scraper* realiza solicitudes a un sitio web, con el fin de evitar causar una carga excesiva que pueda afectar el rendimiento o incluso provocar la caída del servidor. Es importante señalar que, Brown et al. (2024) advierten que los mecanismos de limitación pueden generar efectos secundarios, como la pérdida de registros que podrían estar concentrados en periodos específicos, lo que puede afectar la representatividad y calidad de los datos recolectados. De esta manera, el uso adecuado de *rate limiting* es una práctica técnica esencial que equilibra la necesidad de obtener datos precisos con la responsabilidad de preservar la estabilidad y funcionamiento de las fuentes consultadas.

2.2.7. Inteligencia Artificial Explicable

Según López Martín (2023), la inteligencia artificial explicable (XAI) engloba los métodos y técnicas que hacen comprensibles para los humanos los resultados generados por la inteligencia artificial, a diferencia del enfoque de "caja negra", en el cual las decisiones del sistema no pueden ser explicadas por sus propios diseñadores. En este sentido, la clasificación de dichos métodos no se limita a una única forma, sino que, como proponen Dib y Capus (2025), se puede organizar en torno a un marco de tres dimensiones clave.

Este marco distingue, en primer lugar, entre explicaciones locales, que justifican predicciones individuales, y globales, que ofrecen una visión general del comportamiento del modelo. La segunda dimensión es la de intrínseco vs. extrínseco, donde la explicabilidad intrínseca proviene de modelos transparentes por naturaleza (como la regresión lineal), mientras que la extrínseca utiliza métodos *post hoc* para interpretar "cajas negras" (como el bosque aleatorio). Finalmente, la tercera dimensión que aborda es la de específico del modelo vs. agnóstico del modelo. Los métodos específicos están diseñados para una arquitectura particular, a diferencia de los métodos agnósticos, que son versátiles y pueden aplicarse a cualquier tipo de modelo sin conocer su estructura interna (Dib & Capus, 2025).

Capítulo III. Marco Metodológico.

A continuación, se detallan las metodologías empleadas en el desarrollo de la investigación, utilizadas para orientar tanto la investigación como la construcción del sistema propuesto. Para la investigación, se ha optado por la metodología de Investigación-Acción que permite un enfoque colaborativo y adaptativo para abordar problemas complejos. Mientras que, para el desarrollo del software, se ha seleccionado la metodología ágil *eXtreme Programming* (XP), complementada con el proceso de *Knowledge Discovery in Databases* (KDD) que guía la extracción, procesamiento y análisis de datos. Estas herramientas metodológicas aseguran que la investigación se desarrolle de manera estructurada, eficiente y adaptable a los cambios que puedan surgir durante su ejecución.

3.1. Metodología Investigación-Acción

El presente proyecto se desarrolla bajo la metodología de Investigación-Acción (IA), la cual proporciona un enfoque alternativo a la investigación tradicional, donde se enfatiza que la investigación no se limitará a solo estudiar a los grupos afectados, sino que contribuye a su vez en una solución. Esta metodología surge formalmente gracias al psicólogo Kurt Lewin en 1946, tras la publicación de su libro "*Action Research and Minority Problems*".

La Investigación-Acción es un enfoque colaborativo de indagación que permite a las personas tomar medidas sistemáticas para resolver problemas específicos. Además de desarrollar soluciones efectivas mediante un ciclo de observar, reflexionar y actuar, ajustándose a la complejidad del problema (Stringer,2007). Este método se adapta a las necesidades del *price scraping* en un mercado volátil, como lo es el mercado venezolano. Además, a diferencia de los enfoques de investigación tradicionales, que a menudo siguen un plan rígido, la Investigación-Acción permite ajustar las estrategias y los métodos a medida que se obtiene nueva información y se comprende mejor el problema.

a. Es participativa y democrática: Debido a que involucra activamente a todos los actores relevantes (que bien pueden ser comunidades, organizaciones, individuos afectados) en todas las etapas del proceso investigativo.

b. Pone en práctica el conocimiento en acción: No se limita a la generación de conocimiento teórico o abstracto, sino que este conocimiento se lleva a cabo en pruebas de ciclos iterativos de acción, observación y reflexión, además de ajustarse en base a los resultados obtenidos.

c. Es emergente y evolutivo: No sigue un plan rígido y predefinido, sino que busca implementar soluciones y definir estrategias, en base a cambios imprevistos tras la profundización de la comprensión del problema, al igual que con la obtención de nuevos datos.

d. Aborda cuestiones prácticas sobre los conocimientos adquiridos: La investigación acción, cuenta a su vez con una actitud de cuestionamiento permanente. En donde, no solo busca aplicar el conocimiento, sino también cuestionar cómo se construye y a quién beneficia.

Finalmente, para guiar este proceso de investigación se adoptarán las fases establecidas por Lewin y adaptadas por McNiff y Whitehead (2002) en su obra "*Action Research: Principles and Practice*". A continuación, se describen dichas fases:

a. Planificación: En esta fase inicial, se identifica un área problemática, y se desarrolla un plan para abordar la situación. Donde en el marco de la presente investigación se define, como la falta de un sistema automatizado para la extracción y análisis de datos de los precios en el mercado.

b. Acción: Implica llevar a cabo el plan, a través de estrategias diseñadas, priorizando la adaptabilidad ante obstáculos imprevistos. Su aplicación, respecto al proyecto se refleja a través de la implementación de las prácticas de "*Extreme programming*" (XP) y "*Knowledge Discovery in Databases*" (KDD).

c. Observación: Consiste en la examinación sistemática de los procesos y los resultados, con el fin de recolectar datos referentes a los efectos producidos por la acción, así como también la identificación de posibles obstáculos. En esta investigación, la observación se refleja a través de la recuperación y validación de los datos de precios extraídos.

d. Reflexión: Es un proceso de aprendizaje, que permite comprender mejor el problema, identificar los aciertos y los errores, al igual que dar lugar a nuevas ideas. En esta fase se analizan críticamente los datos recolectados, evaluando el impacto de las acciones, con el fin de extraer conclusiones relevantes. En base al presente proyecto, la reflexión daría lugar a identificar áreas de mejora en la aplicación, ya sea en las técnicas de scraping o mejora en la visualización de los datos.

3.2. Metodología de desarrollo de software

Para el presente proyecto se optó por la metodología ágil *eXtreme Programming* (XP), la cual, de acuerdo con Kent Beck, Cynthia Andrés (2004), promueve prácticas como la programación entre pares, el uso de historias de usuario para capturar los requerimientos de manera sencilla, la integración continua y reuniones semanales, entre otros. Todo esto con el propósito de obtener retroalimentación continua y ajustar rápidamente la dirección del desarrollo, estas prácticas permiten abordar de manera dinámica y flexible los cambios en los requerimientos, asegurando que el software se desarrolle de forma robusta y eficiente.

Cabe destacar que el ciclo de desarrollo de XP, consta de las siguientes etapas (Canós, Letelier y Penadés, 2006):

- El cliente identifica las características o funcionalidades que aportan valor al usuario final o negocio.
- Se estima el esfuerzo necesario para su implementación.
- El cliente prioriza las funcionalidades seleccionadas en base a su aporte y limitantes de tiempo.
- El equipo de desarrollo lleva a cabo las funcionalidades seleccionadas.
- Regresa al paso inicial.

Mientras que, el ciclo de vida ideal de XP consiste en las siguientes fases:

a. Exploración: Se evalúa el alcance del proyecto, llevando a cabo las historias de usuario.

b. Planificación de la entrega: Se hace una estimación de los esfuerzos necesarios llevados a cabo en el desarrollo, al igual que se define un cronograma para las entregas.

c. Iteraciones: Ejecuta los pasos del ciclo de desarrollo mencionados con anterioridad.

d. Producción: El software resultante se pone a prueba en un entorno real.

e. Mantenimiento: Revisión continua del estatus del desarrollo, a la par con las correcciones de errores y mejoras.

f. Muerte del proyecto: Cumplimiento de todas las necesidades del cliente, sin funcionalidades adicionales por agregar.

Además, debido a la naturaleza del proyecto, que implica la extracción y análisis de grandes volúmenes de datos, así como visualización en tiempo real mediante herramientas de inteligencia de negocios como *Power BI*, es necesario implementar también procesos referentes al análisis de datos. En este sentido, se utiliza el proceso de *Knowledge Discovery in Databases* (KDD). El cual tiene como objetivo preparar la data mediante la recopilación y limpieza de grandes volúmenes de datos, para posteriormente convertirlos en información útil, que podrá llevar a cabo la identificación de patrones y tendencias que aporten valor estratégico.

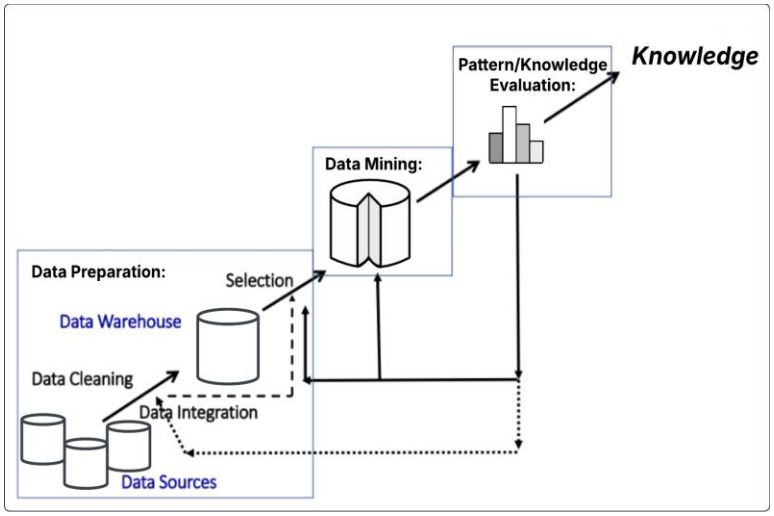
Según los autores Jiawei Han, Jian Pei y Hanghang Tong en su libro "*Data Mining: Concepts and Techniques*" (2022) el proceso KDD se compone de los siguientes pasos iterativos (Ver Figura 1):

a. Preparación de los datos:
    - Se deberá realizar una limpieza con el fin de eliminar el ruido y la inconsistencia entre los datos.
    - Efectuar la integración de los mismos en caso de que provengan de múltiples fuentes.
    - Transformar los datos mediante normalización o discretización, para su posterior uso y análisis.
    - Seleccionar los datos más relevantes para el análisis.
b. Minería de datos: Parte esencial del proceso donde se aplican técnicas para detectar y extraer patrones o modelos entre los datos.

c. Evaluación de patrones/modelos: Se determina si los patrones encontrados contienen información relevante que pueda contribuir en la interpretación de los datos.

d. Presentación del conocimiento: los resultados extraídos se presentan de manera clara para los usuarios, haciendo uso de técnicas avanzadas de representación de información, como gráficos, reportes detallados o tableros interactivos, entre otros.

Figura 1.
Esquema de funcionamiento



Nota. Adaptado de Han, J., Pei, J., & Tong, H. (2022)

Cabe destacar que, la relación entre XP y KDD es complementaria, esto debido a que XP al ser una metodología de desarrollo ágil, proporciona la base para la construcción del sistema de extracción y análisis de datos, mientras que KDD, como proceso, se encarga de la depuración, transformación y modelado de los datos para su interpretación estratégica. Gracias a XP se facilita la implementación ágil de cada fase de KDD, asegurando que la preparación, minería y evaluación de los datos sean optimizadas mediante pruebas continuas y mejoras incrementales.

Asimismo, la integración con Investigación-Acción establece un marco de trabajo que guía la integración de XP y KDD, a través de su ciclo iterativo de planificación, acción, observación y reflexión. En la fase de planificación de la IA, se definen los objetivos del

sistema; en la fase acción, se ejecutan las iteraciones de XP, incorporando los pasos de KDD; en la fase de observación se recopilan datos sobre el rendimiento del sistema, y finalmente en la fase de reflexión, se analizan los resultados. Gracias a este enfoque, la IA, XP y KDD se combinan en un proceso que impulsa la continua evolución del sistema.

3.3. Fases metodológicas del proyecto

Para dar cumplimiento a los objetivos de esta investigación, el desarrollo metodológico se estructuró en un plan de siete fases secuenciales que guiaron la ejecución del proyecto.

Tabla 1.
Fases metodológicas del proyecto

Objetivos	Fases	Actividades
Extracción y estructuración de datos	Análisis comparativo de tecnologías de *scraping*	Se realizó un análisis comparativo de las metodologías de *Web Scraping,* como es la ingeniería inversa de las comunicaciones cliente-servidor para la identificación y consumo de APIs no documentadas. Así como también, la automatización de navegadores para la renderización y extracción de contenido dinámico.
	Planificación de la extracción	La selección de las fuentes de datos web se realizó bajo la premisa de garantizar la disponibilidad de contenido relevante para la investigación y el estricto cumplimiento de sus políticas de uso, verificando que no exista una prohibición expresa a la extracción automatizada de información.
	Desarrollo del proceso de *Web Scraping*	Se llevó a cabo la codificación de los *scrapers* personalizados para cada sitio web, tomando en consideración los tiempos de espera y la gestión de errores. Así como también, la limpieza de los datos.
	Pruebas de rendimiento	Se evaluó el desempeño del proceso de *Web Scraping* e identificar las mejoras necesarias. Con el fin de ajustar el proceso según los resultados obtenidos.
Desarrollo del modelo predictivo	Preparación del *Dataset* y *Feature Engineering*	Se fusionaron los datos obtenidos gracias al *scraper* con los datos del histórico de ventas del *retailer*, mediante técnicas de *product matching* con el objetivo de homologar los productos. Una vez unificada la data, se procedió a enriquecerla mediante la creación de nuevas variables que capturan relaciones y patrones inexistentes en los datos originales.

Objetivos	Fases	Actividades
	Entrenamiento y selección del modelo	Para seleccionar el modelo predictivo más adecuado, se procedió a segmentar los datos en un conjunto de entrenamiento y otro de prueba, reservando este último para una evaluación final. Posterior a ello, el rendimiento de cada modelo ajustado se midió contra el conjunto de prueba para determinar cuál ofrece la mayor precisión predictiva.
	Evaluación de los modelos de aprendizaje automático.	Se calculó y analizó las métricas de validación (R^2 y MAE) sobre el conjunto de prueba para cuantificar la exactitud del modelo.
Elaboración de tableros de control	Selección de las herramientas de visualización	Se realizó un análisis de las posibles herramientas, con el fin de seleccionar aquella que ofrezca un mayor nivel de flexibilidad y personalización para garantizar la correcta interpretación de los resultados y la usabilidad de la herramienta final.
	Implementación de la aplicación web	Se procedió con la implementación del *frontend* de la aplicación, la cual incluyó el desarrollo de componentes de visualización interactivos, tales como tablas de datos dinámicas. Asimismo, se estableció la comunicación con el *backend* a través de una API, permitiendo al usuario final consultar en tiempo real las predicciones del modelo y explorar el conjunto de datos más reciente obtenido del proceso de *scraping*.

Capítulo IV. Resultados.

Este capítulo detalla el proceso de desarrollo y los resultados obtenidos en la construcción de la aplicación para comparar precios de la competencia del sector *retail* de alimentos, tomando como caso de estudio al *retailer* colaborador. Se presenta la implementación por versiones del pipeline completo, que integra el *scraping* de precios de tres competidores con los datos internos del *retail* para alimentar un modelo *de Machine Learning*. El resultado es un *dashboard* interactivo que genera y presenta recomendaciones de precios optimizados, cuyo rendimiento y utilidad son rigurosamente evaluados.

4.1. Visión general y levantamiento de requisitos

El punto de partida del proyecto fue comprender a fondo el contexto del problema y definir claramente las necesidades de los usuarios finales para garantizar que la solución tecnológica estuviera alineada con los objetivos.

El sistema fue diseñado para operar en el sector de *retail* de alimentos en Venezuela, un mercado caracterizado por su alta volatilidad y la creciente relevancia del comercio electrónico. Para asegurar una inteligencia competitiva robusta, se identificaron y monitorearon tres competidores con una fuerte presencia en línea:

- Kromi Market: Una cadena de supermercados consolidada.
- Kalea: Un *retailer* especializado en alimentos y bebidas.
- TuZonaMarket: Un *e-commerce* emergente en el sector.

El monitoreo de estos actores se planificó para ejecutarse de manera automatizada, estableciendo un ciclo de recolección de datos cada cuatro días. Asimismo, esta tarea fue realizada en un horario optimizado (02:00 AM - 06:00 AM) para minimizar el impacto en los servidores de origen. Cada competidor requirió un enfoque técnico distinto debido a la heterogeneidad de sus plataformas web, desde la automatización de navegadores con *Selenium* para sitios dinámicos hasta el consumo de *APIs* internas.

4.1.1. Identificación de perfiles de usuarios y modelado de requisitos

Tras un proceso de levantamiento de información que incluyó entrevistas con la gerencia y observación directa de los procesos de fijación de precios con el *retailer*

colaborador, se identificaron tres perfiles de usuario distintos, cada uno con responsabilidades y necesidades específicas a las que la aplicación debe responder. Gracias a esta segmentación, se asegura que la solución diseñada se integre de manera eficaz en los flujos de trabajo existentes.

a) Perfil 1: Administrador del Sistema

- Descripción: Este perfil es responsable del mantenimiento, configuración y operatividad de la aplicación. Es un rol desempeñado por un miembro del equipo de TI y sus actividades se orientan a que el sistema funcione de manera fiable, segura y eficiente.
- Responsabilidades: Este rol es responsable de administrar la configuración y programación de los *scrapers* garantizando la integridad y disponibilidad de la base de datos, así como gestionar los permisos de acceso de los usuarios al sistema.
- Interacción con el Sistema: Su interacción con el sistema se centra en el *backend* y en la tabla de configuración de la base de datos. A diferencia de los usuarios analíticos, su rol no es consumir los *dashboards*, sino garantizar la operatividad de la infraestructura subyacente que los alimenta.

b) Perfil 2: Analista de datos

- Descripción: Su objetivo principal es garantizar la validez estadística de los resultados del sistema. Para ello, este perfil es responsable de verificar la calidad de la información de entrada y auditar el rendimiento de los modelos predictivos.
- Responsabilidades: Es responsable de auditar y optimizar el sistema predictivo. Su labor comienza con la validación del rendimiento del modelo a través de métricas como MAE y R^2. Además, se encarga de analizar las causas de los errores de predicción atípicos (*outliers*), lo que le permite proponer mejoras al algoritmo o el diseño de nuevas visualizaciones para revelar correlaciones no evidentes en la información.
- Interacción con el sistema: Es un usuario avanzado de la aplicación, el cual se encarga de utilizar el "Explorador de Datos Históricos" y la función de "Exportación a CSV"

con el objetivo de realizar análisis profundos y auditorías de datos. También interactúa con las secciones de interpretación del modelo (como los análisis SHAP) para diagnosticar el comportamiento del algoritmo.

c) Perfil 3: Analista de precios

- Descripción: Este usuario final se encarga de aprovechar al máximo la inteligencia generada por la aplicación para la toma de decisiones. Su objetivo es emplear estos análisis para definir una estrategia de precios que maximice la rentabilidad y mantenga una ventaja competitiva sostenible.
- Responsabilidades: Se encarga de monitorear diariamente las recomendaciones de precios, al igual que identificar las oportunidades de optimización de margen más significativas. También, se ocupa del uso del simulador para evaluar el impacto de posibles movimientos del mercado, con el fin de comunicar y justificar las decisiones de cambio de precios a la gerencia.
- Interacción con el sistema: Es el principal usuario del *dashboard* de "Análisis Predictivo". Se enfoca en la tabla de recomendaciones, los gráficos de oportunidades y el simulador de escenarios. Su interacción es de alto nivel, buscando respuestas directas a preguntas de negocio sin necesidad de profundizar en los detalles técnicos del modelo o la calidad de los datos (confiando en el trabajo del Analista de Datos).

A partir de las necesidades de estos roles, se crearon Historias de Usuario (HU) que fueron priorizadas en un *backlog* de producto. Para establecer la jerarquía de las tareas, se realizó un análisis que consideró el valor de negocio, el esfuerzo de implementación, las dependencias y el riesgo asociado. A continuación, la Tabla 2 presenta de forma resumida las HU y el *backlog* detallado que siguió el desarrollo incremental:

Tabla 2.
Historias de usuarios

ID	Como...	Quiero...	Para...	Módulo Relacionado
HU-01	Analista de Precios	Ver una tabla que compare mi precio, el de la competencia y el precio recomendado	Identificar rápidamente las recomendaciones clave.	Análisis Predictivo
HU-02	Analista de Precios	Visualizar un ranking de los productos con mayor oportunidad de ajuste de precio	Enfocar mi atención en las decisiones de mayor impacto para la rentabilidad.	Análisis Predictivo
HU-03	Analista de Precios	Simular el impacto en el precio recomendado si cambio los precios de mis competidores	Prepararme para escenarios futuros y desarrollar contra-estrategias proactivas.	Análisis Predictivo
HU-04	Analista de Datos	Ver la evolución histórica del precio de un producto, comparando mi empresa con la competencia	Identificar tendencias y validar patrones que el modelo podría estar aprendiendo.	Datos Históricos
HU-05	Analista de Precios	Comparar la distribución general de precios de una categoría contra la competencia	Entender nuestro posicionamiento estratégico general (premium, económico, etc.).	Datos Históricos
HU-06	Analista de Datos	Filtrar todos los análisis por múltiples criterios (producto, fecha, etc.)	Realizar investigaciones profundas y segmentadas para auditorías de datos.	Todos los Módulos
HU-07	Analista de Datos	Acceder a una tabla con todos los datos históricos recolectados	Poder validar la información y diagnosticar errores directamente desde la fuente	Tabla de Productos
HU-08	Analista de Datos	Exportar los datos filtrados a un archivo CSV	Realizar análisis estadísticos avanzados en otras herramientas (Python, R, Excel).	Tabla de Productos
HU-09	Administrador	Que el sistema recolecte los datos de la competencia de forma automática y periódica	Asegurar que los análisis se basen siempre en información reciente y relevante.	Backend (ETL)

Product Backlog

Seguidamente, la Tabla 3 presenta el *Product Backlog* completo del proyecto, con cada ítem mapeado a la HU correspondiente y priorizado para su desarrollo.

Tabla 3.
Backlog del producto

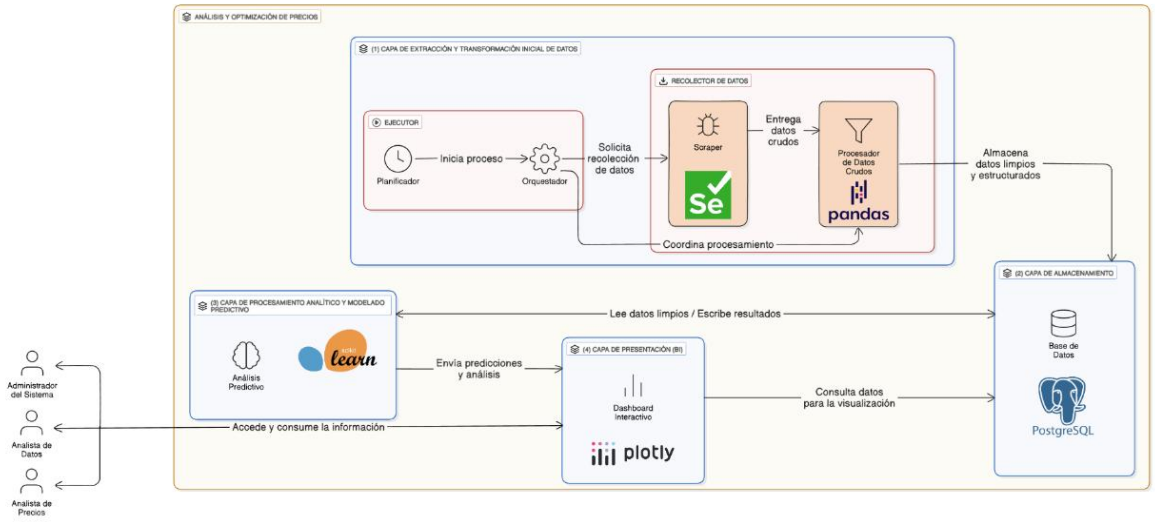
ID	Historia de Usuario (Resumen de la funcionalidad)	Prioridad	Versión Asignada
HU-09	Configurar y automatizar la recolección periódica de datos de la competencia.	Muy Alta	Versión 1
N/A	Desarrollar el modelo de *Machine Learning* para la predicción de precios. (tarea técnica)	Alta	Versión 2
HU-01	Mostrar tabla comparativa de precios (propio, competencia, recomendado).	Alta	Versión 3
HU-02	Visualizar gráficos de las principales oportunidades de optimización de precios.	Alta	Versión 3
HU-04	Mostrar gráfico de evolución histórica de precios para análisis de tendencias.	Media	Versión 3
HU-07	Proveer una vista tabular completa de los datos históricos con búsqueda.	Media	Versión 3
HU-08	Permitir la exportación de datos filtrados a formato CSV.	Media	Versión 3
HU-03	Implementar un simulador de escenarios "*what-if*" para análisis proactivo.	Media	Versión 3
HU-05	Mostrar gráficos de distribución de precios para análisis de posicionamiento.	Baja	Versión 3
HU-06	Implementar filtros avanzados y combinables en todos los módulos visuales.	Baja	Versión 3
N/A	(Tarea de Calidad) Realizar pruebas integrales de rendimiento y precisión del sistema.	Crítica	Versión 4

4.2. Arquitectura del sistema y *stack* tecnológico

A partir de los requisitos definidos, se diseñó una arquitectura modular que comprende un conjunto de tecnologías probadas en el ecosistema de ciencia de datos para garantizar la escalabilidad, mantenibilidad y eficiencia de la solución.

La Figura 2 presenta la arquitectura del sistema, la cual está estructurada en cuatro módulos funcionales: (1) Extracción y transformación inicial de datos, (2) Almacenamiento de datos, (3) Procesamiento analítico y modelado predictivo, y (4) Presentación. A continuación, se detalla la secuencia de operación y las atribuciones de cada uno de estos módulos.

Figura 2.
Diagrama de arquitectura



4.2.1. Módulo 1: Extracción y transformación inicial de datos

El objetivo fue automatizar la recolección de datos de precios de productos desde los sitios web de los *retailers* seleccionados, realizar una limpieza y estandarización inicial, y prepararlos para su almacenamiento.

Componentes y Tecnologías

- Orquestador de Tareas: Se implementó un *script* principal en Python 3.9 que coordina la ejecución de todo el proceso, el cual realiza automatización periódica gestionada mediante el uso de un *cron* en un servidor Linux, programado para ejecutarse diariamente en un horario de bajo tráfico.
- Motor de *Web Scraping:*
    - *Requests*: Utilizado para realizar peticiones HTTP a las páginas web. Es la opción seleccionada por su eficiencia para obtener el contenido HTML de sitios estáticos.
    - *Selenium* con *ChromeDriver*: Empleado para interactuar con sitios web que cargan su contenido dinámicamente mediante *JavaScript*. Simula la

navegación de un usuario para acceder a la información de precios que no está presente en el HTML inicial.

- Procesador de Datos:
    - Pandas: Utilizado para estructurar los datos extraídos en un *DataFrame*, facilitando las operaciones de limpieza, transformación y normalización.

Flujo de Proceso:

1. Inicio Automatizado: El *cron job* invoca el *script* orquestador.
2. Recolección (*Scraping*): El *script* itera sobre una lista predefinida de URLs de *retailers*. Para cada una, determina la estrategia de *scraping* adecuada (*Requests* para páginas estáticas, *Selenium* para dinámicas).
3. Extracción de Datos Crudos: Se extraen los campos de interés (ej. "Harina P.A.N. 1kg", "Bs. 85.50").
4. Transformación y Limpieza (ETL Ligero): Los datos extraídos son cargados en un *DataFrame* de Pandas, a estos se les aplican las siguientes transformaciones:
    - Normalización de Precios: Conversión de precios en formato de texto (ej. "Bs. 85,50") a un tipo numérico de punto flotante (ej. 85.50).
    - Estandarización de Unidades: Extracción y unificación de unidades de medida (ej. "1kg", "1 kg", "1000 gr" se convierten a una tupla estandarizada (1, 'kg')).
    - Asignación de Metadatos: Se añade la fecha y hora de la extracción y el identificador del *retailer* a cada registro.

Salida del Módulo: Un *DataFrame* de Pandas limpio, estructurado y normalizado, listo para ser almacenado en la base de datos.

4.2.2. Módulo 2: Almacenamiento de datos

Objetivo: Almacenar de forma centralizada, segura y persistente los datos históricos de precios, así como los metadatos del sistema, sirviendo como la única fuente válida (*single source of truth*) para los módulos subsecuentes.

Tecnología Principal:

- PostgreSQL 14: Se seleccionó este Sistema de Gestión de Bases de Datos Relacional (RDBMS) por su robustez, su soporte para tipos de datos complejos, sus capacidades

transaccionales (ACID) que garantizan la integridad de los datos, y su alta integración con el ecosistema de Python.

Esquema de datos:

Se diseñó un modelo relacional con tablas clave como:

- websites: Almacena información de los competidores (ID, name, URL).
- productos: Catálogo maestro de productos (ID, name, product_type_id).
- precios_historicos: Tabla transaccional que registra cada precio extraído (ID, product_id, website_id, price, scrape_timestamp, udm_id, currency).
- config_parameters: Se encarga de almacenar los parámetros de configuración de la aplicación (ID, config_key, config_value).
- users: Registra las credenciales de acceso (username, password, role).

Flujo de proceso:

1. Gestión de conexión: Se utiliza la librería SQLAlchemy en Python para gestionar el pool de conexiones a la base de datos de PostgreSQL, optimizando el rendimiento y la seguridad.
2. Carga de datos (Load): El *DataFrame* limpio del Módulo 1 se inserta en la tabla preprocessed_products. a través de operaciones *INSERT*, garantizando así el almacenamiento permanente de los datos.

Salida del módulo: Los datos se encuentran almacenados y estructurados en la base de datos PostgreSQL, disponibles para ser consultados en cualquier momento.

4.2.3. Módulo 3: Procesamiento Analítico y Modelado Predictivo

Objetivo: Aplicar técnicas de minería de datos y aprendizaje automático sobre los datos históricos para descubrir patrones, segmentar productos y generar un modelo de recomendación de precios.

Componentes y tecnologías:

- Librerías de análisis: Pandas y NumPy para la manipulación y preparación de datos.
- Librerías de *Machine Learning*: Scikit-learn para el preprocesamiento, ingeniería de características, entrenamiento y evaluación de modelos.

- Serialización de modelos: Joblib para guardar los modelos entrenados en disco, permitiendo su reutilización sin necesidad de reentrenamiento.

Flujo de proceso (Pipeline de ML):

1. Consulta y creación de dataset analítico: Se extraen los datos de PostgreSQL, uniéndolos para crear una tabla plana (*analytical base table*) que contiene el historial de precios por producto, *retailer* y fecha.
2. Preprocesamiento Avanzado:
    - Homologación de productos (*Product Matching*): Se aplicaron algoritmos de similitud de cadenas, específicamente el token_set_ratio de la librería thefuzz, que es robusto al desorden de las palabras y a la presencia de subconjuntos comunes de términos. Todo ello, con el objetivo de agrupar variantes de nombres de un mismo producto provenientes de diferentes *retailers*.
3. Ingeniería de características (*Feature Engineering*): Se crearon nuevas variables predictivas a partir de los datos existentes, entre las más importantes generadas fueron: comp1_diff, comp2_diff, comp3_diff, las cuales tuvieron como objetivo calcular la diferencia numérica entre el precio del producto de la empresa y su diferencia con el competidor:
4. Modelado predictivo (Regresión): Se entrenó un modelo de regresión denominado *Random Forest Regressor*, el cual fue seleccionado por su capacidad para manejar relaciones no lineales y por su robustez para predecir un precio óptimo, utilizando las características creadas como variables de entrada.
5. Validación y serialización: El modelo se validó utilizando una estrategia de conjunto de prueba ciego (*hold-out*), simulando un escenario de despliegue real con datos temporales. Además, también fue evaluado su rendimiento mediante el uso de métricas clave como MAE y R^2. Adicionalmente, la validación se complementó con un profundo análisis cualitativo mediante técnicas de *Inteligencia Artificial Explicable* (XAI) y un diagnóstico de errores, asegurando no solo la precisión del modelo sino también su interpretabilidad y su capacidad para generar *insights* de negocio.

Salida del Módulo:

- Un conjunto de datos enriquecido con el precio recomendado.
- Un archivo de modelo serializado (.joblib) listo para ser cargado y utilizado por la capa de presentación.

4.2.4. Módulo 4: Presentación y Visualización (BI)

Objetivo: Proporcionar una interfaz web interactiva para que los usuarios finales (analistas de precios, gerentes de categoría) puedan consumir los resultados del análisis, explorar los datos y obtener *insights* para la toma de decisiones.

Tecnología principal:

- Plotly: Librería para la creación de aplicaciones webs con gráficos interactivos y de alta calidad (gráficos de líneas, barras, diagramas de violín).

Funcionalidades Implementadas:

1. Dashboard de analisis de precios:
    - Entrada: Carga el modelo serializado del Módulo 3 y los datos más recientes de la base de datos.
    - Visualización: La interfaz se implementó para mostrar una tabla comparativa principal que desglosa para cada producto su precio actual, el precio promedio de la competencia y el precio recomendado por el modelo. Adicionalmente, se empleó un sistema de codificación por colores para resaltar visualmente las mayores oportunidades de optimización, ya sea para incrementar o reducir precios.
2. Dashboard historico:
    - Entrada: Realiza consultas directas a la base de datos PostgreSQL.
    - Visualización: La herramienta de visualización de datos ofrece gráficos interactivos que facilitan un análisis de precios multifacético. A través de un gráfico de líneas, los usuarios pueden comparar la evolución histórica de los precios de un producto entre distintos *retailers*. Además, un diagrama de violín permite analizar la distribución de precios en una categoría de

productos específica. Finalmente, un gráfico de barras ayuda a identificar de forma consistente al *retailer* que mantiene los precios más bajos o más altos.

3. Tabla de productos:
    - Funcionalidad: Presenta una vista tabular completa de la tabla precios_historicos con filtros dinámicos (por fecha, producto, *retailer*).
    - Salida: Incluye un botón para exportar la vista filtrada a un archivo .csv para análisis más detallados o personalizados en herramientas como Microsoft Excel.
4. *Dashboard* de análisis del modelo:
    - Entrada: Carga el modelo del Módulo 3 y las predicciones realizadas por el mismo.
    - Visualización:
        - Análisis Exploratorio de Datos (*EDA):* Presenta gráficos interactivos diseñados para que el analista de datos realice un Análisis Exploratorio de Datos (EDA) exhaustivo. Mediante histogramas, es posible analizar la distribución y frecuencia de variables clave como precios y ganancias. Además, una matriz de correlación permite visualizar la relación lineal entre todas las variables del modelo, mientras que una tabla descriptiva ofrece un resumen estadístico con métricas fundamentales como la media, la desviación estándar y los cuartiles, facilitando así una comprensión profunda del conjunto de datos.
        - Evaluación del Rendimiento: Ofrece una visión clara de la precisión y fiabilidad del modelo a través de varias visualizaciones clave. Un gráfico de dispersión permite comparar directamente los valores predichos con los valores reales para medir la exactitud. Adicionalmente, un histograma de residuos ayuda a estudiar cómo se distribuyen los errores de predicción, lo que facilita la identificación de sesgos. Finalmente, un gráfico Q-Q sirve para verificar si los errores siguen una distribución normal, un supuesto fundamental para la validez de muchos modelos estadísticos.).

- Interpretabilidad del Modelo (*Explainable AI*): Proporciona herramientas de interpretabilidad para comprender las decisiones del modelo, permitiendo al usuario identificar las características con mayor impacto global en las predicciones de precios a través de un gráfico de resumen SHAP. Asimismo, ofrece la capacidad de analizar cómo varía una predicción al modificar el valor de una sola característica mediante un gráfico de dependencia parcial. Finalmente, facilita el desglose de la contribución de cada variable para una predicción individual de un producto específico, ofreciendo una visión completa de los factores que influyen en cada resultado.

5. Panel de administración:
    - Entrada: Carga los datos de configuración y usuarios de la base de datos.
    - Funcionalidad: Permite al administrador gestionar los usuarios y los parámetros del sistema.

Salida del Módulo: Una aplicación web funcional y accesible a través de un navegador, que centraliza el análisis y las recomendaciones generadas por el sistema.

4.3. Iteraciones del Desarrollo

La aplicación se desarrolló siguiendo una metodología incremental, organizada en versiones o ciclos de entrega. Cada versión tenía como objetivo la construcción y validación de un conjunto cohesivo de funcionalidades, permitiendo un progreso medible y la mitigación de riesgos de manera temprana.

4.3.1. Versión 1: Construcción del pipeline de datos y la capa de persistencia

El propósito de esta primera iteración fue establecer las bases del sistema, estableciendo su capacidad de recolectar datos de la competencia de forma automatizada, procesarlos para asegurar su calidad y almacenarlos de manera persistente y estructurada. Los objetivos específicos fueron:

1. Implementar y validar los extractores (*scrapers*) para las fuentes de datos seleccionadas.
2. Desarrollar un pipeline de limpieza y transformación de datos (ETL).
3. Diseñar e implementar el esquema de la base de datos relacional para el almacenamiento de los datos.
4. Verificar la correcta integración y funcionamiento de todos los componentes mediante pruebas funcionales y unitarias.

4.3.1.1. Implementación del pipeline de extracción, transformación y carga

Esta fase sigue las etapas canónicas del proceso de Descubrimiento de Conocimiento en Bases de Datos (KDD), enfocándose en la selección, preprocesamiento y transformación de los datos.

a) **Selección de fuentes y atributos de datos**:

Se identificaron y delimitaron las fuentes de datos primarias para el análisis:

- Fuentes externas (Competencia): Se seleccionaron los portales de *e-commerce* de tres cadenas de *retail* con significativa presencia en el mercado venezolano: Kromi Market, Kalea y TuZonaMarket.
- Fuente interna: Se utilizó un *dataset* histórico proporcionado por un *retailer* colaborador, conteniendo información de ventas y precios propios.
- Atributos de interés extraídos: Para cada producto en las fuentes externas, se extrajeron los siguientes atributos: Nombre del producto (ej. "Harina P.A.N. 1kg"), Precio de venta al público (ej. "Bs. 85,50"), Identificador del *retailer*, URL de la página del producto, Unidad de Medida (Udm) y Tipo de Moneda.

b) Preprocesamiento y Limpieza de Datos

Dado el carácter no estructurado de los datos web, esta etapa fue crítica para mitigar el principio de "*garbage in, garbage out*" (GIGO). Se partió de los datos recolectados durante un período de seis meses. Los primeros cinco meses constituyeron el conjunto de datos de entrenamiento para el desarrollo del pipeline y el futuro modelado, mientras que el sexto mes se reservó como conjunto de prueba ciego (*hold-out set*) para la validación final del modelo predictivo en la siguiente versión.

Se implementaron reglas de negocio y transformaciones programáticas para solventar las inconsistencias detectadas, como se resume en la Tabla 4.

Tabla 4.
Reglas de negocio implementadas

Problema Detectado	Estrategia de Solución Implementada
Nombres de producto inconsistentes	Se aplicó un algoritmo de *Product Matching* basado en la librería thefuzz para homologar productos equivalentes entre *retailers* (ej. "Harina P.A.N." vs. "Harina PAN 1kg").
Atributos embebidos en texto	Se utilizaron expresiones regulares (RegEx) para extraer y estructurar atributos clave como la marca, cantidad y unidad de medida desde el nombre del producto (ej. de "MAVESA MAYONESA 445 GR" se extrae marca: MAVESA, cantidad: 445, udm: GR).
Unidades de medida variables	Se normalizaron las unidades de medida a un formato estándar (kg, g, l, ml, u) para permitir comparaciones consistentes.
Precios faltantes	Se imputaron los valores faltantes con un marcador nulo (NaN) para su posterior tratamiento en la fase de modelado.

c) Transformación y enriquecimiento del *dataset*

Una vez asegurada la calidad y consistencia de los datos, la fase de transformación se enfocó en reestructurar y enriquecer el *dataset* para maximizar su valor predictivo.

- Ingeniería de características (*Feature Engineering*): En esta subetapa, se crearon nuevas variables diseñadas para capturar la posición competitiva del producto. Las características más relevantes generadas fueron comp1_diff, comp2_diff, comp3_diff las cuales son variables numéricas que representan la diferencia porcentual entre el precio propio y el de cada competidor, aportando una señal más potente para el modelo que los precios absolutos.
- Agregación de datos: Posteriormente, y con el objetivo de preparar el *dataset* para modelos de regresión, se realizó una agregación de datos a nivel de *product_id*. En este proceso, se calculó la suma total de las ganancias (total_ganancia) de cada producto. A su vez, se promediaron las variables de precios, incluyendo el precio del producto (precio_mes_actual), los precios de los tres competidores y las diferencias de precio (*comp_diff*) generadas anteriormente.

4.3.1.2. Diseño e implementación de la capa de persistencia

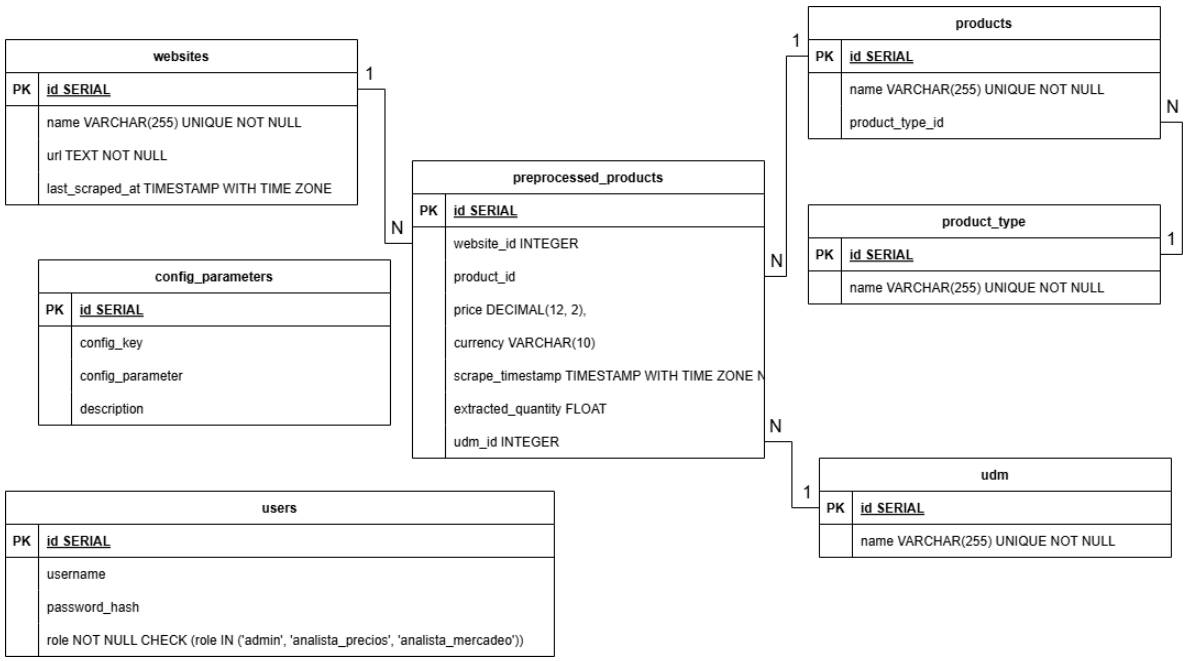
Para dar soporte al almacenamiento persistente, centralizado y seguro de los datos procesados, se diseñó e implementó un modelo de base de datos relacional en PostgreSQL.

a) Modelo Conceptual (Diagrama Entidad-Relación)

El diseño de la base de datos partió de la creación de un modelo conceptual, materializado en el Diagrama Entidad-Relación (Figura 3), el cual modela las entidades fundamentales del dominio del problema y sus interrelaciones.

Figura 3.

Diagrama E-R de la base de datos



De este modelo se desprenden las siguientes entidades principales:

- websites: Representa a los *retailers* monitoreados.
- *products*: Catálogo maestro de productos homologados.
- *product_type, udm*: Tablas catálogo para estandarizar tipos de producto y unidades de medida.
- *preprocessed_products*: Tabla transaccional que almacena cada dato de precio recolectado, constituyendo el histórico central.

- *config_parameters*: Almacena parámetros de configuración del sistema para evitar el uso de parámetros estáticos en el código.
- *users*: Proporciona el grupo de usuarios disponibles en la aplicación.

b) **Modelo lógico relacional y justificación del diseño**

A partir del modelo conceptual, se realizó la traducción a un esquema lógico relacional (Tabla 5), diseñado siguiendo las mejores prácticas de normalización de bases de datos.

Tabla 5.
Esquema relacional

Tabla	Campos y Restricciones	Relación (Cardinalidad)	Tabla referenciada	Descripción
websites	*id (PK, SERIAL), name (UNIQUE), url, last_scraped_at*		-	Almacena los sitios web de los *retailers* a monitorear.
product_type	*id (PK, SERIAL), name (UNIQUE)*		-	Tabla maestra para los tipos de producto (ej. Harina, Arroz).
udm	*id (PK, SERIAL), name (UNIQUE)*		-	Tabla maestra para las unidades de medida (ej. g, kg).
products	*id (PK, SERIAL), name (UNIQUE), product_type_id (FK)*	Muchos a Uno (N:1)	*product_type*(id)	Almacena los productos únicos y homologados.
preprocessed_ products	*Id (PK, SERIAL), product_id (FK), website_id (FK), price, scrape_timestamp,* udm_id *(FK), currency*	Muchos a Uno (N:1)	*products*(id)	Tabla de hechos que guarda cada registro de precio extraído.
		Muchos a Uno (N:1)	*websites*(id)	
		Muchos a Uno (N:1)	udm(id)	
config_param eters	*Id (PK, SERIAL), config_key (UNIQUE), config_value*		-	Almacena parámetros de configuración de la aplicación.
users	*username, password, role*			Tabla maestra para los distintos usuarios de la aplicación.

Es importante destacar que la estructura resultante no es arbitraria; por el contrario, respondió a decisiones de diseño deliberadas para garantizar la mantenibilidad y la integridad. En este sentido, se optó por un esquema altamente normalizado (Tercera Forma

Normal, 3FN) para prevenir anomalías de datos y minimizar la redundancia. Adicionalmente, el uso de claves primarias de tipo SERIAL optimizaron el rendimiento de las operaciones de unión (JOINs). Finalmente, la tabla config_parameters dotó al sistema de una flexibilidad crucial, permitiendo modificar su comportamiento sin necesidad de re-desplegar el código.

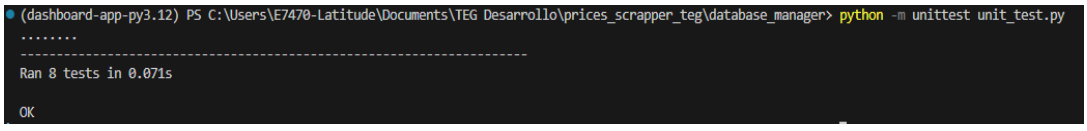
4.3.1.3. Pruebas y validación de la versión 1

Para asegurar la robustez y fiabilidad de los componentes desarrollados, se implementó una estrategia de pruebas continuas a lo largo de toda la iteración.

- Pruebas funcionales de los *scrapers*: Inicialmente, cada extractor fue ejecutado de forma aislada para verificar la correcta extracción de los campos requeridos (nombre, precio, *retailer*) y su capacidad para manejar paginación y contenido dinámico.
- Pruebas unitarias del gestor de dase de datos: Para validar la capa de persistencia, se desarrolló un conjunto de pruebas unitarias con el *framework unittest* de Python. Se utilizó la librería *mock* para aislar el gestor de la base de datos real, permitiendo una validación rápida y en memoria de sus funcionalidades. Esta serie de pruebas cubrió áreas críticas, incluyendo la gestión de la conexión, la creación del esquema y las operaciones CRUD sobre las entidades. Además, se hizo especial énfasis en la lógica de negocio más compleja, validando la función de inserción por lotes (*insert_preprocessed_products_batch*) y la correcta construcción de consultas SQL dinámicas (*get_preprocessed_products*).

La ejecución exitosa de este proceso de pruebas, cuya salida se muestra en la Figura 4, proporcionó una alta confianza en la fiabilidad del *pipeline* de datos y la capa de persistencia, concluyendo así de manera satisfactoria los objetivos planteados para la Versión 1.

Figura 4.
Salida de la suite de pruebas



4.3.2. Versión 2 Desarrollo del Modelo Predictivo

Concluida la fase de construcción del pipeline de datos, la segunda versión se centró en el núcleo del proceso KDD: la minería de datos. El objetivo de esta iteración fue transformar el *dataset* consolidado en un modelo de aprendizaje automático con capacidad predictiva, capaz no solo de estimar precios, sino también de proveer conocimiento accionable sobre los factores que los impulsan.

4.3.2.1. Selección y justificación del algoritmo

La elección del algoritmo de modelado se basó en una evaluación sistemática de la naturaleza del problema y las características de los datos. Dado que el objetivo principal del trabajo fue predecir un precio óptimo a partir de un conjunto de variables históricas y competitivas, la tarea se enmarcó fundamentalmente en un problema de regresión supervisada. Más en detalle, se consideró supervisada porque el modelo se entrenó con datos históricos que ya incluyen la variable objetivo (el precio), y es de regresión porque el valor a predecir es una cantidad numérica continua. En consecuencia, para seleccionar el algoritmo más idóneo, se evaluaron varios candidatos con base en los siguientes criterios clave:

- Capacidad para modelar relaciones no lineales, considerada fundamental en mercados con dinámicas de precios complejas.
- Robustez frente a *outliers* resultó importante dada la naturaleza a menudo ruidosa de los datos de *scraping*.
- Interpretabilidad del modelo, como característica crucial para que los resultados sean confiables y adoptados por los usuarios de negocio.
- Eficiencia computacional como clave para asegurar la escalabilidad futura del sistema.

A partir de estos criterios, se elaboró la siguiente matriz comparativa (Tabla 6) para guiar la selección final.

Tabla 6.
Algoritmos de aprendizaje automático

Algoritmo	Aplicación específica	Ventajas Clave	Limitaciones
Regresión Lineal	Predicción de precios base	Alta interpretabilidad, rápido	Asume linealidad, bajo rendimiento en relaciones complejas
Árbol de Decisión	Segmentación y regresión basada en reglas	Muy interpretable, no requiere escalado	Propenso al sobreajuste, inestable
Random Forest	Predicción de precios precisa en relaciones no lineales	Alta precisión, robustez contra sobreajuste	Menos interpretable que un solo árbol
XGBoost	Predicción de precios de alto rendimiento	Rendimiento de vanguardia, regularización	Modelo de "caja negra", complejo de sintonizar

Tras el análisis, se seleccionó el algoritmo Random Forest para el modelo predictivo de precios, esto debido a que ofrecía la combinación más idónea de características para este problema. Más aún, la capacidad para modelar relaciones no lineales complejas, junto con su robusta arquitectura de ensamblado, evitaría de forma inherente el sobreajuste (*overfitting*), convirtiéndolo así en la opción idónea para un mercado volátil. Además, al operar de forma nativa sobre datos tabulares, simplificaría significativamente el pre-procesamiento de las características.

De igual manera, cabe destacar que el desarrollo y configuración del modelo *Random Forest Regressor* se fundamentó a su vez en metodologías y ejemplos similares presentados en la plataforma *Kaggle*, específicamente en el trabajo de Harsh, S. (2023), los cuales sirvieron como referencia práctica para el diseño y ajuste del modelo predictivo utilizado en esta investigación.

4.3.2.2. Entrenamiento y evaluación cuantitativa del modelo

Una vez seleccionado el algoritmo, se entrenó el modelo Random Forest Regressor con un conjunto de datos pertenecientes a la categoría de víveres que abarcaba seis meses de información (marzo-agosto) del 2025, totalizando 121.887 registros. Así, para validar su capacidad de generalización, el rendimiento del modelo se midió exclusivamente sobre un conjunto de prueba ciego (*hold-out set*), compuesto por los datos del último mes de recolección, los cuales no fueron utilizados durante el entrenamiento. Los resultados cuantitativos de esta evaluación se presentan en la Tabla 7.

Tabla 7.
Métricas de rendimiento del modelo predictivo sobre el conjunto de prueba

Métrica de evaluación	Valor obtenido	Descripción de la métrica
Error Absoluto Medio (MAE)	0.152	El error promedio de las predicciones, en la misma unidad que el precio.
Coeficiente de Determinación (R^2)	0.980	Proporción de la varianza en el precio que es predecible a partir de las variables independientes. Cabe destacar que, un valor de 0.98 indica un ajuste muy bueno.

Los resultados cuantitativos iniciales muestran un alto poder predictivo con un R^2 de 0.98, indicando que el modelo explica el 98% de la variabilidad en los precios, y un MAE de 0.15, que representa un error aceptable para decisiones estratégicas. Sin embargo, estudios relevantes destacan que estas métricas por sí solas son insuficientes para evaluar la robustez real del modelo, ya que un R^2 elevado podría ocultar problemas como el sobreajuste y no reflejar sesgos en los errores (IONOS, 2024). Por ello, se realizaron análisis complementarios como el estudio de residuos, técnicas de inteligencia artificial explicable (SHAP), y gráficos de dependencia parcial, que permitieron validar la estabilidad, interpretar las variables influyentes y detectar anomalías que impactarían la toma de decisiones.

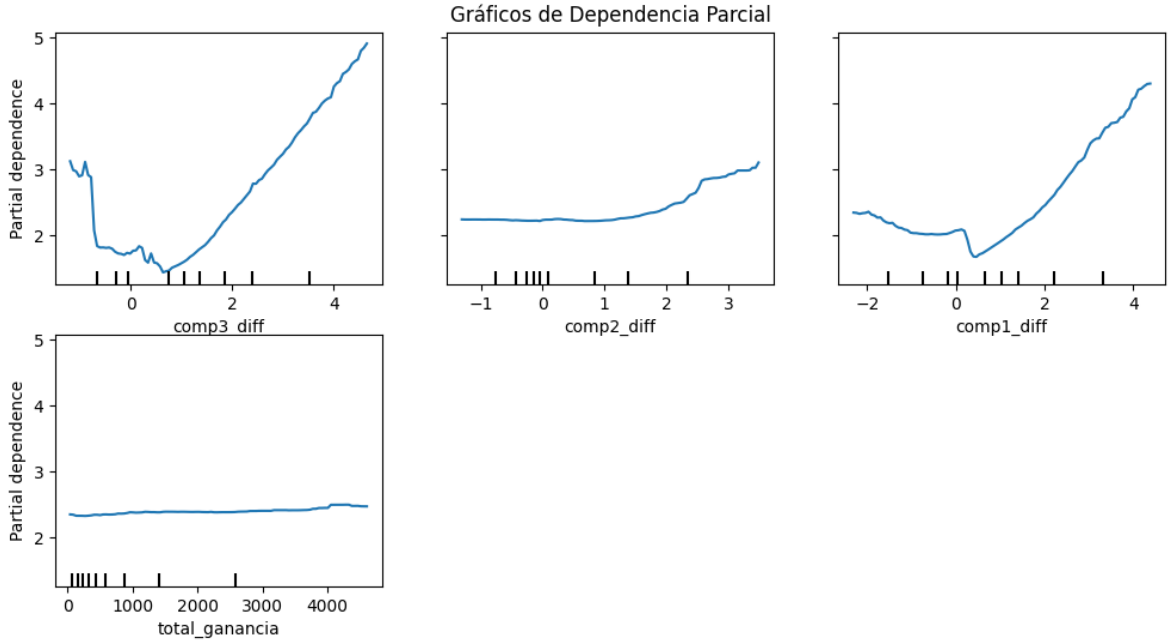
4.3.2.3. Interpretación del modelo mediante Inteligencia Artificial Explicable

Más allá de la precisión predictiva, fue imperativo comprender la lógica interna del modelo para generar confianza y extraer *insights* de negocio. Para este fin, se emplearon técnicas de Inteligencia Artificial Explicable (XAI), específicamente Gráficos de Dependencia Parcial (PDP) y el método SHAP (SHapley Additive exPlanations) tomando como base el marco propuesto por López Martín, P. (2023).

a) Análisis de influencia global de variables (PDP)

Para obtener una visión macro de las relaciones aprendidas por el modelo, se generaron gráficos de dependencia parcial. Estos gráficos (Figura 5) aislaron el efecto marginal de cada variable en la predicción del precio.

Figura 5.
Gráficos de dependencia parcial



El análisis de estos gráficos reveló una relación convexa y no lineal entre las diferencias de precios comp3_diff y comp1_diff y la predicción. Notablemente, cuando la empresa ofrecía precios más baratos, el modelo sugería un precio más alto, identificando una oportunidad de optimización de margen, que puede ser abordada de múltiples formas. Dicha oportunidad implicaría, desde un ajuste gradual del precio hasta una reevaluación completa

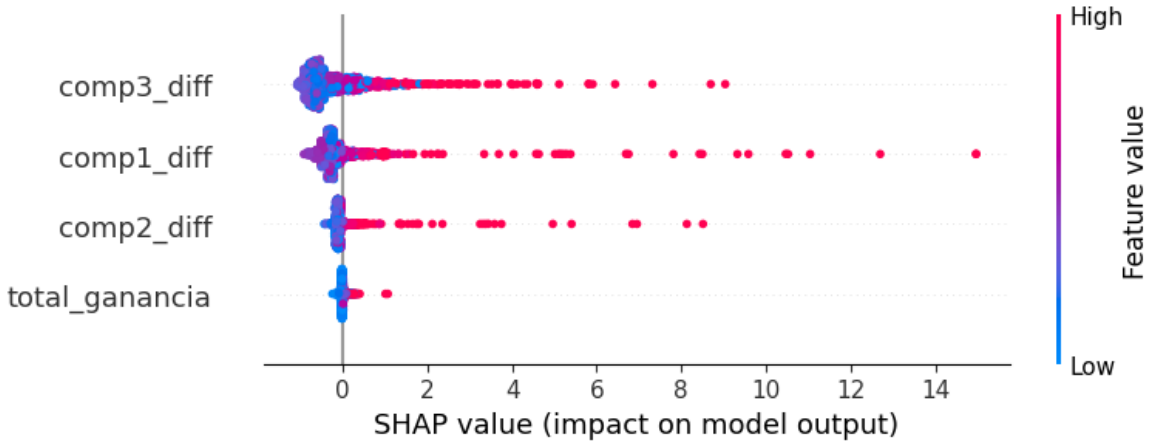
del posicionamiento del producto, cuestionando si se habría subvalorado un artículo o si una estrategia de precios bajos, seguiría siendo pertinente. A la inversa, la planitud en las curvas de comp2_diff y total_ganancia demostró que variaciones en sus valores apenas modificarían el resultado predicho por el modelo, lo que confirmaría su limitada capacidad predictiva.

b) Cuantificación de la importancia y explicación de predicciones (SHAP)

Para profundizar en la visión global del PDP y jerarquizar la influencia de las variables, se utilizó el método SHAP. El gráfico de resumen (Figura 6) confirma que comp3_diff y comp1_diff son, con diferencia, las variables más influyentes. Además, revela una correlación positiva directa: valores altos de diferencia de precios (puntos rojos) empujan la predicción al alza (valor SHAP > 0).

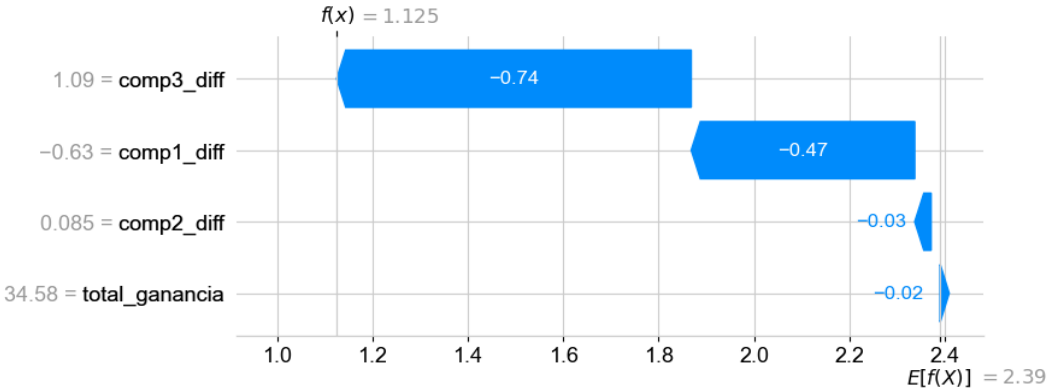
Figura 6.

Gráfico de resumen SHAP del impacto de las variables



El análisis no se detuvo en las tendencias generales; se descendió a nivel de predicción individual. El gráfico de cascada de SHAP (Figura 7) para una observación específica descompone cómo cada variable contribuye a pasar de la predicción base del modelo (E[f(x)] = 2.39) a la predicción final (1.12). Este análisis local corroboró que las decisiones del modelo se basan principalmente en la diferencia con los competidores 1 y 3, demostrando la capacidad del sistema para explicar sus recomendaciones a un nivel granular.

Figura 7.
Desglose de la contribución de las variables para una predicción de ejemplo



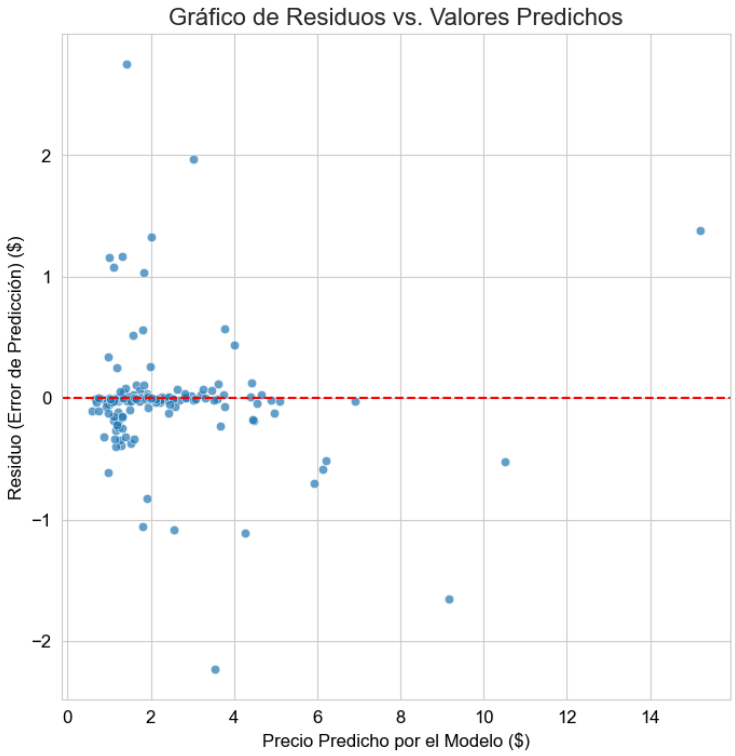
Cabe señalar, que la incorporación del método SHAP se fundamentó y enriqueció gracias al estudio de metodologías similares recogidas en la plataforma *Kaggle*, particularmente el trabajo de Harsh, S. (2023), que sirvió de referencia para la aplicación y adaptación de esta técnica explicativa en el modelo predictivo.

4.3.2.4. Análisis de errores y diagnóstico del modelo

Para validar la eficacia del modelo y comprender sus limitaciones, se realizó un análisis de sus errores de predicción (residuos). Cabe destacar que, este diagnóstico fue esencial, puesto que permitió identificar sesgos sistemáticos o áreas de mejora.

El gráfico de residuos contra valores predichos (Figura 8) exhibe una clara homocedasticidad, con errores distribuidos en una banda horizontal de varianza constante. Este resultado fue un indicador robusto de un buen ajuste, ya que la precisión del modelo resultó consistente a lo largo de todo el rango de precios.

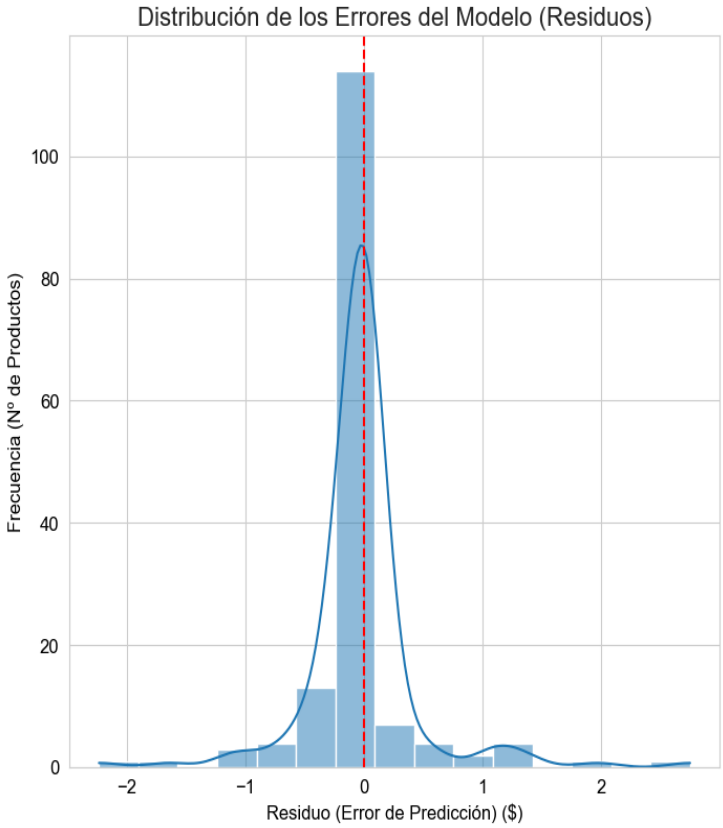
Figura 8.
Comparación de residuos contra valores predichos



Para investigar la distribución de estos errores, se analizaron dos visualizaciones de forma secuencial.

Primero, se generó un histograma (Figura 9) para observar la frecuencia y distribución de los errores. Esta gráfica confirmó que la mayoría de los errores se concentran de manera simétrica alrededor de cero, lo que sugiere una tendencia central en el rendimiento del modelo.

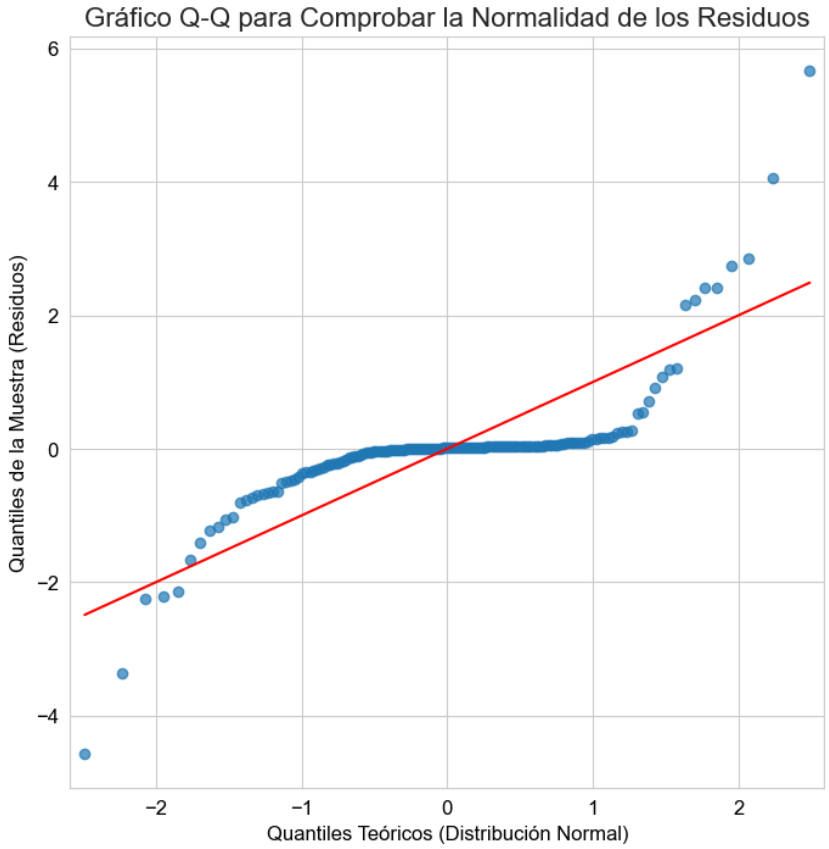
Figura 9.
Distribución de los errores del modelo



A continuación, para profundizar el análisis y comparar la distribución de los errores con una distribución normal, se elaboró un gráfico Q-Q (Figura 10). Esta visualización ratifico la concentración de errores en torno a cero, pero también revelo que las colas de la distribución son más pesadas de lo esperado en una distribución normal. Las desviaciones en los extremos del grafico indican una mayor frecuencia de valores atípicos.

Este patrón es característico de una distribución leptocúrtica, que tiene un pico más alto y colas más largas en comparación con una distribución normal. En el contexto de esta investigación, la naturaleza leptocúrtica de la distribución de errores significó que el modelo era altamente fiable para la mayoría de los productos. No obstante, su rendimiento disminuyó notablemente al enfrentarse a observaciones anómalas o atípicas, que se encuentran en las colas de la distribución.

Figura 10.
Gráfico Q-Q de los residuos contra cuantiles teóricos



En este sentido, la aparente debilidad se reveló como la mayor fortaleza estratégica del sistema. Al identificar los productos que conforman estas "colas pesadas" (Figuras 11 y 12), el modelo actúa como un sistema de diagnóstico automático, señalando las anomalías que representan las mayores oportunidades de optimización de precios, ya sea al alza (subvalorados) o para una revisión estratégica (sobrevalorados).

Figura 11.
Productos sobrevalorados

product_id	name	precio_mes_actual	unit_price_pred	precio_competidor_1	precio_competidor_2	precio_competidor_3	error	diferencia_pred	total_ganancia
2992	IBERIA - PIMIENTA BLANCA MOLIDA 65GR	18.7620	17.1844	0	20.8093	21.5473	1.5776	1.5776	95.7700
1033	FILIPPO VINAGRE BLANCO 500 ML	4.3000	2.9104	0	0	1.5800	1.3896	1.3896	22.5200
1379	HIERBABUENA KG	2.8467	1.8710	0.8996	1.1658	1.6154	0.9756	0.9756	267.6600
2659	HOLSUM - PAN SANDWICH BLANCO 420GR	2.2060	1.5402	0	1.2916	1.3564	0.6658	0.6658	3569.1100
5642	KALDINI - SAL NATURAL ROSADA 390 GR	6.6300	6.0869	0	0	7.4033	0.5431	0.5431	39.7800
4134	CAFE EN GRANO DELLA NONNA 1KG	13.8687	13.3540	0	15.3487	15.3667	0.5147	0.5147	567.7000
1790	FLOR DE ARAUCA - CAFE 500GR	7.0596	6.7249	6.8467	6.6688	0	0.3347	0.3347	3534.0900
4760	DEL MONTE - MAYONESA POUCH 200 GR	2.6021	2.3390	2.0079	0	2.2079	0.2631	0.2631	382.3000
587	RENATA MEZCLA PARA TORTA DE PIÑA 400 GR	1.8325	1.6368	0	0	0.5096	0.1957	0.1957	293.6400
1197	KRAFT MAYONESA 175GR	1.9344	1.7515	1.9544	1.7900	2.0204	0.1829	0.1829	8638.3600

Figura 12.
Productos subvalorados

product_id	name	precio_mes_actual	unit_price_pred	precio_competidor_1	precio_competidor_2	precio_competidor_3	error	diferencia_pred	total_ganancia
3539	LA CAMPIÑA LECHE SEMIDESC VIGIA 900G	12.3669	13.7675	13.2431	0	0	1.4007	-1.4007	1662.4400
3164	KALDINI - ACEITE DE COCO ORGANICO VIRGEN 500ML	8.9360	9.9130	0	8.7760	0	0.9770	-0.9770	2666.9600
2565	HUGGIES - PAÑAL ACTIVE SEC XG 25 UNDS.	12.6723	13.5422	0	15.0323	0	0.8699	-0.8699	221.8000
5385	COCOSETTE MAXI 50GR	0.7957	1.5084	0	6.9600	2.2171	0.7127	-0.7127	345.0600
4445	DE TODITO RESUELTO 400GR	4.0525	4.6180	7.7825	0	5.1875	0.5655	-0.5655	79.5200
291	NESCAFE TRADICIONAL 85GR	4.2496	4.8149	7.1409	0	5.3048	0.5654	-0.5654	1316.4100
1069	ARTESANO ARROZ ARBORIO 1 KG	7.4600	7.9024	11.6823	18.6246	0	0.4424	-0.4424	150.3600
127	UNDERWOOD SALSA PARA PASTAS DIABLITOS 490GR	2.3868	2.8162	0	3.1568	0	0.4294	-0.4294	480.0300

En síntesis, la culminación de la versión 2 no es solo un modelo predictivo con un alto rendimiento cuantitativo, sino una herramienta de inteligencia de negocio interpretable y diagnóstica. El modelo aisló eficazmente las anomalías de precios, permitiendo una intervención estratégica focalizada que de otro modo requeriría un análisis manual extensivo, sentando así las bases para su despliegue en la interfaz de usuario final.

Asimismo, la aplicación colocó a disposición del analista de datos una vista técnica con múltiples herramientas de diagnóstico para su supervisión constante. Entre estas se incluyeron: un Análisis Exploratorio de Datos (EDA), gráficos de rendimiento que comparan los valores reales contra los predichos, la distribución de errores y gráficos Q-Q, así como análisis de interpretabilidad mediante la importancia de variables SHAP, la dependencia parcial (PDP) y el desglose de predicciones individuales. Cabe señalar que, a partir de estos paneles, el analista podía identificar proactivamente cualquier inconveniente o área de mejora, permitiéndole notificar de manera fundamentada la necesidad de realizar ajustes, proponer un cambio de tecnología o reentrenar el algoritmo.

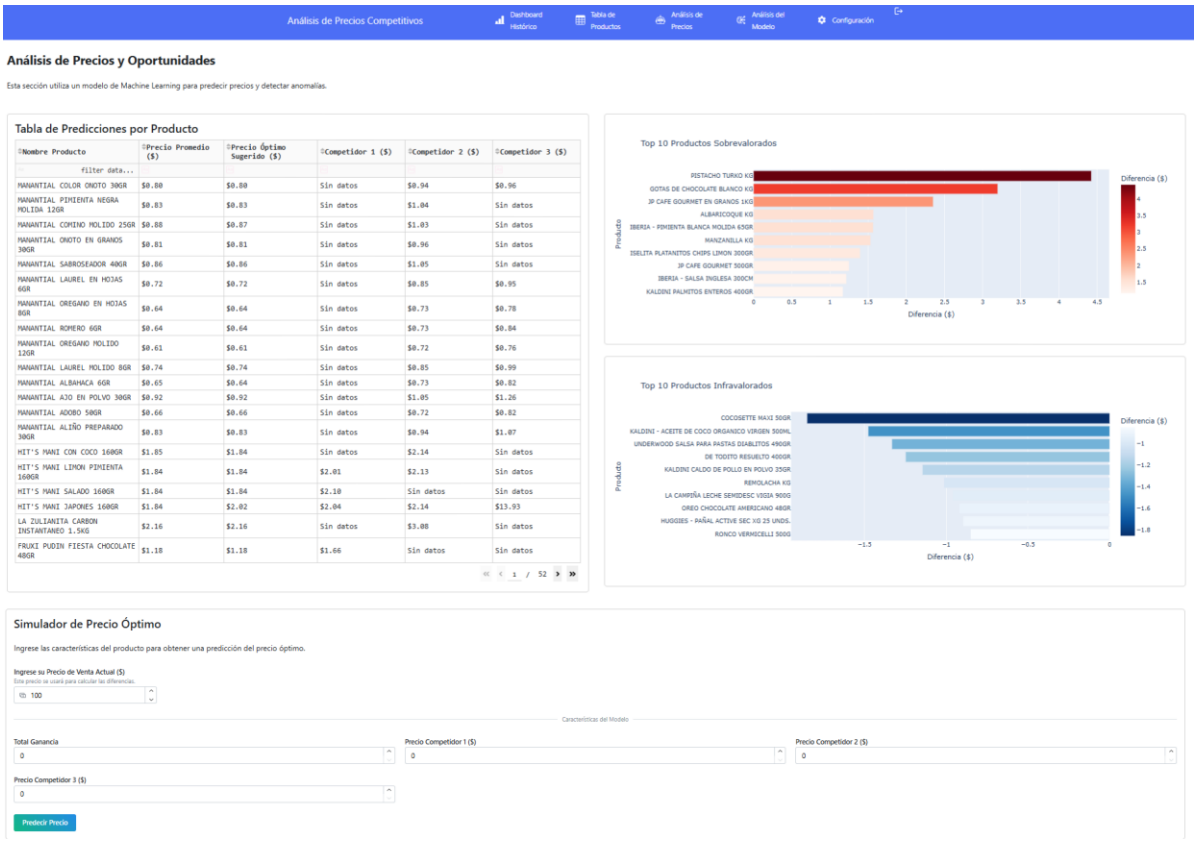
4.3.3. Versión 3: Desarrollo de la capa de presentación

Una vez validados los modelos analíticos y el *pipeline* de datos, el objetivo final de esta versión fue traducir los resultados computacionales en una herramienta de inteligencia de negocio interactiva y de alto valor. Para ello, se diseñó e implementó un *dashboard* web utilizando el *framework Plotly Dash*, siguiendo los principios de diseño centrado en el usuario para asegurar que la información fuera no solo accesible, sino también intuitiva y accionable. La aplicación se estructuró en tres módulos funcionales principales.

4.3.3.1. Modulo 1: Análisis de errores y diagnóstico del modelo

La sección principal de la aplicación, y la más crítica desde el punto de vista estratégico, es el análisis de precios, ya que el objetivo consistió en presentar las recomendaciones del modelo de *Machine Learning* en tiempo real y permitir la exploración de escenarios futuros.

Figura 13.
Pantalla principal de análisis de precios

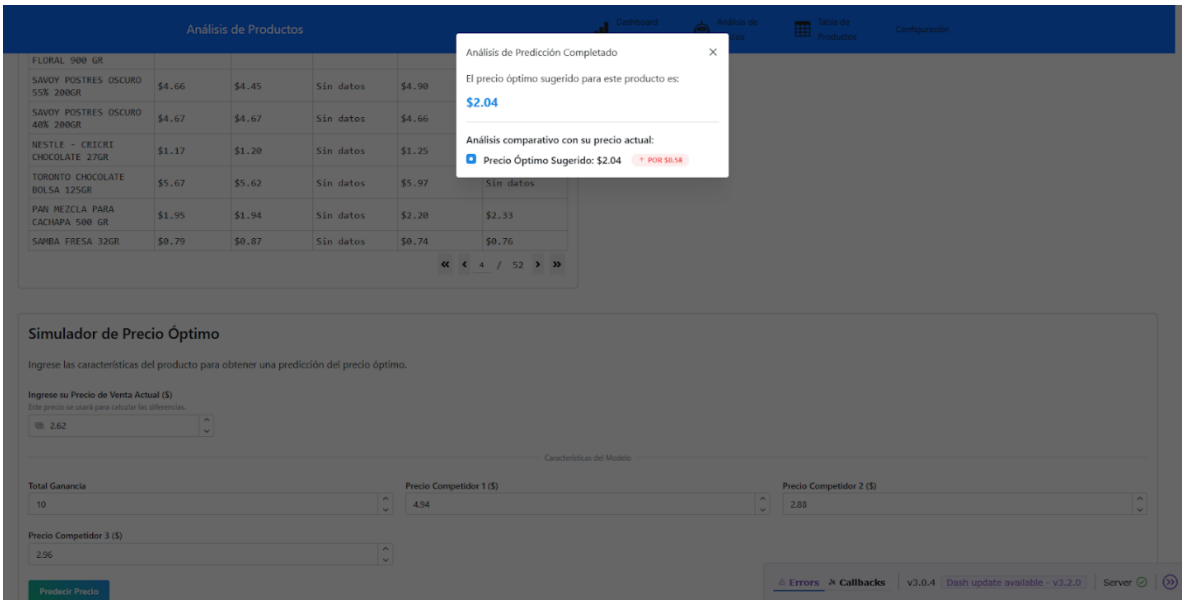


Esta interfaz se conformó por los siguientes elementos de diseño:

- Tabla de recomendaciones: Constituyó el elemento central, ofreciendo una vista comparativa para detallar por producto su precio actual, precio promedio de la competencia y, más importante, el precio recomendado por el modelo. Esta disposición permitió al analista de precios identificar fácilmente las discrepancias y el porqué de las recomendaciones.

- Gráficos de oportunidades: Como complemento a la tabla, se desarrollaron dos gráficos de barras que aislaron y destacaron los productos con la mayor oportunidad de optimización (tanto al alza como a la baja). El objetivo de este componente fue dirigir la atención del analista hacia los casos de mayor impacto potencial, optimizando su tiempo de análisis.
- Simulador de escenarios ("*What-If Analysis*"): Para integrar a la herramienta de capacidad prospectiva, se implementó un simulador interactivo. Este componente estratégico permitió al analista de precios modificar manualmente las variables predictoras clave (ej. Simular una baja de precios de un competidor) y observar la respuesta del modelo en tiempo real. De esta manera, el *dashboard* se transformó de un panel informativo a una herramienta de planificación táctica, permitiendo validar hipótesis y preparar contra-estrategias de forma proactiva.

Figura 14.
Resultado del simulador de precios para un escenario hipotético



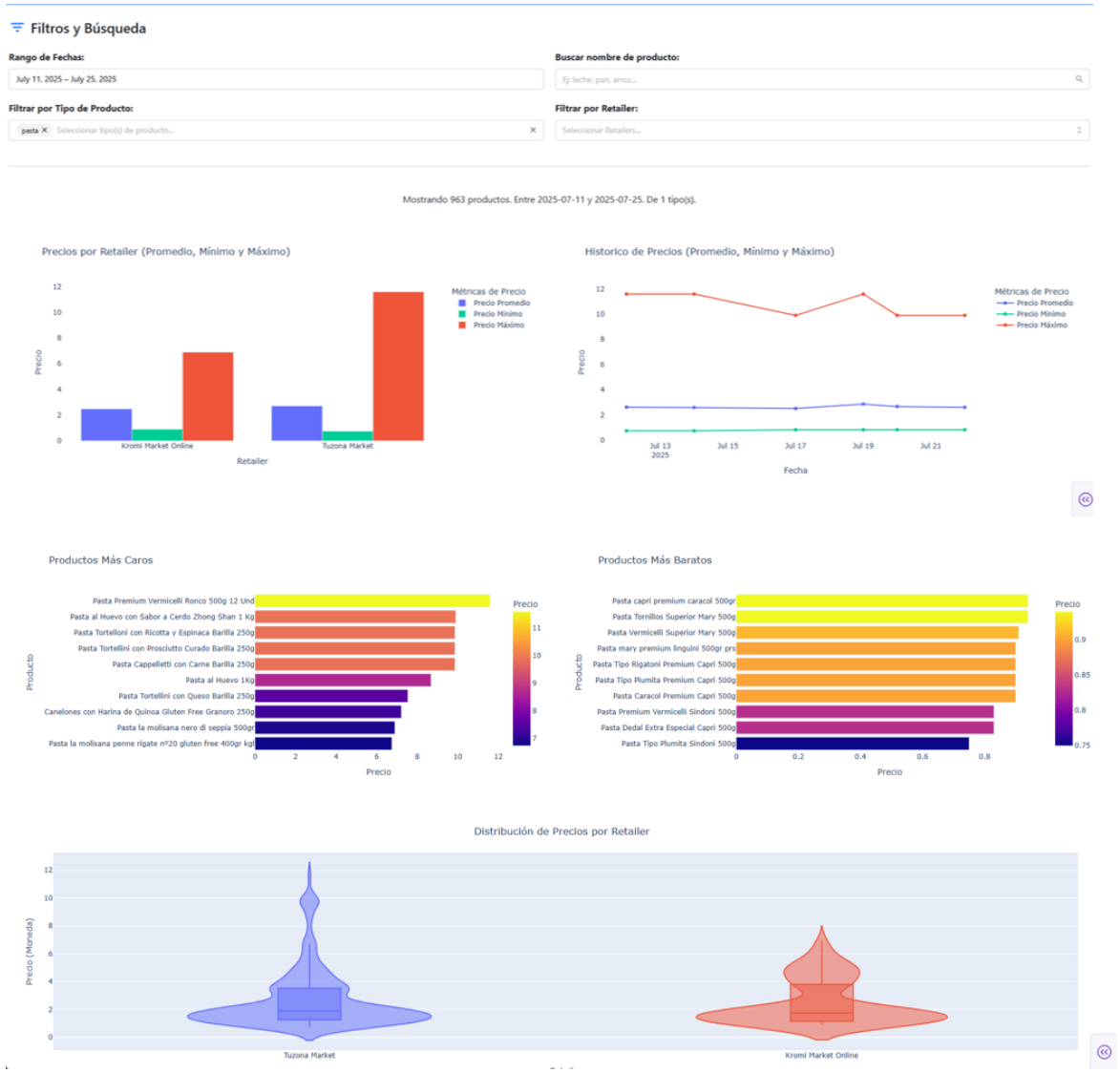
4.3.3.2. Modulo 2: Exploración de datos históricos y análisis competitivo

De manera complementaria al análisis predictivo, se desarrolló un segundo módulo enfocado en la exploración del histórico de datos recolectados. El propósito de esta sección

fue permitir al analista investigar tendencias, entender las estrategias de precios de la competencia a lo largo del tiempo y validar visualmente los patrones que el modelo ha aprendido.

Figura 15.

Dashboard de análisis de precios históricos



Los componentes visuales incluyeron:

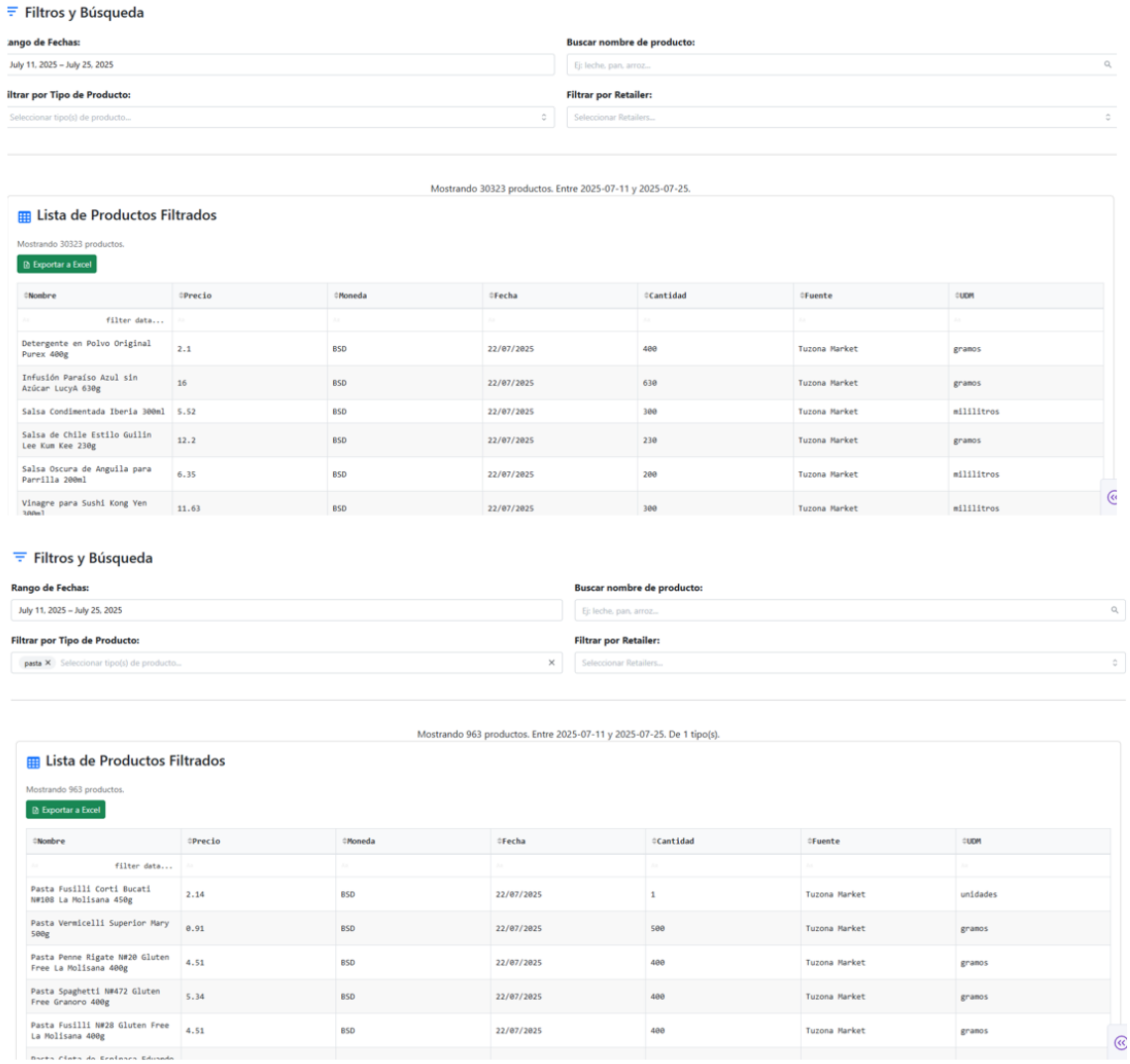
- Gráficos de barras y de violín: Permitieron comparar las distribuciones de precios entre *retailers*, identificando quién es consistentemente más caro o más barato y cuál es la dispersión de su estrategia.

- Gráfico de series temporales: Permitió visualizar la evolución de los precios de un producto a lo largo del tiempo, facilitando la identificación de tendencias, estacionalidad o reacciones a eventos del mercado.
- Filtros interactivos: Todos los gráficos fueron interconectados y responden a filtros dinámicos (por producto, categoría, rango de fechas), permitiendo al analista de precios segmentar la información y realizar análisis profundos sobre nichos de interés.

4.3.3.3. Modulo 3: Explorador y exportación de datos tabulares

Sumado a esto, para facilitar el análisis ad-hoc y la integración con otras herramientas corporativas (como Microsoft Excel), se implementó una sección de “Tabla de Productos” la cual pueden acceder todos los usuarios registrados.

Figura 16.
Vista de la tabla de productos con filtros aplicados



Esta funcionalidad ofreció una vista completa de la base de datos de precios históricos en formato tabular, equipada con capacidades avanzadas de búsqueda y filtrado. Para ello, se incluyó una función que le permite a los analistas exportar la vista de datos seleccionada a un archivo en formato .csv, garantizando que los datos recolectados puedan ser utilizados en otros procesos de negocio sin restricciones.

4.3.3.4. Modulo 4: *Dashboard* de análisis del modelo

Adicionalmente, con el objetivo de facilitar la validación y evaluación en tiempo real del modelo *de Machine Learning*, se implementó un cuarto módulo. Su enfoque es desglosar los hallazgos del modelo mediante el proceso de IA Explicable (descrito en la sección 4.3.2), organizando este análisis en tres vistas interactivas. Asimismo, esta sección solo es accesible por los Analistas de Datos y Administradores.

En primera instancia, una de estas vistas (Figura 17) presenta un panel de control consolidado que resume las tres etapas del análisis. En esta se desarrolló una tabla con las estadísticas descriptivas del conjunto de datos, las métricas de rendimiento clave del modelo (R^2, MAE, MSE) junto a un gráfico de dispersión de valor real vs. predicho, y un resumen visual de la importancia de las características (SHAP) que identifica los factores más influyentes en las predicciones.

Seguidamente, la segunda vista (Figura 18) se enfocó en la exploración detallada de las variables individuales y su efecto en el modelo. Expone los histogramas para analizar la distribución de cada variable numérica y del error de predicción, y presenta un Gráfico de Dependencia Parcial (PDP) que ilustra cómo la predicción es afectada por los cambios en una sola característica, aislando su impacto.

Por su parte, la tercera vista (Figura 19) ofreció herramientas de diagnóstico y transparencia a nivel micro. Para ello, se incluyó una matriz de correlación para evaluar las relaciones lineales entre variables, un gráfico Q-Q para verificar la normalidad estadística de los residuos del modelo, y un desglose detallado de una predicción individual, explicando cómo cada característica contribuye al resultado final para un caso específico.

Figura 17.
Vista general del análisis de modelo (SHAP)

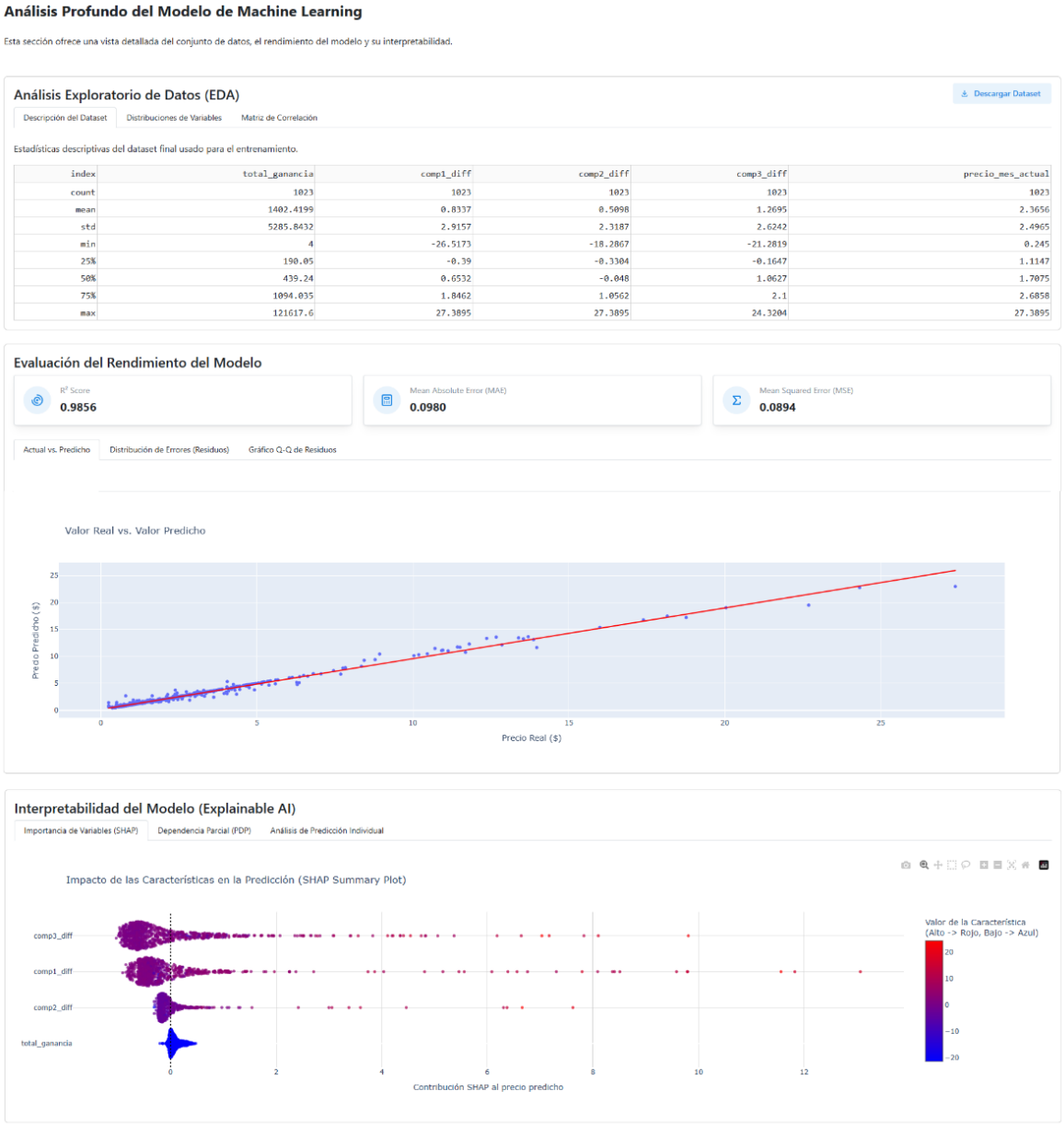


Figura 18.

Análisis de distribuciones y dependencia de variables (PDP)

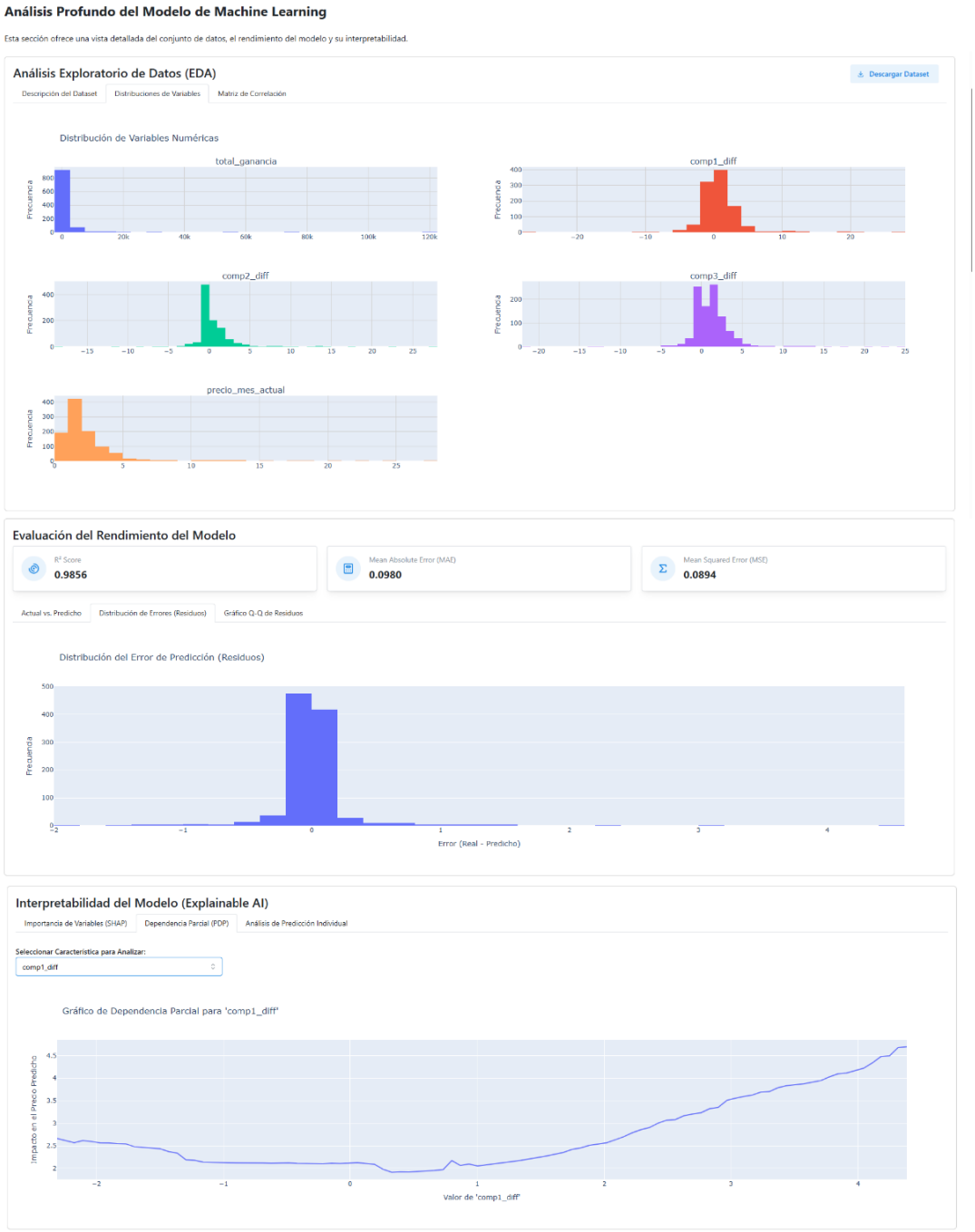
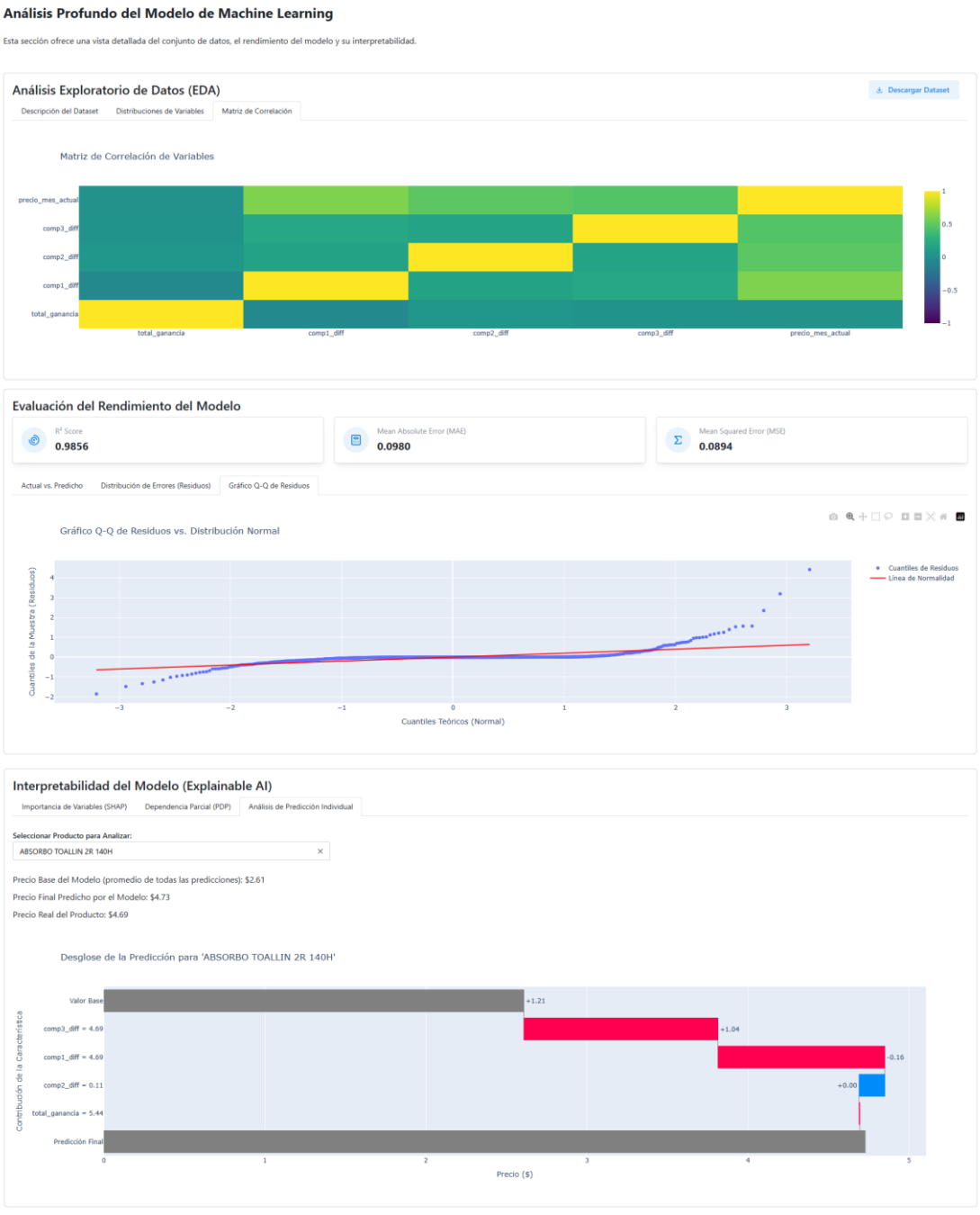


Figura 19.
Análisis de correlaciones y explicabilidad individual



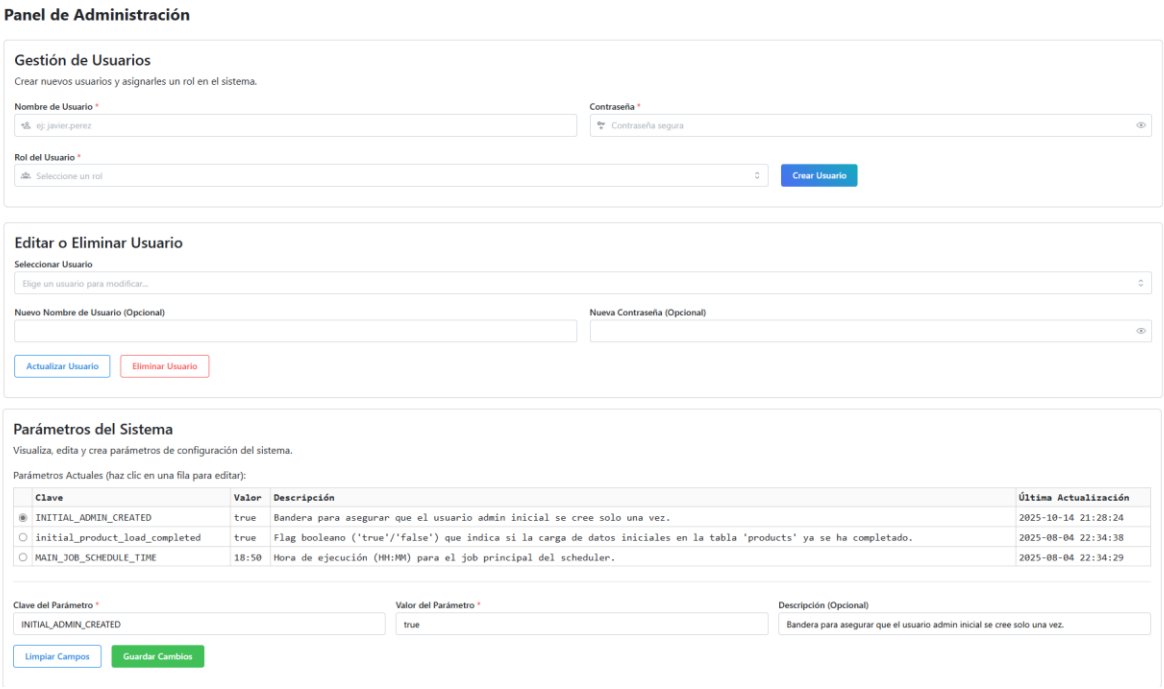
En conjunto, las herramientas visuales distribuidas en estas figuras cumplieron funciones específicas y complementarias para la validación del modelo:

- Gráficos de exploración y distribución: Permitieron una comprensión inicial de los datos, ayudando a analizar la distribución de cada variable (histogramas), visualizar la relación entre ellas (matriz de correlación) y obtener un resumen estadístico completo.
- Visualizaciones de rendimiento del modelo: Proporcionaron una medida clara de la precisión del modelo al permitir comparar las predicciones contra los valores reales (gráfico de dispersión) e identificar la magnitud y el sesgo de los errores de predicción (histograma de residuos).
- Herramientas de IA Explicable (XAI): Presentaron una vista transparente a la "caja negra" del modelo, permitiendo identificar qué características son las más influyentes en las predicciones (gráficos SHAP) y analizar cómo el cambio en una variable específica afecta el resultado final (gráficos de dependencia parcial).

4.3.3.5. Modulo 5: Panel de administración

Finalmente, con el fin de agregar flexibilidad y personalización al sistema se incluyó un panel de administración, al cual solo acceden los administradores del sistema (Figura 20).

Figura 20.
Vista de panel de administración



Este panel permitió ofrecer la capacidad de gestionar los usuarios y sus permisos, así como de modificar parámetros clave del sistema tales como:

- INITIAL_ADMIN_CREATED: Bandera que indica al sistema si debe crear un usuario administrador por defecto al próximo reinicio.
- initial_product_load_completed: Fuerza al sistema a cargar data inicial (Datos de precios de 3 meses) al próximo reinicio.
- MAIN_JOB_SCHEDULE_TIME: Indica la hora en la que se va a ejecutar el proceso de recolección de datos.

4.3.3.6. Caso de Prueba

Para ilustrar el flujo de trabajo de un analista de precios utilizando la herramienta, se presenta el caso del producto "Arroz Mary". El proceso comienza cuando el analista detecta en el panel de "Análisis de Precios" (Figura 21) que el sistema sugiere un precio de $1.79, notablemente más alto que el actual, marcándolo como una gran oportunidad de aumento.

Figura 21.
Vista de identificación de oportunidad (caso de uso)

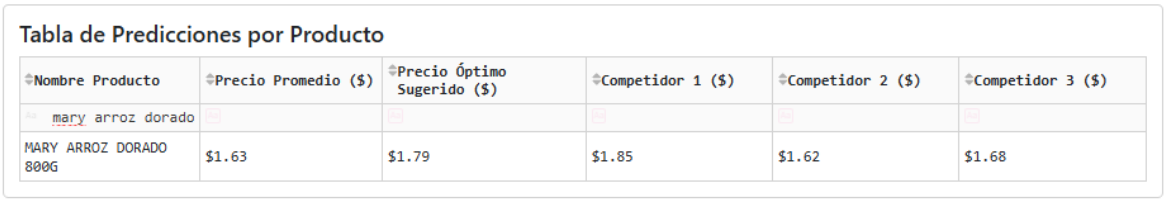
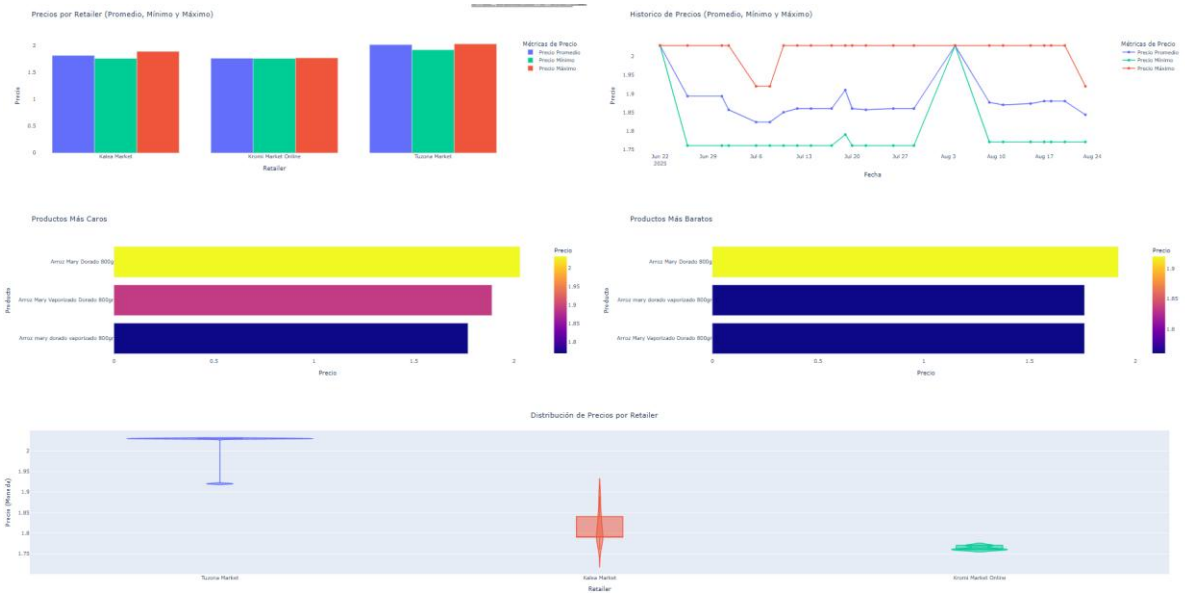


Tabla de Predicciones por Producto

Nombre Producto	Precio Promedio ($)	Precio Óptimo Sugerido ($)	Competidor 1 ($)	Competidor 2 ($)	Competidor 3 ($)
mary arroz dorado					
MARY ARROZ DORADO 800G	$1.63	$1.79	$1.85	$1.62	$1.68

Para verificar esto, el analista consulta el "Dashboard de Datos Históricos" (Figura 22) y confirma que, a lo largo del tiempo, el "Arroz Mary" siempre ha sido más barato que sus competidores. Esto valida la recomendación del modelo, demostrando que el producto ha estado infravalorado de forma constante y que existe una oportunidad real de mejorar el margen de ganancia.

Figura 22.
Vista de investigación histórica (caso de uso)



Con esta evidencia, el analista puede proponer con seguridad un aumento de precio. Sin embargo, la herramienta también abre la puerta a otras estrategias: si la empresa prefiere mantener el precio bajo para atraer clientes, el margen que se sacrifica puede considerarse una inversión en marketing, asimismo si se posiciona el producto en una mejor ubicación en la tienda, podrían reflejarse cambios positivos a nivel de ventas. De esta forma, el analista

puede proponer no solo un precio, sino un conjunto de estrategias de negocio alineadas con los objetivos de la empresa.

En resumen, la herramienta conecta los modelos de datos complejos con las decisiones estratégicas, probando así que no se limita a mostrar información, sino que guía al analista de precios para convertir las predicciones en acciones de negocio concretas. A su vez, la plataforma es de gran valor para el analista de datos, ya que le ofrece herramientas de IA Explicable (XAI) para monitorear y entender el modelo. Esto asegura que el sistema no sea una "caja negra", garantizando su fiabilidad y facilitando su mantenimiento por parte del equipo técnico.

4.3.4. Versión 4: Validación integral y pruebas de rendimiento del sistema

Con todos los módulos funcionales implementados e integrados la iteración final del desarrollo corresponde a la fase de Producción del ciclo de vida de XP. El objetivo de esta versión no fue añadir nuevas funcionalidades, sino someter al sistema completo a una serie de pruebas cuantitativas y cualitativas para verificar su eficiencia, precisión y, en última instancia, su valor estratégico como herramienta de inteligencia de negocio.

4.3.4.1. Pruebas de rendimiento del módulo de extracción de datos (ETL)

Para evaluar la eficiencia y robustez del pipeline de recolección de datos, se midió el rendimiento del módulo de *scraping* bajo condiciones controladas. La prueba consistió en ejecutar un ciclo completo de extracción, transformación y carga (ETL) sobre las tres fuentes de datos de la competencia.

Condiciones de prueba:

- Conexión a internet: 4.1 Mbps (representativa de condiciones reales no óptimas).

Resultados de rendimiento:

- Tiempo total de ejecución: 44 minutos y 28 segundos.
- Volumen de datos recolectados: 5,327 registros de productos.
- Tasa de procesamiento promedio: Aproximadamente 120 productos por minuto.

Los resultados demostraron la eficiencia del módulo de *scraping*, al procesar un volumen considerable de datos en un tiempo razonable. Adicionalmente, durante las pruebas se constató la resistencia del sistema frente a cortes de red de 2-3 segundos durante la extracción de los tres *retailers*, garantizando la finalización del ciclo de extracción de manera fiable. En conclusión, el *backend* de datos cumplió con los requisitos de rendimiento necesarios para mantener la base de datos actualizada de forma frecuente.

No obstante, la fidelidad del sistema se validó no solo por su eficiencia cuantitativa, sino por su capacidad de adaptación a la naturaleza heterogénea de las plataformas web de los *retailers*. Las pruebas revelaron que sitios como Kromi Market y Kalea dependen intensivamente de JavaScript para renderizar su contenido dinámicamente, y presentaron indicios de sistemas de seguridad básicos, contra el tráfico automatizado. En estos casos, se confirmó que el enfoque basado en la automatización de navegador con *Selenium* fue efectivo, demostrando ser indispensable para simular la interacción humana necesaria, navegar por la paginación dinámica y extraer los datos de manera consistente, aunque ello implicara un proceso intrínsecamente más lento.

Asimismo, reconociendo la fragilidad inherente de los scrapers basados en navegador frente a los posibles cambios en la estructura visual (HTML/CSS), se implementó una estrategia más avanzada cuando fue posible. Específicamente, para TuZonaMarket, el análisis durante las pruebas permitió identificar y consumir una API interna que la propia plataforma web utilizaba para cargar sus productos. Esta estrategia de ingeniería inversa, basada en la exploración de las solicitudes de red y fuentes de datos, se fundamentó en metodologías descritas en la guía *'How to reverse engineer website APIs'* de *Apify Blog*" cuya autoría es de Lhoťanová, K., (2025). Esta aproximación probó ser eficiente en el uso de recursos y considerablemente más robusta, dado que las *APIs* tienden a mantener su estructura con mayor estabilidad que las interfaces de usuario.

Finalmente, se validó que el proceso de extracción de datos se llevara a cabo de manera ética y respetuosa. Previo a cualquier extracción, se verificó diligentemente el archivo robots.txt de cada sitio para asegurar el cumplimiento de sus políticas de acceso automatizado. Además, el módulo fue diseñado para emular un comportamiento de navegación humano, incluyendo la implementación de pausas deliberadas (*delays*) entre las

solicitudes, una forma de *rate limiting* auto-impuesto que evita la sobrecarga de los servidores de los *retailers* y garantiza la sostenibilidad de la recolección de datos a largo plazo. Asimismo, el proceso se limitó rigurosamente a la extracción de información de precios disponible públicamente, sin intentar acceder a datos privados o personales.

4.3.4.2. Evaluación cuantitativa del motor analítico (*Machine Learning*)

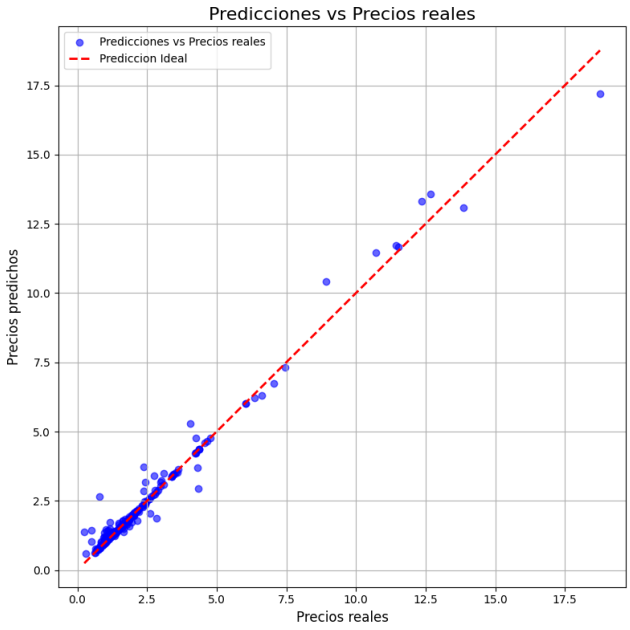
El siguiente paso fue cuantificar la precisión del modelo de Random Forest, que constituye el núcleo analítico de la aplicación. Para ello, se utilizaron métricas estándar de evaluación de regresión sobre el conjunto de prueba ciego (*hold-out set*), que el modelo no había visto durante su entrenamiento.

Métricas de precisión:

- Error Medio Absoluto (MAE): Se obtuvo un MAE de 0.15. Este valor indica que, en promedio, las predicciones del modelo se desvían en solo 0.15 unidades monetarias del precio real. Dada la escala de precios de la mayoría de los productos, este es un nivel de error muy bajo.
- Coeficiente de Determinación (R^2): El modelo alcanzó un R^2 de 0.98. Este resultado es excepcionalmente fuerte e indica que el 98% de la variabilidad en los precios de los productos es explicada por las variables predictoras del modelo.

Visualización del Ajuste: Para complementar estas métricas, la Figura 23 contrasta los precios reales con los precios predichos por el modelo.

Figura 23.
Comparación de precios reales vs. predichos



La fuerte correlación lineal visible en la gráfica, junto con las métricas de MAE y R^2, confirman que el modelo posee un altísimo grado de precisión y un ajuste robusto a los datos. Estos hallazgos cuantitativos proporcionaron una base sólida de confianza en las recomendaciones generadas por el sistema.

4.3.4.3. Validación estratégica y cualitativa en un escenario real

El análisis determinante para el sistema no residió solo en su precisión numérica, sino en su capacidad para generar inteligencia accionable y estratégicamente coherente. Para validar este aspecto, se realizó un análisis cualitativo comparando las predicciones del modelo (generadas antes de procesar los datos de agosto) con la realidad del mercado que se materializó en agosto (Figura 24), utilizando el *dashboard* como herramienta de análisis.

Este ejercicio permitió verificar tres capacidades clave del sistema:

1. Coherencia predictiva (Caso Lentejas Pantera): Basándose en datos hasta julio, el modelo recomendó un precio de $1.85 (Figura 25). Mientras que, en agosto, los competidores fijaron precios de $1.96 y $2.05. Lo que demuestra que la

recomendación del modelo tiene como objetivo posicionar el producto con una ventaja competitiva de precio para capturar demanda

2. Adaptabilidad estratégica (Caso Mayonesa Kraft): La recomendación del modelo de $1.73 (Figura 26) fue una respuesta a la dispersión de precios observada históricamente. Esta predicción se validó como relevante, cuando en agosto un competidor adoptó una estrategia agresiva con un precio de $1.65, demostrando que el modelo no memoriza precios, sino que generaliza relaciones competitivas.
3. Identificación de oportunidades latentes (Caso Arroz Mary): El sistema predijo un precio óptimo de $1.79 (Figura 27), identificando un patrón histórico de subvaloración. Esta predicción se validó en agosto, cuando los precios de todos los competidores se situaron muy por encima de la oferta actual del *retailer*. En este caso, el modelo proyectó con éxito una oportunidad de optimización de margen que se materializó en el mercado real.

En síntesis, la culminación de la versión 4 mostró que la aplicación desarrollada no es solo un conjunto de módulos funcionales, sino un sistema integrado, eficiente y preciso. Las pruebas de rendimiento validan la estabilidad del *backend*, las métricas cuantitativas confirman la precisión del motor analítico, y la validación estratégica cualitativa prueba su capacidad para generar *insights* de negocio valiosos y fiables.

Figura 24.
Valores reales para agosto

name text	price numeric	website_id integer	scrape_timestamp timestamp with time zone
Mayonesa Kraft Premium 175gr	1.85	3	2025-08-09 13:14:33.257219-04
Mayonesa kraft 175gr	1.65	2	2025-08-09 13:14:33.257219-04
Mayonesa Kraft 175g	1.93	1	2025-08-04 21:23:46.215708-04
Lentejas pantera 400gr	2.05	2	2025-08-09 13:14:33.257219-04
Lenteja Pantera 400g	1.96	1	2025-08-04 21:23:46.215708-04
Arroz Mary Vaporizado Dorado 800...	1.83	3	2025-08-09 13:14:33.257219-04
Arroz mary dorado vaporizado 800gr	1.77	2	2025-08-09 13:14:33.257219-04
Arroz Mary Dorado 800g	2.03	1	2025-08-04 21:23:46.215708-04

Figura 25.
Predicción del precio óptimo para el producto 1

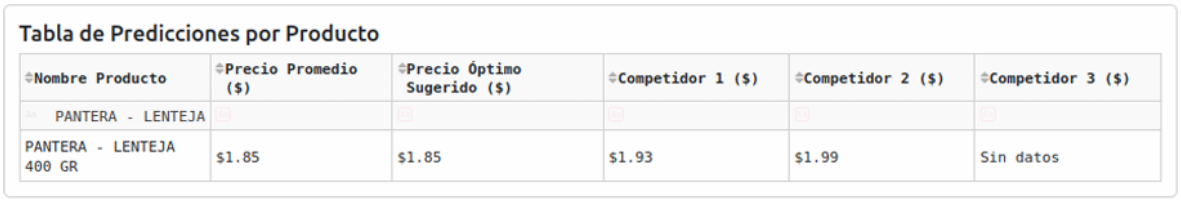


Tabla de Predicciones por Producto

Nombre Producto	Precio Promedio ($)	Precio Óptimo Sugerido ($)	Competidor 1 ($)	Competidor 2 ($)	Competidor 3 ($)
PANTERA - LENTEJA					
PANTERA - LENTEJA 400 GR	$1.85	$1.85	$1.93	$1.99	Sin datos

Figura 26.
Predicción del precio óptimo para el producto 2

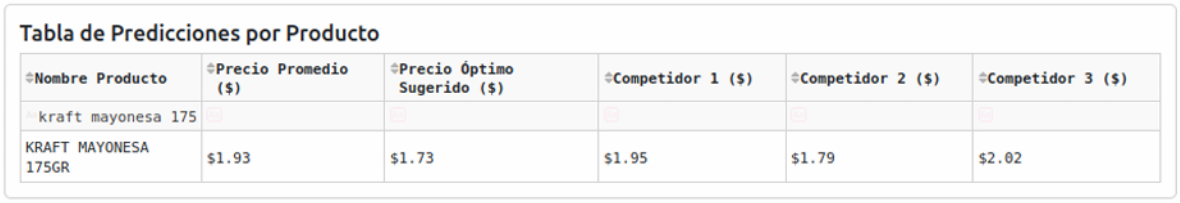


Tabla de Predicciones por Producto

Nombre Producto	Precio Promedio ($)	Precio Óptimo Sugerido ($)	Competidor 1 ($)	Competidor 2 ($)	Competidor 3 ($)
kraft mayonesa 175					
KRAFT MAYONESA 175GR	$1.93	$1.73	$1.95	$1.79	$2.02

Figura 27.
Predicción del precio óptimo para el producto 3

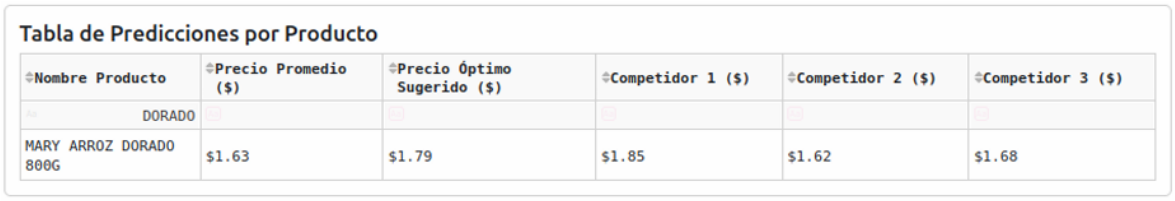


Tabla de Predicciones por Producto

Nombre Producto	Precio Promedio ($)	Precio Óptimo Sugerido ($)	Competidor 1 ($)	Competidor 2 ($)	Competidor 3 ($)
DORADO					
MARY ARROZ DORADO 800G	$1.63	$1.79	$1.85	$1.62	$1.68

4.3.4.4. Pruebas de aceptación

En conformidad con el marco de trabajo de *eXtreme Programming* (XP), se implementó un plan de pruebas de aceptación para verificar que el software cumpliera con las especificaciones requeridas. Los hallazgos de este proceso se detallan en las tablas 8 a la 21.

Tabla 8.
Prueba de aceptación PA001

PA001: Visualización de precios recomendados y oportunidades de optimización	
Funcionalidad	*Dashboard* de análisis predictivo
Fecha de ejecución	22/10/2025
Versión del sistema	3
Entrada esperada	Salida esperada
Acceso al *dashboard* de "Análisis de Precios"	El sistema debe mostrar la tabla comparativa de precios (propio, competencia, recomendado)
Carga de datos del último análisis	El sistema debe presentar los gráficos de barras con los productos de mayor oportunidad de aumento y disminución
Selección de un producto en la tabla	Los detalles del producto seleccionado deben ser consistentes con los datos del modelo

a) Objetivo de la prueba: Validar que el *dashboard* principal muestra correctamente las recomendaciones del modelo, la comparación de precios y las oportunidades de optimización de manera clara y accionable para el Analista de Precios.

b) Pasos para ejecutar la prueba:

1. Iniciar sesión como "Analista de Precios".
2. Navegar al módulo "Análisis de Precios".
3. Verificar que la tabla principal se carga con los datos correctos del último ciclo de predicción.
4. Confirmar que los gráficos de "Mayores Oportunidades" reflejan los productos con mayor diferencia entre el precio actual y el recomendado.

Tabla 9.
Resultados de la prueba PA001

Resultado esperado	Resultado observado
La tabla comparativa se muestra con los precios actual, de la competencia y recomendado.	La tabla se cargó correctamente, mostrando todas las columnas esperadas.
Los gráficos de oportunidades destacan los productos correctos.	Los gráficos de barras se generaron y coincidieron con los datos de la tabla.
El sistema responde de manera fluida al interactuar con los datos.	La interfaz respondió sin demoras a las interacciones del usuario.

c) Estado de la prueba: Aprobada.

d) Comentarios: La funcionalidad principal para el Analista de Precios operó según los criterios de aceptación, no se detectaron errores.

Tabla 10.
Prueba de aceptación PA002

PA002: Simulación de escenarios de mercado ("*What-if*")	
Funcionalidad	Simulador de Precios
Fecha de ejecución	22/10/2025
Versión del sistema	3
Entrada esperada	Salida esperada
Ejecución de la simulación con los parámetros ingresados	El sistema calcula y muestra un precio recomendado en el panel de resultados del simulador
Ingreso de valores no numéricos en los campos de parámetros.	El sistema debe mostrar una validación o ignorar la entrada, impidiendo la ejecución de la simulación.

a) Objetivo de la prueba: Validar que la herramienta de simulación permite al Analista de Precios explorar escenarios hipotéticos de mercado y observar el impacto inmediato en las recomendaciones del modelo.

b) Pasos para ejecutar la prueba:

1. Iniciar sesión como "Analista de Precios".
2. Navegar al módulo "Análisis de Precios".
3. Localizar la sección del simulador.
4. Introducir valores en los campos de entrada correspondientes.
5. Hacer clic en el botón "Predecir precio".
6. Verificar que el nuevo "Precio Recomendado" se muestre claramente en el área de resultados del simulador.

Tabla 11.
Resultados de la prueba PA002

Resultado esperado	Resultado observado
El nuevo precio recomendado se muestra en la sección del simulador tras el cálculo.	La predicción se actualizó correctamente en el panel de resultados del simulador.
El sistema gestiona correctamente las entradas inválidas.	Las entradas no numéricas fueron rechazadas y se mostró un mensaje de error al usuario.

c) Estado de la prueba: Aprobada.

d) Comentarios: El simulador funciono como una herramienta de análisis robusta, cumpliendo con todos los criterios de aceptación.

Tabla 12.
Prueba de aceptación PA003

PA003: Exploración y exportación de datos históricos	
Funcionalidad	Tabla de productos y exportación a Excel
Fecha de ejecución	22/10/2025
Versión del sistema	3
Entrada esperada	Salida esperada
Aplicación de filtros por fecha y producto en la tabla.	La vista de la tabla se actualiza mostrando únicamente los registros que cumplen con los criterios de filtrado.
Clic en botón “Exportar a Excel”	El sistema debe generar y descargar un archivo .xlxs
Apertura del archivo .xlxs descargado	El contenido del archivo debe corresponder exactamente a los datos filtrados en la interfaz web

a) Objetivo de la prueba: Validar que los usuarios pueden explorar el conjunto de datos históricos, aplicar filtros para análisis específicos y exportar los resultados para su uso en herramientas externas.

b) Pasos para ejecutar la prueba:

1. Iniciar sesión.
2. Navegar al módulo "Tabla de productos".
3. Utilizar los campos de búsqueda y los selectores para filtrar los datos por un *retailer* y un rango de fechas específico.
4. Presionar el botón de exportación.
5. Abrir el archivo descargado y comparar su contenido con la vista en pantalla.

Tabla 13.
Resultados de la prueba PA003

Resultado esperado	Resultado observado
Los filtros actualizan la tabla de datos en tiempo real.	Los filtros funcionaron correctamente, actualizando la vista de datos de forma instantánea.
Se descarga un archivo .xlxs al hacer clic en el botón de exportación.	La descarga del archivo .csv se inició y completó exitosamente.
El archivo exportado contiene los datos filtrados.	El contenido del archivo CSV coincidió con los registros mostrados en la interfaz.

c) Estado de la prueba: Aprobada.

d) Comentarios: La funcionalidad de exploración y exportación de datos cumplió con todos los requisitos, garantizando la flexibilidad para análisis ad-hoc.

Tabla 14.
Prueba de aceptación PA004

PA004: Verificación del proceso de ETL automatizado	
Funcionalidad	Ejecución programada del pipeline de datos (*Backend*)
Fecha de ejecución	23/10/2025
Versión del sistema	3
Entrada esperada	Salida esperada
El sistema alcanza la hora programada para la ejecución del *scraper*.	El proceso de *web scraping* se inicia automáticamente sin intervención manual.
Carga de datos del último análisis	Los nuevos datos de precios extraidos se almacenan correctamente en la tabla *preprocessed_products* de la base de datos.
Selección de un producto en la tabla	Se observan nuevos registros con el *scrape_timestamp* correspondiente a la fecha y hora de la ejecución.

a) Objetivo de la prueba: Validar que el pipeline de extracción, transformación y carga (ETL) se ejecuta de forma automática y periódica, asegurando que los análisis se basen siempre en información de mercado reciente.

b) Pasos para ejecutar la prueba:

1. Iniciar sesión como administrador.
2. Navegar al panel de administración
3. Verificar el parámetro *MAIN_JOB_SCHEDULE_TIME* en la configuración.
4. Esperar a que pase la hora de ejecución programada.
5. Acceder a la base de datos de PostgreSQL
6. Ejecutar una consulta sobre la tabla *preprocessed_products* para verificar la existencia de nuevos registros con la fecha actual.

Tabla 15.
Resultados de la prueba PA004

Resultado esperado	Resultado observado
El proceso de ETL se ejecuta automáticamente en la hora programada.	Los logs del sistema y la ejecución del navegador controlado por el *scraper* confirmaron el inicio del proceso en la hora configurada.
Los nuevos datos se insertan en la base de datos.	La inserción de los nuevos registros se realizó con éxito.
Los *timestamps* de los nuevos datos son correctos.	Los nuevos registros tenían el *scrape_timestamp* esperado, validando la actualización de los datos.

c) Estado de la prueba: Aprobada

d) Comentarios: El backend del sistema demostró ser fiable y autónomo, cumpliendo con el requisito crítico de mantener los datos actualizados para garantizar la relevancia de las predicciones.

Tabla 16.
Prueba de aceptación PA005

PA005: Análisis de precios históricos y tendencias de la competencia	
Funcionalidad	*Dashboard* de datos históricos
Fecha de ejecución	23/10/2025
Versión del sistema	3
Entrada esperada	Salida esperada
Ingreso del nombre de un producto especifico en la barra de búsqueda.	El gráfico de series temporales se actualiza para mostrar la evolución del precio de ese producto a través del tiempo, comparando al *retailer* con la competencia.
Selección de una categoría de producto.	Los gráficos de barras y de violín se actualizan para mostrar la distribución de precios de esa categoría entre los diferentes *retailers*.
Aplicación de un filtro por rango de fechas.	Todas las visualizaciones del *dashboard* deben ajustarse para mostrar únicamente los datos correspondientes al período seleccionado.

a) Objetivo de la prueba: Validar que el *dashboard* histórico permite a los analistas filtrar y visualizar datos históricos de manera interactiva para identificar patrones, tendencias y la estrategia de precios de la competencia a lo largo del tiempo.

b) Pasos para ejecutar la prueba:

1. Iniciar sesión.

2. Navegar al módulo "*Dashboard* de Datos Históricos".
3. Ingresar en la barra de búsqueda el nombre de un producto específico.
4. Verificar que el gráfico de líneas se actualice mostrando la evolución de precios de ese producto.
5. Utilizar el filtro de fechas para limitar el análisis a los últimos tres meses.
6. Confirmar que todos los gráficos se reajustan correctamente al nuevo marco temporal.

Tabla 17.
Resultados de la prueba PA005

Resultado esperado	Resultado observado
Los filtros interactivos (barra de búsqueda, fecha) actualizan todos los gráficos en tiempo real.	Los filtros respondieron de manera instantánea y las visualizaciones se actualizaron correctamente.
Las visualizaciones presentan los datos históricos de forma clara y precisa.	Los datos mostrados en los gráficos fueron consistentes con la información de la base de datos.
El *dashboard* permite comparar fácilmente las estrategias de precios entre *retailers*.	Los gráficos comparativos permitieron una clara identificación de tendencias y posicionamiento de la competencia.

c) Estado de la prueba: Aprobada
d) Comentarios: La funcionalidad de análisis histórico cumplió con los criterios, proveyendo una herramienta visual efectiva para la investigación de mercado.

Tabla 18.
Prueba de aceptación PA006

PA006: Diagnostico y validación del modelo predictivo	
Funcionalidad	*Dashboard* de análisis del modelo
Fecha de ejecución	23/10/2025
Versión del sistema	3
Entrada esperada	Salida esperada
Acceso a la URL por el analista de datos	El sistema muestra todas las métricas de rendimiento (R^2, MAE), gráficos de residuos y gráficos de interpretabilidad (SHAP, PDP).
Acceso a la URL por el analista de precios	El sistema debe denegar el acceso mostrando un mensaje de permisos insuficientes.
Revisión de las métricas de rendimiento	Los valores de R^2 y MAE mostrados deben ser consistentes con los resultados del último entrenamiento del modelo.

a) Objetivo de la prueba: Validar que el *dashboard* de análisis del modelo es accesible únicamente por roles autorizados y que presenta de forma correcta las métricas y visualizaciones necesarias para que el Analista de Datos pueda auditar, interpretar y validar la salud del modelo *de Machine Learning*.

b) Pasos para ejecutar la prueba:

1. Iniciar sesión como "Analista de Datos".
2. Navegar al módulo " Análisis del Modelo".
3. Verificar que se cargan los gráficos de rendimiento (Real vs. Predicho) y de interpretabilidad (resumen SHAP).
4. Anotar los valores de las métricas clave.
5. Cerrar sesión e iniciar sesión como "Analista de Precios".
6. Intentar acceder a la URL del *dashboard* de análisis del modelo y verificar que el acceso sea denegado.

Tabla 19.
Resultados de la prueba PA006

Resultado esperado	Resultado observado
Solo los roles autorizados (Analista de Datos, Administrador) pueden acceder al *dashboard*.	El control de acceso basado en roles funcionó correctamente.
Todas las métricas y gráficos del modelo se muestran correctamente.	Las visualizaciones se cargaron sin errores y los datos eran consistentes.
La información es útil para diagnosticar el comportamiento del modelo.	Los gráficos SHAP y de residuos permitieron una interpretación clara del modelo.

c) Estado de la prueba: Aprobada

d) Comentarios: El módulo de análisis del modelo demostró ser una herramienta técnica robusta y segura, cumpliendo su objetivo de proveer transparencia sobre el motor predictivo.

Tabla 20.
Prueba de aceptación PA007

PA007: Gestión de usuarios y parámetros del sistema	
Funcionalidad	Panel de administración
Fecha de ejecución	23/10/2025
Versión del sistema	3
Entrada esperada	Salida esperada
Acceso al panel por el administrador.	El sistema muestra las opciones para gestionar usuarios y editar parámetros de configuración.
Modificación de un parámetro del sistema (ej. *MAIN_JOB_SCHEDULE_TIME*).	El sistema guarda el nuevo valor y muestra un mensaje de confirmación.
Creación de un nuevo usuario	El nuevo usuario es añadido a la base de datos y puede iniciar sesión con sus credenciales y rol asignado.
Acceso a la URL del panel por un no administrador	El sistema debe negar el acceso

a) Objetivo de la prueba: Validar que el panel de administración permite la gestión segura de usuarios y la configuración de parámetros clave del sistema, y que su acceso está estrictamente restringido al rol de Administrador.

b) Pasos para ejecutar la prueba:

1. Iniciar sesión como administrador.
2. Navegar al panel de administración.
3. Cambiar el valor del parámetro de la hora de ejecución del *scraper* y guardar.
4. Verificar que el cambio se ha guardado correctamente y ha tenido efecto sobre la planificación de ejecución del *scraper*.
5. Acceder a la sección de gestión de usuarios y crear una nueva cuenta con el rol "Analista de Precios".
6. Cerrar sesión e intentar iniciar sesión con las credenciales del nuevo usuario para confirmar su creación.
7. Iniciar sesión como "Analista de Datos" e intentar acceder al panel de administración.

Tabla 21.
Resultados de la prueba PA007

Resultado esperado	Resultado observado
El Administrador puede modificar parámetros y gestionar usuarios.	Todas las funcionalidades del panel de administración operaron como se esperaba.
Los cambios en la configuración se guardan y aplican correctamente.	El parámetro modificado fue persistido en la base de datos.
El acceso está restringido únicamente al rol de Administrador.	El control de acceso funcionó correctamente, denegando el ingreso a otros roles.

c) Estado de la prueba: Aprobada
d) Comentarios: El panel de administración es funcional y seguro, permitiendo la correcta gestión del sistema sin requerir intervención directa en la base de datos.

4.4. Análisis de los resultados

La implementación del sistema de análisis de precios para el *retailer* colaborador ha demostrado la viabilidad de transformar datos web dispersos en recomendaciones estratégicas mediante la integración sinérgica de tres componentes tecnológicos fundamentales. El sistema constituye un *pipeline* completo que abarca desde la captura automatizada de información de mercado hasta la presentación de recomendaciones accionables a través de una interfaz web interactiva, específicamente diseñada para abordar los desafíos del sector *retail* venezolano.

Asimismo, la estrategia de *Web Scraping* híbrida implementada demostró ser técnicamente robusta y operacionalmente resiliente ante la heterogeneidad de *plataformas e-commerce* analizadas. La combinación de Selenium WebDriver para sitios con alto contenido dinámico (Kromi Market y Kalea) junto con la ingeniería inversa de APIs REST para fuentes estructuradas (TuZonaMarket) permitió superar las limitaciones técnicas que presentan cada *retailer*. Gracias a esta aproximación integrada, se logró mantener una actualización continua y eficiente del catálogo de productos, procesando 5,327 referencias en ciclos promedio de 44 minutos.

Además, el modelo predictivo basado en Random Forest emergió como el núcleo analítico del sistema, superando las expectativas de precisión con un R^2 de 0.98 y un Error Medio Absoluto de 0.15 unidades monetarias. A su vez, mediante el análisis de residuos, se

identificaron patrones que permitieron clasificar sistemáticamente los productos del *retailer* colaborador en subvalorados, lo que sugiere oportunidades de mejora en el margen, y sobrevalorados, indicando posibles riesgos comerciales. Además, la interpretabilidad proporcionada por SHAP cuantificó el impacto específico de cada variable del mercado en las recomendaciones emitidas.

Por otro lado, la interfaz desarrollada con Plotly Dash facilitó la integración del sistema en la dinámica organizacional al convertir los resultados predictivos en visualizaciones accesibles y útiles. Gracias a su integración nativa con Python, se pudieron implementar funciones avanzadas, como un simulador de escenarios en tiempo real, que permite al equipo de analistas de precios del *retailer* colaborador evaluar de manera proactiva diferentes estrategias bajo condiciones de mercado simuladas. Así, el *dashboard* dejó de ser una herramienta estática de reporte para transformarse en una plataforma interactiva de planificación estratégica.

Finalmente, la evaluación completa del pipeline confirmó que la solución desarrollada constituye un sistema integral para el análisis de precios. En este sentido, las métricas de desempeño, que abarcan desde la eficiencia en la recopilación de datos (120 productos por minuto) hasta la alta precisión predictiva (R^2=0.98), evidencian la solidez técnica de la implementación. Asimismo, casos prácticos específicos, como la optimización de precios para Lentejas Pantera y Mayonesa Kraft, destacaron la aplicabilidad efectiva del sistema para fortalecer la posición competitiva del *retailer* colaborador en el dinámico sector *retail* venezolano, respaldando así la hipótesis central del estudio.

Capítulo V. Conclusiones y Recomendaciones.

5.1. Conclusión General

El desarrollo del sistema de análisis y optimización de precios para el *retailer* colaborador demostró la viabilidad técnica y operativa de implementar soluciones basadas en ciencia de datos en el contexto del *retail* venezolano. De igual forma, la integración exitosa de técnicas de *web scraping, machine* learning y visualización interactiva permitió transformar datos dispersos del mercado en recomendaciones accionables para la gestión de precios, validando el enfoque metodológico que combina *eXtreme Programming* con el

proceso KDD. Asimismo, se garantizó un compromiso ético en la recopilación y manejo de datos, asegurando prácticas responsables que respetan la privacidad y los límites establecidos, lo cual es fundamental para la aceptación y sostenibilidad del sistema en el entorno empresarial.

De esta manera, el sistema logró cumplir con el objetivo principal de proporcionar al *retailer* colaborador una herramienta efectiva para la toma de decisiones de precios basada en datos objetivos del mercado. La solución implementada permite no solo reaccionar ante movimientos competitivos, sino anticipar oportunidades de optimización mediante el análisis predictivo, representando un avance significativo respecto a métodos tradicionales de fijación de precios.

5.2. Conclusiones especificas

5.2.1. Respecto al *pipeline* de datos

La arquitectura de *web scraping* híbrida demostró ser eficaz para enfrentar los desafíos técnicos que plantea la diversidad de plataformas *e-commerce* en el mercado venezolano. La combinación de Selenium para gestionar sitios dinámicos y el consumo de APIs para fuentes estructuradas permitió procesar consistentemente más de 5,000 productos en ciclos inferiores a 45 minutos. Igualmente, el proceso de homologación de productos mediante técnicas *de fuzzy string matching* fue fundamental para garantizar la comparabilidad de los datos entre el *retailer* colaborador y sus competidores. Además, la implementación de reglas de estandarización para nombres, unidades de medida y cantidades facilitó la construcción de un *dataset* consolidado, que sirvió como base confiable para el modelado predictivo.

La selección del algoritmo *Random Forest* demostró ser acertada para el problema de optimización de precios, equilibrando precisión predictiva ($R^2 = 0.98$) con interpretabilidad de resultados. Debido a que el modelo no solo generó recomendaciones precisas, sino que a través de técnicas SHAP permitió comprender los factores detrás de cada predicción, facilitando su adopción por parte del equipo comercial de la empresa colaboradora.

Además, el análisis de residuos reveló que los errores de predicción, lejos de ser una limitación, constituyen una fuente valiosa de inteligencia de negocio. Los productos

identificados como *outliers* representaron oportunidades concretas de optimización, demostrando que el modelo funciona efectivamente como un sistema de diagnóstico para identificar desalineaciones en la estrategia de precios.

5.2.2. Respecto a la interfaz de usuario

El *dashboard* desarrollado con *Plotly Dash* demostró su efectividad para cerrar la brecha entre la complejidad técnica del modelo y las necesidades prácticas de los usuarios finales. La integración de visualizaciones intuitivas con funcionalidades avanzadas como el simulador de escenarios transformó el sistema de una herramienta de análisis a una plataforma de planificación estratégica.

Por otra parte, la validación con datos reales confirmó que las recomendaciones generadas por el sistema mantienen su relevancia en escenarios de mercado dinámicos. Casos como las Lentejas Pantera y la Mayonesa Kraft demostraron la capacidad del modelo para generalizar patrones competitivos y generar recomendaciones que se alinean con estrategias comerciales efectivas

5.3. Logros del trabajo

El principal logro de esta investigación reside en haber desarrollado un sistema integral que aporta a la resolución de un problema empresarial real, mediante la aplicación metódica de técnicas de ciencia de datos y construcción de software. Más específicamente, la solución muestra que es posible implementar una solución *end-to-end* que abarque desde la captura de datos crudos hasta la presentación de información que permita tomar decisiones estratégicas, con base en los estándares de calidad de los resultados obtenidos en cada etapa del proceso.

Metodológicamente, se validó la efectividad de combinar principios ágiles (XP) con procesos estructurados de ciencia de datos (KDD) para proyectos de esta naturaleza. Del mismo modo, el desarrollo por versiones incrementales permitió una entrega temprana de valor, mientras se mantenía el rigor en el procesamiento de datos y validación de modelos. Por su parte, el proceso de investigación fue guiado por la Investigación Acción, involucrando a los participantes como actores principales de dicho proceso.

Cabe hacer mención, en este apartado que cada proceso se realizó siguiendo las normas éticas.

5.4. Limitaciones identificadas

La principal limitación técnica identificada reside en la dependencia del sistema respecto a la estabilidad de las fuentes de datos externas. Particularmente, cambios no anunciados en la estructura de los sitios *web* monitoreados, pueden afectar temporalmente la capacidad de extracción, requiriendo intervención manual para actualizar los selectores de *scraping*.

En cuanto al modelo predictivo, se identificó que su rendimiento se ve afectado por productos con comportamientos atípicos en sus precios, que no son capturados por las variables actuales. Casos como IBERIA - PIMIENTA BLANCA MOLIDA 65GR (encontrado en la figura 11) evidenciaron la necesidad de incorporar variables adicionales relacionadas con categorización de productos y estrategias de marca.

Por otra parte, el proceso de consolidación de datos presentó sus propios desafíos, especialmente en la etapa de *product matching*. Si bien se implementó una técnica de "*fuzzy string matching*" para homologar los productos de diferentes tiendas, este método pese a su efectividad no está exento de imprecisiones. La variabilidad en los nombres de los productos entre los diferentes *retailers* puede ocasionar emparejamientos incorrectos o la omisión de equivalencias válidas, introduciendo ruido en el conjunto de datos sobre el cual se entrenó el modelo.

De manera similar, el alcance de las variables predictivas constituye otra limitación del modelo. El análisis de importancia de características, confirmó que su capacidad predictiva depende de forma predominante de los precios de la competencia. Aunque esto valida la hipótesis central del estudio, esta dependencia implica que el modelo es menos sensible a otras dinámicas de mercado que influyen en la fijación de precios, como promociones, disponibilidad de inventario (*stock*) o días festivos. Por lo tanto, su rendimiento podría verse afectado frente a estrategias competitivas que no se manifiesten exclusivamente a través de ajustes de precios directos.

5.5. Recomendaciones

Como extensión inmediata de este trabajo, se recomienda la incorporación de variables macroeconómicas y estacionales que permitan capturar patrones de demanda más complejos y mejorar la precisión predictiva en productos sensibles a factores externos.

Otra línea de investigación prometedora pudiera considerar en la implementación de modelos de series temporales para capturar tendencias de largo plazo en el comportamiento de precios, complementando el enfoque *cross-sectional* actual con análisis de evolución temporal.

Desde la perspectiva técnica, se sugiere el desarrollo de un módulo de auto-recuperación para el sistema de *scraping* que permita detectar y adaptarse automáticamente a cambios en la estructura de los sitios monitoreados, reduciendo la necesidad de intervención manual.

Finalmente, se propone la integración con sistemas internos del *retailer* colaborador para incorporar variables operativas como niveles de inventario, costos de adquisición y datos de ventas en tiempo real, creando así un ecosistema de gestión de precios aún más integral.

Referencias Bibliográficas

Abodayeh, A., Shihadeh, L., Hejazi, R., Latif, R., & Najjar, W. (2023). *Web Scraping for Data Analytics:* A BeautifulSoup Implementation. 2023 IEEE/WDS/FRJK International Conference. https://doi.org/10.1109/WDS-FRJK1971.2023.00235

Agile Proccedings (2023) *Agile Processes in Software Engineering and Extreme Programming: 24th International Conference on Agile Software Development, XP 2023, Amsterdam,* The Netherlands, June 13–16, 2023, Proceedings. (2023). Suiza: Springer Nature Switzerland.

Almohammadi, B. O. (2019). *How Business Intelligence Can Help You to Better Understand Your Customers.* ResearchGate. Recuperado de https://www.researchgate.net/publication/338326425_How_Business_Intelligence_Can_Help_You_to_Better_Understand_Your_Customers

Boring Owl. (2023, Mayo 31). *The Power of Web Scraping for Business Intelligence.* Software House Boring Owl. Recuperado de https://boringowl.io/en/blog/why-web-scraping-is-important-for-business-intelligence

Bradbury, H. (2015). The SAGE handbook of action research (H. Bradbury-Huang, Ed.; 3a ed.). SAGE Publications.

Chang, C.-Y., & He, X. (2025). The Liabilities of Robots.txt. arXiv preprint arXiv:2503.06035. https://arxiv.org/abs/2503.06035

Cobo, L., Benítez Baldión, S., Perdomo González, J. S., & Novoa Mendoza, H. A. (2024). Web scraping en supermercados para el seguimiento de precios de la cesta básica alimentaria. Ciencia e ingeniería, 12(1), e14533404. https://doi.org/10.5281/zenodo.14533404

D'Souza, M., Agrawal, D., Desai, S., & Joshi, F. (2024). *Web Scraping based Product Comparison Model for E-Commerce Website. Journal of Emerging Technologies and Innovative Research (JETIR).*

Ernest T. Stringer (2007) Action research third edition (3rd. ed.). Sage publications

Han, J., Pei, J., & Tong, H. (2022). *Data mining: Concepts and techniques* (4a ed.). Morgan Kaufmann.

Intelligence Node. (s.f.). *The Importance of Competitor Price Comparison.* Recuperado de https://www.intelligencenode.com/blog/the-importance-of-competitor-price-comparison/

Jean McNiff, Jack Whitehead. (2002). *Action Research: Principles and Practice* (2da ed.)

Joyanes Aguilar, L. (2020). Inteligencia de negocios y analítica de datos. España: Marcombo.

Kaizen Institute. (s.f.). *Optimizing Business Intelligence for Strategic Advantage.* Recuperado de https://kaizen.com/insights/business-intelligence-strategic-advantage/

Kent Beck, Cynthia Andres. (2004). *Extreme Programming Explained: Embrace Change, Second Edition*

Khder, M. A. (2021). *Web Scraping or Web Crawling; State of Art, Techniques, Approaches and Application. International Journal of Advanced Soft Computing and Applications.*

Nagle, T. T., & Muller, G. (2018). *The strategy and tactics of pricing: A guide to growing more profitably (6a ed.). Routledge.*

Nouel, F. & Rodríguez, R. (2024). Adaptación al cambio tecnológico: Diagnóstico de la transformación digital empresarial en Venezuela. Disponible en: https://www.researchgate.net/publication/385659647_ADAPTACION_AL_CAMBIO_TECNOLOGICO_DIAGNOSTICO_DE_LA_TRANSFORMACION_DIGITAL_EMPRESARIAL_EN_VENEZUELA

Nuñez Cartolin, C. A. (2021). *Business Intelligence y su impacto en la productividad del proceso de toma de decisiones de la alta gerencia en la empresa Newocean Technology S.A.C.* (Trabajo de suficiencia profesional). Universidad Privada del Norte, Lima, Perú.

Papadopoulou, M. I. (2023). *Understanding Business Intelligence and Dashboard Creation: An In-depth Analysis and Conceptual Framework (GRI-2023-41843) [Graduate thesis, Aristotle University of Thessaloniki, School of Sciences, Department of Informatics].*

Pertuz, C. M. P. (2022). Aprendizaje automático y profundo en python. Ra-Ma Editorial. Observatorio Venezolano de Finanzas (OVF). (Marzo 2025). https://observatoriodefinanzas.com/inflacion-en-venezuela-febrero-2025-con-un-12-8-mensual-y-117-interanual/

Price2Spy. (2024, nov). *What is Price Scraping?* Price2Spy Blog. Recuperado de https://www.price2spy.com/blog/price-scraping/

Torres Benitez, G. E. (2020). Modelo de inteligencia de negocios como herramienta para la toma de decisiones en el ámbito gerencial: Caso Iterauto PrimIum C.A., Naguanagua Edo. Carabobo 2020 [Trabajo de Grado de Maestría]. Universidad Jose Antonio Paez.

Valecillos, O. (2019). Desarrollo de un sistema de recomendaciones para un sitio de Comercio Electrónico (Trabajo Especial de Grado). Universidad Central de Venezuela, Caracas.

Venigandla, K., Vemuri, N., Thaneeru, N., & Tatikonda, V. M. (2023). *Leveraging AI-Enhanced Robotic Process Automation for Retail Pricing Optimization: A Comprehensive Analysis. Journal Of Knowledge Learning and Science Technology ISSN 2959-6386 (Online)*, *2*(2), 361-370. https://doi.org/10.60087/jklst.vol2.n2.p370

Singh, H. (2023). Retail price optimization. Kaggle. https://www.kaggle.com/code/harshsingh2209/retail-price-optimization/notebook

López Martín, P. (2023). Técnicas de machine learning e interpretabilidad aplicadas al mercado inmobiliario [Trabajo Fin de Grado, Universidad Complutense de Madrid].

Dib, L., & Capus, L. (2025). *Classifying XAI Methods to Resolve Conceptual Ambiguity. Technologies*, 13(9), 390. https://doi.org/10.3390/technologies13090390

Brown, M. A., Gruen, A., Maldoff, G., Messing, S., Sanderson, Z., & Zimmer, M. (2024, oct). *Web scraping for research: Legal, ethical, institutional, and scientific considerations.* https://arxiv.org/abs/2410.23432

Lakshmi, A. (2025, febrero). *Ethical web scraping: A practical guide to responsible data collection*. Scraperapi. https://www.scraperapi.com/web-scraping/ethical/

R cuadrado en R: interpretación y cálculo. (2024, julio). IONOS Digital Guide. https://www.ionos.com/es-us/digitalguide/paginas-web/desarrollo-web/r-squared-in-r/

Lhoťanová, K. (2025, febrero). *How to reverse engineer website APIs*. Apify Blog. https://blog.apify.com/reverse-engineer-apis/