Pontificia Universidad Católica del Ecuador | Sede Ambato

ESCUELA DE HABITAT, INFRAESTRUCTURA Y CREATIVIDAD

Tema:

AUTOMATIZACIÓN DE DATOS CRUDOS MEDIANTE DATA WRANGLING PARA ANALÍTICA DE VENTAS EN LA EMPRESA JOSEPHINE

Proyecto de investigación previo a la obtención del título de Ingeniero en Sistemas de la Información

Línea de Investigación:

SISTEMA DE INFORMACIÓN EMPRESARIAL, TECNOLOGÍA DE LA INFORMACIÓN Y COMUNICACIÓN, TRANSFORMACIÓN DIGITAL EN PYMES

Autor:

Christian Fernando Paredes Tamayo

Director:

Mg. Edison Fernando Meneses Torres

Ambato – Ecuador

Julio 2026

DECLARACIÓN DE AUTENTICIDAD Y RESPONSABILIDAD

Yo: **CHRISTIAN FERNANDO PAREDES TAMAYO**, con cédula de ciudadanía 1804259305, autor del trabajo de graduación titulado: **AUTOMATIZACIÓN DE DATOS CRUDOS MEDIANTE DATA WRANGLING PARA ANALÍTICA DE VENTAS EN LA EMPRESA JOSEPHINE**, previa a la obtención del título profesional de **INGENIERO EN SISTEMAS DE INFORMACIÓN**, en la escuela de **HÁBITAT, INFRAESTRUCTURA Y CREATIVIDAD.**

1. Declaro tener pleno conocimiento de la obligación que tiene la Pontificia Universidad Católica del Ecuador, de conformidad con el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.
2. Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través del sitio web de la Biblioteca de la PUCE Ambato, el referido trabajo de graduación, respetando las políticas de propiedad intelectual de la Universidad.

Ambato, julio 2026

Christian Fernando Paredes Tamayo

CC. 1804259305

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

SEDE AMBATO

APROBACIÓN DEL TRIBUNAL DE GRADO

Tema:

AUTOMATIZACIÓN DE DATOS CRUDOS MEDIANTE DATA WRANGLING PARA ANALÍTICA DE VENTAS EN LA EMPRESA JOSEPHINE

Línea de investigación:

TECNOLOGÍAS DE LA INFORMACIÓN Y LA COMUNICACIÓN

Autor:

Christian Fernando Paredes Tamayo

Edison Fernando Meneses Torres, Ing. Mg. f._____
CC. 1714355482
CALIFICADOR

Liliana del Rocío Mena Hernández, Ing. Mg. f._____
CALIFICADOR

Enrique Xavier Garcés Freire, Ing. Mg. f._____
CALIFICADOR

Francisco Javier Echeverría, Ing. Mg. f._____
DIRECTOR ESCUELA HÁBITAT, INFRAESTRUCTURA Y CREATIVIDAD

Diego Gonzalo Coca Chanalata, Dr. Mg. f._____
PROSECRETARIO PUCE AMBATO

PUCE | AMBATO
PROSECRETARÍA

Ambato – Ecuador
Julio 2026

DEDICATORIA

Al Santo, bendito sea, por ser la luz que ilumina cada uno de mis pasos, por concederme la sabiduría necesaria para alcanzar esta meta y por sostenerme con su gracia infinita a lo largo de toda mi carrera profesional.

A mis padres Christian Fernando Paredes Armas y María Fernanda Tamayo Ramírez por su amor incondicional, el sacrificio y su apoyo inquebrantable. Gracias por creer en mí, por sus consejos por ser siempre el motor que me ha llevado a cumplir mis metas con integridad y perseverancia. Este logro es, ante todo el fruto de la dedicación y esfuerzo que invirtieron en mí. Gracias por ser mi ejemplo de que me puedo superar, por cada sacrificio silencioso para que nunca me faltara nada y por enseñarme que el conocimiento es una herramienta para ayudar a los demás.

Christian Fernando Paredes Tamayo

AGRADECIMIENTO

Agradezco a mi padre, Christian Fernando Paredes Armas, por ser mi guía y el ejemplo de trabajo que sigo cada día; gracias por el apoyo constante y por enseñarme a que mi determinación en algo hace alcanzable cualquier meta profesional.
A mi madre, María Fernanda Tamayo Ramírez por darme siempre el amor infinito de una madre por ser mi refugio cuando necesito paz en mi vida que me sostiene durante los momentos más exigentes; este logro es reflejo de toda tu entrega y los valores que inculcaste en mí.
A mi abuela, Inés Armas, por sus oraciones, su ternura y por ser la base espiritual que siempre me ha dado aliento; gracias por estar presente en cada paso de mi vida con tu sabiduría y cariño.
A mi abuelo, René Paredes, por su ejemplo de rectitud y por transmitirme la fortaleza necesaria para enfrentar los desafíos; tu legado de esfuerzo es una de las razones por las que hoy culmino este proyecto.
A mis mejores amigas, Romina Hidalgo y Doménica Escobar, por ser un pilar fundamental en mi vida; gracias por su lealtad incondicional, por compartir conmigo la vida y por ser ese apoyo sincero que siempre me motiva a seguir adelante.
Christian Fernando Paredes Tamayo

RESUMEN

La Comercializadora ”Josephine”, dedicada a la importación y reventa de ropa casual en cinco sucursales, enfrentaba un desafío crítico en la gestión de su información: la ausencia de un sistema centralizado obligaba al personal administrativo a consolidar manualmente los registros de ventas en formato Excel, generando datos sucios en cuatro categorías recurrentes —registros duplicados, costos unitarios en cero, ventas bajo precio de costo y campos de talla ausentes— que comprometían la veracidad de los reportes y consumían días o semanas por ciclo de consolidación, impidiendo decisiones estratégicas oportunas sobre inventario, rentabilidad y desempeño por sucursal. Para resolver esta problemática, se implementó un *framework* de Data *Wrangling* automatizado en Python con la librería Pandas sobre un corpus histórico de 73.981 registros del período enero 2022 a mayo 2025. Mediante un pipeline ETL de seis etapas se identificaron y gestionaron 11.202 anomalías (15,14 % del universo inicial); el resultado fue un dataset maestro de 72.795 registros íntegros que representan el 98,40 % del corpus original, con tiempos de ejecución reducidos a segundos frente al proceso manual previo y sin requerir GPU ni licencias propietarias. El repositorio procesado fue integrado con Microsoft Power BI a través de una arquitectura desacoplada de tres capas —transformación en Python, almacenamiento en OneDrive y visualización en Power BI— sobre la cual se construyeron cuatro tableros interactivos orientados al resumen ejecutivo de ventas, análisis de rentabilidad por sucursal, monitoreo de ventas bajo costo y rotación de prendas por talla. Esta solución eliminó la brecha informativa preexistente y permitió a la gerencia transitar de una gestión reactiva hacia una cultura orientada a datos, con decisiones respaldadas por información íntegra y oportuna.

Palabras clave: *Data Wrangling, Business Intelligence*, Python, Power BI, calidad de datos, analítica de ventas, ETL, PYMES.

ABSTRACT

”Josephine”Trading Company, dedicated to the importation and resale of casual clothing across five branches, faced a critical information management challenge: the absence of a centralized system forced administrative staff to manually consolidate sales records exported in Excel format from each point of sale, producing dirty data across four recurring categories —duplicate records, zero unit costs, below-cost sales, and missing size fields— that undermined report accuracy and consumed days or weeks per consolidation cycle, preventing management from accessing timely information for strategic decision-making on inventory, profitability, and branch performance. To address this problem, an automated Data Wrangling framework was implemented in Python using the Pandas library as the primary processing engine, applied to a historical corpus of 73,981 commercial transaction records spanning January 2022 to May 2025. Through a six-stage sequential ETL pipeline, 11,202 anomalies were identified and resolved, representing 15.14 % of the initial universe, yielding a master dataset of 72,795 complete records accounting for 98.40 % of the original corpus, with execution times reduced to seconds compared to the previous manual process and without requiring GPU resources or proprietary database licenses. The processed repository was integrated with Microsoft Power BI through a decoupled three-layer architecture —Python transformation, OneDrive institutional storage, and Power BI visualization— upon which four interactive dashboards were built, targeting executive sales summaries, branch profitability analysis, below-cost sales monitoring, and garment turnover by size and branch. This solution eliminated the preexisting information gap and enabled management to transition from reactive, intuition-based decision-making toward a data-driven culture in which every investment and restocking decision is supported by complete, timely, and bias-free information.

Keywords: Data Wrangling, Business Intelligence, Python, Power BI, data quality, sales analytics, ETL, SMEs.

ÍNDICE GENERAL DE CONTENIDOS

INTRODUCCIÓN

En la actualidad, con la transformación digital, la capacidad de las empresas para convertir datos crudos en información estratégica es un factor determinante para que estas puedan ser competitivas (Corporación Universitaria Republicana & Bermúdez Irreño, 2022). Sin embargo, muchas empresas pequeñas y medianas enfrentan como obstáculo significativo la falta de estructuras de datos estandarizados, como consecuencia genera procesos de información ineficientes que derivan en errores humanos.

La comercializadora “Josephine”, dedicada a la importación y reventa de ropa casual, ejemplifica la problemática de la gestión de datos heterogéneos. Actualmente, la organización procesa grandes volúmenes de información provenientes de múltiples puntos de venta en formatos de hojas de cálculo inconsistentes. Esta diversidad de fuentes genera datos con errores de nomenclatura, registros duplicados y fallos de digitación manual. De acuerdo con (Álvarez & Romero, 2025), la dependencia de métodos artesanales para la depuración de información incrementa el uso de horas hombre y compromete la integridad de los reportes. En consecuencia, la falta de una limpieza automatizada afecta directamente la veracidad de los insumos necesarios para la toma de decisiones estratégicas.

A nivel macro, el estado global de la tecnología de la información se destaca por la explotación de datos generados. Según (García-Jiménez, Aguilar-Morales, Hernández-Triano, & Lancaster-Díaz, 2021) los sistemas de información empresarial son actualmente el fundamento para realizar negocios, y su nivel de eficacia depende de la integridad de los datos que alimentan a estos. Las políticas de transformación digital a nivel mundial piden que las organizaciones adquieran herramientas vanguardistas de procesamiento. Sin embargo, como señalan (Ordoñez Abril, Amaya López, Lucio Valencia, & Rodríguez Gómez, 2023) el crecimiento de las fuentes de datos diversas ha creado un “cuello de botella” donde la inteligencia de negocios se ve retrasada por la baja calidad en la información inicial. Esta realidad macroeconómica obliga a las empresas, independientemente de su tamaño, a buscar soluciones para automatización que minimicen el error de parte humana.

A nivel meso, el contexto de las pequeñas y medianas empresas dentro de América Latina y específicamente en Ecuador, se demuestra una brecha tecnológica constante. Muchas organizaciones, a pesar de haber realizado una digitalización de sus

registros, continúan con procesos de forma manual para la elaboración de reportes. Según estudios de(Delen, 2024), la falta de una estabilización de los formatos de intercambio de datos genera una “deuda técnica” que afecta en especial a la agilidad empresarial. En este apartado, se observa que la implementación de un sistema de inteligencia de negocios (BI) a menudo fracasa no por la herramienta de visualización (ejemplo: Power BI) sino por una incorrecta preparación previa de los datos, lo que genera una expectativa poco realista sobre la rapidez de la obtención de los resultados estratégicos.

A nivel micro, las interacciones diarias en la comercializadora “Josephine” demuestran las consecuencias de carecer de procesos automatizados. El personal de administración se encuentra con la recepción de archivos Excel con estructuras inconsistentes, nombres de productos duplicados y errores de digitación. Según (Alvarado-Apodaca, Ramírez-Noriega, Tripp-Barba, Martínez-Ramírez, & Sánchez, 2023), cuando los analistas dedican la mayor parte de su tiempo a la limpieza de datos en lugar de generar un análisis, la percepción de la eficiencia operativa disminuye de manera importante. En “Josephine”, este tiempo operativo impide una respuesta rápida ante las tendencias del mercado de ropa casual, lo que provoca que la toma de decisiones se base en reportes que podrían contener errores.

Investigaciones Nacionales e Internacionales

dentro del ámbito internacional, existen evidencias significativas que muestran que el uso de Python para la optimización de datos en entornos comerciales. Por ejemplo, (Delen, 2024) donde se documentan casos donde la implementación de modelos de análisis avanzados permitió a empresas minoristas transformar procesos con inventarios desorganizados en información predictiva. Con estos se demuestra que el uso de librerías especializadas para el *Data Wrangling* no solo reduce el error humano, sino que establece una base sólida para el uso de herramientas de inteligencia de negocios complejas.

De igual forma, investigaciones desarrolladas dentro de Latinoamérica resaltan que la “limpieza artesanal” de datos es el principal obstáculo de la agilidad en las pequeñas y medianas empresas. Según(Alvarado-Apodaca et al., 2023), la madurez analítica dentro de una organización se mide por su capacidad de automatizar las tareas repetitivas. En este sentido, proyectos similares al *framework* propuesto han demostrado que la transición de hojas de cálculo a scripts automatizados en Python

permite una reducción de hasta el 70 % en lo que es tiempo de preparación de reportes, lo que facilita la integración con plataformas como Power BI.

En el ámbito nacional, la transformación digital en Ecuador ha cobrado importancia en los últimos años, sobre todo en el sector comercial y logístico. Investigaciones señalan que las empresas en ciudades como Ambato se enfrentan a desafíos críticos de interoperabilidad debido a la diversidad de formatos de proveedores. Al igual que la investigación de georreferenciación de (Alvarado-Apodaca et al., 2023) donde la tecnología ayudó a resolver problemas de precisión, la automatización del procesamiento de datos en la comercializadora “Josephine” busca resolver la falta de una estandarización que impide la agilidad con los recursos y una respuesta eficaz ante las demandas del mercado.

En esta investigación se justifica, en la necesidad de superar las limitaciones del procesamiento manual actual, el cual, según (Gómez-Duque, Daza-Torres, & Arias-Pérez, 2023), compromete a la capacidad de los administradores y gerentes para confiar en los sistemas de soporte a las decisiones. Al momento de implementar un framework de Data Wrangling, no solo se tecnifica un proceso operativo, sino que se proporciona a la empresa una herramienta estratégica para mejorar su posicionamiento y competitividad.

Justificación de la Investigación

La implementación de este *framework* de Data *Wrangling* se justifica desde tres dimensiones fundamentales: técnica, operativa y estratégica. Dentro de la perspectiva técnica, el cambio de un manejo de datos basado en hojas de cálculo de forma manual hacia una arquitectura automatizada dentro de Python el cual representaría un salto cualitativo en la infraestructura de datos de “Josephine”. Como señala (Alvarado-Apodaca et al., 2023), al usar scripts nos permite una escalabilidad que el procesamiento de los datos manual no puede ofrecer; esto garantiza que el sistema pueda manejar volúmenes de datos mayores sin afectar la calidad de la información.

Dentro del ámbito operativo, la justificación radica en la optimización del talento humano. Actualmente, el personal administrativo dedica un porcentaje excesivo de su jornada laboral a tareas que son repetitivas en depuración de datos. La automatización le permitiría que este tiempo se utilice en actividades de análisis y tomas de decisiones; esto reduce de manera sustancial el costo de oportunidad y el riesgo de errores humanos inherente a la digitación manual.

Desde la visión estratégica, la estandarización de los datos es el pilar para poder construir la inteligencia de negocios. Contar con información que sea íntegra y veraz en Power BI permitirá a los administradores identificar tendencias de mercado y comportamientos de las ventas en tiempo real. En un mercado que es propenso al cambio como lo es el de la ropa casual, la capacidad de reacción basada en datos confiables es lo que nos asegura la ventaja competitiva y la sostenibilidad de la empresa a largo plazo.

Como objetivo general de esta investigación es implementar una solución de automatización de procesamiento de datos basada en las técnicas de *Data Wrangling* con Python para mejorar la calidad, estandarización e integración de la información de ventas e inventarios de la empresa “Josephine”, con el propósito de fortalecer la analítica de las ventas.

Objetivos específicos

1. Levantar y catalogar las inconsistencias, errores de formato y redundancias presentes en los archivos Excel de ventas e inventarios proporcionados por las distintas sucursales de la empresa.

2. Diseñar y programar scripts en Python (con librerías especializadas) que ejecuten reglas de negocio para la limpieza, tipificación y unión de los datasets fragmentados.

3. Desarrollar un modelo de datos relacional que actúe como repositorio intermedio; esta estructura posibilita la conexión eficiente entre el proceso de limpieza en Python y la capa de visualización en Power BI.

4. Evaluar el desempeño de la solución automatizada mediante la comparación del tiempo de procesamiento, la reducción de inconsistencias y la calidad de los datos obtenidos frente al proceso manual actual.

Metodología y Enfoque del Proyecto

Para el desarrollo de esta investigación, se seleccionó la metodología ágil Kanban, adoptada como marco de gestión visual y organización del flujo de tareas del equipo de desarrollo, en complemento con la metodología CRISP-DM que estructuró las fases técnicas del proyecto (véase apartado 2.3). Este enfoque Kanban permite una gestión visual y un flujo continuo de trabajo, lo que resulta ideal para proyectos

de desarrollo de software donde la priorización de tareas de limpieza y transformación de los datos es tan importante. El uso de un tablero Kanban ayudará a la organización de los procesos de extracción, depuración y carga (ETL), con el fin de garantizar que cada etapa del *framework* sea validada antes de su integración final en los tableros de control.

Este estudio adopta un enfoque mixto. Es cualitativo en su fase inicial, este requiere un diagnóstico estricto de las inconsistencias y los tipos de “datos sucios” presentes en los archivos actuales de la comercializadora. Luego, adquiere un carácter cuantitativo al empezar a evaluar la eficiencia del sistema mediante la medición de tiempos en los que procesa y la precisión al generar reportes a comparación del método artesanal que se presenta en “Josephine”.

Esta investigación es de tipo aplicada y no experimental, debido a que el objetivo principal es resolver el problema práctico y específico mediante la aplicación de conocimientos técnicos de Python y ciencia de datos. El alcance del proyecto cubre desde la identificación de las fuentes de datos heterogéneas hasta la visualización de la información ya depurada; esta cobertura integral tecnifica la cadena de valor de la información en “Josephine”.

CAPÍTULO I. ESTADO DEL ARTE Y LA PRÁCTICA

En el presente capítulo se constituye el fundamento teórico y referencial sobre en el cual se sustenta la investigación. El propósito principal es analizar las corrientes tecnológicas y académicas actuales en las que se manejan la gestión de los datos y la inteligencia de negocios en el entorno empresarial actual. Para ello, se realiza una revisión exhaustiva de conceptos importantes como los sistemas de la información (SI), la calidad del dato (*Data Quality*) y el proceso de *Data Wrangling*, elementos que juntos permiten transformar la información bruta en un activo estratégico para las organizaciones. Este análisis fundamenta el estudio de las dos variables de la investigación: el framework de Data Wrangling automatizado con Python, como variable independiente, y la calidad de la analítica de ventas, como variable dependiente.

A través de este análisis, se busca dimensionar el impacto negativo tanto como limitaciones y vulnerabilidades dentro del proceso de tratamiento de datos de manera manual dentro de las medianas y pequeñas empresas.(Alvarado-Apodaca et al., 2023) La gestión de la información basada principalmente por hojas de cálculo tradicionales facilita a la aparición de errores operativos y genera importantes brechas de eficiencia.

1.1. Sistemas de Información Empresarial

En la era de la transformación digital, los sistemas de información (SI) se han consolidado como algo fundamental dentro de las organizaciones. Un sistema de información es un conjunto de componentes que se relacionan y se encargan de recolectar, procesar, almacenar, y distribuir la información para apoyar la toma de decisiones, la coordinación y el control en una organización(Rodríguez Baque, 2023). La efectividad de un SI está profundamente ligada a la pureza de sus datos. Como señala(Alvarado-Apodaca et al., 2023), un sistema de información es tan fuerte como el dato más débil que este procese. En muchas empresas dentro del sector retail en Ecuador, los sistemas suelen sufrir de un fenómeno conocido como “fragmentación de datos”, donde la información se encuentra en archivos aislados que no tienen comunicación entre sí. Esta falta de integración provoca que el sistema de informa-

ción sea incapaz de cumplir su función de soporte a las decisiones, convirtiéndose en un repositorio de datos aislados en lugar de una herramienta de conocimiento.

La evolución de estos sistemas ha llevado a la necesidad de implementar capas intermedias para el procesamiento. Ya no basta con tener una base de datos; es fundamental contar con mecanismos que aseguren que los datos que alimentan al sistema sean consistentes y se encuentre debidamente estandarizados(Rajpoot, Tiwari, & Vishwakarma, 2026). Se considera imperativo el factor de la “escalabilidad” dentro de los sistemas de información modernos. Un sistema que depende de la intervención humana constante para la limpieza de sus datos está condenado a la deficiencia a medida que el volumen de datos crece.los sistemas de información deben tener la capacidad de adaptarse a las demandas cambiantes del mercado sin comprometer la integridad de la base de datos(Roberts & Massoud, 2025). Para la comercializadora de ropa casual, esto significa que el sistema debe tener la capacidad de procesar cientos de registros diarios de forma estandarizada; de esta manera la infraestructura tecnológica soporta el crecimiento del negocio en lugar de convertirse en una barrera que limite su expansión por la acumulación de errores no resueltos(Unocc Antonio & De La Cruz Llacctahuaman, 2024).

1.2. Inteligencia de Negocios (BI) y Analítica de Ventas

La inteligencia de Negocios (BI) representa el conjunto de estrategias y herramientas que sirven para transformar datos de información, y esta última en conocimiento optimizado para el plan de negocios. (Delen, 2024) enfatiza que el BI no es el producto final, sino un proceso en avance que comienza con la captura del dato y termina con la visualización de indicadores claves (KPIs). Para “Josephine”, el BI se manifiesta con la necesidad de entender que productos tienen mayor aceptación en los clientes y que sucursales presentan mejores márgenes de rentabilidad.

El componente crítico del BI es la analítica de ventas. Según (Delen, 2024), la analítica permite pasar de un enfoque descriptivo a uno diagnóstico. Sin embargo, este salto analítico se hace imposible si los datos de origen están contaminados. En la analítica de ventas para la ropa casual, la estacionalidad y las tendencias de moda son factores que exigen velocidad. Si la empresa tarda días en obtener los datos, la oportunidad de negocio se pierde.

La implementación de Power BI como capa de visualización ha ayudado al acceso a la información en las PYMES. No obstante, la literatura técnica coincide en que el 80 % del éxito de un proyecto de BI depende de la fase de preparación de datos. Esta investigación sostiene que el BI en ”Josephine”no debe quedarse en el tablero final, sino desde la ingeniería del dato inicial. mediante una estandarización de los inventarios y las ventas, el BI podrá generar proyecciones de demanda que realmente sean útiles para la gerencia, lo que reduce el stock que no se mueve y mejora la cadena de suministro.

Una estrategia de *Business Intelligence* bien planteada exige entender el ciclo de vida del dato, desde que entra en el punto de venta hasta su transformación en un indicador visible. La analítica de ventas para el sector retail no puede limitarse a reportes históricos; debe aspirar a una visión en tiempo real que permita la optimización de existencias. Según (Delen, 2024), la diferencia entre una PYME estancada y una competitiva radica en el uso de la analítica descriptiva para corregir desviaciones operativas de forma inmediata. En ”Josephine”, esto se traduce en la capacidad de detectar qué tallas o modelos de ropa presentan menor velocidad de rotación, lo que permite implementar liquidaciones estratégicas o ajustes en las compras a proveedores antes de que el stock se convierta en una pérdida financiera.

Finalmente, el componente humano en el BI juega un papel crucial, pero supeditado a la confianza en la herramienta. Si los gerentes perciben que los tableros de control en Power BI muestran datos inconsistentes debido a la falta de una limpieza previa, abandonarán el uso de la tecnología para volver al juicio subjetivo o ïntuición”. De acuerdo con (Delen, 2024), la .ªlfabetización de datos”(*data literacy*) dentro de una organización solo se logra cuando la herramienta de BI entrega resultados coherentes y verificables. Por lo tanto, la automatización del procesamiento mediante Python no es solo una mejora técnica, sino un paso necesario para fomentar una cultura orientada a datos en la comercializadora, donde cada decisión de inversión esté respaldada por una analítica robusta, libre de los sesgos introducidos por el error humano en las fases de preparación manual.

1.3. Automatización de datos crudos mediante Data Wrangling

La calidad de los datos no es un concepto que se pueda reducir a una sola dimensión. (Hassenstein & Vanella, 2022), lo plantean bien: la calidad se desglosa

en categorías como la intrínseca, la contextual y la representacional. En el caso de ”Josephine”, la calidad intrínseca es decir, la precisión y veracidad de los registros de venta es la que más sufre. La manipulación manual de los archivos genera un efecto dominó: un error en el origen termina distorsionando los reportes finales en Power BI, y la confianza en la herramienta se resiente.

Un estudio fundamental realizado por (Dasari & Varma, 2022), conocido como la regla del diez”, sostiene que el costo de completar una tarea con datos erróneos es diez veces mayor que el costo de realizarla con datos correctos desde el inicio. Esta ineficiencia se manifiesta en ”Josephine.ª través del tiempo que el personal administrativo debe invertir en corregir inconsistencias de formato o registros duplicados antes de poder generar un inventario real. Investigaciones de la *Data Warehousing Institute* (TDWI) estiman que las empresas pierden, en promedio, el 20 % de sus ingresos debido a la mala calidad de los datos, lo que subraya que la depuración no es solo un proceso técnico, sino una medida de protección financiera para la PYME.

La problemática de los ”datos sucios”(dirty data) incluye errores de entrada, datos faltantes y violaciones de reglas de integridad. Como indica (Espinoza Tinoco, Congacha Aushay, & Díaz Ordóñez, 2023), el proceso de limpieza o *Data Cleansing* es una etapa crítica para eliminar anomalías que podrían sesgar los resultados de cualquier modelo analítico. En el sector de la moda casual, donde las tendencias cambian trimestralmente, un dato erróneo sobre la rotación de una prenda puede llevar a compras excesivas de stock innecesario o a la falta de productos de alta demanda; esto afecta directamente la satisfacción del cliente y el flujo de caja.

Desde una perspectiva de ingeniería de sistemas, la calidad del dato también se relaciona con la ”deuda técnica”. Según (McGregor, 2021) posponer la corrección de errores en los flujos de información genera un interés acumulado en forma de procesos cada vez más lentos y difíciles de mantener. En la comercializadora, depender de Excels no estandarizados es una forma de deuda técnica que crece con cada nueva sucursal o proveedor que se suma a la red. Por lo tanto, establecer un estándar de calidad mediante automatización es la única vía para asegurar que la arquitectura de información de la empresa sea escalable y capaz de soportar el crecimiento del volumen de datos sin colapsar bajo el peso de sus propias inconsistencias.

Finalmente, la confianza de los usuarios en los sistemas de información depende enteramente de la consistencia de los resultados. Si la gerencia detecta discrepan-

cias entre los reportes de ventas y la realidad física de las bodegas, se produce una ruptura en la adopción tecnológica. De acuerdo con investigaciones de(Hassenstein & Vanella, 2022), la gobernanza de datos efectiva requiere la implementación de dimensiones de calidad como la çompletitud", que asegura que no existan vacíos en los registros históricos. El framework propuesto en esta tesis actúa como un guardián de estas dimensiones; cada registro procesado por Python cumple con los estándares de limpieza necesarios para que la analítica sea, ante todo, un reflejo fiel y confiable de la actividad comercial de la organización.

1.4. Data Wrangling hacia la analítica de ventas

La convergencia en los procesos de limpieza y transformación de datos y las plataformas de inteligencia de negocios ha dado paso a arquitecturas de integración que superan los modelos tradicionales basados en base de datos locales. En este contexto, la elección de una arquitectura desacoplada y orientada en la nube resulta importante para garantizar la escalabilidad, la disponibilidad y la actualización continua de tableros analíticos sin necesidad de intervención manual en cada ciclo de procesamiento (McGregor, 2021).

La arquitectura propuesta en el presente proyecto se estructura en tres capas funcionales interdependientes. La primera corresponde a la capa de ingestión y transformación, donde los scripts en Python tienen la función de procesar los archivos de ventas de las sucursales de la comercializadora "Josephine" durante el periodo 2022-2025. esta etapa radica en la aplicación de técnicas automatizadas de limpieza que garantizan la integridad de la información procesada (Espinoza Tinoco et al., 2023) bajo este enfoque se aplican las reglas del negocio definidas para corregir anomalías críticas: ventas registradas por debajo del precio de costo, codificaciones inconsistentes de tallas y referencias de productos, y registros con valores nulos dentro de campos estratégicos como unidades vendidas, precio de venta y código de sucursal.

La segunda capa corresponde al repositorio de datos estandarizados. A diferencia de arquitecturas convencionales que dependen de motores de bases de datos relacionales como SQL Server, el modelo adoptado en esta investigación deposita los datos procesados en una carpeta de almacenamiento institucional en la plataforma OneDrive de Microsoft 365. Esta decisión técnica obedece a tres criterios: primero,

la reducción de la deuda técnica al eliminar la dependencia de licencias y configuraciones de servidor; segundo, la disponibilidad multiplataforma que permite el acceso desde cualquier punto de la organización; y tercero, la capacidad nativa de actualización automática que ofrece la integración entre OneDrive y Power BI a través del conector de datos en la nube (Rajpoot et al., 2026).

La tercera capa es la capa de visualización y analítica, implementada en Microsoft Power BI. Los modelos de datos publicados en esta plataforma obtienen la información directamente desde la carpeta de OneDrive institucional mediante una conexión programada; este mecanismo garantiza que los dashboards comerciales, financieros y operativos reflejen siempre los últimos datos procesados por el framework de Data Wrangling sin necesidad de intervención manual adicional. Esta arquitectura desacoplada implica que cualquier actualización en el script de Python —ya sea para incorporar nuevas reglas de negocio o nuevas fuentes de datos— se propaga de manera automática hacia los tableros de control y preserva la integridad del flujo de información en su totalidad.

El valor diferencial de esta propuesta respecto a soluciones convencionales de extracción, transformación y carga (ETL) radica en su orientación hacia la agilidad organizacional descrita por(Gómez & Latournerle, 2022). Al reducir la fricción entre la recolección de datos en campo y su disponibilidad para la toma de decisiones gerenciales, el tiempo de latencia informativa —entendido como el intervalo entre la generación del dato y su visualización analítica— se reduce de forma significativa. Para una empresa del sector de la moda casual, donde la rotación de inventarios y la estacionalidad de la demanda exigen respuestas rápidas, esta capacidad de reacción constituye una ventaja operativa directa.

La arquitectura descrita permite que la analítica de ventas en ”Josephine.^{ev}olucione progresivamente desde un enfoque descriptivo —reportes históricos de unidades vendidas y márgenes por sucursal— hacia un enfoque predictivo que incorpore modelos de proyección de demanda basados en los patrones identificados en el histórico de cuatro años. De acuerdo con(Hassenstein & Vanella, 2022), la consolidación de una base de datos histórica limpia y estandarizada es el requisito previo e insustituible para la construcción de cualquier modelo analítico avanzado. El presente framework provee, precisamente, esa base: un repositorio de datos de ventas íntegro, consistente y accesible, sobre el cual la organización podrá construir capacidades analíticas de mayor complejidad en etapas posteriores de su transformación digital

El panorama investigativo reciente confirma la pertinencia del enfoque adoptado en el presente estudio. A nivel nacional, Sánchez-Lunavictoria et al. (2026) documentan que apenas el 7 % de las MIPYME ecuatorianas emplea efectivamente herramientas de inteligencia de negocios, y que en ciudades como Cuenca la hoja de cálculo Excel continúa operando como el principal —y a menudo único— instrumento de análisis. Este hallazgo, obtenido mediante un diseño documental y descriptivo, coincide plenamente con el diagnóstico situacional levantado en la comercializadora “Josephine” (véase Bloque A del Anexo 1), donde la ausencia de un repositorio centralizado y la dependencia de extracciones manuales en Excel constituían el punto de partida del problema. Sin embargo, mientras dicha investigación se limita a caracterizar la brecha de adopción a nivel de encuesta, el presente proyecto aporta una respuesta técnica implementada y medible: un framework que automatiza precisamente la etapa de preparación de datos identificada como el principal cuello de botella para el tránsito de la MIPYME desde el uso elemental de Excel hacia la analítica de ventas estructurada.

De manera complementaria, Delgado Díaz et al. (2025) emplean una metodología cualitativa exploratoria, basada exclusivamente en entrevistas semiestructuradas a propietarios de PYME, para caracterizar las ventajas percibidas de la inteligencia de negocios en áreas como finanzas y mercadeo. El instrumento de recolección coincide con el empleado en el componente cualitativo de esta investigación (Guía de Entrevista Semiestructurada, Anexo 1); no obstante, el diseño mixto adoptado en el presente estudio incorpora, adicionalmente, un componente cuantitativo mediante el EDA automatizado sobre 73.981 registros reales, lo que permite trascender la percepción subjetiva de los actores y cuantificar objetivamente la magnitud de las anomalías (11.202 registros, 15,14 % del corpus) que la sola entrevista no habría podido dimensionar.

En cuanto a las herramientas técnicas de limpieza de datos, Malla Valdiviezo et al. (2023) revisan mecanismos especializados para el procesamiento de grandes volúmenes de datos, tales como Spark, Talend y Melissa Data Clean Suite, orientados a entornos empresariales con infraestructura robusta. Frente a este panorama, el framework desarrollado para “Josephine” demuestra que es posible alcanzar niveles de depuración comparables (98,40 % de integridad del dataset maestro) empleando exclusivamente Python y Pandas, sin licencias propietarias ni infraestructura de procesamiento distribuido, lo que constituye una alternativa técnica más accesible para el perfil de recursos limitados característico de las PYME del sector comercial ecuatoriano.

CAPÍTULO II. DISEÑO METODOLÓGICO

El presente capítulo describe el diseño metodológico adoptado para el desarrollo del framework de Data Wrangling y su integración con la plataforma de inteligencia de negocios de la comercializadora ”Josephine”. La estructura metodológica responde a la naturaleza aplicada del proyecto, en el que el conocimiento técnico en ciencia de datos y programación en Python se pone al servicio de la resolución de un problema operativo concreto y verificable.

La investigación adopta un enfoque mixto. En su dimensión cualitativa, comprende el diagnóstico exhaustivo de las inconsistencias presentes en los archivos de ventas recibidos de las distintas sucursales, así como el levantamiento de los requerimientos de negocio necesarios para definir las reglas de transformación aplicables. En su dimensión cuantitativa, el trabajo se orienta a la medición objetiva del desempeño de la solución automatizada mediante indicadores como el tiempo de procesamiento, el porcentaje de reducción de registros inconsistentes y la tasa de completitud de los datos resultantes, comparados con los valores registrados durante el proceso manual previo.

2.1. Técnicas e Instrumentos

El estudio es de tipo aplicado, dado que su propósito central es resolver un problema concreto y específico del entorno empresarial mediante la aplicación de conocimientos técnicos de Ciencias de datos e Ingeniería de sistemas. No se persigue la generación de teoría nueva, sino la transferencia de conocimiento existente hacia una solución funcional que mejore la eficiencia operativa de la organización estudiada. Adicionalmente, es de alcance descriptivo-propositivo: describe el estado actual de los datos y propone una solución sistematizada para transformarlo.

El diseño de la investigación es no experimental de campo. No existe manipulación de variables; los archivos de datos y los procesos organizacionales se analizan en su estado natural, tal como se generan en la operación diaria de la empresa. La recolección se realiza directamente en el entorno de la comercializadora ”Josephine”, lo que garantiza la autenticidad y representatividad de los hallazgos. Para la

gestión del desarrollo técnico del *framework*, se adopta la metodología ágil Kanban, que permite organizar las etapas de extracción, limpieza, transformación y carga de datos en un flujo continuo y validado, donde cada módulo del *pipeline* es verificado antes de su integración con la capa de visualización en Power BI (Anderson, 2010). Esta gestión visual de tareas operó de manera complementaria a la metodología CRISP-DM, que estructuró las fases técnicas del proyecto conforme se detalla en el apartado 2.3.

Para esta investigación se identifican dos universos diferenciados que responden a la naturaleza mixta del estudio. El primero corresponde al componente humano: la totalidad del personal administrativo y operativo de la comercializadora ”Josephine”que interviene directamente en los procesos de recepción, consolidación y reporte de datos de ventas e inventarios. El segundo corresponde al componente documental: el conjunto completo de archivos Excel de ventas e inventarios generados por todas las sucursales de la empresa durante el período 2022–2025, que constituyen la materia prima sobre la cual opera el framework de Data Wrangling.

Dado el tamaño reducido y plenamente accesible del componente humano, se aplica una muestra de tipo censal que integra a la totalidad del personal que cumple los criterios de inclusión. Esta decisión elimina el error muestral y maximiza la representatividad del diagnóstico organizacional (Tamayo y Tamayo, 2014). Para el componente documental, se trabaja igualmente con el universo completo de archivos disponibles del período establecido, el Análisis Exploratorio de Datos (EDA) requiere la totalidad del corpus para obtener métricas de error estadísticamente representativas de la problemática real.

Los criterios de inclusión para el componente humano contemplan: vinculación laboral activa al momento de la recolección, participación directa en el proceso de gestión de datos de ventas o inventarios, y un mínimo de tres meses en el cargo. Se excluye el personal en período de prueba y aquel cuyas funciones no guarden relación con el manejo de información comercial. Para el componente documental, se incluyen exclusivamente archivos generados como reportes oficiales de ventas o inventarios por las sucursales de la organización durante el período 2022–2025; se excluyen documentos de uso personal o con procedencia no verificable.

Para el componente cualitativo, se emplea la técnica de la entrevista en su modalidad semiestructurada. Esta técnica permite capturar el flujo de trabajo actual, identificar los tipos de errores más frecuentes en los archivos recibidos y levantar los

requerimientos funcionales del sistema desde la perspectiva del usuario final, información que no es observable directamente en los archivos digitales. El instrumento asociado es la Guía de Entrevista Semiestructurada, dirigida al gerente general y personal administrativo de la empresa (ver Anexo 1).

Para el componente cuantitativo, se emplea la técnica de la observación documental, operacionalizada mediante un EDA automatizado ejecutado con la librería Pandas de Python. Esta técnica permite cuantificar y clasificar objetivamente las anomalías presentes en los archivos de datos bajo las dimensiones de calidad definidas por Hassenstein y Vanella (2022): integridad, consistencia, unicidad, exactitud y completitud. El instrumento asociado es la Ficha de Diagnóstico de Calidad de Datos, que consolida los hallazgos técnicos del EDA en un registro académicamente documentado.

2.2. Instrumento de Recolección de Datos

El instrumento de recolección de datos se construye a partir de la operacionalización de las variables de investigación; cada pregunta e indicador responde directamente a las dimensiones que se desea medir. A continuación, se presenta la tabla de operacionalización de variables y, posteriormente, los instrumentos derivados de ella.

Tabla 2.1. Operacionalización de las variables de investigación.

Variable	Tipo	Dimensión	Indicador	Ítem del instrumento	Técnica
***Framework* de *Data Wrangling* automatizado con Python**	**Independiente**	**Automatización del proceso ETL**	Existencia de scripts de extracción, limpieza y transformación de datos	P7: ¿Existe actualmente algún proceso automatizado para consolidar los archivos de las sucursales?	Entrevista
			Tiempo de procesamiento automatizado versus manual	EDA: tiempo de ejecución del pipeline versus tiempo reportado por el personal	Observación documental
		Estandarización de datos	Aplicación de reglas de nomenclatura unificada en nombres de productos	EDA: porcentaje de registros con nombres de productos inconsistentes antes y después del *framework*	Observación documental
			Unificación de formatos de fecha y campos clave entre sucursales	EDA: porcentaje de registros con formatos de fecha variables antes y después del framework	Observación documental
		Integración con Power BI	Conectividad del modelo de datos limpio con la capa de visualización	P8: ¿Qué herramientas de visualización utiliza actualmente la empresa y con qué nivel de satisfacción funcionan?	Entrevista
Calidad de la analítica de ventas	**Dependiente**	**Integridad del dato**	Porcentaje de campos vacíos u obligatorios ausentes en los registros	EDA: conteo de valores nulos por campo y por archivo analizado	Observación documental
		Consistencia del dato	Porcentaje de registros con formatos heterogéneos en un mismo campo	EDA: varianza de formatos por columna analizada en el corpus documental	Observación documental
		Unicidad del dato	Porcentaje de registros duplicados sobre el total del dataset	EDA: ratio de duplicados detectados por archivo en el período 2022–2025	Observación documental

Fuente: elaboración propia.

Tabla 2.1. Operacionalización de las variables de investigación *(continuación)*

Variable	Tipo	Dimensión	Indicador	Ítem del instrumento	Técnica
		Exactitud del dato	Porcentaje de valores fuera de rango lógico o evidentemente erróneos	EDA: registros con precios negativos, fechas futuras o cantidades imposibles detectados por el script	Observación documental
		Confiabilidad de los reportes	Percepción del personal sobre la veracidad de los reportes actuales de ventas	P4: ¿Ha derivado algún error en los datos en una decisión incorrecta para la empresa?	Entrevista
		Eficiencia operativa	Horas-hombre dedicadas a la limpieza manual de datos por ciclo de reporte	P5: ¿Qué porcentaje del tiempo se consume en corregir errores antes de poder analizar la información?	Entrevista

Fuente: elaboración propia.

A partir de la operacionalización presentada, se construyen los dos instrumentos que se describen a continuación. Ambos se articulan de forma complementaria: la guía de entrevista recoge la dimensión perceptual y organizacional del problema, mientras que la ficha de diagnóstico lo cuantifica desde la evidencia técnica del archivo

El primer instrumento corresponde a la Guía de Entrevista Semiestructurada, aplicada formalmente al Administrador General de la Comercializadora ”Josephine”. El procesamiento analítico de sus respuestas permitió estructurar el diagnóstico situacional bajo cuatro bloques fundamentales:

Bloque A Proceso actual de gestión de datos. El administrador explicó que cada vendedor registra sus transacciones de manera local e independiente en cada una de las cinco sucursales. No existe un repositorio central ni un flujo automatizado que consolide la información en tiempo real; la extracción de datos se hace a pedido, manualmente, en Excel. Esto significa que la gerencia no cuenta con reportes periódicos, y la capacidad de análisis estratégico queda seriamente limitada.

Bloque B Identificación de errores e inconsistencias. La conversación dejó claras cuatro categorías de anomalías que aparecen una y otra vez en las extracciones: duplicados generados automáticamente por el sistema, registros de ”basura”que se cuelan sin control, formatos inconsistentes en fechas y tallas, y columnas enteras que no sirven para la gestión. El procesamiento manual de estos problemas ha tenido consecuencias concretas: reportes con cifras infladas que llevaron a la empresa a comprar stock de baja rotación; esto afectó directamente la eficiencia del capital de trabajo.

Bloque C — Impacto operativo. La administración calcula que el personal puede pasar días enteros, e incluso semanas, solo en tareas repetitivas de depuración, consolidación y unificación de formatos. El volumen de registros es tal que la limpieza manual se convierte en un sumidero de tiempo. El resultado es una desconfianza generalizada hacia los totales de ventas y stock; los ciclos de decisión se ralentizan porque nadie se atreve a trabajar con cifras que saben que pueden estar distorsionadas.

Bloque D Requerimientos del sistema propuesto. Quedó claro que no existe ninguna herramienta de automatización previa, ni scripts ni macros. Con ese punto de partida, se definieron las pautas para el diseño del *framework* de *Business Intelli-*

gence. Hay una condición innegociable: la confidencialidad. Los datos comerciales no pueden exponerse en plataformas externas. El tablero deberá girar en torno a tres ejes: el índice de rotación de las prendas clave, el monitoreo de transacciones que se venden por debajo del costo unitario, y el seguimiento de utilidades netas desglosadas por sucursal y período.

El segundo instrumento fue la Ficha de Diagnóstico de Calidad de Datos. Su aplicación se hizo de forma automatizada mediante un script en Python con la librería Pandas, trabajando directamente sobre el archivo histórico de la empresa: 73.981 registros de transacciones comerciales, correspondientes al período comprendido entre enero de 2022 y mayo de 2025.

Tabla 2.2. Métricas del corpus documental.

Métrica del corpus documental	Valor	% universo inicial	Observación
Universo documental inicial (registros crudos)	**73.981**	100,00 %	Período: enero 2022 – mayo 2025 | 5 sucursales
Universo post-limpieza (dataset maestro)	72.795	98,40 %	Registros íntegros disponibles para analítica
Total de anomalías detectadas y gestionadas	11.202	15,14 %	Suma de las cuatro categorías de anomalía (L1+L2+L4+L5)
L1 — Duplicados eliminados (Unicidad)	1.186	1,60 %	Registros exactamente duplicados; eliminados mediante df.drop_duplicates()
L2 — Costos en cero imputados (Exactitud)	35	0,05 %	Costo_Unitario = 0; imputación matemática por mediana de referencia del período
L4 — Ventas bajo costo marcadas (Consistencia)	2.409	3,26 %	Precio_Venta < Costo_Unitario; flag de auditoría habilitado en el tablero Power BI
L5—Sin talla registradas (Completitud)	7.572	10,24 %	Campo Talla ausente o con valor 'S/T'; registros conservados con Flag_SinTalla = 1

Fuente: elaboración propia.

La distribución por punto de venta permite identificar las fuentes de mayor concentración de errores dentro del corpus. Los valores de los pasos L1, L2 y L4 se estimaron de forma proporcional al volumen de cada sucursal; los valores del paso L5 (Sin Talla) corresponden a los conteos exactos arrojados por el script EDA.

Tabla 2.3. Distribución de registros y anomalías por sucursal.

Sucursal	Registros totales	% corpus	L1 Duplicados	L2 Costo cero	L4 Ventas bajo costo	L5 Sin Talla	Total anomalías
MALL DE LOS ANDES	26.120	35,3 %	419	12	434	1122	**1987**
RIOBAMBA	19.542	26,4 %	313	9	313	846	**1481**
MATRIZ	11.478	15,5 %	184	5	1445	5151	**6785**
SHOPPING	9.039	12,2 %	145	4	120	269	**538**
SUCRE	7.802	10,5 %	125	4	96	184	**409**
TOTAL CORPUS	**73.981**	**100,0 %**	**1.186**	**34***	**2.408***	**7.572**	**11.200***

Fuente: elaboración propia.

Se detalla a continuación, para cada dimensión de calidad evaluada según el marco de referencia de Hassenstein y Vanella (2022), el hallazgo detectado, la función de Python/Pandas ejecutada, el volumen de registros afectados y el tratamiento aplicado por el framework de Data Wrangling.

Tabla 2.4. Dimensiones de calidad, hallazgos y tratamiento aplicado.

Dimensión (Hassenstein y Vanella, 2022)	Paso ETL	Descripción del hallazgo detectado	Función Pandas ejecutada	Registros afectados	% universo inicial	Tratamiento aplicado en el framework	Estado en el dataset maestro
Unicidad	**L1**	Registros exactamente duplicados generados por el sistema de punto de venta. Todas las columnas son idénticas entre el registro original y su copia.	df.duplicated(keep='first').sum() df.drop_duplicates(keep='first', inplace=True)	**1.186**	1,60 %	Eliminación definitiva. Se conserva la primera ocurrencia de cada registro y se descarta la copia exacta.	0 duplicados exactos en el dataset maestro (72.795 registros).
Exactitud	**L2**	Registros con Costo_Unitario igual a cero. Valor imposible desde la lógica de negocio: toda prenda comercializada presenta un costo de adquisición real.	df[df['Costo_Unitario'] ==0] df['Costo_Unitario'] .fillna(mediana_ref)	**35**	0,05 %	Imputación matemática mediante la mediana del costo del mismo producto en el período analizado. Se registra el campo 'Imputado' como flag de auditoría.	0 registros con Costo_Unitario = 0. Campo corregido y auditado.
Consistencia	**L4**	Registros donde el Precio_Venta es inferior al Costo_Unitario, generando utilidad negativa. Concentración severa identificada en la sucursal MATRIZ.	df[df['Precio_Venta'] < df['Costo_Unitario']] df['Flag_Perdida'] = np.where(cond,1,0)	**2.409**	3,26 %	Los registros se conservan en el dataset maestro. Se añade el campo binario Flag_Perdida = 1 para habilitar el monitoreo de ventas bajo costo en el tablero Power BI.	2.409 registros auditados y marcados. Visibles como alerta en el dashboard de Power BI.

Fuente: elaboración propia.

Tabla 2.4. Dimensiones de calidad, hallazgos y tratamiento aplicado *(continuación)*

Dimensión (Hassenstein y Vanella, 2022)	Paso ETL	Descripción del hallazgo detectado	Función Pandas ejecutada	Registros afectados	% universo inicial	Tratamiento aplicado en el framework	Estado en el dataset maestro
Completitud	**L5**	Registros donde el campo Talla está ausente (NaN) o contiene el valor centinela 'S/T' (Sin Talla). La sucursal MATRIZ concentra el 68,0 % de esta anomalía (5.151 registros).	df['Talla'].isna() .sum() (df['Talla']=='S/T') .sum() df['Flag_SinTalla']= cond.astype(int)	**7.572**	10,24 %	Los registros se preservan en el dataset maestro con Flag_SinTalla = 1. No se eliminan, pues representan transacciones válidas de productos sin talla aplicable o errores de ingreso.	7.572 registros marcados. Disponibles para análisis diferenciado de líneas de producto sin talla.
TOTAL	**L1–L5**	Cuatro categorías de anomalía detectadas y gestionadas mediante el framework de Data Wrangling.	—	**11.202**	**15,14 %**	Pipeline ETL completo ejecutado en Python/Pandas sobre el corpus 2022–2025.	**Dataset maestro: 72.795 registros íntegros.**

Fuente: elaboración propia.

2.3. Metodología de desarrollo del sistema

Para la construcción del *framework* de *Data Wrangling* y el posterior diseño del tablero de Inteligencia de Negocios, se seleccionó la metodología CRISP-DM (*Cross-Industry Standard Process for Data Mining*). Esta metodología se articula de manera complementaria con el enfoque ágil Kanban descrito en el apartado 2.1: mientras Kanban organizó la gestión visual y el flujo de tareas del equipo de desarrollo, CRISP-DM estructuró las fases técnicas del ciclo de vida del proyecto de minería de datos, desde la comprensión del negocio hasta el despliegue de la solución. En el ámbito de la Ingeniería en Sistemas y la ciencia de datos, este marco de trabajo iterativo es el estándar predominante, permite estructurar proyectos analíticos con una alineación estricta entre los requerimientos técnicos y los objetivos estratégicos del negocio.

La elección de CRISP-DM se justifica por su capacidad para gestionar ciclos de vida de datos complejos; de esta forma se garantiza que el proceso de extracción, transformación y carga (ETL) automatizado mediante secuencias de código responda de manera directa a las anomalías previamente identificadas en los repositorios de la Comercializadora ”Josephine”.

A continuación, se detalla la operacionalización de las seis fases de la metodología adaptadas al contexto de la presente investigación:

Fase 1: Comprensión del Negocio *Business Understanding*). Esta fase inicial tuvo como objetivo comprender los objetivos y requisitos del proyecto desde una perspectiva administrativa. Se ejecutó mediante la aplicación de la Guía de Entrevista Semiestructurada al Administrador General, lo que permitió identificar las limitaciones del proceso manual previo (reactividad, desconfianza en los reportes e ineficiencia operativa). En esta etapa se definieron los requerimientos clave del sistema: confidencialidad interna, análisis de rotación, identificación de ventas bajo costo y cálculo de utilidades por sucursal.

Fase 2: Comprensión de los Datos *Data Understanding*). En esta fase se recolectó la información inicial y se procedió a evaluar su estado cualitativo y cuantitativo. Se implementó un Análisis Exploratorio de Datos (EDA) en el entorno interactivo Jupyter Notebook. Mediante el uso de la librería Pandas en lenguaje Python, se analizaron los 73.981 históricos correspondientes al período 2022-2025. Esta ex-

ploración permitió poblar la Ficha de Diagnóstico de Calidad de Datos; este análisis cuantificó matemáticamente los problemas críticos de unicidad, exactitud, consistencia, integridad y completitud.

Fase 3: Preparación de los Datos*Data Preparation*). Constituye el núcleo técnico de la investigación. Con base en las 11.202 anomalías detectadas en la fase anterior, se diseñó e implementó el framework automatizado de Data Wrangling. Se codificaron reglas de limpieza específicas (imputación matemática de costos en cero, eliminación de registros duplicados mediante funciones de descarte y estandarización de campos nulos). El procesamiento computacional de estas reglas transformó el universo de datos crudos en un dataset maestro estructurado de 72.795 registros íntegros; así se consolidó la capa de transformación del pipeline ETL.

Fase 4: Modelado *Modeling*). Una vez garantizada la calidad del cien por ciento del dataset resultante, la fase de modelado consistió en la importación de los datos procesados a la herramienta Power BI. En esta etapa se construyó la arquitectura relacional de los datos (esquema de modelado dimensional) y se programaron las medidas analíticas necesarias mediante el lenguaje DAX (Data Analysis Expressions). El modelo se estructuró para soportar los cálculos de indicadores de rendimiento (KPIs) exigidos por la gerencia; de esta forma se asegura un rendimiento óptimo en las consultas visuales.

Fase 5: Evaluación. Previo a la entrega final, el sistema desarrollado fue sometido a un proceso de validación para certificar que el tablero de control respondiera efectivamente a los requerimientos organizacionales del Bloque D del diagnóstico inicial. Se verificó que las métricas de rentabilidad y las banderas de auditoría por pérdidas (ventas bajo costo) coincidieran con la lógica de negocio de la comercializadora; este proceso validó la fiabilidad de la información presentada.

Fase 6: Despliegue *Deployment*). La fase conclusiva comprendió la implementación técnica de la solución. El pipeline de código en Python fue documentado para futuras ejecuciones, y el tablero interactivo de Power BI fue configurado para su operación interna y segura por parte de la administración. Con esto, se consolidó la transición de un modelo de gestión de datos manual y propenso a errores, a un ecosistema analítico automatizado, fiable y orientado a la toma de decisiones estratégicas.

CAPÍTULO III. ANÁLISIS DE LOS RESULTADOS DE LA INVESTIGACIÓN

El presente capítulo expone, de manera integrada y secuencial, los resultados obtenidos durante la implementación del *framework* de *Data Wrangling* y el diseño del tablero de inteligencia de negocios para la Comercializadora ”Josephine”. La exposición se estructura conforme a las cuatro dimensiones centrales del proyecto: la arquitectura tecnológica e implementación del pipeline de limpieza en Python, el modelado dimensional y la integración con la plataforma de visualización Power BI, la evaluación comparativa del desempeño del sistema automatizado frente al proceso manual previo y, finalmente, la descripción funcional de los tableros de control desarrollados. Cada sección articula los datos cuantitativos obtenidos del Análisis Exploratorio de Datos (EDA) con las decisiones de diseño que los originaron; esta articulación demuestra la correspondencia directa entre el diagnóstico de calidad de los datos y la solución técnica implementada.

3.1. Arquitectura Tecnológica e Implementación del Framework de Data Wrangling

La implementación del framework de Data Wrangling se llevó a cabo en el entorno interactivo Jupyter Notebook, con el lenguaje de programación Python en su versión 3.x y la librería Pandas como motor principal de procesamiento. La selección de este entorno respondió a criterios de accesibilidad técnica y reproducibilidad; la solución puede ser ejecutada y auditada sin dependencia de infraestructura especializada, lo cual constituye una ventaja directamente aplicable al contexto de las pequeñas y medianas empresas (McGregor, 2021).

El código fuente completo del framework se presenta en el Anexo 3. El pipeline de procesamiento se articuló en seis etapas secuenciales y verificables, estructuradas conforme a la Fase 3 de la metodología CRISP-DM (Preparación de los Datos). Cada etapa produjo un *checkpoint* intermedio almacenado en formato Parquet, mecanismo que permitió conservar el estado del proceso ante cualquier interrupción y optimizar el consumo de memoria RAM durante el procesamiento del corpus histórico de 73.981 registros correspondiente al período enero de 2022 a mayo de 2025.

Etapa 1. Ingestión y consolidación del corpus documental: Se ejecutó la lectura iterativa de los archivos Excel generados por las cinco sucursales (Mall de los Andes, Riobamba, Matriz, Shopping y Sucre) mediante la función pd.read_excel() con parámetros de tipado explícito. Los archivos fueron concatenados en un único DataFrame maestro mediante pd.concat(); el resultado fue el universo documental inicial de 73.981 registros crudos (véase Anexo 3, Celda 2).

Etapa 2. Tratamiento de duplicados (L1 — Unicidad): Se aplicó la función df.duplicated(keep='first').sum() para la detección de registros exactamente idénticos en la totalidad de sus columnas. Se identificaron 1.186 registros duplicados, equivalentes al 1,60 % del universo inicial, generados sistemáticamente por el software de punto de venta. La eliminación definitiva se realizó mediante df.drop_duplicates(keep='first', inplace=True); se conservó la primera ocurrencia de cada transacción y se descartaron las copias redundantes (véase Anexo 3, Celda 5, corrección L1).

Etapa 3. Imputación de costos en cero (L2 — Exactitud): Mediante la expresión df[df['Costo_Unitario'] == 0] se identificaron 35 registros con Costo_Unitario igual a cero, un valor imposible desde la lógica de negocio de la comercializadora, dado que toda prenda registra un costo de adquisición real. La corrección se realizó mediante imputación matemática por la mediana del costo del mismo producto en el período analizado; se aplicó df['Costo_Unitario'].fillna(mediana_ref). Adicionalmente, se habilitó un campo de auditoría binario ('Imputado': 1/0) para garantizar la trazabilidad de las correcciones realizadas (véase Anexo 3, Celda 5, corrección L2).

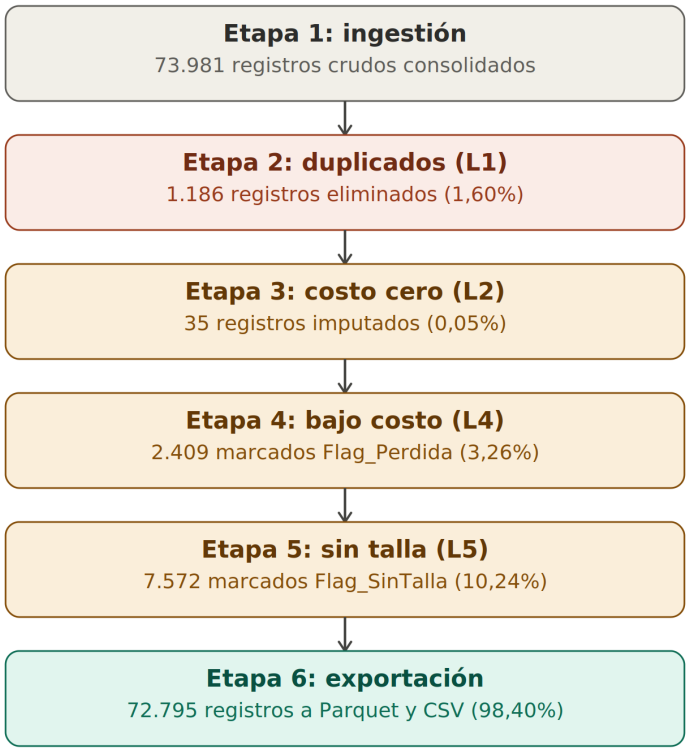
Etapa 4. Marcación de ventas bajo costo (L4 — Consistencia): Se implementó la regla de negocio df[df['Precio_Venta'] < df['Costo_Unitario']] para identificar transacciones en las que el precio de venta resultó inferior al costo unitario, lo que configura una pérdida operativa. Se detectaron 2.409 registros en esta condición, lo que representa el 3,26 % del corpus, con una concentración severa en la sucursal Matriz (1.445 registros, equivalentes al 59,98 % del total de esta anomalía). Los registros no fueron eliminados; en su lugar, se añadió el campo binario Flag_Perdida = 1 mediante np.where(); este campo habilita el monitoreo de dichas transacciones como alertas de auditoría en el tablero de Power BI (véase Anexo 3, Celda 5, corrección L4).

Etapa 5. Tratamiento de registros sin talla (L5 — Completitud): El campo Talla presentó 7.572 registros ausentes o con el valor centinela 'S/T' (Sin Talla), equivalen-

tes al 10,24 % del universo inicial. La concentración más crítica correspondió a la sucursal Matriz, con 5.151 registros afectados (68,0 % de esta anomalía). Los registros fueron preservados íntegramente en el *dataset* maestro y marcados con el campo Flag_SinTalla = 1, dado que representan transacciones válidas de líneas de producto que no aplican clasificación por talla o errores de ingreso que requieren seguimiento diferenciado (véase Anexo 3, Celda 1, función procesar_producto_seguro()).

Etapa 6. Exportación del *dataset* maestro: Una vez completadas las cuatro transformaciones precedentes, el *DataFrame* resultante fue exportado en formato Parquet como repositorio intermedio y, posteriormente, en formato CSV estructurado hacia la carpeta de almacenamiento institucional en OneDrive; este archivo constituye el punto de entrada para la capa de visualización en Power BI (véase Anexo 3, Celda 4 y Celda 5). El *dataset* maestro consolidado quedó compuesto por 72.795 registros íntegros, cifra que representa el 98,40 % del universo inicial.

Figura 3.1. Diagrama de las seis etapas del pipeline ETL.



Fuente: elaboración propia.

La siguiente tabla resume el mapa técnico completo del pipeline ETL implementado; para cada paso se detalla la anomalía tratada, la función de Python/Pandas ejecutada, el volumen de registros afectados y el tratamiento aplicado:

Tabla 3.1. Mapa técnico del pipeline ETL implementado en Python/Pandas.

Paso ETL	Dimensión (Hassenstein & Vanella, 2022)	Función Pandas ejecutada	Registros afectados	% universo	Tratamiento aplicado
L1 — Duplicados	Unicidad	df.drop_duplicates(keep='first')	1.186	1,60 %	Eliminación definitiva de copias exactas
L2 — Costo cero	Exactitud	df['Costo_Unitario'].fillna(mediana_ref)	35	0,05 %	Imputación por mediana del período; flag de auditoría activado
L4 — Ventas bajo costo	Consistencia	np.where(Precio_Venta < Costo_Unitario, 1, 0)	2.409	3,26 %	Conservación con Flag_Perdida = 1; visible como alerta en Power BI
L5 — Sin talla	Completitud	df['Talla'].isna().sum() + (df['Talla']=='S/T').sum()	7.572	10,24 %	Conservación con Flag_SinTalla = 1; análisis diferenciado habilitado
TOTAL ANOMALÍAS	L1 a L5	Pipeline ETL completo	11.202	15,14 %	Dataset maestro: 72.795 registros íntegros (98,40 %)

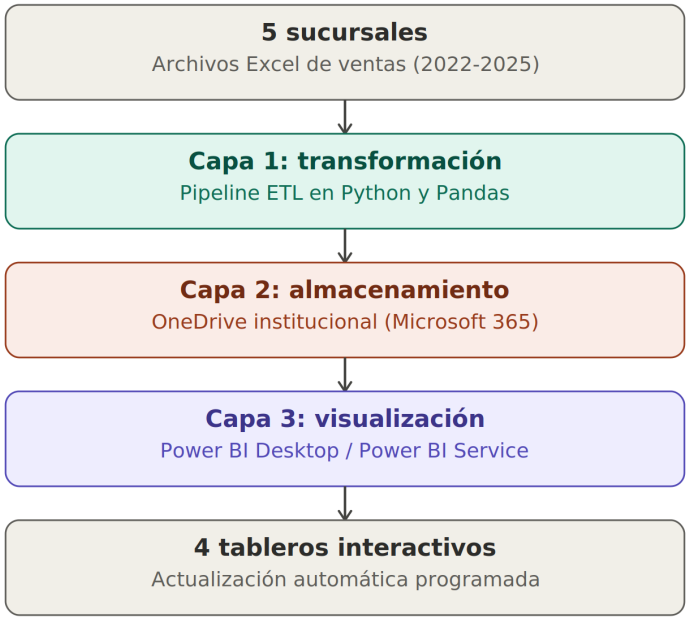
Fuente: elaboración propia.

La eficiencia computacional del pipeline resultó coherente con la naturaleza del entorno de ejecución: al tratarse de operaciones vectorizadas sobre *DataFrames* en memoria RAM, sin dependencia de GPU ni de motores de base de datos externos, el tiempo de procesamiento total del corpus histórico de 73.981 registros se situó en un rango de segundos, frente a los días o semanas que el mismo proceso demandaba de forma manual. Este atributo de eficiencia, asociado a la portabilidad del entorno Jupyter Notebook, posiciona al framework como una solución de alta accesibilidad para organizaciones sin infraestructura tecnológica especializada.

3.2. Modelado Dimensional e Integración de Inteligencia de Negocios

Una vez consolidado el dataset maestro de 72.795 registros íntegros, se procedió a la segunda capa de la arquitectura propuesta: la integración del repositorio de datos procesados con la plataforma de inteligencia de negocios Microsoft Power BI. Esta integración se articuló en torno a tres componentes funcionales: el repositorio de datos en la nube corporativa (OneDrive institucional), el modelo dimensional construido en Power BI Desktop y las medidas analíticas programadas en lenguaje DAX (Data Analysis Expressions).

Figura 3.2. Arquitectura desacoplada de tres capas: Python, OneDrive y Power BI.



Fuente: elaboración propia.

La elección de OneDrive institucional como capa de almacenamiento intermedio obedeció a tres criterios técnicos fundamentales. En primer lugar, la eliminación de la deuda técnica asociada a la configuración y mantenimiento de servidores de bases de datos relacionales con licencias propietarias, los cuales representan una barrera de implementación para organizaciones del perfil de la Comercializadora ”Josephine”. En segundo lugar, la disponibilidad multiplataforma que garantiza el acceso al repositorio de datos desde cualquier punto autorizado de la organización, sin restricciones de ubicación física. En tercer lugar, la capacidad nativa de actualización automática ofrecida por el conector de datos en la nube entre OneDrive y Power BI, que aseguró la propagación inmediata de cualquier actualización del pipeline hacia los tableros de visualización sin necesidad de intervención manual adicional (Rajpoot, Tiwari y Vishwakarma, 2026).

El modelo dimensional adoptado correspondió a un esquema en estrella (star schema), estructura estándar en el modelado de datos para inteligencia de negocios por su eficiencia en el procesamiento de consultas analíticas. El modelo se compuso de una tabla de hechos central (Fact_Ventas) y cuatro tablas de dimensión periféricas, según se describe a continuación:

Tabla de hechos Fact_Ventas: Contiene las métricas cuantificables de cada transacción comercial: unidades vendidas, precio de venta, costo unitario, utilidad bruta, y los campos de auditoría Flag_Perdida y Flag_SinTalla. Cada registro corresponde a una transacción individual del período 2022–2025, con granularidad de línea de venta.

Dimensión Producto (Dim_Producto): Almacena los atributos descriptivos estandarizados de cada prenda: código de producto, descripción, talla, categoría y subcategoría. La estandarización de nombres de producto, ejecutada durante el pipeline ETL mediante mapeos de equivalencia, eliminó las inconsistencias de nomenclatura identificadas en la fase de diagnóstico.

Dimensión Tiempo (Dim_Tiempo): Tabla de calendario generada en Power BI mediante DAX, que desglosa cada fecha de transacción en atributos temporales jerárquicos: año, semestre, trimestre, mes, semana y día. Esta dimensión habilitó el análisis de estacionalidad y tendencia en las series de ventas del período analizado.

Dimensión Sucursal (Dim_Sucursal): Registra los atributos identificativos de cada punto de venta: código de sucursal, denominación y ubicación geográfica. Las cinco sucursales de la organización (Mall de los Andes, Riobamba, Matriz, Shopping y Sucre) quedaron representadas como registros individuales de esta dimensión.

Dimensión Auditoría (Dim_Auditoria): Tabla auxiliar que consolida los registros marcados durante el proceso ETL (Flag_Perdida = 1 y Flag_SinTalla = 1); esto permite su filtrado y análisis independiente en los tableros de control sin afectar la integridad del dataset maestro.

Las relaciones entre la tabla de hechos y las dimensiones se establecieron mediante claves primarias y foráneas con cardinalidad uno a muchos (*:1); esta estructura configura el esquema en estrella que permite a Power BI ejecutar consultas analíticas multidimensionales con máxima eficiencia. Las medidas DAX implementadas incluyeron el cálculo de utilidad bruta (Utilidad = Precio_Venta - Costo_Unitario), el margen de rentabilidad porcentual por sucursal y período, el índice de rotación de prendas y el indicador de ventas bajo costo acumuladas por punto de venta.

La configuración de la actualización programada en Power BI Service garantizó que los tableros reflejaran de manera permanente el estado más reciente del repositorio en OneDrive, sin requerir intervención manual del equipo administrativo. Esta

característica eliminó la brecha informativa que existía previamente, cuando los reportes dependían de procesos manuales reactivos y carecían de una periodicidad establecida.

3.3. Evaluación del Desempeño del Framework

La evaluación del desempeño del framework de Data Wrangling se realizó mediante la comparación directa entre el proceso manual previo, documentado a través de la Guía de Entrevista Semiestructurada aplicada al Administrador General, y el proceso automatizado implementado. Los parámetros de comparación abarcaron cinco dimensiones operativas: tiempo de preparación de datos, tasa de detección de anomalías, confiabilidad de los resultados, periodicidad de los reportes y escalabilidad del proceso.

Los resultados de la evaluación comparativa entre el proceso manual previo y el framework automatizado se presentan en la Tabla 3.2, organizada en torno a nueve dimensiones operativas. Cada dimensión refleja una brecha identificada en el diagnóstico inicial y cuantificada durante la implementación:

Tabla 3.2. Evaluación comparativa del proceso manual previo versus el framework automatizado de Data Wrangling.

Dimensión evaluada	Proceso manual previo	Framework automatizado	Mejora obtenida
Tiempo de preparación de datos por ciclo de reporte	Días o semanas completas (estimación del administrador: ciclos de depuración que impedían el análisis estratégico)	Segundos a minutos (ejecución vectorizada del pipeline ETL sobre 73.981 registros)	Reducción drástica del tiempo operativo; el personal se reorienta hacia el análisis
Tasa de detección de anomalías	Parcial e inconsistente: dependiente de la atención del operador manual; duplicados y basura del sistema no siempre detectados	100 % del corpus procesado sistemáticamente: 11.202 anomalías detectadas (15,14 % del universo inicial)	Detección exhaustiva y reproducible; eliminación del sesgo humano en la identificación de errores
Unicidad del dato (duplicados)	No controlada: reportes inflados por registros duplicados reconocidos por el administrador	1.186 duplicados eliminados (1,60 %); dataset maestro libre de duplicados exactos	Integridad total en la dimensión de unicidad
Exactitud del dato (costos en cero)	No detectada sistemáticamente: valores imposibles permanecían en los reportes finales	35 registros imputados con mediana de referencia (0,05 %); campo de auditoría activado	Corrección auditada y trazable de valores imposibles
Consistencia del dato (ventas bajo costo)	No identificada: decisiones de compra basadas en reportes que incluían pérdidas operativas no visibles	2.409 registros marcados con Flag_Perdida = 1 (3,26 %); visibles como alertas en Power BI	Monitoreo preventivo de transacciones con pérdida; soporte a la auditoría interna
Completitud del dato (registros sin talla)	No gestionada: campos vacíos acumulados sin tratamiento diferenciado	7.572 registros marcados con Flag_SinTalla = 1 (10,24 %); disponibles para análisis diferenciado	Preservación de transacciones válidas con trazabilidad completa
Confiabilidad de los reportes	Baja: la gerencia cuestionaba la veracidad de los totales antes de tomar decisiones; se detectaron discrepancias post-presentación	Alta: dataset maestro de 72.795 registros validados; métricas coherentes con la lógica de negocio verificada	Restauración de la confianza institucional en la información analítica
Periodicidad de los reportes	Reactiva: reportes generados únicamente ante necesidades puntuales; sin ciclo regular establecido	Continua: actualización automática programada entre OneDrive y Power BI; reportes disponibles en tiempo real	Transición de la reportería reactiva a la reportería proactiva y continua
Escalabilidad del proceso	Nula: el tiempo de procesamiento crece linealmente con el volumen; el proceso colapsa ante nuevas sucursales o períodos adicionales	Alta: el pipeline ETL vectorizado en Pandas procesa volúmenes adicionales sin degradación proporcional del tiempo	Arquitectura escalable para el crecimiento organizacional de la empresa

Fuente: elaboración propia.

Los resultados de la evaluación confirman que el framework automatizado resolvió de manera integral las deficiencias identificadas en el diagnóstico inicial. La sustitución del proceso manual por un pipeline ETL codificado en Python/Pandas no solo incrementó la velocidad y la exhaustividad del procesamiento, sino que transformó cualitativamente la naturaleza de la información disponible para la gerencia: de un conjunto de datos parcialmente depurado, de confiabilidad cuestionada y de periodicidad reactiva, a un dataset maestro íntegro, auditable, de actualización continua y directamente conectado con la capa de visualización estratégica.

La concentración de anomalías detectadas en la sucursal Matriz (6.785 anomalías del total de 11.202, equivalentes al 60,57 %) constituyó un hallazgo organizacional de relevancia. Este patrón sugiere una problemática operativa específica en dicho punto de venta, relacionada con el volumen mixto de comprobantes que procesa, la cual fue trasladada como recomendación de auditoría interna a la gerencia de la Comercializadora.

3.4. Tableros de Control y Visualización Estratégica

El componente final de la solución implementada correspondió al diseño y configuración de los tableros de control interactivos en Microsoft Power BI. Los dashboards desarrollados respondieron directamente a los tres ejes prioritarios de visualización identificados en el Bloque D del diagnóstico organizacional: el análisis de la rotación de prendas clave, el monitoreo de ventas bajo costo y el seguimiento de utilidades netas por sucursal y período. El conjunto de tableros conformó un ecosistema analítico integrado que permite a la gerencia de la Comercializadora ”Josephine”tomar decisiones estratégicas basadas en información íntegra, oportuna y visualmente accesible.

La arquitectura de visualización se estructuró en cuatro tableros interactivos, diferenciados por nivel analítico y audiencia objetivo. Cada tablero responde directamente a uno de los ejes estratégicos identificados en el diagnóstico organizacional y se articula con el dataset maestro procesado por el framework de Data Wrangling:

Tablero 1 — Resumen Ejecutivo de Ventas: Este tablero proveyó una vista panorámica del desempeño comercial de la organización para el período 2022–2025. Incorporó tarjetas de indicadores clave (KPI cards) con el total de unidades vendi-

das, los ingresos brutos acumulados, la utilidad total y el número de transacciones procesadas del dataset maestro. La segmentación por año y por sucursal permitió a la gerencia general identificar tendencias de crecimiento o declive con un nivel de granularidad configurable mediante filtros interactivos. Un gráfico de líneas temporal exhibió la evolución mensual de las ventas y evidenció los patrones de estacionalidad del mercado de ropa casual.

Tablero 1. Resumen Ejecutivo de Ventas (vista general de KPIs y tendencia temporal). El tablero consolida los indicadores clave del período 2022–2025 con filtros interactivos por año y sucursal.

Fuente: elaboración propia.

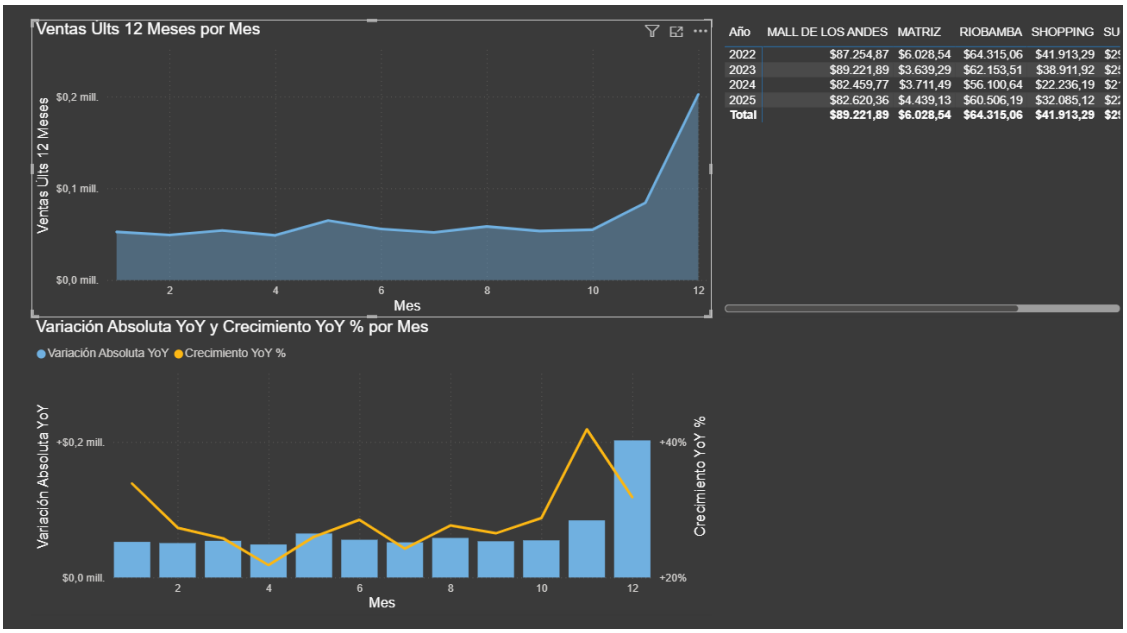
Tablero 2 — Análisis de Rentabilidad por Sucursal: Este panel presentó, de forma comparativa, el desempeño financiero de las cinco sucursales de la empresa. Mediante gráficos de barras apiladas y tablas matriciales con formato condicional, se visualizó la utilidad bruta generada por cada punto de venta en forma mensual, trimestral y anual. El modelo DAX permitió calcular el margen de rentabilidad porcentual para cada sucursal y período; este cálculo identificó los establecimientos con mayor y menor contribución al resultado financiero consolidado. La integración del campo Flag_Perdida habilitó la superposición visual de alertas de ventas bajo costo directamente sobre los gráficos de rentabilidad; esta función facilita la identificación inmediata de los períodos con mayor concentración de pérdidas operativas.

Tablero 2. Análisis de Rentabilidad por Sucursal (margen por punto de venta y alertas de pérdida). Presenta la utilidad bruta comparativa por sucursal con superposición de alertas Flag_Perdida.

Fuente: elaboración propia.

Tablero 3 — Monitoreo de Ventas Bajo Costo (Auditoría): Este tablero constituyó la respuesta directa al requerimiento de monitoreo preventivo de transacciones donde el precio de venta no cubrió el costo unitario. Los 2.409 registros marcados con Flag_Perdida = 1 fueron visualizados mediante un gráfico de dispersión que relacionó el precio de venta con el costo unitario para cada transacción auditada, con codificación de color por sucursal. Una tabla de detalle exportable permitió al equipo administrativo identificar los productos, fechas y sucursales con mayor recurrencia de ventas a pérdida; esta información orienta las acciones correctivas sobre las causas raíz de dicho comportamiento. La concentración detectada en la sucursal Matriz (1.445 registros, 59,98 % del total de esta anomalía) quedó evidenciada visualmente en este panel como un indicador prioritario de auditoría interna.

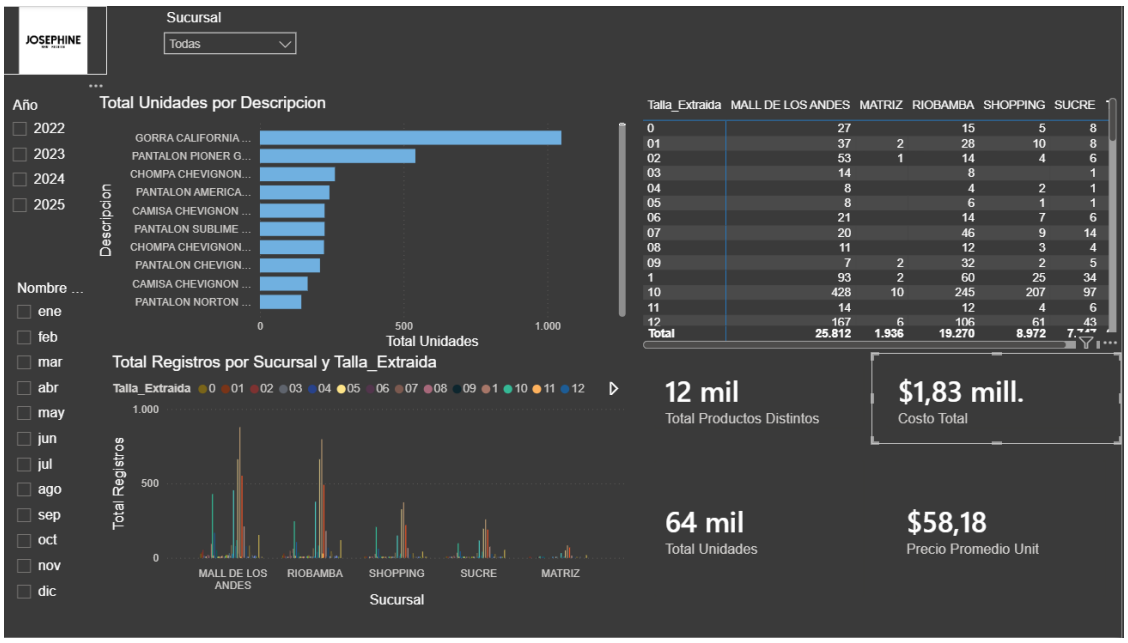
Tablero 3. Monitoreo de Ventas Bajo Costo (dispersión precio vs. costo y detalle por sucursal). Visualiza los 2.409 registros auditados con Flag_Perdida = 1 mediante gráfico de dispersión y tabla exportable.



Fuente: elaboración propia.

Tablero 4 — Rotación de Prendas y Análisis de Producto: Este panel analítico respondió al tercer eje estratégico del requerimiento gerencial: la identificación de los productos de mayor y menor rotación en el catálogo de la comercializadora. Mediante un gráfico de barras horizontales ordenado de forma descendente, se visualizaron las prendas con mayor volumen de unidades vendidas en el período, segmentadas por categoría de producto. Un mapa de calor por talla y por sucursal permitió identificar los segmentos con mayor demanda; esta información orienta las decisiones de reabastecimiento y la estrategia de compras a proveedores. Los 7.572 registros sin talla, marcados con Flag_SinTalla = 1, fueron presentados en una sección diferenciada del mismo tablero; este agrupamiento permite cuantificar el impacto de dicha anomalía sobre los totales por categoría y orienta la corrección en el proceso de ingreso de datos en los puntos de venta.

Tablero 4. Rotación de Prendas y Análisis de Producto (ranking de rotación, mapa de calor por talla y análisis de registros sin talla). Identifica los productos de mayor y menor rotación e incluye el análisis diferenciado de los 7.572 registros sin talla.



Fuente: elaboración propia.

En conjunto, los cuatro tableros de control implementados materializaron el objetivo central de la investigación: transformar el corpus de 73.981 registros crudos, heterogéneos y parcialmente erróneos generados por las sucursales de la Comercializadora ”Josephine”durante el período 2022–2025, en un activo de información estratégico, visualmente accesible y técnicamente confiable. La arquitectura desacoplada OneDrive-Python-Power BI aseguró que este activo sea dinámico: cualquier nueva extracción de datos procesada por el framework de Data Wrangling se propaga automáticamente hacia los tableros sin intervención manual; el resultado es un ecosistema de inteligencia de negocios continuo, auditable y orientado a la toma de decisiones estratégicas en tiempo real.

La implementación descrita en el presente capítulo demostró que la automatización del procesamiento de datos mediante un framework de Data Wrangling en Python no solo resuelve un problema técnico de calidad de datos, sino que actúa como catalizador de una transformación organizacional más profunda: la transición de una cultura de gestión reactiva e intuitiva hacia una cultura orientada a datos, en la que cada decisión de inversión, compra o evaluación de desempeño cuenta con el respaldo de una analítica robusta, íntegra y libre del sesgo introducido por el error humano en las fases de preparación manual de la información (Hassenstein y Vanella, 2022).

CONCLUSIONES

- La aplicación del marco de dimensiones de calidad de datos propuesto por Hassenstein y Vanella (2022) sobre el corpus histórico de 73.981 registros de la Comercializadora “Josephine” permitió no solo diagnosticar la problemática de datos crudos de la organización, sino también validar empíricamente la pertinencia de dicho marco teórico en un contexto de pequeña y mediana empresa del sector retail latinoamericano, escasamente documentado en la literatura revisada. El hallazgo de que el 15,14 % del corpus (11.202 registros) presentara algún tipo de inconsistencia, con una concentración desproporcionada en la sucursal Matriz, constituye evidencia empírica de que la problemática de calidad de datos en las PYMES no se distribuye de manera homogénea, sino que responde a factores operativos específicos de cada punto de venta; este hallazgo aporta un elemento de análisis organizacional que trasciende el diagnóstico técnico inicial y orienta la auditoría interna hacia causas concretas.

- Desde la perspectiva tecnológica, la investigación demuestra que es posible automatizar de manera íntegra un proceso de Data Wrangling mediante un pipeline ETL de seis etapas construido exclusivamente con herramientas de código abierto —Python y Pandas—, sin dependencia de infraestructura especializada como GPU, motores de bases de datos relacionales o licencias propietarias, alcanzando un dataset maestro con el 98,40 % de integridad. Este resultado aporta un modelo técnico replicable para organizaciones de recursos limitados, en el que la accesibilidad y la reproducibilidad del proceso no comprometen su rigor metodológico, y contribuye a cerrar la brecha documentada en la literatura entre las soluciones de inteligencia de negocios orientadas a grandes corporaciones y las necesidades reales de las PYMES.

- La arquitectura desacoplada de tres capas —transformación en Python, almacenamiento en OneDrive institucional y visualización en Power BI—, articulada mediante un modelo dimensional en esquema estrella, constituye un aporte metodológico al evidenciar que la integración de un pipeline de limpieza con una plataforma de inteligencia de negocios no requiere, necesariamente, de un motor de base de datos relacional tradicional. Validada funcionalmente en el caso de estudio, esta arquitectura ofrece una alternativa técnica documentada para el diseño de soluciones de analítica de ventas en organizaciones que carecen de infraestructura de servidores propia.

- La evaluación comparativa confirmó que la brecha limitante para la analítica de ventas en la Comercializadora “Josephine” no residía en la herramienta de visualización, sino en la ausencia de una capa de preparación de datos automatizada y confiable; al resolver esta brecha, la investigación evidencia que la automatización del proceso ETL constituye la condición habilitante —y no solo complementaria— para la adopción efectiva de la inteligencia de negocios en las PYMES. En correspondencia con el objetivo general planteado, el framework desarrollado transformó un proceso de gestión reactivo, propenso al error humano y de baja confiabilidad, en un ecosistema analítico auditable y de actualización continua, sentando las bases técnicas —mediante un repositorio histórico limpio y estandarizado— para una eventual evolución hacia modelos de analítica predictiva, conforme al requisito previo señalado por Hassenstein y Vanella (2022) para el desarrollo de analítica avanzada.

RECOMENDACIONES

- Se recomienda a la Comercializadora ”Josephine.[es]tandarizar el proceso de ingreso de datos en los cinco puntos de venta mediante la definición de un manual operativo de registro, que establezca campos obligatorios —especialmente el campo Talla, responsable del 10,24 % de las anomalías detectadas— y formatos únicos para fechas, códigos de producto y sucursal. Esta medida preventiva reduciría el volumen de anomalías en origen y disminuiría el tiempo de ejecución del pipeline ETL en futuras actualizaciones del dataset maestro.

- Se recomienda programar ejecuciones periódicas del framework de Data Wrangling con una frecuencia mínima mensual, de modo que el dataset maestro en OneDrive se mantenga actualizado y los tableros de Power BI reflejen el estado real de las operaciones comerciales. Dado que el pipeline procesa el corpus completo en segundos, el costo operativo de esta periodicidad es mínimo y representa una mejora directa sobre el esquema reactivo previo, en el que los reportes se generaban únicamente ante necesidades puntuales sin ciclo regular establecido.

- Se recomienda priorizar una auditoría interna sobre la sucursal Matriz, que concentró el 60,57 % del total de anomalías del corpus (6.785 de 11.202), entre las cuales se registran 1.445 ventas bajo costo equivalentes al 59,98 % de esa categoría. Esta revisión debe orientarse a identificar las causas operativas del alto volumen de transacciones con pérdida registradas en ese punto de venta y corregir los procedimientos de emisión de facturas que generan la mayor parte de los registros inconsistentes detectados por el framework.

- Se recomienda a futuras investigaciones en el área de sistemas de información y ciencia de datos extender el presente framework hacia una etapa predictiva. El repositorio histórico limpio de 72.795 registros íntegros del período 2022–2025 constituye la base para la construcción de modelos de proyección de demanda por talla, categoría de producto y sucursal. La consolidación de esta base de datos estandarizada cumple el requisito previo identificado por Hassenstein y Vanella (2022) para el desarrollo de analítica avanzada, lo que posiciona a la organización en condiciones técnicas para una evolución hacia inteligencia de negocios de carácter predictivo.

BIBLIOGRAFÍA

Alvarado-Apodaca, J., Ramírez-Noriega, A., Tripp-Barba, C., Martínez-Ramírez, Y., & Sánchez, I. N. Á. (2023). Inteligencia de negocios en América Latina: Una revisión sistemática de literatura. *Revista de Investigación en Tecnologías de la Información*, *11*(24), 76–89. https://doi.org/10.36825/RITI.11.24.007

Álvarez, M. B., & Romero, M. del C. (2025). Uso de datos e información para la toma de decisiones. Análisis de empresas pymes de la ciudad de Tandil por sector de actividad. *Pymes, Innovación y Desarrollo*, *13*(1), 61–82.

Anderson, D. J. (2010). *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press.

Corporación Universitaria Republicana, & Bermúdez Irreño, C. A. (2022). FACTORES QUE INFLUYEN EN EL ÉXITO DE LA IMPLEMENTACIÓN DE LA TRANSFORMACIÓN DIGITAL. *Revista Ingeniería, Matemáticas y Ciencias de la Información*, *9*(17), 69–75. https://doi.org/10.21017/rimci.2022.v9.n17.a112

Dasari, D., & Varma, P. S. (2022). Employing Various Data Cleaning Techniques to Achieve Better Data Quality using Python. *2022 6th International Conference on Electronics, Communication and Aerospace Technology*, 1379–1383. https://doi.org/10.1109/ICECA55336.2022.10009079

Delen, D. (2024). Landscape of Tools for Business Analytics and Data Science – A Tutorial on KNIME. *Proceedings of the 2024 Pre-ICIS SIGDSA Symposium*. Recuperado de https://aisel.aisnet.org/sigdsa2024/21

Delgado Díaz, N., Alejo Machado, O. J., & López Gutiérrez, J. C. (2025). Las herramientas de inteligencia de negocios potencian la capacidad de toma de decisiones en las PYMES. *Dilemas Contemporáneos: Educación, Política y Valores*, *12*(2), artículo 24. https://dilemascontemporaneoseducacionpoliticayvalores.com/index.php/dilemas/article/view/4545

Espinoza Tinoco, L. M., Congacha Aushay, A. E., & Díaz Ordóñez, J. C. (2023). Calidad de datos con Python: Un enfoque práctico. *Esprint Investigación*, *2*(2), 26–34. https://doi.org/10.61347/ei.v2i2.55

García-Jiménez, A. de-Jesús, Aguilar-Morales, N., Hernández-Triano, L., & Lancaster-Díaz, E. (2021). La inteligencia de negocios: Herramienta clave para el uso de la información y la toma de decisiones empresariales. *Revista de Investigaciones Universidad del Quindío*, *33*(1), 132–139. https://doi.org/10.33975/riuq.vol33n1.514

Gómez, D. del C. S., & Latournerle, N. H. B. (2022). Integridad de los datos. *Revista Iberoamericana de Tecnología Educativa*, *1*(2), 42–54.

Gómez-Duque, L. Á., Daza-Torres, J. D., & Arias-Pérez, J. (2023). Inteligencia de negocios y agilidad organizacional: ¿Son relevantes la toma de decisiones racional e intuitiva? *Estudios Gerenciales*, 181–191. https://doi.org/10.18046/j.estger.2023.167.5542

Hassenstein, M. J., & Vanella, P. (2022). Data Quality—Concepts and Problems. *Encyclopedia*, *2*(1), 498–510. https://doi.org/10.3390/encyclopedia2010032

Malla Valdiviezo, R. O., López Gorozabel, O. A., Arévalo Indio, J. A., & Tóala Briones, C. H. (2023). Mecanismos para el procesamiento de big data: Limpieza, transformación y análisis de datos. *Polo del Conocimiento*, *8*(4), 656–675. https://www.polodelconocimiento.com/ojs/index.php/es/article/view/5457

McGregor, S. E. (2021). *Practical Python Data Wrangling and Data Quality: Getting Started with Reading, Cleaning, and Analyzing Data*. O'Reilly Media, Inc.

Ordoñez Abril, D. Y., Amaya López, S. V., Lucio Valencia, L. P., & Rodríguez Gómez, D. (2023). Innovación en la inteligencia de negocios. Una revisión sistemática de literatura. *ECA Sinergia*, *14*(2), 148–164. https://doi.org/10.33936/ecasinergia.v14i2.5556

Rajpoot, C. S., Tiwari, V., & Vishwakarma, S. K. (2026). Emerging trends of recommender system for e-commerce: A comprehensive review. *Discover Computing*, *29*(1), 63. https://doi.org/10.1007/s10791-026-09923-z

Roberts, I., & Massoud, H. (2025). Sistemas de información contable y decisiones basadas en datos. *Entrelíneas*, *4*(1), e040104. https://doi.org/10.56368/Entrelineas414

Rodríguez Baque, J. C. (2023). *"LOS SISTEMAS DE INFORMACIÓN GERENCIAL Y SU CONTRIBUCIÓN EN LA TOMA DE DECISIONES DE LA MICROEMPRESA "PLASTIMETAL", PERIODO 2021-2022"* (bachelorThesis, Jipijapa-Unesum). Jipijapa-Unesum. Recuperado de http://repositorio.unesum.edu.ec/handle/53000/5267

Sánchez-Lunavictoria, J. C., Bolaños-Logroño, P. F., Mendosa-Mejía, G. M., & Villagómez-Vaca, G. N. (2026). Inteligencia de negocios y análisis de datos para la gobernanza y la democratización del desarrollo productivo en las MIPYMES del Ecuador. *Cuestiones Políticas*, *44*(84), 14–34. https://doi.org/10.5281/zenodo.18734984

Tamayo y Tamayo, M. (2014). *El proceso de la investigación científica* (5ª ed.). Limusa.

Unocc Antonio, A., & De La Cruz Llacctahuaman, M. (2024). *Sistema de información para optimizar el proceso de venta de prendas de vestir de Corporación Marusa E.I.R.L., Lircay-2024*. Recuperado de https://hdl.handle.net/20.500.14502/308

ANEXOS

ANEXO 1

Guía de Entrevista Semiestructurada — Aplicada

Tabla A1.1

Instrumento N.°	1 — Guía de Entrevista Semiestructurada
Informante clave	Administrador General — Comercializadora ”Josephine”
Objetivo	Diagnosticar el proceso actual de gestión de datos de ventas, identificar las principales inconsistencias y levantar los requerimientos funcionales del framework propuesto.
Técnica aplicada	Entrevista semiestructurada
Duración estimada	30 a 45 minutos
Modalidad	Presencial — grabada con consentimiento del informante
Entrevistador	Christian Fernando Paredes Tamayo
Fecha de aplicación	Mayo 2025

Fuente: elaboración propia

Introducción al Entrevistado

.[EI] objetivo de esta entrevista es comprender cómo se gestiona actualmente la información de ventas e inventarios en la empresa, identificar las principales dificultades operativas y conocer sus expectativas respecto a una solución automatizada. La información es estrictamente confidencial y de uso exclusivamente académico.”

BLOQUE A — Proceso actual de gestión de datos

Dimensión evaluada: Automatización del proceso ETL | Eficiencia operativa

Tabla A1.2

Pregunta 1. ¿Podría describir, paso a paso, cómo se reciben y procesan actualmente los reportes de ventas provenientes de las distintas sucursales? ¿Quién es el responsable de consolidarlos y en qué formato llegan?
Respuesta del informante: El administrador indicó que los datos de ventas se registran directamente en el sistema de punto de venta en el momento exacto de cada transacción comercial, en cada una de las cinco sucursales de la empresa. Los responsables del ingreso de la información son los vendedores asignados a cada local, quienes operan el sistema al momento de emitir la factura. Esta modalidad implica que no existe un proceso de consolidación periódica previa: los datos quedan almacenados en el sistema de cada sucursal de forma independiente y deben ser extraídos manualmente en formato Excel cuando se requiere generar un informe. No existe un flujo automatizado de centralización ni un repositorio único que integre la información de todas las sucursales en tiempo real.

Fuente: elaboración propia

Tabla A1.3

Pregunta 2. ¿Con qué frecuencia se elaboran reportes consolidados de ventas o inventarios? ¿Cuánto tiempo aproximado tarda el personal en tener el reporte listo desde que recibe los archivos de las sucursales?
Respuesta del informante: El administrador señaló que no existe una periodicidad establecida para la elaboración de reportes consolidados de ventas o inventarios. Los reportes se generan únicamente de forma reactiva, es decir, cuando surge una necesidad puntual, principalmente cuando se requiere evaluar el inventario disponible para tomar decisiones de reabastecimiento. Esta ausencia de un ciclo regular de reportería implica que la gerencia no cuenta con información actualizada de manera sistemática, lo que limita la capacidad de la empresa para realizar análisis de tendencias, planificación de compras o evaluación del desempeño por sucursal con base en datos oportunos.

Fuente: elaboración propia

BLOQUE B — Identificación de errores e inconsistencias

Dimensión evaluada: Integridad | Consistencia | Unicidad | Exactitud del dato

Tabla A1.4

Pregunta 3. ¿Cuáles son los tipos de errores o inconsistencias más frecuentes que se encuentran en los archivos Excel recibidos? Por ejemplo: nombres de productos escritos de forma diferente, fechas en distintos formatos, celdas vacías o registros repetidos.
Respuesta del informante: El administrador identificó cuatro categorías principales de anomalías recurrentes en los archivos extraídos del sistema: en primer lugar, la presencia de registros duplicados generados por el propio sistema de punto de venta, que replica transacciones sin que el operador lo detecte; en segundo lugar, datos catalogados como ”basura del sistema”, es decir, registros sin utilidad analítica que se incorporan automáticamente en las exportaciones; en tercer lugar, inconsistencias de formato en campos críticos como fechas de venta y tallas de prendas, que varían en su representación entre sucursales o incluso dentro de un mismo archivo; y, por último, la presencia de columnas o campos que no aportan información relevante para los informes de gestión y que incrementan el volumen de los archivos sin valor añadido.

Fuente: elaboración propia

Tabla A1.5

Pregunta 4. ¿Ha ocurrido alguna situación en que un error en los datos haya derivado en una decisión incorrecta, como una compra de inventario innecesaria, una discrepancia contable o un reporte que debió corregirse después de presentarse? ¿Podría describirla brevemente?
Respuesta del informante: El administrador confirmó que se presentaron consecuencias operativas derivadas de la deficiente calidad de los datos. Dado que el tratamiento de los archivos se realizaba de forma completamente manual, en múltiples ocasiones los responsables de la consolidación no lograban identificar y eliminar todos los duplicados ni depurar correctamente los registros de basura del sistema. Esta situación derivó en reportes de ventas e inventarios con cifras infladas, lo que condujo a decisiones de compra de inventario basadas en información incorrecta. Específicamente, se adquirieron unidades adicionales de prendas que, al comparar con la demanda real, no constituían productos de alta rotación, generando un exceso de stock que afectó la eficiencia del capital de trabajo de la empresa.

Fuente: elaboración propia

BLOQUE C — Impacto operativo

Dimensión evaluada: Eficiencia operativa | Confiabilidad de los reportes

Tabla A1.6

Pregunta 5. Del tiempo total que el personal dedica a la gestión de datos, ¿qué porcentaje aproximado estima que se consume en corregir errores, unificar formatos o limpiar información, en lugar de analizar los resultados?
Respuesta del informante: El administrador indicó que el proceso de limpieza y consolidación manual de datos resultaba extremadamente demandante en términos de tiempo. Debido al volumen de información generado por las cinco sucursales, la persona encargada de preparar los reportes podía llegar a invertir días enteros, e incluso semanas completas, únicamente en tareas de depuración, unificación de formatos y eliminación de registros irrelevantes, antes de poder comenzar con el análisis de los datos. Esto implica que la mayor parte del tiempo dedicado a la gestión de información se consumía en actividades operativas de bajo valor analítico, dejando un margen mínimo para el análisis estratégico de los resultados comerciales.

Fuente: elaboración propia

Tabla A1.7

Pregunta 6. ¿Con qué nivel de confianza la gerencia toma decisiones basándose en los reportes actuales de ventas? ¿Existe desconfianza hacia la información que entregan los sistemas o archivos actuales?
Respuesta del informante: El administrador reconoció que existe un nivel significativo de desconfianza hacia los reportes generados con el proceso actual. Esta desconfianza se fundamenta en experiencias concretas: en diversas ocasiones se han detectado, tras la presentación de informes, registros duplicados y datos de basura que distorsionaban las cifras reportadas. La gerencia ha llegado al punto de cuestionar la veracidad de los totales de ventas y los datos de inventario antes de tomar decisiones, lo que ralentiza el proceso de toma de decisiones y genera un clima de incertidumbre respecto a la información disponible. Esta situación refuerza la necesidad de un sistema que garantice la integridad y consistencia de los datos de origen.

Fuente: elaboración propia

BLOQUE D — Requerimientos del sistema propuesto

Dimensión evaluada: Automatización | Integración con Power BI

Tabla A1.8

Pregunta 7. ¿Existe actualmente algún proceso o herramienta que automatice, aunque sea parcialmente, la consolidación o limpieza de los archivos de las sucursales?
Respuesta del informante: El administrador confirmó que no existe ningún proceso automatizado, ni parcial ni total, para la consolidación o limpieza de los archivos de ventas provenientes de las sucursales. Todo el tratamiento de datos se realizaba de manera completamente manual, a través de operaciones directas en hojas de cálculo Excel, sin el apoyo de macros, scripts ni herramientas de automatización. Esta condición implica que el nivel de calidad del dato procesado dependía exclusivamente de la atención y habilidad de la persona encargada, lo que introduce una variabilidad significativa en la calidad de los resultados y hace que el proceso sea altamente susceptible a errores humanos.

Fuente: elaboración propia

Tabla A1.9

Pregunta 8. Si existiera una solución que consolidara y limpiara automáticamente todos los archivos de ventas, ¿qué indicadores considera prioritarios para visualizar en un tablero de control? ¿Existe alguna restricción técnica o de confidencialidad que deba considerarse en el diseño?
Respuesta del informante: El administrador estableció dos condiciones para el diseño del tablero de control. En primer lugar, respecto a la confidencialidad, indicó que la solución debe operar de manera interna dentro de la empresa, sin exponer la información comercial a plataformas o servicios externos no autorizados. En segundo lugar, respecto a los indicadores prioritarios, señaló que el tablero debe enfocarse en tres ejes principales: el análisis de ventas de prendas clave para identificar los productos de mayor rotación y contribución al ingreso; el monitoreo de ventas bajo costo para detectar transacciones donde el precio de venta no cubre el costo unitario; y el seguimiento de utilidades por sucursal y período, que permita evaluar la rentabilidad real de cada punto de venta. Estos indicadores responden directamente a las necesidades de decisión estratégica de la gerencia.

Fuente: elaboración propia

Cierre de la Entrevista

”Muchas gracias por su tiempo y colaboración. La información compartida es de gran valor para el desarrollo de esta investigación y contribuirá directamente al diseño del framework de automatización propuesto para la empresa.”

ANEXO 2

Ficha de Diagnóstico de Calidad de Datos

Análisis Exploratorio de Datos (*EDA*) — Comercializadora ”Josephine”

Instrumento de observación documental cuantitativa | Período 2022–2025 | Ejecutado con Python/Pandas en Jupyter Notebook

Tabla A2.1

Resumen ejecutivo del corpus documental y anomalías detectadas

Se consolidan a continuación los indicadores globales obtenidos mediante el Análisis Exploratorio de Datos (EDA) ejecutado sobre la totalidad del corpus documental histórico de la Comercializadora ”Josephine”, correspondiente al período comprendido entre enero de 2022 y mayo de 2025.

Métrica del corpus documental	Valor	% universo inicial	Observación
Universo documental inicial (registros crudos)	**73.981**	100,00 %	Período: enero 2022 – mayo 2025 | 5 sucursales
Universo post-limpieza (dataset maestro)	72.795	98,40 %	Registros íntegros disponibles para analítica
Total de anomalías detectadas y gestionadas	**11.202**	15,14 %	Suma de las cuatro categorías de anomalía (L1+L2+L4+L5)
L1 — Duplicados eliminados (Unicidad)	1.186	1,60 %	Registros exactamente duplicados; eliminados mediante df.drop_duplicates()
L2 — Costos en cero imputados (Exactitud)	35	0,05 %	Costo_Unitario = 0; imputación matemática por mediana de referencia del período
L4 — Ventas bajo costo marcadas (Consistencia)	2.409	3,26 %	Precio_Venta < Costo_Unitario; flag de auditoría habilitado en el tablero Power BI
L5 — Sin talla registradas (Completitud)	7.572	10,24 %	Campo Talla ausente o con valor 'S/T'; registros conservados con Flag_SinTalla = 1

Nota. Los pasos de limpieza L1–L5 corresponden a las etapas del pipeline de Data Wrangling implementado en la Fase 3 (Preparación de Datos) de la metodología CRISP-DM. El paso L3 corresponde a la estandarización de formatos (fechas, tallas y códigos de producto), sin eliminación de registros. Elaboración propia.

Fuente: elaboración propia

La distribución por punto de venta permite identificar las fuentes de mayor concentración de errores dentro del corpus. Los valores de los pasos L1, L2 y L4 se estimaron de forma proporcional al volumen de cada sucursal; los valores del paso L5 (Sin Talla) corresponden a los conteos exactos arrojados por el script EDA.

Tabla A2.2
Distribución de registros y anomalías por sucursal

Sucursal	Registros totales	% corpus	L1 Duplicados	L2 Costo cero	L4 Ventas bajo costo	L5 Sin Talla	Total anomalías
MALL DE LOS ANDES	26.120	35,3 %	419	12	434	1122	**1987**
RIOBAMBA	19.542	26,4 %	313	9	313	846	**1481**
MATRIZ	11.478	15,5 %	184	5	1445	5151	**6785**
SHOPPING	9.039	12,2 %	145	4	120	269	**538**
SUCRE	7.802	10,5 %	125	4	96	184	**409**
TOTAL CORPUS	**73.981**	**100,0 %**	**1.186**	**34***	**2.408***	**7.572**	**11.200***

Nota. (*) Las diferencias de ±1–2 unidades respecto a los totales globales confirmados (L2 = 35, L4 = 2.409, Total = 11.202) obedecen al redondeo en la distribución proporcional de L1, L2 y L4 entre sucursales. El dato exacto del total global es 11.202 anomalías sobre 73.981 registros crudos. Elaboración propia.
Fuente: elaboración propia.

Se detalla a continuación, para cada dimensión de calidad evaluada según el marco de referencia de Hassenstein y Vanella (2022), el hallazgo detectado, la función de Python/Pandas ejecutada, el volumen de registros afectados y el tratamiento aplicado por el framework de Data Wrangling.

Tabla A2.3

Ficha técnica por dimensión de calidad del dato

Dimensión (Hassenstein y Vanella, 2022)	Paso ETL	Descripción del hallazgo detectado	Función Pandas ejecutada	Registros afectados	% universo inicial	Tratamiento aplicado en el framework	Estado en el dataset maestro
Unicidad	**L1**	Registros exactamente duplicados generados por el sistema de punto de venta. Todas las columnas son idénticas entre el registro original y su copia.	df.duplicated(keep='first').sum() df.drop_duplicates(keep='first', inplace=True)	**1.186**	1,60 %	Eliminación definitiva. Se conserva la primera ocurrencia de cada registro y se descarta la copia exacta.	0 duplicados exactos en el dataset maestro (72.795 registros).
Exactitud	**L2**	Registros con Costo_Unitario igual a cero. Valor imposible desde la lógica de negocio: toda prenda comercializada presenta un costo de adquisición real.	df[df['Costo_Unitario'] ==0] df['Costo_Unitario'] .fillna(mediana_ref)	**35**	0,05 %	Imputación matemática mediante la mediana del costo del mismo producto en el período analizado. Se registra el campo 'Imputado' como flag de auditoría.	0 registros con Costo_Unitario = 0. Campo corregido y auditado.
Consistencia	**L4**	Registros donde el Precio_Venta es inferior al Costo_Unitario, generando utilidad negativa. Concentración severa identificada en la sucursal MATRIZ	df[df['Precio_Venta'] < df['Costo_Unitario']] df['Flag_Perdida'] = np.where(cond,1,0)	**2.409**	3,26 %	Los registros se conservan en el dataset maestro. Se añade el campo binario Flag_ Perdida = 1 para habilitar el monitoreo de ventas bajo costo en el tablero Power BI.	2.409 registros auditados y marcados. Visibles como alerta en el dashboard de Power BI.

Fuente: *Elaboración propia a partir del EDA automatizado ejecutado con la librería Pandas de Python sobre el corpus documental histórico de la Comercializadora "Josephine"(2022–2025). Marco de dimensiones de calidad: Hassenstein, M. J., y Vanella, P. (2022). Data Quality — Concepts and Problems. Encyclopedia, 2(1), 498–510.* https://doi.org/10.3390/encyclopedia2010032

Tabla A2.3. Ficha técnica por dimensión de calidad del dato *(continuación)*

Dimensión (Hassenstein y Vanella, 2022)	Paso ETL	Descripción del hallazgo detectado	Función Pandas ejecutada	Registros afectados	% universo inicial	Tratamiento aplicado en el framework	Estado en el dataset maestro
Completitud	**L5**	Registros donde el campo Talla está ausente (NaN) o contiene el valor centinela 'S/T' (Sin Talla). La sucursal MATRIZ concentra el 68,0 % de esta anomalía (5.151 registros).	df['Talla'].isna() .sum() (df['Talla']=='S/T') .sum() df['Flag_SinTalla']= cond.astype(int)	**7.572**	10,24 %	Los registros se preservan en el dataset maestro con Flag_SinTalla = 1. No se eliminan, pues representan transacciones válidas de productos sin talla aplicable o errores de ingreso.	7.572 registros marcados. Disponibles para análisis diferenciado de líneas de producto sin talla.
TOTAL	**L1–L5**	Cuatro categorías de anomalía detectadas y gestionadas mediante el framework de Data Wrangling.	—	**11.202**	**15,14 %**	Pipeline ETL completo ejecutado en Python/Pandas sobre el corpus 2022–2025.	**Dataset maestro: 72.795 registros íntegros.**

Fuente: *Elaboración propia a partir del EDA automatizado ejecutado con la librería Pandas de Python sobre el corpus documental histórico de la Comercializadora "Josephine"(2022–2025). Marco de dimensiones de calidad: Hassenstein, M. J., y Vanella, P. (2022). Data Quality — Concepts and Problems. Encyclopedia, 2(1), 498–510.* https://doi.org/10.3390/encyclopedia2010032

ANEXO 3

Código Fuente del Framework de Data Wrangling

A continuación se presenta el código fuente completo del pipeline de Data Wrangling, desarrollado en Python sobre Jupyter Notebook (Google Colab), organizado en las cinco celdas funcionales que lo componen. El código corresponde a la versión final ejecutada sobre el corpus histórico 2022-2025 de la Comercializadora “Josephine” y se incluye de forma íntegra para garantizar la reproducibilidad del proceso descrito en el capítulo III.

Celda 1

Configuración inicial y funciones de utilidad. Define las reglas de negocio (mapeo de sucursales y columnas), la función procesar_producto_seguro() que extrae el código base y la talla desde Codigo_Producto (regla rsplit + validación de longitud, base de la anomalía L5 — Sin Talla), y la función reporte_calidad() que documenta el estado del DataFrame tras cada etapa.

```
# ╔═══════════════════════════════════════════════════════════════════════════╗
# ║ CELDA 1 · CONFIGURACIÓN Y FUNCIONES                                       ║
# ║ Ejecutar UNA SOLA VEZ al iniciar el notebook.                             ║
# ║ Define constantes, instala dependencias y declara todas                   ║
# ║ las funciones. No produce salidas de datos.                               ║
# ╚═══════════════════════════════════════════════════════════════════════════╝

# ── Dependencias ──────────────────────────────────────────────────────────

import os, warnings
import pandas as pd
import numpy as np
from google.colab import files

warnings.filterwarnings('ignore')

try:
import openpyxl
except ImportError:
import subprocess
subprocess.run(['pip', 'install', 'openpyxl', '-q'], check=True)
import openpyxl
```

```
# ═══════════════════════════════════════════════════════════════════════
# 1.1 REGLAS DE NEGOCIO — editar aquí y solo aquí
# ═══════════════════════════════════════════════════════════════════════

SUCURSALES = {
'MALL': 'MALL DE LOS ANDES',
'MATRIZ': 'MATRIZ',
'RIOBAMBA': 'RIOBAMBA',
'SHOPPING': 'SHOPPING',
'SUCRE': 'SUCRE',
}

# Regla crítica: columna 8 física del Excel = índice 7 en pandas = Venta_Total
COLUMNAS = {
1: 'Codigo_Producto',
2: 'Fecha_Venta',
5: 'Num_Factura',
7: 'Venta_Total',
9: 'Cantidad',
10: 'Costo_Unitario',
13: 'Descripcion',
}

COLUMNAS_FINALES = [
'Fecha_Venta', 'Num_Factura', 'Codigo_Base', 'Talla_Extraida',
'Descripcion', 'Cantidad', 'Costo_Unitario',
'Precio_Venta_Unit', 'Venta_Total', 'Utilidad_Bruta', 'Sucursal',
]

# Rutas de checkpoint — archivos Parquet intermedios en /content/
CKPT_CRUDO = '/content/ckpt_01_crudo.parquet'
CKPT_TRANSFORMADO = '/content/ckpt_02_transformado.parquet'

# ═══════════════════════════════════════════════════════════════════════
```

```
# 1.2 FUNCIONES DE UTILIDAD
# ═══════════════════════════════════════════════════════════════════════

def detectar_sucursal(nombre_archivo: str) -> str:
”Detecta la sucursal desde el nombre del archivo.”
nombre_upper = nombre_archivo.upper()
for clave, valor in SUCURSALES.items():
if clave in nombre_upper:
return valor
return 'DESCONOCIDA'


def procesar_producto_seguro(row) -> pd.Series:
”
Extrae Codigo_Base y Talla_Extraida desde Codigo_Producto.

Reglas de negocio:
- Separador: último punto → rsplit('.', 1)
- Talla > 3 caracteres → S/T (es ruido, no talla)
- Sin punto / nulo / vacío → S/T
”
codigo_raw = row['Codigo_Producto']

if pd.isna(codigo_raw):
return pd.Series(['S/C', 'S/T'])

codigo = str(codigo_raw).strip()
if not codigo or codigo.lower() == 'nan':
return pd.Series(['S/C', 'S/T'])

if '.' not in codigo:
return pd.Series([codigo, 'S/T'])

base, talla = codigo.rsplit('.', 1)
talla = talla.upper()

return pd.Series([codigo, 'S/T']) if len(talla) > 3 else pd.Series([base, talla])
```

```
def reporte_calidad(df: pd.DataFrame, nombre: str = ”DataFrame”) -> None:
Ïmprime métricas de calidad del DataFrame. Documenta el estado post-wrangling.”
sep = '=' * 55
print(f”\n{sep}”)
print(fREPORTE DE CALIDAD: {nombre}”)
print(sep)
print(fRegistros totales : {len(df):>10,}”)

if 'Talla_Extraida' in df.columns:
n_st = (df['Talla_Extraida'] == 'S/T').sum()
pct = n_st / len(df) * 100 if len(df) else 0
print(fRegistros S/T : {n_st:>10,} ({pct:.1f}%)”)

for col, label in [('Venta_Total', 'Venta_Total = 0 '),
('Venta_Total', 'Venta_Total total ')]:
if col in df.columns:
if 'cero' in label:
print(f”{label} : {(df[col] == 0).sum():>10,}”)
else:
print(f”{label} : {df[col].sum():>10,.2f}”)
break

if 'Venta_Total' in df.columns:
print(f”Venta_Total = 0 : {(df['Venta_Total'] == 0).sum():>10,}”)
print(f”Venta_Total total : {df['Venta_Total'].sum():>10,.2f}”)

if 'Utilidad_Bruta' in df.columns:
print(fÜtilidad negativa : {(df['Utilidad_Bruta'] < 0).sum():>10,}”)
print(fÜtilidad total : {df['Utilidad_Bruta'].sum():>10,.2f}”)

if 'Fecha_Venta' in df.columns:
fechas = df['Fecha_Venta'].dropna()
if len(fechas):
print(fRango de fechas : {fechas.min()} → {fechas.max()}”)

if 'Sucursal' in df.columns:
```

```
print(f”Sucursales : {df['Sucursal'].unique().tolist()}”)

print(f”{sep}\n”)


print(ÇELDA 1 lista — configuración y funciones cargadas.”)
```

Celda 2

Extracción. Lee en crudo los archivos Excel/CSV de cada sucursal (sin transformar), añade la columna de trazabilidad _archivo_origen y guarda el checkpoint intermedio ckpt_01_crudo.parquet. Es re-ejecutable: si el checkpoint ya existe, lo carga directamente en lugar de solicitar los archivos de nuevo.

```
# ╔═══════════════════════════════════════════════════════════════════════════╗
# ║ CELDA 2 · EXTRACCIÓN ║
# ║ Sube uno o varios archivos Excel/CSV, los lee en crudo ║
# ║ (sin transformar) y guarda un checkpoint Parquet. ║
# ║ ║
# ║ CHECKPOINT → ckpt_01_crudo.parquet ║
# ║ Re-ejecutable: vuelve a pedir archivos si el checkpoint ║
# ║ no existe, o lo carga directamente si ya existe. ║
# ╚═══════════════════════════════════════════════════════════════════════════╝


def _leer_archivo_crudo(nombre_archivo: str) -> pd.DataFrame:
”
Lee un archivo Excel o CSV en crudo (header=None) y agrega
la columna _archivo_origen para trazabilidad.
No aplica ninguna transformación de negocio.
”
ext = os.path.splitext(nombre_archivo)[1].lower()
try:
if ext in ('.xlsx', '.xls'):
df = pd.read_excel(nombre_archivo, header=None, engine='openpyxl')
elif ext == '.csv':
df = pd.read_csv(nombre_archivo, header=None, encoding='utf-8')
else:
raise ValueError(f”Formato no soportado: {ext}”)
except Exception as exc:
```

```
print(f.Error al leer '{nombre_archivo}': {exc}")
return pd.DataFrame()

df['_archivo_origen'] = nombre_archivo # trazabilidad de origen
print(f"{nombre_archivo} → {len(df):,} filas crudas")
return df


# ── Lógica de checkpoint ──────────────────────────────────────────────
──────────
if os.path.exists(CKPT_CRUDO):
print(fÇheckpoint encontrado: '{CKPT_CRUDO}'")
print(Çargando datos crudos desde checkpoint (omitiendo re-subida)...")
df_crudo = pd.read_parquet(CKPT_CRUDO)
print(f"{len(df_crudo):,} filas · "
f"{df_crudo['_archivo_origen'].nunique()} archivo(s) en checkpoint")
print("\n Para procesar archivos nuevos, elimina el checkpoint con:")
print(fïmport os; os.remove('{CKPT_CRUDO}')")
else:
print(.EXTRACCIÓN — sube los archivos de ventas (Excel o CSV):")
print("Puedes seleccionar múltiples archivos a la vez.\n")

uploaded = files.upload()

if not uploaded:
raise RuntimeError("No se subió ningún archivo. Re-ejecuta esta celda.")

frames_crudos = []
for nombre in uploaded.keys():
print(f"\n── {nombre}")
df_temp = _leer_archivo_crudo(nombre)
if not df_temp.empty:
frames_crudos.append(df_temp)

if not frames_crudos:
raise RuntimeError("Ningún archivo produjo datos. Revisa los archivos subidos.")

# Concatenar todos en un único DataFrame crudo
```

```
df_crudo = pd.concat(frames_crudos, ignore_index=True)

# ── Guardar checkpoint ──────────────────────────────────────────────
# Convertir columnas numéricas a string para compatibilidad Parquet
df_ckpt = df_crudo.copy()
# Convert any object columns to string to avoid PyArrow type inference issues
for col in df_ckpt.select_dtypes(include='object').columns:
df_ckpt[col] = df_ckpt[col].astype(str)
df_ckpt.columns = df_ckpt.columns.astype(str)
df_ckpt.to_parquet(CKPT_CRUDO, index=False)
print(f"\n Checkpoint guardado: '{CKPT_CRUDO}'")

print(f"\n CELDA 2 lista — {len(df_crudo):,} filas crudas disponibles en df_crudo")
```

Celda 3

Transformación (Data Wrangling). Núcleo técnico del framework. La función _ transformar() ejecuta las siete subetapas T1-T7: renombrado de columnas, validación de columnas críticas, extracción de talla, tipado numérico con coerción segura, cálculo vectorizado de Precio_Venta_Unit y Utilidad_Bruta mediante np.where(), tipado de fecha y enriquecimiento de Sucursal, y filtrado final. El resultado se guarda en el checkpoint ckpt_02_transformado.parquet.

```
# ╔═══════════════════════════════════════════════════════════════════╗
# ║ CELDA 3 · TRANSFORMACIÓN (DATA WRANGLING) ║
# ║ Lee desde el checkpoint crudo, aplica todas las reglas de ║
# ║ negocio y guarda un segundo checkpoint transformado. ║
# ║ ║
# ║ ENTRADA → ckpt_01_crudo.parquet (o df_crudo en memoria) ║
# ║ CHECKPOINT → ckpt_02_transformado.parquet ║
# ╚═══════════════════════════════════════════════════════════════════╝

def _transformar(df_raw: pd.DataFrame) -> pd.DataFrame:
"
Aplica el pipeline de transformación completo a un DataFrame crudo.

Etapas
──────
```

```
T1 Renombrado de columnas (reglas de negocio)
T2 Validación de columnas críticas
T3 Extracción de talla desde Codigo_Producto
T4 Tipado numérico con coerción segura
T5 Cálculo vectorizado de métricas (precio unitario, utilidad)
T6 Tipado de fecha y enriquecimiento de Sucursal
T7 Filtrado de filas inválidas y selección de columnas finales
”
df = df_raw.copy()

# ── T1: Renombrado ──────────────────────────────────────────────────
# Las columnas crudas son enteros (0, 1, 2...); renombramos solo
# las que existan para tolerar archivos con distinto número de cols.
col_map = {k: v for k, v in COLUMNAS.items() if k in df.columns}
df = df.rename(columns=col_map)

# ── T2: Validación de columnas críticas ─────────────────────────────
criticas = ['Codigo_Producto', 'Venta_Total', 'Num_Factura']
faltantes = [c for c in criticas if c not in df.columns]
if faltantes:
print(fÇolumnas críticas ausentes: {faltantes}”)
print(fÇolumnas disponibles: {list(df.columns)}”)

# ── T3: Extracción de talla (regla rsplit + validación longitud) ─
if 'Codigo_Producto' in df.columns:
df[['Codigo_Base', 'Talla_Extraida']] = df.apply(
procesar_producto_seguro, axis=1
)
else:
df['Codigo_Base'] = 'S/C'
df['Talla_Extraida'] = 'S/T'

# ── T4: Tipado numérico ─────────────────────────────────────────────
for col in ('Venta_Total', 'Cantidad', 'Costo_Unitario'):
df[col] = pd.to_numeric(df.get(col, 0), errors='coerce').fillna(0)
```

```
# ── T5: Métricas vectorizadas ──────────────────────────────────────
# np.where evita la división por cero sin un apply fila por fila
df['Precio_Venta_Unit'] = np.where(
df['Cantidad'] > 0,
df['Venta_Total'] / df['Cantidad'],
0
)
df['Utilidad_Bruta'] = df['Venta_Total'] - (df['Costo_Unitario'] * df['Cantidad'])

# ── T6: Fecha + Sucursal ───────────────────────────────────────────
if 'Fecha_Venta' in df.columns:
df['Fecha_Venta'] = pd.to_datetime(
df['Fecha_Venta'], errors='coerce'
).dt.date

if '_archivo_origen' in df.columns:
df['Sucursal'] = df['_archivo_origen'].apply(detectar_sucursal)
df = df.drop(columns=['_archivo_origen']) # columna auxiliar, ya no necesaria
else:
df['Sucursal'] = 'DESCONOCIDA'

if 'Descripcion' not in df.columns:
df['Descripcion'] = ''

# ── T7: Filtrado y selección final ─────────────────────────────────
cols_ok = [c for c in COLUMNAS_FINALES if c in df.columns]
df_final = df[cols_ok].copy()

if 'Num_Factura' in df_final.columns:
df_final = df_final.dropna(subset=['Num_Factura'])
df_final = df_final[
df_final['Num_Factura'].astype(str).str.strip() != ''
]

if 'Fecha_Venta' in df_final.columns:
df_final = df_final.sort_values('Fecha_Venta', na_position='last')
```

```
return df_final.reset_index(drop=True)


# ── Lógica de checkpoint ──────────────────────────────────────────────
if os.path.exists(CKPT_TRANSFORMADO):
print(fÇheckpoint encontrado: '{CKPT_TRANSFORMADO}'")
print(Çargando datos transformados (omitiendo re-transformación)...")
df_transformado = pd.read_parquet(CKPT_TRANSFORMADO)
print(f"{len(df_transformado):,} registros limpios cargados")
print("\n Para re-transformar desde cero, elimina el checkpoint con:")
print(fïmport os; os.remove('{CKPT_TRANSFORMADO}')")
else:
# Leer desde checkpoint crudo si df_crudo no está en memoria
if 'df_crudo' not in dir() or df_crudo is None:
if not os.path.exists(CKPT_CRUDO):
raise RuntimeError(
"No hay datos crudos disponibles. Ejecuta primero la Celda 2."
)
print(fÇargando datos crudos desde '{CKPT_CRUDO}'...")
df_crudo_raw = pd.read_parquet(CKPT_CRUDO)
# Restaurar columnas enteras (Parquet las convierte a string)
int_cols = {}
for c in df_crudo_raw.columns:
try:
int_cols[c] = int(c)
except (ValueError, TypeError):
int_cols[c] = c
df_crudo_raw = df_crudo_raw.rename(columns=int_cols)
else:
df_crudo_raw = df_crudo.copy()
# Asegurarse de que las columnas sean enteros donde corresponda
int_cols = {}
for c in df_crudo_raw.columns:
try:
int_cols[c] = int(c)
except (ValueError, TypeError):
```

```
int_cols[c] = c
df_crudo_raw = df_crudo_raw.rename(columns=int_cols)

print(”TRANSFORMACIÓN — aplicando pipeline de Data Wrangling...\n”)

# Procesar archivo por archivo para reportar progreso individual
frames_limpios = []
origenes = df_crudo_raw['_archivo_origen'].unique() \
if '_archivo_origen' in df_crudo_raw.columns \
else ['(desconocido)']

for origen in origenes:
print(f”── {origen}”)
if '_archivo_origen' in df_crudo_raw.columns:
bloque = df_crudo_raw[df_crudo_raw['_archivo_origen'] == origen].copy()
else:
bloque = df_crudo_raw.copy()

bloque_limpio = _transformar(bloque)
frames_limpios.append(bloque_limpio)
print(f”{len(bloque):,} filas crudas → {len(bloque_limpio):,} registros limpios”)
reporte_calidad(bloque_limpio, origen)

df_transformado = pd.concat(frames_limpios, ignore_index=True)

# ── Guardar checkpoint ──────────────────────────────────────────────
df_transformado.to_parquet(CKPT_TRANSFORMADO, index=False)
print(fÇheckpoint guardado: '{CKPT_TRANSFORMADO}'”)

print(f”\n CELDA 3 lista — {len(df_transformado):,} registros transformados en df_
transformado”)
```

Celda 4

Carga. Consolida todas las sucursales desde el checkpoint transformado, genera el reporte de calidad global y el resumen agregado por sucursal (registros, venta total, utilidad, registros sin talla, margen porcentual), y exporta el archivo Excel consolidado.

```
# ╔═══════════════════════════════════════════════════════════════════════╗
# ║ CELDA 4 · CARGA — CONSOLIDACIÓN, REPORTE FINAL Y DESCARGA ║
# ║ Lee desde el checkpoint transformado, consolida todas las ║
# ║ sucursales, genera el reporte de calidad global y descarga ║
# ║ el Excel final. ║
# ║ ║
# ║ ENTRADA → ckpt_02_transformado.parquet (o df_transformado) ║
# ║ SALIDA → VENTAS_CONSOLIDADAS_RETAIL.xlsx (descarga auto.) ║
# ╚═══════════════════════════════════════════════════════════════════════╝

# ── Cargar datos: memoria primero, checkpoint como fallback ──────────
if 'df_transformado' not in dir() or df_transformado is None:
if not os.path.exists(CKPT_TRANSFORMADO):
raise RuntimeError(
”No hay datos transformados disponibles. Ejecuta primero la Celda 3.”
)
print(fÇargando desde checkpoint '{CKPT_TRANSFORMADO}'...”)
df_consolidado = pd.read_parquet(CKPT_TRANSFORMADO)
else:
df_consolidado = df_transformado.copy()

# ── Ordenar consolidado por fecha ────────────────────────────────────
if 'Fecha_Venta' in df_consolidado.columns:
df_consolidado = df_consolidado.sort_values(
'Fecha_Venta', na_position='last'
).reset_index(drop=True)

# ── Reporte de calidad global ────────────────────────────────────────
print(-”* 55)
print(ÇONSOLIDADO FINAL — TODAS LAS SUCURSALES”)
print(-”* 55)
reporte_calidad(df_consolidado, Çonsolidado”)

# ── Resumen por sucursal ─────────────────────────────────────────────
```

```
————————————
if 'Sucursal' in df_consolidado.columns and 'Venta_Total' in df_consolidado.columns:
print(RESUMEN POR SUCURSAL”)
print(”* 55)
resumen = (
df_consolidado
.groupby('Sucursal', sort=False)
.agg(
Registros =('Num_Factura', 'count'),
Venta_Total =('Venta_Total', 'sum'),
Utilidad =('Utilidad_Bruta','sum'),
Sin_Talla =('Talla_Extraida', lambda x: (x == 'S/T').sum()),
)
.assign(Margen_Pct=lambda d: (d['Utilidad'] / d['Venta_Total'] * 100).round(1))
.sort_values('Venta_Total', ascending=False)
)
print(resumen.to_string())
print()

# ── Guardar y descargar Excel final ──────────────────────────────────────
ARCHIVO_SALIDA = 'VENTAS_CONSOLIDADAS_RETAIL.xlsx'

with pd.ExcelWriter(ARCHIVO_SALIDA, engine='openpyxl') as writer:
# Hoja principal con todos los datos
df_consolidado.to_excel(writer, sheet_name='Ventas_Consolidadas', index=False)

# Hoja de resumen por sucursal (si existe)
if 'resumen' in dir():
resumen.to_excel(writer, sheet_name='Resumen_Sucursales')

print(f.Archivo generado: '{ARCHIVO_SALIDA}'”)
print(İniciando descarga...\n”)
files.download(ARCHIVO_SALIDA)

# ── Vista previa ─────────────────────────────────────────────────────────
————————————
print(”Vista previa — primeros 10 registros:”)
```

```
display(df_consolidado.head(10))

# ── Estado final de checkpoints ──────────────────────────────────────
print("\n Checkpoints disponibles en /content/:")
for ckpt in (CKPT_CRUDO, CKPT_TRANSFORMADO):
if os.path.exists(ckpt):
size_kb = os.path.getsize(ckpt) / 1024
print(f"{ckpt} ({size_kb:.1f} KB)")

print(f"\n CELDA 4 lista — pipeline completo.")
print(f"{len(df_consolidado):,} registros exportados a '{ARCHIVO_SALIDA}'")
```

Celda 5

Consolidación maestro y limpieza para Power BI. Aplica las correcciones de calidad sobre el consolidado anual completo: eliminación de duplicados exactos (df.drop_duplicates()), conversión de Costo_Unitario = 0 a NaN, generación de la columna Año_Mes para los ejes de tiempo del modelo dimensional, y el flag binario Venta_Bajo_Costo. Genera el archivo DATASET_MAESTRO_VENTAS_22_25.xlsx que alimenta la capa de Power BI.

```
# ╔═══════════════════════════════════════════════════════════════════╗
# ║ CELDA 5 · CONSOLIDACIÓN MAESTRO + LIMPIEZA PARA POWER BI ║
# ║ Sube los archivos anuales ya procesados por el pipeline ETL. ║
# ║ ║
# ║ Aplica 4 correcciones de calidad detectadas en auditoría: ║
# ║ L1 · Eliminar filas 100 % duplicadas (718 detectadas) ║
# ║ L2 · Costo_Unitario = 0 → NaN (8 registros sin costo real) ║
# ║ L3 · Añadir columna Año_Mes para ejes de tiempo en Power BI ║
# ║ L4 · Flag 'Venta_Bajo_Costo' para filtrar ventas a pérdida ║
# ║ ║
# ║ SALIDA → DATASET_MAESTRO_VENTAS_22_25.xlsx ║
# ╚═══════════════════════════════════════════════════════════════════╝

print("Sube los archivos anuales (VENTAS_CONSOLIDADAS_RETAIL...xlsx)")
print("Puedes seleccionar los 4 a la vez.\n")
uploaded = files.upload()
```

```
frames = []
for nombre_archivo in uploaded.keys():
print(f”Procesando: {nombre_archivo}”)
try:
df = pd.read_excel(nombre_archivo)
except Exception:
try:
df = pd.read_csv(nombre_archivo, encoding='latin1')
except UnicodeDecodeError:
df = pd.read_csv(nombre_archivo, encoding='cp1252')
except Exception as e_csv:
print(f.Error al leer {nombre_archivo}: {e_csv}”)
continue

if 'Fecha_Venta' in df.columns:
df['Fecha_Venta'] = pd.to_datetime(df['Fecha_Venta']).dt.date

frames.append(df)
print(f”{len(df):,} registros cargados”)

if not frames:
raise RuntimeError(”No se subieron archivos válidos.”)

df_maestro = pd.concat(frames, ignore_index=True)
print(f”\n Total registros antes de limpieza : {len(df_maestro):,}”)

# ═══════════════════════════════════════════════════════════════
═══════════════
# L1 · ELIMINAR DUPLICADOS EXACTOS
# ═══════════════════════════════════════════════════════════════
═══════════════
antes = len(df_maestro)
df_maestro = df_maestro.drop_duplicates()
eliminados = antes - len(df_maestro)
print(f”🗑 L1 Duplicados eliminados : {eliminados:,}”)

# ═══════════════════════════════════════════════════════════════
═══════════════
```

```
# L2 · COSTO_UNITARIO = 0 → NaN
# Registros sin costo registrado no deben distorsionar el margen.
# Power BI excluye NaN de promedios y porcentajes automáticamente.
# ═══════════════════════════════════════════════════════════════
if 'Costo_Unitario' in df_maestro.columns:
cero_costo = (df_maestro['Costo_Unitario'] == 0).sum()
df_maestro['Costo_Unitario'] = df_maestro['Costo_Unitario'].replace(0, np.nan)
print(f"L2 Costo_Unitario=0 → NaN : {cero_costo:,} registros")

# ═══════════════════════════════════════════════════════════════
# L3 · COLUMNA Año_Mes PARA EJES DE TIEMPO EN POWER BI
# Formato YYYY-MM garantiza orden cronológico correcto en gráficos.
# ═══════════════════════════════════════════════════════════════
if 'Fecha_Venta' in df_maestro.columns:
fechas_dt = pd.to_datetime(df_maestro['Fecha_Venta'])
df_maestro['Año_Mes'] = (
fechas_dt.dt.year.astype(str) + '-' +
fechas_dt.dt.month.astype(str).str.zfill(2)
)
# Insertar Año_Mes justo después de Fecha_Venta

cols = list(df_maestro.columns)
idx = cols.index('Fecha_Venta') + 1
cols.insert(idx, cols.pop(cols.index('Año_Mes')))
df_maestro = df_maestro[cols]
print(f"L3 Columna Año_Mes añadida : {df_maestro['Año_Mes'].nunique()} meses
únicos")
```

```
# ═══════════════════════════════════════════════════════════════════════
# L4 · FLAG Venta_Bajo_Costo
# Identifica ventas donde Precio_Venta_Unit < Costo_Unitario.
# Permite filtrar liquidaciones en los KPIs de margen de Power BI.
# ═══════════════════════════════════════════════════════════════════════
if 'Precio_Venta_Unit' in df_maestro.columns and 'Costo_Unitario' in df_maestro.columns:
df_maestro['Venta_Bajo_Costo'] = (
df_maestro['Precio_Venta_Unit'] < df_maestro['Costo_Unitario']
).map({True: 'Sí', False: 'No'})
bajo_costo = (df_maestro['Venta_Bajo_Costo'] == 'Sí').sum()
print(f"🏷 L4 Flag Venta_Bajo_Costo : {bajo_costo:,} registros marcados como 'Sí'")

# ── Ordenar por fecha ──────────────────────────────────────────────────
if 'Fecha_Venta' in df_maestro.columns:
df_maestro = df_maestro.sort_values('Fecha_Venta').reset_index(drop=True)

print(f"\n Total registros finales : {len(df_maestro):,}")
print(fÇolumnas : {list(df_maestro.columns)}")

# ── Exportar ──────────────────────────────────────────────────────────
NOMBRE_SALIDA = 'DATASET_MAESTRO_VENTAS_22_25.xlsx'
df_maestro.to_excel(NOMBRE_SALIDA, index=False)
print(f"\n Archivo generado: '{NOMBRE_SALIDA}'")
print(Ïniciando descarga...")
files.download(NOMBRE_SALIDA)
```